ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronic and Communication Engineering (IJERECE) Vol 3, Issue 6, June 2016

Authentication by Zero Knowledge Protocol

[1] K. Prasanth [2] E.T. Jabajasphin
[1] PG Scholar [2] Associate Professor,
Department of Electronics and communication Engineering,
Saveetha engineering college , Thandalam,
Kanchipuram District.

Abstract: **- In wireless sensor network (WSN) various types of sensors are used and deployed in the network to collect useful physical parameters and some highly sensitive information is then been transmitted between the nodes and to the base station, without any human intervention. Hence, message authenticity and security are major requirements in WSN. Since the cryptographic schemes used for wired networks tend to exhaust wireless sensor network resources, they cannot be directly used in sensor networks. Here the Zero Knowledge Protocol (ZKP) is implemented in the network for the authentication and verification of sender sensor nodes before transmitting any sensitive information. In proposed scheme an optimal number of challenge questions are also used to maintain a balance between the added security and the increase in cost. Increase in the number of challenge question makes up to the reduced key size thus providing an improved security. The proposed scheme was assessed based on the MATLAB simulation and an analysis was performed.**

Index terms: **ZKP, WSN, authentication, sensor node.**

I. INTRODUCTION

Sensor nodes are basically battery powered with limited computation capability, Cluster Heads are nodes that are more powerful than sensor nodes and Base Stations are resource abundant. The key job of Nodes is to continuously collect data for events of interests and deliver the data to a designated Cluster Head. The major job of Cluster Head is to aggregate all the data's received and to send to a Base Station. Implementing these authentication mechanisms becomes very difficult because of their design constrains, due to limited resources and their physically insecure nature. The network autonomously must be able to identify and prevent itself from these attacks. Due to an insecure physical hardware and light weight operating systems, the network is prone to clone attack. In case of WSNs it is very easy for an adversary to capture or clone nodes and place them into the network by copying the cryptographic information. Malicious packet injection is also very common through man in the middle attack. Few of the highly regarded cryptographic mechanism such as RSA used in wired networks which proposed solutions for the above mentioned attacks cannot be used in WSNs because of the lack of memory and computation power and constrains on energy consumption, making it inappropriate.

II. RELATED WORK

Majority of the existing works in the field of WSNs security such as SPINS [11], LEAP [13], and TinySec [12] rely on symmetric encryption and secure establishment of their keys is yet another serious challenge. Hence an authentication mechanism would fail to comply if the attacker attains a secret key or password even in encrypted form. To overcome this issue, zero knowledge protocol (ZKP) [5] is used where zero information about the security key is exchanged between the communicating nodes. Zero knowledge proof method, implementation of identification schemes [6], and its variants have been suggested for authentication purposes in various domains. ZKP was first introduced by Goldwasser in [5] and it is proven for its efficiency in cryptography with small computational requirement. It can well be applied in places of authentication and secure key exchange. ZKF has extensively been used in many contexts; the usage of ZKP in Wireless sensor network is shown in [1-3].

In [1] ZKP is implemented for WSN considering the security attacks in WSN. It has been shown that ZKP proves efficient for the clone and man in the middle attacks. ZKP's use in WSN was proposed by [9], an Identification scheme for Base Nodes (IBN), where a group of sensor nodes cooperatively authenticates, also super-imposed disjunctive matrices were used for the finger print generation. The overall communication cost using the same would be high and in [2] a small version of

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronic and Communication Engineering (IJERECE) Vol 3, Issue 6, June 2016

ZKP is proposed for the wireless body area network (WBAN) systems, Tiny-ZKP [REF]. It is proven effective for such small networks but not for WSN systems. In this paper, we propose a ZKP model, which provides increased security, reduced communication cost with slight increase in computation cost. In this paper, the method for generation of fingerprint or the secret key provides better security and is very simple to use when compared to the technique in [2].

III. ZERO KNOWLEDGE PROTOCOL

In cryptography, a zero-knowledge protocol is a method by which one party (the prover) can prove to another party (the verifier) that a given statement is true, without conveying any additional information apart from the fact that the statement is indeed true. Notice that the notion only applies if the statement being proven is the fact that the prover has such knowledge. This is a particular case known as zero-knowledge proof of knowledge, and it nicely illustrates the essence of the notion of zero-knowledge proofs: proving that one possesses a certain knowledge is in most cases trivial if one is allowed to simply reveal that knowledge; the challenge is proving that one has such knowledge without revealing it or without revealing anything else.

For zero-knowledge proofs of knowledge, the protocol must necessarily require interactive input from the verifier, usually in the form of a challenge or challenges such that the responses from the prover will convince the verifier if and only if the statement is true (i.e., if the prover does have the claimed knowledge). This is clearly the case, since otherwise the verifier could record the execution of the protocol and replay it to someone else: if this were accepted by the new party as proof that the replaying party knows the secret information, then the new party's acceptance is either justified - the replier 'does' know the secret information - which means that the protocol leaks knowledge and is not zero-knowledge, or it is spurious.

Research in zero-knowledge proofs has been motivated by authentication systems where one party wants to prove its identity to a second party via some secret information (such as a password) but doesn't want the second party to learn anything about this secret. This is called a "zero-knowledge proof of knowledge". However, a password is insufficiently random to be used in many schemes for zero-knowledge proofs of knowledge.

Authentication using Zero Knowledge Protocol

The zero knowledge proof relies on the fact that during the entire process of authentication the secret key or password is not revealed to the receiver node. The nodes receive information only after successful authentication from the receiver node. There is a zero knowledge protocol mechanism between each node and cluster heads. Hence there is an authentication process before any transmission of data takes place.

The entire process of authentication (figure 1) can be explained in the following steps:

- ***Step1:*** The prover P chooses a random number r, calculates x (eq. 3)
- ***Step2:*** The prover P then send x to the verifier
- ***Step3***: Now the verifier requests for the prover's protocol key vp from the base station accompanied by its own protocol key vv = sv^2mod N where sv is the secret key of the verifier.
- ***Step4:*** Now the base station calculates the protocol keyof the verifier z = sv^2mod N, using the secret keystored for corresponding node id.
- ***Step5***: The base station then compares value z and protocol key received from the verifier. If both keys are equal, it authenticates the verifier.

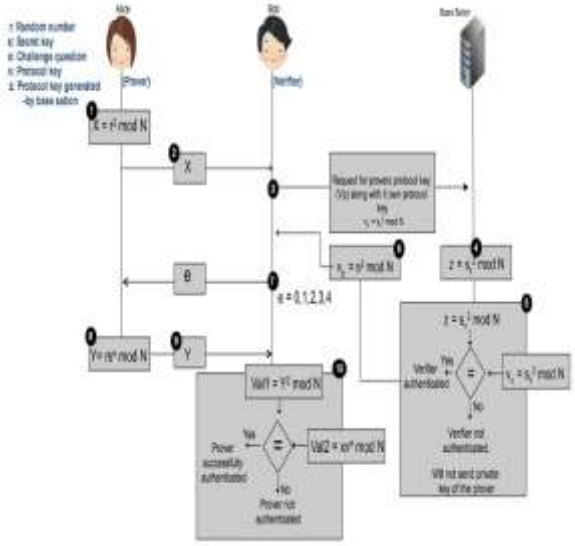

Figure1.Proposed Authentication Mechanism

- ***Step6:*** After successful authentication the base station computes the protocol key of the prover and sends it to the verifier. Transferring of protocol key in a multi-hop network will not affect his security as an adversary cannot deduce secret key from the protocol key.

- ***Step7:*** The verifier V now chooses a random challenge question e (e=0, 1, 2, 3, 4), asks and ask the prover P for $Y = rs^e \bmod N$.
- ***Step8:*** Prover then calculates Y using random number r, secret key s and the challenge question e.
- ***Step9***: Prover then sends back Y as a response to thechallenge question.
- ***Step10:*** the verifier then compute and two values val1 $=Y^2 \bmod N$ and val2 $= xv^e \bmod N$ with each other.Val1 and val2 will only be equal if the secretprovided by the prover matches the secret keyprovided by the base station. As BS is the trustedparty, the key match will prove the authenticity ofthe prover P.

The secret key s_j for a node is calculated by a using a random hash function (eq 1).

$$s_j = \sum_{i=1}^{k} (C_j + P_i) mod\ F \quad (1)$$

The authentication process is initialized by the prover by generating a random number r and calculates $x = r^2 \bmod N$ (eq. 2).

$$x = r^2 \bmod N \quad (2)$$

Prover then sends x to the verifier (Node 4). Now the verifier will request the base station for the protocol key v_2 (eq. 3)of the prover and sends its own protocol key $v_4 = s_4^2 \bmod N$along with it.

$$V_2 = s_2^2 \bmod N: \quad (3)$$

The base station will first compute the protocol key of the verifier using the stored secret key of the same (s_4) and then compares it with the protocol key received from the verifier. If both values are found equal, then the BS replies back with the protocol key (v_2) of the prover. This mechanism is termed as two-way authentication in our proposed method. This mechanism is implemented to authenticate the verifier node before the base station shares any protocol key, hence improving security in the network. The verifier will ask a random challenge question to the prover after receiving the protocol key of the prover from the base station. This challenge question will be a random value of e. Based upon the challenge question asked by the verifier, theprover replies with Y (eq. 4). The verifier then calculates andcompares two values val1 (eq. 5) and val2 (eq. 7).

$$Y = rs_2^e \bmod N \quad (4)$$

$$val1 = Y^2 \bmod N \quad (5)$$

$$val1 = Y^2 \bmod N = (rs_2^e)^2 \bmod N = r^2 s_2^{2e} \bmod N \quad (6)$$

$$val2 = xv_2^e \bmod N \quad (7)$$

$$val2 = xv_2^e \bmod N = r^2 \bmod N (s_2^2)^e \bmod N = r^2 s_2^{2e} \bmod N \quad (8)$$

For example for challenge question e=1:

$$val1 = r^2 s_2^2 \bmod N \quad (9)$$

$$val2 = r^2 s_2^2 \bmod N \quad (10)$$

The secret key $s_2$in (eq. 5) comes from the proverand the secret key $s_2$in (eq. 7) comes from the basestation. This way both val1 and val2 will only be equal ifboth the secret keys match with each other i.e. if the proveris legitimate and is using the secret key stored during the pre-deployment stage. By using this protocol the verifier can compare and match the secret key of the prover from two sources without even learning the actual value of the secret key of the prover.

This complete process of authentication, except for protocol key exchange,isrepeated K times and for each round, a new random number rand random challenge question e is chosen. The protocol also requires the response to a challenge to be provided within a time limit such that it becomes computationally infeasible for an impersonator to answer to the challenge by using brute force method. This authentication mechanism is performed before every initialization of data transmission. For every authentication process, a new public key N is generated by the base station. The entire process of authentication is shown in figure 2, where Alice acts as the prover and Bob acts as verifier and public key N is shared among them by the base station.

In ZKP, the challenge question e plays a very crucial part to authenticate the sender node. If an adversary in the beginning of the authentication process possesses the challenge question e, it can easily prove itself to be a genuine sender. Hence it is very important that the authentication process is repeated multiple times and challenge key e should be chosen randomly for each round. For an instance, if a false claimant knows the challenge question e before the start of the authentication process, it can claim to be a genuine node by generating an arbitrary number a and sends $x = a^2/v^e \bmod N$ to the verifier. Upon receiving the expected value of e, the false claimant sends Y = a. The verifier will then computes $val1 = Y^2 \bmod N$, which becomes $a^2 \bmod N$. It also computes $val2 = xv^e \bmod N$ which becomes $a^2/v^e \bmod N$. This process will result in val1 and val2 to be equal. This will lead to false authentication. In [2] only two challenge questions were used (e = 0, 1).

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronic and Communication Engineering (IJERECE) Vol 3, Issue 6, June 2016

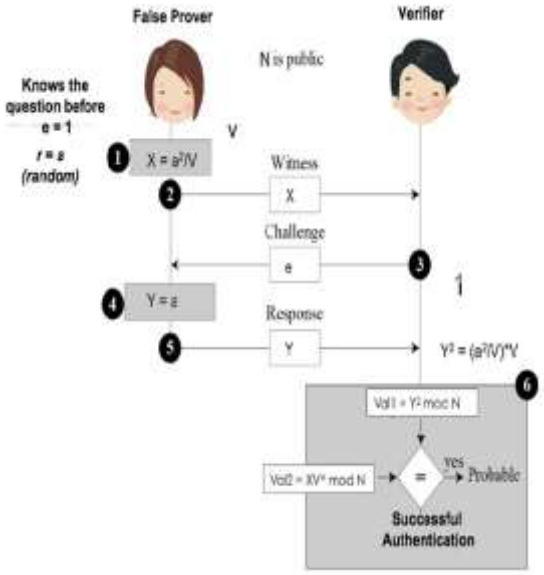

Figure 2. False claimant authentication in ZKP

Here the number of challenge questions increased to 5 (e = 0, 1, 2, 3, 4). By having more options for e value we have increased security of the protocol. If we have only one option for e i.e. e=0, then at all rounds the adversary can authenticate itself and ZKP will fail. If wehave two options for e, i.e. e=0 and e=1, we will have a good security but even then the randomness can be predicted by high probability. So if we have more options for e, the security is increased y several folds, but simultaneously the computation cost will also increase. So through our simulation we found an optimal number of challenge questions to maintain a balance between the number of challenge questions and that of the computation cost. By increasing the number of challenge questions we were able to reduce the number of rounds the authentication must be performed.

In [2] when e has 2 values the optimal number of rounds was10, which makes the probability of successful authentication by a malicious node to be $(1/2)^{10}$. Here ZKP able to improve the security with lesser number of rounds (K), by increasing the number of challenge questionse. After successful authentication of the node K times, the data transfer phase starts, wherein the nodes exchange information.

Results Discussion

Here a model of ZKP scheme is developed in MATLAB, and various performance analyses were done.

1) Computation cost: Computation cost for proposed protocol can be derived as (eq. 11), similar to the work in [18], where t is the number of rounds, l being number of challenge questions and k is the multiplicity of challenge.

2)

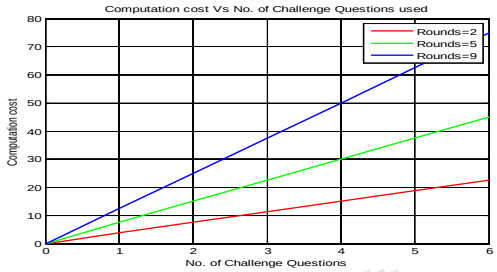

Figure 3. Computation cost vs no of challenge questions

The simulation results of the proposed network model and its computation results are shown below.

Computation cost = t.l.(k + 1)/4 (11)

Figure 3 shows the number of challenge questions versus the computation plot for different number of rounds, we can notice that as number of challenge question increases, the computation cost also increases. We can also see that computational cost will increase more rapidly with the increase in number of rounds.

2) Security: In figure 4, ZKP performs better when numbers of challenge questions are increased. Security in the system is measured with increase in challenge questions and we found that with increase in number of challenge questions, lesser number of rounds are required to provide optimum security than in [2].

Security = $2^{(k.t.l/2)}$

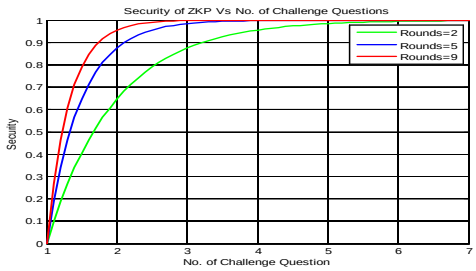

Figure 4.Security of the proposed ZKP model.

REFERENCES

[1] K.R. Venugopal and L.M. Patnaik, "Authentication in WirelessSensor Networks Using Zero Knowledge

Protocol", ICIP 2011,CCIS 157, pp. 416¸S421, 2011.Springer-Verlag Berlin Heidelberg 2011.

[2] Siba K. Udgata, AlefiahMubeen, Samrat L. Sabat, "WirelessSensor Network Security model using Zero Knowledge Protocol" Proceedings of ICC, IEEE International Conference, 2011,pages 1-5.

[3] Limin Ma, Yu Ge, Yuesheng Zhu, "TinyZKP: A Lightweight Authentication Scheme Based on Zero-Knowledge Proof for Wireless Body Area", Wireless PersCommun DOI 10.1007/s11277-013-1555-4, Springer Science+Business Media New York 2013.

[4] Joseph Binder, Hans Peter Bischof, "Zero Knowledge Proofs ofIdentity for Ad Hoc Wireless Networks An In-Depth Study",Technical Report, 2003. http://www.cs.rit.edu/sb7384/zkpsurvey.pdf

[5] Goldwasser, S., Micali, S., Rackoff, "The Knowledge Complexity of Interactive Proof Systems". SIAM J. Computing 18,186â A¸S208 (1989).

[6] Feige, U., Fiat, A., Shamir, "A.: Zero Knowledge Proofs ofIdentity". J. Cryptology 1, 77âS94 (1988).

[7] Goldreich, O., Micali, S., and Wigderson, "Proofs That YieldNothing But Their Validity Or All Languages in NP HaveZeroknowledge Proof Systems", Journal of the ACM, Vol. 38,No. 1, pp.691-729, 1991.

[8] Tuyls, Pim T. (Mol, BE), Murray, Bruce (Eastleigh, GB), "Efficient Implementation of Zero Knowledge Protocols" ,UnitedStates NXP B.V. (Eindhoven, NL) 7555646 ,June 2009,http://www.freepatentsonline.com/7555646.html.

[9] MuneeraHashim, Santhosh Kumar G., and Sreekumar A, "Authentication in Wireless Sensor Networks Using Zero Knowledge Protocol" ICIP 2011, CCIS 157, pp. 416-¸S421, 011. C Springer-Verlag Berlin Heidelberg 2011.

[10] Anshul, D., Roy, S.S.: "A ZKP-based Identification for BaseNodes in Wireless Sensor Networks". In: 2005 ACM Symposium on Applied Computing, pp. 319-¸S323. ACM Press, NewYork (2005).

[11] A. Perrig, R.Szewczyk, V.Wen, D.Culler, and J.Tygar.â˘AIJSpins: Security protocol for sensor networks,I In proceedings of Seventh Annual International Conference on MobileComputing and Networks, July 2001.

[12] C. Karlof, N. Sastry, and D. Wagner, Tinysec: a link layersecurity architecture for wireless sensor networks, in SenSys,2004, pp. 162-175.

[13] S. Zhu, S. Setia, and S. Jajodia, Leap+: Efcient security mechanisms for large-scale distributed sensor networks, TOSN, vol.2, 2006.