Vulnerability Scorecard Report

Published: March 2011

Vulnerability Scorecard Report

Contents

Executive Summary

This report benchmarks how security vulnerabilities in Microsoft products compare to those in select competitors' products. The platforms covered include Internet browsers, client operating systems, server operating systems, and database platforms. This study leverages data from the National Vulnerability Database (NVD), the industry standard source of security vulnerability data. It includes comparative analysis tables and charts that allow the reader to form their own conclusions about the existence and distribution of vulnerabilities in common software platforms.

This report analyzes the following platforms:

- Web browsers by the following software companies:
  - Google Corporation (Chrome)
  - Mozilla (Firefox)
  - Microsoft® Corporation (Windows® Internet Explorer®)
  - Apple Corporation (Safari)
- Client operating systems by the following software companies:
  - Apple Corporation (MAC OS X)
  - Ubuntu Linux
  - Microsoft Corporation (Windows®)
- Server operating systems by the following software companies:
  - Red Hat (Enterprise Linux)
  - Microsoft Corporation (Windows)
- Database programs by the following software companies:
  - Microsoft Corporation (SQL Server®)
  - MySQL (now owned by Oracle)
  - Oracle (Database 11g)

Reporting Period

Throughout this report, half-yearly and quarterly time periods are referenced using the nHyy or nQyy formats, respectively, where yy indicates the calendar year and n indicates the half or quarter. For example, 1H10 represents the first half of 2010 (January 1 through June 30), and 2Q10 represents the second quarter of 2010 (April 1 through June 30). To avoid confusion, note the referenced reporting period or periods when considering the report's statistics. All of the data in the NVD version 2.0 data files that were published in 2010 was included in the analysis. The NVD data files that were used in this report were downloaded from the NVD on January 1, 2011.

Methodology

The methodology used in this report was to define four collections of features and approximately similar functionality that address common user expectations in such a way that the features and functionality are sufficiently similar for comparison purposes.

It is a complex challenge to provide a balanced comparison across different platforms. However, the research team made every effort to assure that the evaluated platforms had comparable functionality. The analyses started with default installations and settings, and adjusted them to bring all of the evaluated platforms to the same level of functionality. For example, open source products typically create new releases to fix vulnerabilities, and operating systems from Microsoft and Macintosh provide patches throughout the lifetime of the operating systems. Also, creating a fair comparison between the evaluated platforms in a group requires some adjustment to the Common Platform Enumeration (CPE) definitions, because these definitions are defined in the NVD differently. In addition, "published date" information entered into the NVD may not be accurate.

Some feature exceptions were noted in both Macintosh and Windows installations, such as CUPS and Microsoft Direct-X®, which ensure that the operating environments work in a normal or standard mode. It's noteworthy that the operating systems can operate without these capabilities, but would not reflect a reasonably good user experience.

Additional detail is available in the appendix on the data sources used and rationale for the choice made in this analysis.

All of the raw data was pulled directly from the NVD – specifically, the 2009/2010 version 2.0 files. In addition, vulnerability data such as OVAL and the CERT were also analyzed to compare findings and determine accuracy of the vulnerability profile; Note: Only the raw Common Vulnerabilities and Exposures (CVE) database was complete. All other government reports were summaries of the CVE data.

Web Browsers

This section of the Vulnerability Scorecard Report focuses on the web browsers that are listed in the following table.

Table 1. Vulnerability Scorecard Report Web Browsers

Web browser	Release date
Chrome 5 Chrome 6 Chrome 7 Chrome 8	2010-05-21 2010-09-02 2010-10-21 2010-12-2
Firefox 3.6	2010-01-21
Windows Internet Explorer® 7 Windows Internet Explorer® 8	2006-10-01 2009-03-18
Safari 4 Safari 5	2008-06-02 2010-06-07

Browser Trends

The following figure displays the 2010 vulnerability trends for the web browsers in this report.

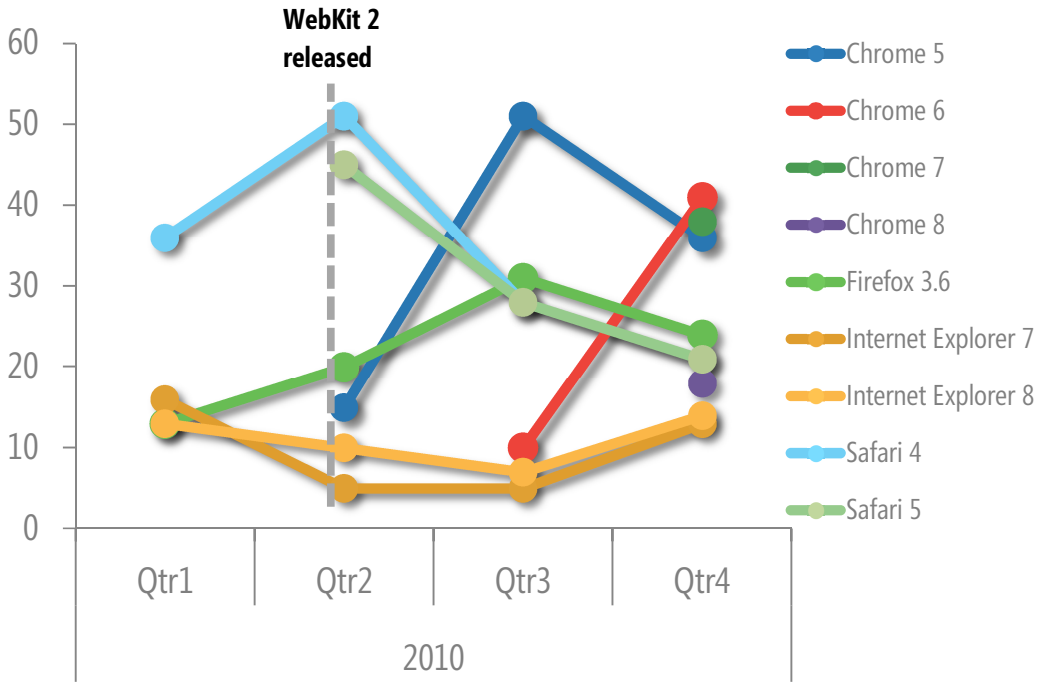

Figure 1. Web browser trends

The impact of the release of WebKit 2 (an open source web browser engine) in the second quarter on the increased number of vulnerabilities for Chrome and Safari is described below. Subsequent charts and analysis describe in more detail the events that contributed to these trends.

Table 2. Web Browser Unique Vulnerability Totals Published by the NVD in 2010

2010	Chrome 5	Chrome 6	Chrome 7	Chrome 8	Firefox 3.6	Internet Explorer 7	Internet Explorer 8	Safari 4	Safari 5
1Q10					13	16	13	36	
2Q10	15				20	5	10	51	45
3Q10	41	10			31	5	7	28	28
4Q10	36	41	38	18	24	13	14	21	21
Total	102	51	38	18	88	39	44	136	94

The preceding table includes the vulnerabilities that were reported by the NVD database according to the "published date" in 2010 of each vulnerability in the database. This is the data used to create figure 1. The browser release dates in Table 1 show that only some of the browsers considered were on the market at the start of the year. Vulnerabilities for others, such as Safari 5 and all of the Chrome releases, were only published after their formal release date.

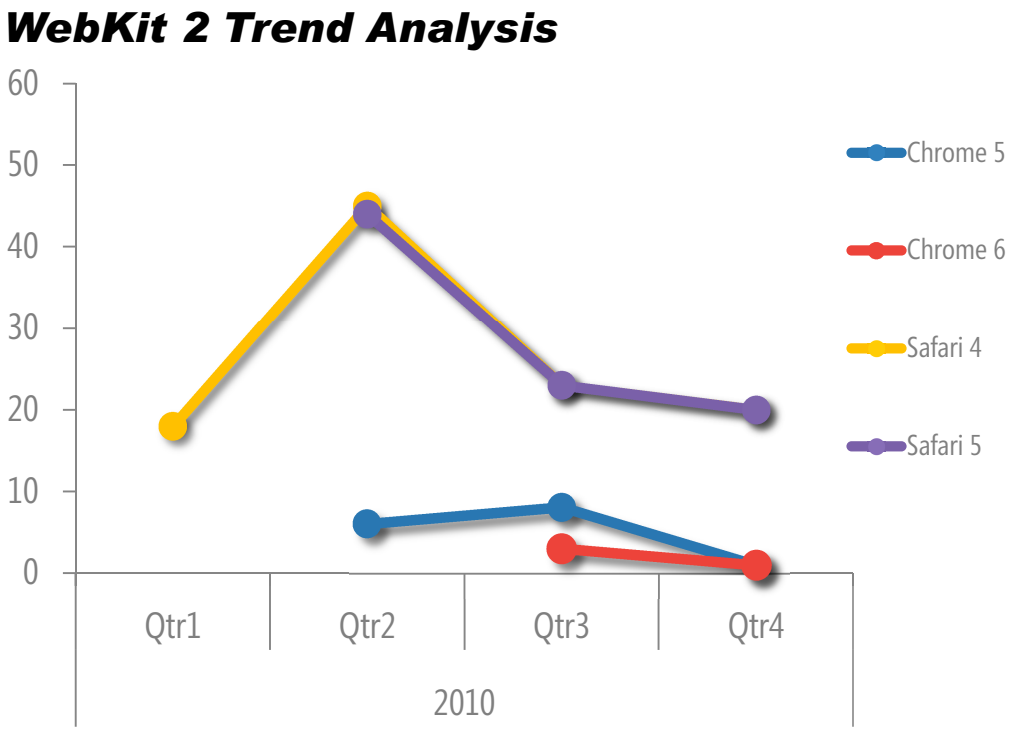

Figure 2. WebKit2 vulnerabilities

Table 3. WebKit2 Unique Vulnerability Totals Published by the NVD in 2010

2010	Chrome 5	Chrome 6	Chrome 7	Chrome 8	Safari 4	Safari 5
1Q10					18	
2Q10	6				45	44
3Q10	8	3			23	23
4Q10	1	1	0	0	20	20
Total	15	4	0	0	106	87

WebKit 2 vulnerability trend observations:

- The WebKit 2 open source browser engine core update was released on April 8, 2010. This complete redesign was created to support a separate isolated Java process for web content and several other enhancements. Browsers such as Chrome and Safari were affected by several vulnerabilities.
- The vulnerabilities reported for Safari appear to apply to all versions of these browsers that were released during the 1Q10 and 4Q10 periods. The vulnerabilities reported for Chrome apply to versions 5 and 6, and appear to have been corrected in Chrome versions 7 and 8. For more information, see Figure 4 later in this section that displays CVEs published for Chrome during 4Q10 of 2010.

Google Chrome Trend Analysis

The following figure displays 2Q10 through 4Q10 vulnerability trends for the Chrome browsers in this report.

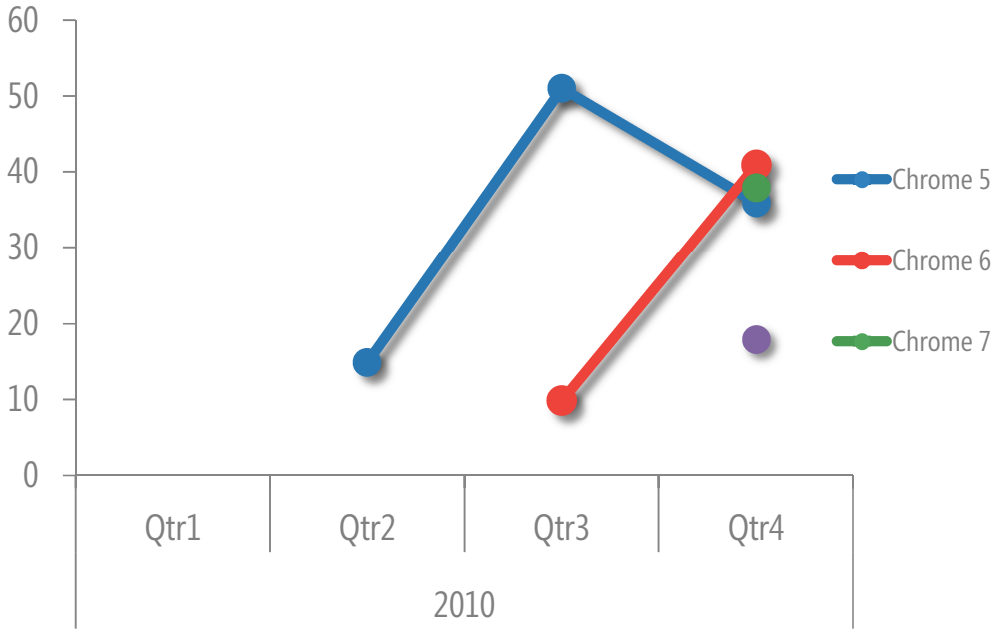

Figure 3. Google Chrome trends

The following table includes the unique vulnerability totals for Chrome that were published by the NVD in 2010.

Table 4. Chrome Unique Vulnerability Totals

2010	Chrome 5	Chrome 6	Chrome 7	Chrome 8
1Q10				
2Q10	15			
3Q10	51	10		
4Q10	36	41	38	18
Totals	**102**	**51**	**38**	**18**

The rapid release of Chrome browsers included three version refreshes in less than four months, which could indicate that security corrections in new releases were included along with software updates.

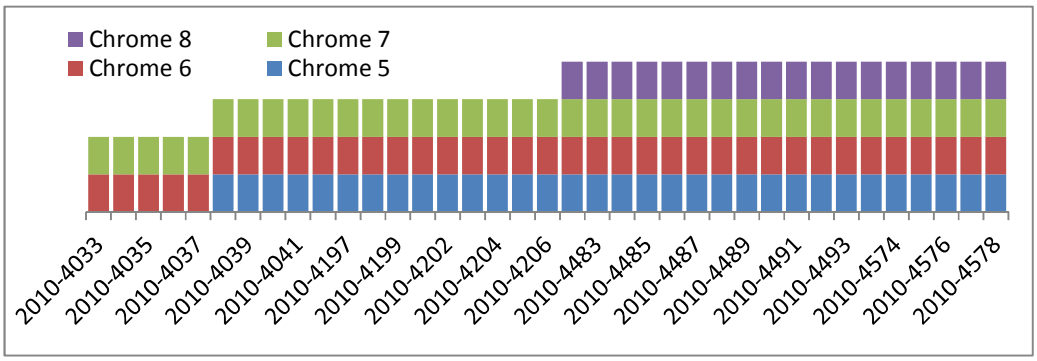

Figure 4. CVEs published for Chrome during 4Q10 of 2010

The CVEs in Figure 4 show that Chrome 5, 6, 7, and 8 were affected by CVE-2010-4083, CVE-2010-4085, CVE-2010-4087, and several others after they were publicly released. These five vulnerabilities allow attackers to potentially use a malicious image or document to change data or deny user access.

Firefox Trend Analysis

The following figure displays 1Q10 through 4Q10 vulnerability trends for Firefox.

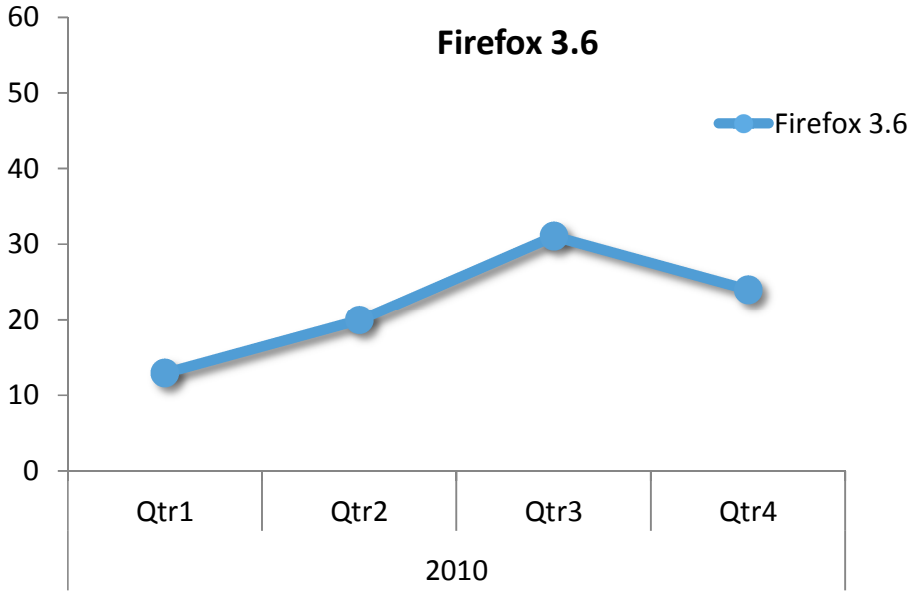

Figure 5. Firefox 3.6 trends

The following table includes the unique vulnerability totals for Firefox 3.6 that were published by the NVD in 2010.

Table 5. Firefox 3.6 Vulnerability Totals in 2010

2010	Firefox 3.6
1Q10	13
2Q10	20
3Q10	31
4Q10	24
Total	**88**

Firefox vulnerability observations:

- A number of vulnerabilities in Firefox published in 2H10 had the potential to execute arbitrary code or allow elevation of privilege in the Document Object Model (DOM), including the following vulnerabilities: CVE-2010-1208, 1209, 1211, 1212, 1214, 1215, 1988, 2752, 2753, 2755, 2760, 2766, 2767, 2770, 3131, 3166, 3167, 3168, and 3169. Most of these CVEs were assigned update numbers after the release of a security update, so the corresponding vulnerabilities were mitigated before they had been verified.
- Like many other open-source projects, Mozilla does not release updates for current versions. Instead, Mozilla issues new versions of the browser with updates that add new functionality and try to correct existing vulnerabilities.

Internet Explorer Trend Analysis

The following figure displays 1Q10 through 4Q10 vulnerability trends for Internet Explorer.

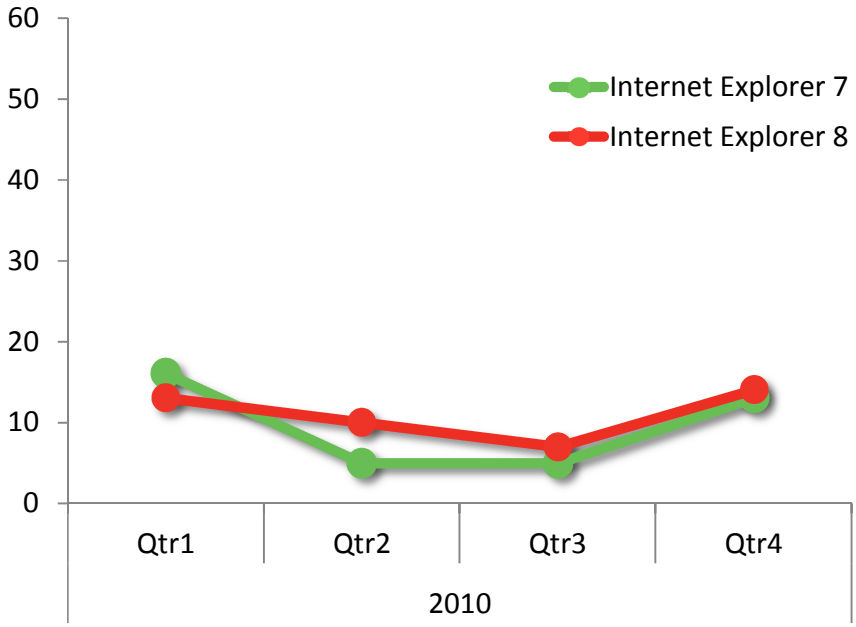

Figure 6. Trend results for Internet Explorer

The following table includes the unique vulnerability totals for Internet Explorer that were published by the NVD in 2010.

Table 6. Internet Explorer Vulnerability Totals in 2010

2010	Internet Explorer 7	Internet Explorer 8
1Q10	16	13
2Q10	5	10
3Q10	5	7
4Q10	13	14
Total	**39**	**44**

Internet Explorer vulnerability observations:

- Two vulnerabilities were reported at the end of the year that could result in the execution of arbitrary code in the Cascading Style Sheets (CSS) parser in mshtml.dll, CVE-2010-3971 and CVE-2010-3346. A similar vulnerability was reported for Internet Explorer 8, CVE-2010-3345, which results in a memory handling issue.
- In 4Q10 (on October 12, specifically), 10 vulnerabilities were disclosed and reported in Microsoft Security Bulletin MS10-071. The security update for this bulletin addressed the vulnerabilities by modifying the way Internet Explorer handles objects in memory, CSS special characters, HTML sanitization, the AutoComplete feature, the Anchor element, and scripts during certain processes.

Safari Trend Analysis

The following figure displays 1Q10 through 4Q10 vulnerability trends for Safari.

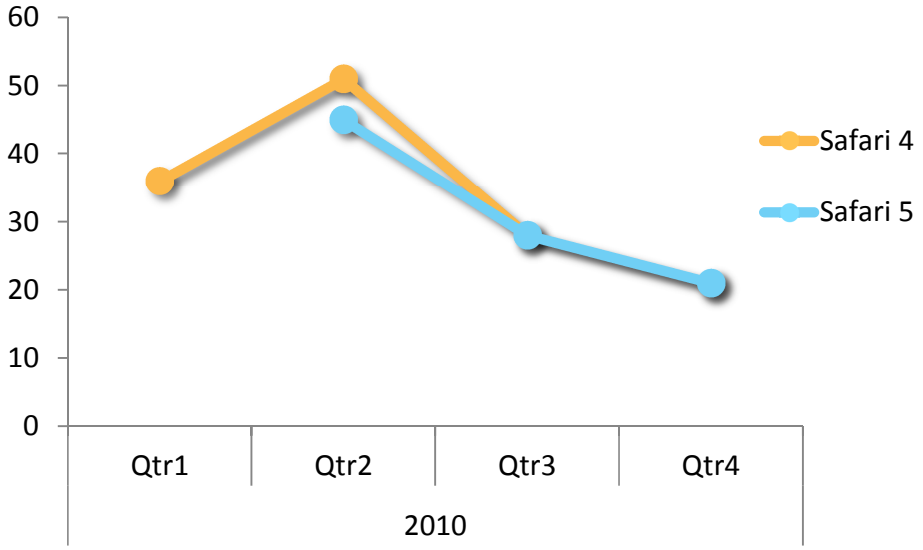

Figure 7. Trend analysis results for Safari

The following table includes the unique vulnerability totals for Safari that were published by the NVD in 2010.

Table 7. Safari Vulnerability Totals in 2010

2010	Safari 4	Safari 5
1Q10	36	
2Q10	51	45
3Q10	28	28
4Q10	21	21
Total	**136**	**94**

Safari vulnerability trend observations:

- Safari had the greatest number of identified vulnerabilities as published by the NVD for the evaluated browser platforms in this report.
- Six vulnerabilities (CVE-2010-1176, 1177, 1178, 1179, 1180, and 1181, which relate to execution of arbitrary code and denial of service) that were published on Mar. 29, 2010, were confirmed on products such as the iPod touch and the iPhone that use iOS, the Apple mobile operating system. These vulnerabilities are included in the count because they represent the new generation of computing applications for Apple and use Safari as their browser.

Client Operating Systems

This section of the Vulnerability Scorecard Report focuses on the client operating systems in the following table.

Table 8. Vulnerability Scorecard Report Client Operating Systems

Client operating system	Release date
MAC OS X 10.6 (Snow Leopard)	2009-08-28
Ubuntu 10.04 (Linux 2.6.32)	2010-04-28
Windows 7	2009-10-22

Trend Analysis

The following figure displays 2010 vulnerability trends for the client operating systems in this report.

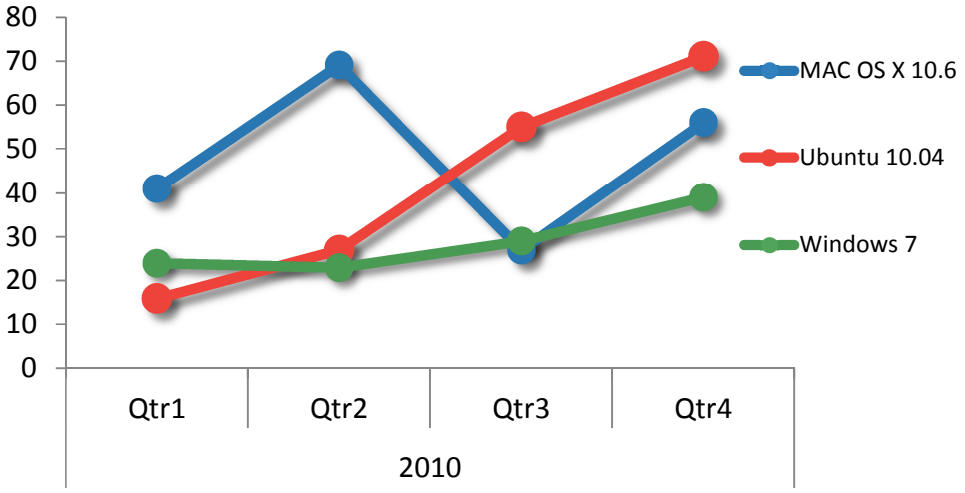

Figure 8. Trend analysis results for client operating systems

The following table includes the unique vulnerability totals for client operating systems in this report that were published by the NVD in 2010.

Table 9. Client Operating System Totals of Unique Vulnerabilities

2010	MAC OS X 10.6	Ubuntu 10.04	Windows 7
1Q10	36	16	24
2Q10	25	27	23
3Q10	30	55	29
4Q10	58	71	39
Total	**149**	**169**	**115**

The end of 2010 included several vulnerability postings by security firms on different submission schedules that affected the rate of their inclusion into the NVD.

Macintosh Operating Systems Evaluated Platform

The following default features are included in each Macintosh client OS X release:

- Airport utilities: provide wireless services for OS X.
- Apple type services: the Unicode imaging services for typeface and text in OS X.
- Core graphics: the 2D and 3D rendering engine.
- The Common UNIX Printing System (CUPS), as distributed by Apple: provides printing services.
- ImageIO, as distributed with Macintosh: provides image file format processing and compatibility.
- iTunes, as distributed with Macintosh: provides music and sound services.
- QuickTime, as distributed with Macintosh: provides the video processing services for OSX.
- Safari browser version 5 for MAC OS X 10.6.

All relevant Safari vulnerabilities from the NVD were included in the Macintosh vulnerability count.

The following vulnerability observations were noted:

- An increased number of vulnerabilities reported for Mac OS X 10.6 (Snow Leopard) since the release of this operating system on August 28, 2009 included vulnerabilities to key features, such as the iTunes and graphics subsystems for OSx and CVE-2010-1777, which is a remote buffer overflow vulnerability in iTunes.
- The following QuickTime vulnerabilities were also published for Snow Leopard: CVE-2010-4009, CVE-2010-3800, CVE-2010-3801, and CVE-2010-3802. These vulnerabilities allowed attackers to execute arbitrary code and cause remote denial-of-service attacks via a crafted movie file.
- The large spike of vulnerabilities in Mac OS X 10.6 in 1Q10 was created by APPLE-SA-2010-03-29-1 Security Update 2010-002.

Ubuntu Client Operating Systems Evaluated Platform

Refer to Appendix A, “Data Sources,” for all services used in the evaluation of the Ubuntu distributions. Ubuntu 10.04 was chosen for the evaluation instead of Ubuntu 10.10 because Ubuntu 10.04 is a Long Term Service (LTS) release, which means it is a fully supported version. Ubuntu 10.10 was only released in early 2010. From release and stability perspectives, Ubuntu 10.4 is more similar to the other evaluated operating systems in this report.

In addition to the default Ubuntu components included in the Ubuntu client operating system that was evaluated for this report, the following major features are included:

- CUPS – Common Unix Printing System
- Firefox
- Linux kernel 2.6.32
- Perl scripting
- FTP – File transfer Protocol
- OpenSSL – cryptographic modules

The following vulnerability observations were noted:

- The increasing number of Ubuntu vulnerabilities are a result of the vulnerabilities in Firefox (see the “Firefox Trend” section earlier in this report)
- The Linux 2.6.32 kernel included several elevation of privilege and denial-of-service vulnerabilities, such as CVE-2010-3850 and CVE-2010-4342.

Windows Client Operating Systems Evaluated Platform

In addition to the default Windows components included in the Windows client operating system releases in this report, the following features are included:

- Microsoft .NET Framework (2.0 for Windows Vista® and 3.5 for Windows 7).
- Microsoft DirectX versions 7, 8, and 9.
- Internet Explorer 7 is a part of Windows Vista, and Internet Explorer 8 is a part of Windows 7. Vulnerabilities published against Internet Explorer in the NVD are included in the count for the corresponding version of the Windows client operating system.
- Microsoft Silverlight® versions 2 and 3.
- Windows Media® Player 10 and other versions.
- Windows Media encoder, foundation, and format runtime.
- Microsoft XML Core Services (MSXML), a set of services that allow applications written in JScript®, Visual Basic® Scripting Edition (VBScript), and Microsoft development tools to run Windows-native XML-based applications.

The following vulnerability observations were noted:

- CVE-2010-3966 is a vulnerability to the BranchCache feature that allows local users to gain privileges via a Trojan horse.
- CVE-2010-3940 is a vulnerability in win32k.sys in the kernel-mode.
- CVE-2010-3965 is a vulnerability in the search path in Media Encoder 9.

Server Operating Systems

This section of the Vulnerability Scorecard Report focuses on the server operating systems that are listed in the following table.

Table 10. Vulnerability Scorecard Report Server Operating Systems

Server operating system	Release date
Red Hat Enterprise Linux (RHEL) V5 (Linux 2.6.18) Update 5.5	2007-03-14 2010-03-30
Windows Server® 2008	2008-02-27
Windows Server 2008 R2	2009-10-22

Trend Analysis

The following figure displays the 2010 vulnerability trends for the server operating systems in this report.

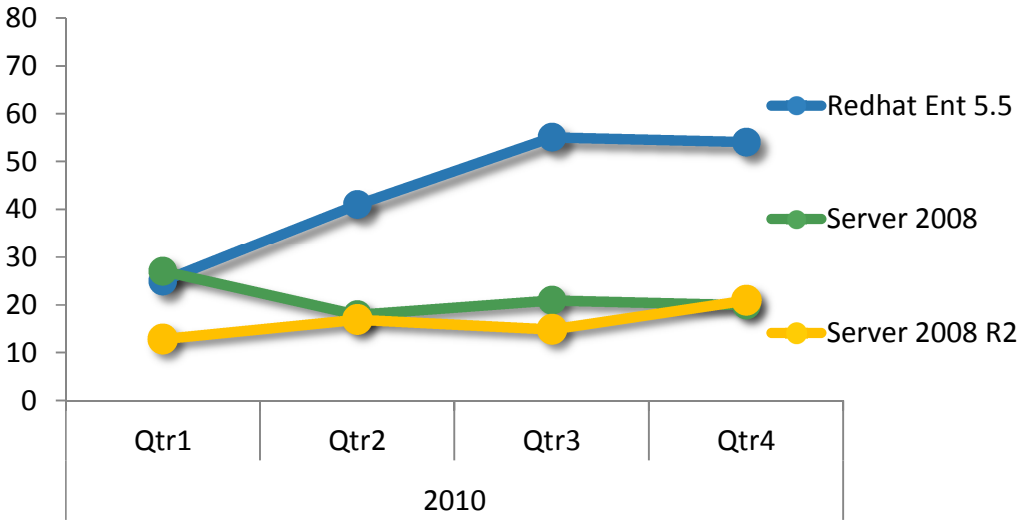

Figure 9. Trend analysis results for server operating systems

The following table includes the unique vulnerability totals for server operating systems in this report that were published by the NVD in 2010.

Table 11. Server Operating System Totals of Unique Vulnerabilities

2010	Red Hat Enterprise 5.5	Windows Server 2008	Windows Server 2008 R2
1Q10	25	27	13
2Q10	41	18	17
3Q10	55	21	15
4Q10	54	20	21
Total	**175**	**86**	**66**

Red Hat Linux Operating Systems Evaluated Platform

For Linux distributions, the following components are included in the evaluation. See Appendix A, “Data Sources,” for details. For more information about the Red Hat Errata page, see Appendix A, “Cross-Platform Vulnerability Complexity.” This list includes:

- Perl components
- Linux Kernel 2.6.18
- CUPS. Common Unix Printing System
- FTP. File Transfer Protocol
- Bash. The default shell
- Iputils. The basic networking applets (ping, traceroute, and so on)

The following vulnerability observations were noted:

- Red Hat Enterprise Linux 5.5 exploits increased from 2Q10 to 4Q10 with integer and buffer overflow vulnerabilities.
- CVE-2010-3848 is a stack-based buffer overflow in the econet_sendmsg function in net/econet/af_econet.c.
- CVE-2010-3874 is a heap-based buffer overflow in the bcm_connect function in net/can/bcm.c in the Controller Area Network (CAN) implementation in the Linux kernel. This vulnerability might allow local users to cause a denial of service (memory corruption) via a connect operation.
- In 4Q10, 13 Firefox vulnerabilities were published by the NVD against Red Hat Enterprise Linux 5.5 for the first time. However, this analysis did not include these browser vulnerabilities for this evaluated platform. As a result, the count for 4Q10 remained about the same as the count for 3Q10.

Windows Server Operating Systems Evaluated Platform

The Windows Server vulnerability counts include the following installation modules:

- The Windows NT 6.1 release.
- Windows PowerShell® for scripting.
- Microsoft .NET 3.5.
- Graphics subsystem (win32k.sys and GDI+).
- HTTP/TCP/IP stack.
- Remote file system, include branch support (SMB).
- Printing server.
- Active Directory® Domain Services for authentication.
- Hyper-V™ virtualization.
- CNP – Crypto Next Generation as well as CAPI.
- WMI administrative tools.

The following vulnerability observations were noted:

- CVE-2010-2739 is a buffer overflow in the win32k.sys function.
- Some buffer overflows discovered in Windows Server 2008 did not affect Windows Server 2008 R2, such as those published in CVE-2010-2739 and CVE-2010-3227.
- The vulnerabilities discovered in Windows Server 2008 R2, but not Windows Server 2008, could only be exploited by an application installed on the server. For example CVE-2010-3944 is a win32k.sys vulnerability in the kernel-mode drivers in Microsoft Windows Server 2008 R2.

Database Systems

This section of the Vulnerability Scorecard Report focuses on the database systems in the following table.

Table 12. Vulnerability Scorecard Report Database Systems

Database	Release date
Microsoft SQL Server® 2008	2008-8-06
Microsoft SQL Server 2008 R2	2010-4-21
MySQL 5	2004-12-01
Oracle Database 11g	Uncertain, see the following “Oracle Database Evaluated Platform” section.

Trend Analysis

The following figure displays the 2010 vulnerability trends for the database systems in this report.

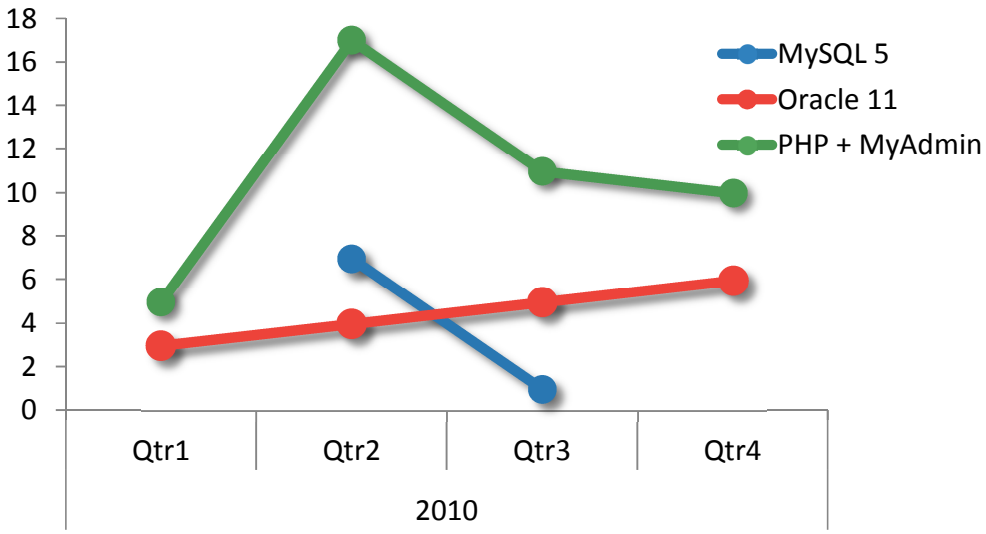

Figure 10. Trend analysis results for database systems

The following table includes the unique vulnerability totals for database systems in this report that were published by the NVD in 2010.

Table 13. Database System Totals of Unique Vulnerabilities

2010	SQL Server 2008	SQL Server 2008 R2	MySQL 5	Oracle 11	PHP + MyAdmin
1Q10	0	0	0	3	5
2Q10	0	0	7	4	17
3Q10	0	0	1	5	11
4Q10	0	0	0	6	10
Total	**0**	**0**	**8**	**18**	**43**

Microsoft SQL Server Evaluated Platform

The Microsoft SQL Server platforms are the public release versions of SQL Server 2008 and SQL Server 2008 R2.

- No vulnerabilities were reported for SQL Server 2008 and SQL Server 2008 R2 since Q4 of 2009, when some GDI+ vulnerabilities were published by the NVD. NVD vulnerabilities for 2009 are not included in this report.

MySQL Evaluated Platform

Evaluating MySQL 5 required some open source components, such as Perl, which had very few vulnerabilities published in 2010. In addition, the management of MySQL uses PHP for scripting and MyAdmin tool set for web administration. These items are optional, but highly desirable since the lack of the tools makes management of MySQL more complex. Because similar functionality is available from Microsoft SQL Server and Oracle, the vulnerabilities for PHP and PHP + MyAdmin are included in the analysis but listed separately from MySQL.

The following vulnerability observations were noted:

- CVE-2010-1158 is a PERL Integer overflow in the regular expression engine that will allow attackers to cause a denial of service in the PERL engine. Red Hat has indicated that this vulnerability will not be patched because the impact of the correction will introduce compatibility issues.
- CVE-2010-1621 affects the mysql_uninstall_plugin function in sql/sql_plugin.cc in MySQL. This vulnerability exposes a privileges check that can allow remote attackers to uninstall arbitrary plugins.
- CVE-2010-1849 allows attackers to cause a denial of service in MySQL in the my_net_skip_rest function.

Oracle Database Evaluated Platform

The release date for the Oracle Platforms is indeterminate because Oracle defines versions by functionality rather than by release data. For some time, vulnerabilities specific to Oracle 11g were reported separately. However, recent vulnerabilities reported to the NVD have all been labeled simply Oracle 11.

The following vulnerability observations were noted:

- Oracle vulnerabilities that were published as a result of Oracle security updates on the following dates: January 12 (3), April 13 (4), July 13 (5) and October 13 (6).
- Oracle vulnerabilities that were published in October (CVE-2010-2387, 2407, 2411, 2412, and 2415), which were described as an unspecified threat to the confidentiality and integrity of the database.
- CVE-2010-2419, a Java Virtual Machine vulnerability that allows a remote authenticated user to access the database.

Summary

The Vulnerability Scorecard Report provides information about vulnerability counts for the calendar year 2010 in comparable evaluated platforms (collections of features and functionality) that are of general interest to the IT community. The latest National Vulnerability Database (NVD) data was used at the end of the fourth quarter of 2010 (4Q10) as the source of data in this report. This report includes comparative analysis information on leading open source and proprietary client and server operating systems, web browsers, and database operating systems. Future versions of this report will include comparative analysis information on leading open source, Apple, and Microsoft applications.

Appendix A: Data Sources

This appendix of the Vulnerability Scorecard Report provides terms and definitions from the Microsoft Security Intelligence Report (SIR), a glossary of terms derived from the NVD, and information about the NVD Vulnerability Severity Ratings scale used in this report.

Vulnerabilities

Vulnerabilities are weaknesses in software that allow an attacker to compromise the integrity, availability, or confidentiality of that software. Some of the worst vulnerabilities allow attackers to run arbitrary code on the compromised system.

The SIR analyzes new vulnerabilities that are disclosed and has been examining trends in vulnerability disclosures since 2006. A disclosure, as the term is used in the SIR, is the revelation of a software vulnerability to the public at large. It does not refer to any sort of private disclosure or disclosure to a limited number of people. Disclosures can come from a variety of sources, including the software vendor itself, security software vendors, independent security researchers, and even malware creators.

Vulnerability Disclosure Date vs. Publication Date

Vulnerabilities are counted and charted for trends based upon the date when the vulnerability was first disclosed.

Another key date that is associated with each vulnerability is its publication date, which is the date the vulnerability is first assigned a Common Vulnerabilities and Exposures (CVE) identifier and published in the Mitre CVE list or the NIST National Vulnerability Database. Typically, the length of time between the publication and disclosure dates is relatively short and has little impact on the trend analysis. For example, from 2005 through the end of 2007, less than five percent of the vulnerabilities disclosed in each half-year period were published more than 30 days beyond the end of the period.

However, there have been times when a significant percentage of vulnerabilities disclosed during a period were not published until the following year—enough to have a noticeable effect on the reported disclosure trend.

Vulnerability Severity

In general, large numbers of disclosed vulnerabilities create significant challenges for IT security administrators who have deployed the affected products. Not all vulnerabilities are equal, however, and an analysis of vulnerability severity can help IT professionals understand and prioritize the nature and severity of the threats they face from newly disclosed vulnerabilities.

The Common Vulnerability Scoring System (CVSS) is a standardized, platform-independent scoring system for rating IT vulnerabilities, developed by a coalition of security professionals from around the world who represent the commercial, non-commercial, and academic sectors. Currently in its second version, the system assigns a numeric value between 0 and 10 to vulnerabilities according to severity, with higher scores representing greater severity. See “NVD Vulnerability Severity Ratings” later in this appendix for information that defines the high, medium, and low ranges within this system that are used in this report.

Vulnerability Complexity

Some vulnerabilities are easier to exploit than others, and vulnerability complexity is an important factor to consider in determining the magnitude of the threat that a vulnerability poses. A High severity vulnerability that can only be exploited under very specific and rare circumstances might require less immediate attention than a lower severity vulnerability that can be exploited more easily.

Security investigators consider both severity and complexity when determining the appropriate response to a vulnerability. Access complexity is one of the metrics used to calculate the CVSS base score for a vulnerability. CVSS version 2.0 uses three complexity designations: Low, Medium, and High. These definitions are from section 2.1.2 of Peter Mell, Karen Scarfone, and Sasha Romanosky's, "A Complete Guide to the Common Vulnerability Scoring System Version 2.0."

Vulnerability Scorecard Report Glossary

This glossary of terms for the Vulnerability Scorecard Report is mostly taken from Appendix A, "SCAP Resources" in the *System Center Configuration Manager Extensions for SCAP User Guide,* which was released by the Microsoft SA-SC team as a Solution Accelerator in July 2009. Supplementary information has been added from the NVD website.

Table A1. Vulnerability Scorecard Report Glossary

Term	Definition
NVD	National Vulnerability Database (NVD). The NVD contains data feeds for each standard that the security community can use without licensing. The NVD and the associated National Checklist Program also contain SCAP security checklist data that organizations can use with SCAP compatible tools to automate vulnerability management and compliance activities.
NIST	The U.S. National Institute of Standards and Technology defines and maintains SCAP and the FDCC SCAP data streams for SCAP standards. NIST defines how to use the open standards within SCAP context and the mappings between the SCAP component enumeration standards. However, NIST does not control the underlying standards that are used within the protocol.
SCAP	Security Content Automation Protocol (SCAP) that provides a method to use specific existing standards to enable automated vulnerability management, measurement, and policy compliance evaluation.
CVE	Common Vulnerabilities and Exposures (CVE), a component enumeration standard of SCAP.
CCE	Common Configuration Enumeration (CCE), a component enumeration standard of SCAP.
CPE	Common Platform Enumeration (CPE), a component enumeration standard of SCAP that is a structured naming scheme for information technology systems, software, and packages.
CVSS	Common Vulnerability Scoring System (CVSS), a component enumeration standard of SCAP. The standard impact score (CVSS) is derived from a standard product name (CPE), the configuration issues that affected that product (CCE), and from a given software vulnerability (CVE) or configuration issue (CCE).

Term	Definition
XCCDF	Extensible Configuration Checklist Description Format (XCCDF), a component enumeration standard of SCAP. XCCDF is a machine-readable language format that is used to write SCAP security checklist data that consists of configuration checklists.
OVAL	Open Vulnerability and Assessment Language (OVAL), a component enumeration standard of SCAP.
FDCC	Federal Desktop Core Configuration (FDCC)

NVD Vulnerability Severity Ratings

The NVD provides severity rankings of **Low**, **Medium**, and **High** in addition to the numeric CVSS scores. These qualitative rankings are mapped based on the following numeric CVSS score ranges:

- **Low** severity vulnerabilities have a CVSS base score ranging from 0.0 to 3.9.
- **Medium** severity vulnerabilities have a CVSS base score ranging from 4.0 to 6.9.
- **High** severity vulnerabilities have a CVSS base score ranging from 7.0 to 10.0.

Red Hat Errata Advisories - A Sample of Open Source Program Methodology

This section includes information about the Red Hat Enterprise Linux Life Cycle and Red Hat Errata Advisories. Red Hat supports major releases of Red Hat Enterprise Linux (RHEL) during stated time periods. Each major release of RHEL is denoted by a single number, for example, RHEL 4, RHEL 5, RHEL 6. The RHEL Life Cycle identifies the various levels of maintenance for each major release of RHEL over a period of up to 10 years from the initial release date, which is often referred to as the general availability (GA) date.

Red Hat published the RHEL Life Cycle in an effort to provide as much transparency as possible and may make exceptions from these policies as conflicts may arise.

Software changes to RHEL are delivered via individual updates known as *Errata Advisories*, generally through Red Hat Network (RHN) or other authorized portals. Errata Advisories can be released individually on an as-needed basis or aggregated as a Service Pack (or SP, also referred to as a *Minor Release* or *Update*). Errata Advisories may contain security, bug fixes or feature enhancements. Red Hat may release any Errata Advisories independent of customer reported issues and will generally do so for critical impact security issues. All Errata Advisories are tested and qualified against the appropriate Red Hat Enterprise Linux release. All released Errata Advisories remain accessible to active subscribers for the entire Life Cycle. Within each major release of Red Hat Enterprise Linux, any Errata Advisory is only applied incrementally to the previously released Errata Advisories.

Red Hat makes commercially reasonable efforts to maintain binary compatibility for the core runtime environment across all Service Packs and asynchronous errata advisories. Red Hat may elect to make exceptions to the compatibility goal for critical impact security or other significant issues. In addition, major releases of Red Hat Enterprise Linux contain a limited set of backward-compatible libraries from the previous major releases to allow for the easy migration of applications. Exceptions may apply for controlled re-bases, packages that are provided primarily to satisfy dependencies, or desktop applications.

For more information about Red Hat Errata advisories, see https://access.redhat.com/support/policy/updates/errata/

Cross-Platform Vulnerability Complexity

This section of the appendix explores cross-platform vulnerability complexities, and includes information on the following topics:

- Criteria for Defining Evaluation Platforms for Analysis
- A Conservative Evaluation Platform Approach

Criteria for Defining Evaluation Platforms for Analysis

The objective of this section is to define a clear exposition of the reasons for assigning vulnerabilities to platforms.

Platforms are released to the public to provide a collection of functions and services desired by customers. All platforms are released and are exposed to issues from prior versions or with artifacts from the design and development process that result from unwanted or unneeded features. Regardless of whether a particular behavior results from a prior version or the design process, it may present unanticipated vulnerabilities that are discovered after the platform is released. Such vulnerabilities are increasingly being reported to the National Vulnerability Database (NVD). These vulnerabilities may have existed when the platform was publicly released, but they are not counted until they have been reported to the NVD database.

The Common Platform Enumeration (CPE) is designed to have “the capability to completely and unambiguously characterize the software systems, hardware devices and network connections which comprise an enterprise’s computing infrastructure.” The NVD uses the CPE to enumerate its “vulnerable software list.” Some of the vulnerabilities are assigned to a platform of interest (for example, Windows Vista), but some are assigned to components that are released with the platform (for example, DirectX). It is also important to note that some vulnerabilities affect a large number of platforms, such as the GDI+ vulnerabilities discovered in the fourth quarter of 2009 (4Q09).

A Conservative Evaluation Platform Approach

One critical objective of this project is to define the criteria for grouping vulnerabilities into platforms that can be justifiably compared. The credibility of this project depends on the choices made to determine which default features (as identified by the CPE definition of the NVD) are included in each platform and the rationale used to inform these choices. Microsoft has taken a conservative approach by including all default functionality for analysis with Microsoft platforms. If a feature or capability need was unclear, its corresponding CVE record was excluded. This approach was taken to make the functional comparisons between all of the evaluated platforms as close and fair as possible.

An evaluation platform in the context of this report is a collection of software capabilities that work together to deliver comparable functionality to customers.

A simple usability comparison was used to balance the study, in dissimilar environments. Broadly speaking, the platform comparison groups of interest for this analysis are:

1. Internet browsers running on client operating systems.
2. Client operating systems for individual users.
3. Server operating systems run by administrators.
4. Database platforms that support server applications managed by a database administrator.

For this analysis, it is important to note that the platform and browser comparisons of Microsoft products with well-defined and infrequent release cycles stand somewhat in contrast to the open source products that may have new releases every few months.

This contrast exists because the objectives of these respective platform and browser release schedules are different: some open source releases are designed to introduce new functionality, but many are specifically designed to address vulnerabilities reported in the NVD. For commercial products, such as those from Microsoft, vulnerabilities are typically addressed by software updates rather than new product releases.

The Common Platform Enumeration (CPE) entries for the Common Vulnerabilities and Exposures (CVE) available in the NVD were used in most cases to determine which vulnerabilities to assign to an evaluated platform. Specific other criteria are described in each “Evaluated Platform” section above. The choice of which vulnerabilities to include for each platform and browser in this report is based on the intended function of each evaluation group using the following criteria:

- Whether a given vulnerability is based on the intended exploit that an attacker could take advantage of in the platform or browser to perform some unintended purpose.
- The function of the collection of software features are matched between platforms and browsers in each group to make the comparisons between them as legitimate as possible. See the previous “A Conservative Platform Evaluation Approach” subsection for more information on how this rational is applied.
- The decision to include a particular vulnerability in the NVD, which is based on existing or potential exploits that attackers could use to make the software perform some unintended function. Only those vulnerabilities that apply to one of the evaluation platforms are included in this analysis. A review of the excluded vulnerabilities should be undertaken each quarter to ensure that all vulnerabilities that impact any of the evaluation platform are counted.

Appendix B: Ubuntu Security Notice

New Client side Vulnerabilities disclosed in February. This information is provided as supporting information only. Note this Notice requires Linux kernel rebuild to correct these high risk Vulnerabilities.

```
==========================================================
Ubuntu Security Notice USN-1054-1         February 01, 2011
linux, linux-ec2 vulnerabilities
CVE-2010-0435, CVE-2010-4165, CVE-2010-4169, CVE-2010-4249
==========================================================

A security issue affects the following Ubuntu releases:

Ubuntu 10.04 LTS
Ubuntu 10.10

This advisory also applies to the corresponding versions of
Kubuntu, Edubuntu, and Xubuntu.

The problem can be corrected by upgrading your system to the
following package versions:

Ubuntu 10.04 LTS:
  linux-image-2.6.32-28-386       2.6.32-28.55
  linux-image-2.6.32-28-generic   2.6.32-28.55
  linux-image-2.6.32-28-generic-pae  2.6.32-28.55
  linux-image-2.6.32-28-ia64      2.6.32-28.55
  linux-image-2.6.32-28-lpia      2.6.32-28.55
  linux-image-2.6.32-28-powerpc   2.6.32-28.55
  linux-image-2.6.32-28-powerpc-smp  2.6.32-28.55
  linux-image-2.6.32-28-powerpc64-smp  2.6.32-28.55
  linux-image-2.6.32-28-preempt   2.6.32-28.55
  linux-image-2.6.32-28-server    2.6.32-28.55
  linux-image-2.6.32-28-sparc64   2.6.32-28.55
  linux-image-2.6.32-28-sparc64-smp  2.6.32-28.55
  linux-image-2.6.32-28-versatile  2.6.32-28.55
  linux-image-2.6.32-28-virtual   2.6.32-28.55
  linux-image-2.6.32-312-ec2      2.6.32-312.24

Ubuntu 10.10:
  linux-image-2.6.35-25-generic   2.6.35-25.44
  linux-image-2.6.35-25-generic-pae  2.6.35-25.44
  linux-image-2.6.35-25-omap      2.6.35-25.44
  linux-image-2.6.35-25-powerpc   2.6.35-25.44
  linux-image-2.6.35-25-powerpc-smp  2.6.35-25.44
  linux-image-2.6.35-25-powerpc64-smp  2.6.35-25.44
  linux-image-2.6.35-25-server    2.6.35-25.44
  linux-image-2.6.35-25-versatile  2.6.35-25.44
  linux-image-2.6.35-25-virtual   2.6.35-25.44

After a standard system update you need to reboot your computer
to make all the necessary changes.
```

ATTENTION: Due to an unavoidable ABI change the kernel updates have been given a new version number, which requires you to recompile and reinstall all third party kernel modules you might have installed. If you use linux-restricted-modules, you have to update that package as well to get modules which work with the new kernel version. Unless you manually uninstalled the standard kernel metapackages (e.g. linux-generic, linux-server, linux-powerpc), a standard system upgrade will automatically perform this as well.

Details follow:

Gleb Napatov discovered that KVM did not correctly check certain privileged operations. A local attacker with access to a guest kernel could exploit this to crash the host system, leading to a denial of service. (CVE-2010-0435)

Steve Chen discovered that setsockopt did not correctly check MSS values. A local attacker could make a specially crafted socket call to crash the system, leading to a denial of service. (CVE-2010-4165)

Dave Jones discovered that the mprotect system call did not correctly handle merged VMAs. A local attacker could exploit this to crash the system, leading to a denial of service. (CVE-2010-4169)

Vegard Nossum discovered that memory garbage collection was not handled correctly for active sockets. A local attacker could exploit this to allocate all available kernel memory, leading to a denial of service. (CVE-2010-4249)