2007

Bluetooth-base Worm Modeling And Simulation

Haiou Xiang
University of Central Florida

BLUETOOTH-BASE WORM MODELING AND SIMULATION

by

HAIOU XIANG
B.A. Chongqing University, 1998

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science
in School of Electrical Engineering and Computer Science
in College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2007

ABSTRACT

Bluetooth is one of the most popular technologies in the world in the new century. Meanwhile it attracts attackers to develop new worm and malicious code attacking Bluetooth wireless network. So far the growth of mobile malicious code is very fast and they have become a great potential threat to our society. In this thesis, we study Bluetooth worm in Mobile Wireless Network. Firstly we investigate the Bluetooth technology and several previously appeared Bluetooth worms, e.g. “Caribe”,”Comwar”, and we find the infection cycle of a Bluetooth worm. Next, we develop a new simulator, Bluetooth Worm simulator (BTWS), which simulates Bluetooth worm’ behaviors in Mobile wireless networks. Through analyzing the result, we find i) In ideal environment the mobility of Bluetooth device can improve the worm’s propagation speed, but combining mobility and inquiry time issue would cause a Bluetooth worm to slow down its propagation under certain situation. ii) The number of initially infected Bluetooth devices mostly affects the beginning propagation speed of a worm, and energy issue can be ignored because the new technology can let Bluetooth device keeping work for a long time. iii) Co-channel interference and setting up monitoring system in public place can improve the security of Bluetooth wireless network.

ACKNOWLEDGMENTS

I wish to thank Dr. Cliff Zou who gives me useful advice for simulating method of worm and review my thesis.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 Background of Network Worm

A computer worm is a self-replicating computer program. It uses a network to sent copies of itself to other nodes (computer terminals on the network) and it may do so without any user intervention [5]. Unlike a virus, it does not need to attach itself to an existing program. The first worm, the Christmas Tree EXEC, appeared on a worldwide network in 1987, which spread across both IBM's own international network and BITNET [6]. Actually, Christmas Tree EXEC was technically a Trojan horse (a program that installs malicious software while under the guise of doing something else). The first worm that caused massive disruption of the internet was the Morris worm, written by a computer science graduate student at Cornell University in 1988 [7].

When human enters the 21st century, the worm stars its engine and brings disaster to the internet. On July 13, 2001, the most famous Internet worm, Code Red [9], was noticed because of its unbelievable spreading speed. On July 18 Security company eEye Digital Security discovered the flaw in IIS that Code Red exploits. Code Red worm exploited a vulnerability in the indexing software distributed with IIS [11, 12], spreading itself using a common type of vulnerability known as a buffer overflow. It does this by using a long string of a repeated character 'N' to overflow a buffer, allowing the worm to execute an arbitrary code and infect the machine. Then the infected host attempts to connect to TCP port 80 of randomly generated IP addresses in order to propagate the worm. At the same time, the worm starts 100 worm threads in memory. When the date

is between the 20th and 27th of a month, the worm starts a denial-of-service attack on www.whitehouse.gov. On July 19, it infected 359,000 [8, 10] hosts in internet. The worse situation is that more than 2,000 new hosts were infected each minute. 43% of all infected hosts were in the United State, and 19% of all compromised machines were the .NET Top Level Domain, followed by .COM with 14% [9].

With the development of computer technology, the Internet worms have developed into different types as well. An email worm [13] spreads via email messages. Typically, the worm code contains in email attachment or the email may contain a link to an infected website. Once a user activates the worm, it can use “social engineering”, such as user’s contact address book, to propagate itself. In modern society, a lot of younger like to chat in internet, so an IRC worm [14] uses the chat channels to spread infected files. Another popular internet application is Peer-to-Peer (P2P) application. A file-sharing networks worm places itself in a shared folder and spreads via the P2P network [15]. Internet worms not only disrupt the network traffic, but also have payload to implement many kinds of attacks, such as installing backdoor, deleting system files, or encrypting files.

Internet worms have already become a major threat in the internet due to their faster spreading and its serious devastating. According to the report of London-based market intelligence firm Mi2g in 2003 ‘Code Red’ worm brought almost $2.6 billion in productivity cost, and SQL ‘Slammer’ worm caused between $950 million and $1.2 billion in lost productivity in its first five days worldwide. These costs do not include labors costs and cleanup costs. In 2006, from the FBI’s survey [16] from 2,066

organizations, “This would be 2.8 million U.S. organizations experiencing at least one computer security incident. With each of these 2.8 million organizations incurring a $24,000 average loss, this would total $67.2 billion per year.” In this survey, worms, viruses and Trojan horses were most costly computer crime.

1.2 Background of Bluetooth Network Worm

Wireless has already been one of the most important technologies in 21 Century. Mobile phone is not just a telephone and becomes an intelligent device with multi-function. Smart devices, such as PDAs, smart phones, on-board car computers, and even new generation appliances are now equipped with communications functions. Nowadays, human live in a huge Wireless network and are entering a wireless era. In the meantime, wireless technology open a new window to attacker, and parts of attacking techniques had been immigrated to wireless network.

IEEE 802.11 (Wi-Fi) and Bluetooth are the primary wireless technology in internet. Although both of technologies were implemented in 1990’s, the first mobile virus appeared in June 2004, and it was called ‘Caribe’ [17]. Caribe was written for the Symbian OS and spread via Bluetooth [38]. In July 2004, antivirus company discovered another mobile virus, ‘Duts’ [18], which is the first mobile virus to infect the Windows CE OS. One more month later, the first backdoor virus for mobile platforms appeared, called ‘Brador’. Then several Trojan viruses were developed for Symbian platform, such as ‘Mosquit’, ‘Locknut’, ‘Dampig’ [39, 40, 41], and so on. Until January 2005, a new mobile virus, ‘Comwar’ [19], brought new functionality – the first malicious program with the ability to propagate via MMS. From above introduction, we notice the speed of

increasing number of mobile virus grows significantly. Figure 1 shows mobile virus, 15 mobile virus and 27 variants were discovered during one year.

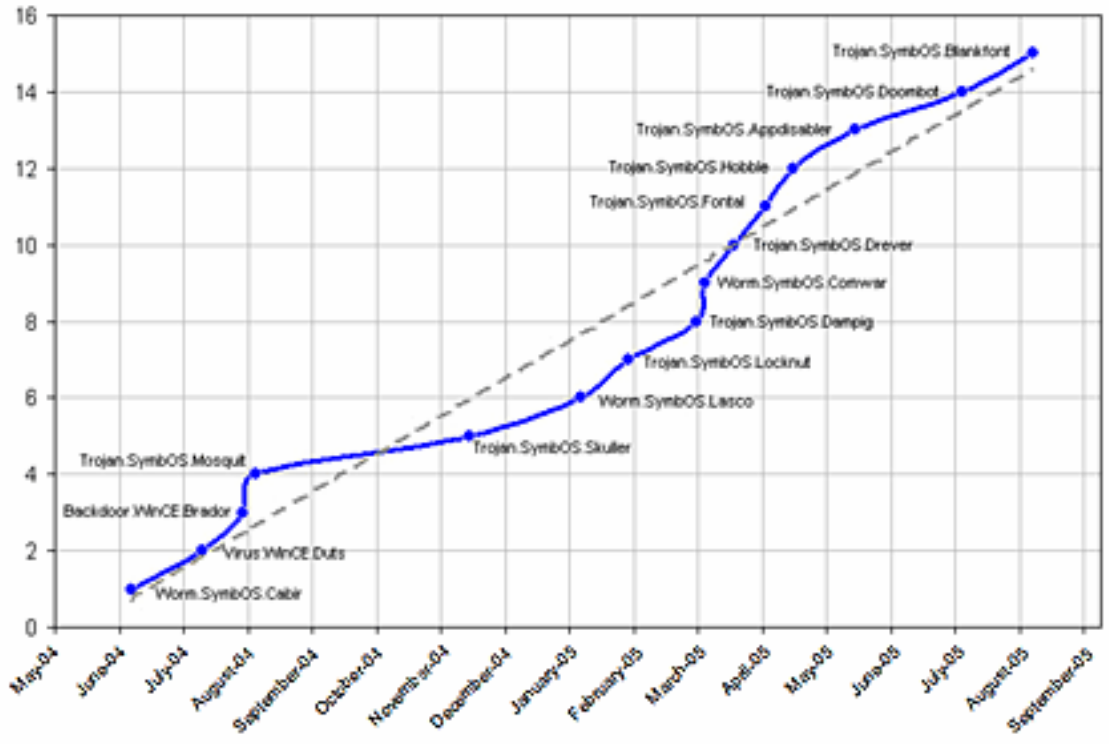

Figure 1 Increase in The Number of Known Mobile Virus Families
(Source: Viruslist.com - An overview of mobile device security)

In section 1, we know the internet worm is popular in wired networks, but in mobile networks, there are only two popular worms, ‘Caribe’ and ‘Comwar’. The Cabir worm is the first Bluetooth worm that runs in Symbian mobile phones that support Series 60 platform. The Cabir worm replicates over Bluetooth connections and arrives to phone messaging inbox as caribe.sis file what contains the worm. When user clicks the caribe.sis and chooses to install the file the worm activates and starts looking for new devices to infect over Bluetooth. When the Cabir worm finds another Bluetooth device, it

will send infected SIS files to its neighbor, and lock to that phone so that it won't look other phones even when the target moves out of range. The Comwar worm uses MMS or Bluetooth technologies to propagate that operates on Symbian Series 60 devices, either. ComWar not only spread over Bluetooth but also MMS. If it is the first hour of the 14th of any month, the threat resets the device.

In 2006, over 600 million Bluetooth-enabled devices were shipped [32], and there are more than a billion Bluetooth units to be installed. People use Bluetooth at home for internet or intranet, use Bluetooth headset for cell phone, number of Bluetooth hot-spots also are set up in Coffee, restaurant and cinema. Bluetooth Indeed give people convenience, but it also provides a chance for attacker to spread worm in wider range.

1.3 Properties and Characteristics of Bluetooth

1.3.1 History of Bluetooth Technology

Bluetooth wireless technology is a short-range communications technology intended to replace the cables connecting portable and/or fixed devices while maintaining high levels of security. In 1994, the Swedish company Ericsson initiated the Bluetooth Technology movement [20]. "The original intention was to make a wireless connection between something like an earphone and a cordless headset and the mobile phone," Haartsen said who is Bluetooth co-inventor. In 1998 the Bluetooth Special Interest Group (SIG) is formed and officially adopts the project name "Bluetooth" as the name of the technology. From 1999 to 2004, Bluetooth SIG adopts three Bluetooth Core Specification Versions, now it is Version 2.0 + Enhanced Data Rate [21].

1.3.2 Features of Bluetooth Technology

1.3.2.1 Unlicensed Spectrum

Bluetooth Technology operates in the unlicensed industrial, scientific and medical (ISM) band at 2.4 to 2.485 GHz, using a spread spectrum, frequency hopping, full-duplex signal at a nominal rate of 1600 hops/sec.

1.3.2.2 Efficient Interference

Adaptive frequency hopping (AFH) capability in Bluetooth Technology reduces the interference between wireless technologies sharing the 2.4GHz spectrum. This adaptive hopping allows for more efficient transmission with the spectrum, providing users with greater performance. The signal hops among 79 frequencies at 1 MHz intervals to give a high degree of interference immunity.

1.3.2.3 Three Operating Range

Class 3 radios: up to 1 meter

Class 2 radios: 10 meters (most using in mobile devices)

Class 1 radios: 100 meters (industrial use cases)

1.3.2.4 Low Power Consumption

Most users used Class 2 radios, so its power is 2.5mW. In addition, the Bluetooth device is allowed radios to be power down when inactive.

1.3.2.5 Data Rate

In Bluetooth Core Specification Version 1.2, Data Rate is set to 1 Mbps; however, in Version 2.0 + EDR, it increases to 3 Mbps.

1.3.3 Operation of Bluetooth Technology

Bluetooth core system consists of an RF transceiver, baseband, and protocol stack. The system offers services that enable the connection of devices and the exchange of a variety of data classes between these devices. Figure 2 shows the Core_System_Architecture.

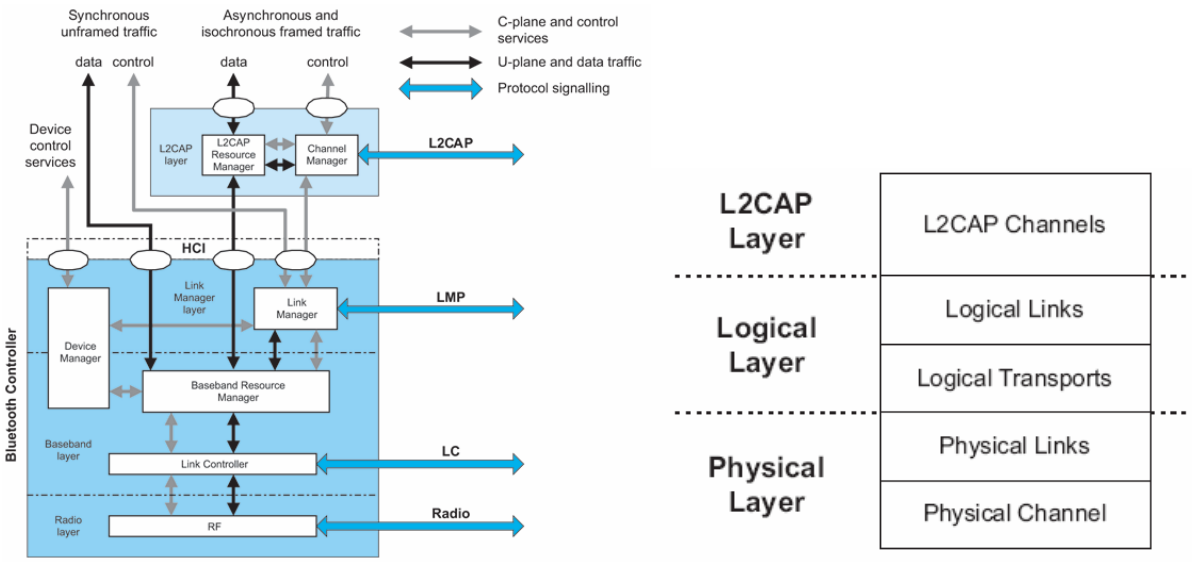

Figure 2 Core System Architecture of Bluetooth
(Source: Bluetooth Specification 2.0)

In physical layer (Radio layer), the Bluetooth RF operates in unlicensed ISM band at 2.4GHz. The system employs a frequency hop transceiver to combat interference and fading, and provides many FHSS carriers. When Bluetooth devices operate in Radio layer, they shape to a group in which each device is synchronized to a

common clock and frequency hopping pattern. One device is called the master that provides the synchronization reference. All others are called slaves. The master and the slaves form a piconet. Devices in a piconet use a specific frequency hopping pattern that is a pesudo-random ordering of the 79 frequencies in the ISM band. The data is stored in package and is transmitted by a number of consecutive time slots. In addition, the physical link is formed between any two devices that transmit packets in either direction.

Above physical layer there is Logical layer (Baseband layer) and L2CAP layer. In the Logical layer, logical link can control flow, acknowledgement/repeat mechanisms, sequence numbering and scheduling behavior, and logical transports carry different types of logical links.

The highest layer is L2CAP layer that provides a channel-based abstraction to applications and services. It carries out segmentation and reassembly of application data and multiplexing and de-multiple channels over a shared logical link. L2CAP has a protocol control channel that is carried over the default ACL logical transport.

1.4 Bluetooth Worm Infection

Before Bluetooth worm propagate in wireless network, the attacker need discover vulnerable node. In Bluetooth operations, a Bluetooth-enabled device uses the inquiry procedure to discover nearby devices, or to be discovered by devices in their locality. When a Bluetooth-enabled device tries to find new devices, it enters inquiry sub-state. In this sub-state, it shall repeatedly transmit the inquiry message at different hop frequencies. If a Bluetooth-enabled device allows itself to be discovered, it shall

regularly enter the inquiry scan sub-state to respond to inquiry messages. When the inquiry message is received in the inquiry scan sub-state, the recipient shall return an inquiry response (FHS) packet containing the recipient's device address and other parameters. The entire inquiry procedure is asymmetrical, and does not use any of the architectural layers above the physical layer.

After neighbors discovering, two Bluetooth-enabled devices enter paging (connecting) procedure. In order to set up a connection between two devices, only the Bluetooth device address is required. In the page scan sub-state, the device shall select the scan frequency according to the page hopping sequence determined by the device's address. Because there are master and slave in a piconet, the master enters page sub-state in page scan sub-state. The master tries to coincide with the slave's scan activity by repeatedly transmitting the paging message consisting of the slave's device access code (DAC) in different hop channels. On receiving the page message, the slave enters the slave response sub-state that the salve device transmits a slave page response message. Then the master receives a slave page response message, it enters the master response sub-state. The master shall transmit an FHS packet to slave device. If the slave's response is received by the master, the master enters the connection state and starts to transmit data.

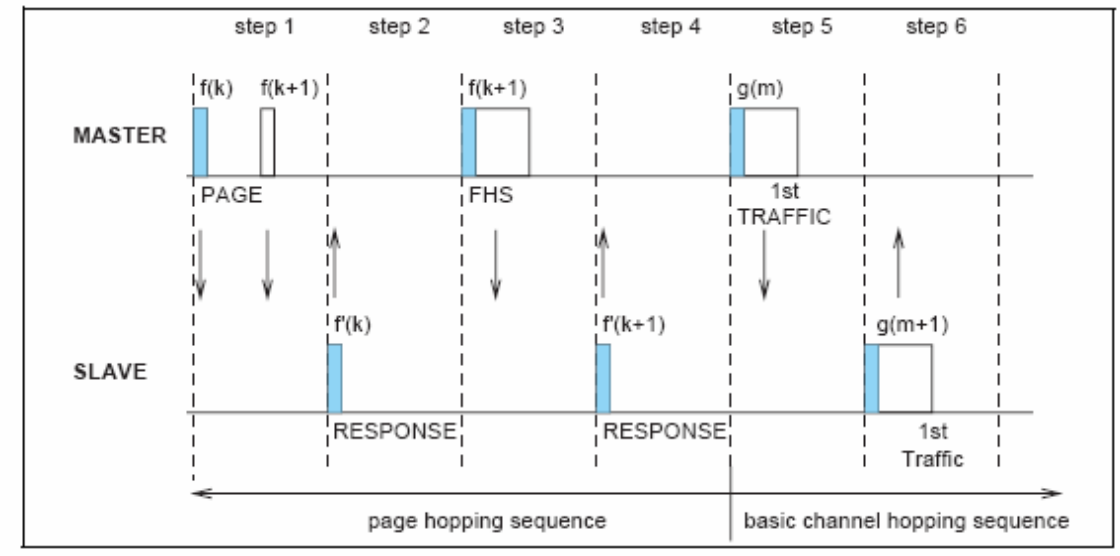

Figure 3 Messaging at Initial Connection
(Source: Bluetooth Specification 2.0)

For a regular Bluetooth device, they are usually in Idle (sleep state). In this paper, however, attacker always is in inquiry state. Firstly it broadcasts the inquiry message. When Bluetooth enabled devices response the inquiry request, attacker generates a neighbor list table. It extracts one of neighbor to set up the connection as slave device. If successful connection, infected file will be sent to vulnerable device. Then attacker disconnect with infected device. During the process of replicating infected file and disconnecting, there exists a timer, when the timer expired, the attacker device automatically stop the connection and try to connect other neighbor in table. If the neighbor list table is empty, attacker will broadcast a new inquiry message. When the user of vulnerable device runs the infected file, it is infected and starts to broadcast an inquiry message to find new vulnerable neighbors. Figure 4 describes the entire procedure of Bluetooth worm infection.

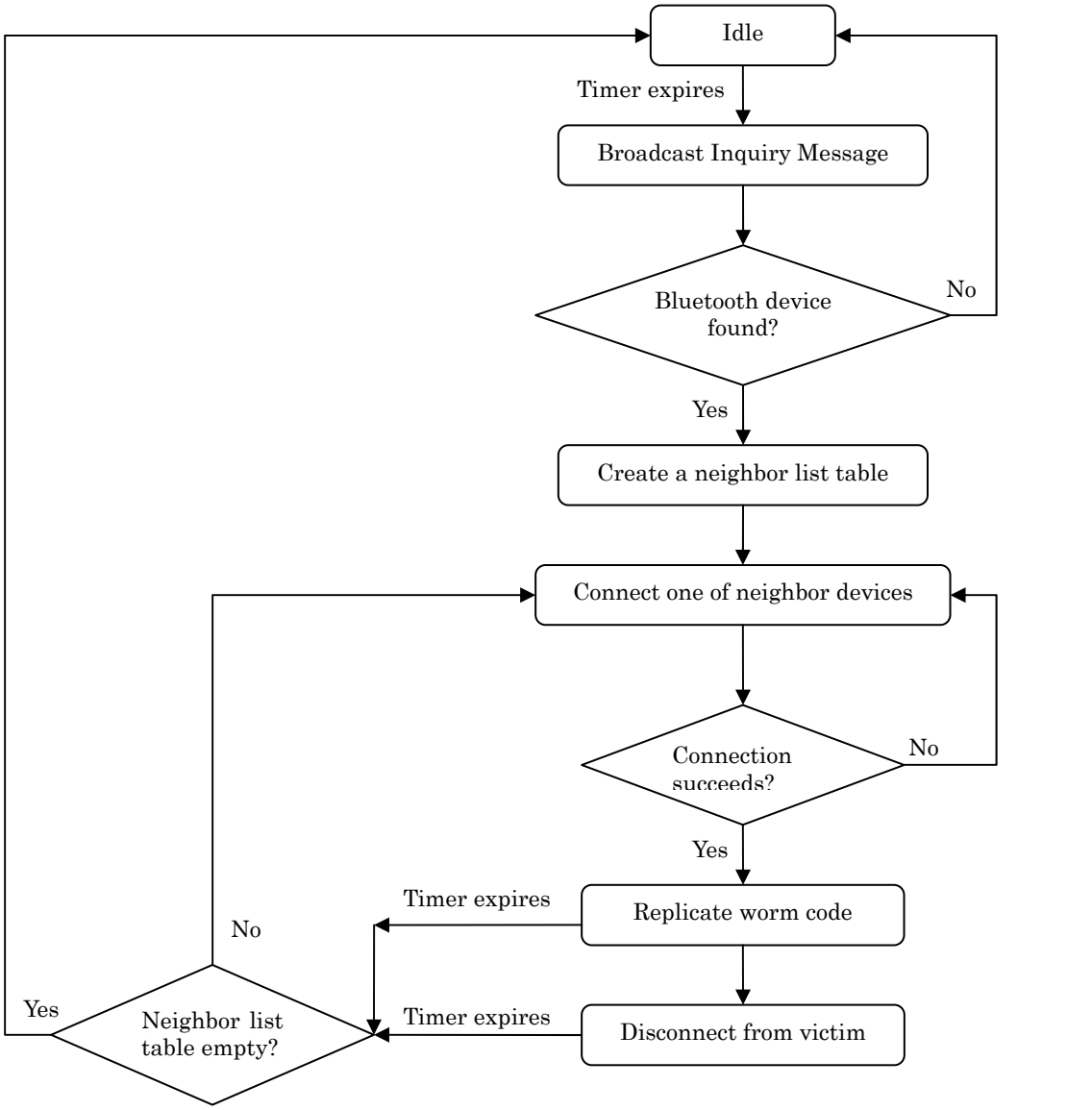

Figure 4 Bluetooth Worm Infection Life Cycle

1.5 Wireless Network Simulator Introduction

In computer network research, network simulation is a technique where a program simulates the behavior of a network. There exist several Free/Open Source

network simulators, such as NS, OMNet++, GloMoSim, Shunra, NetSim, and OPNET. Most of them include the wireless network simulation.

In addition, IBM developed BlueHoc [22] simulator that is a Bluetooth extension for NS (ver2.1b6). It implements basic features of Bluetooth baseband, Logical Link Control and Adaptation Protocol (L2CAP) and Link Manager Protocol (LMP), and it adds eight C++ classes to ns to support device discovery, paging and connection establishment functions, and it has trace support and graphical interface. After BlueHoc, MIT developed another Bluetooth simulator, Blueware [23]. It is still an ns extension and is based on the BlueHoc simulator. Therefore, Blueware uses the most of original code, but it adds large number of new code. In particular, Blueware provides an easy-to-program interface to various scatter net formation, link scheduling schemes and their related algorithm, TSF and LCS. Blueware works with NS (ver 2.1b7a).

Although BlueHoc and Blueware implement most of basic functions of Bluetooth, they were developed in 2002 and just work at NS's old version, the newest NS2 [24] is ver2.31 (released Mar 10, 2007), and also they support Bluetooth specification 1.1, the newest Bluetooth specification is ver2.0 + Enhanced Data Rate. Because both of above simulator cannot satisfy the new NS2 and Bluetooth specification, University of Cincinnati developed a new Bluetooth simulator, UCBT [25], which works at NS-2 (ver2.28 even later version) and partially support Bluetooth specification 2.0. As same features with BlueHoc and Blueware, UCBT is NS-2 based simulator and supports basic Bluetooth functions as well. It, however, adds more than 28,000 lines of C++ codes, and it adapts to the PAN profile with Bluetooth Network Encapsulation Protocol (BNEP) and

Enhanced Data Rate (EDR) specification, and it takes clock drift into account. UCBT is the most accurate, complete and up-to-date open-source Bluetooth simulator.

1.6 BTWS (Bluetooth Worm Simulator)

From pervious section, we learn that NS-2 and OPNET are so large and complex, and they are good at simulating small size of network, but for large scale network they run slowly and low efficiency. BlueHoc and Blueware are too old to compatible for new C++ complier. UCBT is a full implementation of the Bluetooth protocol stack, but it is specially developed for Bluetooth scatter net research and does not support worm propagation model. In our simulation, we focus on propagation of Bluetooth worm. Simulator need support worm behaviors. Unfortunately, there is no simulator to provide worm model.

Therefore, I develop new simulator, BTWS (Bluetooth Worm Simulator). In our research, we mainly consider how the Bluetooth worm propagates quickly in wireless network and what parameters infect the spreading speed of the worm. In BTWS, we don’t concern scatter net and energy issue. BTWS uses some NS-2 codes that are wireless class, so every Bluetooth device is a node with speed, location and communication range properties. BTWS has several advantages. It can calculate the worm propagation quickly. It isn’t a NS-2 extension, so its size is so small and support both small and large scale wireless network.

1.7 Contribution of Thesis

In this thesis, we research on worm propagation behaviors in wireless network and focus on wireless Bluetooth technology. Our primary contributions in this thesis are below:

- From Bluetooth technology specification, we learn the mechanisms of the Bluetooth worm propagation.
- Develop a new simulation – BTWS (Bluetooth Worm Simulation), which is based on part of Bluetooth technology specification 2.0 and uses some of NS-2's code.
- Simulate two sizes of wireless network, Local scale and Wide scale.

--In local scale simulation, we simulate worm propagation with different properties, such as density, speed, the number of initially infected nodes, contact degree and inquiry time. Faster speed helps worm propagation, but longer inquiry time and co-channel interference reduce the worm propagation.

--In Wide scale simulation, we implement five scenarios to learn that there will be a positive impact for Bluetooth worm propagation if Bluetooth-enabled devices often transfer among different groups.

In section 1, this paper introduces the background of computer worm in wired networks and wireless mobile networks. In section 2, we discuss some related work with the Bluetooth worm. We simulate the worm behaviors in local scale and wide scale network environments in section 3. Then, in section 4, we try to find defense method for Bluetooth worm. Finally, in section 5, we present our conclusion.

CHAPTER 2: RELATED WORK

Researcher had already done a lot of work on Internet worm including wire and wireless network, and most of them focus on analyzing, modeling and simulating. But for Bluetooth wireless network, there are few papers, especially on worm propagation behaviors analyzing. Yan et al. [1] propose a baseline worm model to analyze the speed of Bluetooth worm propagation. Yan also use the radio propagation model to calculate signal attenuation. Through simulation he analyzes the effects of speed, density and network size and either did dynamics analysis in different scenarios. He found that mobility may not be key feature for Bluetooth worm propagation, and link instability owing could reduce the worm spreading speed, and the inference factor even slow down the worm propagation in high density network. Except of Bluetooth network, Yan considers the out-of-band propagation. An intelligent attacker cannot only use Bluetooth technology, but also he can use GSM/CDMA and GPRS technology to accelerate worm spreading [27].

Mickens et al. [2] find the standard worm propagation models cannot satisfy with mobile network, so he introduced new model, called probabilistic queuing. He uses several examples to demonstrate the failure of the Kephart-White model [30] in mobile environments, which cannot capture the non-trivial connectivity variances and is insensitive to node speed in mobile environments. Probabilistic queuing model treats node mobility as a first-order concern. It provides an accurate threshold condition related with the virulence of malicious code to the likelihood, and it also provide accurate estimates of these persistent infection levels.

Bose et al. [3] notice the growth of the SMS/MMS and Bluetooth technology will bring the more mobile viruses and worm in mobile environment. They study MMS and Bluetooth devices vulnerabilities in-depth and developed a fine-grained agent-based mal-ware modeling (AMM) framework to study the worm propagation. They use the SMS usage characterization collected call data records and SS7 traces [31] from a large cellular carrier to simulate the worm spreading. The results show the growth rate of a mobile virus exploiting SMS messages is small, but the growth rate increases significantly when these handsets are highly vulnerable to Bluetooth exploits.

Su et al. [4] and other researcher at University of Toronto did a preliminary investigation of the worm infection in Bluetooth environment. They implement real experiments, which use PDAs to scan other Bluetooth devices in Mall and subway. The results of tracing activities show it is very quickly and easy for Bluetooth worm to spread in a popular place. Moreover, mobility cannot impact the worm exploiting vulnerability and whatever the direction is. In addition, they use trace-driven simulation to do experiment in a large scale network [29]. They find Bluetooth worm can infect 10,000 devices in a few days and spreads more rapidly in day than does in night. N srl et al. [37] creates a BlueBag device to convert attack and scan Bluetooth device. They demonstrate the existence of a very high risk potential, created by low awareness, ever-increasing functionalities and complexity, and by the feasibility of targeted, covert attacks through Bluetooth-enabled malware.

Other researchers do not study especially in Bluetooth technology but in other wireless network. Khayam et al. [33] develop a new model, topologically-aware worm propagation model for wireless sensor networks. It takes the MAC layer interference into consideration. In his simulation, he also performs the Box-Muller transformation [34] to generate Gaussian random variable for simulating the fading affected of neighbor nodes. Finally, the new model accurately predicts the result of simulation. However this model just fit in stationary environment. Hoh et al [35] study the worm propagation in ad hoc with wide-area network. He proposes a new architecture for an intrusion response system by developing and analyzing location-based quarantine boundary estimation techniques. The detection probability of this technique is greater than 95% and a false-alarm rate of less than about 35%. Wagner et al. [36] investigates the behavior of the worm propagation and design worm simulation to predict its spreading potential in order to defense worm by early detection.

CHAPTER 3: SIMULATION

In my experiment, there are two parts: local wireless networks and wide wireless networks. Bluetooth technology has a limitation that devices with Bluetooth-function just communicate within an operating range; therefore, the Bluetooth worm can spread faster in a high-density public place than in a low-density walking road. In the simulation, most experiments simulate the Bluetooth worm in a local group. However, in the real world, people often transfers from one place to another place, and cell phone carriers cover range as large as possible in order to achieve the biggest revenue. So I need to simulate the Bluetooth worm propagation in a wide environment as well.

In a local wireless network, all nodes (individual Bluetooth device) are arranged in an area and their movement is limited to that local network, and no node can move outside, which means a stationary environment. On the other hand, in wide wireless networks, there exist many groups, and nodes in each group can be exchanged, which means a mutable environment.

Table 1 Parameter List

Parameter Name	Unit	Description
Node	Number	Bluetooth-enabled device
Initially infected node	Number	The number of worm source
Square Area	m^2	Simulation Environment Area
Density	Node / Square Area	The number of nodes in Simulation Environment Area
Speed	meter / second	The speed of Bluetooth-enabled device
Operating range	meter	Communicating range between two Bluetooth-enabled devices
Contact degree	Number of slave per master	The number of slave nodes per master node
Propagation Time	second	Bluetooth worm spreading time
Healthy rate	%	The immunizing Bluetooth devices / Total Bluetooth devices
Inquiry time	second	Time of scanning neighbor + setting up connection + Time of transferring Infected file
Infected rate	%	Simulation stop when infected nodes reach infected rate X Total nodes

Table 2 Default Parameters Assumption

Square Area	1000 X 1000		Simulation Num	50
Nodes	200		Operating Range	10 m
Initially infected Node	3		Infected rate in simulation	95%
Speed	2 m/s			

Parameters Table lists the most of parameters used in our simulation, and default parameters table presents the default assumption. There is small difference settings between every simulation and it will be described in each part.

3.1 Simulation in Local Wireless Network

I assume the local network is in 1000 X 1000 square areas. Each node (Individual Bluetooth-enabled device) has several properties, such as position, destination position, and speed and infection status. All nodes in this area are mobile, and they move from one position to a random destination position, and when nodes arrive to the destination, they will randomly move to another position and continue this process. During the simulation, each node calculates the distance between itself and other nodes, if it finds one node is within operating range, it adds the neighbor into its neighbor list table. Next, it checks the neighbor's infection status. If the neighbor node is not infected, it will establish connection and transfer the infected file, otherwise, it iterates other neighbors in the table. In the experiment, we simulate the Bluetooth worm's behavior with different parameters, such as speed and operating range. In Bluetooth specification, inquiring and paging sessions find neighbors and then set up connection with them. This procedure is a dominate part in Bluetooth communication. Therefore, we consider two cases, ideal case (discovering neighbors + set up connection time + transfer file time ≈ zero) and real case (discovering neighbors + set up connection time + transfer file time > zero). In addition, contact degree is one factor to be simulated, and there are still two cases, one of which is that contact degree is one and in the other contact degree is up to seven. Except for node's properties, we also simulate different scenarios including different density of the network, different numbers of initially infected nodes.

As we know, one of the important factors of Bluetooth technology is energy, but in our experiment we do not consider the energy model. All of simulations are under the ideal energy model that every node is alive during the simulation. We will discuss the reasons later.

3.1.1 Propagation Speed

The Bluetooth worm has similar properties with other computer worms, which self-replicate as fast as possible. The first experiment simulates propagation speed so that we can study the behavior of the worm spreading process. I use the default assumption to do this experiment and calculate the number of infected nodes per 100 s. In Figure 5, the two sides, the beginning and the end of worm propagation, the spreading speed is low because there are few infected node at the start time and at the end it is difficult to find un-infected neighbors. The important phase is the middle of worm propagation, and the infected nodes significantly replicate themselves to attack vulnerabilities.

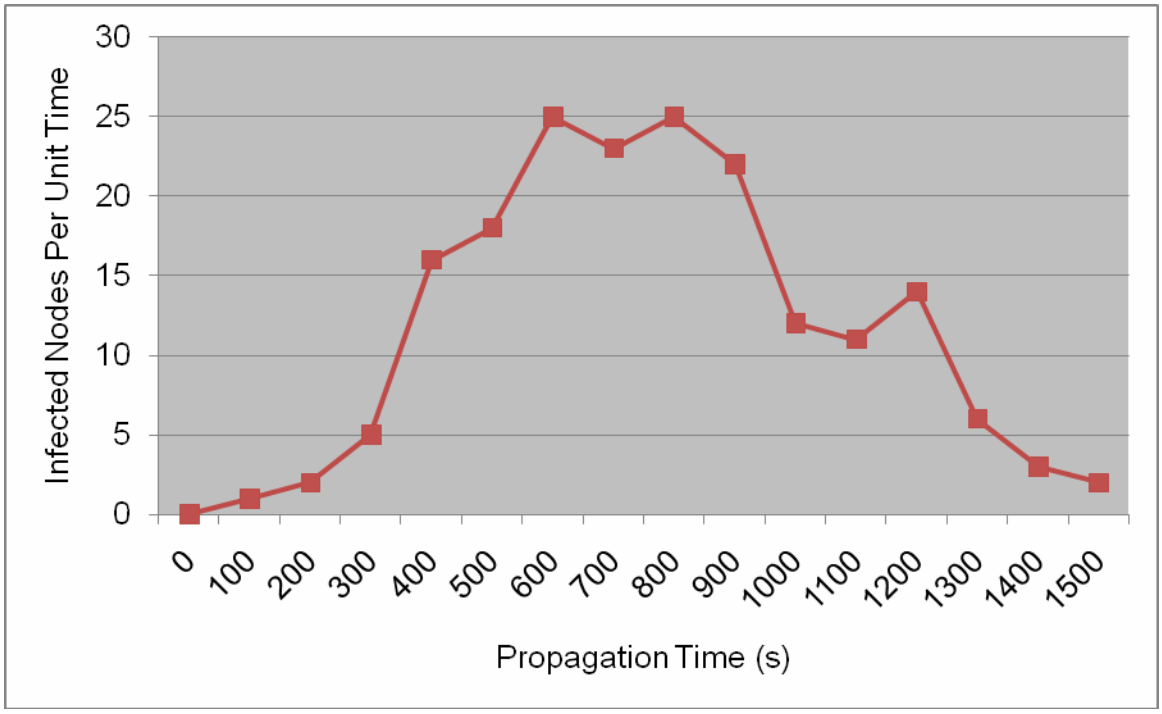

Figure 5 Bluetooth worm propagation speed per unit time

3.1.2 Density Effect

Density of the group is a very important factor. Because we assume the area size is constant, we change the density by putting a different number of nodes in this square area. In our simulation, we chose 50, 80, 100, 130, 150, 180, 200, 230, 250, 280 and 300 nodes in this square area. The properties and movement of each node under default settings are random movement with random speed, and its spreading worm behaviors follow the rules mentioned before.

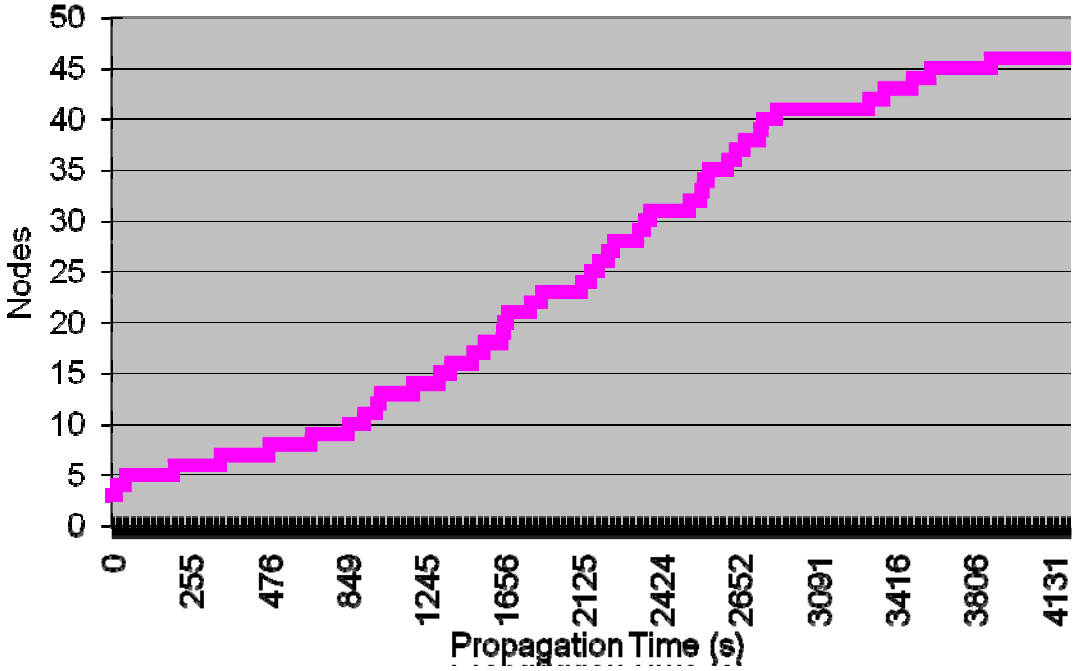

Figure 6 Propagation Curve of 50 Nodes (Infected Rate: 95%)

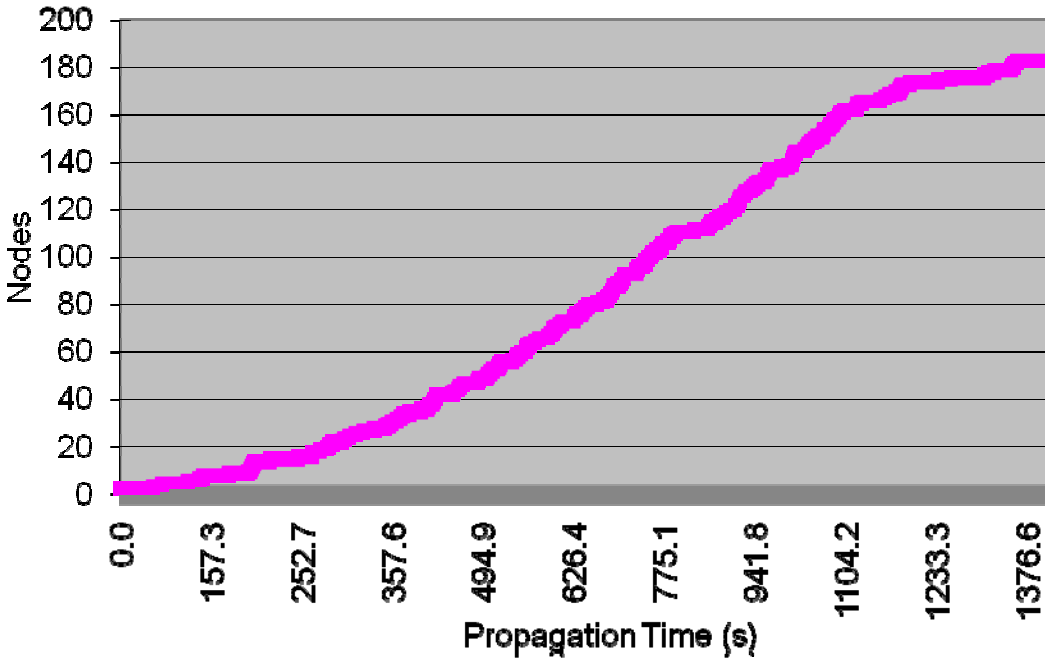

Figure 7 Propagation Curve of 200 Nodes (Infected Rate: 95%)

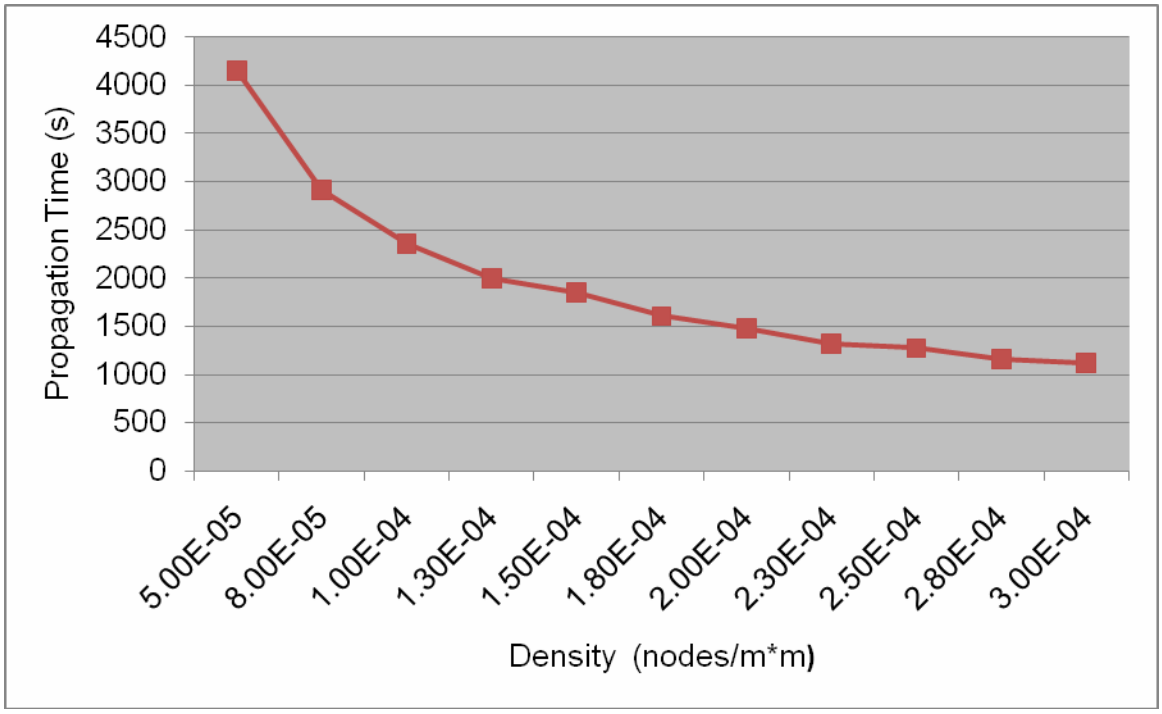

Figure 8 Propagation Time in Different Density

Figure 6 and Figure 7 describe the relationship between the time of worm propagation and infected nodes. The results in Figure5 and 6 are the same as in the first experiment and is the another ways to present worm propagation speed, which is low-spreading speed at the beginning and end of propagation, and in the middle of the worm propagation the spreading process is fast. This behavior is the common property for worms in wired networks. Figure 8 presents a curve between the propagation time and density of the network. Obviously, with increasing the density of the network, in other words, increasing the number of nodes, worm spreading speed is faster and more nodes are infected. When only 50 nodes are in 1000 X 1000 square areas, the density is 5.00E-5. It is a low density value, so it takes over 4000 seconds for malicious nodes to infect 95% of vulnerable nodes within its operating range. The number of nodes

increases to 80, and the propagation time decreases to 2914 seconds. When the number of nodes is 100, the propagation time almost is half of the one in 50 nodes. However, the potential tendency stops when nodes continue increasing. From 200 nodes to 300 nodes, the propagation time is greater than 1000 seconds, and it is hard to continue to decrease the speed of worm propagation.

3.1.3 Effect of the moving speed of Bluetooth device

This paper discusses worm spreading in mobile wireless networks, so the speed of each node is one of the important parameter in our simulation. We set the speed as 1, 2, 8, 15, 20, 25, 30, 35, 40, 45 and 50 m/s; these are eleven different maximum speeds. The speed 1 m/s is to simulate people taking a walk, and speed 2 m/s is regular speed of a person. The maximum speed 8 m/s simulate a running person. From 15 m/s to 50 m/s, they cannot present any real scenarios and our purpose is to find the trend of node's speed argument for worm propagation.

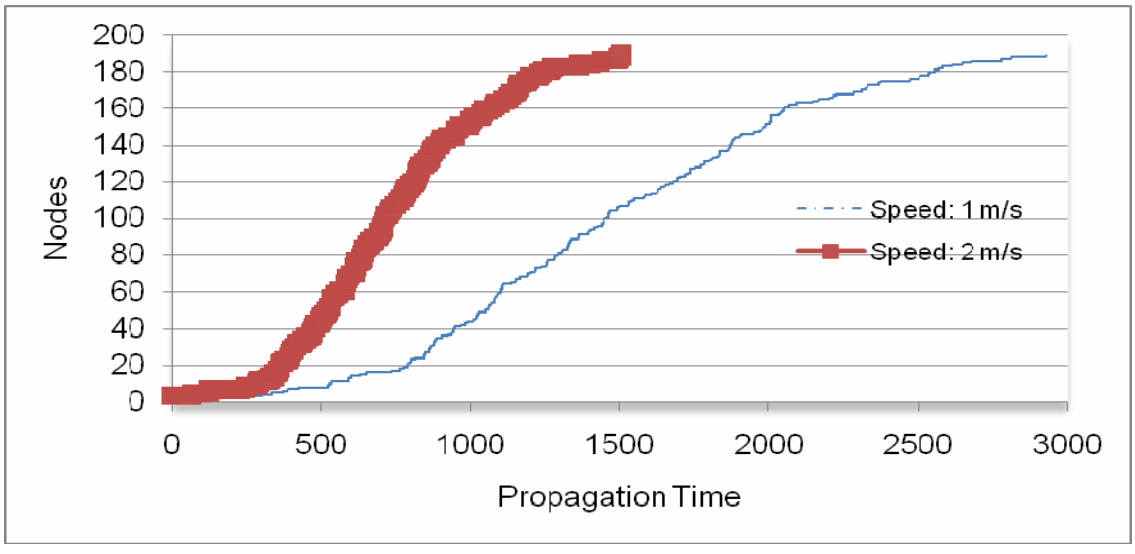

Figure 9 Propagation Curve of Speed: 1 m/s and 2 m/s

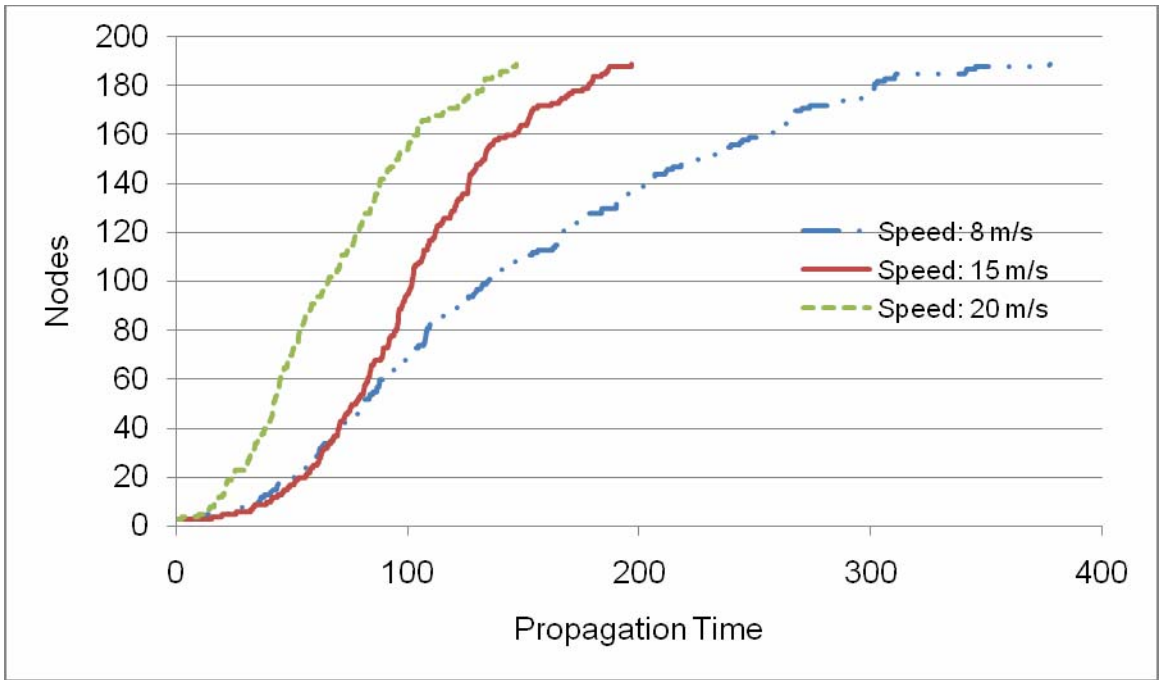

Figure 10 Propagation Curve of Speed: 8m/s, 15 m/s and 20 m/s

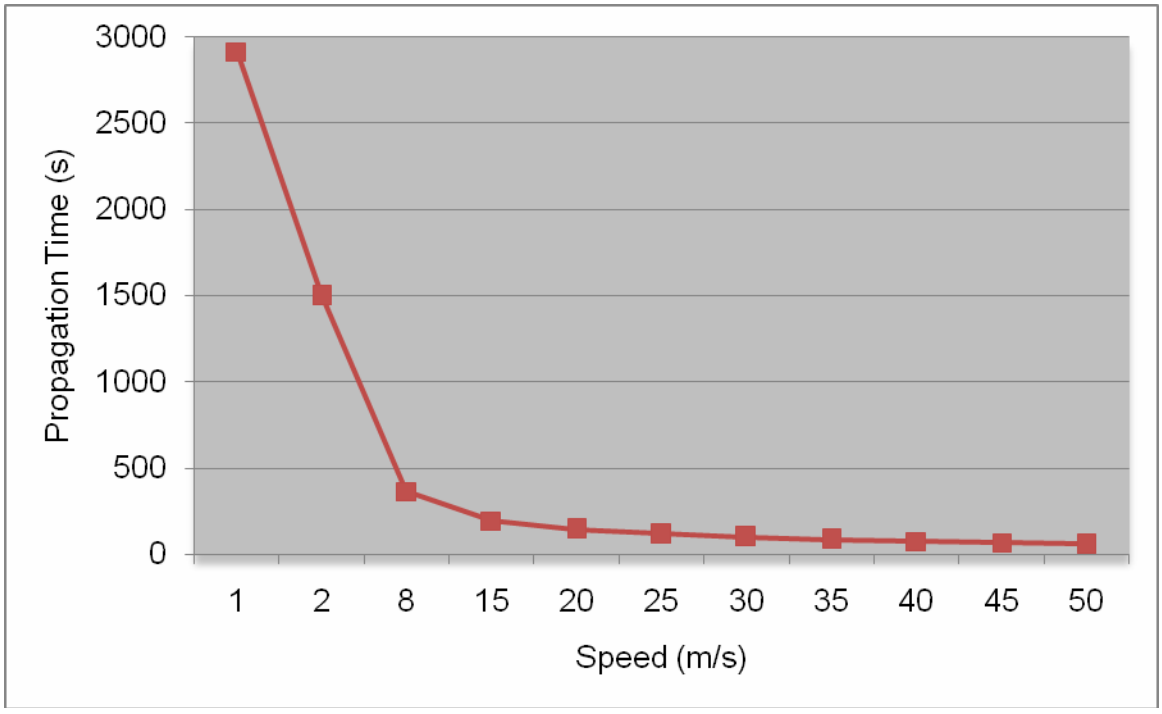

Figure 11 Propagation Time in Different Speed

The key of mobile wireless networks is that nodes can move, hence, the speed indubitably is a major factor. Figure 9 and Figure 10 show the comparing curve among different speeds, and Figure 11 gives us speed and propagation curve. When the speed changes from 1 m/s to 2 m/s and from 2 m/s to 8 m/s, the infected speed has huge increasing. At the beginning of this curve, there is almost a linear relationship between the speed of node and the propagation time. This is very useful for attackers and they can try to move faster in order to accelerate the worm propagation. Above 8 m/s, however, the speed seems not be the key parameter to affect the worm propagation. Although the propagation time still decreases with the speed increasing, the value is too small to help attackers to infect rapidly. Therefore, the speed in a certain range (e.g. smaller than 8 m/s) can let the large number of nodes infected in mobile wireless network.

3.1.4 Operating Range effect

Whenever you use any kind of wireless technology, all of them have the operating range. Only two nodes within the operating range can set up connection and transfer files with each other. Once out of the operating range, they have no any relationship between them. With the development of wireless technology, operating range could extend to a wide area.

Bluetooth technology has three kinds of operating range depending on the device class:

Class 3 radios – have a range of up to 1 meter or 3 feet;

Class 2 radios – most commonly found in mobile devices – have a range of 10 meters or 30 feet;

Class 1 radios – used primarily in industrial use cases – have a range of 100 meters or 300 feet.

In this experiment, we simulate all three standard operating ranges to predict the effects in the future. We set 200 nodes in 1000 X 1000 square areas, the maximum speed is 2 m/s and the number of initially infected nodes are 3.

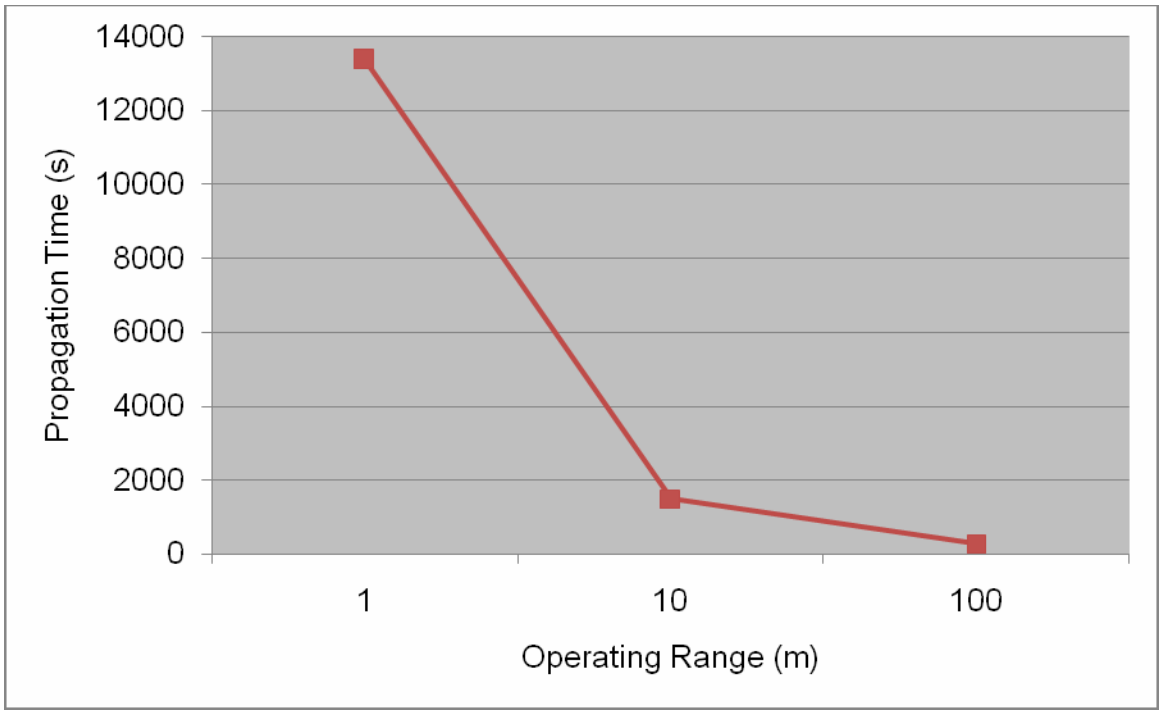

Figure 12 Propagation Time of Range Effect

Simulation result:

-- Operating Range: 1, Propagation time: 13382.9 s

-- Operating Range: 10, Propagation time: 1488.65 s

-- Operating Range: 100, Propagation time: 274.41 s

Except for the speed of the node, the operating range is another unavoidable factor in Bluetooth networks. In modern society, there is the great number of Bluetooth devices, and most users use Class 2 (10 meter operating range). When the operating range is 1 meter, attackers take over 3.7 hours (over 13,000 seconds) to infect 189 nodes. If we use Class 2 standard operating range, attackers only take 1/10 time to infect the same number of nodes. This is an unbelievable decrease, but this is not the end of road. In the future, Bluetooth device would change to Class 3 (operating range is up to 100 meters), and our experiment shows the propagation time is below 300 seconds to infect 189 nodes.

The operating range increases 10 times from 1 meter to 10 meters, and the propagation time decrease 10 times as well. Although the propagation time only decreases 5 times from Class 3 to Class 2, the worm-spreading process is so fast that the network security engineer cannot response to it before huge damage. Therefore, the Bluetooth worm is a potential threat for next-generation Bluetooth device.

3.1.5 Initial Infected Nodes

The price of Cell phones, smart phones and PDAs never stops decreasing because manufactures continually develop new model devices, and it is very common for a person to hold several Bluetooth devices. Therefore, attackers could use several Bluetooth devices as seeds to start worm propagation at the same time. The different numbers of initially infected nodes affect the speed of the worm propagation, and in Figure 13 we can clearly understand this effect.

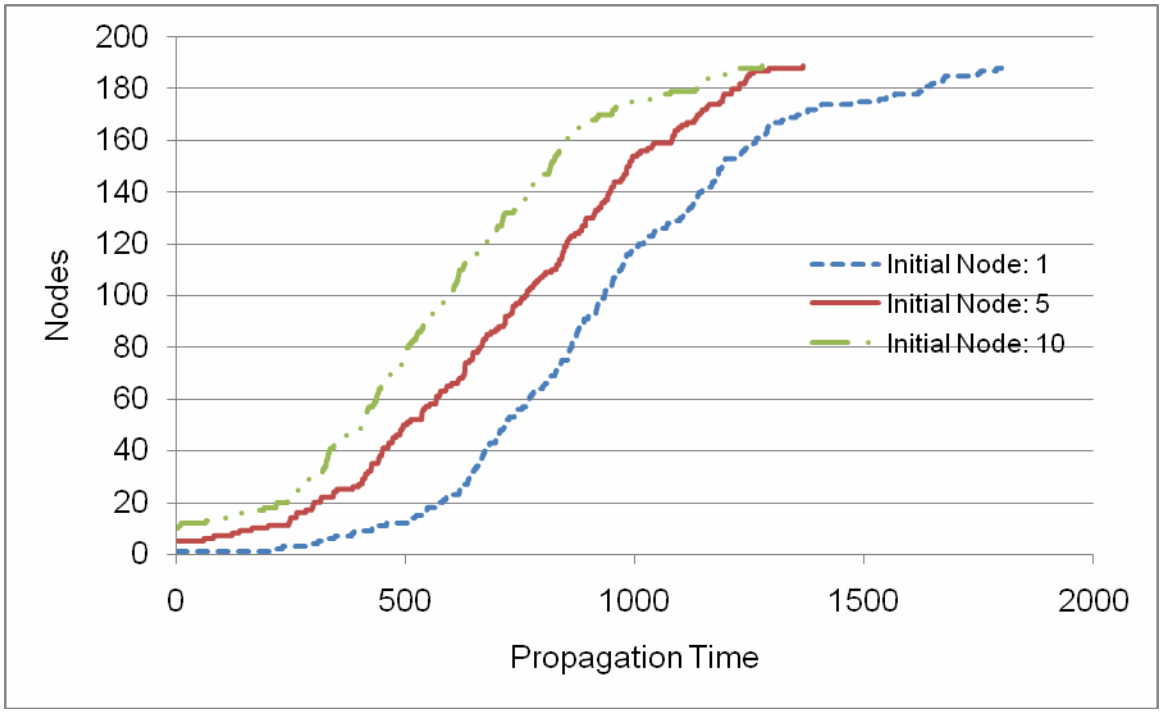

Figure 13 Propagation Curve in Different Initially infected Nodes: 1, 5, 10

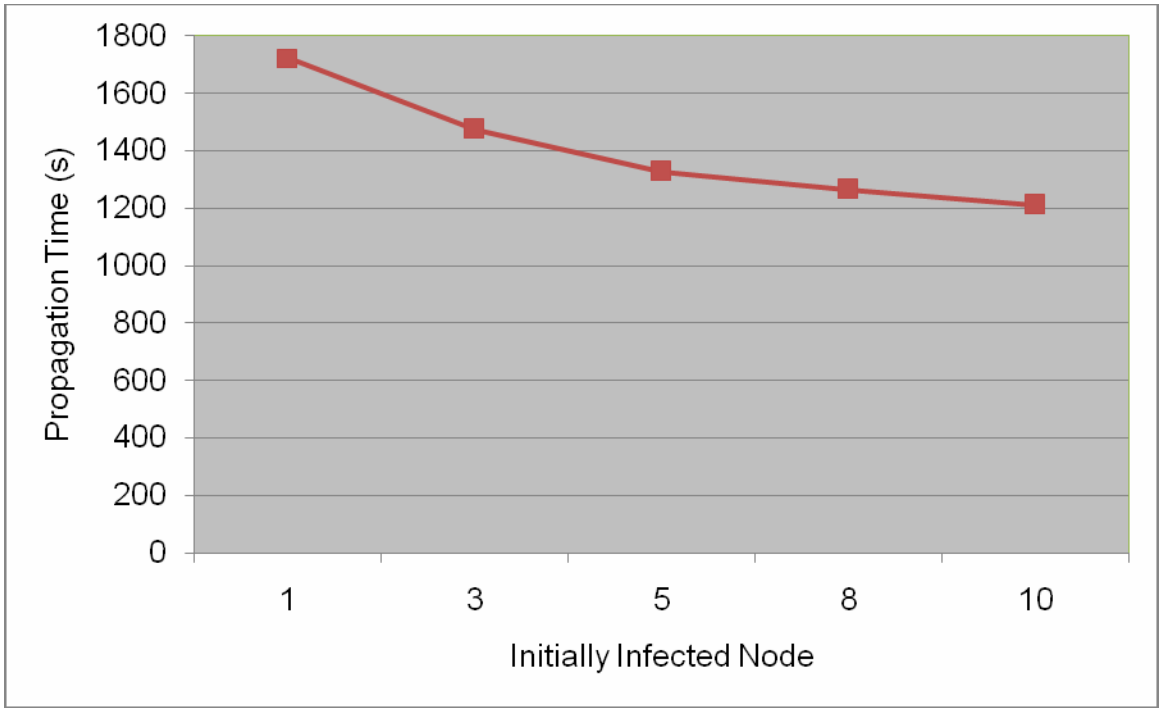

Figure 14 Initially infected Nodes and Propagation Time

From Figure 14 the number of initially infected nodes is set from 1 to 10, and the propagation time becomes short when we add initially infected nodes, but the decreasing value is not too big. The primary reason is that when the number of initially infected nodes increases, the opportunity of discovering vulnerable nodes decreases. We can explain it by the worm propagation figure. When the worm infects over 90% of nodes in the simulation area, the speed of worm propagation begins to decrease significantly, so it is hard for an infected node to find a healthy, vulnerable neighbor node. Therefore, increasing the number of initially infected node only helps worm spreading at the beginning stage, while the most nodes are infected, the worm spreading changes to low value. Above all we can know the number of initially infected nodes is not a key factor of the worm propagation.

3.1.6 Contact Degree

In this section, we consider the contact degree of the Bluetooth device. We assume that a Bluetooth device can set up only one connection with another Bluetooth device, which means at one unit time attackers just infect one Bluetooth device. In previous simulations, there was no such limitation. In this simulation, the contact degree is set to 1. If an attacker finds several vulnerable neighbors, it only connects the one neighbor with minimum distance between itself and them. In addition, in real wireless networks, not all cell phones or PDAs have Bluetooth function. Because the devices without Bluetooth function are immune to the Bluetooth worm, the 20% of cell phones without Bluetooth function cannot be infected. We call 20% as the Healthy Rate. In this experiment, we simulate 0%, 20%, 40% and 60% Healthy Rate, respectively. Figure 15

shows a relationship between the propagation time and the Healthy Rate. Figure 16 compare two different contact degree. The basic arguments: Nodes 200, Speed 2m/s, initially infected Nodes 3, Range 10m.

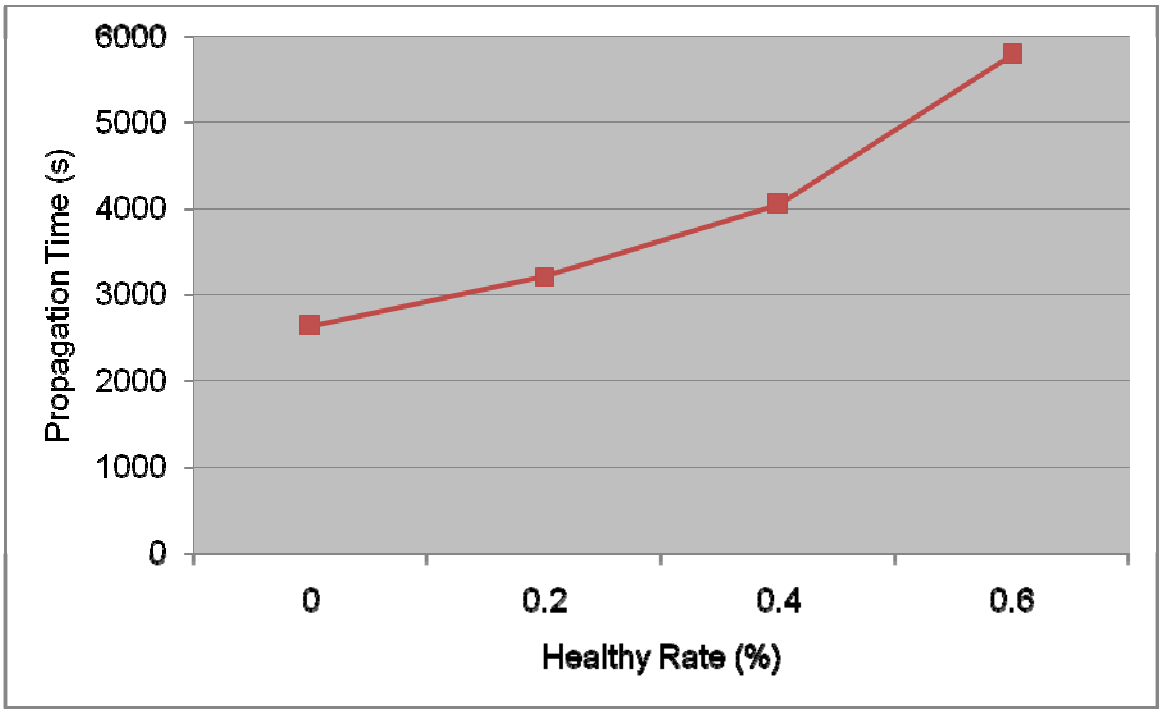

Figure 15 Propagation Time and Healthy Rate (200 Nodes, Speed 2m/s)

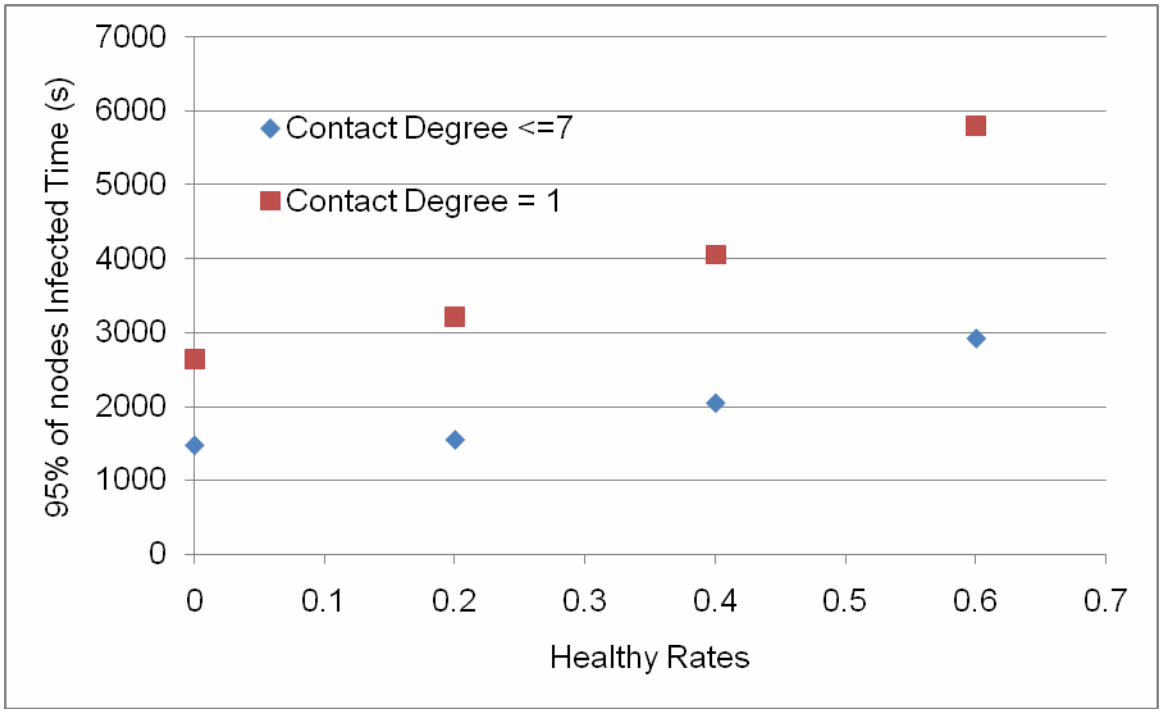

Figure 16 Healthy Rate and Propagation Time (95% nodes are infected)

We can clearly find that if the Healthy Rate is high, the worm has to take more time to infect vulnerable nodes. Originally, when the healthy rate increases so as to decrease the total number of vulnerable nodes, it seems that the worm should take less time to infect a smaller portion of nodes. On the contrary, the worm needs to take more time to spread itself in the networks. In Figure 16, we compare two scenarios, such as contact degree = 1 and contact degree <= 7, and it presents High contact degree can improve the speed of the worm propagation.

3.1.7 Inquiry Time Effect

In the density, speed and initially infected nodes simulation, all experiments are under the ideal model. In the previous simulation, we consider a constraint, the contact degree of each node, but all of them assume no time to use for inquiring and setting up

connection. A Bluetooth device, however, cannot ignore the above parameters. Normally, both processes of inquiring and setting up connection should take several hundred mms, even several seconds. We simulate the inquiry time as 0, 0.5,1, 1.5 and 2 seconds. Then in second simulation, we assume each node takes 1 second to inquiry neighbors, set up connection and finish the infected file transmission, but we change the node's speed.

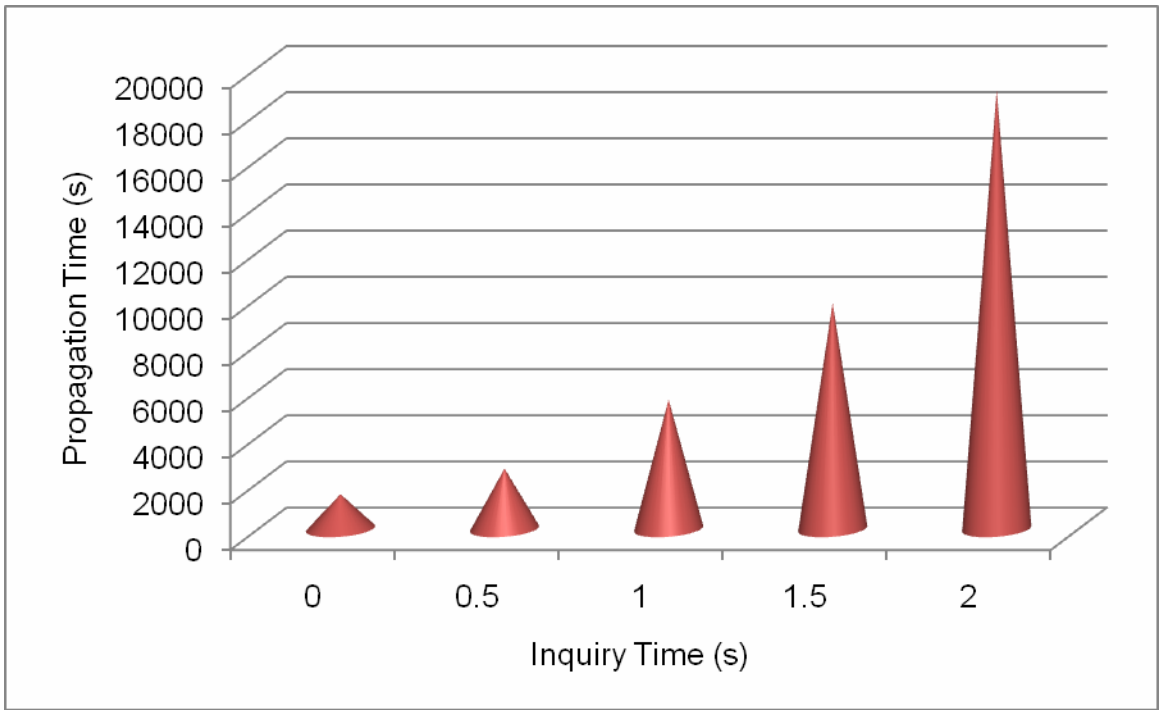

Figure 17 Inquiries Time and Propagation Time (Speed 2 m/s)

In Figure 17, if the inquiry time is more than 0.5 second, the propagation time is over two times longer than in the previous inquiry time experiment. When inquiry time is 2 seconds, the malicious node needs to take 5 hours to infect 95% of nodes in the Bluetooth networks. Because every node always moves from one place to another and the operating range is 10 m/s, sometimes a malicious node discovers a vulnerable

neighbor, but both nodes may separate before the malicious node infect the vulnerable node. This kind of case often happens in the real world. Therefore, the inquiry time limits the worm propagation in Bluetooth networks.

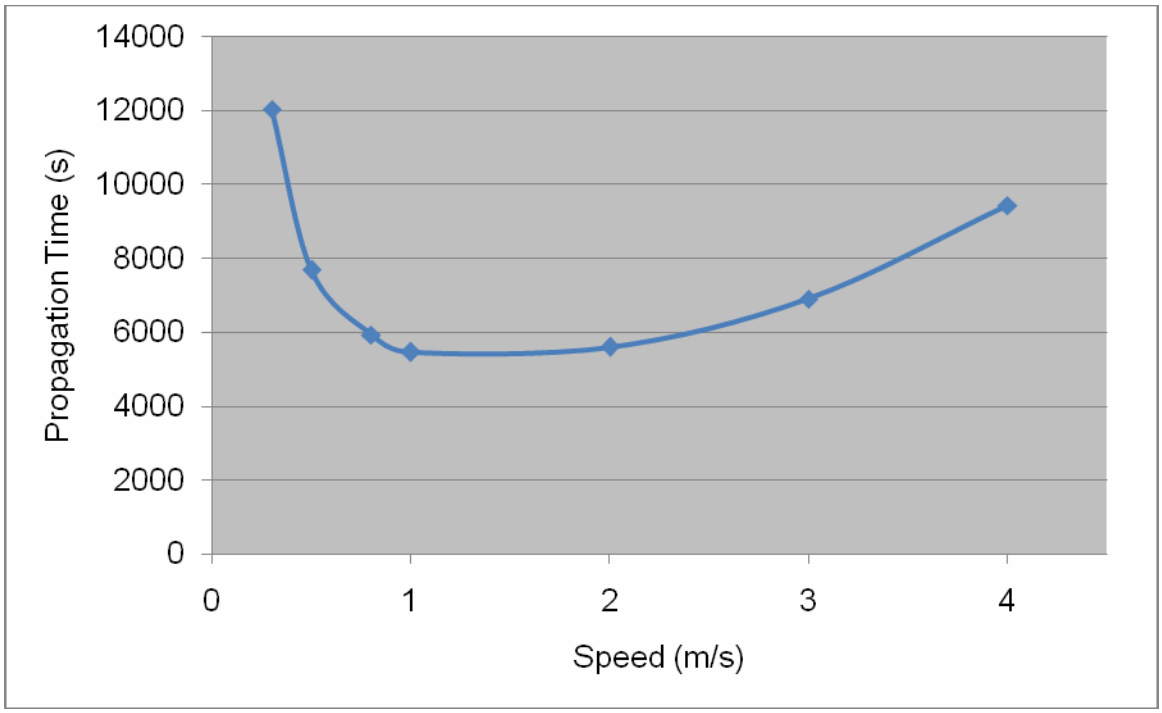

Figure 18 Speed and Propagation Time (Inquiry Time = 1s)

Figure 18 gives us an interesting curve which is an inverted parabola shape. We set the node's speed as 0.3, 0.5, 0.8, 1.0, 2.0, 3.0 and 4.0 m/s when the inquiry time is 1 second. From 0.3 to 1 m/s, the propagation time decreases, however, from 1 to 4 m/s, malicious nodes use more time to propagate. From the previous speed simulation, we know the node's speed is one of the most important arguments for worm propagation. When we increase speed from 1 to 8 m/s, the propagation time is decreased greatly. In this simulation, however, we get an opposite result and the only difference is that we take the inquiry time argument into consideration. Analyzing the curve, we can find the

reason is the same as the effect of changing inquiry time. Because operating range is a constant argument, if we increase the node's speed, the operating time between two nodes decreases at same time. From this simulation, we learn increasing the node's speed is not a necessary condition for accelerating worm propagation.

The lowest point in this curve is 1 m/s. In this simulation, we also calculate the two values, the average targets per neighbors (TPN) and the average targets per propagation time (TPT). We record the number of targets in its neighbors and the number of targets in each node.

Table 3 TPN and TPT

Speed (m/s)	TPN	TPT
0.8	0.142	0.066
1.0	0.125	0.074
2.0	0.069	0.076
3.0	0.038	0.058

The faster the speed of the Bluetooth device, the more neighbors the Bluetooth device finds. This was true in the previous speed simulation. However, it changes if we add the inquiry time into simulating parameter. When the speed increases from 0.8 m/s to 2.0 m/s, the Bluetooth device can find more vulnerable targets per unit time. But once the speed is more than 3.0 m/s, TPT goes to the opposite direction. That is the reason for the shape of the curve in Figure 18. What is the lowest point? It is the point with speed 1.0 m/s, but the TPT under speed 1.0 m/s is smaller than the TPT under speed

2.0. We also need to check TPN in this case. The lower speed (1.0 m/s) infects more neighbors to vulnerable targets, but the faster speed (2.0 m/s) cannot successfully infect its neighbors because its neighbors do not stay for enough time within operating range.

3.1.8 Co-Channel Interference and Failure Rate

Many environmental factors could cause failure. Sometimes two Bluetooth devices terminate communication during paring states so as not to be connected, and sometimes connection has been set up, but transferring the file failed. This thesis focuses on co-channel interference [26] in Bluetooth networks from peripheral Bluetooth devices located in proximity of the ten-meter range. The probability of interference increases as each new user connects to a Bluetooth network within the ten-meter range. Bluetooth efficiency can suffer a drastic drop when too many Bluetooth devices are active in a small area due to collisions. Data collisions require retransmits and with retransmits speed or data throughput is degraded. Therefore, co-channel interference increases the failure rate as well. We assume different failure rates with different numbers of neighbors. If a Bluetooth device finds one neighbor and no others, we set the Failure Rate as 0; if there are 1 to 2 other neighbors within their operating range, the failure rate is 15%; when there are above 2 other neighbors, the Failure Rate is 25%.

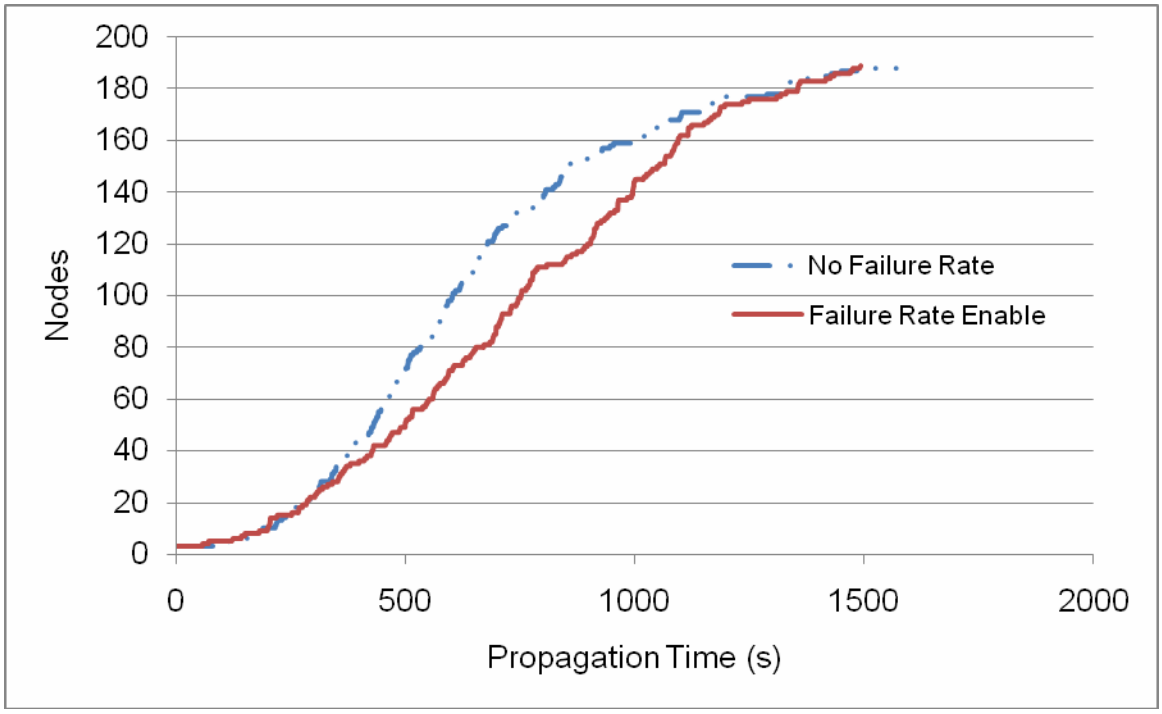

Figure 19 Propagation Curve in Failure Rate and No Failure Rate

Simulation result:

-- No-failure (ideal): Propagation time: 1475.52 s

-- Failure-enabled: Propagation time: 1563.85 s

There is no big deference between the No-failure and Failure-enabled in our experiment. Although both simulations use almost the same time to infect the same number of vulnerable nodes, the Failure-enabled scenario still makes the spreading speed slow. In Figure 19, in the middle of the propagation process, a solid line takes longer time to propagate the same number of nodes as a two-dot line. In crowded places, e.g. office cubicles, airports and movie cinema, there exists high-density Bluetooth devices within ten-meter range. The co-channel interference would be a

serious problem because of the high failure rate. In this situation, co-channel interference helps us to stop worm propagation and decrease the spreading speed.

3.1.9 Speed and Inquiry Time Combination

In the previous simulation, we considered speed and inquiry time separately. Now we take two arguments into consideration at the same time. Speed could impact the inquiry time, and a user may take more time to set up connection than another with lower speed. In this simulation, there are three assumptions as below.

- Speed: 0 ~ 1.0 m/s, Inquiry time: 0.5 s
- Speed: 1.0 ~ 2.0 m/s, Inquiry time: 1.0 s
- Speed: > 2.0 m/s, Inquiry time: 1.5 s

Table 4 Speed and Inquiry Time

Speed (m/s)	Propagation Time (s)
1	3590.89
2	3917.62
5	23643.3

Obviously, when the speed is over 5 m/s, and propagation time increase significantly. Compared to the previous experiment (only considering speed argument), attackers just use 1000 seconds to spread a worm. Therefore, so far, in a real wireless environment, it is hard for an attacker to rapidly implement Bluetooth worm propagation.

3.1.10 Energy Issue

Bluetooth technology is designed to have low power consumption. The Bluetooth device is classified into three power classes. Table 5 describes the specification in detail.

Table 5 Power Class

Power Class	Maximum Output Power (Pmax)	Minimum Output Power
1	100 mW (20 dBm)	1 mW (0 dBm)
2	2.5 mW (4 dBm)	0.25 mW(-6 dBm)
3	1 mW (0 dBm)	N/A

.

The most commonly used radio is Class 2 which uses 2.5mW of power. Actually, there are no Bluetooth devices, and there are only Bluetooth-enabled devices. These include Bluetooth headsets, Bluetooth-enabled laptops, Bluetooth-enabled PDAs or Bluetooth input devices. In some Bluetooth systems (e.g. laptops and appliances using AC power), the Bluetooth will not be a noticeable drain on the system. However, in some cases, the Bluetooth will dominate current consumption in a device. This is especially true for simple devices such as a Bluetooth headset or a Bluetooth mouse.

Can the Bluetooth-enabled device work without charging? We choose headsets to study the energy lifetime of Bluetooth devices because they completely use Bluetooth function without other primary power consumptions. According to market products, Bluetooth headsets typically offer 2-10 hours talking time or 25-250 hours standby. In this simulation, we do not consider the standby case, and we assume attackers continue to scan neighbors and transfer the infected file without time interval. From the experiment result, the worst case is that a malicious node takes more than 5 hours to

infect 95% of vulnerabilities in Bluetooth networks. However, in most of my simulations, the propagation time is below 1.5 hours (5000 seconds). It is possible for attackers to infect all vulnerabilities before the power of Bluetooth-enabled device is used up. If we use a Bluetooth-enabled device with AC power, there is no necessary to consider energy issue.

In the future, when Bluetooth technology uses Power Class 1 that is just 1 mW output power, the Bluetooth-enabled device has longer lifetime and can provide attackers more chance to spread the Bluetooth worm. Therefore, in this thesis, we don't take care of the Bluetooth energy (power) issue.

3.2 Simulation in Wide Wireless Network

In the real world, human activities are not just in a local group. Usually, people go to work at company or school, and at weekend they go to movie cinema or park, even from one city to another or one country to another. They always change groups or places and not belong to a static group. Therefore, in wide wireless network simulations, we assume some nodes in one group will transfer to another, and the wide network simulation provides different density groups. We also use the Poisson distribution as the transfer model, and assume that each group has a fix departure rate. For each group, it runs the local wireless simulation. The difference between wide networks and local networks is that the density of each group is variable during the simulation.

As above mentioned, we simulate five kinds of scenarios related with real people activities. There are 20 groups with different nodes, such as 50, 100, 150 and 200.

-- Node Transfer scenario: People travel among different groups.

-- Without node transferring scenario: All activities happen in local groups, and people do not go out of their groups.

-- Add New Group scenario: Sometimes people could go to the one place at same time, for example, movie cinema, national park, or arena. In this simulation people are random selected from random groups to build a new group, and we assume there are at least 3 infected nodes in the new group.

-- Remove New Group scenario: On the contrary, when the movie and game are over, people go to different places. We simulate one group dismisses and people in this group randomly join into other exist groups.

-- Adding and Removing Group scenario: This case assumes two events, group adding and group removing, happen in one simulation. We assume they would not happen at the same time as well.

Table 6 Large Scale Simulation

Event	Nodes	Propagation time
Add	2138	850
Without Transfer	1126	2000
Add And Remove	1640	2000
Only Transfer	2055	2000
Remove	2138	2000

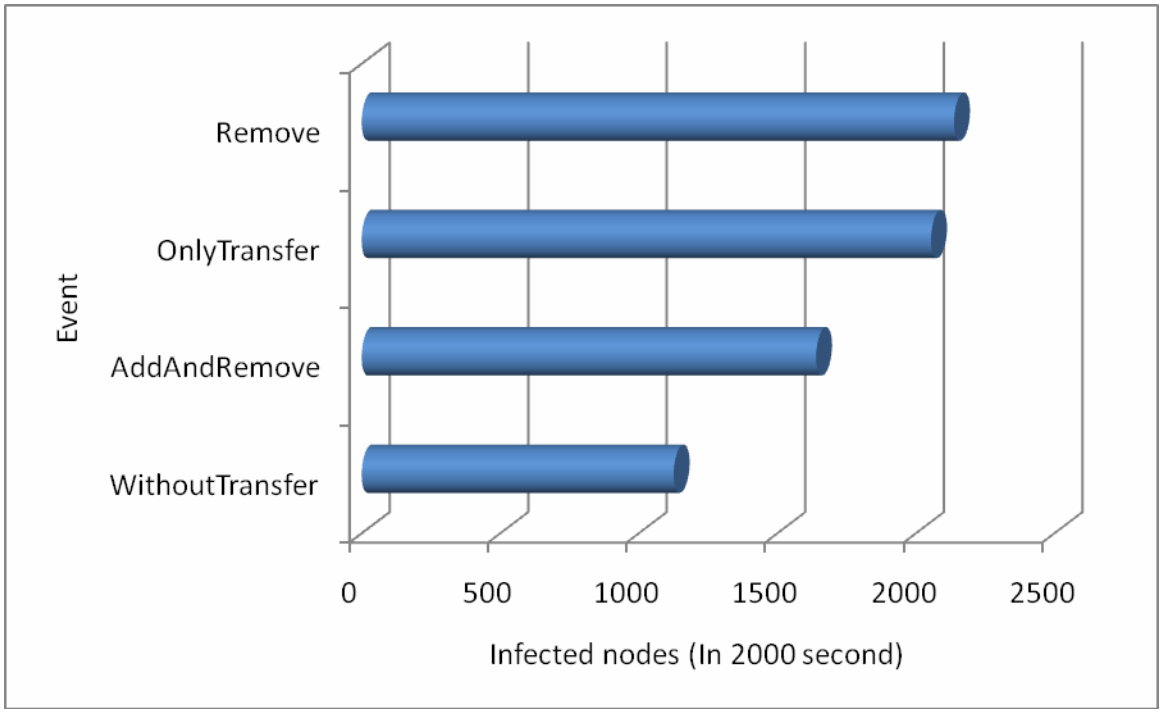

Figure 20 Large Scale Simulation (Max Simulation Time: 2000 s)

In Figure 20, we learn that the Adding new group event is the most helpful to accelerate the worm propagation, and the next event is the Removing group event. And then are only transferring event, adding and Removing Event. The last one is the event without any transferring activity among groups. Those results tell us mobile property is the most important factor for Bluetooth worm propagation. Even in wide scale wireless environment, mobility in different groups still is a positive impact for the worm spreading.

CHAPTER 4: DEFENSE

Product developers that use Bluetooth wireless technology in their products have several options to implement security issues. There are three modes of security for Bluetooth access between two devices.

Security Mode 1: non-secure

Security Mode 2: service level enforced security

Security Mode 3: link level enforced security

The manufacturer of each product determines these security modes. Devices and services also have different security levels. For devices, there are two levels: "trusted device" and "un-trusted device." A trusted device, having been paired with one's other device, has unrestricted access to all services. With regard to services, three security levels are defined: services that require authorization and authentication, services that require authentication only and services that are open to all devices.

Consumers can do a number of things to protect their data. If users have a phone that is vulnerable to Bluetooth virus, they should contact the phone's manufacturer to get developed software patches to fix the vulnerability. In addition, if users are still concerned about a device being targeted, they can turn the device to non-discoverable mode when not using wireless Bluetooth technology and in unknown areas. Users can also ensure their data is secure by not "pairing" with unknown devices. If a user were to receive an invitation to pair with another device, and asked to put in a PIN code, but was unsure of what device was inviting to pair, the user should not pair, and only pair with known devices.

However, attackers still can find the address of the user even if user just pair with trusted devices. He can overhear the initial pairing process between two Bluetooth-enabled devices, and then use brute-force or other methods to guess the security key [28] and masquerade as the second device during a connection. If successfully setting up connection. It can transfer an infected file to implement worm propagation. According to above simulation, in mobile network, attackers need to guess the security key very quickly within the 10-meter operating range. If the PIN code is from manufacture that is just four digital, it is very easy for attackers to spread worm in few seconds. If users set the PIN code more complex, attackers has to take hours even several days to find the right PIN. Hence a complex PIN code is one way to prevent Bluetooth worm.

All computer worms including the Bluetooth worm have a same property, spreading rapidly. It is un-normal behavior in networks. Also, in our simulations, we find the co-channel interference is a negative factor for the Bluetooth worm spreading. Therefore, one defense method is to set a monitor and interference system in popular places since the Bluetooth worm attacking is implemented mostly in this kind of place. When the system finds a suspect device that continually pairs and transfers files to others, firstly the system sends a warning message to the virus center for further analysis. Meanwhile, the system tries to response all pairing quests in order to generate a co-channel interference problem in this wireless Bluetooth network. This process reduces the spreading speed of the Bluetooth worm so as to let the virus center develop patches to prevent the worm propagation in time.

CHAPTER 5: CONCLUSION

Computer worm spread by Bluetooth technology still is a new way in the wireless network, and so far it does not bring the huge damage for wireless networks. Since all Bluetooth worms need be activated by a manual operation, it cannot spread rapidly. However, Bluetooth technology will be growth quickly in order to communicate in larger range, to transfer bigger size package and more rapidly, and smart phones will become a mobile micro device with full computer's functions. Those new techniques will boost the Bluetooth worm propagation in wireless networks.

To reduce the Bluetooth worm damage, even to predict the Bluetooth worm, this thesis studies its behaviors based on new simulator – BTWS. The mobility of Bluetooth-enabled devices in Mobile wireless network is the primary effect of the worm spreading, and if attackers control their speed at 1 or 2 m/s so as to achieve the maximum worm propagation. The Inquiry time is another key feature of Bluetooth technology, and it determines whether or not the worm can infect vulnerable devices. If the new-generation Bluetooth technology can significant reduce the inquiry time, it will be easy for Bluetooth devices to communicate to each other, but it is also a disaster for the wireless network security. In the large wide wireless environment, attackers could change its location in order to spread the worm quickly. For network security issue, it is hard for security engineer to find the source of the worm due to its mobility. In the future, we can build a worm propagation model to further study its behavior in order to improve the defense system.

REFERENCES

[1] Guanhua Yan and Stephan Eidebenz. Bluetooth Worms: Models, Dynamics, and Defense Implications. In the 22nd Annual Computer Security Applications Conference (ACSAC), 2006.

[2] James W. Mickens and Brian D. Noble. Modeling Epidemic Spreading in Mobile Environments. In the international conference of Web information systems engineering (Wise), September 2, 2005

[3] Abhijit Bose, Scott Boehmer and Kang G. Shin. On Mobile Viruses Exploiting Messaging and Bluetooth Services. In Securecomm and Workshops, 2006

[4] Jing Su, Kelvin K. W. Chan, Andrew G. Miklas, Kenneth Po, Ali Akhavan, Stefan Saroiu, Eyal de Lara†, Ashvin Goel. A Preliminary Investigation of Worm Infections in a Bluetooth Environment. In The 4th ACM workshop on Recurring malcode, 2006.

[5] Computer Worm. http://en.wikipedia.org/wiki/Computer_worm.

[6] Peter Denning. Computers Under Attack: Intruders, Worms, and Viruses. ACM Press, 1990.

[7] Eugene Spafford. The internet worm program: An analysis. Computer Communication Review, 19(1), January 1989.

[8] Moore, David; Colleen Shannon. The Spread of the Code-Red Worm (CRv2). CAIDA Analysis. Retrieved on 2006-10-03.

[9] ANALYSIS: .ida "Code Red". eEye Digital Security Company. EWormhttp://research.eeye.com/html/advisories/published/AL20010717.html

[10] David Moore, Colleen Shannon, and k claffy. Code-red: a case study on the spread and victims of an Internet worm. In The Second Internet Measurement Workshop, pages 273–284, November 2002.

[11] Netcraft. The Netcraft Survey, http://www.netcraft.com.

[12] Stuart Staniford, Vern Paxson, and Nicholas Weaver. How to Own the Internet in Your Spare Time. In Proceedings of the 11th USENIX Security Symposium. USENIX, August 2002.

[13] CC Zou, DW Gong. Computer Communications and Networks, 2004. ICCCN 2004.

[14] F-Secure Virus Descriptions: Antimarc. http://www.f-secure.com/v-descs/antimarc.shtml

[15] Lidong Zhou, Lintao Zhang, Frank McSherry, Nicole Immorlica, Manuel Costa, and Steve Chien. A First Look at Peer-to-Peer Worms: Threats and Defenses. Proceedings of the IPTPS, 2005

[16] 2005 FBI Computer Crime Survey. http://www.cpppe.umd.edu/Bookstore/Documents/2005CSISurvey.pdf

[17] Worm.SymbOS.Cabir.a. http://www.viruslist.com/en/viruslist.html?id=1689517

[18] Virus.WinCE.Duts.a. http://www.viruslist.com/en/viruslist.html?id=1874404

[19] Worm.SymbOS.Comwar.a. http://www.viruslist.com/en/viruses/encyclopedia?virusid=75541

[20] C Bisdikian. An overview of the Bluetooth wireless technology IEEE COMMUN MAG, 2001

[21] Core Specification v2.0 + EDR. http://www.bluetooth.com/NR/rdonlyres/1F6469BA-6AE7-42B6-B5A1-65148B9DB238/840/Core_v210_EDR.zip

[22] BlueHoc: Bluetooth Performance Evaluation Tool. http://bluehoc.sourceforge.net/

[23] Blueware: Bluetooth Simulator for ns. http://nms.lcs.mit.edu/projects/blueware/software/

[24] The Network Simulator - ns-2. http://www.isi.edu/nsnam/ns/

[25] UCBT - Bluetooth extension for NS2 at the University of Cincinnati. http://www.ececs.uc.edu/~cdmc/ucbt/

[26] J. P. Lynch Jr. Co-channel interference in Bluetooth piconets. Master's thesis, Virginia Polytechnic Institute and State University, 2002.

[27] C. Guo, H. J. Wang, and W. Zhu. Smart-phone attacks and defenses. In Proceedings of HotNets III, November 2004

[28] O. Whitehouse. Bluetooth: Red Fang, Blue Fang, 2004. http://www.cansecwest.com/csw04/csw04-Whitehouse.pdf.

[29] N. Eagle and A. Pentland. Reality Mining: Sensing Complex Social Systems. Journal of Personal and Ubiquitous Computing, June 2005.

[30] J. Kephart and S. White. Directed-graph epidemiological models of computer viruses. In Proceedings of the IEEE Computer Symposium on Research in Security and Privacy, pages 343–359, May 1991.

[31] Vidyut Samanta, “A study of mobile messaging services,” UCLA Master’s Thesis, 2005.

[32] Mike Foley. Stacking Up High-speed Bluetooth Against Certified Wireless USB. http://www.byte.com/documents/s=10114/byt1175264350917/0402b.htm

[33] S. A. Khayam and H. Radha. A topologically-aware worm propagation model for wireless sensor networks. In Proceedings of The 2nd International Workshop on Security in Distributed Computing Systems (SDCS-2005), 2005.

[34] G. E. P. Box and M .E. Muller, “A Note on the Generation of Random Normal Deviates,” Annals Math. Stat, vol. 29, pp. 610–611, 1958.

[35] B. Hoh and M. Gruteser. Computer Ecology: Responding to Mobile Worms with Location-Based Quarantine Boundaries. In International Workshop on Research Challenges in Security and Privacy for Mobile and Wireless Networks, 2006.

[36] A. Wagner, T. Dubendorfer, B. Plattner, and R. Hiestand, “Experiences with Worm Propagation Simulations,” 10th ACM CCS Workshop on Rapid Malcode (WORM '03), 2003.

[37] N Srl. Studying Bluetooth Malware Propagation. csdl.computer.org

[38] B Dwan. The mobile phone virus. Network Security, 2004

[39] Iain Thomson. Mosquito Trojan set to infect mobiles. Vnunet.com, Aug 2004

[40] John Leyden. Dampig Trojan menaces Symbian mobiles. http://www.theregister.co.uk/2005/03/07/dampig_symbian_trojan/

[41] Trojan.SymbOS.Locknut.a. http://www.viruslist.com/en/viruses/encyclopedia?virusid=73046