IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 17, Issue 6, Ver. IV (Nov – Dec. 2015), PP 44-54
www.iosrjournals.org

Ndsp: Neighbor Assisted Distributed Self-Healing Protocol for Compromised Node Recovery in Wireless Sensor Networks

Silochana Devi, RohitVaid
[1,2]*CSE Department M. M. Engineering College, M. M. University, Mullana, Ambala, India-133207*

Abstract: *Wireless sensors network (WSN) is collection of sensors nodes and a base station. All the sensor nodes in the network collect data from the environment and sent this data to the base station (BS). This data is transmitted in hope by hope fashion, i.e. each sensor sent the data to its one hope neighbor in the direction of BS on wireless medium. As themedium is wireless which is already less securedthus more prone to attacks. Therefore security is one of the major issues in WSN. An adversary is a node which steals information stored in the memory of sensor nodes. This paper presentsa neighbor assisted distributed self-healing protocol (NDSP) for compromised node recoveryin wireless sensor networks. Presented NDSP protocol allows a compromised sensor node to continuously and collectively recover from a compromise stage to a normal stage. Detection of compromised node is out of the scope of this research paper. Simulation and analytical results proves that presented scheme to recover a compromised sensor node is both effective and efficient.*

Keywords: *WSN, BS, Recovery*

I. Introduction To Wireless Sensor Networks

Wireless sensor networks can virtually work in any environment, especially where wired connections are not possible. It is consistsof a number of sensor nodes that are used to gather information from the environment, and actuator nodes that are used to change the behavior of the environment. Wireless sensors network in sensor node are usually random deployed in the interested area. When the node is deployed in area does not effect in other area. Wireless networks can also be deployed in extreme environmental conditions and may be prone to enemy attacks These sensors are consist of low power, limited memory, and energy constrained due to their small size.

From a technical perspective, a sensor is a device that translates parameters or events in the physical world into signals that can be measured and analyzed. During development of a wireless sensor node, it is ensured that there is always adequate energy available to power the system. Power is stored either in batteries or capacitors. Batteries used for power supply can be both rechargeable and non-rechargeable. Sensor nodes can use limited supply of energy performing computations and transmitting information in a wireless environment. As such, energy-conserving forms of communication and computation are essential. Sensor node lifetime shows a strong dependence on the battery lifetime.

Wireless Sensor Networks (WSNs) is to collect information from the physical world. It is consists of base stations and a number of wireless sensors.It also represent the mobility of nods, it means movement of node one node to other node in interested area. When the change of node no effect are other sensor node. They are performing easier.Sensor nodes sense and report the state of the environment while actuator nodes gather data from sensors and are able to act on the environment. It is expected to be self-organized and potentially operate autonomously in unattended environments, with basic and Minimal directives from the user that might be remotely connected to the scene. In the wireless communication the number of sensor nodes deployed in a monitoring region may be in the order of hundreds or thousands while such size is not necessary for actuator nodes since they have higher capabilities and can act on larger areas.The unreliable Sensor nodes represents the sensor node are limited energy and small size .There are always risk that a node will fail or physically damage. This will be causes of unreliability in the network.

While many sensors connect to controllers and processing stations directly (e.g., using local area networks), an increasing number of sensors communicate the collected data wirelessly to a centralized processing station. This is important since many network applications require hundreds or thousands of sensor nodes, often deployed in remote and inaccessible. Therefore, a wireless sensor has not only a sensing component, but also on-board processing, communication, and storage capabilities. With these enhancements, a sensor node is often not only responsible for data collection, but also for in-network analysis, correlation, and fusion of its own sensor data and data from other sensor nodes for example, using ultrasound, infrared, or radio frequency technologies with varying data rates and latencies. Finally, some devices may have access to additional supporting technologies, for example, Global Positioning System (GPS) receivers. It is represent the random node is deployed in unattended environment; they are usually operated by battery to perform any type of operation. It is also impossible to change or recharge their batteries once they are deployed. In the WSNs the

DOI: 10.9790/0661-17644454

node is deployed randomly in interested area without careful planning. Once node deployed, they are perform the autonomously configure themselves into the communication network. Although deployed in an ad hoc manner they need to be self-organized and self-healing and can face constant reconfiguration. The WSNs represent theMany-to-One Traffic Pattern, in most of the sensor network applications, when the need of same data in base station then the every sensor node sends from the data in the direction of a base station. It process cycle is called many to one traffic process

WSNs have been employed in many applications, such as environment monitoring and health care.There are many applications in monitoring environmental and earth sensing parameters. These are Forest Fire Detection, Air pollution Water quality monitoring, and Landslide detections, HealthApplications, Areamonitoring, Security and Surveillance.

II. Review Of Literature

This section presents various works that has been done in the field of distributed self-healing, data secrecy, in WSNs. The reviews of previous work help us to analyses the problems in the existing system and we can get some help to develop our proposed system.

V. Naik, et al. [1] proposed a Local secret maintenance in sensor networks. In this paper present a simple protocol for secret maintenance between a pair of network. It is show that if the current secret between the pair is somehow disclosed and previous key are not compromised and nor can future secret compromised. Local keys are updated periodically to mitigate the effect of sensor compromise. Thisprotocol provides both forward and backward security for communication between pair of sensors. However, the scheme's security relies on a somewhat unrealistic assumption that the adversary is unable to compromise both sensors simultaneously. The conclusion of paper whisper is the first piece of work in which neighboring node to node local secret maintained with the property of forward secrecy achieved is using session key only.

Hu, F. et al. [2] proposed a Security in wireless sensor networks. Data secrecy is a fundamental security issue in sensor networks and encryption is the standard way to achieve it Many research efforts have yielded techniques for establishing pair wise keys used to secure sensor-to-sensor and sink-to-sensor communication

D. Ma et al. [3] presents unattended sensors and sensor networks have become subject of attention in the security research community and various aspects of security have been explored. Parno, et al. proposed two distributed algorithms where sensors (without interference of sink) work collectively to detect node replication attack. Security and privacy in data-centric sensor networks typically running in unattended mode have been recently studied in. Unattended wireless sensor networks (UWSNs) operating in hostile environments face the risk of compromise. Unable to offload collected data to a sink or some other trusted external entity, sensors must protect themselves by attempting to mitigate potential compromise and safeguarding their data .UWSNs have also been considered the context of minimizing storage and bandwidth overhead due to data authentication [1].

A generalization of this is the cryptographic literature; in this scheme they consider notions of intrusion-resilience [4] and key insulation [5] refer to techniques of providing both forward and backward security to mitigate the effect of exposure of decryption keys. However, these techniques are unsuitable for solving the problem at hand, control of the adversary. Data integrity is an equally important issue which is normally considered in random with data secrecy. However, in this paper, we ignore data integrity. This is because we distinguish between read-only and read-write adversaries. The former is assumed to compromise sensors and leave no evidence behind: it merely reads all memory and storage. In contrast, a read-write adversary can delete or modify existing – and/or introduce its own fraudulent data. We consider a read-only adversary to be more realistic, especially since it aims to remain stealthy. A stealthy adversary has an incentive (and the ability) to visit the UWSN again and again, while a non-stealthy one might be unable to do so once an attack is detected and corresponding measurements are taken.

Author in [6] presents Self-Healing strategies. The strategies they perform to investigate two cooperative self-healing strategies that allow an UWSN to recover from compromise and maintain secrecy of collected data. Because the cure comes from peer sensors, the network exhibits a self-healing property that emerges through collaboration of all nodes – something no individual node can provide. In this paper we propose DISH (Distributed Self-Healing), a scheme where unattended sensors collectively attempt to recover from compromise and maintain secrecy of collected data. DISH does not absolutely guarantee data secrecy; instead, it offers probabilistic tunable degree of secrecy which depends on variables such as: adversarial capability (number of nodes it can compromise at a given time interval), amount of inter-node communication the UWSNs can support, and number of data collection intervals between successive sink visits. We believe that this work represents the first attempt to cope with the powerful mobile adversary in UWSNs. We also show that, in this context, healing capabilities are subject to a 0-1 law. Results obtained from our analysis are supported by extensive simulations, showing that the proposed protocols are very effective in self-healing, despite the power of the mobile adversary. Finally, some issues related to UWSN deployment are addressed, while some open research problems calling for further investigation are introduced

Huhns et al. [7] have proposed a Software agent-based self-healing architecture presented the concept of multi-agent redundancy to fabricate software adaptation. Software engineering intersects multi-agent systems in many ways. Such as, multi-agent systems can be used to assist conventional software systems or traditional software engineering techniques can be used to build multi-agent systems. The benefits of using agents as building blocks for conventional software are agents can dynamically compose in a system when all the components of the system are unknown till runtime. Also, as agents can be added to a system in run time, software can be customized over its lifetime, even by the end-users too. This can produce more robust systems.

Blundo, et al. [8] proposed Self-healing Key Distribution Scheme**.** In this work, the author includes the proposed forward and backward secrecy. Forward secrecy is usedto prevent a revoked user from continued accessing thesession key even if it keeps receiving the broadcast messages. Backward secrecy is used to prevent a new user from decoding messages broadcasted before it joins the group. When a group requires forward and backward secrecy, the session key must be changed for every membership change.

Canetti and Herzberg [9] proposed a generic deterministic Scheme is proposed that maintains secrecy in the presence of a mobile adversary. In this scheme, a node must communicate with all other nodes to update its state at each round. This might work for small wired networks, but due to communication overhead the proposal would not scale to the envisaged UWSN size. Also, since sensor communication is wireless and broadcast in nature, Eavesdropping is easy, which makes our analysis very different from that in unattended sensors and sensor networks have become subject of attention. The initial work introduced the UWSN scenario, defined the mobile adversary and discussed a number of challenges in the new scenario. This work is later extended to include the case where the adversary's goal is to indiscriminately erase all sensor data.

Authors in [10] proposed a distributed scheme to detect node replication attack in sensor network. In this attack, an adversary uses the credentials of a compromised node to introduce replicas of a node in the network.

III. Problem Description

In this section, we describe network assumptions and different states of sensor nodes in the compromised WSN.

A. Network Assumptions

All the sensors in the network know the public key of BS PK_{BS}. As soon as a sensornode S_i collects data d_i^r in round r, S_i encrypts this data using following procedure: $E_i^r = Enc(PK_{BS}, R_i^r, d_i^r, r, S_i, ...)$ where R_i^r refer to one time random number generated by sensor node S_i in round 'r' include in each randomized encryption operation. Each sensor node encrypts the data using public key of BS whereas BS decrypts this data using its own private key K_{BS}. Each sensor node uses its own seed value $Seed_i$ to generate the random number R_i^r in round r. The random number is generated by applying one way hash function $\int_h$ on seed value $Seed_i$ .The function $\int_h$ is applied r^{th} time on seed value $Seed_i$ to generate the random number R_i^r used in round r.

B. Type of compromised node attack

Once an adversary compromised a sensor node S_i in the network, it steels all the information stored in the memory of S_i. As the sensor node S_i is compromised, its secret seed value to generate the random number R_i^r is also disclosed to adversary. So any key which is generated by a compromised sensor node S_i is alsodisclosed to the adversary. The adversary is only interested in steeling the information stored in the memory of sensor node S_i, adversary never interferes or steel any message communicated over wireless medium between two nodes weather the nodes are compromised or not.

So once a node is compromised, it is no use to generate a random number R_i^r using its compromised seed value. At this stage, it is time to replace the seed value $Seed_i$ using new seed value $Seed_i'$. This is done by replacing the seed value $Seed_i$ of i^{th} sensor with the seed value of $Seed_j$ of j^{th} sensor.

C. Types of Sensor Nodes in compromised WSNs

There are three types of node in the network. If the network is not compromised, only single type of nodes are presents in the network, i.e. healthy nodes. But once the network is compromised and it is under the control of an adversary node, all the compromised sensor nodes changes there states in to a different states. There are three types of nodes presents in the compromised WSNs which are as under:

a. Healthy Node: A node is healthy if its internal data and seed values to generate a key are confidential from adversary node.

b. Sick Node: A node is in Sick state when data stored in the node memory is available to adversary but the seed value to generate the key is safe from adversary.

c. Compromised Node: A node is in compromised state when both data stored in the memory of node along with seed value to generate the key generation material are available to adversary.

IV. Ndsp Protocol

All the sensor nodes are programmed for collecting data periodically. Each sensor waits for a pre-determined time to upload data to the mobile sink node. Each time a sink visits a sensor node; all the network security parameters of all the sensors are securely reinitialized which includes all cryptography key as well as initial seed value for random number generation, etc. Adversary can compromised at most k number of sensors out of total N number of sensors in the interval of single collection. Adversary's only interest is in the secret of compromised node it does not interfere in the communication of any sensors in does not try to change anything in them. Adversary movements are unpredictable and untraceable.

The intrusion detection process is out of scope of the research work. In the presented scheme, the unattended sensor nodes collectively attempt to recover from compromise state to sick or healthy state to maintain network performance in terms of secrecy of collected data. Because the cure comes from peer sensors, the network exhibits a self-healing property that emerges through collaboration of all nodes.

Figure 1 shows the system model of neighbor assisted distributed scheme for compromised node recover in wireless sensor networks. The process starts with the intrusion detection. If the system detects an intruder in the network, then it starts repairing the network. The network is assumed to be a normal network only if no more than 70% of the nodes are compromised. So if entire network is not compromised then system repairs all the sick and compromised sensors by refilling the seed values of such nodes with the seed value generated by healthy and sick sensors as shown in Figure 2.

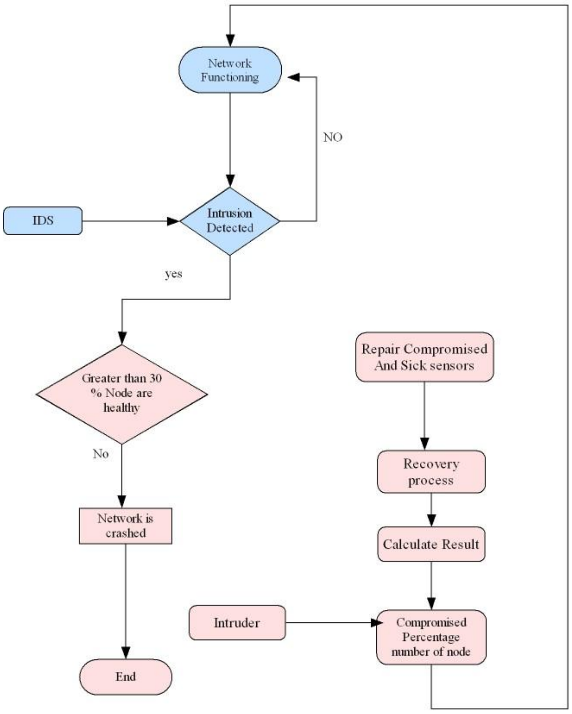

Figure 1: System model for compromised sensors recovery

D. Compromised and sick node recovery

Figure 2 shows the neighbor assisted distributed scheme for compromised node recover in wireless sensor networks.A healthy (H) sensor helps in replacing the seed value of both sick (S) and compromised (C) sensors. A healthy sensor selects its randomly neighbor 'R'. If the selected neighbor 'R' is already in a healthy state then it does not change the state of 'R'.But if the state of 'R' is sick state then it will automatically changes into healthy state. On the other hand if 'R' is in compromised state, then the state of 'R' is changes into sick state.

A sick sensor helps in replacing the seed value of compromised sensor only. But the sensor is already sick so a single sick sensor is unable to recover a compromised state into a sick state. Two sick sensors are required for the recovery of a compromised sensor from compromised state into a sick state. Figure 2 shows the recovery process where a healthy sensor gives recovery to a sick and compromised sensor. And similarly two sick sensors give recovery to a compromised sensor.

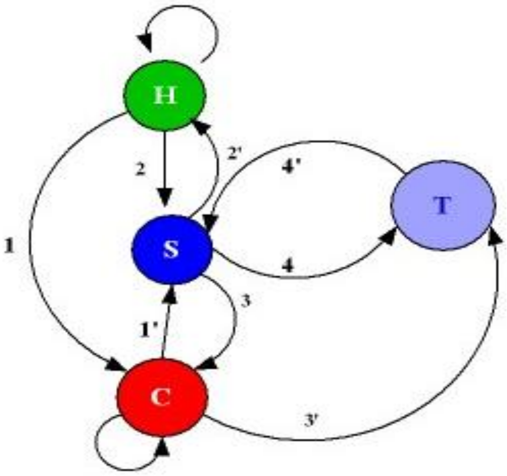

Figure 2: Network self-healing for compromised node recovery

V. Simulation Environment

Figure 3 shows an environment of wireless sensor networks. The size of network is 100X100 meter square in 2-dimentional area 'A'. Thirty sensors are deployed randomly in this area. So the network consist of a set of sensor nodes 'S' such that $S=\{S_1, S_2, ..., S_{30}\}$ .The network is static where each sensor s_i is located at its random location identified by a coordinate (x_i , y_i). Each sensor is identified by a unique number known as Identification (ID) number.

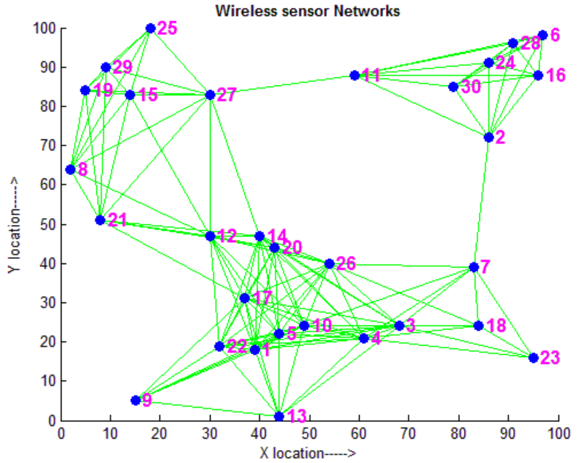

Figure 3: Randomly deployed wireless sensor networks

Each sensor node identifies its one hope neighbors that falls within its limited communication range. Figure 3 shows a link between a sensor node and its one hope neighboring sensor nodes. Neighbor of each node are shown in Table 2.

Table 2: Single Hope neighbor

Node	Neighbors
1	3,4,5,9,10,12,13,14,17,20,22,26
2	6,7,11,16,24,28,30
3	1,4,5,7,10,13,14,17,18,20,22,23,26
4	1,3,5,7,10,13,14,17,18,20,22,23,26
5	1,3,4,9,10,12,13,14,17,20,22,26
6	2,11,16,24,28,30
7	2,3,4,10,18,23,26
8	12,15,19,21,25,27,29
9	1,5,10,13,17,22

10	1,3,4,5,7,9,12,13,14,17,18,20,22,26
11	2,6,16,24,27,28,30
12	1,5,8,10,14,15,17,20,21,22,26,27
13	1,3,4,5,9,10,17,22
14	1,3,4,5,10,12,17,20,21,22,26,27
15	8,12,19,21,25,27,29
16	2,6,11,24,28,30
17	1,3,4,5,9,10,12,13,14,20,21,22,26
18	3,4,7,10,23,26
19	8,15,21,25,27,29
20	1,3,4,5,10,12,14,17,21,22,26
21	8,12,14,15,17,19,20,27,29
22	1,3,4,5,9,10,12,13,14,17,20,26
23	3,4,7,18
24	2,6,11,16,28,30
25	8,15,19,27,29
26	1,3,4,5,7,10,12,14,17,18,20,22
27	8,11,12,14,15,19,21,25,29
28	2,6,11,16,24,30
29	8,15,19,21,25,27
30	2,6,11,16,24,28

A. Compromised Network

In each round, the adversary node attacks over the network. It then controls a number of sensor nodes from the network. These sensor nodes are known as compromised sensors. Figure 4 shows a compromised wireless sensor networks. Three types of nodes are presents in the compromised wireless sensor network, i.e. Healthy (H), Sick (S) and Compromised (C). All three types of nodes (Healthy, Sick and Compromised) are randomly selected to show the behavior of the network. Each types of sensor nodes selected randomly in the compromised wireless sensor network are given below:

ID of Compromised sensors: 3, 4, 5, 15, 17, 21, 23, 24, 27, 29.

Sick sensors: 6, 8, 12, 13, 14, 16, 19, 22, 25, 28.

Healthy sensors: 1, 2, 7, 9, 10, 11, 18, 20, 26, 30.

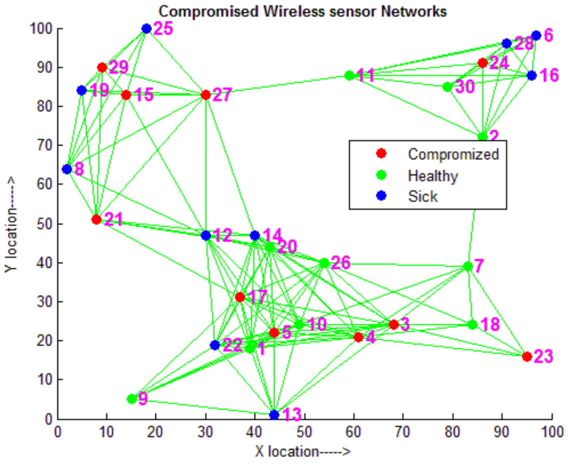

Figure 4: Network with Compromised Nodes

B. Network Recovery

Figure 5 represent the process of network recovery in presented NDSP protocol. If a node is compromised then it is recovered only when it update its seed value from a healthy or two sick sensors. Similarly a sick sensor updates its seed value from healthy sensors. A sensor node from compromised state is changes into sick state only when it updates its seed value from ahealthy sensor or from two different sick sensors, then the stage of compromised sensor becomes sick. Similarly a sick sensor update its seed value with the help of a healthy sensor, it also becomes healthy. Figure 6 shows a scenario where each sensor node selects its one randomly selected neighbor to update its seed value.

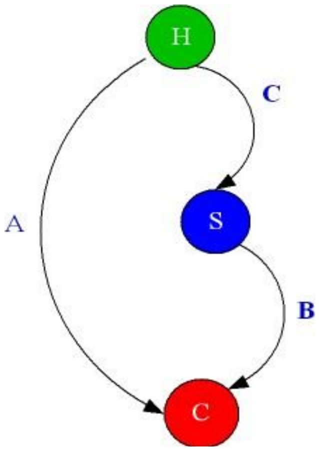

Figure 5: Exchange of seed value in compromised network

a. Network before Recovery

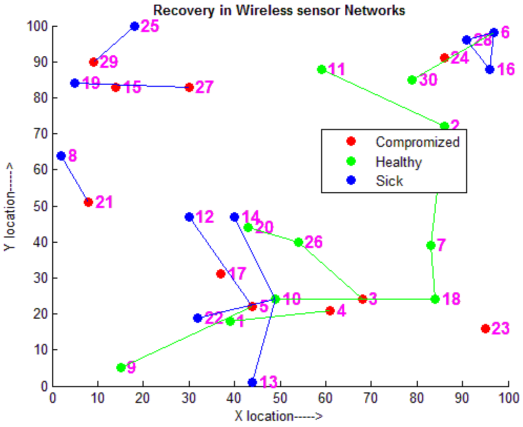

Figure 6: Healthy and sick sensors updating seed value of their one random neighbor

Table 3 show the status of each sensor nodes which are selected by a Healthy sensor node as a randomly selected neighbor.

Table 3: Recovery given by a healthy sensor to its one randomly selected neighbor

ID of Healthy sensor	State of neighboring sensors
1	4: C
2	7:H
7	18:H
9	5:C
10	18:H
11	2:H
18	7:H
20	26:H
26	3:C
30	6:S

Similarly each sick sensor also selects its one random neighbor to change their state as shown in Table 4. In this way each healthy and sick sensors participate in recovery process.

Table 4: Recovery given by a sick sensor to its one randomly selected neighbor

ID of Healthy sensor	State of neighboring sensors
6	28:S
8	21:C
12	5:C
13	10:H
14	10:H
16	6:S
19	27:C
22	10:H
25	29:C
28	16:S

b. Network after Recovery

Figures 7 shows recovered wireless sensor network. The ID of each types of sensor node after recovery is given in Table 4. Table 5 shows the status of those nodes that have recovered their state from compromise state to sick state or healthy state and similarly from a sick state to a healthy state.

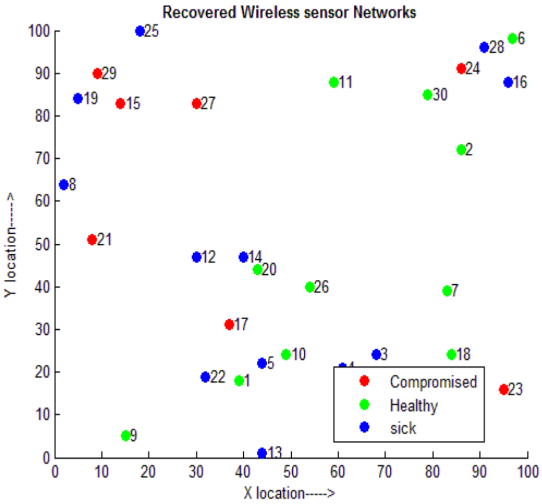

Figure 7: Status of nodes after recovery

Table 5: Sensor nodeIDafter recovery

Compromised Sensors	Healthy Sensors	Sick Sensors
15	1	3
17	2	4
21	6	5
23	7	8
24	9	12
27	10	13
29	11	14
	18	16
	20	19
	26	22
	30	25
		28

Table 6: Sensor state after recovery

Sensors ID with state before recovery	Recovered state
3:C	S
4:C	S
5:C	S
6:S	H

VI. Simulation Results

Network lifetime is measured in terms of number of healthy, compromised and sick sensor nodes with respect to different number of compromised sensor nodes.

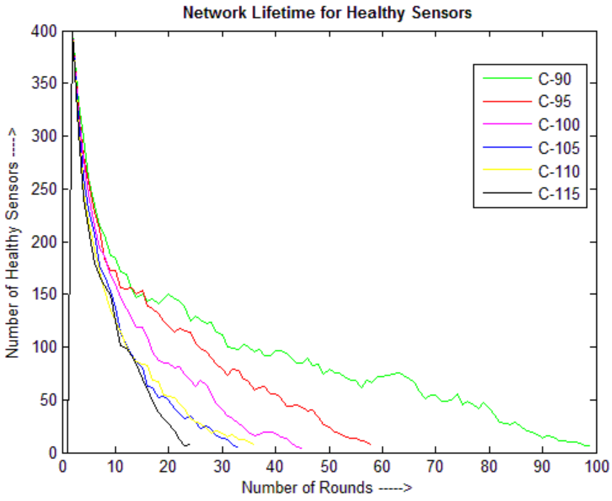

Figure 8: Network Lifetime of Healthy Sensor with increasing Number of Compromised Sensor Nodes

A. Network Life Time With Different Compromise Ratio

Network lifetime in terms of healthy sensor nodes is measured as the number of sensor nodes still healthy by recovery process in presence of compromised sensor nodes. Number of compromised nodesare increased from 90 sensors to 95,100,105,110,115to chck the behavior of the network. The simulation is run for the network till more than 10% nodes are healthy or in other words more than 90% of the total number of nodes are not compromisedas shown in Figure 8. The lifetime of network is 100 number of rounds when number of compromised sensor nodes in each round are 90 whereas more than 80% nodes are die befor 25 number of rounds when the number of compromised sensor nodes in each round are 115.

Similarly number of compromised sensor nodes with different compromised ration is shown in Figure 9. When the number of compromised sensor nodes in each round is 115, the total numbers of sensor nodes that are compromised in the network are increased to 360 sensors after 25 numbers of rounds. On the other hand when the number of compromised sensor nodes in each round is 90 then the total numbers of compromised sensor nodes in the network are 360 sensors after 95 numbers of rounds.

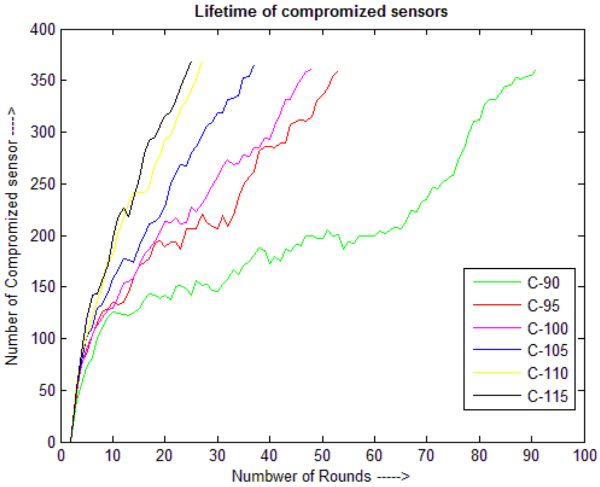

Figure 9: Lifetime of compromised Sensor with Variable Number of Compromised Sensor Nodes

Figure 10 shows network lifetime for sick sensor with same compromised ratio, i.e. 90 compromised sensor to 115 number of compromised sensor in each round. The numbers of sick sensors increases in each round for the first 15 rounds only. When 80% of the sensors are compromised, the numbers of sick sensors decreases due to network recovery by healthy and sick sensors. Otherwise all the healthy sensors in the network changes to sick state and number of sick and compromised sensors increases.

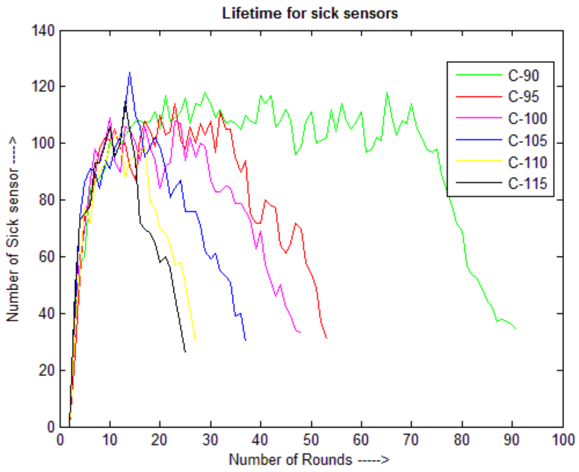

Figure 10: Lifetime of sick Sensor with Variable Number of Compromised Sensor Nodes

B. Status of Sick, Healthy and Compromised Sensors with Fix Compromise Ratio

The network behavior with fix compromised ratio, i.e. 50 numbers of compromised sensors in each round are shown in Figure 11. The figure represents total number of sick, healthy and compromised sensors in each round where X-axis represents the number of rounds and Y-axis represents total number of healthy, sick or compromised sensor nodes. Compromised sensors are represented by red line in the graph. Healthy and sick sensors are represented by green and blue lines respectively. In case of compromised sensors, 330 sensor nodes are compromised till round number 22. In case of healthy sensors, the number of sensor nodes rises to 400 between 0-3 rounds and then falls back to 20 nodes at 65th round which clearly represents the network lifetime of healthy sensors in the graph. In case of sick sensors, the number of sensor nodes rises to 75 between 0-5 rounds and goes on varying (rise and fall) after fifth round. The number of sick sensors falls to 0 nodes after 78thround.

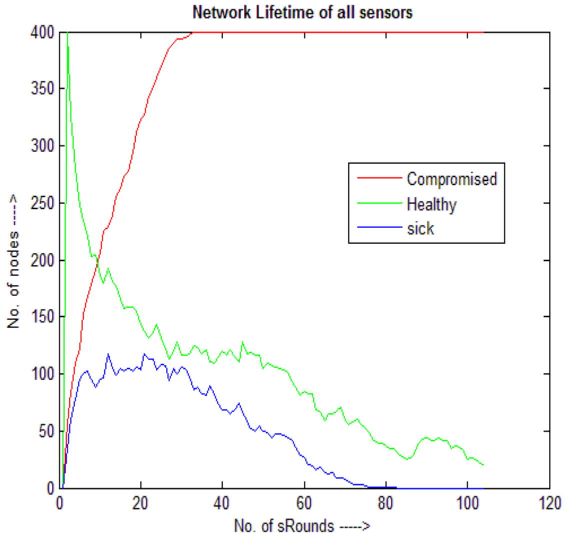

Figure 11: Network Lifetime With Static Number of Compromised Sensor Nodes

VII. Conclusion

In this research paper, we presented a 'NDSP: Neighbor assisted Distributed Self-healing Protocol'for compromised node recovery in wireless sensor networks.The scheme is used in compromised wireless sensor networks to recover most of the compromised sensor nodes from compromised and sick state. The recovery scheme presented in NDSP recoversa single sensor node by a healthy or sick sensors but our future work for recovery is multi node recovery where a healthy sensor gives recovery to its more than one random neighbor in the network. The future model of NDSP is also based on the concept of centralized intrusion detection.

References

[1] . V. Naik, S. Bapat, and M. Gouda, Whisper, “local secret maintenance in sensor networks," in Principles of Dependable System, 2003.

[2] . Hu, F.and Sharma, N. 2005,”Security considerations in ad hoc sensor networks, “Ad Hoc Networks (Elsevier) 3, 1, 69–89.

[3] . R. D. Pietro, C. Soriente, A. Spognardi, and G. Tsudik,” data survival in unattended sensor networks," in IEEE PERCOM'08, March 2008.

[4] . G. Itkis and L. Reyzin, Sibir, “signer-base intrusion-resilient signatures." in Crypto'02, 2003.

[5] . Y. Dodis, J. Katz, S. Xu, and M. Yung, “Key-insulated public key cryptosystems," in Eurocrypt'02, May 2002.

[6] . MA, D. and Tsudik, G. “DISH: Distributed self-healing in unattended wireless sensor networks”. In 10th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS’08). 47–62, 2008.

[7] . M.N.Huhns, V.T. Holderfield, R.L. Gutierrez, “Robust software via agent-based redundancy”,AAMAS'03, 2003

[8] . Blundo, C, D, Arco, P. and Listo, “A New Self-healing Key Distribution Scheme. Proc. Eighth IEEE Int. Symp. Computer Commun, vol. 2, pp. 803–808, 2003.

[9] . Canetti, R. and Herzberg, “Maintaining security in the presence of transient faults In 14th Annual IACR Crypto Conference (Crypto’94). 425–438

[10] . B. Parno, A. Perrig and V. Gligor, “Distributed detection of node replication attacks in sensor networks” In IEEE Symposium on Security and Privacy, 2008.