Seminar Report on Study of Viruses and Worms

H.Shravan Kumar(05329018)
KReSIT,
I.I.T. Bombay.
Email: shravan@it.iitb.ac.in

Guide: Prof. Bernard Menezes.

Abstract— **One of the most high profie threats to information integrity is the *computer virus*. In this paper, I am presenting what are viruses, worms, and Trojan horses and their differences, different strategies of virus spreading and case studies of Slammer and Blaster worms.**

I. Introduction

The internet consists of hundreds of millions of computers distributed around the world. Millions of people use the internet daily, taking full advantage of the available services at both personal and professional levels. The internet connectivity among computers on which the World Wide Web relies, however renders its nodes on easy target for malicious users who attempt to exhaust their resources or damage the data or create a havoc in the network.

Computer Viruses, especially in recent years, have increased dramatically in number. One of the most high-profile threats to information integrity is the Computer Virus. Surprisingly, PC viruses have been around for two-thirds of the IBM PC's lifetime, appearing in 1986. With global computing on the rise, computer viruses have had more visibility in the past few years. In fact, the entertainment industry has helped by illustrating the effects of viruses in movies such as ”Independence Day”, ”The Net”, and ”Sneakers”. Along with computer viruses, computer worms are also increasing day by day. So, there is a need to immunise the internet by creating awareness in the people about these in detail. In this paper I have explained the basic concepts of viruses and worms and how they spread.

The basic organisation of the paper is as follows. In section 2, give some preliminaries: the definitions of computer virus, worms, trojan horses, as well as some other malicious programs and also basic characteristics of a virus. In section 3, detailed description: describe Malicious Code Environments where virus can propagate, Virus/Worm types overview where different types have been explained, and Categories of worm where the different forms of worm is explained in broad sense. In section 4, File Infection Techniques which describe the various methods of infection mechanisms of a virus. In section 5, Steps in Worm Propagation describe the basic steps that a normal worm will follow for propagation. In section 6 Case studies: two case studies of Slammer worm and blaster worm are discussed.

II. Preliminaries

A. Virus:

A self-replicating program. Some definitions also add the constraint saying that it has to attach itself to a host program to be able to replicate. Often Viruses require a host, and their goal is to infect other files so that the virus can live longer. Some viruses perform destructive actions although this is not necessarily the case.Many viruses attempt to hide from being discovered.

A virus might rapidly infect every file on individual computer or slowly infect the documents on the computer, but it does not intentionally try to spread itself from that computer (infected computer) to other. In most cases, that's where humans come in. We send e-mail document attachments, trade programs on diskettes, or copy files to file servers. When the next unsuspecting user receives the infected file or disk, they spread the virus to their computers, and so on.

B. Worms:

Worms are insiduos because they rely less (or not at all) upon human behaviour in order to spread themselves from one computer to others. The *computer worm* is a program that is designed to copy itself from one computer to another, leveraging some network medium: e-mail, TCP/IP, etc. The worm is more interested in infecting as many machines as possible on the network, and less interested in spreading many copies of itself on a single computer (like a computer virus). The prototypical worm infects (or causes its code to run on) target system only once; after the initial infection, the worm attempts to spread to other machines on the network.

Some researchers define worms as a sub-type of Viruses. In early years the worms are considered as the problem of Mainframes only. But this has changed after the Internet become wide spread; worms quickly accustomed to windows and started to send themselves through network functions.

Some categories that come under worms are
•Mailers and Mass-Mailer worms
•Octopus
•Rabbits

C. Trojan Horses:

A Trojan Horse is a one which pretend to be useful programs but do some unwanted action. Most trojans activate when they are run and sometimes destroy the structure of the current drive (FATs, directories, etc.) obliterating themselves in the process. These does not require a host and does not replicate.

A special type is the backdoor trojan, which does not do anything overtly destructive, but sets your computer open for remote control and unauthorised access.

D. Others:

There are other types of malicious programs apart from Viruses, Worms and Trojan Horses. Some of them are described below.

1) Logic Bombs:: A logic bomb is a programmed malfunction of a legitimate application. These are intentionally inserted in otherwise good code. They remains hidden with only their effects are being visible. These are not replicated. Bugs do everything except make more bugs.

2) Germs:: These are first-generation viruses in a form that the virus cannot generate to its usual infection process. When the virus is compiled for the first time, it exists in a special form and normally does not have a host program attached to it. Germs will not have the usual marks that the most viruses use in second-generation form to flag infected files to avoid reinfecting an already infected object.

3) Exploits:: Exploit is specific to single vulnerability or set of vulnerabilities. Its goal is to run a program (possibly remote, networked) system automatically or provide some other form of more highly previliged access to the target system.

E. Characteristics:

The following are some of the characteristics of Viruses:

1) Size - The sizes of the program code required for computer viruses are very small.
2) Versatility - Computer viruses have appeared with the ability to generically attack a wide variety of applications.
3) Propagation - Once a computer virus has infected a program, while this program is running, the virus is able to spread to other programs and files accessible to the computer system.
4) Effectiveness - Many of the computer viruses have far-reaching and catastrophic effects on their victims, including total loss of data, programs, and even the operating systems.
5) Functionality - A wide variety of functions has been demonstrated in virus programs. Some virus programs merely spread themselves to applications without attacking data files, program functions, or operating system activities. Other viruses are programmed to damage or delete files, and even to destroy systems.
6) Persistence - In many cases, especially networked operations, eradication of viruses has been complicated by the ability of virus program to repeatedly spread and reoccur through the networked system from a single copy.

III. Detailed Description

A. Malicious Code Environments

It is important to know about the particular execution environments to understand about Computer Viruses. A successful penetration of the system by a viral code occurs only if the various dependencies of malicious code match a potential environment. The following are some of the various malicious code environments

1) Computer Architecture Dependency
2) CPU Dependency
3) Operating System Dependency and Operating System version Dependency
4) File System Dependency
5) File Form Dependency
6) Interpreted Environment Dependency
7) Vulnerability Dependency
8) Date and Time Dependency
9) Just-In-Time Dependency
10) Achieve Format Dependency
11) File Format Extension Dependency
12) Network Protocol Dependency
13) Source Code Dependency
14) Self Contained Environment Dependency

B. Virus/Worm types overview

These are the main categories of Viruses and worms:

1) Binary File Virus and Worm – File virus infect executables (program files). They are able to infect over networks. Normally these are written in machine code. File worms, are also written in machine code, instead of infecting other files, worms focus on spreading to other machines.
2) Binary Stream Worms – Stream worms are a group of network spreading worms that never manifest as files. Instead, they will travel from computer to computer as just pieces of code that exist only in memory.
3) Script File Virus and Worm – A script virus is technically a file virus, but script viruses are written as human readable text. Since computers cannot understand text instructions directly, the text first has to be translated from text to machine code. This process is called ”Interpretation”, and is performed by separate programs on computer.
4) Macro Virus – Macro Viruses infect data files, or files that are normally perceived as data files, like documents and spreadsheets. Just about anything that we can do with ordinary programs on a computer we can do with macro instructions. Macro viruses are more common now-a-days. These can infect over the network.
5) Boot Virus – The first known successful computer viruses were boot sector viruses. Today these are rarely used. These infect boot sectors of hard drives and floppy

disks and are not dependent on the actual operating system installed. These are not able to infect over networks. These take the boot process of personal computers. Because most computers don't contain Operating System in their Read Only Memory (ROM), they need to load the system from somewhere else, such as from a disk or from the network (via a network adapter).

6) Multipartite Viruses – Multipartite Virus infect both executable files and boot sectors, or executable and data files. These are not able to infect over the networks.

C. *Categories of Worm*

Worms are broadly categorised into three types. They are:

1) E-mail (and other application) worms – These worms when executed on a local system, take advantage of the user's e-mail capabilities to send themselves to others. The first e-mail worm was found in 1987, with the *Christmas tree* trojan horse. At the early stages these were using local mail programs and/or mail Api's on a compromised machine to send out copies of themselves to one or more addresses. Later e-mail worms contained their own SMTP engines so that they were not (as) dependent on the mail capabilities of the compromised machine. Soon after they started using spoof mail headers.
2) Windows file sharing worms – These take the advantage of the Microsoft Windows peer-to-peer service that is enabled whenever Windows determines networking hardware is present in a system. It uses Server Message Block (SMB) protocol and sometimes the Common Internet File System (CIFS), which was originally designed for trusted workgroups. File sharing worms are rarely seen in isolation as they are usually created along with other attacks also as well configure firewall can stop the file sharing outside of the organisation. These are growing recently over the past two years.
3) Traditional worms – These do not require user intervention. These often uses direct connections over TCP-IP based protocols to exploit vulnerabilities in operating systems and applications. Most of the traditional worms have exploited Unix-based operating systems such as Linux. Recently only these are affecting Microsoft operating systems. These exploit the vulnerabilities to propagate, and the time between the time of announcement of a vulnerability and its exploitation by a worm has been shrinking.

IV. File Infection Techniques of viruses

The following are the common strategies that virus writes used over the years to invade into the new host systems:

1) Overwriting Viruses – These locate another file on the disk and overwrite with their own copy. This is the easiest approach and these can do a great damage when they overwrite all the files in the system. These cannot be disinfected from a system. Infected files must be deleted and should be restored from backups. These don't change the size of the host.
2) Random Overwriting Viruses – This is another rare variation of the overwriting method does not change the code at the top of the file but it chooses a random location in the host program and overwrites that location. In this case it may be possible that the code is not even get control during the execution. In both cases , the host program is lost during the virus attack, and often crashes before the virus code executes.
3) Appending Viruses – In this technique the virus code is appended at the end of the program and the first instruction of the code is changed to a jump or call instruction which will be pointing to the starting address of the viral code.
4) Prepending Viruses – A common virus infection technique uses the principle of inserting virus code at the front of host programs. Such viruses are called Prepending Viruses. This is a simple infection technique and is often successful. Virus writers wrote much of this kind on various operating systems, causing major outbreaks in many.
5) Classical Parasitic Virus – This is a variation of prepender technique. These overwrite the top portion of the program with virus code and the top portion is being copied at the end of the program.
6) Cavity Viruses – These typically don't increase the size of the program they infect. Instead they will overwrite a part of the code that can be used to store the virus code safely. Normally these overwrite areas of files that contain zeros in binary files. These are often slow spreaders in DOS systems.
7) Compressing Viruses – This is a special technique where the content of host program is compressed. Compressor Viruses are sometimes beneficial because such viruses might compress the infected program to a much shorter size saving disk space.
8) Amoeba Infection Technique – This is a rarely seen infection technique where the head part of the viral code is stored at the starting of the host program and the tail part is stored after the end of the host program.

V. Steps in Worm Propagation

Each Worm has a few essential components, such as target locator, infection propagation modules, and a couple of nonessential modules, such as remote control, update interface, life cycle-manager, and payloads.

1) *Target Locator:-* For a worm to propagate first it must discover the existence of a machine. There are many techniques by which a worm can discover new machinesto exploit. They are
   a) Scanning: This entails probing a set of addresses to identify the vulnerable hosts. Two simple forms of scanning are *Sequential scanning* (working through an address block using ordered set of addresses)

and *Random scanning* (trying addresses out of a block in pseudo-random fashion).

b) Pre-generated Target Lists: An attacker could obtain a target list in advance, creating a ”hit-list” of a probable victims with good network connections. This list is being created well before the release of worm. There are some scanning techniques that just see for particular criteria such as the operating system that the machine is running, what are the servers running, what is the version of operating systems etc. Stealthy scans, Distributed scanning, DNS searches, Just listen and also there are some public surveys that list such as Netcraft Survey.

c) Externally Generated Target Lists: An externally generated list is one which is maintained by a separate server, such as a matchmaking service;s metaserver. This can also be used to speed the worm propagation. This worm has not yet in the wild.

d) Internal Target Lists: Many applications contain information about the other hosts providing vulnerable services. Such target lists can be used to create ’topological’ worms, where the worm searches for the local information to fine new victims by trying to discover the local communication topology.

e) Passive: These does not seek out victim machines. Instead, they either wait for potential victims to contact the worm or rely on user behaviour to discover new targets. Although potentially slow these worms produce no anomalous traffic patterns during the target discovery, which potentially makes them high stealthy.

2) *Infection Propagator:-* A very important strategy of the worm uses to transfer itself to a new node and get control on remote machine. Most worms will assume that one has a copy of certain window machine and send a worm with such compatible system.

3) *Remote Control and Update Interface:-* Another important component of a worm is remote control using a communication module. Without such a module, the worm’s author cannot control the worm network by sending control messages to the worm copies. such remote control can allow the attacker to use the worm as a DDoS (distributed denial of service) tool on the zombie network against several unknow targets. The attacker is interested in changing the behaviour of the worm and even sending new infection strategies to as many compromised nodes as possible.

4) *Life-Cycle Manager:-* Some writers prefer to run a version of a computer worm for a preset period of time. On the other hand, many worms have bugs in their life-cycle manager component and continue to run without ever stopping.

5) *Payload:-* This is optional but common component of a worm. An increasingly popular payload is a DDoS attack against a particular website. These can utilise the compromised systems as a ”super computer”. Recently it is becoming popular to install an SMTP (Simple Mail Transfer Protocol) spam relay as the payload of a worm.

6) *Self-Tracking:-* Many virus authors are interested in seeing how many machines the virus can infect and also they want others to track the path of virus infections.

VI. Case Studies

A. Slammer Worm

Slammer worm sometimes called as Sapphire was the fastest computer worm in history till now. It began his journey on January 25, 2003. It began spreading through the Internet infected more than 90 percent of vulnerable hosts within 10 minutes, causing a significant disruption to financial, transportation, and government institutions and precluding any human-based response.

1) Vulnerability: Microsoft’s database server SQL Server or Microsoft SQL Server Desktop Engine(MSDE) 2000 exhibits two buffer overrun vulnerabilities that can be exploited by a remote attacker without ever having to authenticate to the server. These are being attacked based on the Stack overflow and heap overflow techniques.

2) Target Selection: It used random scanning for selecting IP addresses, there by selecting vulnerable systems. Random scanning worms intially spread exponentially, later infection slows as the worms continually retry infected or immune addresses. Slammer is bandwidth-limited, in contrast to Code Red which is latency-limited.

3) Infection Propagator: It carries only 376 bytes of code where there is a simple, fast scanner. Along with the headers of the protocol it will of total size of 404 bytes. It used UDP protocol for propagation so it can transmit the entire packet in a single transfer. It uses 1434 port to transfer packets. It doesnot write itself into the system. It exists only as network packets and in running processes on the infected computers.

4) Payload: This does not contain any additional malicious content in the form of backdoors, etc. The speed at which it attempts to re-infect systems to create a denial-surface of attack.

5) Network Propagation: When the SQL server receives a malicious request, the overrun in the server’s buffer allows the worm code to be executed. After the worm has entered into the vulnerable system,, first it gets the addresses to certain functions then start an infinite loop to scan for the other vulnerable hosts on the internet. This performs pseudo-random number generation formula using the GetTickCount() value to generate an IP address that is used as target thereby, spreading furher into the network and infecting the vulnerable machines. These don’t check for the multiple instances of the worm affected the system.

This could have been a great damage if it would have carried any malicious code with it. There are few wrong things that this wormauthor did such as in the pseudo random number generation algorithm the author used the following equation $x^1 = (x * 214013 + 2531011) mod 2^{32}$ here the author substituted a different value for 2531011 increment value:hex 0xFFD9613C. This value is equivalent to -2531012 when interpreted as a twos-complement decimal.

6) Prevention: This can be prevented using a firewall which blocks 1434 port as the worm infects through this port only.

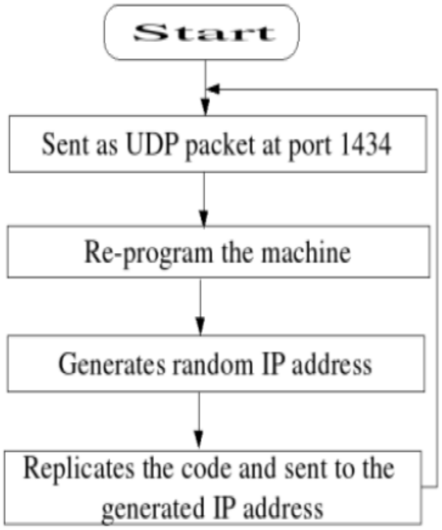
Start
Sent as UDP packet at port 1434
Re-program the machine
Generates random IP address
Replicates the code and sent to the generated IP address

Fig. 1. Overview of Slammer

B. Blaster Worm

It is a multi stage worm first observed on August 11, 2003. It affected between 200,000 and 500,000 computers.

1) Vulnerability: It exploited a remote procedure call (RPC) vulnerability of Microsoft Windows 2000 and Windows XP operating systems which were made public in July 2003.

2) Intialization: The worm when launched, opens a mutex called ”BILLY” that is used to prevent multiple infections of the same machine and sets a registry key which ensures that it is started every time the system reboots.

3) Target Selection: In the intialization phase it decides whether it will exploit code for Microsoft XP with 80% probability or the one for Windows 2000. It first scans with 60%, an IPv4 address of the form X.Y.Z.0 with X, Y, Z are chosen at random. With 40% probability, and address of the form X.Y.Z1.0 derived from the infected computer’s local address X.Y.Z.U is chosen. Z1 is set to Z unless Z1 is greater than 20, in which case a random values less than 20 is subtracted from Z to get Z1. The destination IP is incremented after each scan.

4) Infection Propagator: If TCP connection to a destination 135 port is opened, the exploit code is sent to victim. If the machine was vulnerable it can start listening on 4444/TCP and allows remote command execution. unpatched windows automatically reboots XP. Next it intiates a TCP connection to 4444 port, if successful, using TFTP(Trivial File Transfer Protocol - which is a smaller version of FTP) the mblast.exe file is transfered. After that if TFTP requests are not blocked, on UDP port 69 the worm code is being downloaded. Infected host stops TFTP daemon after transmission or after 20 secs of inactivity. If successful it sends a command mblast.exe on the already open TCP connection to port 4444 of the victim.

5) Payload: The payload of the worm for RPC step is as follows– 72 bytes for RPC, 1460 bytes for ”request” and a 244 bytes of TCP packet, Along with these there is 40-48 bytes for TCP/IP which makes the worm to 1976 to 2016 bytes. The worm code is of 6176 bytes. along with the overhead of headers it will come to 6592 bytes on the IP layer.

6) Prevention: This can be prevented by using the firewall that blocks traffic to incoming to port 135/TCP or 4444 port or TFTP port and by applying the operating system patch against the RPC vulnerability.

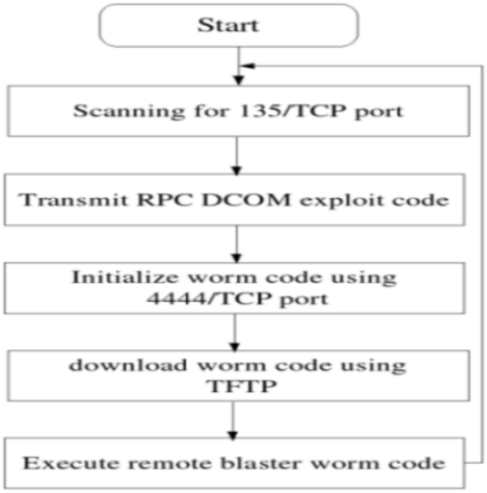
Start
Scanning for 135/TCP port
Transmit RPC DCOM exploit code
Initialize worm code using 4444/TCP port
download worm code using TFTP
Execute remote blaster worm code

Fig. 2. Overview of Blaster

VII. Conclusion

I have gone through the basic definitions of Viruses and Worms, then discussed in about the different malicious code environments. After that I have discussed about the different types of viruses and worms, then discussed in detail about the various ways of virus and worm propagation techniques. I have also looked into two case studies of slammer and blaster worms.

The ability of attackers to rapidly gain control of vast numbers of internet hosts poses an immense risk to overall security of the internet. Now-a-days the virus writers are more concentrating on writing worms as they have got great capability to spread over the network in few minutes. There are various upcoming techniques in worm propagation such as polymorphic worms which are really a big threat to the internet community. Worms can be written such that they can be affected only to a particular region or country. There are worms which will keep quiet for a specific amount of time and attack at random times.

These worms can also be used to create Distributed Denial of Service (DDoS) which is a real threat to the websites and the network traffic.

VIII. Acknowledgement

I would like to thank my guide Prof. Bernard Menezes, for his continuous support through out this work. I would also thank Peter Szor for his excellent book of ”The Art of Computer Viruses and Defence”. I would also thank one and all who helped me in doing this work either directly or indirectly.

References

[1] Peter Szor, *The Art of Computer Virus and Defence*, Harlow, England: Addison Wesely Professional, 2005.

[2] Norman, *Norman book on Computer Virus*, Norman ASA, 2003.

[3] Dan Xu, Xiang Li, and Xian Fan Wang, *Mechanisms for Spreading of Computer Virus on the Internet: An Overview*, IEEE Computer Society 2004, 601-606.

[4] Darrell M. Kienzie, and Matthew C. Elder, *Recent Worms: A Survey and Trends*, Washington, DC, USA: WORM-2003.

[5] David Moore, Vern Paxson, Stefan Savage, Colleen, Stuart Staniford and Nicholas Weaver, *Inside the Slammer Worm*, IEEE Security and Privacy, 2003.

[6] Thomas Subendorfer, Arno Wagner, Theus Hossmann, and Bernhard Plattner, *Flow-Level Traffic Analysis of the Blaster and Sobig Worm Outbreaks in an Internet Backbone*, Springer-Verlag Berlin Heidelberg 2005.

[7] Nicholas Weaver, Vern Paxson, Stuart Staniford, and Robert Cunnigham, *A Taxonomy of Computer Worms*, Washington, DC, USA: WORM-2003.

[8] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.