An Analytical Survey of Recent Worm Attacks

Vishrut Sharma

Member of ACM, IEEE

ABSTRACT

In this paper, I will present a broad overview of recent worm attacks starting from the year 2000 through 2011. Some of the most 'notorious' worms were discovered during this period including code red, slammer, and conficker.

Though this paper does not contain any original research, it is meant to provide malware researchers with well-documented information of some worms that caused havoc during the said period. After sifting through thousands of entries on virus information repositories, I present, in this paper, only those that I considered novel in their approach, primarily from a technical perspective.

As a summary to the paper, I have presented a broad overview of trends that I extracted from this raw data and also focussed on the ever increasing destructive potential of worms.

KEYWORDS

Worms, virus, recent attacks, self-propagating, malwares

1. Introduction

The outbreak of Morris worm in 1988 marked a new era of self-replicating malware. In May, 2000, the ILOVEYOU worm also known as VBS/Loveletter appeared and soon infected millions of computers worldwide. The worm came through e-mail with the simple subject of "ILOVEYOU" and an attachment "LOVE-LETTER-FOR-YOU.". The file extension was hidden by default, leading unsuspecting users to think it was a normal text file. Upon opening the attachment, the worm sent a copy of itself to everyone in the Windows Address Book and with the user's sender address. It also made a number of malicious changes to the user's system. This marked the beginning of an era of fast, self-propagating, and self-replicating worms.

In March 2001, cnet declared that 2001 would be "The Year of the Worm" [2]. They predicted that fast-moving, self-replicating code would become the weapon of choice for those wanting to inflict widespread damage on the Internet. As it turns out, 2001 saw a renaissance in worm creation. This culminated in the release of Nimda, an incredibly sophisticated worm that made headlines worldwide [1].

The goal of this study was to enhance knowledge about recent trends in worm development and attempt to predict future worm developments. In this paper, I present my findings about recent worms, their technical specifications, etc.

In this paper, I've discussed the past and present of worms and related malicious code with special focus on worms like Code Red, Nimda, Slammer, and Conficker.

The following section presents the definition and categories of a computer worm.

2. Definition and Type of Worms

In general, worm is a type of malware that can self-propagate and/or self-replicate over a network of computers. Self-propagation means it can propagate through a network without human intervention. Self-propagation is what makes a worm differ from a virus. A virus, on the other hand, usually infects non-mobile files i.e. those files that when carried from one computer to another computer through a media, may propagate the virus attack thus making virus propagation very slow as compared to worm.

In recent years, many different categories of worms, based on their programming and payloads, have been found. Broadly, worms can be categorised as follows:-

- **Email Worms**
- **Instant Messaging Worms**
- **Internet Worms**
- **IRC Worms**
- **File-sharing Network Worms**
- **PDF Worms**

The following sections explain each of these types in detail. However, this is not a strict classification scheme. Some worms, like Nimda, fall in two or more categories.

2.1 Email Worms

Such worms spread via infected email messages. Any form of attachment or link in an email may contain a link to an infected website. In the first case activation starts when the user clicks on the attachment while in the second case the activation starts when clicking the link in the email.

Known methods to spread are:

- MS Outlook services

- Direct connection to SMTP servers using their own SMTP API

- Windows MAPI functions

Manuscript received November 5, 2011
Manuscript revised November 20, 2011

Table 1: some infamous worms, date of their detection, category, and method of infection. [6], [7], [8]

Worm Name	***Date of Detection***	***Category***	***Method of Infection***
VBS/Loveletter@MM	May 4, 2000	Email/IRC	Running the email attachment received either accidentally or intentionally will install it to the local system, and also to all available drives.
W32/CodeRed.f.worm	July 12, 2001	Internet	Exploited Microsoft IIS 5.0 IDQ path overflow vulnerability
W32/Nimda.gen@MM	September 18, 2001	Email/Internet	Through email and exploiting various vulnerabilities present in Microsoft IIS Servers
W32/Klez.gen@MM	November 9, 2001	Email	Exploits vulnerability in MSOutlook and Outlook Express & tries executing itself when you open or preview the message.
W32/SQLSlammer.worm	January 25, 2003	Internet	Exploited buffer overflow vulnerability in Microsoft SQL Server.
W32/Sobig.f@MM	August 19, 2003	Email/File-sharing Network	Propagates via email (contains its own SMTP engine) and attempts to spread via accessible network shares.
W32/Bagle.gen	January 18, 2004	Email	Spreads as email attachment.
W32.MyDoom@mm	January 26, 2004	Email	A mass-mailing worm that arrives as an attachment with the file extension .bat, .cmd, .exe, .pif, .scr, or .zip.
W32/Netsky.j@MM	March 8, 2004	Email	A mass-mailing worm that uses its own SMTP engine to send itself to the email addresses it finds when scanning hard drives and mapped drives.
W32/Sasser.worm.a	April 30, 2004	Internet	Attempts to exploit the vulnerability described in Microsoft Security Bulletin MS04-011. It spreads by scanning the randomly selected IP addresses for vulnerable systems.
Perl/Santy.worm	December 21, 2004	Internet	Attempts to spread to Web servers running versions of the phpBB 2.x bulletin board software prior to 2.0.11, which are vulnerable to the PHPBB Viewtopic.PHP PHP Script Injection Vulnerability.
W32/IRCbot.worm!MS05-039	August 16, 2005	Internet/IRC	Opens a back door and exploits the Microsoft Windows Plug and Play Buffer Overflow Vulnerability (described in Microsoft Security Bulletin MS05-039) on TCP port 445.
Win32.Nyxem.e	January 17, 2006	Email/File-Sharing Network	A mass-mailing worm that attempts to spread through network shares and lower security settings. On the third day of every month it attempts to rewrite files with certain extensions with custom text.
W32.Rontokbro.AN@mm	April 22, 2006	Email	A mass-mailing worm.
W32/Koobface.worm	August 3, 2008	Internet	Spreads primarily through social networking sites as links to videos. When a user visits the website that is hosting the video, they are prompted to download a video codec or other necessary update, which is actually a copy of the worm.
W32/Conficker.worm	January 13, 2009	Internet	Exploits the MS08-067 Microsoft Windows Server Service vulnerability in order to propagate.
W32.Stuxnet	July 13, 2010	Internet	Targets systems running WinCC SCADA software. It spreads utilizing CVE-2010-2568 which allows arbitrary code execution via a crafted .lnk file. This has been noted to spread via removable USB drives.
W32.Morto	August 28, 2011	Internet	Attempts to spread using the Remote Desktop Protocol.
W32.Duqu	October 18, 2011	Not known	Not known

Attackers have been using email to propagate malicious code from as early as 1987, the year when Christmas Tree Trojan horse appeared. Malwares that use email to propagate are generally referred to as mailers. Unfortunately, these mailers are used even today. In fact, in past few years these mailers have gained more popularity since email is the most efficient way for propagation of malicious codes in order to compromise a sizeable amount of hosts on the Internet.

Some worms that used this method for propagation in the last 10 years are VBS/Loveletter@MM, W32/Pandem.worm, W32/Nimda.gen@MM, W32/Mydoom.f@MM, etc.

2.2 Instant Messaging Worms

Such worms spread via instant messaging applications like by sending links to infected websites to everyone on the local contact list. The only difference between these and email worm is the way chosen to send the links. Some worms that used this method for propagation are W32/YahLover.worm, W32/Sdbot.worm!im, etc.

2.3 Internet Worms

These worms scan all available network resources using local operating system services and/or scan the Internet for vulnerable machines. Attempt will be made to connect to these machines and gain full access to them.

Another way is that the worms scan the Internet for machines still open for exploitation i.e. machines which are not patched. Data packets or requests will be sent which install the worm or a worm downloader. If succeeded the worm will execute and compromises the system.

This is one of the most efficient ways of propagation of worms just after email. Some worms that made use of this method are W32/SQLSlammer.worm, W32/Pandem.worm, W32/Conficker.worm, W32/Nimda.gen@MM, W32/CodeRed.f.worm, etc.

2.4 IRC Worms

Same as the Instant Messaging worms but it infects and uses IRC (Internet Relay Chat) in order to propagate. Some worms under this category are IRC/Stages.worm, W32/Pandem.worm, VBS/Loveletter@MM, etc.

2.5 File-sharing Network Worms

These worms copy themselves into a shared folder, most likely located on the local machine. Now the worm is ready to download via the P2P network thus making it spread.

Most worms today exploit windows file-sharing and for this they make use of the Common Internet File System (CIFS) protocol to gather information and compromise the target host. CIFS is an extension to Server Message Block (SMB) protocol. Since this protocol was designed to allow small workgroups to share files in a trusted environment more emphasis was given to resource-sharing than security and this loophole is reflected in the worm attacks.

Some worms that made use of this are W32/Autorun.worm.g, W32/Mydoom.f@MM, W32/Pandem.worm,W32/Sobig.e@MM, etc.

2.6 PDF Worms

In 2001, a new exploit came into existence that used Adobe Acrobat PDF format as a platform. However, it only worked under the full 'developer' version of Acrobat. The common Acrobat Reader program was not affected by this worm. The worm operated as a VB script embedded within a PDF file [4]. Since 2001, the number of such exploits has increased but fortunately no worm has appeared yet. It seems that this category of worms is yet to unleash.

As we can see in the above examples of worms, some worms appear in different categories such as VBS/Loveletter@MM is both an IRC worm and an Email worm. This happens because of the payload that was used by the worm author. Similarly, W32/Pandem.worm appears in 3 different categories viz. Email worm, Internet worm and File-sharing Network worm. Such worms have different propagation method and hence fall in different categories.

3. Worm Characteristics

A worm may be characterised based on its propagation method and payload. General consensus is that a worm is a form of malware that can self-propagate through a network without any human intervention. But, some worms were discovered lately that needed some user action in order to propagate, thus modifying this definition to “A worm is a form of malware that can propagate through a network with or without any human intervention”. Various characteristics of a worm are known and agreed upon by researchers worldwide, some of them are given below.

- Malicious Code: We all know that worms are malicious in nature. Some people say that there have been “good worms” too that breaks into systems in order to repair them. The first computer worm created at XEROX PARC was actually a self-propagating maintenance program [5], although, the connotation of worm now is one of an uninvited program that executes malicious code to perform an unauthorized, harmful or undesirable act.

- Self-propagation: Worms can actively propagate over a network and this makes them unique. The earlier malwares often relied on humans to carry storage media such as pen-drive or floppy disk from one system to another, a worm, on the other hand, attacks another computer directly over some network interface. File-infecting viruses that infect local files that happen to be remotely mounted from another machine are generally not considered worms because they are not actively aware of the network [1].
- User Action: Traditionally, virus has required user intervention to spread from one machine to another (e.g., copying via pen-drive). Whereas, worm is said to be a malware that self-propagates i.e. it requires no human intervention.

However, some people say there are two categories of worm: one that requires user intervention to propagate (e.g., opening an email message or attachment) and one that does not require user intervention [2]. The degree of user intervention varies: some worms that do not require the user to actively execute or open malicious code files require the user to take other seemingly unrelated actions, such as rebooting or running a mail program.

4. Trends in Worm Programming

While studying about worms I found some qualitative and quantitative trends in the programming of worms. These trends are given in sections below.

4.1 Quantitative Trends

To be able to determine approximate quantitative trends, I made use of Trend Micro’s Threat Encyclopaedia [9] to study about worms that can be classified according to the types I mentioned. I searched for some keywords such as ‘@mm’ (mass-mailer) or ‘@m’ (mailer) for worms that propagate through email, ‘net’ for Internet worm, ‘IRC’ for IRC worm, ‘IM’ for IM worm, apart from studying all the other worms. Since no access to the database files was available I made use of ordinary web search for this. Some trends that I inferred show that the most common propagation method is still through email apart from propagation through Internet. IM worm made its presence not until 2005 when WORM_BROPIA came into existence alongwith its several variants. Though, the first IRC worm was released in 1997 this method of propagation doesn’t seem to be that popular when compared to propagation through email. File-Sharing Network worms or P2P worms have been there from the year 2000 and has become one of the most favoured propagation methods since then, the latest worm utilising this method of propagation being Duqu that appeared in September 2011. In September 2011, a new threat emerged when Kaspersky Labs detected malicious QR codes [10]. Soon this could be another propagation vector for worms.

4.2 Qualitative Trends

Based on my study of hundreds of worms, I observed some qualitative trends. These trends are my personal observations and may differ from person to person.

a. Technological Commoditization

Advances in technology soon become more of a commodity for every other worm that appears after the first significant worm. The best example could be that of email propagation of worms. The first malicious code with email propagation capability appeared in the year 1987, then it would have been considered a complex code since it allowed self-propagation based on email which no one had ever thought of. Now, email propagation has become a routine. Anyone can now design an email worm by either making a new worm on top of an existing one or using toolkits. File-Sharing network worms were first seen in the year 2000 but now it has become one of the most favoured propagation techniques. Maybe in the coming few years, worms may start propagating using some other technique such as QR codes.

b. Propagation Vector and Convergence

Although Internet and Email have been the two most efficient ways of propagating a worm, last few years have seen a growth in different propagation methods such as that using Instant Messaging applications or IRC, etc. These days worms are designed with multiple capabilities such as that of a Trojan Horse or a Backdoor hence making them less distinguishable. Worms such as Duqu enabled attackers with different services such as that of a backdoor. So, it seems that the malware technology is now converging from different types of malwares to one having different capabilities.

c. Propagation Speed

The speed of propagation has increased since the Morris worm. While Code Red infected around 2000 hosts per minute during its peak infection rate the Witty worm that came in 2004 infected approximately 12000 hosts in a minute. The Conficker worm that made its presence in late 2008 is still in the wild and has infected over 7 million computers worldwide making it the largest infection till date. This worm may have finally reached the ultimate propagation speed, even much greater than that of SQL/Slammer worm. Unfortunately enough, it seems anti-virus softwares have lagged behind malwares when it comes to speed.

d. Platforms

The worm authors seem to be much interested in writing worms that run on Windows system based on the fact that Windows OS is a widely used operating system whether for personal computers or servers. Lately, there have been worms that infected Windows servers exploiting vulnerabilities such as those in IIS, etc. On the other hand, operating systems such as Macintosh, or different distributions of Linux such as Ubuntu are less vulnerable to worm attacks since they are used less as compared to Windows.

Summary

Worms have been an interesting piece of program to the researchers worldwide. Since the first worm in 1988 there have been huge technological advancements in worm writing. The propagation methods, speed of propagation, payloads, etc. have all increased. There’s no doubt in saying that the next worm attack will be much more damaging and technologically advanced. When the cloud technology came into existence some people predicted that this may be the most secure technology ever but worm authors proved them wrong. The Conficker worm was the first worm that penetrated the cloud. This tells us about the ultimate intelligence that worm authors possess. In order to counter-attack these malwares there is a growing need to develop more enhanced and efficient anti-virus softwares. Until this day, a worm can propagate much faster than the average detection rate of the best anti-virus and this is a major hindrance. The malware authors have now shifted to other technologies too such as mobile. Although, there can’t be a system which has no vulnerability but a software can be made that can launch a counter-attack in case a vulnerability is being exploited.

References

[1] Darrel M. Kienzle, Matthew C. Elder. “Recent Worms: A Survey and Trends”. WORM '03 Proceedings of the 2003 ACM workshop on Rapid malcode.

[2] Lemos, R. “Year of the Worm: Fast-spreading code is weapon of choice for Net vandals”. CNET News.com, http://news.com.com/2009-1001-254061.html, March 2001.

[3] http://virusall.com/computer%20worms/worms.php

[4] F-Secure. F-Secure Computer Virus Information Centre. http://www.f-secure.com/v-descs/pdf.shtml

[5] J. Shoch and J. Hupp. “The Worm Programs: Early Experience with a Distributed Computation”. Communications of the ACM, Vol. 25 No. 3 (March 1982), 172-180.

[6] ThreatExpert. http://www.threatexpert.com

[7] McAfee Threat Library. http://www.mcafee.com/threat-intelligence/malware/latest.aspx

[8] Symantec Security Response. http://www.symantec.com/security_response/

[9] Trend Micro Threat Encyclopaedia. http://about-threats.trendmicro.com/

[10] Kaspersky Lab. http://usa.kaspersky.com/about-us/press-center/press-blog/malicious-qr-codes-attack-methods-techniques-infographic

I completed the degree of Bachelor of Technology in Computer Science and Engineering in the year 2010 and then pursued PG Diploma in Information Systems & Cyber Security and completed it in February, 2011. I’ve two research publications. My first paper with title “A Comparative Analysis of Exact String-Matching Algorithms for Virus Signature Detection”, co-authored by Mr. Amit Kumar and Dr. Shishir Kumar, got published in the proceedings of First International Conference on Emerging Trends in Soft Computing and ICT (SCICT - 2011). Second paper with title “A Theoretical Implementation of Blended Program Analysis for Virus Signature Extraction” got published recently in the proceedings of 45th IEEE Carnahan Conference on Security Technology (ICCST - 2011). My area of interest includes hacking, malware analysis and research, reverse engineering, and programming in Microsoft technologies (VB, VB.Net, C#.Net)