part 1

//////////*1000+ HACKING TRICKS & TUTORIALS - ebook By Mukesh Bhardwaj Blogger - Paid Version - only @ TekGyd | itechhacks | Mukeshtricks4u*///////

Like any other field in computer science, viruses have evolved -a great deal indeed- over the years. In the series of press releases which start today, we will look at the origins and evolution of malicious code since it first appeared up to the present.

Going back to the origin of viruses, it was in 1949 that Mathematician John Von Neumann described self-replicating programs which could resemble computer viruses as they are known today. However, it was not until the 60s that we find the predecessor of current viruses. In that decade, a group of programmers developed a game called Core Wars, which could reproduce every time it was run, and even saturate the memory of other players' computers. The creators of this peculiar game also created the first antivirus, an application named Reeper, which could destroy copies created by Core Wars.

However, it was only in 1983 that one of these programmers announced the existence of Core Wars, which was described the following year in a prestigious scientific magazine: this was actually the starting point of what we call computer viruses today.

At that time, a still young MS-DOS was starting to become the preeminent operating system worldwide. This was a system with great prospects, but still many deficiencies as well, which arose from software developments and the lack of many hardware elements known today. Even like this, this new operating system became the target of a virus in 1986: Brain, a malicious code created in Pakistan which infected boot sectors of disks so that their contents could not be accessed. That year also saw the birth of the first Trojan: an application called PC-Write.

Shortly after, virus writers realized that infecting files could be even more harmful to systems. In 1987, a virus called Suriv-02 appeared, which infected COM files and opened the door to the infamous viruses Jerusalem or Viernes 13. However, the worst was still to come: 1988 set the date when the "Morris worm" appeared, infecting 6,000 computers.

From that date up to 1995 the types of malicious codes that are known today started being developed: the first macro viruses appeared, polymorphic viruses … Some of these even triggered epidemics, such as MichaelAngelo. However, there was an event that changed the virus scenario worldwide: the massive use of the Internet and e-mail. Little by little, viruses started adapting to this new situation until the appearance, in 1999, of Melissa, the first malicious code to cause a worldwide epidemic, opening a new era for computer viruses.

part 2

This second installment of 'The evolution of viruses' will look at how malicious code used to spread before use of the Internet and e-mail became as commonplace as it is today, and the main objectives of the creators of those earlier viruses.
Until the worldwide web and e-mail were adopted as a standard means of communication the world over, the main mediums through which viruses spread were floppy disks, removable drives, CDs, etc., containing files that were already infected or with the virus code in an executable boot sector.

When a virus entered a system it could go memory resident, infecting other files as they were opened, or it

could start to reproduce immediately, also infecting other files on the system. The virus code could also be triggered by a certain event, for example when the system clock reached a certain date or time. In this case, the virus creator would calculate the time necessary for the virus to spread and then set a date –often with some particular significance- for the virus to activate. In this way, the virus would have an incubation period during which it didn't visibly affect computers, but just spread from one system to another waiting for 'D-day' to launch its payload. This incubation period would be vital to the virus successfully infecting as many computers as possible.

One classic example of a destructive virus that lay low before releasing its payload was CIH, also known as Chernobyl. The most damaging version of this malicious code activated on April 26, when it would try to overwrite the flash-BIOS, the memory which includes the code needed to control PC devices. This virus, which first appeared in June 1998, had a serious impact for over two years and still continues to infect computers today.

Because of the way in which they propagate, these viruses spread very slowly, especially in comparison to the speed of today's malicious code. Towards the end of the Eighties, for example, the Friday 13th (or Jerusalem) virus needed a long time to actually spread and continued to infect computers for some years. In contrast, experts reckon that in January 2003, SQLSlammer took just ten minutes to cause global communication problems across the Internet.

Notoriety versus stealth

For the most part, in the past, the activation of a malicious code triggered a series of on screen messages or images, or caused sounds to be emitted to catch the user's attention. Such was the case with the Ping Pong virus, which displayed a ball bouncing from one side of the screen to another. This kind of elaborate display was used by the creator of the virus to gain as much notoriety as possible. Nowadays however, the opposite is the norm, with virus authors trying to make malicious code as discreet as possible, infecting users' systems without them noticing that anything is amiss.

pat 3

This third installment of 'The evolution of viruses' will look at how the Internet and e-mail changed the propagation techniques used by computer viruses.

Internet and e-mail revolutionized communications. However, as expected, virus creators didn't take long to realize that along with this new means of communication, an excellent way of spreading their creations far and wide had also dawned. Therefore, they quickly changed their aim from infecting a few computers while drawing as much attention to themselves as possible, to damaging as many computers as possible, as quickly as possible. This change in strategy resulted in the first global virus epidemic, which was caused by the Melissa worm.

With the appearance of Melissa, the economic impact of a virus started to become an issue. As a result, users -above all companies- started to become seriously concerned about the consequences of viruses on the security of their computers. This is how users discovered antivirus programs, which started to be installed widely. However, this also brought about a new challenge for virus writers, how to slip past this protection and how to persuade users to run infected files.

The answer to which of these virus strategies was the most effective came in the form of a new worm: Love Letter, which used a simple but effective ruse that could be considered an early type of social engineering. This strategy involves inserting false messages that trick users into thinking that the message includes anything, except a virus. This worm's bait was simple; it led users to believe that they had received a love

letter.

This technique is still the most widely used. However, it is closely followed by another tactic that has been the center of attention lately: exploiting vulnerabilities in commonly used software. This strategy offers a range of possibilities depending on the security hole exploited. The first malicious code to use this method –and quite successfully- were the BubbleBoy and Kakworm worms. These worms exploited a vulnerability in Internet Explorer by inserting HTML code in the body of the e-mail message, which allowed them to run automatically, without needing the user to do a thing.

Vulnerabilities allow many different types of actions to be carried out. For example, they allow viruses to be dropped on computers directly from the Internet -such as the Blaster worm-. In fact, the effects of the virus depend on the vulnerability that the virus author tries to exploit.

part 4

In the early days of computers, there were relatively few PCs likely to contain “sensitive” information, such as credit card numbers or other financial data, and these were generally limited to large companies that had already incorporated computers into working processes.

In any event, information stored in computers was not likely to be compromised, unless the computer was connected to a network through which the information could be transmitted. Of course, there were exceptions to this and there were cases in which hackers perpetrated frauds using data stored in IT systems. However, this was achieved through typical hacking activities, with no viruses involved.

The advent of the Internet however caused virus creators to change their objectives, and, from that moment on, they tried to infect as many computers as possible in the shortest time. Also, the introduction of Internet services -like e-banking or online shopping- brought in another change. Some virus creators started writing malicious codes not to infect computers, but, to steal confidential data associated to those services. Evidently, to achieve this, they needed viruses that could infect many computers silently.

Their malicious labor was finally rewarded with the appearance, in 1986, of a new breed of malicious code generically called “Trojan Horse”, or simply “Trojan”. This first Trojan was called PC-Write and tried to pass itself off as the shareware version of a text processor. When run, the Trojan displayed a functional text processor on screen. The problem was that, while the user wrote, PC-Write deleted and corrupted files on the computers’ hard disk.

After PC-Write, this type of malicious code evolved very quickly to reach the stage of present-day Trojans. Today, many of the people who design Trojans to steal data cannot be considered virus writers but simply thieves who, instead of using blowtorches or dynamite have turned to viruses to commit their crimes. Ldpinch.W or the Bancos or Tolger families of Trojans are examples of this

part 5

Even though none of them can be left aside, some particular fields of computer science have played a more determinant role than others with regard to the evolution of viruses. One of the most influential fields has been the development of programming languages.

These languages are basically a means of communication with computers in order to tell them what to do. Even though each of them has its own specific development and formulation rules, computers in fact understand only

one language called "machine code".

Programming languages act as an interpreter between the programmer and the computer. Obviously, the more directly you can communicate with the computer, the better it will understand you, and more complex actions you can ask it to perform.

According to this, programming languages can be divided into "low and high level" languages, depending on whether their syntax is more understandable for programmers or for computers. A "high level" language uses expressions that are easily understandable for most programmers, but not so much for computers. Visual Basic and C are good examples of this type of language.

On the contrary, expressions used by "low level" languages are closer to machine code, but are very difficult to understand for someone who has not been involved in the programming process. One of the most powerful, most widely used examples of this type of language is "assembler".

In order to explain the use of programming languages through virus history, it is necessary to refer to hardware evolution. It is not difficult to understand that an old 8-bit processor does not have the power of modern 64-bit processors, and this of course, has had an impact on the programming languages used.

In this and the next installments of this series, we will look at the different programming languages used by virus creators through computer history:

- Virus antecessors: Core Wars

As was already explained in the first chapter of this series, a group of programs called Core Wars, developed by engineers at an important telecommunications company, are considered the antecessors of current-day viruses. Computer science was still in the early stages and programming languages had hardly developed. For this reason, authors of these proto-viruses used a language that was almost equal to machine code to program them.

Curiously enough, it seems that one of the Core Wars programmers was Robert Thomas Morris, whose son programmed -years later- the "Morris worm". This malicious code became extraordinarily famous since it managed to infect 6,000 computers, an impressive figure for 1988.

- The new gurus of the 8-bits and the assembler language.

The names Altair, IMSAI and Apple in USA and Sinclair, Atari and Commodore in Europe, bring memories of times gone by, when a new generation of computer enthusiasts "fought" to establish their place in the programming world. To be the best, programmers needed to have profound knowledge of machine code and assembler, as interpreters of high-level languages used too much run time. BASIC, for example, was a relatively easy to learn language which allowed users to develop programs simply and quickly. It had however, many limitations.

This caused the appearance of two groups of programmers: those who used assembler and those who turned to high-level languages (BASIC and PASCAL, mainly).

Computer aficionados of the time enjoyed themselves more by programming useful software than malware. However, 1981 saw the birth of what can be considered the first 8-bit virus. Its name was "Elk Cloner", and was programmed in machine code. This virus could infect Apple II systems and displayed a message when it infected a computer.

Computer viruses evolve in much the same way as in other areas of IT. Two of the most important factors in understanding how viruses have reached their current level are the development of programming languages and the appearance of increasingly powerful hardware.

In 1981, almost at the same time as Elk Kloner (the first virus for 8-bit processors) made its appearance, a new operating system was growing in popularity. Its full name was Microsoft Disk Operating System, although computer buffs throughout the world would soon refer to it simply as DOS.

DOS viruses

The development of MS DOS systems occurred in parallel to the appearance of new, more powerful hardware. Personal computers were gradually establishing themselves as tools that people could use in their everyday lives, and the result was that the number of PCs users grew substantially. Perhaps inevitably, more users also started creating viruses. Gradually, we witnessed the appearance of the first viruses and Trojans for DOS, written in assembler language and demonstrating a degree of skill on the part of their authors.

Far less programmers know assembler language than are familiar with high-level languages that are far easier to learn. Malicious code written in Fortran, Basic, Cobol, C or Pascal soon began to appear. The last two languages, which are well established and very powerful, are the most widely used, particularly in their TurboC and Turbo Pascal versions. This ultimately led to the appearance of “virus families”: that is, viruses that are followed by a vast number of related viruses which are slightly modified forms of the original code.

Other users took the less ‘artistic’ approach of creating destructive viruses that did not require any great knowledge of programming. As a result, batch processing file viruses or BAT viruses began to appear.

Win16 viruses

The development of 16-bit processors led to a new era in computing. The first consequence was the birth of Windows, which, at the time, was just an application to make it easier to handle DOS using a graphic interface.

The structure of Windows 3.xx files is rather difficult to understand, and the assembler language code is very complicated, as a result of which few programmers initially attempted to develop viruses for this platform. But this problem was soon solved thanks to the development of programming tools for high-level languages, above all Visual Basic. This application is so effective that many virus creators adopted it as their ‘daily working tool’. This meant that writing a virus had become a very straightforward task, and viruses soon appeared in their hundreds. This development was accompanied by the appearance of the first Trojans able to steal passwords. As a result, more than 500 variants of the AOL Trojan family -designed to steal personal information from infected computers- were identified.

part 7

This seventh edition on the history of computer viruses will look at how the development of Windows and Visual Basic has influenced the evolution of viruses, as with the development of these, worldwide epidemics also evolved such as the first one caused by Melissa in 1999.

While Windows changed from being an application designed to make DOS easier to manage to a 32-bit platform and operating system in its own right, virus creators went back to using assembler as the main language for programming viruses.

Versions 5 and 6 of Visual Basic (VB) were developed, making it the preferred tool, along with Borland Delphi (the Pascal development for the Windows environment), for Trojan and worm writers. Then, Visual C, a powerful environment developed in C for Windows, was adopted for creating viruses, Trojans and worms. This last type of

malware gained unusual strength, taking over almost all other types of viruses. Even though the characteristics of worms have changed over time, they all have the same objective: to spread to as many computers as possible, as quickly as possible.

With time, Visual Basic became extremely popular and Microsoft implemented part of the functionality of this language as an interpreter capable of running script files with a similar syntax.

At the same time as the Win32 platform was implemented, the first script viruses also appeared: malware inside a simple text file. These demonstrated that not only executable files (.EXE and .COM files) could carry viruses. As already seen with BAT viruses, there are also other means of propagation, proving the saying "anything that can be executed directly or through a interpreter can contain malware." To be specific, the first viruses that infected the macros included in Microsoft Office emerged. As a result, Word, Excel, Access and PowerPoint become ways of spreading ‘lethal weapons’, which destroyed information when the user simply opened a document.

Melissa and self-executing worms

The powerful script interpreters in Microsoft Office allowed virus authors to arm their creations with the characteristics of worms. A clear example is Melissa, a Word macro virus with the characteristics of a worm that infects Word 97 and 2000 documents. This worm automatically sends itself out as an attachment to an e-mail message to the first 50 contacts in the Outlook address book on the affected computer. This technique, which has unfortunately become very popular nowadays, was first used in this virus which, in 1999, caused one of the largest epidemics in computer history in just a few days. In fact, companies like Microsoft, Intel or Lucent Technologies had to block their connections to the Internet due to the actions of Melissa.

The technique started by Melissa was developed in 1999 by viruses like VBS/Freelink, which unlike its predecessor sent itself out to all the contacts in the address book on the infected PC. This started a new wave of worms capable of sending themselves out to all the contacts in the Outlook address book on the infected computer. Of these, the worm that most stands out from the rest is VBS/LoveLetter, more commonly known as ‘I love You’, which emerged in May 2000 and caused an epidemic that caused damage estimated at 10,000 million euros. In order to get the user’s attention and help it to spread, this worm sent itself out in an e-mail message with the subject ‘ILOVEYOU’ and an attached file called ‘LOVE-LETTER-FOR-YOU.TXT.VBS’. When the user opened this attachment, the computer was infected.

As well as Melissa, in 1999 another type of virus emerged that also marked a milestone in virus history. In November of that year, VBS/BubbleBoy appeared, a new type of Internet worm written in VB Script. VBS/BubbleBoy was automatically run without the user needing to click on an attached file, as it exploited a vulnerability in Internet Explorer 5 to automatically run when the message was opened or viewed. This worm was followed in 2000 by JS/Kak.Worm, which spread by hiding behind Java Script in the auto-signature in Microsoft Outlook Express, allowing it to infect computers without the user needing to run an attached file. These were the first samples of a series of worms, which were joined later on by worms capable of attacking computers when the user is browsing the Internet.