IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Replica Node Attacks Detection in Mobile Sensor Networks Using Efficient and Distributed Scheme

R. Kalichamy[*1], S.Arulselvi[2], Dr.T. Kirankumar[3]

[*1,2,3] Bharath University, Chennai, India

kalicsamy@gmail.com

Abstract

The most wireless sensor networks are composed of unshielded sensor nodes. An adversary can easily attack, analyze and clone the unshielded sensor nodes and create replicas and insert them in the networks. This gives the adversary to carry on large class of insidious attacks like disrupting communication, subverting data aggregation, eavesdropping etc… in this research we resist against node replication attacks in mobile sensor networks. In this paper, we propose a new protocol to detect the replicas in mobile WSNs. in this protocol, efficient and distributed scheme and sequential probability ratio test are used to security that the replica nodes enters in to the monitoring area containing number of nodes under consideration the path travelled by the replica nodes are monitored and other nodes present I the area are prevented from attack using the efficient and distributed scheme.

Keywords: Replica detection, sequential analysis, mobile sensor networks, security.

Introduction

Wireless Sensor Networks are networks made up of tiny embedded devices. Each device is capable of sensing, processing and communicating. The networks can be made up of hundreds or thousands of devices that work together to communicate the information that they obtain. Each node is responsible for covering a particular area by sensing. The node then sends the results to a sink node that collects the data. Nodes are used to relay the information, allowing the message to use multiple hops to reach the sink node. In order to process the information effectively, the network must have good coverage and the sink node must have good connectivity. Wireless Sensor Networks are frequently ad hoc, meaning that nodes can be added at any time and configure themselves to be part of the existing network. Any node can act as a relay to pass messages along in the network. This works well for applications that add new sensors to replace those that have used up their battery life, or need to add more nodes for better coverage.

Wireless Sensor Network (WSN) is a kind of Ad hoc network, and is often infrastructure independent. Large amount of cheap motes construct the network by collaborating with each other, and the data sensed by each mote congregate to the base station. The transmission power for each mote is often low to avoid interfacing with each other. Thus the range of communication or range of connectivity is limited to some extent. In a quite large WSN, the readings from each mote may have to arrive at the base station via several steps. In sensor network, finding and maintaining a high efficient multi-hop routing algorithm is very important to guarantee the high reliability and low energy consuming. WSN is a kind of data centrical network.

Specialized Sensor nodes

Specialized sensors are used in many applications, including asset tracking. These sensors are very small and must operate for a long time on a battery supply. Thousands of sensors are usually involved in this type of application.

Generic Sensor nodes

Generic sensors are used in many applications, including security applications such as motion detection in doors and windows. These sensors are very small and must operate for a long time on a battery supply. Not much data processing is required for this application and low communication rates are required. Hundreds of sensors are usually involved in this type of application.

High-bandwidth Sensor nodes

High-bandwidth sensors are used for video, acoustic and chemical applications that require more resources for communications and computations. Battery power is often not enough for these applications. In order to operate for the long term, they must be plugged into electrical power.

Gateway nodes

Gateway sensors are used to link the wireless sensor network to the internet. They contain more memory and data logging capabilities. The gateway node is intended to be capable of generic processing possessing the flexibility to connect to the network with a variety of interfaces. The Star gate device is an example of a Gateway device developed by Intel. It contains several megabytes of RAM and persistent storage measured in Gigabytes. The Star gate has USB, JTAG, RS232, Compact flash, Ethernet, and a PCI interface.

Makeup of a wireless sensor network application

It is common for applications to combine different types of sensors in their network. A security system may use motion sensors, cameras and a gateway interface to collect and process the data collected by the wireless sensor network.

Duck Island is a habitat monitoring system that uses different types of wireless sensors.

Security Requirements of Wireless Sensor Networks

Following are the basic requirements for provisioning security in wireless sensor networks

- **Data Confidentiality**: Certain readings observed and generated by a sensor node can be classified as sensitive data and therefore must be protected from eavesdropping by rogue sensors and/or intruders. A standard approach to protect the confidentiality of sensory data is to encrypt it using a cryptographic key. The resource constrained nature of sensor nodes makes it a challenge to generate, store, and use cryptographic keys of any kind, asymmetric or symmetric
- **Data Authentication**: The authentication of messages exchanged between the sensor nodes is necessary to ensure protection against hoax messages that may be injected into the network by an adversary. Such an attack may have catastrophic consequences considering the mission critical nature of sensor applications.
- **Data Integrity**: Data integrity ensures that the received data is not modified or tampered with on its way from the sender to the receiver. For instance, in a bush sensing network, an adversary may attempt to alter sensor readings to trigger an alarm which otherwise would have been initiated only for actual emergency scenarios.
- **Data Freshness**: An old set of messages i.e. sensor readings may be replayed by an adversary to mock a potential emergency in a normal situation. Therefore, it is essential to ensure the freshness of all data exchanged within the sensor networks.
- **Data Availability**: Sensor nodes deployed in un-trusted environments for carrying out critical operations must be able to survive the expected battery lifetimes. Premature exhaustion of the limited battery lives of sensor nodes may have a catastrophic event on operations of the entire network. An adversary may attempt to launch an attack against valuable resources in the sensor network to exhaust their energy resources, and cause the network to be disabled from continuing to operate and carry out its designated tasks pertaining to environment sensing and detection. Such an attack leads to denied access for the base station to sensory data, that may be crucial for critical applications. Therefore, these types of attacks are referred to as Denial of Service (DoS) attacks. The DoS attack may or may not be launched from a single end point of the network, wherein a single compromised node or a node belonging to an adversary, repeatedly sends hoax requests to a legitimate target sensor node with the intent of exhausting its limited energy resources. On the contrary, an intelligent attacker may launch the attack from multiple ends of the network by compromising enough available resources to ensure high success in the attack process. The distributed nature of this attack is called a Distributed Denial of Service (DDoS) attack.Several popular schemes such as the Standard Network Encryption Protocol (SNEP) and TESLA (Micro Timed Efficient Stream Loss-tolerant Authentication) have been proposed in the literature to satisfy the data authentication, freshness, and confidentiality requirements for provisioning security in wireless sensor networks. However, very little research has been done to address the issue of availability of sensor nodes under an attack.

Background

We consider a two-dimensional mobile sensor network where sensor nodes freely roam throughout the network. We assume that every mobile sensor node’s movement is physically limited by the system-configured maximum speed, Vmax. We also assume that all direct communication links between sensor nodes are bidirectional. This communication model is common in the current generation of sensor networks. We assume that every mobile sensor node is capable of obtaining its location information and also verifying the locations of its neighboring nodes. This can be implemented by employing secure localization methods. We assume that the clocks of all nodes are loosely synchronized. This can be achieved with the help of secure time synchronization protocols . We also assume that the nodes in the mobile sensor network communicate with a base station. The base station may be static or mobile, although we focus on a static base station for our simulations, as long as the nodes have a way to communicate reliably to the base station on a regular basis

- **Potential Problems With Wireless Sensor Networks**

We define a mobile replica node u0 as a node having the same ID and secret keying materials as a mobile node u. An adversary creates replica node u0 as follows: He first compromises node u and extracts all secret keying materials from it. Then he prepares a new node u0, sets the ID of u0 to the same as u, and loads u's secret keying materials into u0. There may be multiple replicas of u, e.g., u01; u02; . . . , and there may be multiple compromised and replicated nodes. Our goal is to detect the fact that both u and u0 (or u01; u02; . . .) operate as separate entities with the same identity and keys.

- **Definition of Replica**

In wireless networks, a particularly dangerous attack is the replica node attack , in which the adversary takes thesecret keying materials from a compromised node, generatesa large number of attacker-controlled replicas that share the compromised node's keying materials and ID, and then spreads these replicas throughout the network. With a single captured node, the adversary can create as many replica nodes as he has the hardware to generate. Note that replica nodes need not be identical robots, a group of static nodes can mimic the movement of a robot and other mobile nodes or even humans with handheld devices could be used.

- **Prevention of replica attacks**

A straightforward solution to stop replica node attacks is to prevent the adversary from extracting secret key materials from mobile nodes by equipping them with tamper-resistant hardware. Several software-based replica node detection schemes have been proposed for static sensor networks. In this paper, we propose a novel mobile replica detection scheme based on the Sequential Probability Ratio Test (SPRT)

- **Attacker models**

We assume that an adversary may compromise and fully control a subset of the sensor nodes, enabling him to mount various kinds of attacks. For instance, he can inject false data packets into the network and disrupt local control protocols such as localization, time synchronization, and route discovery process. Furthermore, he can launch denial-of-service attacks by jamming the signals from benign nodes. However, we place some limits on the ability of the adversary to compromise nodes. We note that if the adversary can compromise a major fraction nodes of the network, he will not need nor benefit much from the deployment of replicas. To amplify his effectiveness, the adversary can also launch a replica node attack, which is the subject of our investigation. We assume that the adversary can produce

many replica nodes and that they will be accepted as a legitimate part of the network. We also assume that the attacker attempts to employ as many replicas of one or more compromised sensor nodes in the network as will be

effective for his attacks.

- **Work of attacker**

The attacker can allow his replica nodes to randomly move or he could move his replica nodes in different patterns in an attempt to frustrate our proposed scheme. We also assume that the base station is a trusted entity. This is a reasonable assumption in mobile sensor networks, because the network operator collects all sensor data and can typically control the nodes operation through the base station. Thus, the basic mission of the sensor network is already completely undermined if the base station is compromised.

Materials and Methods

Existing system

In previous sector we use only fixed sensor network .In existing we use sequential testing for Replica node detection process. Only small number of location claim for detection of replica node.

Proposed system

In this research are effective and efficient in terms of the communication/computation/storage overheads. In this research more number of location to be search in mobile sensor network. The mobile sensor network has moveable nodes so the replica node in moving . In propose we use secured key for digital signature ,it based on public key scheme.

Efficient and Distributed Scheme

- **Description**

The replica node that enters the area under consideration is not only viewed. The path that replica moves is also monitored and the damage of other nodes are prevented as replica moves around the path.

We present a practical interactive conference key distribution system based on public keys, which is `proven' secure provided the Diffie-Hellman problem is intractable. The system authenticates the users and allows them to compute their own conference key. A certain number of interactions is required, but the number of rounds is independent of the number of conference users. All users involved perform the same amount of computation and communication. Our technique for authentication can be extended and used as the basis for an authentication scheme which is 'proven' secure against any type of attack, provided the discrete logarithm problem is intractable.

Traditional Key Management Approaches

Some general key distribution and management approaches are not suitable for wireless sensor networks. First, trivially storing in each node a pair wise key for every other node poses a high memory requirement unaffordable for sensor nodes. Second, online key distribution and management offered by the base station is inefficient for wireless sensor networks due to high communication overhead. Third, public-key algorithms such as RSA, Diffie-Hellman, and Elliptic Curve Cryptography (ECC) are too expensive to current sensor nodes for high energy consumption and computation overhead. Experimental results from existing research show that the execution time of public key- based operations, such as encryption and decryption, is of the order of seconds or even 10seconds . Moreover, wireless sensor networks may not be able to provide the desired public-key infrastructure (PKI) for key distribution. We have to either distribute public keys into nodes through the base station online, which may cause high communication overhead, or predistribute public keys into nodes offline, which may need some scheme like what we present in this paper to improve its efficiency.

Related Work

Drawbacks of traditional key management approaches

The key agreement problem is a part of the key management problem, which has been widely studied in general network environments. There are three types of general key agreement schemes: trusted-server scheme, self-enforcing scheme, and key pre-distribution scheme.The trusted-server scheme depends on a trusted server for key agreement between nodes, e.g., Kerberos. This type of scheme is not suitable for sensor networks because there is usually no trusted infrastructure in sensor networks. The self-enforcing scheme depends on asymmetric cryptography, such as key agreement using public key certificates. However, limited computation and energy resources of sensor nodes often make it undesirable to use public key algorithms, such as Diffie-Hellman key agreement or RSA as pointed out. The third type of key agreement scheme is key pre-distribution, where key information is distributed among all sensor nodes prior to deployment.

Key management scheme for distributed networks

Eschenauer and Gligor proposed the basic scheme by predistributing random keys into nodes. The drawback is that one pair wise key may be shared by multiple links.

Random key predistribution for sensor networks

Chan and Perrig presented two schemes. In their q composite scheme, multiple keys are required to establish a secure link, which makes a trade-off between connectivity and security. In their random pair wise-key scheme, a unique pair wise key is assigned to each node and every one of a random set. This scheme provides high security but poses an upper bound on network size.

Pairwise key distribution scheme for wireless sensor networks

Du proposed the pair wise key pre-distribution scheme based on both the basic scheme and Blom's scheme, from which it inherits the threshold property.

Key pre-distribution with deployment knowledge in static sensor networks

Du and Liu and Ning, independently proposed to utilized deployment knowledge to improve the performance of key establishment. Our scheme outperforms Du's deployment knowledge scheme in terms of connectivity and security. Liu and Ning's polynomial-based key predistribution scheme also has the threshold property for the use of bivariate polynomials, which is a special form of Blom's scheme.

A Probabilistic approach for secure communication in wireless sensor networks

Zhu presented LEAP by introducing a weaker model,which assumes that there exists a short time interval within which nodes can establish pair wise keys securely. However, this time interval is often very hard to estimate accurately. Once it is overestimated, all links may be compromised. Probabilistic Key Sharing discussed most of the proposed symmetric key cryptography protocols for establishing a pair wise shared key between two nodes make use of an on-line key server. Mitchell and Piper proposed a solution based on probabilistic key sharing that does not depend on such an on-line server. However, the storage complexity imposed on

each participant in their scheme seems to be unaffordable in the context of ad hoc networks. The probabilistic keying scheme in our protocol is similar to schemes that have been used by other researchers. Eschenauer and Gligor introduced a key management scheme based on probabilistic key sharing for distributed sensor networks (DSN) with central key servers (e.g., base stations). Chan extended this scheme by presenting three new mechanisms for key establishment in sensor networks based on the framework of probabilistic key predeployment, including a mechanism for pair wise shared key establishment called multipath key reinforcement. Our work differs from the previous ones in several aspects. First, in our scheme, a node can deduce the set of keys it shares with any other node (which may be an empty set) only based on the latter's to identity. In contrast, the approaches require each node exchange the ids of the keys it possesses with its neighbors. Keys are allocated to each node using a probabilistic scheme that enables every pair of nodes to share one or more keys with certain probability. The keys directly shared between any two nodes can thus be used to encrypt messages exchanged between them. Even if two nodes do not share any keys directly, our probabilistic key sharing scheme enables them to communicate securely using logical paths obtained via a logical path
discovery process.

Comparison of different key management approaches for wireless sensor networks

WSNS are ideal candidates for applications such as military target tracking, home security monitoring, and scientific exploration in dangerous environments. Typically, a sensor network consists of a potentially large number of resource constrained sensors, which are mainly used to collect data (e.g. temperature) from the environment, and a few control nodes, which may have more resources and may be used to control the sensors
and/or connect the network to the outside world (e.g. a central data processing server). Sensors usually communicate with each other through
wireless communication channels. Sensor networks may be deployed in hostile environments, especially in military applications.

In such situations, the sensors may be captured, and the data/control packets may be intercepted and/or modified. Therefore, security services such as authentication and encryption are essential to maintain the network operations. However, due to the resource constraints on the sensors, many security mechanisms such as public key cryptography are not feasible in sensor networks. Indeed, providing security services in sensor networks is by no means a trivial problem; it has received a lot of attention recently. A fundamental security service is the establishment of a symmetric, pairwise key shared between two sensors, which is the basis of other security services such as encryption and authentication. Several key pre-distribution techniques have been developed recently to address this problem. Eschenauer and Gligor proposed the basic probabilistic key pre-distribution, in which each sensor is assigned a random subset of keys from a key pool before the deployment of the network. By doing this, two sensors can have a certain probability to share at least one key. Chan developed the q-composite key pre-distribution and the random pair wise keys schemes.

The q-composite key pre-distribution scheme is based on the basic probabilistic scheme, but it requires two sensors share at least q pre-distributed keys to establish a pair wise key. The random pair wise keys scheme pre-distributes random pair wise keys between a particular sensor and a random subset of other sensors, and has the property that compromised sensors do not lead to the compromise of pair wise keys shared between non compromised sensors. However, these approaches still have some limitations. For the basic probabilistic and the q-composite key predistribution, a small number of compromised sensors may reveal a large fraction of pair wise keys shared between non-compromised sensors. Though the random pair wise keys scheme provides perfect security against node captures, the maximum supported network size is strictly limited by the storage capacity for pair wise keys and the desired probability to share a key between two sensors. Liu and Ning developed a framework to pre-distribute pair wise keys using bi-variate polynomials and proposed two efficient instantiations, a random subset assignment scheme and a grid-based key pre-distribution scheme, to establish pair wise keys in sensor networks. Sensor networks usually consist of a large number of ultra-small autonomous devices. Each device, called a sensor node, is battery powered and equipped with integrated sensors, data processing capabilities, and short-range radio communications.

In typical application scenarios, sensor nodes are spread randomly over the deployment region under scrutiny and collect sensor data. Examples of sensor network projects include Smart Dust and WINS.Sensor networks are being deployed for a wide variety of applications, including military sensing and tracking, environment monitoring, patient monitoring and tracking, smart environments, etc. When sensor networks are deployed in a hostile environment, security becomes extremely important, as they are prone to different types of malicious attacks. This key agreement problem is a part of the key management problem, which has been widely studied in general network environments. There are three types of general key agreement schemes: trusted-server scheme, self-enforcing scheme, and key pre-distribution scheme.

The trusted-server scheme depends on a trusted server for key agreement between nodes, e.g., Kerberos. This type of scheme is not suitable for sensor networks because there is usually no trusted infrastructure in sensor networks. The self-enforcing scheme depends on asymmetric cryptography, such as key agreement using public key certificates. However, limited computation and energy resources of sensor nodes often make it undesirable to use public key algorithms, such as Diffie-Hellman key agreement or RSA, as pointed out. The third type of key agreement scheme is key pre-distribution, where key information is distributed among all sensor nodes prior to deployment. If we know which nodes are more likely to stay in the same neighborhood before deployment, keys can be decided a priori. However, because of the randomness of the deployment, knowing the set of neighbors deterministically might not be feasible. There exist a number of key pre-distribution schemes. A naive solution is to let all the nodes carry a master secret key. Any pair of nodes can use this global master secret key to achieve key agreement and obtain a new pair wise key.

This scheme does not exhibit desirable network resilience. If one node is compromised, the security of the entire sensor network will be compromised. Some existing studies suggest storing the master key in tamper-resistant hardware to reduce the risk, but this increases the cost and energy consumption of each sensor.

Based on the Eschenauer-Gligor scheme Chan, Perrig and Song proposed a q-composite random key pre-distribution scheme. The difference between this scheme and the Eschenauer-Gligor scheme is that q common keys

(q_1), instead of just a single one, are needed to establish secure communications between a pair of nodes. It is shown that, by increasing the value of q, network resilience against node capture is improved, i.e., an attacker has to compromise many more nodes to achieve a high probability of compromised communication. Du, Deng, Han, and Varshney proposed a new key pre-distribution scheme, which substantially improves the resilience of the network compared to the existing schemes. This scheme exhibits a nice threshold property. when the number of compromised nodes is less than the threshold, the probability that any nodes other than these compromised nodes are affected is close to zero. This desirable property lowers the initial payoff of smaller scale network breaches to an adversary, and makes it necessary for the adversary to attack a significant proportion of the network. A similar method is also developed by Liu and Ning.

A survey on key distribution and authentication for resource-starved devices in mobile environments is given. The majority of these approaches rely on asymmetric cryptography, which is not a feasible solution for sensor networks. Several other methods based on asymmetric cryptography are also proposed. Zhou and Hass propose a secure ad hoc network using secret sharing and threshold cryptography. Kong also proposes localized public-key infrastructure mechanisms, based on secret sharing schemes. Distributed sensor networks have received a lot of attention recently due to their wide application in military as well as civilian operations. Example applications include target tracking, scientific exploration, and monitoring of nuclear power plants. Sensor nodes are typically low-cost, battery powered, and highly resource constrained, and usually collaborates with each other to accomplish their tasks. Eschenauer and Gligor proposed a probabilistic key pre-distribution scheme recently for pair wise key establishment.

The main idea was to let each sensor node randomly pick a set of keys from a key pool before deployment so any two sensor nodes have a certain probability of sharing at least one common key. Chan further extended this idea and developed two key pre-distribution techniques: q-composite key pre distribution and random pair wise keys scheme. The q-composite key predistribution also uses a key pool but requires two sensors compute a pair wise key from atleast q predistributed keys they share. Some general key distribution and management approaches are not suitable for wireless sensor networks. First, trivially storing in each node a pair wise key for every other node poses a high memory requirement unaffordable for sensor nodes. Second, online key distribution and management offered by the base station is inefficient for wireless sensor networks due to high communication overhead. Third, public-key algorithms such as RSA, Diffie- Hellman, and Elliptic Curve Cryptography (ECC) are too expensive to current sensor nodes for high energy consumption and computation overhead.

Experimental results from existing research show that the execution time of public key- based operations, such as encryption and decryption, is of the order of seconds or even 10 seconds. Moreover, wireless sensor networks may not be able to provide the desired public-key infrastructure (PKI) for key distribution. We have to either distribute public keys into nodes through the base station online, which may cause high communication overhead, or predistribute public keys into nodes offline, which may need some scheme like what we present in this project to improve its efficiency.

SPRT(Sequential Probability Ratio Test)

This section presents the details of our technique to detect replica node attacks in mobile sensor networks. In static sensor networks, a sensor node is regarded as being replicated if it is placed in more than one location. If nodes are moving around in network, however, this technique does not work, because a benign mobile node would be treated as a replica due to its continuous change in location. Hence, we must use some other technique to detect replica nodes in mobile sensor networks. Fortunately, mobility provides us with a clue to help resolve the mobile replica detection problem. Specifically, a benign mobile sensor node should never move faster than the system configured maximum speed, Vmax.

SPRT Functions

As a result, a benign mobile sensor node's measured speed will appear to be at most Vmax as long as we employ a speed measurement system with a low rate of error. On the other hand, replica nodes will appear to move much faster than benign nodes and thus their measured speeds will likely be over Vmax because they need to be at two (or more) different places at once. Accordingly, if the mobile node's measured speed exceeds Vmax, it is then highly likely that at least two nodes with the same identity are present in the network. We propose a mobile replica detection scheme by leveraging this intuition. Our scheme is based on the Sequential Probability Ratio Test which is a statistical decision process.

Replica Node Detection Using SPRT

The SPRT can be thought of as one dimensional random walk with the lower and upper limits. Before the random walk starts, null and alternate hypotheses are defined in such a way that the null hypothesis is associated with the lower limit while the alternate one is associated with the upper limit. A random walk starts from a point between two limits and moves toward the lower or upper limit in accordance with each observation. If the walk reaches (or exceeds) the lower or upper limit, it terminates and the null or alternate hypothesis is selected, respectively. We believe that the SPRT is well suited for tackling the mobile replica detection problem since we can construct a random walk with two limits in such a way that each walk is determined by the observed speed of a mobile node. The lower and upper limits can be configured to be associated with speeds less than and in excess of Vmax, respectively. We apply the SPRT to the mobile replica detection problem as follows: Each time a mobile sensor node moves to a new location, each of its neighbors asks for a signed claim containing its location and time information and decides probabilistically whether to forward the received claim to the base station. The base station computes the speed from every two consecutive claims of a mobile node and performs the SPRT by considering speed as an observed sample. Each time the mobile node's speed exceeds (respectively, remains below) Vmax, it will expedite the random walk to hit or cross the upper (respectively, lower) limit and thus lead to the base station accepting the alternate (respectively, null) hypothesis that the mobile node has been (respectively, not been) replicated. Once the base station decides that a mobile node has been replicated, it revokes the replica nodes from the network. Let us first describe the detection scheme and then analyze its security and performance.

Security

AODV defines no special security mechanisms. So an impersonation attack can easily be done. Or even simpler, a misbehaving node is planted in the network. There are a few proposals how to solve this problem, but it is very hard because AODV is not a source based routing protocol and such a solution would introduce a tremendous overhead.

Implementations

There are two types of different implementations, user space daemons and kernel modules. The first implementation requires to maintain an own routing table and was first implemented in the Mad hoc Implementation by Fredrik Lilieblad, Oskar Mattsson, Petra Nylund, Dan Ouchterlony, Anders Roxenhag running on a Linux 2.2 kernel but does not supports multicast. A bit later the University of Uppsala published user space daemon implementation called AODV-UU , which runs fairly well on Linux with a 2.4 kernel. Today many different implementations of AODV exist.

Classifier: analyses the packet and hands it over to the correct successor

Routing Agent object (Rt agent): implements the used routing protocol as AODV or DSDV

Link Layer object (LL): supports data link protocols and mechanisms such as packet fragmentation and reassembly, queuing, link-level retransmissions, piggybacking etc.

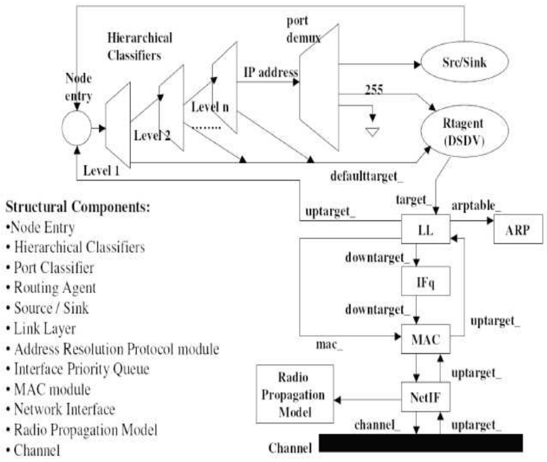


Figure1. Schematic of Hierarchical Wireless Node

Address Resolution Protocol module (ARP): finds and resolves the IP address of the next – hop/node into the correct MAC address. The MAC destination address is set into the MAC header of the packet.

Interface Priority Queue (IFQ): gives a priority to routing protocol packets by running a filter over the packets and removing those with a specified destination address.

Medium Access Protocol module (MAC): provides multiple functionalities such as carrier sense, collision detection and avoidance etc.,

Network Interface (Net IF): is an interface for a mobile node to access the channel. Each packet leaving the Net IF is stamped with the meta-data in its header and the information of the transmitting interface such as transmission power, wavelength etc. to be used by the propagation model of the receiving Net IF.

Radio Propagation Model: uses Free-space attenuation (1/r2) at near distances and an approximation to two rays ground (1/r4) at far distances by default. It decides whether a mobile node with a given distance, power of transmission and wavelength can receive a packet. By default, it implements an Omni directional antenna, which has unit gain for all directions.

Traffic Generator

Till now we discussed about the event scheduler of ns-2 and the raw architecture of a mobile node. But for network simulation we also need some load on the net. The data packets are always injected over an agent as TCP or UDP, which is aggregated to a node. For emission, the agent sends the packet to the entry point of its node. For reception, the agent receives the packet over the nodes classifiers. But the agent is not jet the source of the data. A Process supplies the data or in Figure 14 more specific a traffic generator. For the simulations a CBR (Constant Bit Rate) traffic generator with an UDP agent was used. With this configuration it is possible to study the real performance of the ad hoc network without any undesired and unknown influences of other protocols. This comes very close to real world behaviour.

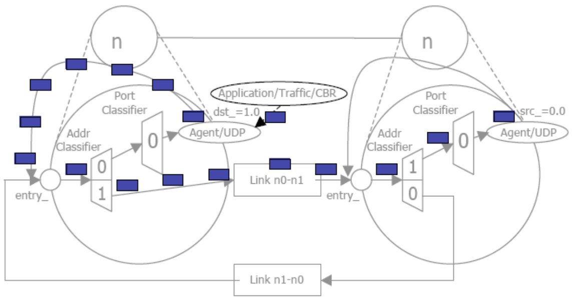


Figure.2. Schematic of Traffic Generator and Packet Flow

Packet Header

I would like to give also an overview of the packet header stack used by a typical packet in the simulation. The common header (hdr_cmn) takes care of the basic information, the simulator needs for a packet, as type, unique id, size or timestamp. The headers below correspond to the used protocols on the corresponding layers.

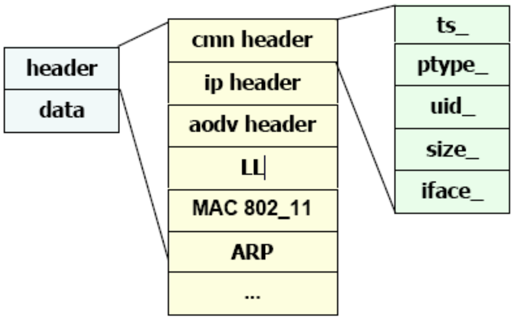
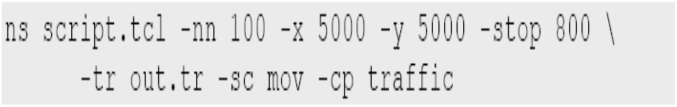


Figure 3.Packet header stack

USAGE OF NS-2

An ns-2 simulation is controlled by a TCL scripts, which contains all necessary parameters and configurations. Additionally the ´opt´ parameters within the TCL script can be modified from the command line as shown below.



```
ns script.tcl -nn 100 -x 5000 -y 5000 -stop 800 \
     -tr out.tr -sc mov -cp traffic
```

The TCL script specifies the path of movement and connection files to be loaded as well as the path to the trace files, usually a nam and a tr file, which are the product of a simulation.

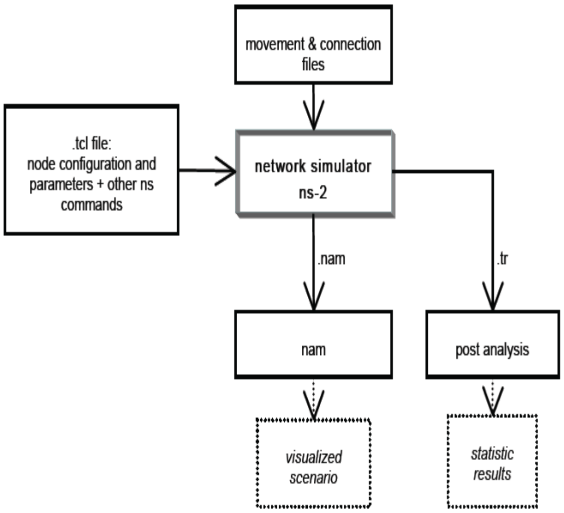

movement & connection files
.tcl file: node configuration and parameters + other ns commands
network simulator ns-2
.nam
.tr
nam
post analysis
visualized scenario
statistic results

Figure 4. NS-2 Usage Diagram

Results and Conclusion

Simulation

As the simulation time increases the probability density Ratio has better results for the the mobile replica node when compared to the normal mode which is a stationary mode

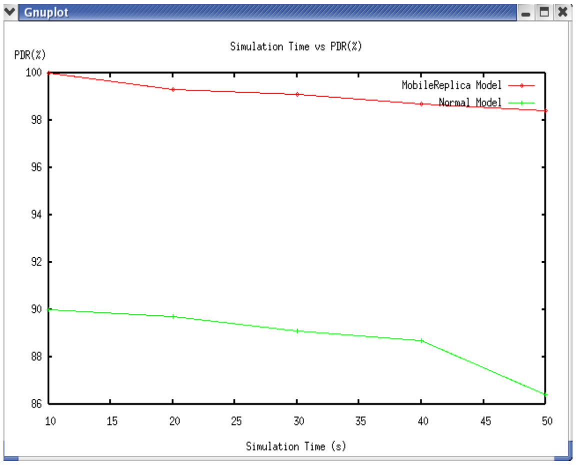


Fig.5. Simulation time Vs PDR

Simulation Time Vs Dropped Packets

As simulation time increases the Dropped Packet level decreases for the mobile replica node when compared to the normal model

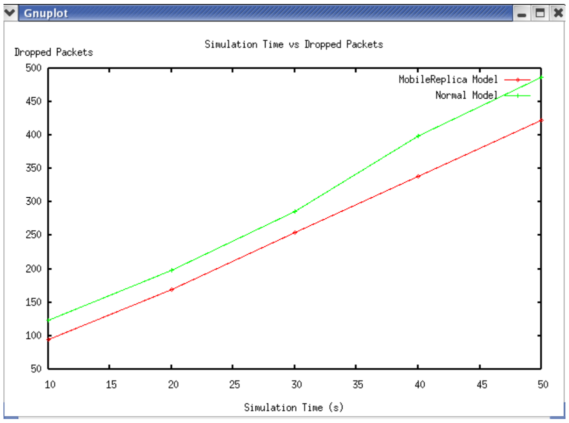


Fig 6. Simulation time Vs Dropped packets

Simulation Time VS Received Packets

As the Simulation time increases the rate of received packets increases for the mobile Replica node as the error have been reduced as shown above. This rate is better than the rate of normal model.

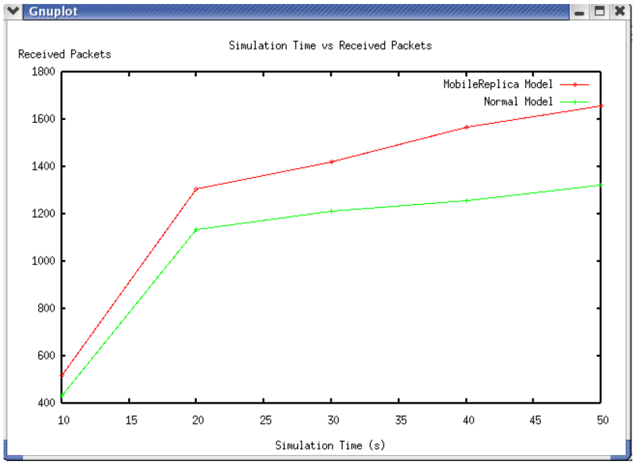


Fig 7.Simulation time Vs Received Packets

Speed Vs Average Number OF Claims

As the simulation speed increases the average no of claims decreases for the mobile replica node. This result is better than the previous case for normal model.

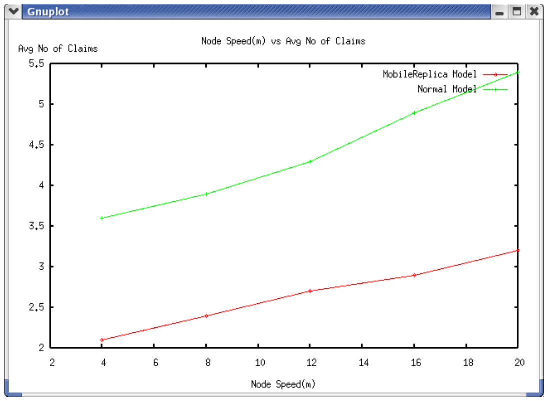


Fig 8.Speed Vs Avg no of claims

Number OF Claims Vs Probability

As the number of claims increases the probability for error decreases for the mobile replica node. This result is better than the previous case for normal model

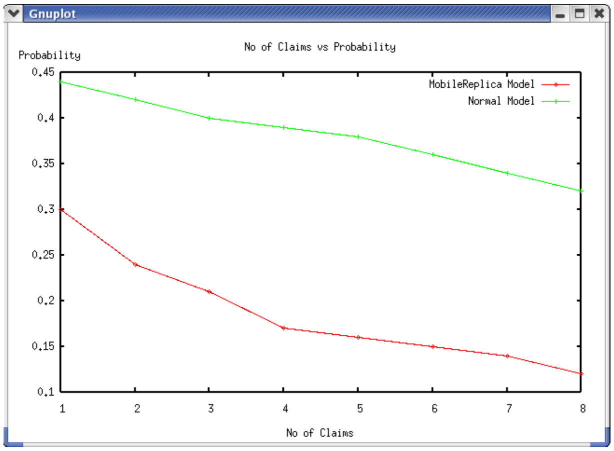


Fig 9. Number of Claims vs Probability

Number of Monitoring Agent Vs Average Number of Claims

Number of monitoring agent increases then obviously Number of claims decreases for the mobile replica node .this is safer when compare to the normal model as show in figure.

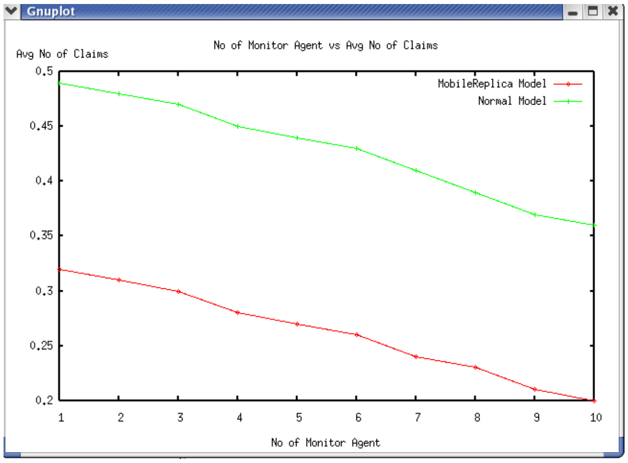


Fig 10. Number of monitoring agent Vs Average Number of claims

Distance Vs Average Number of Claims

As the distance increases number of claims decreases in the replica node when compared to normal mode.

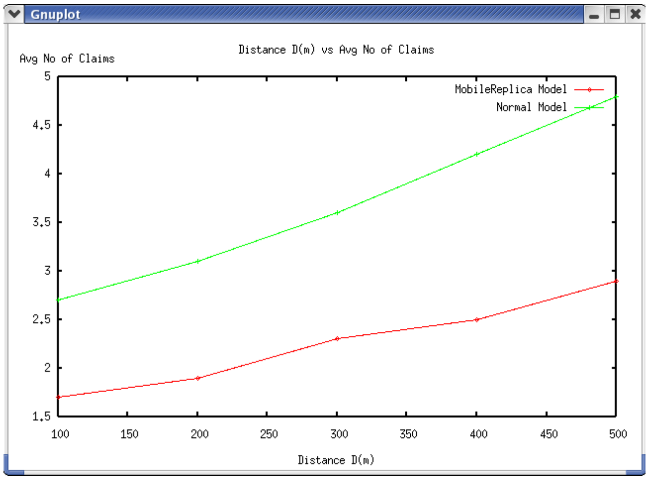


Fig11.Distance Vs Average number of claims

Future Work

It can be noted from the fig.12 that the replica nodes are monitored thoroughly and other nodes are prevented from attack.

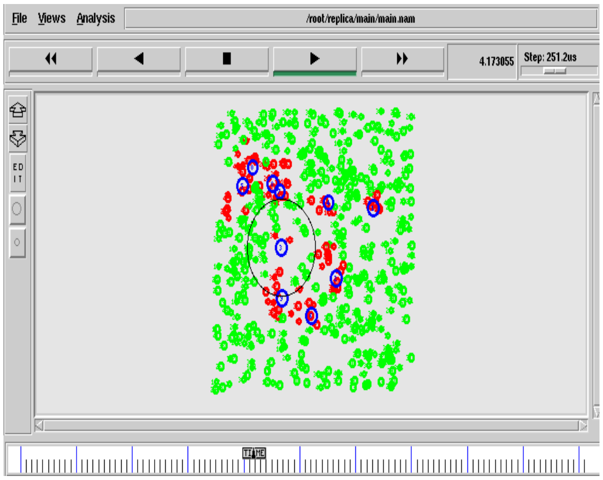


Fig 12. Repica node detection using Efficient and Distributed Scheme

Hence all nodes all away from attacks from the replica nodes.Here we have considered only limited number of nodes .Number of nodes can be increased only for a particular level.

In future we are trying to implement the technique with infinite number of nodes and increase the energy levels with higher accuracy and lower Bit Error Rate (BER).

Acknowledgement

Thanks to Dr. T.Kiran Kumar, who comforted me in various replica node detection and prevention issues simulating this paper and also Mrs. s. Arulselvi for her help in seeding me regarding the practical implementation and its pros and cons. The insightful and detailed feedback and suggestion by them which improved content and presentation of this paper.

References

[1] Liu and P. Ning, "TinyECC: A Configurable Library for Elliptic Curve Cryptography in Wireless Sensor Networks," Proc. Seventh IEEE Int'l Symp. Information Processing in Sensor Networks (IPSN), pp. 245-256, Apr. 2008.

[2] C.-M. Yu, C.-S. Lu, and S.-Y. Kuo, "Efficient and Distributed Detection of Node Replication Attacks in Mobile Sensor Networks," Proc. IEEE Vehicular Technology Conf. Fall (VTC Fall), Sept. 2009.

[3] J. Ho, D. Liu, M. Wright, and S.K. Das, "Distributed Detection of Replicas with Deployment Knowledge in Wireless Sensor Networks," Ad Hoc Networks, vol. 7, no. 8, pp. 1476-1488, Nov. 2009.

[4] K. Xing, F. Liu, X. Cheng, and H.C. Du, "Real-Time Detection of Clone Attacks in Wireless Sensor Networks," Proc. IEEE Int'l Conf. Distributed Computing Systems (ICDCS), pp. 3-10, June 2008.

[5] Wireless communication principles and practice by Theodore Rappaport prentice hall,1996.