Authentification Robuste par Mot de Passe Basée sur une Analyse Probabiliste de Risque d'Attaque par Déni de Service Dans les Réseaux de Capteurs Sans Fil

Youssou Faye* — Ibrahima Niang** — Hervé Guyennet*

* Département d'informatique
Université de Franche Comté
Besançon Cedex
FRANCE
yfaye@femto-st.fr, herve.guyennet@femto-st.fr

** Département Mathématique-Informatique
Université Cheikh Anta Diop
Dakar
SENEGAL
iniang@ucad.sn

RÉSUMÉ. Pour plusieurs types d'applications des Réseaux de Capteurs Sans Fil (RCSFs), produire une variété de fonctions de sécurité avec une faible consommation de ressources devient un vrai défi. Récemment, Vaidya et al. ont proposé un protocole d'authentification robuste pour les RCSFs qui permet aux utilisateurs légitimes d'envoyer leurs requêtes à tout capteur. Dans ce papier, nous démontrons les vulnérabilités lors d'attaque par déni de service (DoS) et par falsification sur le protocole de Vaidya et al.. Nous proposons une solution efficace dans l'environnement des RCSFs, qui conserve tous les avantages de Vaidya et al., et améliore sa sécurité. Nous utilisons le modèle de l'attaquant pour déterminer la probabilité de risque à partir de laquelle notre solution est énergétiquement meilleure.

ABSTRACT. For various Wireless Sensor Networks (WSNs) applications, providing a variety of security functions with limited energy resources and low power capabilities is a very challenge. Recently, Vaidya et al. proposed an authentication protocol which allows legitimate users to query sensor data at every sensor node of the network. In this work, we show that, Vaidya et al.'s scheme suffers from the risk of forgery and Denial of- Service (DoS) attacks. To cope with them, we propose a new solution which is quite adequate for power and resource constrained WSN. Our scheme not only retains all the advantages in Vaidya and al.'s scheme but also enhances its security. We use in our implementation the probability risk analysis owing to the DoS attack model to show to which level schemes justify better energy.

MOTS-CLÉS : Réseaux de Capteurs Sans Fil, contrôle d'accès, authentification, mot de passe.

KEYWORDS : Wireless Sensor Networks, access control, authentication, password.

1. Introduction

Le RCSF est souvent destiné au contrôle d'espaces géographiquement limités. En général, les données récoltées par les capteurs comme par exemple la température ne sont pas confidentielles. Dans certaines applications, les requêtes sont envoyées à une station de base ou à la passerelle du réseau. Cependant, pour les applications temps réel ou critiques, comme par exemple les applications militaires, les données critiques doivent être protégées contre toute utilisation frauduleuse, et accessibles en temps réel non seulement depuis la station de base ou la passerelle du réseau, mais parfois aussi depuis n'importe où dans le réseau à travers les capteurs en mode ad hoc. Dans ce contexte, le contrôle d'accès au réseau devient nécessaire [1]. L'authentificaion des utilisateurs est la solution la plus utilisée dans les réseaux traditionnels. Cependant, dans les RCSFs, elle reste moins étudiée à cause des contraintes énergétiques, de mémorisation, de calcul et de transmission.

Vaidya et al.[2], une des dernières solutions basée sur un mot de passe fort est divisé en quatre phases : une phase d'enregistrement, une phase de login, une phase d'authentification et une phase de changement de mot de passe. Dans cet article, nous montrons que, la solution de Vaidya et al., de par son manque de vérification du mot de passe lors de la phase de login, est vulnérable lors d'attaque par DoS. De même, elle manque de protection contre la falsification des estampilles temporelles. Nous proposons une solution qui maintient tous ses avantages et améliore sa sécurité par l'intégration d'un mécanisme de vérification de mot de passe et de chiffrement des estampilles basés sur les fonctions de hachage et l'opérateur OU-exclusif

Le reste du papier est organisé comme suit. La section 2 présente l'état de l'art. Une description de la solution de Vaidya et al. [2] est fournie à la section 3. Une analyse de leur solution et une présentation de celle proposée sont respectivement décrites dans les sections 4 et 5. La sécurité du nouveau protocole est analysée à la section 6. La section 7 décrit une implémentation de notre solution et celle de Vaidya et al.[2] avec une analyse basée sur la probabilité de risque. Enfin, nous concluons le papier et identifions des perspectives de recherche à la section 8.

2. Etat de l'art

Le contrôle d'accès a toujours été un problème classique dans beaucoup d'applications et systèmes informatiques existants. L'authentification à distance des utilisateurs a été depuis longtemps la solution de base la plus utilisée. Initialement, les solutions d'authentification d'utilisateurs [3,4,5,6,7] proposées dans l'environnement des cartes à puce étaient inspirées de Lamport(1981) [8], à la différence qu'aucune table de vérification n'était pas stockée dans le système distant pour la validité du login de l'utilisateur. Ces solutions utilisent une approche par mot de passe avec un login statique. Certaines d'entre elles [4] utilisent la techniques d'un mot de passe faible. Elles présentent l'avantage d'une mémorisation facile du mot de passe. Cependant la cryptographie à clé publique utilisée reste le principal inconvénient pour une application dans l'environnement des capteurs. Par contre, d'autres solutions [3,5,6,7] utilisent la technique d'un mot de passe fort et sont basées uniquement sur les fonctions de hachage et l'opérateur OU-exclusif, ce qui facilite leur implémentation dans les RCSFs. Leur inconvénient réside dans la difficulté de mémorisation du mot de passe. Des solutions comme [9] implantent la technique du mot de passe fort avec un login dynamique afin de se protéger contre l'usurpation de lo-

gin. Ce qui permet le libre changement de login et de mot de passe. Cependant, Lee et al.[10] ont montré que ces protocoles basés sur un login dynamique sont vulnérables aux attaques telles que la répétition, la contrefaçon de login et la fabrication. Leur solution proposée pour les cartes à puce utilise la même technique des fonctions de hachage et du OU-exclusif. Pour une adaptation dans les RCSFs, Wong et al. [11] proposent une solution basée sur Lee et al.[10] moins coûteuse en calcul. La solution de Tseng et al. [13] montre les insufisances de Wong et al. [11] et améliore sa sécurité. C'est ainsi que Vaidya et al., dans [2], proposent une version plus robuste de [12] basée sur Wong et al. [11].

3. Passage en revue du protocole de Vaidya et al.

Dans les prochaines sections, nous allons décrire les trois premières phases de la solution de Vaidya et al. [2] en utilisant les notations du Tableau 1. Seule la phase de changement de mot de passe n'est pas décrite car elle rest inchangée dans les deux solutions.

Tableau 1. *Notations*

Symboles	Description
UD	dispositif de l'utilisateur (PDA, PC etc..)
GW	noeud passerelle
LN	un Noeud capteur Login
H( )	une fonction de hachage à sens unique
$\oplus$	l'opérateur OU-exclusif
$\|$	l'opérateur de concatenation
Succ_Reg	message d'enregistrement avec succès
Acc_login	message d'accepation du login
Succ_Change	message de changement avec succès
x	la clé de la passerelle
UID	l'identité de l'utilisateur
PW	mot de passe choisi par l'utilisateur
TS	temps d'enregistrement d'un noeud
t , T_0,T_1,T_2,T_3	temps actuels enregistrés par un des noeuds
$\triangle$T	respectivement délai de transmission
$\longrightarrow$	transmission de message

Phase d'Enregistrement (PE)

PE1-L'utilisateur (UD) au voisinage de la passerelle (GW) choisit librement un mot de passe PW et calcule le haché vpw = H(PW).

PE2-Au temps TS, UD envoie son identité UID et vpw à la GW en mode sécurisé.

PE3-La GW calcule X = H(UID$\|$x), stocke (UID, vpw, X, TS), puis répond à UD que l' enregistrement est effectué avec succès (Succ_Reg (X)), et distribue ensuite les paramètres (UID, X, TS) aux capteurs de login (LN) capables de fournir des interfaces à l'UD pour se logger.

Phase de Login (PL)

PL1-L'UD au voisinage d'un LN calcule A = H(vpw$\|$t) et soumet (UID, A, t) au LN.

PL2- Après réception de la requête à T_0, le LN vérifie la validité du UID et si T_0 - t$\geq\triangle$T.

PL3-Si les conditions sont vérifiées,le LN récupère le paramètre A correspondant et calcule C_K = (X $\oplus$A $\oplus$ T_0), puis il envoie (UID, C_K, T_0, t) à la GW.

Phase d'Authentification (PA)

PA1- A la réception du message de login, la GW vérifie la validité de UID et t. S'ils sont valides, alors la GW vérifie si T_1-T_0 $\geq$ $\triangle$T et T_0-t $\geq$ $\triangle$T afin d'éviter la répétition de la requête de login. S'ils sont valides, la GW récupère le vpw et le paramètre A correspondant, puis calcule A'= H(vpw$\|$ t) et C_K' = (X $\oplus$ A'$\oplus$ T_0). Si $C_K \neq C_K$', le message de

login est rejeté, sinon, la GW calcule V_M = H(X$\|$A'$\|$ T_1).
PA2- La GW envoie un message d'acceptation (Acc_login, V_M, T_1) au LN.
PA3- Le LN calcule V'_M = H(X$\|$A$\|$ T_1), et si V_M =V'_M , il calcule Y_K = H($V'_M\|$ T_2).
PA4- Le LN envoie (Acc_login, Y_K, T_1, T_2) à l'UD.
PA5-Après réception du message au temps T_3,l'UD vérifie si T_1-$T_0 \geq \triangle$T et T_0-t $\geq \triangle$T. Si ces conditions vérifiées, l'UD calcule $V"_M$ =H(X$\|$A$\|T_1$) et Y'_K = H($V"_M\|T_2$), si Y_K =Y'_K, l'UD commence à obtenir les données, sinon le message d'acceptation de login (Acc_login) est rejeté.

La Figure 1(a) montre le modèle de communication dans Vaidya et al.[2].

4. Vulnérabilités du protocole de Vaidya et al.

Nous allons démontrer que Vaidya et al.[2] est vulnérable lors d'attaque par DoS et par falsification d'estampilles temporaires. Durant la phase de login, seuls le UID et le temps t sont vérifiés par le LN, le DoS peut survenir de deux façons. Premièrement l'intrus peut intercepter ou écouter un UID valide puis le soumet avec un faux mot de passe. Puisque le LN vérifie seulement le UID, il va retransmettre ce message via les capteurs à la GW située à plusieurs sauts. Deuxièment, ce scénario peut arriver autrement. Généralement, les mots de passe sont masqués lors de leur saisie, si un UD se trompe de saisie de mot de passe, le même effet se reproduit. La transmission étant généralement l'opération la plus coûteuse en énergie, la propagation de fausses requêtes doit être évitée.

Vaidya et al.[2] ont supposé qu'un attaquant ayant capturé un LN, obtient UID, X, TS, et qu'il lui est aussi possible d'écouter le message (UID, A, t) afin de démontrer une attaque par falsification sur Wong et al. [11]. Partant de ces suppositions, leur protocole, de façon différente, sera vulnérable à l'attaque par falsification des temps transmis. Puisque seul le délai de transmission est vérifié, l'attaquant peut créer deux faux temps T'_0 et t' dont la différence respecte le délai de propagation en y ajoutant un petit nombre ξ partout sur les deux temps T_0 et t déjà transmis en clair dans le réseau. Ainsi il effectue T'_0=T_0+ξ ,t'=t+ξ, ensuite calcule C'_K = H(X$\oplus$A$\oplus T'_0$) puis envoie le message (UID, $C'_K T'_0$,t'. Puisque (T_1 - T'_0)$\leq\triangle$T et (T'_0-t')$\leq\triangle$T, le message passe.

5. Authentification Robuste Basée sur la Probabilité d'Attaque

Dans cette section, nous proposons un protocole d'Authentification Robuste Basée sur la Probabilité d'Attaque (ARBPA) afin de résoudre les faiblesses notées dans Vaidya et al.[2]. Notre solution comporte les mêmes phases que Vaidya et al.

Phase d'Enregistrement (PE)

PE1-Dans Vaidya et al., l'UD choisit un mot de passe PW, puis calcule vpw=H(PW) et envoie vpw avec son identité pour se loguer. Et pour le reste du protocole, le PW n'est plus utilisé. Il n'est pas nécessaire de calculer vpw, qui est autant vulnérable que PW. C'est ainsi que dans ARBPA, l' UD choisit librement son mot de passe PW.
PE2-Au temps TS, l'UD soumet son UID avec PW à la GW en mode sécurisé.
PE3-La GW calcule X = H(UID$\|$x) puis répond à l'UD le message Succ_Reg(X). Elle stocke les paramètres (UID, H(PW), X, TS), et distribue (UID, X, H(PW),TS) aux LNs.

Phase de Login (PL)

PL1- L'UD calcule A= H(H(PW)$\|$t) et soumet (UID, A, t) au LN.
PL2-Après réception de la requête au temps T_0, le LN vérifie : si l'identité UID est non

valide ou A $\neq$ H(H(PW)||t), ou T_0 - t $\geq\triangle$, alors le message de login est rejeté.
PL3-Le LN calcule C_K = (X$\oplus$A$\oplus T_0$) et t'=H^2(PW)$\oplus$t, et envoie (UID,C_K,T_0,t') à GW.

Phase d'authentification

PA1-La GW vérifie si le UID et le temps t sont valides, puis elle calcule t=t'$\oplus H^2$(PW), ensuite vérifie si T_1 - $T_0\geq\triangle$T et T_0 - t$\geq\triangle$T. Si ces connditions sont valides,la GW récupère H(PW) et le paramètre A puis calcule A' =H(H(PW)||t) et C'_K = (X$\oplus$A'$\oplus$ T_0). Le message de login est rejeté si $C_K \neq C'_K$, sinon la GW calcule V_M = H(X||A'||T_1).
PA2- La GW envoie le message (Acc_login, V_M, T_1) au LN et stocke t.
PA3-Le LN calcule V'_M = H(X||A|| T_1), et si $V_M = V'_M$, il calcule Y_K = H(V'_M||T_2).
PA4-Le LN envoie le message (Acc_login, Y_K, T_1, T_2) à l'utilisateur UD.
PA5-Après réception du message au temps T_3, l'UD vérifie si T_1 - $T_0\geq\triangle$T et T_0 - t$\geq\triangle$T. Si les conditions sont valides, l'UD calcule, $V"_M$ = H(X||A||T_1) et Y'_K = H($V"_M$||T_2), puis vérifie si $Y_K = Y'_K$. Si la condition est vraie, l'UD commence à obtenir les données, sinon il rejette le message d'acceptation de login.

La Figure 1 (b) montre le modèle de communication dans ARBPA.

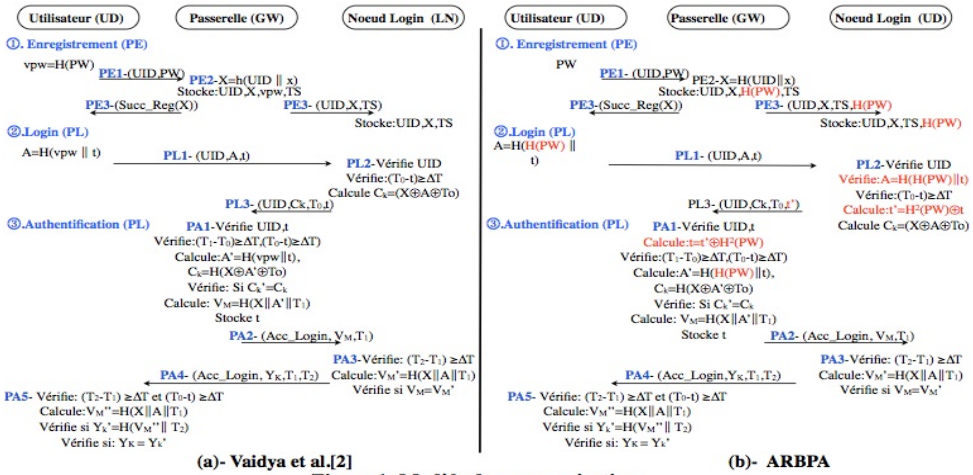

Figure 1: Modèle de communication

6. Sécurité de ARBPA

Dans cette partie, nous allons démontrer que notre solution est résistante aux attaques mentionnées. Cette section se termine par une étude comparative avec d'autres solutions.

6.1. Gestion des attaques par Déni de Service et par falsification

Le Deni de Service : en donnant aux LNs la possibilité de vérifier le mot de passe, notre solution se protège contre le DoS. Puisque le LN stocke H(PW), après réception du message (UID, A,t), il calcule A'=H(H(PW)||t). Si A'=A, alors le mot de passe est correct.

La falsification : pour éviter la falsification des estampilles temporaires entre le LN et la GW, nous les envoyons en mode sécurisé avec une utilisation du OU-exclusif (XOR). Ainsi le LN, après avoir reçu le message de login, calcule une fausse estampille t'=H^2(PW)$\oplus$t) puis envoie le message (UID,C_K,T_0,t') au lieu de (UID,C_K,T_0,t) à GW. Ce qui rend l'estampille t confidentiel. Puisque H^2(PW)$\oplus$t$\oplus H^2$(PW)=t, la GW calcule t=t'$\oplus H^2$(PW).

6.2. Etude comparative

Le Tableau 2, donne une comparaison en termes d'opérations de hachage, de OU-exclusif et de nombre de communications multi-sauts de quelques solutions.

Tableau 2. *Comparaison du nombre d'opérations effectuées*

Protocoles	Nombre total d'opérations
Wong et al. [11]	$7T_H$+$4T_{XOR}$+$3C_{MH}$
Tseng et al.[13]	$5T_H$+$4T_{XOR}$+$3C_{MH}$
Vaidya et al. [12]	$8T_H$+$4T_{XOR}$+$3C_{MH}$
Vaidya et al. [2]	$11T_H$+$4T_{XOR}$+$3C_{MH}$
ARBPA	$15T_H$+$7T_{XOR}$+$3C_{MH}$

T_H : temps pour exécuter la fonction de hachage H().
T_{XOR} : temps pour exécuter l'opération OU-exclusif .
C_{MH} : délai de communication multi-sauts entre le LN et la GW.

D'après le Tableau 2, on peut remarquer que la solution proposée a un coût supérieur de quatre opérations de hachage (T_H) et de trois opérations OU-exclusif (T_{XOR}) à celle de Vaidya et al.[2]. Notons que le coût en énergie de l'opération OU-exclusif est largement inférieur à celui du hachage.

Par ailleurs, si on considère les requêtes avec une UID valide mais avec un mot de passe erroné qui proviendrait de l'intrus ou d'une erreur de saisie de la part d'un utilisateur légitime. Pour ARBPA, cette requête sera freinée au niveau du LN, et pour les autres solutions, il sera propagé jusqu'à la GW. Une comparaison est effectuée sur le Tableau 3.

Tableau 3. *Comparaison du nombre d'opérations effectuées.*

Protocoles	Nombre total d'opérations
Vaidya et al.[12]	$4T_H$+$2T_{XOR}$+$2C_{MH}$
Vaidya et al. [2]	$4T_H$+$2T_{XOR}$+$2C_{MH}$
ARBPA	$3T_H$

7. Implémentation et Validation Expérimentale

Nous avons implémenté ARBPA et Vaidya et al. [2] avec TinyOS. Le programme est testé sur la plateforme MicaZ, et le simulateur Avrora est utilisé pour mesurer la consommation d'énergie. Le but est d'estimer la consommation énergie en sappuyant sur le nombre de sauts entre le LN et la GW, et de la probabilité d'une fausse requête.

Premièrement, nous avons évalué la consommation d'énergie en fonction du nombre de sauts entre le LN et la GW en se basant sur le Tableau 2 où les deux protocoles sont considérés sans aucune probabilité d'attaques. Pour chaque paramètre de données comme UID,A,PW, C_k etc., on a utilisé 16 bits. Pour le hachage, nous avons utilisé une implémentation de la fonction de hachage universelle PolyR décrite dans le papier de Ted and al.[14], comme une interface TinyOS. Sur la Figure 2, on remarque ARBPA consomme plus d'énergie que Vaidya et al.. A l'étape deux, nous nous intéressons à l'effet de propagation d'une fausse requête sur la consommation énergétique. A partir du Tableau 3, l'énergie consommée en fonction du nombre de sauts entre le LN et la GW est représentée sur la Figure 3. Cette énergie reste constante pour notre solution car la fausse requête ne se propage pas et augmente en fonction du nombre de sauts dans Vaidya et al. [2].

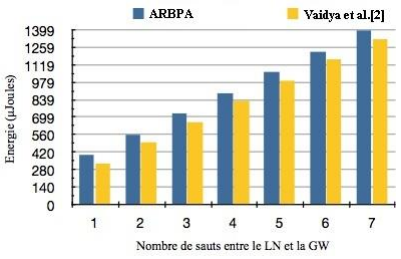

Figure 2 : Consommation énergétique basée sur le Tableau 2

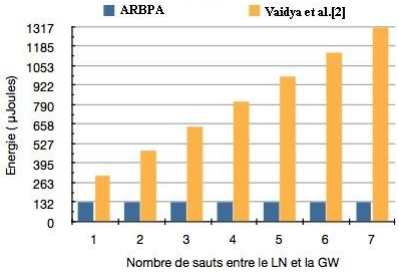

Figure 3 : Consommation énergétique basée sur le Tableau 3.

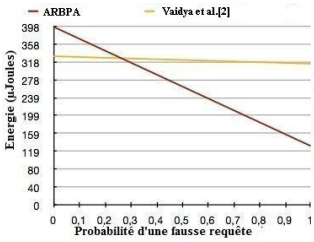

Figure 4. Energie à 1 saut entre LN et la GW

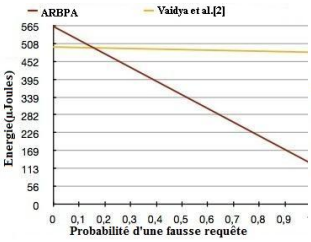

Figure 5. Energie à 2 sauts entre LN et la GW

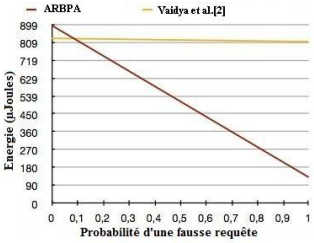

Figure 6. Energie à 4 sauts entre LN et la GW

Troisièment, nous introduisons la probabilité d'une fausse requête. Nous avons déterminé analytiquement par une formule la consommation énergétique en fonction de la probabilté d'une fausse requête. Selon une probabilité de distribution uniforme P_i d'une fausse requête, E_i désigne l'énergie consommée dans la formule suivante.

$$E_i = P_i * E_w + (1-P_i) * E_v$$

E_v=Energie consommée sans fausse requête (Figure2)
E_w=Energie consommée avec une fausse requête (Figure3)
E_i=Energie consommée avec une probabilité P_i
P_i=Probabilité de fausse requête

A partir de cette formule et du nombre de sauts entre le LN et la GW, nous avons déterminé la probabilité à partir de laquelle chaque solution justifie une meilleur consommation énergétique. Les résultats sont présentés sur la Figure 4 pour un saut, Figure 5 pour deux sauts et Figure 6 pour quatre sauts. On peut voir que notre solution présente des avantages significatifs. Les meilleurs résultats sont atteints à partir d'une probabilité $P_i \geq 0{,}3$ pour un saut, $P_i \geq 0{,}13$ pour deux sauts $P_i \geq 0{,}09$ pour quatre sauts.

8. Conclusion

Dans cet article, la solution proposée conserve tous les avantages de Vaidya et al.[2] et améliore sa sécurité par la protection contre le DoS et l'attaque par falsification. D'une part, sans aucun risque (probabilité d'attaque égale à 0), elle justifie d'une meilleure sécurité avec seulement un coût énergétique additionnel de quatre opérations T_H et de trois opérations T_{XOR}. D'autres part, on a démontré dans l'implémentation, qu'elle est énergétiquement meilleure en consommation. Pour une architecture donnée d'un RCSF basée sur le nombre de sauts entre le LN et la GW, nous avons aussi déterminé la probabilité à partir de laquelle notre solution présente une meilleure consommation énergétique. Nos travaux futurs vont dans le sens de vérifier les propriétés de sécurité avec un outil de vali-

dation. Ces propriétés sont le *secret* de H(PW), l'*authentification* respective entre la GW et LN et entre le LN et l'UD, et la *détection du rejeu* entre le LN et la GW.

9. Bibliographie

[1] Y. FAYE, I. NIANG, T. Noêl « A Survey of Access Control Schemes in Wireless Sensor Networks », *World Academy of Science, Engineering and Technology,*, n° Issue 59 : 2011, Paris, France, Pages 814-823, November 2011.

[2] BINOD VAIDYA, M. Chen , J. Rodrigues « Improved Robust User Authentication Scheme for Wireless Sensor Networks, Wireless Communication and Sensor Networks (WCSN) » , *2009 Fifth IEEE Conference : December 15-19*,

[3] A. K. AWASTHI,S. LAL, « A remote user authentication scheme using smart cards with Forward Secrecy, », *IEEE Transactions on Consumer Electronics,*, vol. 49, n° 4 pp.1246-1248, Nov. 2003.

[4] M. HWANG, C. Chang, K. Hwang, « An E1Gamal-like cryptosystem for enciphering large messages », *IEEE Trans .on Knowledge and Data Engineering*, vol. 14, n° 2, pp.445-446, 2002..

[5] C. C. LEE, L. H. Li, M. S. Hwang « A remote user authentication scheme using hash functions », *ACM Operating Systems Review*, vol. 36, n° 4, pp.23-29, 2002.

[6] J. J. SHEN, C. W. LIN, M. S. Hwang « A modified remote user authentication scheme using smart cards », *IEEE Trans. on Consumer Electron.*, vol. 49,n° 2, pp.414-416, May 2003.

[7] H. M. SUN, « An Efficient remote user authentication scheme using smart cards », *IEEE Trans. on Consumer Electron.*, vol. 46, n° 4, pp. 958-961, Nov. 2000.

[8] L. LAMPORT « Password authentication with insecure communication, Communications of the ACM, 1981. », *Commun ACM* , n° 1981 ; vol.24, no.11, pp.770-772.

[9] M.L. DAS, A. SAXENA, V.P. Gulati « A Dynamic ID-based Remote User Authentication Scheme », *IEEE Transactions on Consumer Electronics*, vol. 50, n° 2, 2004.

[10] C. LEE, C.H. Lin,, C. Chang « An Improved Low Communication Cost User Authentication Scheme for Mobile Communication », *Proceedings of the IEEE 19th International Conference on Advanced Information Networking and Applications (AINA 2005)*, n° Taiwan, 2005.

[11] K. WONG,Y. ZHENG, J. Cao, S Wang « A dynamic user authentication scheme for wireless sensor networks », *In Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC 06)*, vol. 1, n° Jun. 2006, pp. 244-251.

[12] B. VAIDYAJ.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 mobile networks (Q2SWinet 2009), Tenerife, Spain, Oct.*, n° 2009, pp 88-91.

[13] H. R. TSENGJAN, R. H., W. Yang « An improved dynamic user authentication scheme for wireless sensor networks. », *In Proceedings of the IEEE Global Communications Conference (GLOBECOM07), Nov. 2007*, n° 2007 ;986-990.

[14] TED KROVETZ,P. ROGAWAY, « Fast Universal Hashing with Small Keys and No Preprocessing : The PolyR Construction », *D. Won (Ed.) : ICISC 2000, LNCS 2015, pp. 73-89, 2001. Springer-Verlag Berlin Heidelberg 2001*,