Forestalling Meticulous Jam Attacks Using Packet-Hiding Techniques

P. Srimanchari[1], V.Perumal[2], Dr. M. Saroja Ph.D[3], Dr. M. Venkatachalam,Ph.D[4]
1. Assistant Professor, 2. M.Phil Research Scholar
Department of Computer Science
3. Associate Professor, 4. Professor & HOD
Department of Electronics
Erode Arts and science college, Erode- 638009, India
Email: srimanchari@gmail.com

ABSTRACT

The open nature of the wireless medium leaves it liable to intentional interference attacks, generally said as jam. This intentional interference with wireless transmissions is used as a launch pad for mounting Denial-of-Service attacks on wireless networks. Typically, jam has been self-addressed beneath associate external threat model. However, adversaries with internal information of protocol specifications and network secrets will launch low-effort jam attacks that are troublesome to notice and counter. during this work, we have a tendency to address the matter of jamming attacks in wireless networks. In these attacks, the resister is active just for a brief amount of your time, by selection targeting messages of high importance. In our work two offender nodes (node that creates jamming) and introduce one new node i.e sender node. The new node(jammer node) is at intervals the twenty five nodes. Victimization that new sender node we have to eliminate the offender nodes absolutely. We have a tendency to conclude that however jam happens within the network and approach of elimination of the offender nodes victimization new sender node. We propose mistrial approach for avoid flooding packets in jammer network. We conclude the performance between the mistrial and damping approach for avoid jamming packets We have a tendency to illustrate the benefits of {selective jam|spot-jamming|jamming|electronic jamming|jam} in terms of network performance degradation and resister effort by to beat the sender in network with the assistance of recent jamming node. We illustrate the benefits of jam|spot-jamming|jamming|electronic-jamming|jam} in terms of network performance degradation and human effort by to beat the sender in network with the assistance of recent jamming node.

Keywords **: Denial-of-Service, Spot Jamming, Electronic Jamming, Wireless networks, Packet classification**

Date of Submission: 25, July 2014 Date of Acceptance: 02, August 2014

1.INTRODUCTION

In the world of computers, **Networking** is the practice of linking two or more computing devices together for the purpose of sharing data. Networks are built with a mix of computer hardware and software. Networking includes communication with other users, centralization of software and account maintenance, and mobility of users. Whenever we have more than one computer being used at the same location, networking them together makes a lot of sense. Not only can we transfer files between them quickly and easily, but they can also share expensive resources like laser printers, hard disc arrays, backup tape drives, CD and DVD burners, scanners, internet connections and so on. Sharing of files from source to the destination is often referred as “file sharing” in networking. The router is the primary component, which is used to transfer all such files. While transferring the files, these routers are compromised by the attackers and hence it becomes malicious in nature. Therefore there arrives a problem in the delivery of files, because of these malicious routers. So we aim in detecting the existence of compromised routers and isolate them from the routing fabric by using the mobile agents called as AntNets.

There are two main types of network categories which are:

- Server based
- Peer-to-peer

In a server based network, there are computers set up to be primary providers of services such as file service or mail service. The computers providing the service are called servers and the computers that request and use the service are called client computers.

In a peer-to-peer network, various computers on the network can act both as clients and servers. For instance, many Microsoft Windows based computers will allow file and print sharing. These computers can act both as a client and a server and are also referred to as peers. Networks are combination of peer-to-peer and server based networks. The network operating system uses a network data protocol to communicate on the network to other computers. The network operating system supports the applications on that computer. A Network Operating System (NOS) includes Windows NT, Novell Netware, Linux, UNIX and others.

- ➢ Local area network (LAN), which is usually a small network constrained to a small geographic area.
- ➢ Wide area network (WAN) that is usually a larger network that covers a large geographic area.
- ➢ Metropolitan area networks (MAN), are large computer networks usually spanning a city.

Network topology is the study of the arrangement or mapping of the elements i.e. links or nodes of a network, especially the physical (real) and Logical (virtual) interconnections between nodes. Network topology signifies the way in which devices in the network see their logical relations to one another. The use of the term "logical" here is significant. That is, network topology is independent of the "physical" layout of the network. Even if networked computers are physically placed in a linear arrangement, if they are connected via a hub, the network has a Star topology, rather than a bus topology. In this regard the visual and operational characteristics of a network are distinct; the logical network topology is not necessarily the same as the physical layout. Networks may be classified based on the method of data used to convey the data; these include digital and analog networks.

1.1 Wireless sensor network

A **wireless sensor network (WSN)** consists of spatially distributed autonomous sensors to *monitor* physical or environmental conditions, such as temperature, sound, pressure, etc. and to cooperatively pass their data through the network to a main location. The more modern networks are bi-directional, also enabling *control* of sensor activity. The development of wireless sensor networks was motivated by military applications such as battlefield surveillance; today such networks are used in many industrial and consumer applications, such as industrial process monitoring and control, machine health monitoring, and so on.

The WSN is built of "nodes" – from a few to several hundreds or even thousands, where each node is connected to one (or sometimes several) sensors. Each such sensor network node has typically several parts: a radio transceiver with an internal antenna or connection to an external antenna, a microcontroller, an electronic circuit for interfacing with the sensors and an energy source, usually a battery or an embedded form of energy harvesting. A sensor node might vary in size from that of a shoebox down to the size of a grain of dust, although functioning "motes" of genuine microscopic dimensions have yet to be created. The cost of sensor nodes is similarly variable, ranging from a few to hundreds of dollars, depending on the complexity of the individual sensor nodes. Size and cost constraints on sensor nodes result in corresponding constraints on resources such as energy, memory, computational speed and communications bandwidth. The topology of the WSNs can vary from a simple star network to an advanced multi-hop wireless mesh network. The propagation technique between the hops of the network can be routing or flooding.

Recent developments in digital circuitry, wireless communication, and Micro Electro-mechanical Systems (MEM), have made possible the integration of sensing, communication, and power supply into an inch-scale sensor devices. Thus, the investigation for development of robust, easy deploying, micro sensor networks has attracted a great deal of attention . Wireless sensor networks are energy-limited and application-specific. These two characteristics pose new challenges in the network design. Inch-scale sensor devices are expected to operate over years with limited power supply.

Thus, the energy consumption becomes the foremost design consideration, while other constraints, such as throughput, latency, and fairness, become relatively less important. On the other hand, sensor networks are considered for a diverse range of civil and military applications, such as environmental Monitoring, home networking, medical vital signs monitoring and smart battlefield, among others. These requirements suggest that the classical Open System Interconnect (OSI) paradigm may not be suitable for sensor networks, but rather a methodology of cross layer communications network design may be more appropriate,..

2. RELATED WORK

WIRELESS networks rely on the uninterrupted availability of the wireless medium to interconnect participating nodes. However, the open nature of this medium leaves it vulnerable to multiple security threats. Anyone with a transceiver can eavesdrop on wireless transmissions, inject spurious messages, or jam legitimate ones. While eavesdropping and message injection can be prevented using cryptographic methods, jamming attacks are much harder to counter. They have been shown to actualize severe Denial-of-Service (DoS) attacks against wireless networks. In the simplest form of jamming, the adversary interferes with the reception of messages by transmitting a continuous jamming signal , or several short jamming pulses. Typically, jamming attacks have been considered under an external threat model, in which the jammer is not part of the network. Under this model, jamming strategies include the continuous or random transmission of high-power interference signals. However, adopting an “always- on” strategy has several disadvantages. First, the adversary has to expend a significant amount of energy to jam frequency bands of interest. Second, the continuous presence of unusually high interference levels makes this type of attacks easy to detect [1].

Conventional antijamming techniques rely extensively on spread-spectrum (SS) communications, or some form of jamming evasion (e.g., slow frequency hopping, or spatial retreats. SS techniques provide bit-level protection by spreading bits according to a secret pseudonoise (PN) code, known only to the communicating parties. These methods can only protect wireless transmissions under the external threat model. Potential disclosure of secrets due to node compromise neutralizes the gains of SS. Broadcast communications are particularly vulnerable under an internal threat model because all intended receivers must be aware of the secrets used to protect transmissions. Hence, the compromise of a single receiver is sufficient to reveal relevant cryptographic information. In earlier address the problem of jamming under an internal threat model. We consider a sophisticated adversary who is aware of network secrets and the implementation details of network protocols at any layer in the network stack. The adversary exploits his internal knowledge for launching selective jamming attacks in which specific messages of “high importance” are targeted. For example, a jammer can target route-request/route-reply messages at the routing layer to prevent route discovery, or target TCP acknowledgments in a TCP session to severely degrade the throughput of an end-to-end flow. To launch selective jamming attacks, the adversary must be capable of implementing a “classify-then-jam” strategy before the completion of a wireless transmission [1].

Such strategy can be actualized either by classifying transmitted packets using protocol semantics, or by decoding packets on the fly . In the latter method, the jammer may decode the first few bits of a packet for recovering useful packet identifiers such as packet type, source and destination address. After classification, the adversary must induce a sufficient number of bit errors so that the packet cannot be recovered at the receiver .Selective jamming requires an intimate knowledge of the physical (PHY) layer, as well as of the specifics of upper layers [1].

Recent advancements in wireless technologies have caused a shift in computing away from the traditional wired Internet, towards new paradigms of mobile computing. As wireless progressively becomes ubiquitous, affordable, and part of our daily lives, a plethora of security challenges will arise that were not present in the traditional network paradigm. Even now, for existing wireless networks, security is often cited as a major technical barrier that must be overcome before widespread adoption of mobile services can occur. Security and privacy for wireless systems is complicated by the fact that wireless devices can be cheaply and easily purchased. The combination of the commodity nature of wireless technologies and an increasingly sophisticated userbase ultimately means that adversaries will be able to easily gain access to communications between wireless devices either by purchasing their own device and running it in a monitor mode, or by employing slightly more sophisticated software radios capable of monitoring a broad array of radio technologies. Further, adversaries are now empowered to easily mount a variety of security attacks, such as injecting false data into the network, launching denial of service attacks, or even disrupting the routing and delivery of legitimate data [2].

Many wireless security threats are being addressed through appropriately designed network security architectures . These technologies are, essentially, modifications of traditional security services, such as confidentiality, authentication, and integrity, to the wireless domain. The wireless medium, however, introduces many threats that are not simply addressable through conventional security mechanisms. One important class of security threats that may be launched by adversaries, which are difficult to address through conventional network security techniques, are denial of service attacks. Traditionally, denial of service is concerned with filling user-domain and kernel-domain buffers . However, in the wireless domain, due to the shared nature of the wireless medium, the adversary is empowered to prevent others from even communicating. An adversary can simply disregard the medium access protocol and continually transmit on a wireless channel. By doing so, he either prevents users from being able to commence with legitimate MAC operations, or introduces packet collisions that force repeated backoffs, or even jams transmissions. Such MAC and PHY-layer security threats for wireless networks have been known for some time, and recently the issue of MAClayer weaknesses in 802.11 has been revisited by a recent announcement by the Australian CERT [2].

Signaling and control channels are essential to the operation of wireless communication networks. Such networks are constrained by the limited radio-frequency bandwidth and energy available to the mobile devices. Therefore, wireless networks implement various control mechanisms to conserve the limited resources. Many networks employ shared control channels for sending system control information. For example, the GSM cellular communication system has multiple control channels for different functionalities[2].

The broadcast channels (BCH), such as BCCH, SCH, and FCH, carry the network/cell identity, the structure of the current control channels and synchronization information. The common control channels (CCCH), such as AGCH, and PCH, are used for subscriber channel assignment and paging notification. A subscriber has to first lock to the appropriate channel of nearby base station by monitoring the broadcast control channels, send out connection requests to the base station and get an assignment of traffic channel before being able to initiate a call. The BCH and some CCCH in GSM are located at very specific timeslots and physical frequency band (usually TS0 on a single 200KHz band) such that a subscriber can easily listen to them. However, this makes the system vulnerable. An attacker can launch a denial of service attack by jamming the control channels. It is a highly energy efficient and effective attack for the attacker compared to jamming the whole frequency band to stop the communication. We simulate the scenario using Qualnet Simulator. The result shows that by jamming 1 timeslot (out of 8 timeslots) of BCCH in every 51 frames on a single 200KHz band, the attack prevents all the mobile stations from communicating with each other. This leads to a jammer four order of magnitude more efficient than a jammer that is not aware of the GSM structure. Similarly, Hass et al. discover and study the attacks against control channels in Personal Communications Services (PCS) network [4]. Wireless networks are highly sensitive to denial of service attacks , . The broadcast nature of wireless communication exposes the physical layer of the system to jamming. The traditional anti-jamming strategy has been extensively relying on spread spectrum technique . Very little work has been done from a system level to countermeasure jamming.

In our previous work, we propose a novel system architecture based on mechanism-hopping to increase the wireless network robustness against cross-layer jamming . Xu et al. study the effect and detection of jamming at MAC and PHY layer in wireless sensor networks in . Geng et al. survey the denial of service attacks against wireless networks and propose a policy based networking framework to defend against DDoS for mobile systems . Resilience and identification of internal attackers is very difficult. To the best of our knowledge, we are the first to investigate the problem of control channel jamming by traitors. We use results from coding theory to assign keys in our approach that guarantees the resilience and identification of traitors . Due to the shared use of the communication medium, wireless radio communications are not only vulnerable to traditional attacks such as eavesdropping and message synthesis, but also to active jamming attacks . In a signal jamming attack, the attacker emits a jamming signal while the legitimate transmission is taking place, thus achieving a denial-of-service (DoS) by blocking, modifying, annihilating, or overwriting the original signal. Well-known, effective countermeasures against signal jamming attacks are spread-spectrum techniques, in particular Direct-Sequence Spread Spectrum (DSSS) and Frequency Hopping Spread Spectrum (FH). For these techniques to work, the receivers are required to share secret keys with the sender prior to their anti-jamming communication; these keys enable them to derive identical spreading codes or hopping sequences. Shared secrets are also the basis of proposed anti-jamming broadcast schemes. The requirement of pre-shared secret keys, however, imposes limits on the use of common spread spectrum techniques for anti-jamming communication in scenarios where such secret keys cannot be preshared (but which instead rely on, e.g., public-key certificates). This problem (i.e., the lack of techniques for jamming resistance without shared secret keys) was recently observed in both and in the context- of pair wise communications. Wireless networks are progressively becoming more affordable, and consequently are being deployed in a variety of different modalities, ranging from wireless local area networks to mesh and sensor networks. As these networks gain popularity, providing security and trustworthiness will become an issue of critical importance. Many wireless security threats may be addressed through appropriately designed network security architectures, which are essentially modifications of traditional security services, such as confidentiality, authentication, and integrity to the wireless domain. Wireless networks, however, are susceptible to threats that are not able to be adequately addressed via cryptographic methods. One serious class of such threats are attacks of radio interference[3].

The shared nature of the wireless medium, combined with the commodity nature of wireless technologies and an increasingly sophisticated user-base, allows wireless networks to be easily monitored and broadcast on. Adversaries may easily observe communications between wireless devices, and just as easily launch simple denial of service attacks against wireless networks by injecting false messages. Traditionally, denial of service is concerned with user-domain and kernel-domain users . However, in the wireless domain, the adversary is empowered to launch more fundamentally severe types of denial of service that block the wireless medium and prevents other wireless devices from even communicating[4].

Radio interference attacks are not addressable through conventional security mechanisms. An adversary can simply disregard the medium access protocol and continually transmit on a wireless channel. By doing so, he either pre- vents users from being able to commence with legitimate MAC operations, or introduces packet collisions that force repeated backs, or even jams transmissions. Such MAC and PHY-layer security threats for wireless networks have been known for some time, and the issue of MAC-layer weak- nesses in 802.11 has been revisited by a recent announcement by the Australian CERT [5].

In order to ensure the dependability of future deployments of wireless networks, mechanisms are needed that will allow wireless networks of all types to cope with the threat of at tacks of radio interference, or simply RF jamming attacks. The rest stage to defending a wireless network is to understand what types of attacks are feasible, and how these attacks may be diagnosed [5].

3. DESCRIPTION OF PROPOSED SCHEME

To address the problem of jamming under an internal threat model and consider a sophisticated adversary who is aware of network secrets and the implementation details of network protocols at any layer in the network stack. The adversary exploits his internal knowledge for launching selective jamming attacks in which specific messages of high importance are targeted.

For example, a jammer can target route-request/route-reply messages at the routing layer to prevent route discovery, or target TCP acknowledgments in a TCP session to severely degrade the throughput of an end-to-end flow.

In our work two attacker nodes(node which creates jamming) and introduce one new node i.e Jammer node. The new node(jammer node) is within the 25 nodes. Using that new jammer node we have to eliminate the attacker nodes fully. We conclude that how jamming occurs in the network and way of elimination of the attacker nodes using new jammer node. The jammer may decode the first few bits of a packet for recovering useful packet identifiers such as packet type, source and destination address.

After classification, the adversary must induce a sufficient number of bit errors so that the packet cannot be recovered at the receiver. An intuitive solution to selective jamming would be the encryption of transmitted packets (including headers) with a static key. However, for broadcast communications, this static decryption key must be known to all intended receivers and hence, is susceptible to compromise.

Moreover, even if the encryption key of a hiding scheme were to remain secret, the static portions of a transmitted packet could potentially lead to packet classification.

3.1. Real Time Packet Classification

At the Physical layer, a packet m is encoded, interleaved, and modulated before it is transmitted over the wireless channel. At the receiver, the signal is demodulated, de-interleaved and decoded to recover the original packet m. Nodes A and B communicate via a wireless link. Within the communication range of both A and B there is a jamming node J. When A transmits a packet m to B, node J classifies m by receiving only the first few bytes of m. J then corrupts m beyond recovery by interfering with its reception at B.

3.2 A Strong Hiding Commitment Scheme

A strong hiding commitment scheme (SHCS), which is based on symmetric cryptography. Assume that the sender has a packet for Receiver. First, S constructs commit(message) the commitment function is an off-the-shelf symmetric encryption algorithm is a publicly known permutation, and k is a randomly selected key of some desired key length s (the length of k is a security parameter). Upon reception of d, any receiver R computes.

3.3 Cryptographic Puzzle Hiding Scheme

A sender S has a packet m for transmission. The sender selects a random key k , of a desired length. S generates a puzzle (key, time), where puzzle() denotes the puzzle generator function, and tp denotes the time required for the solution of the puzzle. Parameter is measured in units of time, and it is directly dependent on the assumed computational capability of the adversary, denoted by N and measured in computational operations per second. After generating the puzzle P, the sender broadcasts (C, P). At the receiver side, any receiver R solves the received puzzle to recover key and then computes.

3.4 Hiding based on All-Or-Nothing Transformations

The packets are pre-processed by an AONT before transmission but remain unencrypted. The jammer cannot perform packet classification until all pseudo-messages corresponding to the original packet have been received and the inverse transformation has been applied. Packet m is partitioned to a set of x input blocks m = {m1, m2, m3….}, which serve as an input to an The set of pseudo-messages m = {m1, m2, m3,…..} is transmitted over the wireless medium.

3.5 Avoid Jamming using Jammer Node

Here we create two attacker nodes(node which creates jamming) and introduce one new node i.e Jammer node. The new node(jammer node) is within the 25 nodes. Using that new jammer node we have to eliminate the attacker nodes fully. We conclude that how jamming occurs in the network and way of elimination of the attacker nodes using new jammer node.

4.0 Software Module Implementation

NS (version 2) is an object-oriented, discrete event driven network developed at UC Berkley written in OTCL. NS is primarily useful for simulating local and wide area networks. The purpose is to give a new user some basic idea of how the simulator works, how to create new network components, etc., mainly by giving simple examples and brief explanations based on our experiences. Although all the usage of the simulator or possible network simulation setups may not be covered in this project, the project should help a new user to get start quickly.

4.1 Overview

NS is an event driven network simulator developed at UC Berkley that simulates variety of IP networks. It implements network protocols such as TCP and UDP, traffic source behavior such as FTP, Telnet, web, CBR and VBR, router queue management mechanism such as Drop Tail, RED and CBQ, routing algorithms such as Dijkstra, and more. NS also implements multicasting and some of the MAC layer protocols for LAN simulations. The NS is now a part of the VINT Project that develops tools for simulation results display, analysis and converters that convert network topology generated by well-Known generators to NS formats. Currently, NS(version 2) written in OTCL(TCL script language with object-oriented extensions developed at MIT) is available. This document explains briefly about basic structure of NS and explains in detail how to use NS mostly by giving examples.

As shown in figure, in a simplified user's view, NS is object-oriented TCL (OTCL) script interpreter that has a simulation event scheduler and network component, and object libraries, and network setup (plumbing)module libraries(actually, plumbing modules are implemented as member function of the base simulator object). In other words, to use NS, you program in OTCL script language. To setup and run a simulation network, a user should write a OTCL script that initiates an event schedule, setups the network topology using the network objects and plumbing functions in the library, and tells traffic sources when to start and stop transmitting packets to the event scheduler.

The term plumbing is used for a network setup, because setting up a network is plumbing possible data paths among network among network object by setting the "neighbor" pointer of an object to the address of the appropriate object. When a user wants to make a network object, an object can be made either by writing a new object or by making a compound object from the object library, and plumb the data path through the object. This may sound like complicated job, but the plumbing OTCL modules actually make a job very easy. The power of NS comes from this plumbing.

Another major component of NS beside network objects is the event scheduler. An event in NS is a packet ID that is unique for a packet with scheduled timing and the pointer to an object that handles the event. In NS, an event scheduler keeps track of simulation time and fires all the events in the event queue scheduled for the current time by invoking appropriate network components, which usually are the ones who issued the events, and let them do the appropriate action associated with packet pointed by the event. Network components communicate with one another by passing packets; however this doesn't consume actual simulation time. All the network components that need to spend some simulation time handling a packet(i.e need a delay) use the event scheduler by issuing a event for the packet and waiting for the event to be fired to itself doing further action of handling the packets.

For example, a network switch component that simulates the switch with 20microseconds of switching delay issues an event for a packet to be switched to the scheduler as an event 20microsecond later. The scheduler after 20microsecond dequeues the event and fires it to the switch component, which then passes the packet to an appropriate output link component. Another use of an event scheduler is timer.

For example, TCP needs a timer to keep track of packet transmission time out for retransmission (transmission of a packet with the same TCP packet number but different NS packet ID). Timers use event schedulers in a similar manner that delay does. The only difference is that timer measures a time value associated with the packet and does an appropriate action related to the packet after a certain time goes by, and does not simulate a delay.

NS is return not only in OTCL but in C++ also. For efficiency reason NS separates the data path implementation from control path implementation. In order to reduce packet and event processing time (not simulation time), the event scheduler and the basic network component objects in the data path are written and compiled using c++.

These compiled objects are made available to the OTCL interpreter through an OTCL linkage that creates a matching OTCL object for each of the c++ objects and makes the control function and the configurable variables specified by c++ object act as a member function and member variables of the corresponding OTCL project.

In this way, the controls of the c++ object are given to Otcl. It is also possible to add member functions and variables to a c++ linked Otcl object. The objects in c++ that do not need to be controlled in a simulation or internally used by another object do not need to be linked to otcl. Likewise an object (not in the data path)can be entirely implemented in otcl. Figure shows an object hierarchy example in c++ and otcl. One thing to note in the figure is that for C++ objects that have an Otcl linkage forming a hierarchy, there is a matching otcl object hierarchy very similar to that of c++.

5 RESULTS AND DISCUSSIONS

Figure 5.1: Jammer network creation

Figure 5.2: Packet transition between jammer node

Figure5.3 : avoid the attackers from jammer network

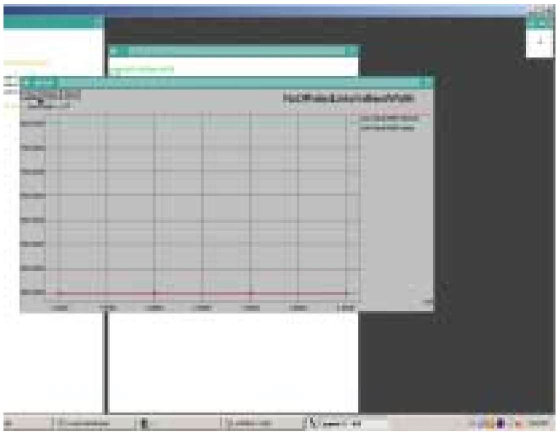
Figure 5.4 Number of failed link vs bandwidth

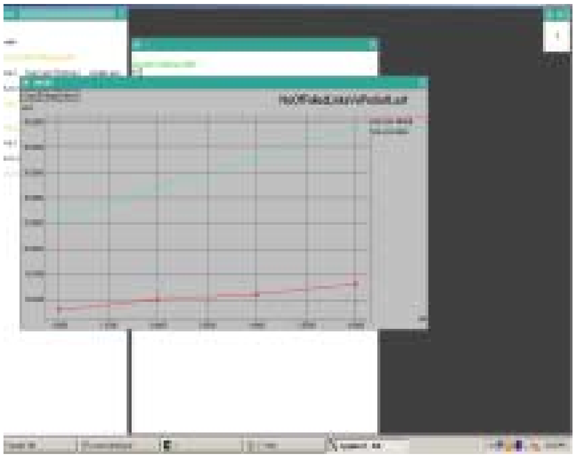
Figure 5.5 Failed link vs packet lost

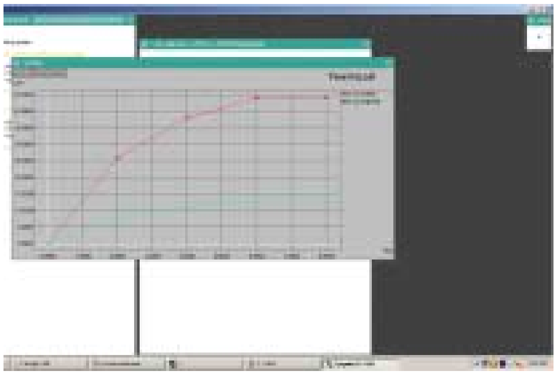
Figure 5.6 Time vs Lost

CONCLUSION

An internal adversary model in which the jammer is part of the network under attack, thus being aware of the protocol specifications and shared network secrets. We showed that the jammer can classify transmitted packets in real time by decoding the first few symbols of an ongoing transmission. We evaluated the impact of selective jamming attacks on network protocols such as TCP and routing. Our findings show that a selective jammer can significantly impact performance with very low effort through new jammer node in that wireless network.

REFERENCE

[1]. Packet-Hiding Methods for Preventing Selective Jamming Attacks, Alejandro Proaño, University of Arizona, Tucson Loukas Lazos, University of Arizona, Tucson

[2]. Channel Surfing and Spatial Retreats: Defenses against Wireless Denial of Service Wenyuan Xu, Timothy Wood, Wade Trappe, YanyongZhang Wireless Information Network Laboratory (WINLAB)Rutgers, The State University of New Jersey,73 Brett Rd.Piscataway, NJ 08854

[3]. Jamming-resistant Broadcast Communication without Shared Keys, ETH Zurich D-INFK Tech. Report 609 – March 12, 2009. Accepted for publication at Usenix Security Symp. 2009, Christina Popper,System Security Group,ETH Zurich, Switzerland,poepperc@inf.ethz.ch

[4]. Distributed IDS in Case of Continuous Attack and Performance Analysis of Access points. A.K.Kakakhel and N.Anjum,City University of Science and Technology,Peshwar,Pakistan.

[5]. Traffic Rerouting Strategy against Jamming Attacks in WSNs for Microgrid,Yujin Lim,1 Hak-Man Kim,2 and Tetsuo Kinoshita3

Authors Biography

P. Srimanchari received the B.Sc degree in Computer Science from Erode Arts College for Women, M.Sc degree in Computer Science from Vivekanandaha Arts College, Namakkal and the M.Phil degree in Computer Science from Bharathiar University, Coimbatore, India, in 1998, 2000 and 2004, respectively. She is currently pursuing a Ph.D degree in Computer Science at the Bharathiar University of Coimbatore, India. She is presently working as Assistant Professor in the Department of Computer Science, Erode Arts and Science College, Erode, Tamilnadu, India. Her research interests include Mobile Computing and Wireless Network Technology.

Mr. V. Perumal has obtained B.sc degree from Saraswathi Narayanan college-perungudi ,Madurai kamaraj university and obtained M.Sc degree in Computer Science from Pasumpon Muthuramali nga thevar College-Usilampatti, adurai kamaraj university . Presently he is pursuing M.Phil in Computer Science from Erode Arts&Science College- Rangampalayam. His research interest includes Computer Network, Network Security.

Dr. M. Saroja is currently working as Associate Professor in the Department of Electronics, Erode Arts & Science College, Erode, Tamilnadu, India. She received her Ph.D from Bharathiar University, Coimbatore. She has twenty years of experience in teaching as well as research. She has presented papers at Twenty Three International and National Conferences and has published research articles in twenty five leading Journals. She is an active researcher and is usually associated with reputed Academic Forums and Associations of research interest. She is currently involved in a UGC major research project. Her research interests are Thin Films, Mobile Computing and Wireless Sensor Network. She is also a life member of ISTE.

Dr. M. Venkatachalam received Ph.D from PSG College of Technology, Coimbatore. He is currently working as Professor and HOD in the Department of Electronics, Erode Arts & Science College, Erode, Tamilnadu, India. He has twenty four years of experience in teaching as well as Research. He has presented twenty five research papers at International and National Conferences and has published thirty five research articles in leading Journals. She is currently involved in a UGC major research project. His research interests are Thin Films, VLSI Design, Mobile Computing and Wireless Sensor Network. He is a also life member of ISTE