Alquds University
Deanship of Graduate Studies
M. Sc. In Computer Science


## Thesis Approval

# Global Early Detection Contentment System

Prepared By: Manal Mahmoud Omar Tamimi
Registration: 20311744

Supervisor: Dr. Badie Sartawi

Master thesis submitted and accepted, Date: 24/2/2007
The names and signatures of the examining committee members are as follow:

| | | |
|---|---|---|
| 1-Head of Committee | Dr. Badie Sartawi | Signature: ..... |
| 2-Internal Examiner | Dr. Nidal Kafri | Signature: ..... |
| 3-External Examiner | Dr. Bahjat Qazaz | Signature: ..... |

Jerusalem-Palestine

1428/2007

## ملخص

ديدان الانترنت يمكن أن تنتشر بسرعة كبيرة خلال الانترنت. هذه السرعة الكبيرة تجعل من المهم جدا وقف هذه الدودة في أقل وقت ممكن في محاولة لكسب الحرب ضدها.

هذا البحث يثبت أنه من الممكن تصميم نظام تدخل يمكن أن يقطع طريق الدودة السريعة. هذا التدخل يعرف بأنه تقنية الهدف منها إيقاف أو تقليل انتشار الدودة لدى اكتشافها. مجال هذا البحث محدودا بالديدان التي تعتمد على قائمة مسبقة (*Hit List*) والتي تعتمد بروتوكول *UDP*.

تقنية التدخل تقترح توليفة من بنية تحتية والخوازميات اللازمة لعملها. بالإضافة إلى إنذار أولي، تم اعتماد مستوى أعلى من الأمن وسمي طبقة الحراس. الحراس هي عقد متخصصة يمكن أن تعمل بشكل متوازي في الانترنت وترتبط على شكل شجرة. مهمة هذه الطبقة من الحراس تشريع الإنذار الأولي، تفعيل مستوى إنذار ثاني، واتخاذ أفعال للحماية وللتصحيح.

لتحقيق الوقاية يتم سد الثغرات في جميع الأجهزة المستهدفة من الدودة. أما من أجل تصحيح الوضع لدى اكتشاف الاصابة فيتم حجز الجهاز المصاب مؤقتا.

هذا البحث يستعمل طريقة فعالة في كشف وجود الدودة تسمى *destination port matching technique.*

## Abstract

Internet worms can spread very rapidly through the Internet. The fast spreading nature of worms makes it very important to stop them as soon as possible. Such methods will ultimately help in wining the war against worms.

This research proves that it is possible to design an intervention system that can interrupt the spread of flash worm. Intervention is defined as a technique that attempts to stop or limit the worm spread as a result of infection detection. The scope of the research is limited with hit-list worms that use UDP packets.

An intervention technique is proposed that combines both infrastructure and algorithmic techniques. In addition to a first-level warning (router-based), a higher level of security was added, which is called guards layer. Guards are specialized nodes that are distributed evenly around the internet, and connected in a tree structure form. Their job is to validate a first-level warning, generate a second-level warning, and take both preventive and corrective actions.

Preventive action is provided as active patching of all possible victims. In other words, the intervention system will generate a code that aims at patching only machines that are targeted by the worm. The correction action on the other hand is based on temporarily blocking all packets outgoing from an infected node. To

detect a possible worm infection a technique called "destination-port matching" technique was utilized that is very fast and efficient.

## Table of Contents

# Chapter 1

# Introduction

The increasing threat of internet worms makes it a very important research area. Many researchers work in this field. Different results for different techniques are proposed.

This thesis proposes a model that deals with trying to stop or at least minimize the damage that could be caused by one of the most serious worms which is named as a flash worm (Staniford, 2004).

This thesis starts by giving the needed background about the worms in chapter2. Chapter3 illustrates the previous related work. Then in chapter4 the suggested model is described in detail.The detailed design of the model can be found in chapter 5. The results of the implementation with analysis are in chapter 6. Conclusion and the future work are in chapter7.

## 1.1 Worms History

"Worms were first used as a legitimate mechanism for performing tasks in a distributed environment. Network worms were considered promising for the performance of network management tasks in a series of experiments at the Xerox Palo

Alto Research Center in 1982"(Donn Seeley Department).

Worms were first noticed in December 1987,as a potential computer security threat when the Christmas Tree Exec attacked IBM mainframe. The Christmas Tree Exec brought down both the world-wide IBM network and BITNET, but it wasn't a true worm. The Christmas Tree Exec was a trojan horse with a replicating mechanism, which sent an email titled Christmas card to the user that included executable (REXX) code. If executed the program draw a Xmas tree on the display, and it also sent a copy to everyone on the user's address lists.

November 2, 1988 was the date of the release of a true Internet worm. " November 3, 1988 is already coming to be known as Black Thursday. System administrators around the country came to work on that day and discovered that their networks of computers were laboring under a huge load. If they were able to log in and generate a system status listing, they saw what appeared to be dozens or hundreds of "shell" (command interpreter) processes. If they tried to kill the processes, they found that new processes appeared faster than they could kill them. Rebooting the computer seemed to have no effect within minutes after starting up again, the machine was overloaded by these mysterious processes"(Donn Seeley Department). Internet worm attacked Sun and DEC UNIX systems that were attached to the

Internet. It utilized the TCP/IP protocols, common application layer protocols, operating system bugs, and a variety of system administration flaws to propagate. Various problems with worm management resulted in extremely poor system performance and a denial of network service (konczal, 1994).

After a short period, the Father Christmas worm was first appeared and released onto the worldwide DECnet Internet in December of 1988. Father Christmas worm was considered as a true worm, it attacked VAX/VMS systems on SPAN and HEPNET, also it utilized the DECnet protocols and a variety of system administration flaws to propagate. This worm added an additional feature, it reported successful system penetration to a specific site.

"In 1996, the first Word macro virus appeared and became quickly widespread. This was due to two reasons: the far greater tendency for people to exchange documents, as opposed to executables, and the accidental inclusion of the virus on at least two Microsoft CDs. For the most part these were just annoyances, but they showed how the blurring of data and programs could create fertile ground for mobile code.

All this changed in 1999 when the Melissa worm appeared. Unlike previous macro viruses, this one would spread in a semi-active manner. When an infected file was

opened for the first time, it looked through all Outlook address books and sent a copy of itself to the first 50 individuals. This was the first major e-mail worm and it quickly spread around the globe. Although generally benign in intent (the worm itself only sent a small message and the payload was simply a Simpsons joke,) the process of transferring so many messages overwhelmed some e-mail servers. The Melissa worm clearly illustrated the dangers of mixing code and data: items perceived by the user as benign data could contain malware.

After Melissa, mail worms have become annoyingly common, complete with toolkits. There have been some improvements in social engineering (ILOVEYOU and AnnaKornikova showed how proper subject choice can make a difference in the successful proliferation of a worm,) more comprehensive searches for new addresses, included SMTP routines, and payload (Magistr tries to actively disrupt removal and SirCam demonstrated how worms could be used for espionage.) But otherwise these worms are rather unoriginal. Nevertheless, since they continue to spread, they have remained a significant problem and must be considered a major threat.

Active worms have recently returned to prominence. The first one that attracted major attention, Code Red, demonstrated how swiftly a relatively simple worm can

spread on the current Internet infrastructure: it effectively achieved complete infection in a little over twelve hours, even with the aborted early release of a buggy version. Code Red exploited a recently discovered (but patchable) buffer overflow attack in Microsoft's Internet Information Server. It spread far and fast because of the "on by default" nature of IIS with many versions of Windows NT and 2000. It also included multithreaded scanning routines that improve throughput and effectively keep it from being trapped by tarpits (such as LaBrea), which are blocks of IP addresses that attempt to slow down scanning by automated tools by seeming to respond to connection requests while actually doing nothing.

Code Red 2 ended up being significantly more disruptive then Code Red even if the change in infection strategy was relatively mild. Instead of searching only randomly selected addresses, Code Red 2 preferentially probed for machines on the same subnet and nearby subnets. As a result, once a single machine within a corporate firewall was infected, it would quickly probe virtually every machine within the firewall and since it was attacking an on-by-default service, Code Red 2 quickly infested entire corporate networks.

But even before the release of Code Red, active worms were spreading, albeit with less publicity. A good example was the March 2000 release of the 911 worm.

It spread through Windows shares, a slow but autonomous mechanism. It would search portions of the Net for remotely writeable C drives and then inject a copy of itself into the victim's startup routines. The next time the victim computer started up their system, the worm would begin running. Worms that use only this mechanism are comparatively slow spreading because of the need to have the machine rebooted before infection is complete"(Weaver, 2001).

## 1.2 Motivation of this Research

The increasing threat of the worms to the security of the Internet is the real motive to do this research. The researchers in the field worked hard in an attempt to suppress the damage caused by many different types of worms, good results were acheived, but flash worms are still challengeable, their very high speed made it impossible for any human reaction to stop them. This research is an attempt to add an additional weapon to secure the Internet against flash worms.

## 1.3 Contribution to the field

The contribution of this research to the field of Inernet security is the design of an automatic global intervention system that can suppress flash worms' spread as a result of infection detection.The intervention technique combines both infrastructure and algorithmic techniques. Preventive and corrective actions are also applied.

# Chapter 7

# Conclusion

According to our knowledge, this is the first research that attempts to suppress a flash worm. The proposed intervention system combines multiple architecture and algorithmic techniques to cut the worm's way, including: destination-port matching, automatic immunization, worm code analysis, and host blocking.

In addition to effectiveness, our intervention system is efficient both during normal and warning times. The effectiveness lies in applying multiple prevention technique at the same time, e.g. host blocking, immunization, and warning. Efficiency of the system is achieved on multiple levels: detection, warning, and immunization. Port matching detection technique is very efficient, since it requires very easy port number comparison. When an infection is detected by one guard, it will inform other guards in very quick way because guards are connected in a binary tree structure. In addition, detection routers don't block any host until a warning is issued. Thus under normal conditions, routers don't have extra load. More important this interventions system won't flood the network with a self-replicating patching worm, only targeted hosts (spotted in worm hit list) will be patched.

This system employees two levels of detection: router-level and guard-level. This feature prevents false alarms and increase detection accuracy.

Experiments are done using a special developed simulator. The experiments show that if the hit list contains more monitored nodes then there will be more rescued nodes, which emphasise the importance of distributing guards around the world. Also, the experiments show that the level of the first monitored node has a clear effect on the results, so if the detection happend in a high level (closer to the root) then the better the chance to rescue more nodes. It was noticed that in the high levels of the infection tree, the delay is not a big problem, on the other hand, the delay in the bottom of the tree is more serious. One of our results illustrates that the length of the hit list at the detected node has a big effect on the number of rescued nodes, so the longer the hit list at the detection time the more the rescued nodes.

## 7.1 Future work

Other types of worms will be studied. The division of the hit list between the guards will be changed in a way that let any guard patch nodes that are nearest to it.