Jeff Jones Security Blog
http://blogs.csoonline.com/blog/jeff_jones
http://blogs.technet.com/security
November 27, 2007

BROWSER VULNERABILITY ANALYSIS

OF INTERNET EXPLORER AND FIREFOX

By Jeffrey R. Jones

Security Guy

Jones on Security

Jeff Jones Security Blog
http://blogs.csoonline.com/blog/jeff_jones
http://blogs.technet.com/security
November 27, 2007

Contents

EXECUTIVE SUMMARY

For most people, their web browser is central to their interaction with the Internet, connecting to global web sites and helping them consume online services providing everything from booking flights to banking services to online shopping. This reality makes browsers a key tool when evaluating the security experience of users as the browser interprets Web content and programs delivered from around the world.

Over the past few years, there has been much discussion of the need for improvements in browser security, but few hard data studies performed to support assertions concerning the security of available browsers.

This report documents the results of my analysis of Internet Explorer and Firefox vulnerabilities over the past few years since Internet Explorer 6 on Windows XP SP2 became available and Mozilla launched Firefox.

Over the past 3 years, supported versions of Internet Explorer have experienced fewer vulnerabilities and fewer High severity vulnerabilities than Firefox, a result that stands in contrast to early assertions by Mozilla that Firefox "*won't harbor nearly as many security flaws as those that have Microsoft's Internet Explorer.*"[1]

The report in detail examines vulnerabilities over the past 3 years, breaks them down by severity, looks at version-over-version trends for each browser and finally examines how each browser is doing in terms of unfixed vulnerabilities.

[1] Mozilla President Baker in CNET article Mozilla: We're more secure than Microsoft.

OVERVIEW

When thinking about vulnerabilities, browsers are one of the most sensitive pieces of software on a computer, as they are the door through which users interact with the Internet. They are used to interpret Web content developed outside of user control by professionals and amateurs that may have everything from benign to malicious intentions. So, what are browser vendors doing about keeping their products secure?

Microsoft introduced a focus on security improvement as part of their Trustworthy Computing initiative, launched in January 2002, and made changes to improve the security of Internet Explorer. This resulted in shipping Internet Explorer with an Enhanced Security Configuration in Windows Server 2003, several security advances in Internet Explorer 6 in Windows XP SP2 and more recently, a whole new set of security features in Internet Explorer 7.

Similarly, when Mozilla launched the first version of Firefox in November 2004, security was a key part of the value benefit that they described, and statements from that time indicate that the Mozilla team was thinking about security as an important aspect of their efforts.

While there are other browsers that one could consider, Internet Explorer and Firefox ultimately represent the forefront of efforts and claims with respect to browsers and security, and likely suffer higher levels of scrutiny by security researchers than other browser options.

Ultimately, security professionals recognize that flaw-free software is an aspiration. With that context in mind, I've undertaken an analysis of Internet Explorer and Firefox software vulnerabilities, along with related vendor actions and policies that could impact user risk.

ALL SUPPORTED BROWSERS

Let's start out with a high level view and then proceed with digging deeper into issues further along in the report.

Mozilla released Firefox 1.0 in November 2004 and has subsequently released Firefox 1.5 and Firefox 2.0. These three versions make up the supported Firefox versions in the three years from November 2004 to October 2007. The time period covered in this report is through the end of October 2007.

In that same timeframe, Microsoft has supported Internet Explorer 5.01 SP3 and SP4, Internet Explorer 6.0 Gold, SP1, SP2, and Windows Server 2003 edition, plus Internet Explorer 7.

Since the release of Firefox 1.0 in November 2004, Mozilla has fixed 199 vulnerabilities in supported Firefox products – 75 HIGH severity, 100 MEDIUM severity and 24 LOW severity. In the same timeframe, Microsoft has fixed 87 total vulnerabilities affecting all supported versions of Internet Explorer – 54 HIGH severity, 28 MEDIUM severity, and 5 LOW severity. This is charted in Figure 1.

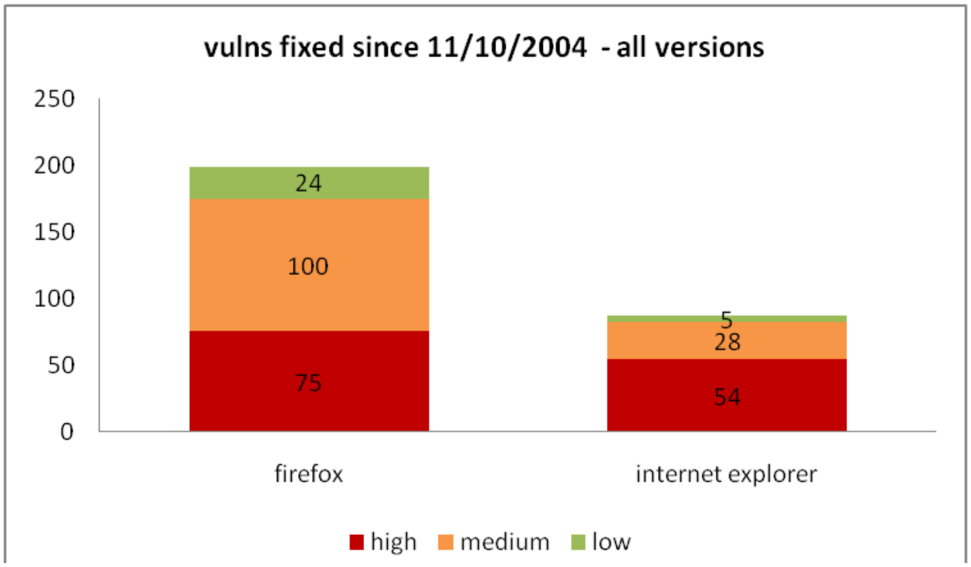

Figure 1: Vulns fixed in all supported versions

The chart shows results in stark contrast to early predictions that Firefox would have many fewer vulnerabilities than Internet Explorer, and more than anything, highlights that security quality should be a concern for all vendors across the industry.

This chart displaying fixed vulnerabilities naturally poses the question of how the respective vendors have done on publicly disclosed, but unfixed issues as well. I will come back to address that question later in the report, as we dig deeper.

LIFECYCLE SUPPORT POLICIES

In performing this analysis, I learned of significant differences in lifecycle support policies between the vendors that have potential security implications.

Mozilla released Firefox 1.0 in November 2004, Firefox 1.5 in November 2005, and Firefox 2.0 in October 2006. Only Firefox 2.0 is currently supported with security fixes from Mozilla, as it is has been Mozilla's policy to support a previous version for six months after a new (major) version is released. So, according to its original schedule, Firefox 3.0 was scheduled to ship in November 2007, which meant Firefox 2.0 support would end in May 2008[2]. To put this in perspective, if Microsoft had this same policy, then support of Internet Explorer 6 would have ended in May 2007, or similarly Internet Explorer 5.01 support would have ended in 2001.

In contrast, Microsoft generally releases a browser in conjunction with a new operating system release and commits to supporting that version for the lifecycle of the product – now 10 years for business products. Major versions do have service packs and the Microsoft policy is to support a previous service pack for at least one year after a new service pack is released.

[2] While a revised schedule has not officially been announced by Mozilla, they have announced that three Beta releases are planned and the current estimate for Firefox 3.0 is "early 2008."

Microsoft released Internet Explorer 6 for Windows XP SP2 in August 2004 and Internet Explorer 7 in October 2006 (for Windows XP SP2 – Internet Explorer 7 Vista released with Windows Vista in November 2007). Both versions of Internet Explorer are currently supported by Microsoft. Figure 2 shows a timeline of browser releases since November 2004, along with end of life for those products no longer in support.

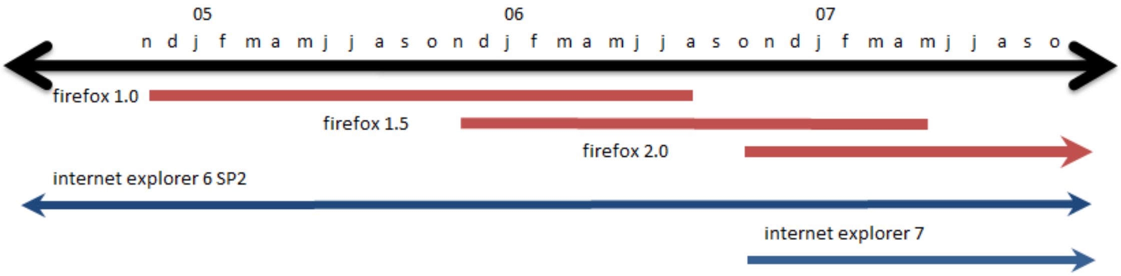

Figure 2: Firefox and Internet Explorer releases since November 2004

Actually, though not shown in the diagram, Internet Explorer 5.01 SP4 is also still supported for those Windows 2000 users that have made the decision never to upgrade their browser to a different release.

One key factor of lifecycle is simply the fact that "unsupported" versions of products don't get patches developed for them. This is equally true for all vendors, but shorter lifecycles mean more people may still be running an unsupported version and be exposed. To explain this comment, let's look at an example using Microsoft IE6 SP2. Imagine that after IE7 was released last October that one month later support for IE6 would end. How likely is that everyone will have upgraded by the end of that month? What if it was six months? Is it likely some consumers or companies might not have upgraded by the end of the six month grace period?

Lifecycle issues are also illustrated by looking at Enterprise Linux distributions. Ubuntu 6.06 LTS integrates Firefox 1.5 and has a security support commitment until 2009. Novell SUSE Linux Enterprise Desktop 10 (SLED10) integrates Firefox 1.5 and has a security support commitment until 2013. Red Hat shipped Red Hat Enterprise Linux Desktop 5 (RHEL5) with Firefox 1.5 in March 2007 with a security support commitment until 2014.

However, Mozilla stopped support of Firefox 1.5 in May 2007, only 2 months after RHEL5 shipped and has this statement on their web site:

> **Firefox 1.5 is no longer supported and the last update, Firefox 1.5.0.12, is affected by several vulnerabilities fixed in newer versions of the program. All users are urged to upgrade to the newest version of Firefox.**

So, though Firefox 1.5 isn't actually a supported product by Mozilla anymore, and thus won't get security fixes, distribution users may have to continue using it in order to maintain their support policy. This presents somewhat of a security dilemma for the distribution vendors. They are faced with a couple of choices.

One choice is to self-support and issue distribution-specific patches. Both Red Hat and Ubuntu released patches for their versions of Firefox 1.5 in July 2007. The Red Hat advisory specifically says their patch contains "backported patches", presumably backported from the Firefox 2.0 patch. It should be noted however that the vulnerabilities patched by each vendor only overlap partially.

Ubuntu 6.06 LTS	CVE-2007-3089, CVE-2007-3285, CVE-2007-3656, CVE-2007-3734, CVE-2007-3735, CVE-2007-3736, CVE-2007-3737, CVE-2007-3738 CVE-2007-3844, CVE-2007-3845
Red Hat EL 5	CVE-2007-3089, CVE-2007-3656, CVE-2007-3734, CVE-2007-3735, CVE-2007-3736, CVE-2007-3737, CVE-2007-3738

At the time of this writing, the National Vulnerability Database entry for CVE-2007-3844 indicates Red Hat is investigating and may address the issue in a future update. There is no comment on the other two issues.

The other choice is to force users to upgrade to the version supported by Mozilla, which seems to be the route taken by Novell. In March 2007, they released updated packages for SLED10 containing Firefox 1.5.0.10 addressing several security vulnerabilities. Their next Firefox "patch" in June was a set of replacement packages upgrading to Firefox 2.0.0.4. In the past, Red Hat has taken this route as well, forcing upgrades from Firefox 1.0 to Firefox 1.5 in Red Hat RHEL4 in July 2006.

Lifecycle considerations are likely more important to corporate enterprises, as they sometimes have custom web applications and are hesitant to upgrade between major releases very often, and even then may have a relatively long transition plan.

Home users have a different issue in that they just need to make sure they are aware of when a product end-of-life occurs so that they can upgrade or they may be exposed. Again, I reiterate that this true for any software product that reaches end of life, but with shorter lifecycles, users may have expectations of longer support lifecycles.

THREE YEARS OF BROWSER VULNERABILITIES

Given what we just learned concerning browser lifecycles, let's next look at what the situation would have been for a browser user over the past two years from the time Firefox launched. There are two basic types of users – boundary cases, if you will – we can consider for both Firefox and Internet Explorer:

- Users that upgrade as quickly as they can
- Users that stick with the version they have as long as they can

Using the lifecycle timeline in Figure 2 as a guide starting from November 2004, these are the scenarios we get:

1. FFa (upgrade soonest) Firefox 1.0 from Nov 2004 to Nov 2005. Firefox 1.5 from Nov 2005 to Oct 2006. Firefox 2.0 from Oct 2006 to Oct 2007.

2. FFb (upgraded latest) Firefox 1.0 from Nov 2004 to Apr 2006. Firefox 1.5 from Apr 2006 to May 2007. Firefox 2.0 from May 2007 to Oct 2007.
3. IEa (upgrade soonest) Internet Explorer SP2 from Nov 2004 to Oct 2006. Internet Explorer 7 from Oct 2006 to Oct 2007.
4. IEb (upgrade latest) Internet Explorer SP2 from Nov 2004 to Oct 2007.

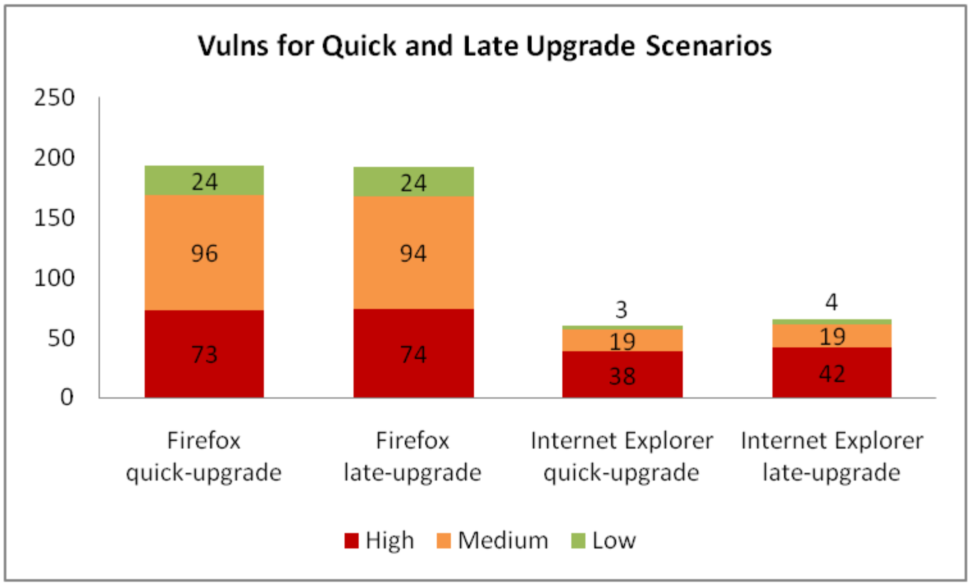

Figure 3: Vulnerabilities for quick and late upgrade scenarios

As shown in Figure 3, it made little difference (0.5%) to Firefox users, in terms of security vulnerabilities, whether they upgraded as soon as a new version came out or waited until support for their current version ended. While the difference for Internet Explorer users was also relatively small (8%), upgrading[3] to IE7 quicker did result in four fewer High severity issues overall.

Perhaps the most striking observation from Figure 3 is that either Internet Explorer scenario in this time period resulted in fewer vulnerabilities in total than just the High severity Firefox vulnerabilities. This again stands in contrast to early predictions for fewer Firefox vulnerabilities.

INTERNET EXPLORER TRENDS

Microsoft shipped Internet Explorer 6 SP2 in August 2004 and in the three years since then has fixed a total of 79 vulnerabilities – 50 High / 24 Medium / 5 Low – or an average of about 2.1 per month.

Microsoft shipped Internet Explorer 7 in October 2006 for Windows XP SP2 and in November 2006 as part of Windows Vista. In the nearly one year since release, Microsoft has fixed a total of 17

[3] These Internet Explorer upgrade numbers assume a user stays on Windows XP SP2. If the user also upgraded the operating system to Windows Vista, there were three fewer vulnerabilities.

vulnerabilities in IE7 – 14 High / 3 Medium – or an average of about 1.4 per month. Only 14 of the vulnerabilities have affected the Vista release, so that rate is slightly lower.

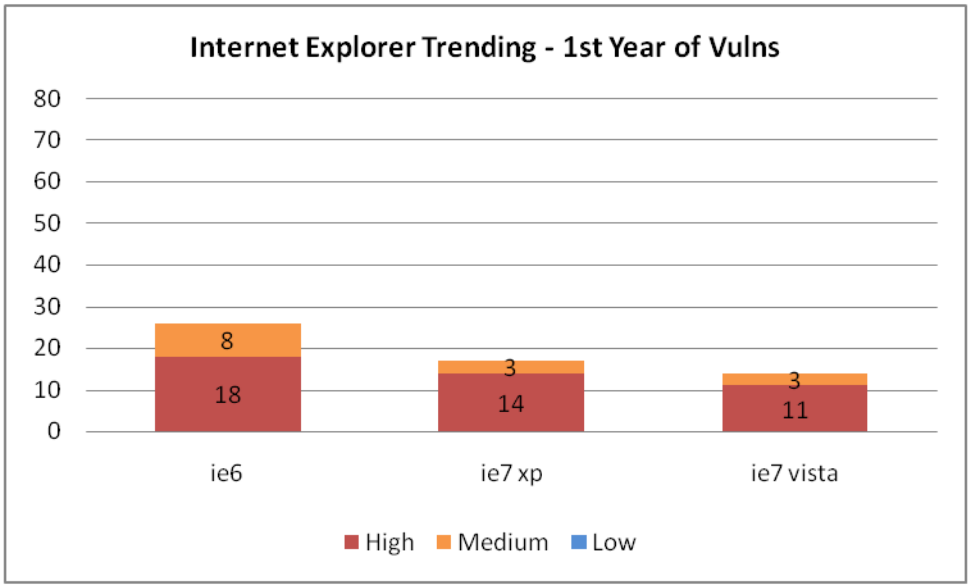

Figure 4: Internet Explorer version vulnerability trending

Figure 4charts the first year of vulnerability fixes for Internet Explorer 6 and Internet Explorer 7, on both Windows XP SP2 and Windows Vista. The data indicates that the latest version of Internet Explorer has improved security in terms of fewer vulnerabilities than previous releases, with the Vista version being a bit better than the XP SP2 version.

FIREFOX TRENDS

Mozilla shipped Firefox 1.0 in November 2004 and ended support in April 2006. In the 17 months of support, Mozilla fixed 88 vulnerabilities in Firefox 1.0 – 36 High /33 Medium / 19 Low – or an average of about 5.2 vulnerabilities per month.

Mozilla shipped Firefox 1.5 in November 2005 and ended support in May 2007 (one month later than originally planned). In the 18 months of support, Mozilla fixed 107 vulnerabilities in Firefox 1.5 – 46 High /53 Medium / 8 Low – or an average of about six vulnerabilities per month.

Mozilla shipped Firefox 2.0 in October 2006, so it has been available for 12 months now. In the 12 months of support, Mozilla has fixed 56 vulnerabilities in Firefox 2.0 – 13 High / 42 Medium / 1 Low – or an average of about 3.75 vulnerabilities per month.

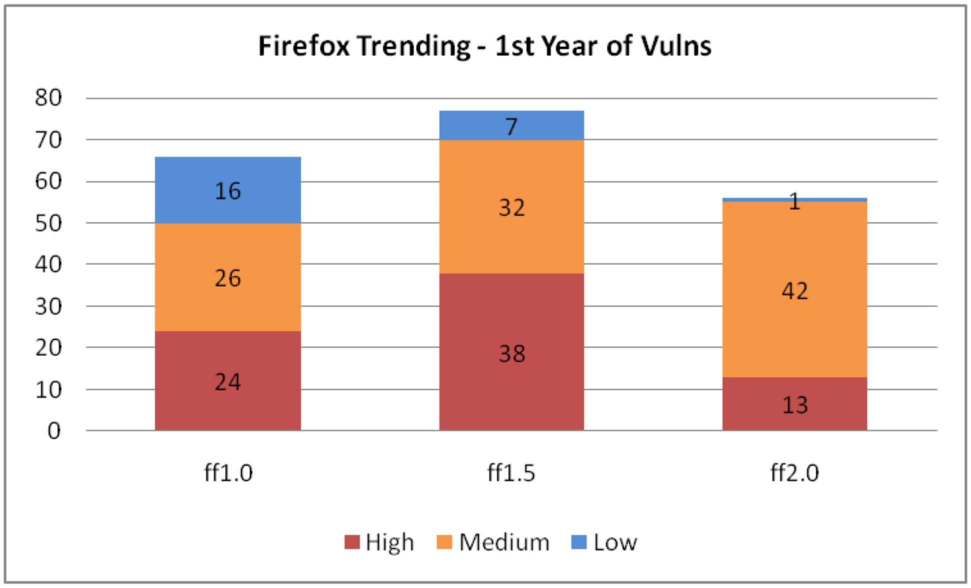

Figure 5: Firefox version trending - 1st year of vulns

Figure 5 charts the first year of fixed vulnerabilities for the three releases of Firefox since it began shipping in 2004. Mozilla indicated (interview here) that security was one of two focus areas (along with user experience) for Firefox 2.0. As we can see in the chart, assuming that the unfixed issues for each version are relatively consistent, then Firefox 2.0 is an improvement in terms of security quality over the previous Firefox releases.

UNFIXED VULNERABILITIES

As has been pointed out to me when I've previously published analyses of vulnerabilities fixed by vendors, the results only show part of the picture without performing an analysis of unfixed vulnerabilities. Unfixed vulnerabilities are more challenging to enumerate and analyze than fixed vulnerabilities since, in the later case, we can look to vendor advisories to enumerate the issues that have been addressed. However, with a lot of manual work, the analysis can be done.

To develop a list of unfixed vulnerabilities for the latest versions of Firefox and Internet Explorer, I utilized this process:

1. Compiled a list of vulnerabilities identified as affecting the respective browser (IE or Firefox) in the National Vulnerability Database (NVD) (http://nvd.nist.gov).
2. Marked a vulnerability fixed if either a vendor advisory noted it as fixed, or if the NVD referenced an advisory as addressing the issue. The latter was necessary because vendor advisories do not always list issues addressed by CVE identifier. For example, MFSA2005-50 does not mention any vulnerability by CVE identifier. However, the NVD entry for CVE-2005-2265 identifies MFSA2005-50 as the patch advisory for that issue. (fyi, this step benefitted only Firefox.)

3. Scrubbed the remaining list to remove rejects and duplicates as acknowledged by the NVD, plus issues where the browser was listed in error or the actual vulnerable product was not the browser. For example, CVE-2007-3657 lists Firefox 2 as an affected product, but it also indicates that other researchers have disputed the issue, so I didn't count this for Firefox. Similarly, CVE-2007-1377 lists Firefox as affected, but the flaw is actually in an Adobe plug-in, so I did not count it.
4. Looked at various other references to the issue to try and determine if it applied to the browser version in question.

There are some caveats on the list of unfixed issues that I would like to call out. There are additional vulnerabilities listed in the NVD for Firefox 1.0 and Firefox 1.5 for which I cannot verify a fix from the vendor, which were disclosed prior to the release of Firefox 2. They may still exist in Firefox 2 or they may have been silently addressed as part of one of the subsequent version releases, I simply can't tell. Similarly, there are vulnerabilities listed in the NVD for Internet Explorer 6 and previous that were disclosed prior to the release of Internet Explorer 7 which may or may not have been addressed as part of the IE7 release – again I can't tell, except in a few cases where some researchers have pointed out that an IE6 issues still exists in IE7, so I counted those cases. Perhaps this report will spur further research in this area for browsers.

Results are charted in Figure 6 for the current set of Firefox 2 and Internet Explorer 7 vulnerabilities that have been disclosed and are listed in the NVD, but have yet to receive a patch from the vendor.

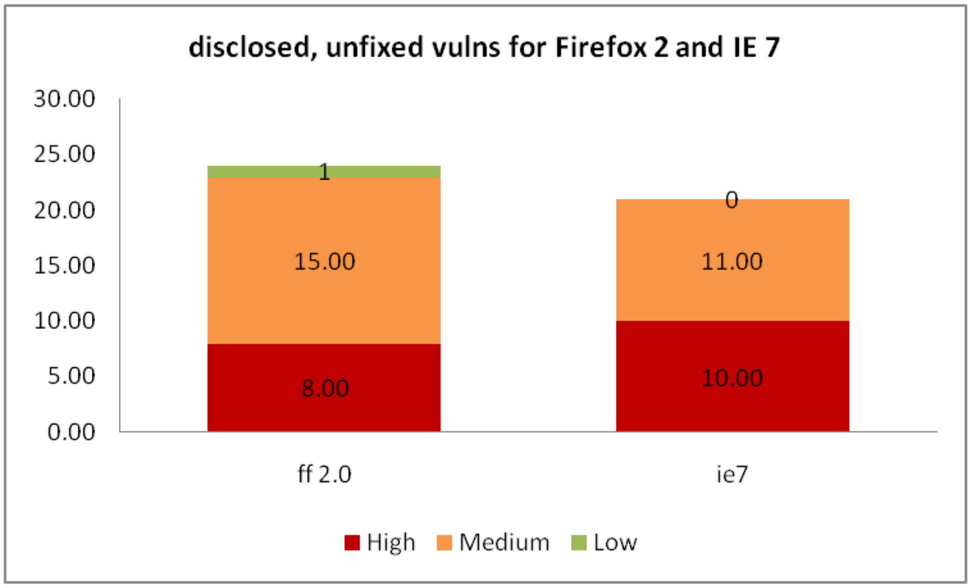

Figure 6: Disclosed and unfixed vulnerabilities for Firefox 2 and IE 7

Actually, two of the IE7 issues only affect the Windows XP platform and one of the issues only affects the Windows Vista platform, so the platform-specific totals would be slightly lower. However, I didn't have the ability to distinguish platform-specific issues for Firefox, so I thought it better to simply chart the totals.

FINAL OBSERVATIONS

Microsoft has publicly emphasized the need for improved security and implemented steps to raise the bar against attackers in browser security. Microsoft efforts have resulted in shipping Internet Explorer with an Enhanced Security Configuration in Windows Server 2003, several security advances in Internet Explorer 6 in Windows XP SP2 and more recently, a whole new set of security features and architectural improvements in Internet Explorer 7.

Mozilla launched the first version of Firefox in November 2004 with security articulated as a key part of the value benefit that they described and since then have taken further steps, such as hiring "Chief Security Something or Other" Window Snyder to help drive further security improvements within the community.

While the data trends show that both Internet Explorer and Firefox security quality is improved in the latest version, it also demonstrates that, contrary to popular belief, Internet Explorer has experienced fewer vulnerabilities than Firefox.

While the results in this study showing fewer vulnerabilities in Internet Explorer might be surprising to some, to others the results will simply be a confirmation that improving security is a hard job even with the best of intentions. Further, it shows that with commitment and focused effort, vendors can make progress in improving computer security for software products. As someone who has been closely involved in Microsoft security improvement efforts over the past five years, I believe the improved security quality demonstrated in Internet Explorer is a result of the Microsoft Security Development Lifecycle (SDL) and implementation of the security commitment under the Trustworthy Computing Initiative.

ABOUT THE AUTHOR

Jeff Jones is a Security Strategy Director in Microsoft's Trustworthy Computing group. In this role, Jeff draws upon his security experience to work with enterprise CSOs and Microsoft's internal security teams to drive practical and measurable security improvements into Microsoft process and products. Prior to his strategic position at Microsoft, Jeff was the vice president of product management for security products at Network Associates where his responsibilities included PGP, Gauntlet and Cybercop products, and several improvements in the McAfee product line. These latest positions cap a 20 year hands on career in security, performing risk assessments, building custom firewalls and being involved in DARPA security research projects while part of Trusted Information Systems. Jeff is a frequent global speaker and writer on security topics ranging from the very technical to more high level, CxO-focused topics such as Security TCO and metrics.

Jeff actively encourages readers to challenge his assumptions, analysis and conclusions and provide critical feedback – but asks for equal (or better) rigor in methodology and analysis to support the challenges, as opposed to enthusiastic espousal of unsupported evangelistic fervor.

APPENDIX: DATA SOURCES AND METHODOLOGY

The efforts to identify and fix vulnerabilities lacked a common naming mechanism until a consortium led by the Mitre Corporation began publishing the Common Vulnerabilities and Exposure (CVE) list, in an attempt to drive a common naming mechanism that could be leveraged by multiple vulnerability databases and security products. The CVE naming conventions and process has achieved success in being the most comprehensive list of vulnerabilities across software products of all types and worldwide. In this report, I use the CVE naming convention when identifying individual vulnerabilities.

The analysis in this report uses a set of data that has been compiled, customized and cross-checked using several sources of data available on the Internet:

- Microsoft Security Bulletins as published at http://www.microsoft.com/technet/security/current.aspx and associated web pages.
- Mozilla Foundation Security Advisories as published at http://www.mozilla.org/security/announce/ and associated web pages.
- The National Vulnerability Database (NVD) , a database superset of the Mitre CVE list (http://cve.mitre.org) which provides additional objective information concerning vulnerabilities was the source utilized for severity ratings and exploit complexity assessment. The NVD is also sponsored by the US Department of Homeland Security and makes their data downloadable in an XML format at http://nvd.nist.gov/download.cfm.
- Many security websites were utilized for detailed verification and validation of vulnerability details, and especially dates for when the issue was first discussed publicly. Some of the most commonly utilized were: www.securityfocus.com, the Bugtraq mailing list, www.secunia.com, and www.securitytracker.com, but there were many others.

Leveraging these and many other sources, I compiled a database of vulnerabilities for the browsers analyzed and a database of disclosure dates for vulnerabilities to use in determining which year, month, and day that each vulnerability was disclosed publicly and broadly for the first time.

Note that more detail is also provided in the section entitled “Unfixed Vulnerabilities” with respect to compiling unfixed vulnerabilities.

Note that in this report, “disclosure” is used to mean broad and public disclosure and not any sort of private disclosure or disclosure to a limited number of people.