Operating Systems Security

Microsoft Windows Vulnerability Report

Ajit Gaddam
ajit@ospreysecurity.com

Abstract**—** Classic Operating System (OS) protection techniques are no longer able to protect against the constant evolving threat landscape to any IT infrastructure. Increasingly, untrustworthy software can reside on top of an OS, including active web code and remotely exploited applications, which means that it is no longer sufficient to protect users from each other.

This paper analyzes the vulnerability disclosures and security updates during the year 2007 for the Microsoft Windows Vista Operating System when compared to its predecessor, Microsoft Windows XP, along with other modern client Operating Systems of that era - Red Hat, Ubuntu, and Apple Mac OS X.

The results of this analysis based on the Vulnerability Count Metric and Days of Risk suggest that Windows Vista is the most secure Operating System when compared to the other leading Desktop Operating Systems for the year 2007. The analysis also reveals that Firefox 2.x on Ubuntu platform was the most secure browser for the year 2007 in terms of the lowest Days of Risk and vulnerability profile.

Keywords—operating systems, security, Microsoft, OSX, RedHat, Linux, vulnerability, risk

I. Introduction

With Operating System Security, there is a need for preventing the unauthorized reading or modification of data, or the unauthorized use of resources. In the classical sense, security leverages the idea of protecting and isolating users/applications from each other. The operating system and other application programs authenticate a user and the user is the basis of protection policies. The assumption is that applications perform according to their specifications.

The rise of the Internet, however, has drastically reduced the trust that can be placed in software running on our Operating Systems. First, a tremendous amount of active content can be run simply by loading a Web page or reading an e-mail, thereby lowering the barriers to entry for malware. In addition, the use of spyware and adware, disguised as legitimate software, is on the rise. Local applications are interpreting and accessing more and more remote content, increasing the likelihood that a malicious user can exploit vulnerabilities to take control of the program, the operating system, and finally the machine itself. Operating System vendors use the term “secure” to describe the state of their products. However, the reality is that how secure a system really is lies in the *implementation* of the Operating System itself.

While the results in this paper represent only the vulnerability dimension of security risk, they do provide insight into the aspects of security quality that are under the control of the vendors – code security quality and security response. These metrics however, must be considered in combination with several other important qualitative factors when choosing a platform based upon security maintenance and likelihood of a security breach in your environment.

Beyond patches and vulnerabilities, there are “softer” qualities of security that are difficult to quantify but impact deployed security. Qualities like security lifecycle support, bulletin descriptiveness, default security features, and the like all have a direct impact on deployed role security.

II. Measuring Operating System Security

How does one measure the security of an Operating System or for that matter, measure Information Security? After all, the common mantra seems to be “You can’t manage what you can’t measure” and “what gets measured gets done.” So, by measuring the number of vulnerabilities and their severity, can we determine how secure an Operating System is? Can these metrics predict future risk trends? What do these numbers actually tell us? Moreover, if these numbers are lower this year from last year or their first year, can the vendor claim success about providing enhanced security?

This Operating System Security paper utilizes the approach presented by Jeff Jones of Microsoft on Windows Visa vulnerabilities [1] while the approach for metrics is based on an ISSA paper on Seven Myths about Information Security Metrics [2]. Measuring anything makes it easier to drive improvements but measuring the wrong things leads to improving the wrong things. The hard part is not measurement per se but figuring out the suite of parameters that need altering and to measure and work on them all, acknowledging that many of the measures are interdependent. Information Security is a complex field with ramifications throughout the organization. It is unrealistic to expect to find a few simple measures.

Since Operating System security is all about reducing risk and process outputs such as better audit reports, reduction in virus incidents, and reduction in vulnerabilities are all worthwhile sources of metrics. In addition, using the '*vulnerability count*' and '*days-of-risk*' will provide a vulnerability analysis that could be incorporated with other factors such as various kinds of controls and defense-in-depth measures to provide for reduced risk for different kinds of environments whether in an enterprise or in a home environment.

A. Days of Risk as a Metric

Days-of-Risk (DoR) is a measurement of the time-period of greatly increased risk from when a vulnerability has been publicly disclosed (and thus known and available to script-kiddies and other malicious attackers) until a vendor patch is available to close the vulnerability.

The vulnerability lifecycle chart below from Bruce Schneier [2] illustrates key points on where risk might increase dramatically. Essentially, days-of-risk as it is commonly used is the time from "vulnerability announced" until the "Vendor patches vulnerability" is pointed out in this chart below.

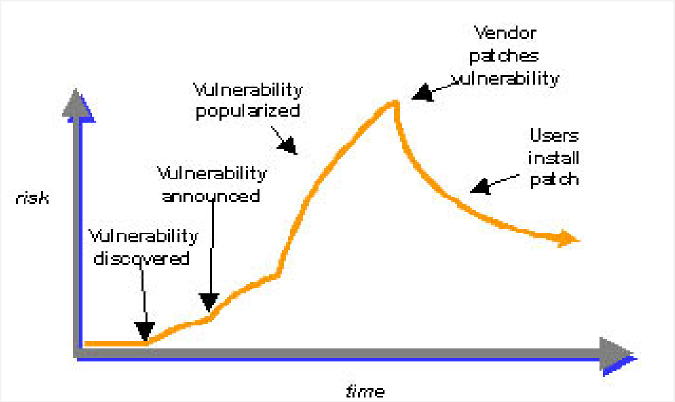

Fig. 1. Vulnerability Lifecycle chart

The potential usefulness of days-of-risk as a metric is around monitoring security risk reduction along with providing an incentive to vendors to expedite patches. Though days-of-risk relates closely to vendor maintenance and security response process, it is worth noting that there are actions that can be taken to help reduce the total number of vulnerabilities and move towards reducing the opportunities for vulnerability discovery [3].

B. Quantitative Metrics Description & Assumptions

Any comparisons of various Operating System security measurement historically was made using quantitative and readily available data and counts of "security advisories or bulletins" issued by the different vendors.

While these counts are popular, vendors control the number of vulnerabilities addressed by a single security advisory. Therefore, to compensate the inherent weakness of just using raw bulletin counts, *Days of Risk* is used along with a role-based approach of measuring security of these different operating systems using a most-likely deployed client configuration environment. This analysis will also take advantage of Linux ability to create and deploy a minimum set of components, a security advantage it has over Windows.

The overall vulnerability information as of this publication for 2007 vulnerabilities was obtained from Microsoft TechNet reports, vendor listings, and other security bug reporting lists [4-16]. The following is the set of conditions that was used to gather the vulnerability information and patch information

i. It accounts for patches and fixes only released by Microsoft. Similarly, for Red Hat and the other Operating Systems, only the install patches released by the vendor are considered.

ii. The "first public" date for a vulnerability is the date at which the vulnerability was first released on a public list or a web site (Bugtraq, Red Hat, Microsoft, Full-disclosure, Security Focus, k-otik) devoted to security, or a publicly accessible list of bugs or problems posted to the home site of a package or its mailing list.

iii. Dates of patches are based on the release date for the distribution of interest.

iv. Release dates for a vulnerability patch or fix are specific to a distribution/architecture. If a fix for a component (ex: libpng) is released on 01/01/2007 for a certain Linux distribution (ex: Gentoo Linux) and a fix for the same issue is released for Red Hat on 01/10/2007, the release date for the fix on Red Hat will be 01/10/2007. This is not applicable for the Windows platform.

v. For past issues, the release date for a patch is the first published vendor report that includes the patch for the applicable platform for which the patch fully fixed the vulnerability. If the patch had to be re-issued to address some portion of the security issue, the later date is used.

C. Time Period

The methodology used for this comparison can be applied to any fixed time-period for comparisons. For Server 2003 analysis, vulnerabilities disclosed earlier than 2007 are used if any only if Microsoft has released a fix for these issues in 2007. Similarly, a vulnerability announced in 2007 but fixed in 2008 will not be considered.

When considering the relative security of Windows Server 2003, it is important not just to consider *what* is installed, but also *how* it is installed. Thus, the context in which a role is deployed is important when considering the long-term security of a solution.

D. *Mitre CVE List*

In this analysis, the CVE or CAN identifier of a vulnerability is used. CVE stands for Common Vulnerabilities and Exposures and provides a standardized taxonomy for all publicly known vulnerabilities and exposures. In this analysis, a vulnerability is considered distinct if it has its own CVE number. In rare instances, it is possible for a vulnerability to not have been assigned a CVE number. In such a case, this vulnerability is not considered for this analysis.

III. Windows Operating System Vulnerability Comparison

A. *Windows Visa – Year 2007*

With the previous Windows Operating Systems, it never supported the principle of least privilege even in their discretionary access control systems. While there was some concept of a completely privileged security domain and a completely unprivileged security domain, Microsoft introduced many security measures with its latest flagship Operating System, Windows Vista.

Looking at security updates for Windows Vista during the year 2007, Microsoft released a total of 22 Security Bulletins and corresponding patches in the year 2007 affecting components of Windows Vista. These fixed 44 different vulnerabilities. For Windows Vista, the average Days of Risk for these vulnerabilities was 163.69 days.

Excluded components: Development (.Net Frameworks)

Included components: Browser (Internet Explorer 7), Windows Media Player 11, DirectX 10.0

To get a better feel for the frequency and impact of these security updates for administrators throughout the year, the histogram of Patch Events is shown on the graph below and accounts for the first fifty-two weeks of availability. There were ten, and no week had more than one security update. Charted out as Patch Events, Figure 2 shows what year 2007 looked like for Security Administrators for Windows Vista and days of risk presented in figure 3.

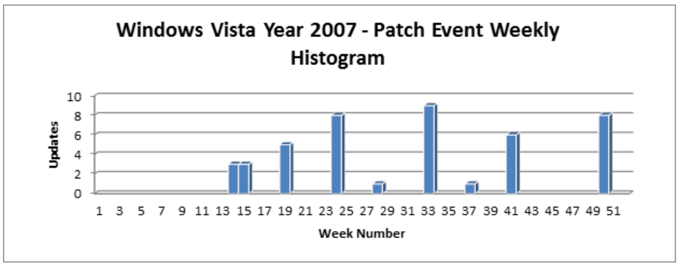

Fig. 2. Windows Vista Patch Event Weekly Histogram (year 2007)

The number of vulnerabilities for Windows Vista and the days of risk is shown below.

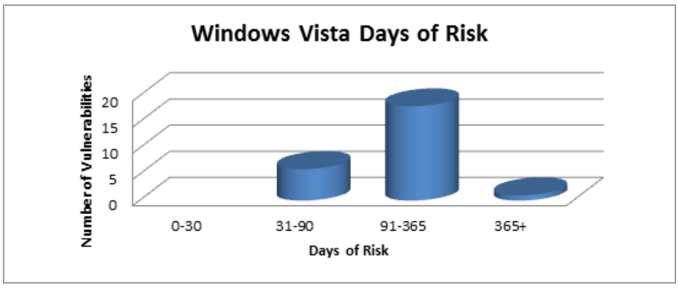

Fig. 3. Windows Vista Days of Risk (year 2007)

B. *Windows XP – Year 2007*

Windows XP shipped on October 25th 2001 and only matured as a secure Operating System especially after the release of Service Pack 2.

Excluded components: Development (.Net frameworks), Web (IIS)

Included components: The version of Windows XP considered is Windows XP Professional SP2 with Internet Explorer 6 Service Pack 1, Windows Media Player 10, and DirectX 9.0.

Windows XP in year 2007 had 61 different vulnerabilities that were patched over 39 different security bulletins. The average Days of Risk for these Windows XP vulnerabilities was 161.52 days. Charted out as Patch Events, Figure 4 shows what year 2007 looked like for Security Administrators for Windows XP and days of risk presented in figure 5.

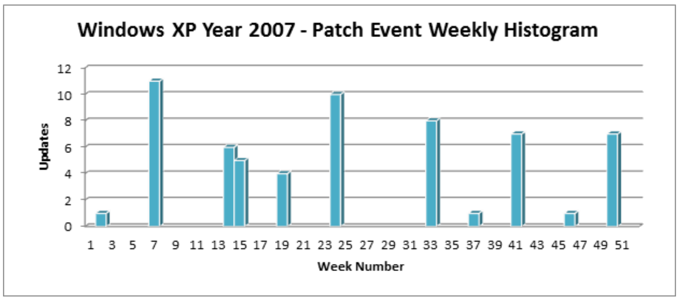

Fig. 4. Windows XP Patch Event Weekly Histogram (year 2007)

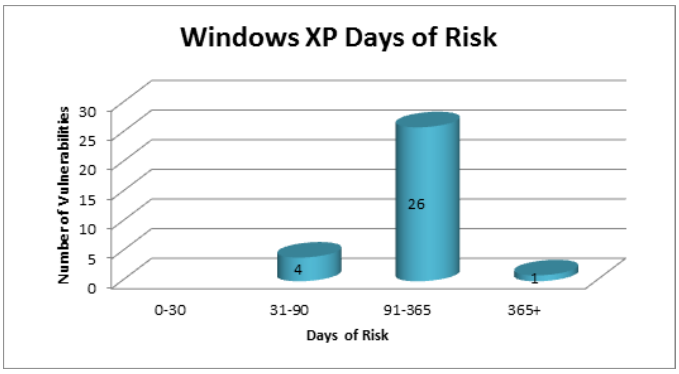

Fig. 5. Windows XP Days of Risk (year 2007)

C. *Side-by-side comparison of Windows Vista and Windows XP*

With the basic analysis completed for Windows Vista and Windows XP, we now have enough information to compare them. First, let us look at a chart that shows the total number of vulnerabilities fixed for Windows Vista and Windows XP side-by-side.

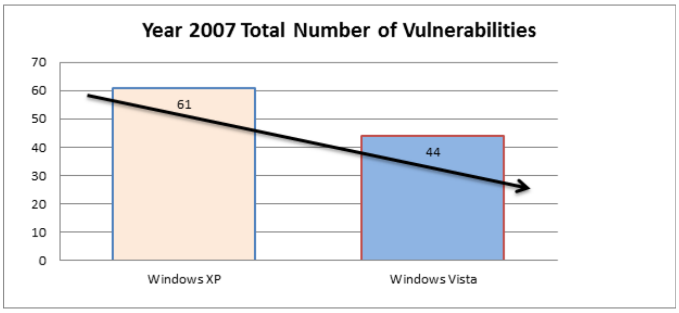

Fig. 6. Windows Vista and Windows XP comparison of total # of vulnerabilities (year 2007)

We can see a reduction in vulnerabilities from Windows XP (61) to Windows Vista (44) in the year 2007.

Next, let us examine the impact that security updates had for administrators by looking at Patch Events in Figure 4. Windows Vista's 44 security updates occurred across nine Patch Events in nine different weeks in the year 2007. Windows XP's 61 security updates occurred across 12 Patch Events in 12 different weeks in the year 2007.

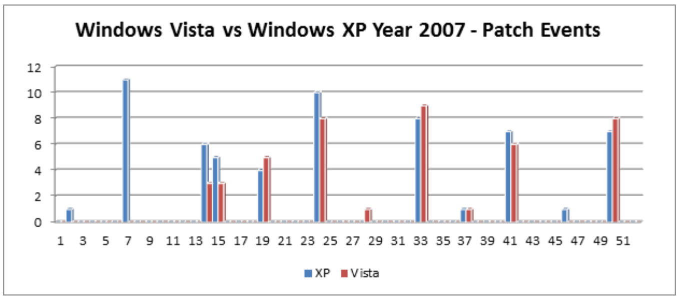

Fig. 7. Windows Vista vs. Windows XP comparison of patch events (year 2007)

Graphed out visually, it is easy to see that there is reduced work to manage security risk with Windows Vista in 2007 when compared to Windows XP in the year 2007.

Table 1 below is a summary table of the key data discussed above.

TABLE I. WINDOWS VISTA VS. WINDOWS XP PATCH EVENT COMPARISON

Metric	**Windows Vista**	**Windows XP**
Vulnerabilities fixed	44	61
Security Updates	21	39
Patch Events	9	12
Weeks with at least 1 Patch Event	9	12

IV. WINDOWS VISTA VS. OTHER OPERATING SYSTEMS

Other workstation products compared in this paper besides Microsoft Windows (Vista / XP) are those that offered long-term support options. These are Red Hat, Ubuntu, and Mac OS X 10. The comparison is performed using the flagship version that has been running for the whole of 2007.

A. *RedHat Enterprise Linux (RHEL)*

Red Hat shipped Red Hat Enterprise Linux 5 in March 2007. However, it will not be included in this analysis since it did not have a full year worth data as of this publication. Instead, the version of Red Hat that is used is Red Hat Enterprise Linux 4 Workstation (rhel4ws).

This analysis will not count the vulnerabilities for all the components for the product that Red Hat ships and supports as Red Hat Enterprise Linux 4 WS. To accommodate this idea, only a reduced set of components is used in comparisons.

Excluded components: Any component that is not installed by default, which includes all optional "server" components that ship with rhel4ws.

- Excluded components include text-internet, graphics (the gimp stuff) and office

(OpenOffice) and Development Tools (gcc, etc) installation groups.

- The rpm command is also used to list out all packages that get installed and used that package list to filter vulnerabilities for inclusion.

This process results in a Gnome – windows workstation that includes standard system management tools, Firefox for browsing, sound, and video support, but excludes all server packages, as well as OpenOffice and other optional stuff that a Windows system wouldn't have by default.

Included Components: Browser (Firefox), Mail (SpamAssassin, Mutt), Messaging (GnomeMeeting), Security (Kerberos, shadowutils, SASL), Utilities (BusyBox, util-linux, unzip, GIMP)

Additional Excluded Components: Development (gcc, gdb, qt, libpng, Ruby, Perl, Python, php), Office Apps (Open Office, Thunderbird, Mailman, SquirrelMail, FetchMail, SpamAssassin, Mutt), Database (mysql, postgesql), Utilities (SeaMonkey, xpdf, gpdf, kpdf, TeTeX,CUPS), Graphics (GIMP, gtk2, ImageMagick), Security (GnuPG, Wireshark, Tcpdump, OpenSSH), Web (Apache, OpenLDAP, W3C libwww, htdig, Squid), Browsers (Opera, ELinks, Konqueror), Entertainment (HelixPlayer, FLAC)

This reduced rhel4ws build is then examined for comparison:

- During the year 2007, Red Hat issued 67 Security Advisories affecting the rhel4ws reduced set of components on 43 different days during 31 different weeks. If limited to only those security advisories containing issues rated Critical or Important by Red Hat, there were 29 Security Advisories released on 26 different days during 21 different weeks.
- Red Hat fixed 160 different vulnerabilities affecting the reduced rhel4ws set of components. If limited to those rated Critical or Important by Red Hat, the number drops down to 114 vulnerabilities.

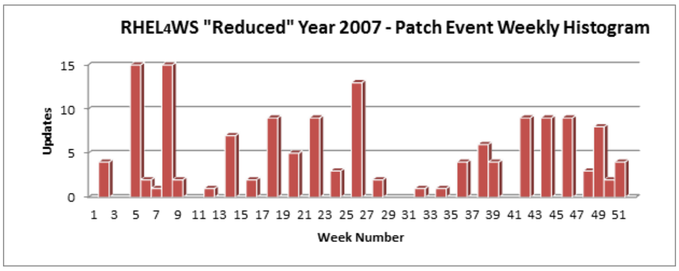

Fig. 8. RHEL4WS Patch Event Weekly Histogram (year 2007)

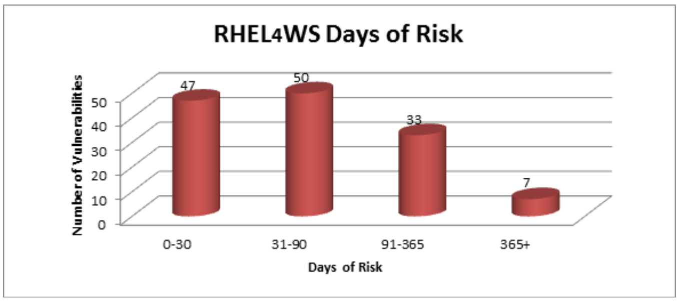

Fig. 9. RHEL4WS Days of Risk (year 2007)

B. Ubuntu 6.06 LTS

Ubuntu releases new versions every six months, and supports those releases for 18 months with daily security fixes and patches to critical bugs. The most recent version, Ubuntu 7.10 (Gutsy Gibbon), was released on 18 October 2007.

There are also Long Term Support (LTS) releases, which have three years support for the desktop version and five years for the server version. The most recent major LTS version, Ubuntu 6.06 (Dapper Drake), was released on June 1, 2006.

The next major LTS version will be 8.04 (Hardy Heron), scheduled for release in April 2008. With a full year worth vulnerability data to analyze and with Long Term Support, Ubuntu 6.06 LTS is considered for this analysis.

1) Ubuntu 6.06 LTS – Reduced Component Set

Similar to the component set reduction performed for RHEL4WS, for this analysis of Ubuntu 6.06 LTS, excluded components include those that do not have comparable functionality shipping with Windows Vista or Windows XP.

Ubuntu for example, with their advisory USN – 511 – 1 released a patch for a Kerberos Vulnerability. However, this only reduced the scope of the vulnerability, without fully solving it. Three days later, they announce and release a full fix. It is hard to fault someone who are rushing to get a patch out which at least tried to mitigate the risk their customers are facing at the earliest. Another example is if a patch breaks something such as USN – 544 -1, that fixed vulnerabilities in Samba. However, some fixes introduced regression in smbfs mounts. Ubuntu released a patch fixing this regression later the same day.

Excluded Components: Browser (w3m, Konqueror, ELinks), Office/Productivity (OpenOffice, Thunderbird, koffice, fetchmail, libwpd, Tomboy), Messaging (ksirc, Ekiga), Applications (kword, KTorrent, Inkscape, Gnome, rdesktop, raccoon, Poppler, Ghostscript, VMWare, TeTex, CUPS), Database (postgreSQL), Web (Squid, MoinMoin, NAS, snmpd, pptpd, libpng, dovecot, Apache, Jasper), Development (PHP, Python, libgd2, Qt, vim, Tk, PCRE, Perl, Mono,pwlib), Security (Enigmail, tcpdump, GnuPG, Nagios), Entertainment (XMMS, FLAC), Graphics (GIMP, ImageMagick, Cairo, GD Library), Other Libraries (t1lib)

Included Components: Browser (Firefox), Utilities (Emacs), Entertainment (xine,libsndfile), Security (Kerberos)

This reduced Ubuntu build is then examined for comparison:

- During the year 2007, Ubuntu issued 65 security advisories covering the reduced desktop build of Ubuntu 6.06. These fixes were released on 54 different days during the year 2007 in 37 different weeks
- During the year 2007, Ubuntu fixed 168 vulnerabilities affecting the reduced Ubuntu desktop set of components.

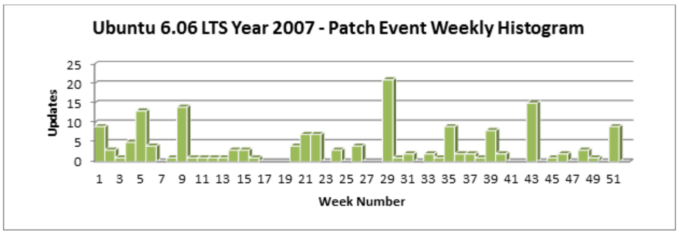

Fig. 10. Ubuntu 6.06 LTS Patch Event Weekly Histogram (year 2007)

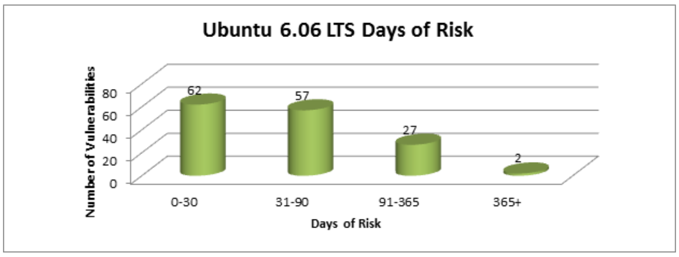

Fig. 11. Ubuntu 6.06 LTS Days of Risk (year 2007)

C. *Apple Mac OS X 10.4*

Mac OS X v10.4 "Tiger" was released on April 29, 2005. Among the new features, Tiger introduced Spotlight, Dashboard, Smart Folders, updated Mail program with Smart Mailboxes, QuickTime 7, Safari 2, Automator, VoiceOver, Core Image, and Core Video.

On January 10, 2006, Apple released the first Intel-based Macs along with the 10.4.4 update to Tiger. This operating system functioned identically on the PowerPC-based Macs and the new Intel-based machines, with the exception of the Intel release dropping support for the Classic environment.
For this analysis Mac OS X 10.4.8 and beyond is considered. This version update was released on September 29 2006 and has the full 2007 vulnerability data to analyze. Apple announced the Mac OS X 10.4.9 on March 13th 2007 and shipped the next generation of OS X, the Mac OS X v 10.5 “Leopard” on 10/26/2007 which does not have a one-year track record data for this analysis.

Apple does not have a good Security Bulletin naming system. For example, there are multiple instances of the same Security Bulletin number describing different security bulletins addressing different vulnerabilities on different products.

Excluded: Any Beta products, Server related, iTunes, Xcode, Java, Airport, CUPS, Development (Perl, Python, Ruby, XQuery), Applications/Software (Shockwave, Flash, PDFKit, VPN, VideoConference, Ftpd, OpenSSH)

Included: Safari (Browser), Multimedia (QuickTime), iChat, iPhoto

- During the year 2007, Apple released nine Security Updates (15 Security Updates if QuickTime is included) affecting Mac OS X 10.4.
- These updates fixed a total of 154 vulnerabilities in shipping components of Mac OS X 10.4

This number of vulnerabilities increases to 187 if QuickTime is included in this count of security vulnerabilities.

Below is the patch event chart for Mac OS X 10.4 for the year 2007.

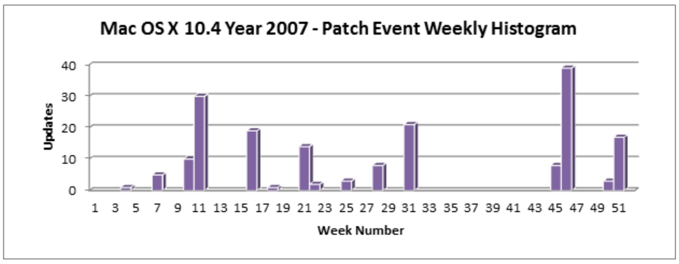

Fig. 12. Mac OS X 10.4 Patch Event Weekly Histogram (year 2007)

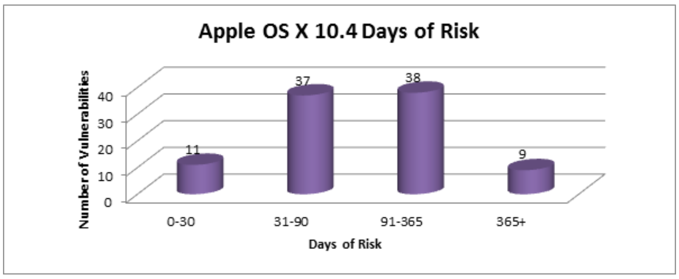

Fig. 13. Apple Mac OS X 10.4 Days of Risk (year 2007)

V. Comparison of All Operating Systems

With the basic analysis completed for Windows Vista and other industry products, we now have enough information to compare them. First, let us look at a chart that examines the impact that security updates had for administrators by looking at Patch Events.
Note: For Apple OS X, from now on, any analysis would include the numbers that include QuickTime vulnerabilities.

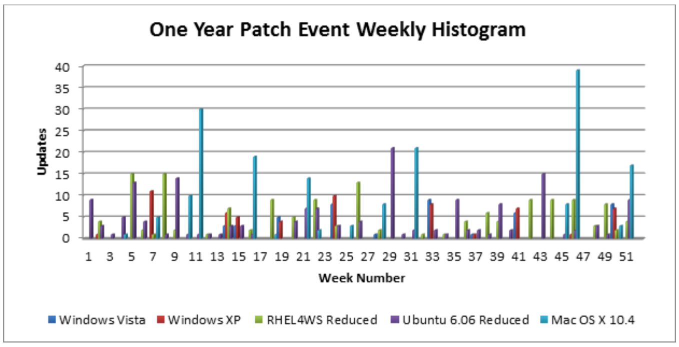

Fig. 14. Comparison of all OS on Patch Event Weekly Histogram

Table 2 below shows another view for comparison of the metrics.

TABLE II. COMPARISON OF ALL OPERATING SYSTEMS ON THEIR PATCH EVENT WEEKLY

Metric	Vista	XP	Red Hat rhel4ws	Ubuntu 6.06	OS X 10.4
Vulnerabilities Fixed	42	66	160	168	187
Security Updates	21	39	67	65	15
Patch Events	9	12	43	54	19
Weeks with at least 1 Patch Event	9	12	31	37	15

A. *Days of Risk for all Operating Systems*

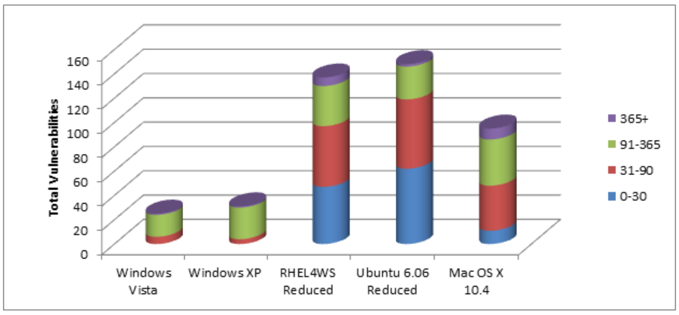

Fig. 15. Days of Risk for all Operating Systems (year 2007)

TABLE III. DAYS OF RISK FOR ALL OPERATING SYSTEMS

#Days	Vista	XP	RHEL4 WS	Ubuntu 6.06	Mac OS X 10.4
0-30			47	62	11
31-90	6	4	50	57	37
91-365	18	26	33	27	38
365+	1	1	7	2	9

VI. WEB BROWSER SECURITY

All the current stable web browsers in use today, Internet Explorer 6, Internet Explorer 7, Firefox 2.x, Safari 3.x are not fully equipped to deal with all the malware on today's Internet.

The next generation of browsers such as Internet Explorer 8 and Firefox 3 plan to do a tighter integration with anti-malware and anti-fraud mechanisms such as IE8 incorporating Windows Vista's protected mode and IE8 using a sandbox mechanism and malware blockers. Despite those moves, vulnerabilities and malicious hacker attacks that use the browser as the entry point to desktops continue to rise as indicated by the current number of browser exploits. In hacking contests to hack three different notebooks running Mac OS X, Ubuntu, and Vista, network attacks against all three failed the first day. However, when the contest was opened up to include browser exploits, Mac OS X failed in 2 minutes [17], Windows Vista running IE7 went next and then Ubuntu running Firefox all get hacked.

Following is the Web Browser Security Vulnerability data and days of Risk for the browsers for the year 2007. The browsers assessed were part of the Operating Systems assessed in this paper. They are Microsoft's Internet Explorer 6 (Windows XP), Internet Explorer 7 (Windows Vista), Firefox 2.x (Red Hat), Firefox 2.x (Ubuntu) and Safari (Mac OS X 10.4)

The data especially the average days of risk is highly accurate for Firefox (Red Hat and Ubuntu) and very close for Internet Explorer 6 & 7 (Windows XP, Windows Vista) where initial disclosure dates for over 90% of the vulnerabilities were utilized. However, for Safari, metrics and dates were only available for only 6/24 (25%) vulnerabilities. So, the 78 days of risk could be low or could go higher. This again reflects the closed loop security mentality of Apple.

Coming back to the data, the best browser for year 2007 was Mozilla Firefox on the Ubuntu operating system, which although was exposed to 56 different vulnerabilities, had the lowest average days of risk at ~75.

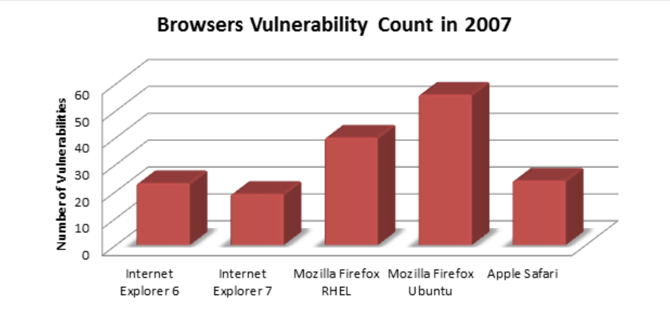

Fig. 16. Browser Vulnerability Count in 2007

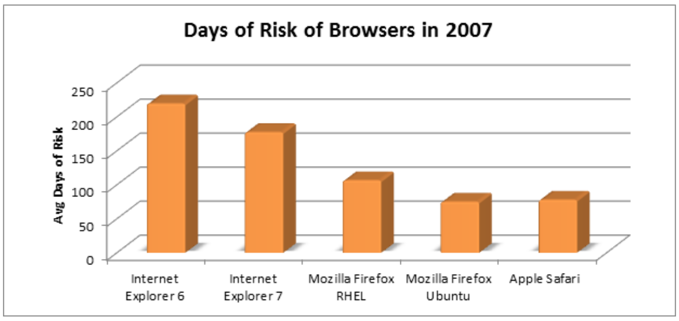

Fig. 17. Days of Risk of Browsers in 2007

TABLE IV. AVERAGE DAYS OF RISK + NUMBER OF VULNERABILITIES FOR WEB BROWSERS

Browser	Number of Vulnerabilities	Average Days of Risk
Internet Explorer 6	23	219.53
Internet Explorer 7	19	177.46
Mozilla Firefox RHEL	40	106.6
Mozilla Firefox Ubuntu	56	75.15
Apple Safari	24	78

VII. CONCLUSION

In conclusion, does the high vulnerability count indicate that the Mac OS X is the most vulnerable of all the Operating Systems evaluated? Of course not… As of this publication in 2007, there are less than 200 known viruses targeting the Mac platform compared to the many hundreds for Windows. A malicious hacker motivated by financial incentives would create malware or a computer virus that has the highest coverage possible by targeting a dominant platform such as Windows. Having control over as many hosts as possible would help a malicious hacker in launching a DDoS attacks or sending out spam or installing spyware on those machines where each click or install would pay the malicious hacker/spammer.

Recent reports [18] are indicating the increasing Mac market share and this might tip it over an infection point, which in turn would bring more active attention from malware writers. Adam J. O'Donnell, PhD, Director of Emerging Technologies at Cloudmark and has recently been using game theory to analyze at what point Macs become more targeted for malicious attack. He states,

> *"Game theory shows that an inflection point will come when the rate at which a malware author can reliably compromise a PC rivals that of the Mac market share. It is at this time you will see monetized, profitable Mac malware start popping up."*

Derek Schatz says it best when it may be possible to think about a relative security nirvana by patching your Operating System diligently, locking down the configuration and being careful with where you surf and what you trust on the Internet. For the average user, it is hard to make an OS secure but at the same time preserving usability, it does not matter whether the Operating System is Windows or Linux or Mac OS. None is measurably better over the other and they only differ in how many security researchers/ malicious hackers are paying attention to it. Sure, there are some highly secure Operating Systems such as OpenBSD or SELinux, or Trusted Solaris, but may not be compatible with the majority of desktop applications.

Therefore, Enterprises and regular users will continue to fight the never-ending cycle of patching as new flaws continue to be found in their installed base of PC's. We lose this battle a little more each month. Beyond patches and vulnerabilities, there are "softer" qualities of security that are difficult to quantify but impact deployed security. Qualities like security lifecycle support, bulletin descriptiveness, default security features, and the like all have a direct impact on deployed role security.

REFERENCES

[1] J.Jones, "Windows Vista One Year Vulnerability Report", technet.com, January 2008

[2] G. Hinson, "Seven myths about information security metrics" in ISSA Journal, July 2006.

[3] B.Schneier, "Vulnerability lifecycle chart" in Cryptogram newsletter, September 2000

[4] J. Jones, "Basic guide to days of risk" in CSO Online blog, July 2007

[5] Microsoft TechNet security listings <:http://www.microsoft.com/technet/security/current.aspx>

[6] RedHat Security advisories: https://rhn.redhat.com/errata/rhel4ws-errata-security.html

[7] Ubuntu Security: http://www.ubuntu.com/usn

[8] Apple Security: http://docs.info.apple.com/article.html?artnum=305391

[9] Firefox vulnerabilities: http://www.mozilla.org/projects/security/known-vulnerabilities.html#Firefox

[10] Trapkit security advisories <http://www.trapkit.de/advisories>

[11] Zero day initiative <http://www.zerodayinitiative.com/advisories/>

[12] Securiteam CVE listings <http://www.securiteam.com/cves/2006/>

[13] iDefense vulnerability listing report for year 2007 <http://labs.idefense.com/intelligence/vulnerabilities/?intYear=2007>

[14] eeye research advisory listing and publications <http://research.eeye.com/html/advisories/published/index.html>

[15] Secunia research <http://secunia.com/secunia_research/>

[16] ISS CVE listing https://webapp.iss.net/Search.do?keyword=CVE-2007-3826&searchType=keywd&x=13&y=8

[17] ITWorld "Mac Hacked first in content", March 2008 http://security.itworld.com/5013/mac-hacked-first-in-contest-080327/page_1.html

[18] Bloginfosec posting "Are we less secure now that before", March 2008 <http://www.bloginfosec.com/2008/03/18/are-we-less-secure-now-than-before/>