Viruses and spyware are two kinds of usually malicious software that you need to protect your computer against. You need antispyware technology to help prevent spyware, and you need to keep it regularly updated.

Some threats to the security of Windows are;

I. ***Worms***
II. ***Spyware***
III. ***Viruses***

- ✓ **Security in Microsoft Windows**

While the Windows 9x series offered the option of having profiles for multiple users, they had no concept of access privileges, and did not allow concurrent access; and so were not true multi-user operating systems. In addition, they implemented only partial memory protection. They were accordingly widely criticised for lack of security.

The Windows NT series of operating systems, by contrast, are true multi-user, and implement absolute memory protection. However, a lot of the advantages of being a true multi-user operating system were nullified by the fact that, prior to Windows Vista, the first user account created during the setup process was an administrator account, which was also the default for new accounts. Though Windows XP did have limited accounts, the majority of home users did not change to an account type with fewer rights – partially due to the number of programs which unnecessarily required administrator rights – and so most home users ran as administrator all the time.

Windows Vista changes this by introducing a privilege elevation system called User Account Control. When logging in as a standard user, a logon session is created and a token containing only the most basic privileges is assigned. In this way, the new logon session is incapable of making changes that would affect the entire system.

When logging in as a user in the Administrators group, two separate tokens are assigned. The first token contains all privileges typically awarded to an administrator, and the second is a restricted token similar to what a standard user would receive. User applications, including the Windows Shell, are then started with the restricted token, resulting in a reduced privilege environment even under an Administrator account. When an application requests higher privileges or "Run as administrator" is clicked, UAC will prompt for confirmation and, if consent is given (including administrator credentials if the account requesting the elevation is not a member of the administrators group), start the process using the unrestricted token.

Windows Defender

On January 6, 2005, Microsoft released a beta version of Microsoft AntiSpyware, based upon the previously released Giant AntiSpyware. On February 14, 2006, Microsoft AntiSpyware became Windows Defender with the release of beta 2. Windows Defender is a freeware program designed to protect against spyware and other unwanted software. Windows XP and Windows Server 2003 users who have genuine copies of Microsoft Windows can freely download the program from Microsoft's web site, and Windows Defender ships as part of Windows Vista.

File Permissions

At windows version from window NT3 have been based on a file system permission system referred to as AGLP (Account, Global, Local, and Permission). In essence, where file permissions are applied to the file and folder in the form of a local group, which then has other global group.

CHAPTER NINE

INTRODUCTION

The fact that an operating system is computer software makes it prone to error just as any human creation. Programmers make mistakes, and inefficient code is often implemented into programs even after testing. Some developers perform more thorough testing and generally produce more efficient software. Therefore, some operating systems are more error prone while others are more secure.

1.1Security in Computer

The branch of computer technology known as information security as applied to computers and networks is the computer security. The objective of computer security includes protection of information and property from theft, corruption, or natural disaster, while allowing the information and property to remain accessible and productive to its intended users. The term computer system security means the collective processes and mechanisms by which sensitive and valuable information and services are protected from publication, tampering or collapse by unauthorized activities or untrustworthy individuals and unplanned events respectively.

The technologies of computer security are based on logic. As security is not necessarily the primary goal of most computer applications, designing a program with security in mind often imposes restrictions on that program's behavior.
Security in this regard could be referred to as the protection mechanism used to safeguard information in the computer. Security has many facets and the three of the more important ones are

1. The threats
2. The nature of intruders
3. Accidental data loss

1. **Threats**: From security perspective, computer systems have three general goals, with corresponding threats.

(i) Data Confidentiality: Concerned with having secret data remain secret meaning if the owner of some data has decided that these data are available to certain people and no others, the system should guarantee that release of the data to unauthorized people does not occur.

(ii) Data Integrity: Means that unauthorized users should not be able to modify any data without the owner's permission. If a system cannot guarantee that data deposited in it remain unchanged until the owner decides to change them, it is not worth much as an information system.

(iii) System Availability: Means that nobody can disturb the system to make it usable. If a computer is an internet server, sending a flood of requests to it may cripple it by eating up all of its CPU time just examining and discarding incoming requests. Another aspect of security problem in operating system is Privacy that is protecting individuals from misuse of information about them.

(2) **The nature of intruders:** In the security literature, people who are nosing around places where they have no business being are called ***intruders*** or sometimes called ***adversaries.*** Intruders act in two different ways

1 ***Passive Intruders****:* just want to read files they are not authorised to read.

2 ***Active Intruders***: are more malicious; they want to make unauthorised changes to data.

When designing a system to be secure against intruders, it is important to keep in mind the kind of intruder one is trying to protect against. Some categories are

(a) Casual prying by nontechnical users.

(b) Snooping by insiders.

(c) Determined attempts to make money.

(d) Commercial or military espionage.

Another category of security pest that has manifested itself in recent times is the ***Virus***. The term virus is a piece of code that replicates itself and usually does some damage.

(3) **Accidental data loss:** In addition to threats caused by malicious intruders, valuable data can be lost by accident. Some of the accidental data losses are

(1) ***Acts of God***: fires, flood, earthquakes, wars, etc.

(2) ***Hardware or software errors***: CPU malfunction, unreadable disks or tapes, telecommunication errors, program bugs.

(3) ***Human errors:*** incorrect data entry, wrong tape or disk mounted, wrong program run, lost disk or tape, or some other mistake.Most of these can be dealt with by maintaining adequate backups, preferably far away from the original data.

1.2 Operating system design, security and complexity

An operating system is a software component that acts as the core of a computer system. It performs various functions and is essentially the interface that connects your computer and its supported components. Various operating systems are in use today to satisfy the ever changing customer demands. Nevertheless the most widespread operating systems are: **Microsoft Windows, Linux/Unix and Macintosh**. **Windows** are mostly used as personal computers, **Linux/Unix** are mainly open source while **Apple Macs** are often used for graphic designs or other specialist applications. Irrespective of their application or use, all operating systems up to date have been subject to security compromises or likewise failures. It is a fact that the majority of hacking tools, viruses, worms or Trojan horses are written for Windows, but this is merely due to the fact that Windows occupy almost 90% of the global market.

The security of the operating system is therefore a necessity for the overall system security. Today most commercially developed operating systems provide security through authentication of the users, maintenance of access control mechanisms, separation of kernel and user spaces and providing trusted applications to modify or manage system resources. However the above security features are inadequate to protect the operating system from attacks in today's environment.

SECURITY ISSUES IN THE LINUX OPERATING SYSTEMS

While companies like Microsoft and Apple own the commercial software market, a variety of non profit organizations or intellectual individuals contribute constantly to the proliferation of the ***open source software***. In terms of operating systems, Linux is the example par excellence of all free ***open source systems***, Introduced by **Linux Torvalds**, at the time a student. Linux was the first fully functional operating system that was offered for free under the ***open source*** agreement to the public. Following this event,

with the participation of thousand of admirers across the globe, a myriad of ***open source*** Linux based operation systems flooded the cyber world.

There are different reasons that motivate thousands of people around the globe to participate in open source projects and release software to the public. Intellectual gratification, pleasure of creativity or of solving complex and challenging tasks, are of some the driving forces in this domain. Whatever the reasons, the benefits of using open source are manifold. Open-source software powers many of the web sites on the Internet, corporate computer, servers used for research and development, it can be found in digital video recorders , telephones, personal digital assistants (PDAs) watches, networking hardware, MP3 players and automobiles

Nevertheless, open source software does not come without issues or disadvantages. Even though the codes are available to thousands of eyes for scrutiny, there is no guarantee for security or optimal performance. Although that is ok for simple home applications it is not the case for enterprise, commercial or critical applications. Also because of lack of standardization and complex licensing issues, open source software is prone to misuse or abuse. Standardization is hard to achieve, because open source creators are completely free in their choice of design, implementation or adherence to existing standards. Usually standardization is enforced by market forces and industry regulators; however in the case of open source software both these factors do not exercise enough pressure to drive the process. Version proliferation is another major open source issue. As a matter of fact, this matter does not concern only open source software but commercial software as well. Nevertheless, the effect on the open source software is more evident. Developing many versions of a program in a short period not only confuses users but also requires a steep learning curve. This is true in particular in the case of constant changes of graphical user interfaces and navigation concepts from one version to the next one. At the same time, constant introductions of new versions of a software package do not affect in positive way its reputation since the user might believe this is a sign of instability.

Another issue that affects ICT today is that of the implementation of open source software. Because, open source is developed by the co-operation of different individuals, it is hard to establish a proper working relationship with the person in charge (if there is any). Furthermore, technical support, documentation issues, no access to advice, are some of the problems that a company or individual that uses open source software might face. Hardware

and software compatibilities influence also the processes ever since most of hardware manufacturers do not expose their trade secrets, therefore not allowing access to their codes to open source developers. Open source developers have no other choice but to design and release their own code (hardware drivers) therefore contributing to the complexity and lack of standardization.

Future prognosis on the Linux operating system

The open source phenomenon is definitely influencing in a positive way ICT and probably the trend will not change in the future. Open source projects are available to "millions of eyes" for scrutiny, improvement or testing. Nevertheless, it is likely that in the future will continue to experience the same issues mentioned above with some improvements in the area of standardization. Some open source will definitely transform in commercial software provided that they have matured enough and captured a significant market share. Red Hat Linux for example, is a typical example of how open source software becomes commercial under the right circumstances.

SECURITY ISSUES IN THE WINDOWS OPERATING SYSTEM

Security is the main problem that Windows operating systems are facing since their introduction. Lack of vision from its developers regarding security is probably the main reason behind this issue. The first windows were designed to be simple and productive but not very secure. Although new operating systems versions were introduced within the last 5-10 years, the same issues with security persisted. In my opinion, because of market pressure and product development circles, it was almost impossible for Microsoft to totally change their operating system approach. Instead they continued to build on top of each previous model. Unfortunately, their operating systems are still vulnerable affecting significantly ICT applications worldwide.

To understand the impact of operating system vulnerabilities on ICT suffice to look at the case of "SQL slammer', a worm released on the web in 2003 (Forte, 2003). Slammer, also known as "SQL hell", is a worm that affected Microsoft Windows operating systems in January 2003 affecting within ten minutes 75 thousands machines worldwide. Slammer exploited vulnerability in SQL server and desktop engine slowing down communications and

affecting businesses financially worldwide. Following this incident several modified slammer versions were released online. This is not an isolated episode that demonstrates the lack of security vision and poor operating system design. Viruses and worms like Melissa, code red, sasser, nimda, donut, spida or slapper have also impacted information communication telecommunications globally. In 2007 Computer Economics, a well known research company conducted a research on the impact of the malware globally estimating a $ 13 billion in financial losses in 2006 only. It is quite obvious how ICT and global communications are affected by malware which attributes its success to operating system vulnerabilities.

Microsoft has been able to take care of malware attacks by releasing patches or services packs. Although, it looks like this is the right approach to this problem it does eradicate the problem and provide temporarily relief from threats. We need a holistic approach to deal with the root of the problem not with it consequences. As a matter of fact operating system patches manage to avoid the threat temporarily, since what they usually do is a mere change of names or locations of important operating system files used by malware. In other occasions Microsoft has even discontinued shipping certain programs with its operating systems as a security measure against malware, therefore not dealing with the main problems: design and security.

In 2007, Microsoft released officially to the public Windows Vista and in October 2009 Windows seven. Although, the graphical user interfaces look impressive, both operating systems are still vulnerable to malware or system hacking. A very simple example is the ‘the sticky key backdoor’, one of Vista’s vulnerabilities. Vinoo Thomas, a McAfee researcher, in 2007 released a blog online informing the public about the Sticky Keys vulnerability. Vista apparently does allow the modification of sethc.exe file (located at: C:/windows/system32/sethc.exe) and no integrity checks are performed before execution. **Authentication** can simply be bypassed by replacing this file with cmd.exe using a live CD like Backtrack or direct logging and entering windows explorer (Vinoo, 2007).

Moreover, Vista activation mechanism has been broken almost one year after its official release. The same security scenario applies to Windows seven. This operating system can be bypassed in the same way as Windows Vista (using the installation disk and entering recovery mode). Furthermore online news of a zero day attack is spreading around. Certainly, this poor

security performance of Microsoft Windows, the most used operating system worldwide, does not sound promising for ICT and its future.

Future prognosis on the windows operating system

Operating system design is a factor that will influence ICT in the times to come. The main reason is security. If we take in consideration the fact that digital globalization is facilitating the distribution of malware and the number of internet users is rising exponentially, we should expect more sophisticated attacks on the windows operating systems and ICT. Under these circumstances, we should review the windows operating system design strategy, focusing on security and build reliable operating system.

CHAPTER NINE:
SECURITY ISSUES IN THE UNIX OPERATING SYSTEMS

There are a number of elements that have lead to the popularity of the UNIX operating system in the world today. The most notable factors are its portability among hardware platforms and the interactive programming environment that it offers to users. In fact, these elements have had much to do with the successful evolution of the UNIX system in the commercial market place. As the UNIX system expands further into industry and government, the need to handle UNIX system security will no doubt become imperative. For example, the US government is committing several million dollars a year for the UNIX system and its supported hardware. The security requirements for the government are tremendous, and one can only guess at the future needs of security in industry. In this paper, we will cover some of the more fundamental security risks in the UNIX system. Discussed are common causes of UNIX system compromise in such areas as file protection, password security, networking and hacker violations. In our conclusion, we will comment upon ongoing effects in UNIX system security and their direct influence on the portability of the UNIX operating system.

In the UNIX operating system environment, files and directories are organized in a tree structure with specific access modes. The setting of these modes through permission bits (as octal digits), is the basis of UNIX system security. Permission bits determine how users can access files and the type of access they are allowed. There are three user access modes for all UNIX system files and directories: the owner, the group, and others. Access to read, write and execute within each of the user types is also controlled by permission bits. Flexibility in file security is convenient, but it has been criticized as an area of System security compromise.

FILE AND DIRECTORY SECURITY

In the UNIX operating system environment, files and directories are organized in a tree structure with specific access modes. The setting of these modes, through permission bits (as octal digits), is the basis of UNIX system security. Permission bits determine how users can access files and the type of access they are allowed. There are three user access modes for all UNIX system files and directories: the owner, the group, and others. Access to read, write and execute within each of the user types is also controlled by permission bits (Figure 1). Flexibility in file

security is convenient, but it has been criticized as an area of system security compromise.

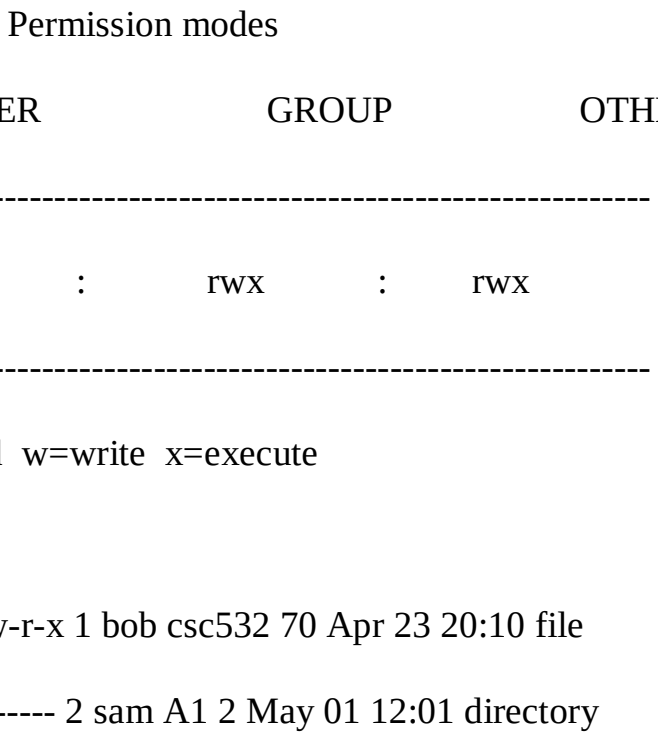

FIGURE 1. File and directory modes: File shows Bob as the owner, with read and writes permission. Group has write permission, while others has read and execute permission. The directory gives a secure directory not readable, writeable, or executable by Group and Others.

Since the file protection mechanism is so important in the UNIX operating system, it stands to reason that the proper setting of permission bits is required for overall security. Aside from user ignorance, the most common area of file compromise has to do with the default setting of permission bits at file creation. In some systems the default is octal 644, meaning that only the file owner can write and read to a file, while all others can only read it. (3) In many "open" environments this may be acceptable. However, in cases where sensitive data is present, the access for reading by others should be turned off. The file utility umask does in fact satisfy this requirement. A suggested setting, umask 027, would enable all permission for the file owner, disable write permission to the group, and disable permissions for all

others (octal 750). By inserting this umask command in a user .profile or .login file, the default will be overwritten by the new settings at file creation. The CHMOD utility can be used to modify permission settings on files and directories. Issuing the following command,

```
chmod u+rwd,g+rw,g-w,u-rwx file
```

will provide the file with the same protection as the umask above (octal 750). Permission bits can be relaxed with chmod at a later time, but at least initially, the file structure can be made secure using a restrictive umask. By responsible application of such utilities as umask and chmod, users can enhance file system security. The Unix system, however, restricts the security defined by the user to only owner, group and others. Thus, the owner of the file cannot designate file access to specific users. As Kowack and Healy have pointed out, "The granularity of control that (file security) mechanisms is often insufficient in practice (...) it is not possible to grant one user write protection to a directory while granting another read permission to the same directory. (4) A useful file security file security extension to the Unix system might be Multics style access control lists. With access mode vulnerabilities in mind, users should pay close attention to files and directories under their control, and correct permissions whenever possible. Even with the design limitations in mode granularity, following a safe approach will ensure a more secure Unix system file structure.

DIRECTORIES

Directory protection is commonly overlooked component of file security in the Unix system. Many system administrators and users are unaware of the fact, that "publicly writable directories provide the most opportunities for compromising the Unix system security" (6). Administrators tend to make these "open" for users to move around and access public files and utilities. This can be disastrous, since files and other subdirectories within writable directories can be moved out and replaced with different versions, even if contained files are unreadable or unwritable to others. When this happens, an

unscrupulous user or a "password breaker" may supplant a Trojan horse of a commonly used system utility' *For example:*

Imagine that the /bin directory is publicly writable. The perpetrator could first remove the old version (with rm utility) and then include his own fake su to read the password of users who execute this utility.

Although writable directories can destroy system integrity, readable ones can be just as damaging. Sometimes files and directories are configured to permit read access by other. This subtle convenience can lead to unauthorized disclosure of sensitive data: a serious matter when valuable information is lost to a business competitor.
As a general rule, therefore, read and write access should be removed from all but system administrative directories. Execute permission will allow access to needed files; however, users might explicitly name the file they wish to use. This adds some protection to unreadable and unwritable directories. So, programs like lp file.x in an unreadable directory /ddr will print the contents of file.x, while ls/ddr would not list the contents of that directory.

USER AUTHENTICATION

Another area is the user authentication. In the UNIX system, authentication is accomplished by personal passwords. Though passwords offer an additional level of security beyond physical constraints, they lend themselves to the greatest area of computer system compromise. Lack of user awareness and responsibility contributes largely to this form of computer insecurity. This is true
of many computer facilities where password identification, authentication and authorization are required for the access of resources, and the Unix operating system is no exception. Password information in many time-sharing systems are kept in restricted files that are not ordinarily readable by users. The UNIX system differs in this respect, since it allows all users to have read access to the /etc/passwd file where encrypted passwords and other user information are stored.

DATA ENCRYPTION

Although the Unix system implements a one-way encryption method, and in most systems a modified version of the data encryption standard (DES), password breaking methods are known. Among these methods, brute-force attacks are generally the least effective, yet techniques involving the use of heuristics (good guesses and knowledge about passwords) tend to be successful. For example, the /etc/passwd file contains such useful information as the login name and comments fields. Login names are especially rewarding to the "password breaker" since many users will use login variants for passwords (backward spelling, the appending of a single digit etc.). The comment field often contains items such as surname, given name, address, telephone number, project name and so on. To quote Morris and Grampp in their landmark paper on Unix system security: The authors made a survey of several dozen local machines, using as trial passwords a collection of the 20 most common female first names, each followed by a single digit. The total number of passwords tried was therefore 200. At least one of these 200 passwords turned out to be a valid password on every machine surveyed. If an intruder knows something about the people using a machine, a whole new set of candidates is available. Family and friend's names, auto registration numbers, hobbies, and pets are particularly productive categories to try interactively in the unlikely event that a purely mechanical scan of the password file turns out to be disappointing.

Thus, given a persistent system violator, there is strong evidence, that he will find some information about users in the /etc/passwd file. With this in mind, it is obvious that a password file should be unreadable to everyone except those in charge of system administration.

SECURITY ISSUES IN THE MACINTOSH OPERATING SYSTEM

It's been called one of the safest operating systems of all times, but the Mac's OS X Tiger may not be as safe as it seems. Mac's OS X Tiger has become a favorite among Mac users for its bells and whistles and its UNIX based architecture. From a power user to newbie, Tiger provides both comfort and security for all OS X users. Some of the flaws found in its security is as follows:

- *FAILING TO USE ITS SOFTWARE UPDATE*

Regularly updating Tiger's software is one of the easiest ways to keep your computer protected from the latest exploits and malicious Internet content. In January of this year, a couple of computer guru's published "The Month of Apple Bugs"(MoAB) -- a website dedicated to pointing out 31 of OS X's vulnerabilities and security flaws. After reviewing the website, Apple acted promptly and has since released several updates addressing the critical bugs. With software updates turned off, there's a good chance the computer could fall victim to one of MoAB's exploits.

- *MINDLESSLY SURFING WITH SAFARI*

Although much safer than Microsoft's Internet Explorer, Tiger's default web browser Safari is not immune to security flaws. To obtain the safest Internet browsing experience, a few of Safari's features should be modified: Make sure all "AutoFill" options are disabled, and always use "Private Browsing" on each of the computer's accounts. Although surfing without "Private Browsing" enabled could save you some time, in the long run you're simply opening yourself up to greater security risks.

To be ultra conservative with web browsing, one can disallow all cookies and remove all existing cookies via the "Show Cookies" dialog. (Keeping in mind that some websites require cookies for complete functionality). By not accepting cookies, one may be limiting web browsing experience, so one is torn in between securing his/her system or enjoying the web experience.

- *INCORRECTLY CONFIGURING SECURITY PREFERENCES*

Tiger's security panel features a handful of security preferences which permit users to select varying levels of security based upon their particular usage requirements. Obviously, however, when configuring your security preference it's important to understand what each option does, and the benefits of a particular setting: One of the most important and often overlooked preferences is whether to permit automatic login. Requiring a password to wake the computer is imperative in preventing unauthorized access to unattended computers. "Disabling automatic login is necessary for any level of security. If you enable automatic login, an intruder can automatically log in without having to authenticate. Even if you automatically log in with a very restricted user account, this makes it much easier to perform malicious actions on the computer."

In addition to requiring a user to login after the computer has been asleep, it is also important to require an additional login wherever an important system

wide preference is being changed. In order to prevent faulty administration, either from a malicious user or just from an unwitting friend who accidentally makes a system change, it is important to require an extra step of authentication when altering system preferences. After all, we can all make mistakes when toggling options, but by requiring an extra authentication step whenever a system preference is changed, you can make sure that many of these types of errors never occur.

- *LEAVING UNUSED HARDWARE DEVICES ENABLED*

Most of us are no longer worried by our Internet connections, but instead connect to the Internet through multiple styles of broadband connections. For example, instead of being tied down to Ethernet cables at home, users are taking advantage of wireless connections using their laptops from anywhere, and connecting their Bluetooth devices to their computer for extra support. While having different types of broadband connections is great for the user, it's awful for the security of the computer. To protect your computer, you should make sure to "disable any unused hardware devices listed in Network preferences. Enabled, unused devices (such as AirPort and Bluetooth) are a security risk." While we're not suggesting that you do away with these great feature altogether, we are saying that when they're not in use, you should turn them off.

- *TROJAN HORSE ALERT*

Recently, a new variant of the Hell Raiser Trojan Horse, which was identified as OSX/HellRTS.D, has been discovered. Experts have analyzed this new variant, and it is detected in the latest MacScan spyware definitions update as HellRaiser Trojan Horse 4.2. MacScan has detected previous variants of this trojan horse since 2005. HellRaiser is a trojan horse that allows complete control of a computer by a remote attacker, giving the attacker the ability to transfer files to and from the infected computer, pop up chat messages on the infected system, display pictures, speak messages, and even remotely restart or shut down the infected machine. The attacker can search through the files on the infected computer, choosing exactly what they want to steal, view the contents of the clipboard, or even watch the user's actions on the infected computer.

In order to become infected, a user must run the server component of the Trojan horse, which can be disguised as an innocent file. The attacker then

uses the client component of the Trojan horse to take control of the infected system.

SECURITY ISSUES IN SOLARIS OPERATING SYSTEM

Solaris or Oracle Solaris as it is now known is a UNIX-based operating system introduced by Sun Microsystems in 1992 as the successor to SunOS. The prominent flaw in Solaris operating system is the multiple security vulnerabilities in PostgreSQL Shipped with Solaris 10 which allows the Elevation of Privileges or Denial of Service (DoS)

Multiple Security vulnerabilities affecting the PostgreSQL software shipped with Solaris 10 may allow a local or remote user who has access to the PostgreSQL server to cause a Denial of Service (DoS) to the PostgreSQL instance or the server it runs on (due to excessive resource consumption), or to gain elevated privileges on the server.

Regular Expression Denial-of-Service

(CVE-2007-4772, CVE-2007-6067, CVE-2007-4769):

Three separate issues in the regular expression libraries used by PostgreSQL allow malicious user to initiate a denial-of-service by passing certain regular expressions in SQL queries.

First, users could create infinite loops using some specific regular expressions.

Second, certain complex regular expressions could consume some excessive amounts of memory.

Third, out-of-range backref numbers could be used to crash the backend.

All of these issues have been patched.

DBLink privilege Escalation (CVE-2007-6601):

DBLink functions combined with local trust or ident authentication could be used by malicious user to gain superuser privileges.

This issue has been fixed and does not affect users who have not installed DBLink (an optional module), or who are using password authentication for local access. This same problem was addressed in the previous release cycle.

These issues can occur in the following releases

SPARC (Scalable Processor Architecture) Platform

Solaris 10 PostgreSQL 8.1

Without patch 123590-08

Solaris 10 PostgreSQL 8.2

Without patch 136998-02

X86 Platform

Solaris 10 PostgreSQL 8.1
Without patch 123591-08
Solaris 10 PostgreSQL 8.2
Without patch 136999-02

Solaris 8 and 9 do not ship with PostGreSQL and are not impacted by this issue.
A user exploiting this vulnerability must have an account on the PostgreSQL server.
This issue affects PostGreSQL versions 7.4x prior to 7.4.19, 8.0.x prior to 8.0.15, 8.1.x prior to 8.1.11 and 8.2.x prior to 8.2.6.

CONCLUSION

Information and Communication Technologies (ICT) will provide benefits to our society for years to come. The proliferation of these technologies or their decline will be affected amongst all by security issues on the area of operating system design and security, open source issues, and design complexity. Therefore, designing better operating systems, improving on their security, are some of the challenges for the future. If we take in consideration the fact that digital globalization is facilitating the distribution of malware and the number of internet users is rising exponentially, we should expect more sophisticated attacks on the windows operating systems and ICT. Under these circumstances, we should review the windows operating system design strategy, focusing on security and build reliable operating system.

The open source phenomenon is definitely influencing in a positive way ICT and probably the trend will not change in the future. Open source projects are available to “millions of eyes” for scrutiny, improvement or testing. Nevertheless, it is likely that in the future will continue to experience the same issues mentioned above with some improvements in the area of standardization.

Some open source will definitely transform in commercial software provided that they have matured enough and captured a significant market share. Red Hat Linux for example, is a typical example of how open source software becomes commercial under the right circumstances.

CHAPTER TEN: