Microsoft Security Intelligence Report

Volume 13

January through June, 2012

www.microsoft.com/sir

Microsoft Security Intelligence Report

This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

This document is provided "as-is." Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

Authors

Danielle Alyias
Microsoft Trustworthy Computing

Dennis Batchelder
Microsoft Protection Technologies

Joe Blackbird
Microsoft Malware Protection Center

Joe Faulhaber
Microsoft Malware Protection Center

David Felstead
Bing

Roger A. Grimes
Microsoft IT Information Security and Risk Management

Paul Henry
Wadeware LLC

Jeff Jones
Microsoft Trustworthy Computing

Jimmy Kuo
Microsoft Malware Protection Center

Marc Lauricella
Microsoft Trustworthy Computing

Jenn LeMond
Microsoft IT Security and Risk Management

Nam Ng
Microsoft Trustworthy Computing

Daryl Pecelj
Microsoft IT Information Security and Risk Management

Anthony Penta
Microsoft Windows Safety Platform

Tim Rains
Microsoft Trustworthy Computing

David Ross
Microsoft Trustworthy Computing

David Seidman
Microsoft Trustworthy Computing

Weijuan Shi Davis
Windows Business Group

Holly Stewart
Microsoft Malware Protection Center

Matt Thomlinson
Microsoft Trustworthy Computing

Terry Zink
Microsoft Exchange Online Protection

Contributors

Doug Cavit
Microsoft Trustworthy Computing

Enrique Gonzalez
Microsoft Malware Protection Center

Heather Goudey
Microsoft Malware Protection Center

Angela Gunn
Microsoft Trustworthy Computing

Satomi Hayakawa
CSS Japan Security Response Team

Greg Lenti
CSS Security Readiness & Response Team

Le Li
Microsoft Windows Safety Platform

Ken Malcolmson
Microsoft Trustworthy Computing

Hideya Matsuda
CSS Japan Security Response Team

Takumi Onodera
Microsoft Premier Field Engineering, Japan

Kathy Phillips
Microsoft Legal and Corporate Affairs

Hilda Larina Ragragio
Microsoft Malware Protection Center

Laura A. Robinson
Microsoft Information Security & Risk Management

Richard Saunders
Microsoft Trustworthy Computing

Jasmine Sesso
Microsoft Malware Protection Center

Frank Simorjay
Microsoft Trustworthy Computing

Mark Simos
Microsoft Consulting Services

Norie Tamura
CSS Japan Security Response Team

Kurt Tonti
Microsoft Information Security & Risk Management

Henk van Roest
CSS Security EMEA

Patrik Vicol
Microsoft Malware Protection Center

Steve Wacker
Wadeware LLC

Iaan Wiltshire
Microsoft Malware Protection Center

Dan Wolff
Microsoft Malware Protection Center

The Microsoft Pass-the-Hash Working Group

Table of Contents

About this report

The *Microsoft Security Intelligence Report (SIR)* focuses on software vulnerabilities, software vulnerability exploits, and malicious and potentially unwanted software. Past reports and related resources are available for download at www.microsoft.com/sir. We hope that readers find the data, insights, and guidance provided in this report useful in helping them protect their organizations, software, and users.

Reporting period

This volume of the *Microsoft Security Intelligence Report* focuses on the first and second quarters of 2012, with trend data for the last several years presented on a quarterly basis. Because vulnerability disclosures can be highly inconsistent from quarter to quarter and often occur disproportionately at certain times of the year, statistics about vulnerability disclosures are presented on a half-yearly basis.

Throughout the report, half-yearly and quarterly time periods are referenced using the *n*H*yy* or *n*Q*yy* formats, where *yy* indicates the calendar year and *n* indicates the half or quarter. For example, 1H12 represents the first half of 2012 (January 1 through June 30), and 4Q11 represents the fourth quarter of 2011 (October 1 through December 31). To avoid confusion, please note the reporting period or periods being referenced when considering the statistics in this report.

Conventions

This report uses the Microsoft Malware Protection Center (MMPC) naming standard for families and variants of malware and potentially unwanted software. For information about this standard, see “Microsoft Malware Protection Center Naming Standard” on the MMPC website.

Trustworthy Computing: Security engineering at Microsoft

Amid the increasing complexity of today's computing threat landscape and the growing sophistication of criminal attacks, enterprise organizations and governments are more focused than ever on protecting their computing environments so that they and their constituents are safer online. With more than a billion systems using its products and services worldwide, Microsoft collaborates with partners, industry, and governments to help create a safer, more trusted Internet.

Microsoft's Trustworthy Computing organization focuses on creating and delivering secure, private, and reliable computing experiences based on sound business practices. Most of the intelligence provided in this report comes from Trustworthy Computing security centers—the Microsoft Malware Protection Center (MMPC), Microsoft Security Response Center (MSRC), and Microsoft Security Engineering Center (MSEC)—which deliver in-depth threat intelligence, threat response, and security science. Additional information comes from product groups across Microsoft and from Microsoft IT (MSIT), the group that manages global IT services for Microsoft. The report is designed to give Microsoft customers, partners, and the software industry a well-rounded understanding of the threat landscape so that they will be in a better position to protect themselves and their assets from criminal activity.

Deceptive downloads: Software, music, and movies

Malware authors go to great lengths to distribute their wares, and they invest significant resources into finding victims and avoiding detection by antimalware products. Attackers experiment with different methods and mechanisms for distributing malware, ranging from exploits to pure social-engineering–based approaches. Recently, the Microsoft Malware Protection Center (MMPC) has observed a growing trend of malware infection associated with unsecure supply chains—the websites, protocols, and other channels by which software and media files are informally distributed, both legally and illegally. Unsecure distribution mechanisms range from underground sites where pirated software and media are openly exchanged, to legitimate websites that make shareware or free music files available for public download. In some cases, malware has even been discovered preinstalled on computers sold at retail.[1] Any mechanism by which untrusted parties can distribute files to a wider audience without sufficient safeguards in place is a potential vehicle for malware dissemination.

This section of the *Microsoft Security Intelligence Report* examines how attackers take advantage of these unsecure supply chains to distribute malware to victims around the world, with data and analysis about the problem based on Microsoft antimalware telemetry. It also provides guidance that computer users and administrators can use to help protect themselves from malware distributed through unsecure supply chains.

Detecting malware associated with unsecure supply chains

Through analysis of the data reported by Microsoft antimalware products running on computers that have been opted in to data collection,[2] it is possible to discern patterns of activity that show a correlation between unsecure supply chains and malware. In some cases, this correlation may simply involve malware samples that have the same names as certain files that are known to be disseminated on file-distribution sites and networks—spreading malware by claiming it is something else is a time-honored tactic used by attackers.

In other cases, a correlation can be drawn from the presence on the reporting computer of other threat families—including Win32/Keygen, Win32/Pameseg,

[1] See "Operation b70: Nitol Malware Research and Analysis," a report by the Microsoft Digital Crimes Unit, for additional details about one such incident.
[2] See "Appendix B: Data sources" on page 113 for links to privacy statements for the products and services that provided the data for this report.

and Win32/Gendows—that are strongly associated with file distribution activity. These indicator families were detected on 16.8 percent of all computers reporting detections in the first quarter of 2012, increasing to 17.2 percent of computers in the second quarter. Some of these indicator families are considered potentially unwanted software rather than malware, but all can be taken as evidence that file distribution activity has probably occurred. By looking at malware detected alongside the indicator families and comparing it with malware detections reported by computers that *don't* also report detections of indicator families, MMPC researchers can estimate the extent and impact of attackers' abuse of the file distribution supply chain.

Malware and unsecure software distribution

The most commonly reported threat family in 1H12 was Win32/Keygen, a detection for tools that generate keys for various software products. Software pirates often bundle a key-generator utility with a well-known application and then distribute the package using a torrent client or by uploading the package to a file distribution site. A user who downloads the package runs the key-generator utility to create a product key that will supposedly allow the software to be used illegally. Its widespread impact—of the 105 countries or regions covered in this report, 98 percent listed Keygen as one of the top 10 families detected in 1H12—and its strong association with unsecure file distribution activity make it a good indicator family to use to examine how attackers exploit such activity to distribute malware.

An examination of Keygen reports shows a diverse list of popular software products being targeted, as indicated by some of the file names used by the Keygen executable:

- keygen.exe
- Windows Loader.exe
- mini-KMS_Activator_v1.1_Office.2010.VL.ENG.exe
- AutoCAD-2008-keygen.exe
- SonyVegasPro Patch.exe
- Nero Multimedia Suite 10 - Keygen.exe
- Adobe.Photoshop.CS5.Extended.v12.0.Keymaker-EMBRACE.exe
- Call.of.Duty.4.Modern.Warfare.Full-Rip.Skullptura.7z
- Guitar Pro v6.0.7+Soundbanks+Keygen(Registered) [kk].rar
- Half Life CDkeygen.exe

Installing pirated software bears significant risks. In many cases, the distributed packages contain malware alongside (or instead of) the pirated software, which takes advantage of the download and install process to infect the computers of users who download the bundles. More than 76 percent of computers reporting Keygen detections in 1H12 also reported detections of other threat families, which is 10 percent higher than the average co-infection rate for other families. (See "Malware statistics" on page 7 for additional information.)

The tactic of bundling malware with software on unsecure file distribution sites and networks is not limited to pirated commercial software—attackers sometimes take advantage of traffic in freely distributed software as well. In 1H12, the MMPC observed 35 different threat families being distributed using the file name install_adobeflash.exe, which purports to be an installation package for the freely distributed Adobe Flash Player. Threats that make use of this technique in 1H12 included notable families such as Win32/Sirefef, Win32/Bancos, and Win32/FakeRean. (See "Threat families" beginning on page 53 for more information about these and other threats.)

Similar tactics are used by attackers who engage in so-called paid archive schemes, in which users are convinced or tricked into paying for software that might otherwise be available for free. The most commonly detected threat family in 1H12 in Russia, Ukraine, and several other countries and regions in eastern Europe and western Asia was Win32/Pameseg, a family of programs that claim to install various popular software packages. A user who launches a Pameseg installer is instructed to send an SMS text message to a premium number (typically at a cost of between 5 and 20 US dollars, although the installer usually claims that it will be free of charge) to successfully install the program. Among the top file names used by Pameseg installers in 1H12 were several that resembled the names of programs that can be legally downloaded and installed for free, in addition to paid commercial programs:

- Adobe Photoshop CS5 key-rus.exe
- avast_free.exe
- DirectX11.exe
- kb909241x.exe
- LoviVkontakte.exe
- powerpoint-setup.exe
- Skype.exe
- SkypeSetup.exe
- vksaver.exe
- willarchive.exe

For more information about Pameseg and paid archive schemes, see the following entries in the MMPC blog (blogs.technet.com/mmpc):

- Easy Money: Program:Win32/Pameseg (part one) (November 14, 2011)
- Easy Money: Program:Win32/Pameseg (part two) (November 21, 2011)

Other hacking tools that are frequently used to distribute malware with shared or pirated software include:

- Win32/Gendows. A tool that attempts to activate Windows 7 and Windows Vista operating system installations.
- Win32/Patch. A family of tools intended to modify, or "patch," programs that may be evaluation copies or unregistered versions with limited features, for the purpose of removing the limitations.
- Win32/Wpakill. A family of tools that attempt to disable or bypass WPA (Windows Product Activation), WGA (Windows Genuine Advantage) checks, or WAT (Windows Activation Technologies) by altering Windows operating system files, terminating processes, or stopping services.

Music, movies, and malware

Like software, popular movies and music are often traded on unsecure file distribution sites and networks. As with software, attackers have taken advantage of the illegal trafficking in media files to spread malware.

The ASX/Wimad family is a generic detection for malicious URL script commands found in Advanced Systems Format (ASF) (a file format used by Windows Media) that download arbitrary files. Several of the file names used by Wimad files suggest a global hit parade of popular music:

- - - 1 Alejate De Mi - Camila.mp3
- - Lady Gaga - Telephone (feat. Beyonce).mp3
- - - Alexandra Stan - - - Mr. Saxobeat.mp3
- 0 Merche - Si Te Marchas.mp3
- 09. Pitbull - Back In Time (From Men In Black III).mp3
- 09 Back In Time - Pitbull.mp3
- Oasis - Stop Crying Your Heart Out.mp3
- - - Moves Like Jagger - Maroon 5 Christina Aguilera.mp3
- עמיר בן עיון עומד בשער.mp3 [Amir Benayun, "Standing at the Gate"]
- Rumer - Slow.mp3

Current popular films are also well-represented in the list of Wimad file names:

- The Avengers 2012 720p BDRip QEBS7 AAC20 MP4-FASM.avi
- Prometheus 2012 DVDRip.avi
- Wrath of the Titans 2012 DVDRip aXXo.avi
- Battleship 2012 DVDRip.avi
- What to Expect When You're Expecting 2012.BRRip.XviD-KAZAN.avi
- The Hunger Games 2012 TRUE FRENCH DVDRIP XViD FiCTiON L S79.avi
- Sherlock.Holmes.2.A.Game.of.Shadows.2012.DVDRip.XviD-26K-0123.avi
- The Five-Year Engagement 2012 HDRip XviD-HOPE.avi
- Project X 2012 TRUE FRENCH DVDRIP XViD FiCTiON L S79.avi
- Amazing SpiderMan 2012 DVDRiP XviD.avi

Malware statistics

Computers reporting detections of the six indicator families mentioned (Keygen, Wimad, Pameseg, Wpakill, Gendows, and Patch) have a higher malware detection rate than those that don't.[3] Figure 1 lists the families that were most commonly detected alongside the indicator families in 1H12.

[3] See "Appendix B: Data sources" on page 113 for information about the Microsoft products and

Figure 1. Threat families most commonly detected on computers displaying evidence of unsecure file distribution in 1H12, by absolute number of computers and by percentage of all computers displaying such evidence

Family	Most significant category	1Q12	1Q12 %	2Q12	2Q12 %
Win32/Autorun	Worms	849,108	10.5%	937,747	11.3%
JS/Pornpop	Adware	637,966	7.9%	661,711	8.0%
Win32/Obfuscator	Misc. Potentially Unwanted Software	515,575	6.4%	606,081	7.3%
Blacole	Exploits	561,561	7.0%	512,867	6.2%
Win32/Dorkbot	Worms	492,106	6.1%	522,617	6.3%

- Win32/Autorun is a generic detection for worms that spread between mounted volumes using the Autorun feature of Windows. Recent changes to the feature in Windows XP and Windows Vista have made this technique less effective,[4] but attackers continue to distribute malware that attempts to target it.

- JS/Pornpop is a detection for specially crafted JavaScript-enabled objects that attempt to display pop-under advertisements in users' web browsers. Initially, Pornpop appeared exclusively on websites that contained adult content; however, it has since been observed to appear on websites that may contain no adult content whatsoever.

- Win32/Obfuscator is a generic detection for programs that have had their purpose disguised to hinder analysis or detection by antivirus scanners. Such programs commonly employ a combination of methods, including encryption, compression, anti-debugging and anti-emulation techniques.

- Blacole is a multiplatform family of exploits that target vulnerabilities in popular products and components and are delivered through malicious or compromised webpages. (See page 23 for more information about Blacole.)

- Win32/Dorkbot is a worm that spreads via instant messaging and removable drives. It also contains backdoor functionality that allows unauthorized access and control of the affected computer. Dorkbot may be distributed from compromised or malicious websites using PDF or browser exploits.

See "Malware and potentially unwanted software" beginning on page 39 for more information about threat detection patterns around the world.

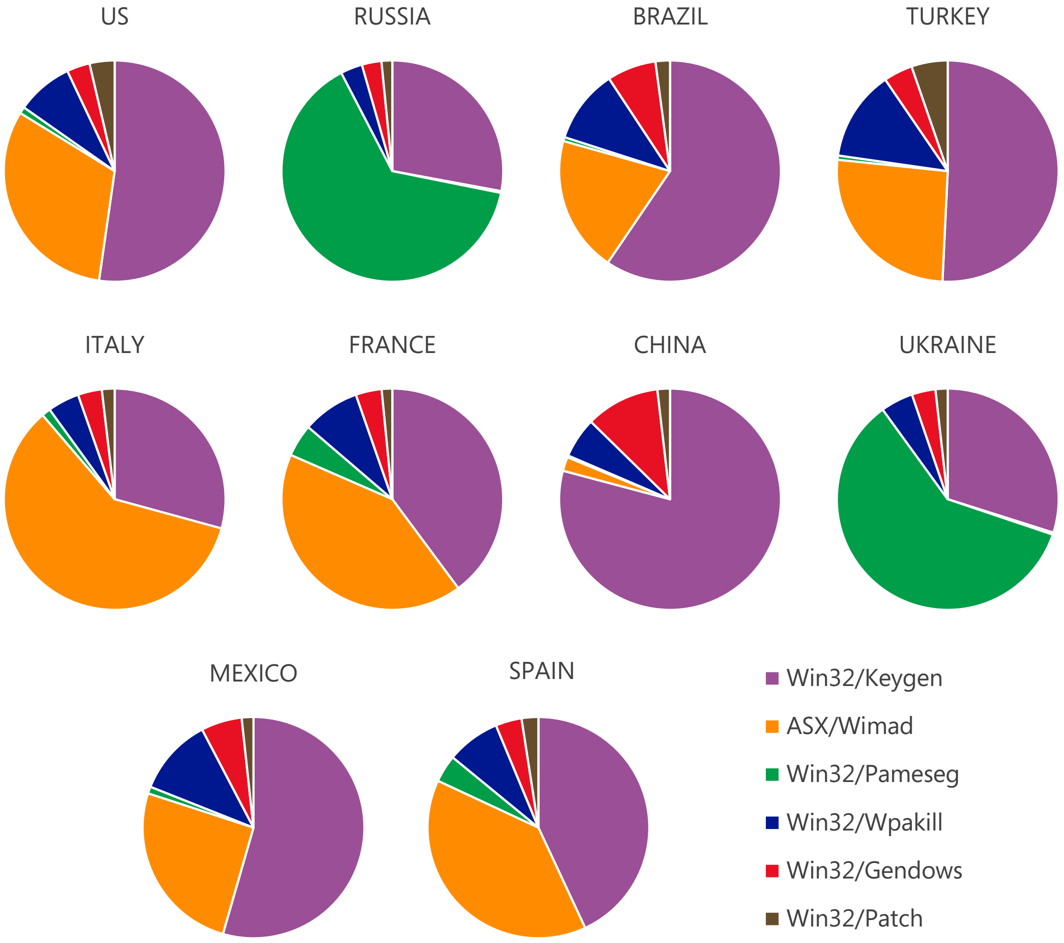
services that generated the telemetry used for this analysis.

[4] See support.microsoft.com/kb/971029 for more information about these changes.

Regional variations

Detections of the indicator families described in this section vary between different countries and regions. In Russia, Pameseg is detected far more often than the others; in some other locations, such as Italy and France, Wimad is in the top position. Figure 2 illustrates how these families are detected in different proportions in several different locations.

Figure 2. Relative detections of the indicator families discussed in this section in the 10 countries/regions with the most detections in 2Q12



Guidance: Defending against supply chain threats

Organizations and IT departments can use various processes and technological solutions to minimize the risk they face from malware transmitted through unsecure supply chains. Processes include the following:

- Create policies that state what constitutes acceptable and unacceptable downloading and use of third-party tools and media. Institute policies that govern the download and execution of music, movies, and game media. Create and enforce disciplinary actions for repeat policy offenders.
- Block peer-to-peer (P2P) applications from communicating into or out of the organization's internal network.
- Ensure that all new hardware is purchased by an internal procurement team. Procurement processes might include formatting computers and devices upon receipt and reinstalling the operating systems from known good images. Such images should include antimalware software, intrusion detection tools, software firewalls, monitoring and reporting tools, and other security software, all of which should be enabled by default.

Technology solutions to implement include the following:

- Use the AppLocker feature in Windows to create blacklists for potentially unsafe applications, programs, and scripts on client computers.
- On proxy servers, implement rules to block known malicious websites as well as other websites that violate the organization's acceptable media usage policy for content such as music, movies, games, shopping, pornography, and so on.
- Regularly update the organization's hardware and software standards, and limit the amount of old hardware and software. A 64-bit computer running Windows 7 and Internet Explorer 9, for example, is inherently more secure than a 32-bit computer running Windows XP and Internet Explorer 6 because of technologies such as ASLR, DEP, and SmartScreen Filter.

Vendors should use code signing and digital rights management to ensure customers can trust and confirm the authenticity of downloads.

Individual users can protect themselves by running antimalware software from a reputable vendor and keeping it up to date, and by only downloading software and content from trustworthy sources. Software updates and free software should only be obtained from the original vendors or from known, reputable sources. Using Internet Explorer with SmartScreen Filter enabled can help provide protection from malicious downloads.

Worldwide threat assessment

Vulnerabilities

Vulnerabilities are weaknesses in software that enable an attacker to compromise the integrity, availability, or confidentiality of the software or the data that it processes. Some of the worst vulnerabilities allow attackers to exploit the compromised system by causing it to run malicious code without the user's knowledge.

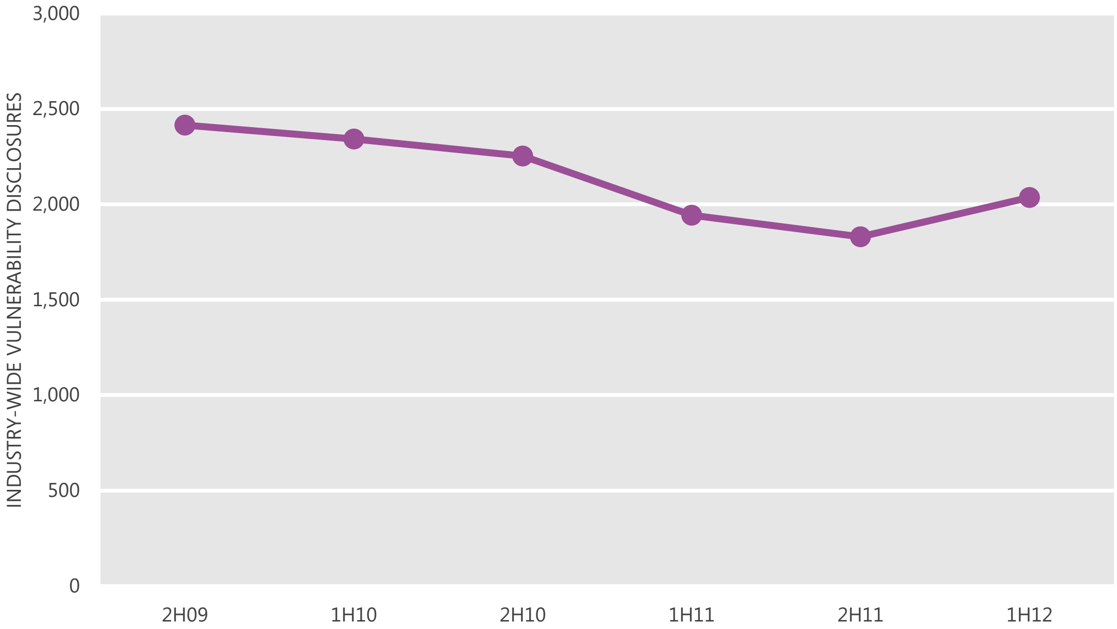
Industry-wide vulnerability disclosures

A *disclosure*, as the term is used in the *Microsoft Security Intelligence Report*, is the revelation of a software vulnerability to the public at large. Disclosures can come from a variety of sources, including the software vendor, security software vendors, independent security researchers, and even malware creators.

The information in this section is compiled from vulnerability disclosure data that is published in the National Vulnerability Database (nvd.nist.gov), the US government repository of standards-based vulnerability management data. It represents all disclosures that have a published CVE (Common Vulnerabilities and Exposures) identifier.

Figure 3 illustrates the number of vulnerability disclosures across the software industry for each half-year period since 2H09. (See "About this report" on page vi for an explanation of the reporting period nomenclature used in this report.)

Figure 3. Industry-wide vulnerability disclosures, 2H09–1H12

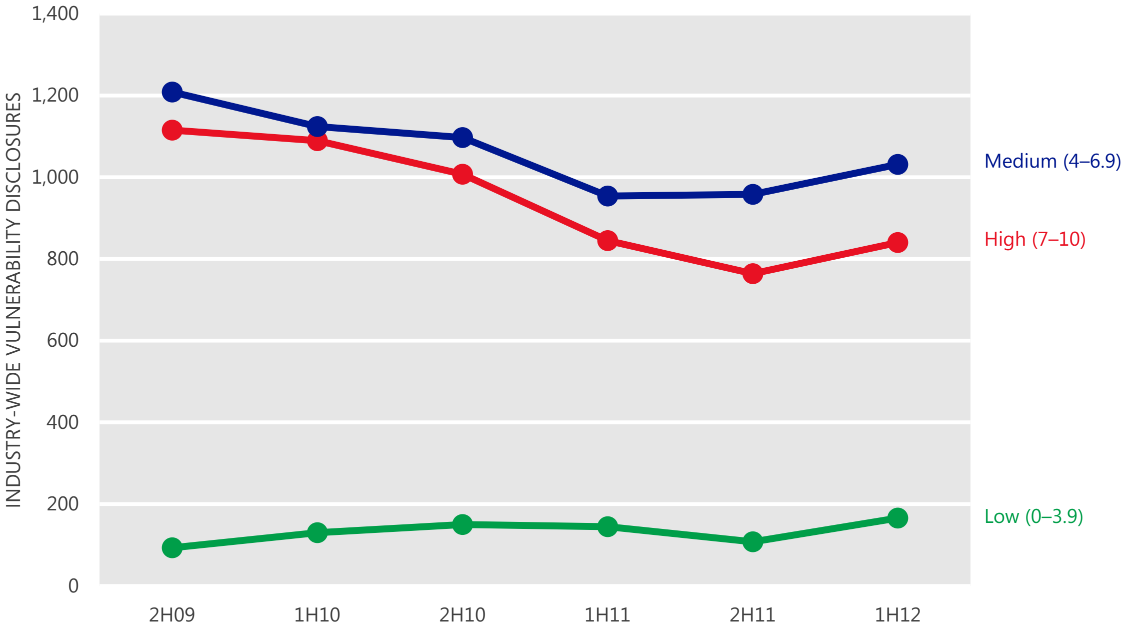


- Vulnerability disclosures across the industry in 1H12 were up 11.3 percent from 2H11, and 4.8 percent from 1H11, mostly because of an increase in application vulnerability disclosures. (See "Operating system, browser, and application vulnerabilities" on page 17 for more information.)
- This increase reverses a trend of small declines in every six-month period from 2H09 to 2H11.

Vulnerability severity

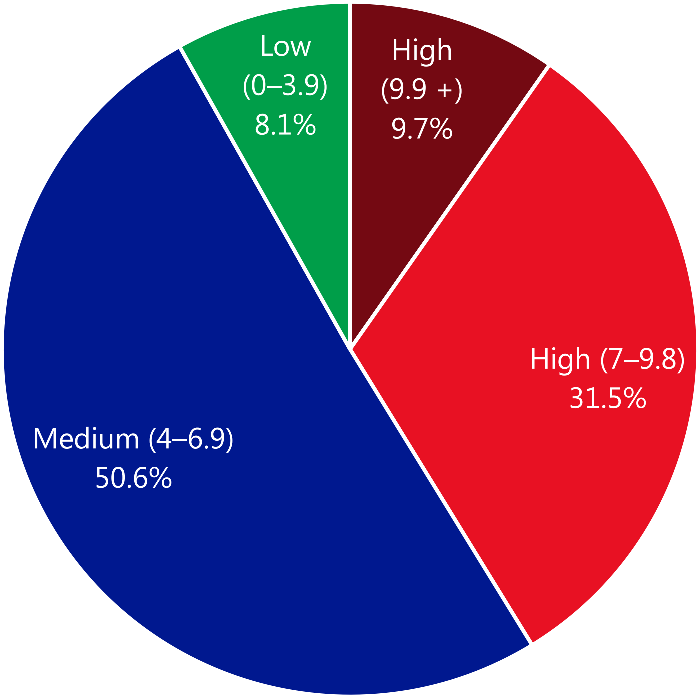
The Common Vulnerability Scoring System (CVSS) is a standardized, platform-independent scoring system for rating IT vulnerabilities. The CVSS base metric assigns a numeric value between 0 and 10 to vulnerabilities according to severity, with higher scores representing greater severity. (See Vulnerability Severity at the *Microsoft Security Intelligence Report* website for more information.)

Figure 4. Industry-wide vulnerability disclosures by severity, 2H09–1H12



- Vulnerability disclosures in each of the three CVSS severity classifications rose by a roughly similar amount, as shown in Figure 4.
- Medium-severity vulnerabilities again accounted for the largest number of disclosures at 1,031, a 7.6 percent increase from 2H11.
- High-severity vulnerabilities increased 9.9 percent from 2H11, but remained slightly below their 1H11 level. They accounted for a lower percentage of total vulnerabilities than at any time since 2H08.
- Low-severity vulnerabilities increased 53.7 percent from 2H11 and accounted for a larger share of total vulnerabilities than at any time since before 2H09.
- Mitigating the most severe vulnerabilities first is a security best practice. High-severity vulnerabilities that scored 9.9 or greater represent 9.7 percent of all vulnerabilities disclosed in 1H12, as Figure 5 illustrates. This figure was a slight increase from 2H11, when vulnerabilities scoring 9.9 percent or greater accounted for 9.6 percent of all vulnerabilities.

Figure 5. Industry-wide vulnerability disclosures in 1H12, by severity

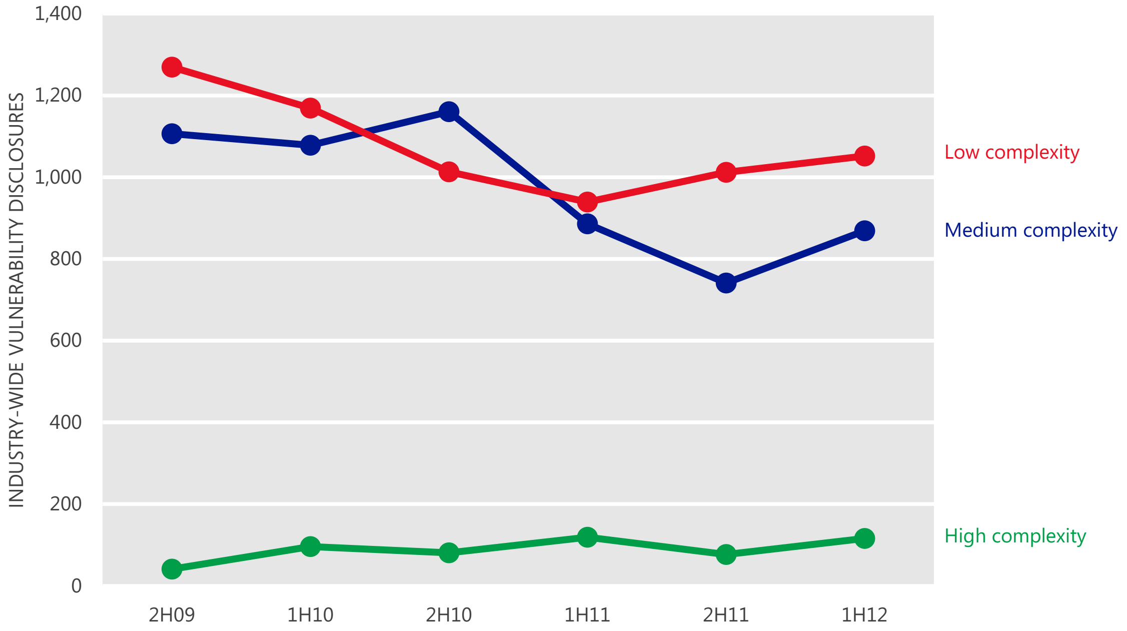


Vulnerability complexity

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.

The CVSS assigns each vulnerability a complexity ranking of Low, Medium, or High. (See Vulnerability Complexity at the *Microsoft Security Intelligence Report* website for more information about the CVSS complexity ranking system.) Figure 6 shows complexity trends for vulnerabilities disclosed since 2H09. Note that Low complexity in Figure 6 indicates greater risk, just as High severity indicates greater risk in Figure 4.

Figure 6. Industry-wide vulnerability disclosures by access complexity, 2H09–1H12



- A total of 1,052 Low-complexity vulnerabilities—those that are the easiest to exploit—were disclosed in 1H12, more than in any period since 1H10. However, as a percentage of the total, low-complexity vulnerabilities accounted for 51.6 percent of all disclosures in 1H12, down from 55.3 percent in 2H11.
- After decreasing significantly for two periods, Medium-complexity vulnerability disclosures increased to 42.7 percent of all disclosures in 1H12, up from 40.5 percent in 2H11.
- High-complexity vulnerability disclosures increased slightly to 116 in 1H12, up from 77 in 2H11. Disclosures of High-complexity vulnerabilities have been stable or slightly increasing over the past several years, but still only account for 5.7 percent of all vulnerabilities disclosed in 1H12.

Operating system, browser, and application vulnerabilities

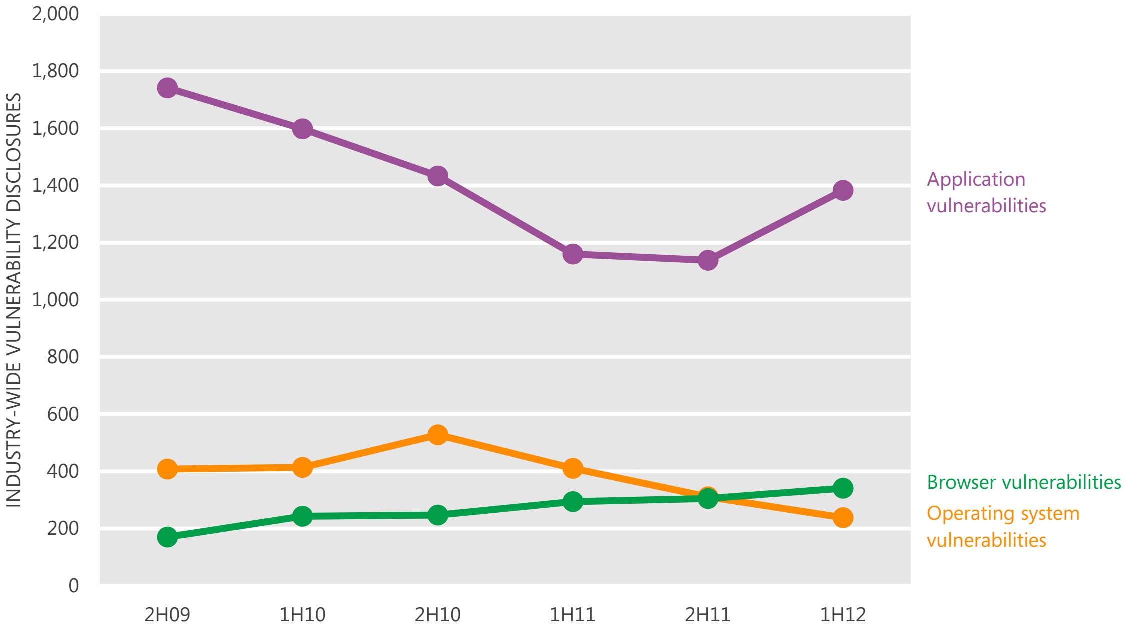
Comparing operating system vulnerabilities to non-operating system vulnerabilities requires determining whether a particular program or component should be considered part of an operating system. This determination is not always simple and straightforward, given the componentized nature of modern operating systems. Some programs (media players, for example) ship by default with some operating system software but can also be downloaded from the software vendor's website and installed individually. Linux distributions, in particular, are often assembled from components developed by different teams, many of which provide crucial operating functions, such as a graphical user interface (GUI) or Internet browsing.

To facilitate analysis of operating system and browser vulnerabilities, the *Microsoft Security Intelligence Report* distinguishes among three different kinds of vulnerabilities:

- *Operating system vulnerabilities* are those that affect the Linux kernel; or that affect components that ship with an operating system produced by Microsoft, Apple, or a proprietary Unix vendor, and are defined as part of the operating system by the vendor, except as described in the next paragraph.
- *Browser vulnerabilities* are those that affect components defined as part of a web browser, including web browsers that ship with operating systems, such as Internet Explorer and Apple's Safari, along with third-party browsers, such as Mozilla Firefox and Google Chrome.
- *Application vulnerabilities* are those that affect all other components, including components published by operating system vendors and other vendors. Vulnerabilities in open source components that may ship with Linux distributions (such as the X Window System, the GNOME desktop environment, GIMP, and others) are considered application vulnerabilities.

Figure 7 shows industry-wide vulnerabilities for operating systems, browsers, and applications since 2H09.

Figure 7. Industry-wide operating system, browser, and application vulnerabilities, 2H09–1H12

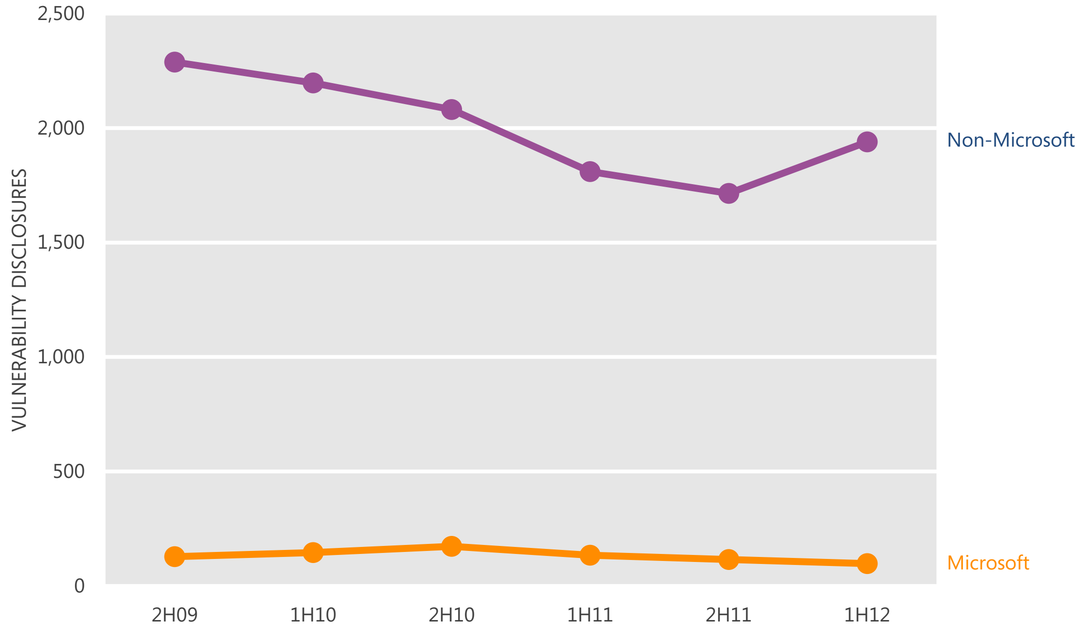


- After a steady decrease of several periods, application vulnerabilities increased significantly in 1H12, representing over 70 percent of all disclosures for the period.
- Browser and operating system vulnerabilities, which were nearly equal in 2H11, have switched places compared to previous periods. Operating system vulnerabilities dropped to the lowest level since 2003, while vulnerabilities in web browsers continue a multi-year trend upwards.

Microsoft vulnerability disclosures

Figure 8 charts vulnerability disclosures for Microsoft and non-Microsoft products since 2H09.

Figure 8. Vulnerability disclosures for Microsoft and non-Microsoft products, 2H09–1H12



- Although industry-wide vulnerability disclosures increased in 1H12, disclosures of vulnerabilities in Microsoft products continued to decrease slightly, accounting for 4.8 percent of all disclosures during the period, down from 6.3 percent in 2H11.
- Vulnerability disclosures for Microsoft products have decreased by 56.1 percent since 2H10.

Guidance: Developing secure software

The Security Development Lifecycle (www.microsoft.com/sdl) is a software development methodology that incorporates security and privacy best practices throughout all phases of the development process with the goal of protecting software users. Using such a methodology can help reduce vulnerabilities in the software and help manage vulnerabilities that might be found after deployment. (For more in-depth information about the SDL and other techniques developers can use to secure their software, see Protecting Your Software in the "Managing Risk" section of the *Microsoft Security Intelligence Report* website.)

Exploits

An *exploit* is malicious code that takes advantage of software vulnerabilities to infect, disrupt, or take control of a computer without the user's consent and usually without the user's knowledge. Exploits target vulnerabilities in operating systems, web browsers, applications, or software components that are installed on the computer. In some scenarios, targeted components are add-ons that are pre-installed by the computer manufacturer before the computer is sold. A user may not even use the vulnerable add-on or be aware that it is installed. Some software has no facility for updating itself, so even if the software vendor publishes an update that fixes the vulnerability, the user may not know that the update is available or how to obtain it and therefore remains vulnerable to attack.[5]

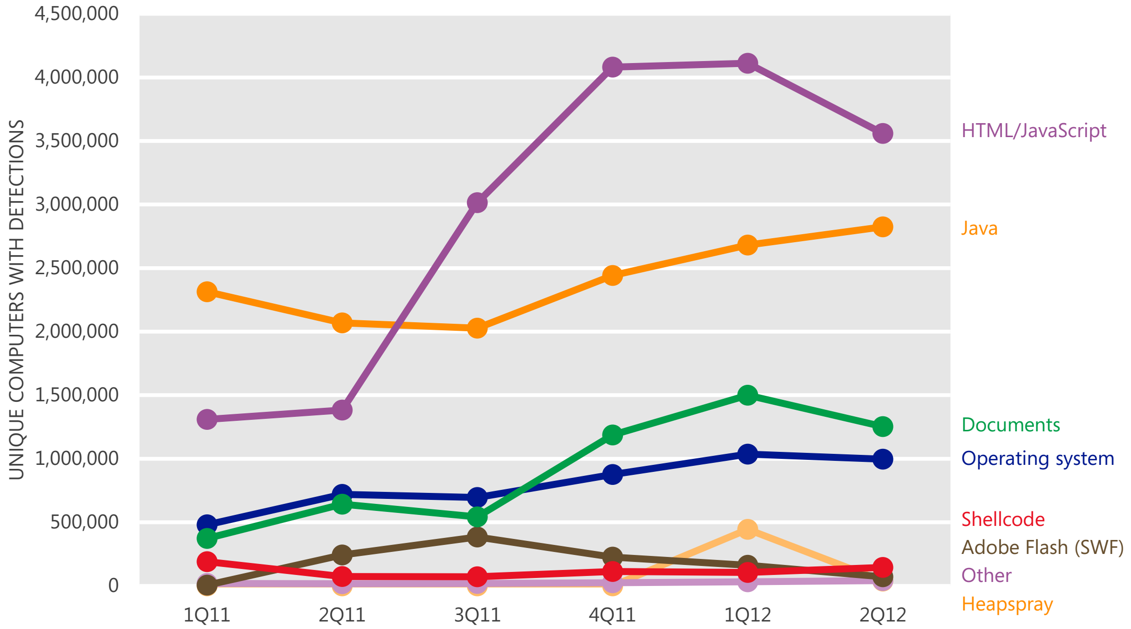
Software vulnerabilities are enumerated and documented in the Common Vulnerabilities and Exposures (CVE) list (cve.mitre.org), a standardized repository of vulnerability information. Here and throughout this report, exploits are labeled with the CVE identifier that pertains to the affected vulnerability, if applicable. In addition, exploits that affect vulnerabilities in Microsoft software are labeled with the Microsoft Security Bulletin number that pertains to the vulnerability, if applicable.[6]

Figure 9 shows the prevalence of different types of exploits detected by Microsoft antimalware products each quarter from 1Q11 to 2Q12, by number of unique computers affected. (See "Appendix B: Data sources" on page 115 for more information about the products and services that provided data for this report.)

[5] See the Microsoft Security Update Guide at www.microsoft.com/security/msrc/whatwedo/securityguide.aspx for guidance to help protect your IT infrastructure while creating a safer, more secure computing and Internet environment.

[6] See technet.microsoft.com/security/bulletin to search and read Microsoft Security Bulletins.

Figure 9. Unique computers reporting different types of exploits, 1Q11–2Q12



- The number of computers reporting exploits delivered through HTML or JavaScript remained high during the first half of 2012, primarily driven by the continued prevalence of Blacole, the most commonly detected exploit family in 1H12. (More information about the multiplatform Blacole family is provided in the next section.)
- Java exploits, the second most common type of exploit detected in 1H12, increased throughout the period, driven by increased detection of exploits for CVE-2012-0507 and CVE-2011-3544.
- Exploits that target vulnerabilities in document readers and editors were the third most commonly detected type of exploit during 1H12, primarily because of detections of exploits that target older versions of Adobe Reader that are not up-to-date on the latest security updates.

Exploit families

Figure 10 lists the exploit-related families that were detected most often during the first half of 2012.

Figure 10. Top exploit families detected by Microsoft antimalware products in 1H12, by number of unique computers with detections, shaded according to relative prevalence

Exploit family	Platform or technology	3Q11	4Q11	1Q12	2Q12
Blacole	HTML/JavaScript	1,054,045	2,535,171	3,154,826	2,793,451
CVE-2012-0507*	Java	–	–	205,613	1,494,074
Win32/Pdfjsc	Documents	491,036	921,325	1,430,448	1,217,348
Malicious IFrame	HTML/JavaScript	1,610,177	1,191,316	950,347	812,470
CVE-2010-0840*	Java	1,527,000	1,446,271	1,254,553	810,254
CVE-2011-3544	Java	–	331,231	1,358,266	803,053
CVE-2010-2568 (MS10-046)	Operating System	517,322	656,922	726,797	783,013
JS/Phoex	Java	–	–	274,811	232,773
CVE-2008-5353	Java	335,259	537,807	295,515	215,593
ShellCode	Shell code	71,729	112,399	105,479	145,352

* This vulnerability is also used by the Blacole kit; the totals given here for this vulnerability exclude Blacole detections.

- Blacole, a family of exploits used by the so-called Blackhole exploit kit to deliver malicious software through infected webpages, was the most commonly detected exploit family in the first half of 2012 by a large margin. Prospective attackers buy or rent the Blacole kit on hacker forums and through other illegitimate outlets. It consists of a collection of malicious webpages that contain exploits for vulnerabilities in versions of Adobe Flash Player, Adobe Reader, Microsoft Data Access Components (MDAC), the Oracle Java Runtime Environment (JRE), and other popular products and components. When the attacker loads the Blacole kit on a malicious or compromised web server, visitors who don't have the appropriate security updates installed are at risk of infection through a drive-by download attack. (See page 89 for more information about drive-by download attacks.)

  Figure 11 lists some of the vulnerabilities targeted by Blacole exploits during the first half of 2012:

Figure 11. Specific vulnerabilities targeted by the Blacole exploit kit in 1Q12 and 2Q12, by number of unique computers reporting detections of each one

CVE identifier	Class	1Q12	2Q12
CVE-2010-1885	Operating system	1,826,343	1,947,770
CVE-2012-0507	Java	306,939	1,845,862
CVE-2011-2110	Adobe Flash (SWF)	1,843,382	1,809,242
CVE-2007-5659	Documents	616,436	1,771,008
CVE-2009-0927	Documents	616,436	1,771,008
CVE-2006-0003	Operating system	445,055	1,761,074
CVE-2009-4324	Documents	445,045	1,760,955
CVE-2008-2992	Documents	445,045	1,760,955
CVE-2011-3544	Java	2,055,592	1,756,129
CVE-2011-0611	Adobe Flash (SWF)	168,072	1,401,569
CVE-2010-0188	Documents	441,858	737,296
CVE-2012-1723	Java	255,095	687,933
CVE-2010-0840	Java	940,249	609,116
CVE-2012-1889	Operating system	255,095	605,028
CVE-2010-0886	Java	416,120	521,590
CVE-2010-4452	Java	406,106	516,432

- All of the vulnerabilities listed in Figure 11 were addressed by security updates from the affected vendors between 2006 and 2012.

- The most commonly detected Blacole exploits during both quarters targeted CVE-2010-1885, a vulnerability that affects the Windows Help and Support Center in Windows XP and Windows Server 2003. Microsoft issued Security Bulletin MS10-042 in July 2010 to address this issue.

- CVE-2012-0507, a vulnerability in the Oracle Java Runtime Environment (JRE), was added to the Blacole kit in late March of 2012 and accounted for the second highest number of exploits attributed to the kit in 2Q12. More information about this exploit is available later in this section.

- CVE-2011-2110, a vulnerability in Adobe Flash Player, accounted for the second largest number of Blacole exploits detected in 1Q12 and the third largest number in 2Q12. Adobe released Security Bulletin APSB11-18 in June 2011 to address the issue.

  - Blacole exploitation of CVE-2011-3544, a vulnerability in the Java Runtime Environment, is decreasing, as Blacole authors have shifted their focus to newer exploits. It accounted for the largest number of Blacole exploits detected in 1Q12, but fell 14.6 percent to ninth place in 2Q12. Oracle released a security update in October 2011 to address the issue.

  For more information about Blacole, see the entry "The Rise of the 'Blackhole' Exploit Kit: The Importance of Keeping All Software Up To Date" on the Microsoft Security Blog (blogs.technet.com/security), as well as the following entries in the MMPC blog at blogs.technet.com/mmpc:

  - Get gamed and rue the day (October 25, 2011)
  - Disorderly conduct: localized malware impersonates the police (December 19, 2011)
  - Plenty to complain about with faux BBB spam (January 12, 2012)

- Exploitation of CVE-2012-0507 by families other than Blacole accounted for the second largest number of exploit attempts detected in 2Q12. This vulnerability allows an unsigned Java applet to gain elevated permissions and potentially have unrestricted access to a host system outside its sandbox environment. Oracle released a security update in February 2012 to address the issue.

  The CVE-2012-0507 vulnerability is a

Defending against Blacole exploits

The Blacole exploit kit targets a large number of exploits in web browsers and browser plug-ins in an effort to infect vulnerable computers through drive-by download attacks. Effectively defending against Blacole exploits can be challenging for IT departments and individual users.

Many antimalware solutions can block the Blacole kit directly when it is detected, before any of the exploits included in the kit have a chance to work. Using an antimalware solution from a reputable provider and keeping it up to date provides some protection against exploitation even when vulnerable software is installed. For better protection, ensure that all of the software in your environment is up to date and that security updates from all relevant vendors are installed quickly after they are published.

IT departments can increase their level of protection against Blacole exploits by using intrusion detection and prevention systems (IDS/IPS) to monitor for and block exploitation of the vulnerabilities targeted by the kit, including the ones listed in Figure 11 on page 24. Other vulnerabilities exploited by Blacole include CVE-2009-1671, CVE-2010-0842, CVE-2010-1423, CVE-2010-3552, and CVE-2012-4681. Configure your firewall to block any sites that have been compromised by the Blacole kit. Many enterprise firewall products use reputation services that can help automate the blocking of known malicious sites. If Blacole-related attacks are detected, use the detection telemetry to help you prioritize the deployment of security updates across your environment.

logic error that allows attackers to run code with the privileges of the current user, which means that an attacker can use it to perform reliable exploitation on other platforms that support the JRE, including Apple Mac OS X, Linux, VMWare, and others. On Mac OS X, CVE-2012-0507 exploits have been observed to install MacOS_X/Flashback, a trojan that gained notoriety in early 2012.

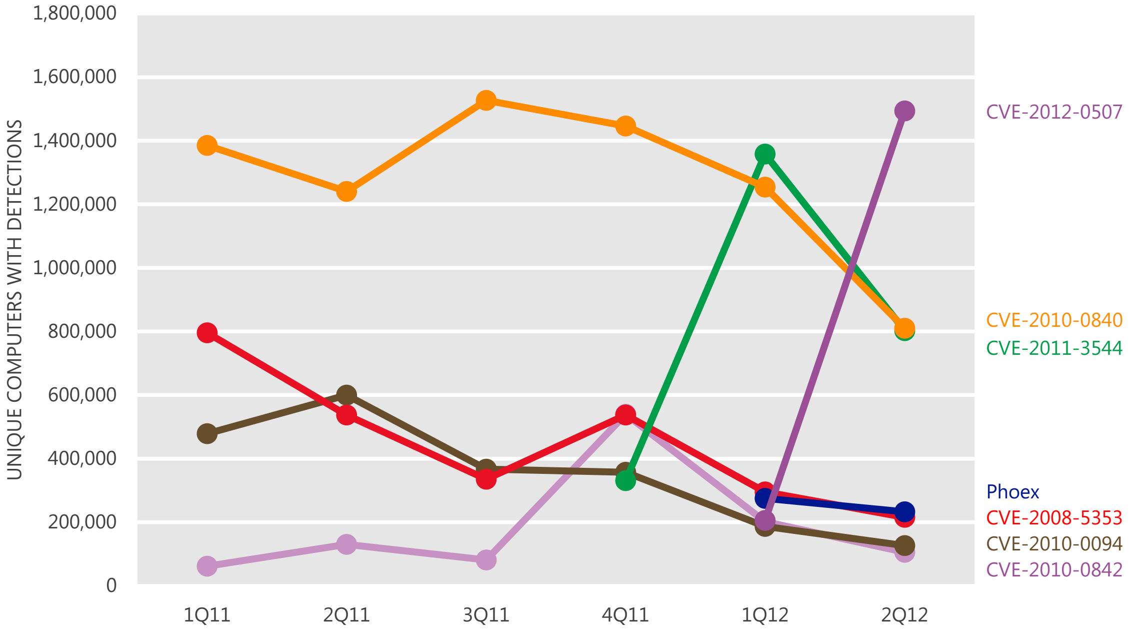
For more information about this vulnerability, see the entry "An interesting case of JRE sandbox breach (CVE-2012-0507)" (March 20, 2012) in the MMPC blog.

- Win32/Pdfjsc, a detection for specially crafted PDF files that exploit vulnerabilities in Adobe Reader and Adobe Acrobat, accounted for the second highest number of exploit detections in 1Q12 and the third highest in 2Q12. See page 29 for more information about Pdfjsc.

Java exploits

Figure 12 shows the prevalence of different Java exploits by quarter.

Figure 12. Trends for the top Java exploits detected and blocked by Microsoft antimalware products in 1H12

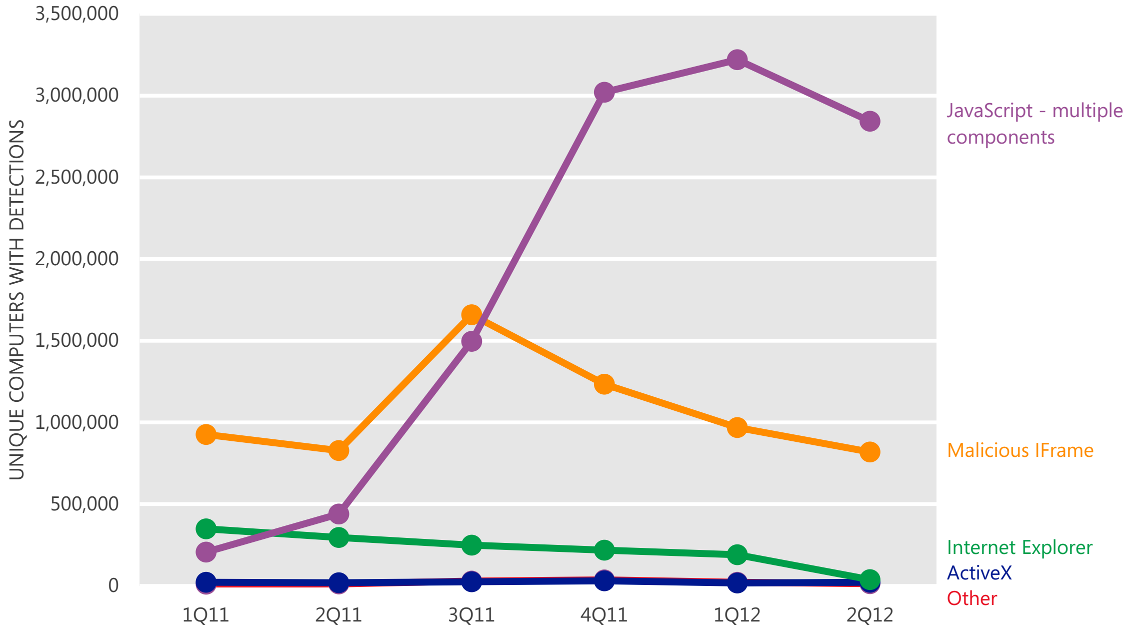


- CVE-2012-0507, the multiplatform JRE vulnerability added to the Blacole exploit kit in March 2012, accounted for the largest number of Java exploits detected and blocked in 2Q12.
- CVE-2011-3544 accounted for the largest number of Java exploits detected and blocked in 1Q12 and in the first half of the year as a whole, although it fell to third place in 2Q12. Like CVE-2012-0507, this JRE vulnerability can be relatively easily and reliably exploited across multiple platforms, which may be why the authors of the Blacole kit have decided to target them.
- The prominence of exploits for recently disclosed vulnerabilities such as CVE-2012-0507 (first disclosed in February 2012) and CVE-2011-3544 (first disclosed in October 2011) marks a change from previous periods, when the lists of the top exploits were dominated by much older vulnerabilities. Even relatively recent Java vulnerabilities such as these represent a potentially large target for attackers, given the generally low rate of adoption of recent Java security updates. (See “Security update adoption rates” on page 35 for more information.)
- CVE-2010-0840, a Java Runtime Environment (JRE) vulnerability first disclosed in March 2010 and addressed with an Oracle security update the same month, was the most commonly detected Java vulnerability throughout 2011 but fell to second place in both 1Q11 and 2Q11. This vulnerability was exploited by older versions of the Blacole exploit kit but has been removed from more recent releases, which probably helps explain the decline in detections.

HTML and JavaScript exploits

Figure 13 shows the prevalence of different types of HTML and JavaScript exploits during each of the six most recent quarters.

Figure 13. Types of HTML and JavaScript exploits detected and blocked by Microsoft antimalware products, 1Q11–2Q12

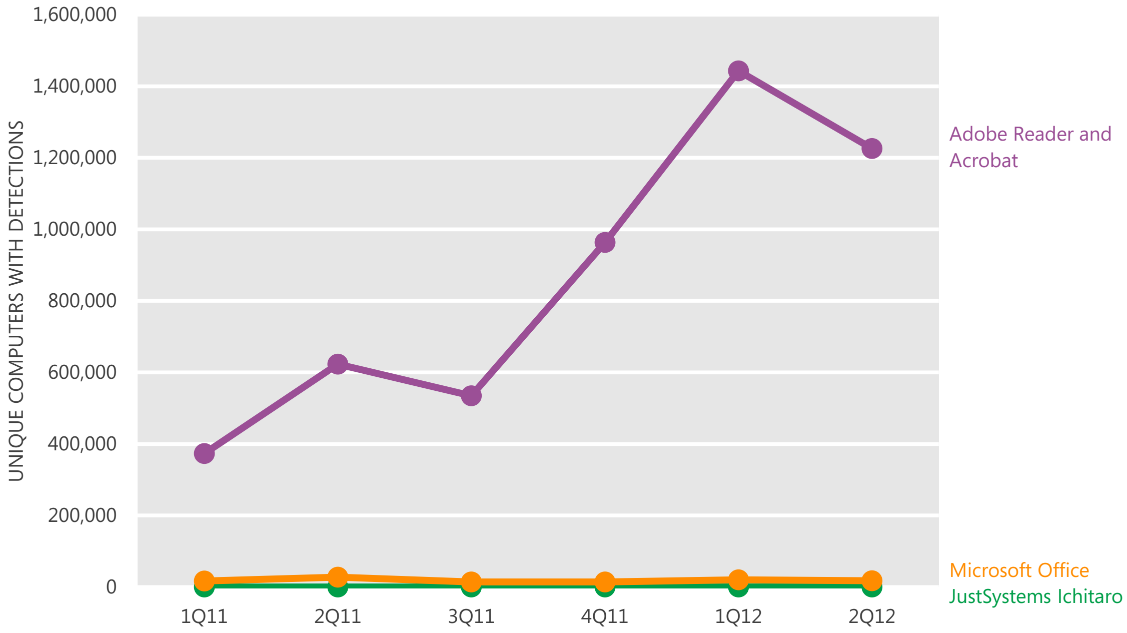


- The use of malicious JavaScript code designed to exploit one or more web-enabled technologies accounted for nearly three-fourths of HTML and JavaScript exploits detected in the first half of 2012, primarily because of the Blacole exploit kit. (See page 23 for more information about Blacole.)
- Detections of exploits that involve malicious HTML inline frames (IFrames) decreased slightly throughout the period, continuing a trend of moderate declines since 3Q11. These exploits are typically generic detections of inline frames that are embedded in webpages and link to other pages that host malicious web content. These malicious pages use a variety of techniques to exploit vulnerabilities in browsers and plug-ins; the only commonality is that the attacker uses an inline frame to deliver the exploits to users. The exact exploit delivered and detected by one of these signatures may be changed frequently.
- Detections for specific Windows Internet Explorer exploits decreased from more than 190,000 in 1Q12 to less than 40,000 unique computers in 2Q12. The decrease was primarily caused by fewer detections of CVE-2010-0806, a vulnerability in versions of Internet Explorer 6 and 7 that was addressed by Microsoft Security Bulletin MS10-018 in March 2010.
- ActiveX and other types of browser exploitation remained comparatively low.

Document parser exploits

Document parser exploits are exploits that target vulnerabilities in the way a document editing or viewing application processes, or parses, a particular file format. Figure 14 shows the prevalence of different types of document parser exploits during each of the six most recent quarters.

Figure 14. Types of document parser exploits detected and blocked by Microsoft antimalware products, 1Q11–2Q12



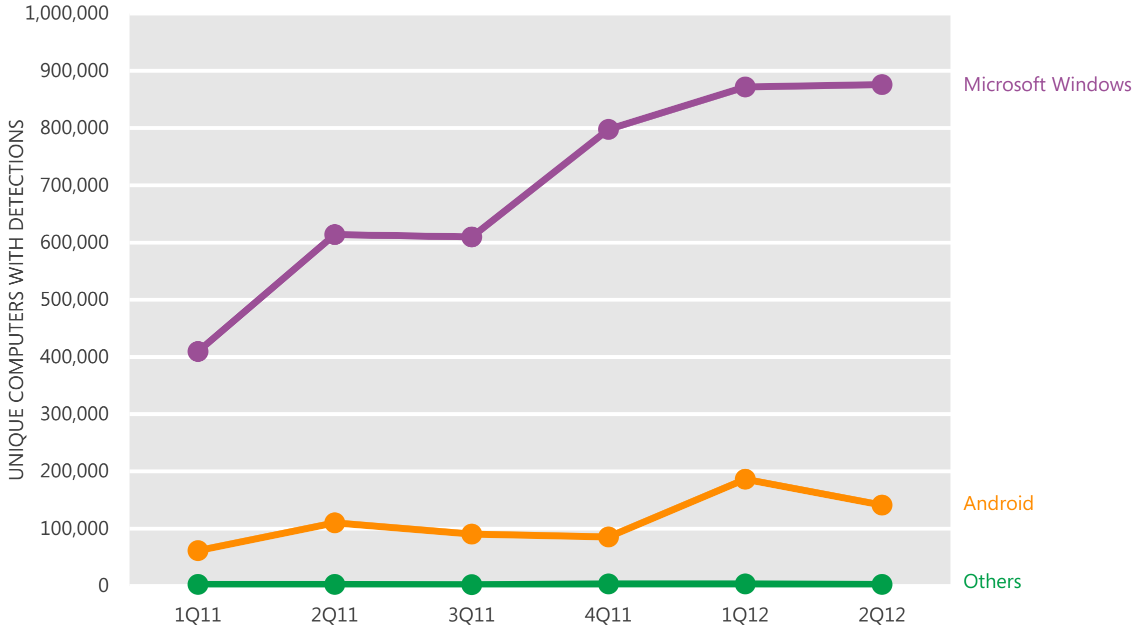
- Exploits that affect Adobe Reader and Adobe Acrobat accounted for most document format exploits detected in 1H12. Most of these exploits were detected as variants of the generic exploit family Win32/Pdfjsc.
    - During 1Q12, more than two-thirds of Pdfjsc activity (led by the variants Exploit:Win32/Pdfjsc.YN and Exploit:Win32/Pdfjsc.RF) targeted CVE-2010-0188, a vulnerability in versions of Adobe Reader 8 and 9 and Adobe Acrobat 8 and 9. Adobe published Security Bulletin APSB10-07 in February 2010 to address the issue. In the second quarter, malicious PDF documents with JavaScript that targeted multiple browser components became more common. Exploit:Win32/Pdfjsc.RM, which contains a malicious script detected as a variant of the JS/Mult generic family, was the most commonly detected Pdfjsc variant in 2Q12.

    - As with many of the exploits discussed in this section, Pdfjsc variants are also known to be associated with the Blacole exploit kit. (See page 23 for more information about Blacole.) In most cases, the vulnerabilities targeted by these exploits had been addressed several months or years earlier with security updates or new product versions.
    - Detections of Pdfjsc decreased from 1Q12 to 2Q12. As Microsoft detection signatures for the Blacole family have improved, a number of malware samples that previously would have been caught by the generic Pdfjsc signatures are being recognized as Blacole variants instead, which explains part of the decrease.
- Exploits that affect Microsoft Office and Ichitaro, a Japanese-language word processing application published by JustSystems, accounted for a small percentage of exploits detected during the period.

Operating system exploits

Although most operating system exploits detected by Microsoft security products are designed to affect the platforms on which the security products run, computer users sometimes download malicious or infected files that affect other operating systems. Figure 15 shows the prevalence of different exploits against operating system vulnerabilities that were detected and removed by Microsoft antimalware products during each of the past six quarters.

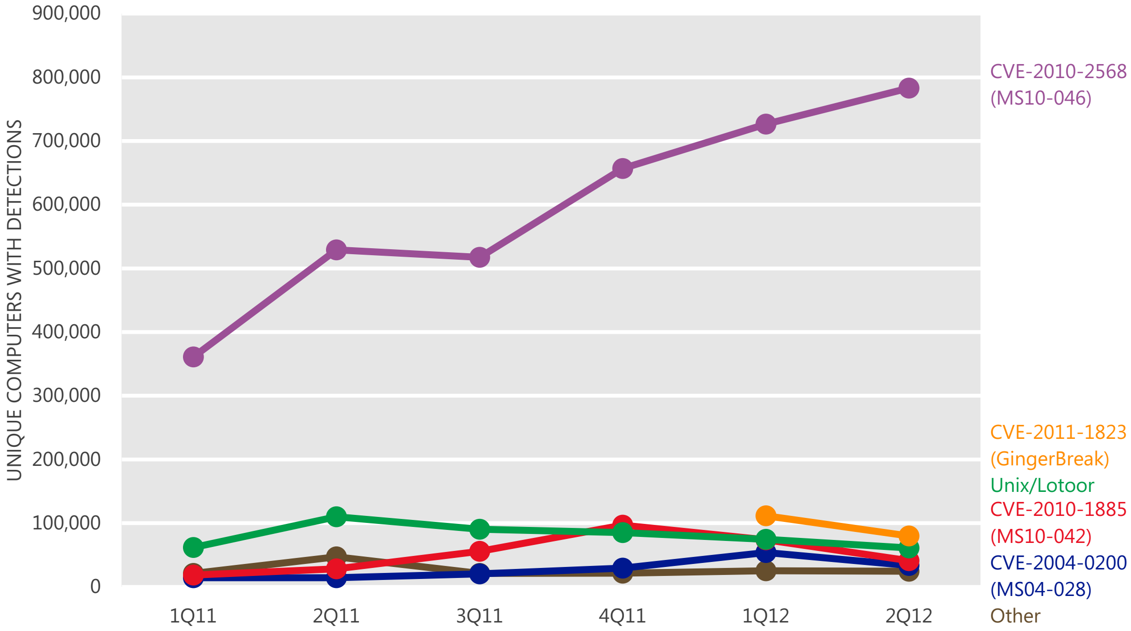
Figure 15. Exploits against operating system vulnerabilities detected and blocked by Microsoft antimalware products, 1Q11–2Q12



- Most exploit attempts targeting Windows that were detected in 1H12 targeted CVE-2010-2568, a vulnerability in Windows Shell addressed by Microsoft Security Bulletin MS10-046. See Figure 17 on page 33 for more information about these exploit attempts.
- Detections of exploits that affect the Android mobile operating system published by Google and the Open Handset Alliance increased in 1Q12 because of CVE-2011-1823, a vulnerability that can be used to obtain root permissions on a vulnerable device. Microsoft security products detect these threats when Android devices or storage cards are connected to computers running Windows, or when Android users download infected or malicious programs to their computers before transferring the software to their devices. See page 33 for more information about these exploits.

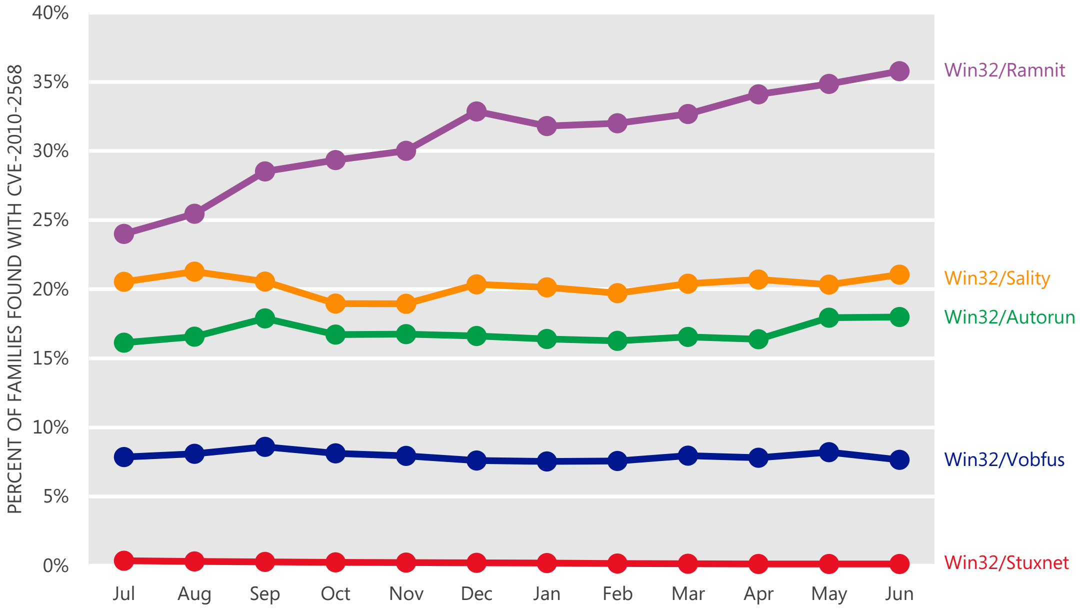
For another perspective on these exploits and others, Figure 16 shows trends for the individual exploits most commonly detected and blocked or removed during each of the past six quarters.

Figure 16. Individual operating system exploits detected and blocked by Microsoft antimalware products, 1Q11–2Q12, by number of unique computers exposed to the exploit



- Exploits that target CVE-2010-2568, a vulnerability in Windows Shell, accounted for more than 85 percent of Windows exploit detections in the first half of 2012. An attacker exploits CVE-2010-2568 by creating a malformed shortcut file that forces a vulnerable computer to load a malicious file when the shortcut icon is displayed in Windows Explorer. The vulnerability was first discovered being used by the malware family Win32/Stuxnet in mid-2010. It has since been exploited by a number of other malware families, many of which predated the disclosure of the vulnerability and were subsequently adapted to attempt to exploit it, as shown in Figure 17.

Figure 17. Families commonly found with CVE-2010-2568, July 2011–June 2012



Win32/Ramnit, Win32/Sality, Win32/Autorun, and Win32/Vobfus are all consistently detected alongside a file designed to exploit the CVE-2010-2568 vulnerability. All of the families are known to use this exploit to spread through removable drives. The increasing prevalence of Ramnit, a family of multicomponent malware that infects files and steals sensitive information, has been the primary cause of the increase in CVE-2010-2568 detections, as Figure 17 shows.

- Most detections that affect Android involve a pair of exploits that enable an attacker or other user to obtain root privileges on vulnerable Android devices. Device owners sometimes use such exploits intentionally to gain access to additional functionality (a practice often called rooting or jailbreaking), but these exploits can also be used by attackers to infect devices with malware that bypasses many of the typical security systems.
  - CVE-2011-1823 is sometimes called the GingerBreak vulnerability because of its use by a popular rooting application by that name (detected separately as Exploit:AndroidOS/GingerBreak). It is also used by AndroidOS/GingerMaster, a malicious program that can allow a remote attacker to gain access to the mobile device. GingerMaster may be bundled with clean applications, and includes an exploit for the CVE-

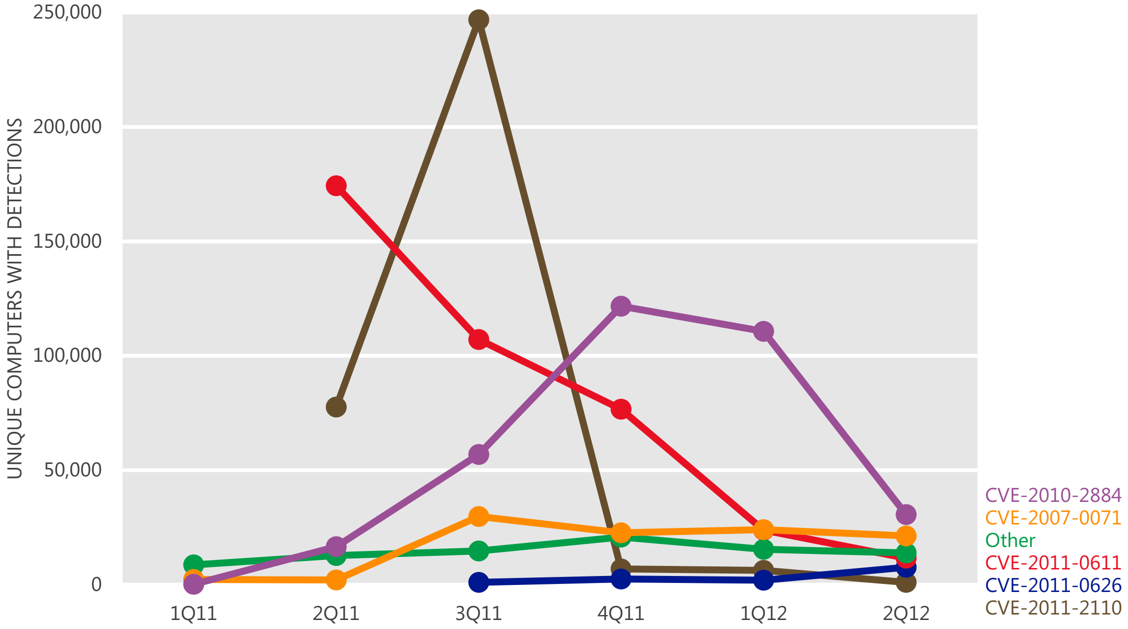
2011-1823 vulnerability disguised as an image file. Google published a source code update in May 2011 that addressed the vulnerability.

- Detections of Unix/Lotoor declined in 1H12, but continued to account for a significant minority of Android exploits. Lotoor is dropped by TrojanSpy:AndroidOS/DroidDream.A, a malicious program that often masquerades as a legitimate Android application and can allow a remote attacker to gain access to the device. Google published a source code update in March 2011 that addressed the vulnerability.

Adobe Flash Player exploits

Figure 18 shows the prevalence of different Adobe Flash Player exploits by quarter.

Figure 18. Adobe Flash Player exploits detected and blocked by Microsoft antimalware products, 1Q11–2Q12, by number of unique computers exposed to the exploit



- Following a surge in detections that peaked in 3Q11, detections of exploits that target vulnerabilities in Adobe Flash Player have decreased significantly in every subsequent quarter, with no single vulnerability accounting for more than 35,000 computers with detections by 2Q12.

- CVE-2010-2884 was discovered in the wild in September 2010 as a zero-day vulnerability, and Adobe released Security Bulletin APSB10-22 on September 20 to address the issue. Significant exploitation of the vulnerability peaked in 4Q11 and remained close to the peak in 1Q12, then declined by almost three-quarters in 2Q12. This decline is likely caused by more computers receiving the security update, combined with an overall saturation of exploitable targets.
- CVE-2007-0071 accounted for the second largest number of Adobe Flash Player exploitation attempts detected in 1Q12 and 2Q12. Adobe released Security Bulletin APSB08-11 on April 8, 2008 to address the issue. Detections of CVE-2007-0071 exploitation attempts during the first half of 2012 were likely caused by the inclusion of exploits for the vulnerability in exploit kits along with exploits for other, more recent Adobe Flash Player vulnerabilities.
- CVE-2011-0611 was discovered in April 2011 when it was observed being exploited in the wild; Adobe released Security Bulletin APSB11-07 on April 15 and Security Bulletin APSB11-08 on April 21 to address the issue. After peaking in 3Q11, detections of CVE-2011-0611 exploits declined to very low levels in the fourth quarter and remained low throughout the first half of 2012.

Security update adoption rates

As in previous volumes, this edition of the *Microsoft Security Intelligence Report* shows that most exploits detected by Microsoft antimalware products target vulnerabilities for which a security update existed at the time of the infection attempt. This fact implies that many computers do not have security updates installed, even for critical vulnerabilities that are being actively exploited. Using data gathered in October 2011 from thousands of computer users worldwide who have agreed to provide data to Microsoft for research purposes, Microsoft analyzed the distribution of missing security updates across a variety of dimensions.[7]

[7] For this study, Windows was considered to be updated if ntoskrnl.exe was updated and other applications were considered to be updated if the main application executable (.exe) was updated. Limitations in the data source precluded analyzing some common software, such as browser software: in some cases (as with Mozilla Firefox and Google Chrome) the browser manufacturer does not provide public documentation of the file version numbers used for security updates, and in others (as with Windows Internet Explorer) the .exe file version is not always incremented when a security update is installed.

Figure 19. Security update status of Windows and popular applications running on computers worldwide, October 2011

Security update status	Microsoft Windows	Microsoft Word	Adobe Reader	Oracle Java	Adobe Flash Player
Missing latest update*	34%	39%	60%	94%	70%
Missing three latest updates	16%	35%	46%	51%	44%

* As of October 2011. At the time, the most recent updates to Adobe Flash Player, Adobe Reader, and Java had been released within one month; the most recent Windows kernel update had been released 9 months prior; and the most recent updates to Word had been released one year prior.

- Windows was the product with the fewest out-of-date computers. 34 percent of computers analyzed were missing the most recently released Windows kernel update, and 16 percent were missing the three most recently released updates.
- Java was the product with the most out-of-date computers. 94 percent of computers analyzed were missing the most recently released Java update, and 51 percent were missing the three most recently released updates.
- Windows computers and those running Microsoft Word that were missing recent security updates were generally running out-of-support versions of the products, or their users had installed service packs but no additional security updates. The latter scenario was most common with Windows XP: 11 percent of the Windows XP computers that were analyzed were still running SP2 (Service Pack 2, which is no longer supported) or SP3 with no post-SP Windows security updates installed.
- Users of Oracle Java, Adobe Reader, and Adobe Flash who were missing security updates did not display a characteristic pattern as to which versions they were using, with approximately equal portions of the unpatched user base on almost every version released within the last several years.
- Adobe released a new version of the Adobe Reader Updater, the update mechanism used for Adobe Reader in April 2010.[8] Analysis found that:
  - 28 percent of Adobe Reader users were using versions released prior to the April 2010 Updater revision.
  - Of Adobe Reader users with the April 2010 Updater revision installed, only 44 percent were missing the latest security update. 12 percent were missing all three of the most recently released security updates.

[8] Another revised version was released in March 2012, after the data for this analysis was collected. Since this study was conducted, Adobe has enabled automatic updates by default for Adobe Reader and introduced a silent update mechanism for Flash Player.

- Seven percent of Adobe Reader users and nine percent of Microsoft Word users are using a major version of the application for which Adobe and Microsoft (respectively) are no longer creating security updates. Almost all of the computers analyzed were running currently supported major versions of Windows, including Windows XP and more recently released versions.
- Users of the latest Microsoft operating systems were significantly more likely than others to have the most recent security updates installed for most products. Windows 7 users were 20 percent more likely than Windows XP users to have the most recent Windows updates installed, 40 percent more likely to have Microsoft Word security updates installed, 37 percent more likely to have the most recent Adobe Reader updates installed, and 60 percent more likely to have the most recent Oracle Java updates installed. (Adobe Flash users' update status did not vary significantly with operating system version.)
- Windows Vista and Windows XP users with Automatic Updates enabled or who regularly visit Windows Update were more than twice as likely to have the latest Microsoft updates installed compared to those who don't.[9]
- In a separate analysis of malware data collected by the MMPC in January 2012, at least 36 percent of the malware that was analyzed contained functionality to disable Windows Update. Malware uses many techniques to accomplish this goal, including:
  - Changing registry settings
  - Modifying DNS to reroute traffic intended for Windows Update servers
  - Modifying the computer to use a Windows Server Update Services (WSUS) server controlled by the attacker
  - Tampering with last-checked timestamps so that Windows Update will not check for new updates
- No correlation was found between installation of an antivirus product and update status. It was hypothesized that some antimalware programs might interfere with update installation, but no evidence was found to support this hypothesis. Users should not avoid installing antimalware software out of concern about update problems.

[9] For the purposes of this analysis, a user was considered to be a Windows Update user if the automatic update utility (called Automatic Updates or Windows Update) was enabled on the computer, or if the user had recently visited the Windows Update website or Control Panel item. Windows 7 users could not be included in this analysis for technical reasons, but their update patterns are likely to be similar to those observed on other platforms.

- The data set did not allow Microsoft researchers to determine whether a correlation exists between software piracy and update status, or between a user’s Internet connection speed and update status.

To improve security update adoption rates, users are advised to:

- Avoid older software releases that are no longer supported by their publishers.
- Ensure all service packs are installed for Microsoft products.
- Ensure that automatic updaters, including Windows Automatic Updates, are enabled and functioning, particularly after cleaning a malware infection. Microsoft recommends configuring computers to use Microsoft Update instead of Windows Update to help ensure they receive timely security updates for Microsoft products.

Malware and potentially unwanted software

Except where specified, the information in this section was compiled from telemetry data that was generated from more than 600 million computers worldwide and some of the busiest services on the Internet. (See "Appendix B: Data sources" on page 115 for more information about the telemetry used in this report.)

Global infection rates

The telemetry data generated by Microsoft security products from computers whose administrators or users choose to opt in to provide data to Microsoft includes information about the location of the computer, as determined by IP geolocation. This data makes it possible to compare infection rates, patterns, and trends in different locations around the world.[10]

[10] For more information about this process, see the entry "Determining the Geolocation of Systems Infected with Malware" (November 15, 2011) on the Microsoft Security Blog (blogs.technet.com/security).

Figure 20. The locations with the most computers reporting detections and removals by Microsoft desktop antimalware products in 1H12

	Country/Region	1Q12	2Q12	Chg. 1Q to 2Q
1	United States	9,407,423	12,474,127	32.6% ▲
2	Brazil	3,715,163	3,333,429	-10.3% ▼
3	Korea	2,137,136	2,820,641	32.0% ▲
4	Russia	2,580,673	2,510,591	-2.7% ▼
5	China	1,889,392	2,000,576	5.9% ▲
6	Turkey	1,924,387	1,911,837	-0.7% ▼
7	France	1,677,242	1,555,522	-7.3% ▼
8	United Kingdom	1,648,801	1,509,488	-8.4% ▼
9	Germany	1,544,774	1,486,309	-3.8% ▼
10	Italy	1,361,043	1,341,317	-1.4% ▼

- In absolute terms, the locations with the most computers reporting detections tend to be ones with large populations and large numbers of computers.

- Detections in the United States increased 32.6 percent from 1Q12 to 2Q12, primarily because of increased detections of Win32/FakePAV, the most commonly detected rogue security software family in the world in 1H12. FakePAV accounted for more than 45 times as many detections in the United States in 2Q12 as in the previous quarter. (See “Rogue security software” beginning on page 57 for more information on FakePAV and similar families.) Increased detections of JS/IframeRef, Java/CVE-2012-0507, Win32/Keygen, and Win32/Autorun also contributed to the increase.

- Detections in Korea increased 32.0 percent from 1Q12 to 2Q12, primarily because of increased detection of the trojan downloader family Win32/Pluzoks. See page 43 for more information.

- Detections in Brazil declined 10.3 percent from 1Q12 to 2Q12, primarily because of decreased detections of the trojan downloader Win32/Banload and the trojan family Win32/Bancos. Bancos is a data-stealing trojan that primarily targets customers of Brazilian banks; it is frequently downloaded to the target computer by Banload. Fewer detections of the adware family JS/Pornpop also contributed to the decline.

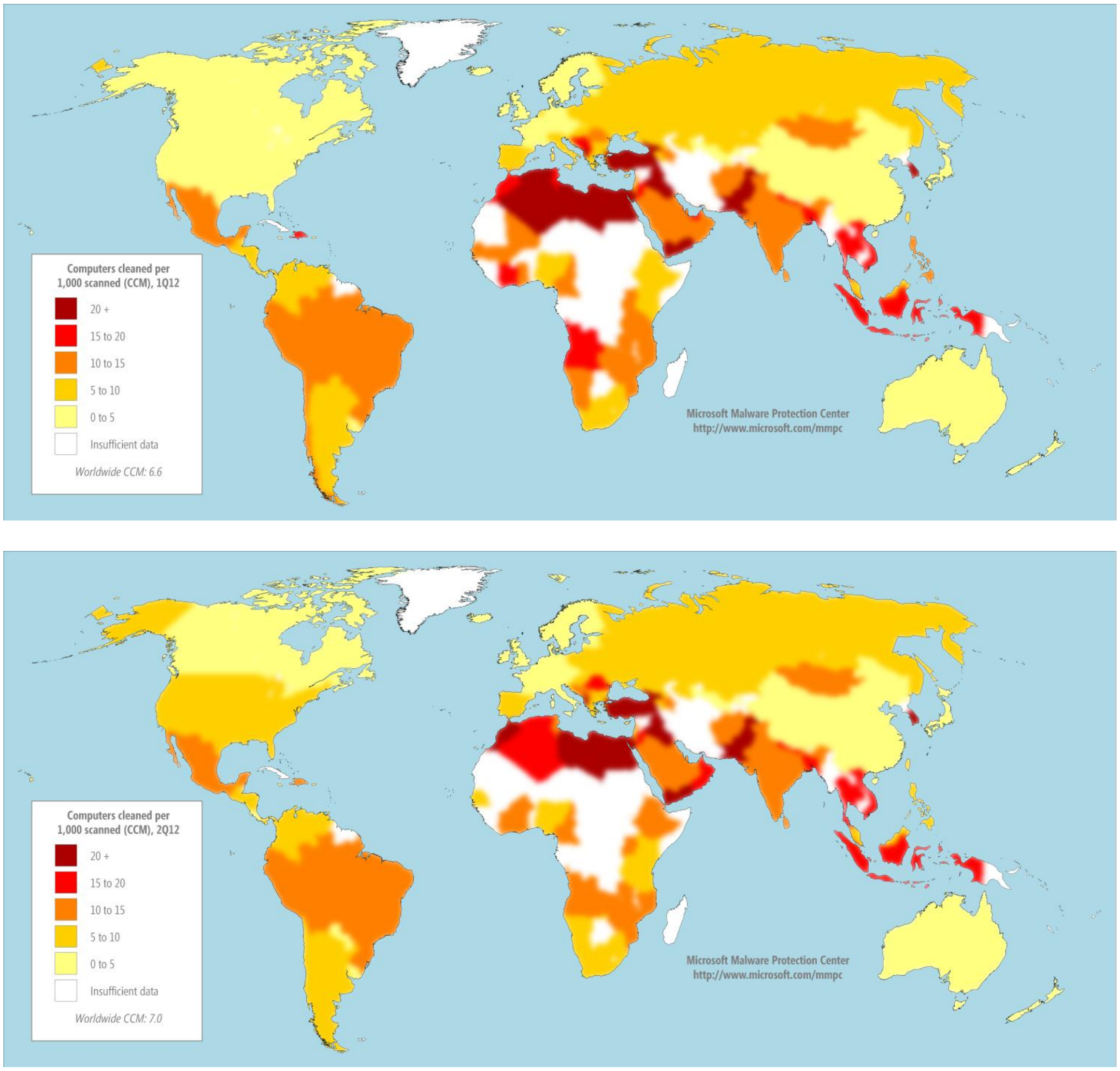
- Detections in the United Kingdom declined 8.4 percent from 1Q12 to 2Q12, primarily because of decreased detections of the adware families Pornpop and Win32/Hotbar, which decreased 28.3 and 32.7 percent, respectively.

Detections of JS/BlacoleRef, a trojan family associated with the Blacole exploit kit, also decreased in 2Q12 after increasing significantly from 4Q11 to 1Q12.

- Detections in France declined 7.3 percent from 1Q12 to 2Q12, primarily because of fewer detections of Hotbar and Blacole.
- Detections in China increased 5.9 percent from 1Q12 to 2Q12, primarily because of a new adware family, JS/Popupper.

For a different perspective on infection patterns worldwide, Figure 21 shows the infection rates in locations around the world in *computers cleaned per mille* (CCM), which represents the number of reported computers cleaned for every 1,000 executions of the Microsoft Malicious Software Removal Tool (MSRT). Normalizing the data this way makes it possible to compare malware infection rates of different locations without skewing the data because of differences in populations and install bases. See the *Microsoft Security Intelligence Report* website for more information about the CCM metric.

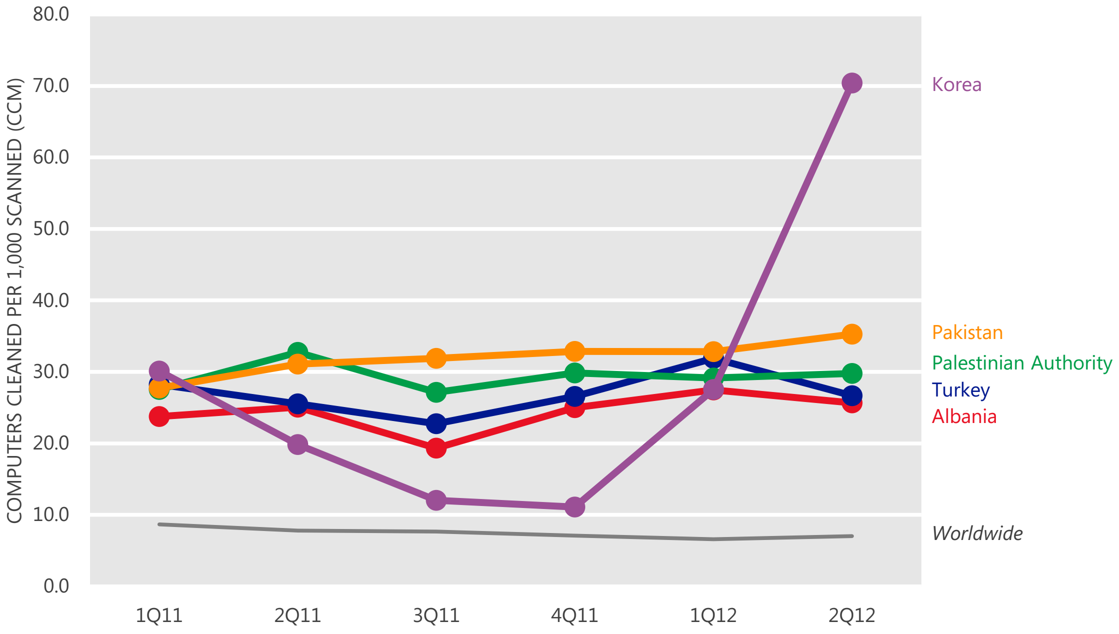
Figure 21. Infection rates by country/region in 1Q12 (top) and 2Q12 (bottom), by CCM


Computers cleaned per 1,000 scanned (CCM), 1Q12
20 +
15 to 20
10 to 15
5 to 10
0 to 5
Insufficient data
Worldwide CCM: 6.6
Microsoft Malware Protection Center
http://www.microsoft.com/mmpc
Computers cleaned per 1,000 scanned (CCM), 2Q12
20 +
15 to 20
10 to 15
5 to 10
0 to 5
Insufficient data
Worldwide CCM: 7.0
Microsoft Malware Protection Center
http://www.microsoft.com/mmpc

Detections and removals in individual countries/regions can vary significantly from quarter to quarter. Increases in the number of computers with detections can be caused not only by increased prevalence of malware in that location, but also by improvements in the ability of Microsoft antimalware solutions to detect malware. Large numbers of new antimalware product or tool installations in a location also typically increase the number of computers cleaned in that location.

The next three figures illustrate infection rate trends for specific locations around the world, relative to the trends for all locations with at least 100,000 MSRT executions each quarter in 1H12.

Figure 22. Trends for the five locations with the highest malware infection rates in 1H12, by CCM (100,000 MSRT executions minimum)



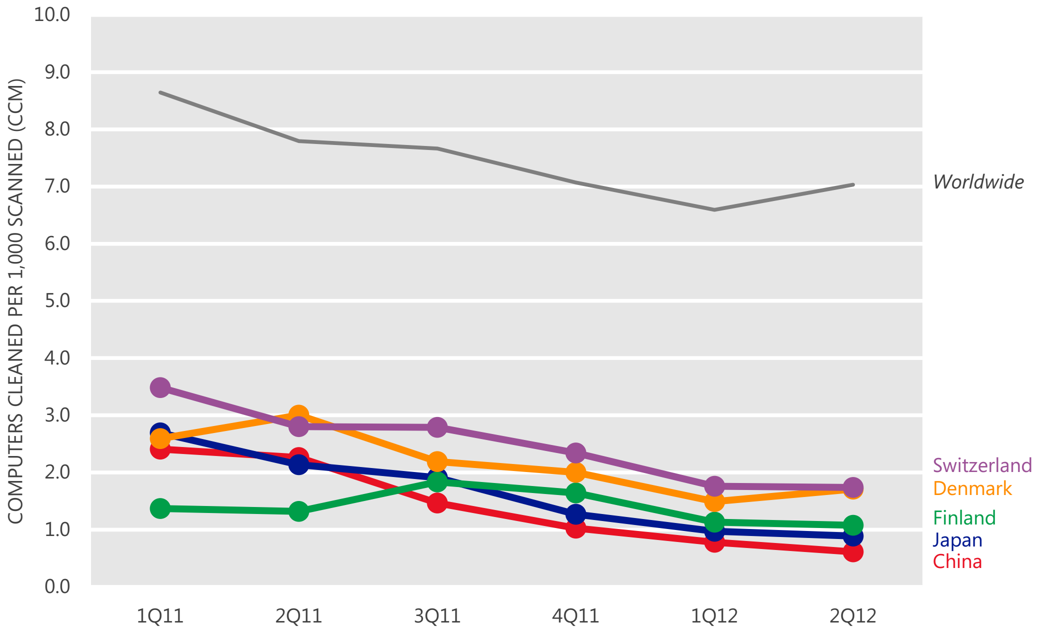
- Korea's CCM increase from 27.5 in 1Q12 to 70.4 in 2Q12 is one of the largest quarter-to-quarter increases ever reported for a large country or region in the *Microsoft Security Intelligence Report*. The change was primarily caused by increased detection of the trojan downloader family Win32/Pluzoks. When installed on an infected computer, Pluzoks attempts to connect to a web server in the .kr top-level domain (TLD) assigned to Korea, and the overall detection pattern for Pluzoks—93.3 percent of Pluzoks detections in 1H12 were located in Korea—suggests that distribution of the family has been strongly targeted. Detection signatures for Pluzoks were added to the MSRT in March 2012, which led to a large increase in the number of detections for the family.

Figure 23. Prevalent families in Korea in 2Q12, by infection rate (CCM)

Family	Most significant category	3Q11	4Q11	1Q12	2Q12
Win32/Pluzoks	Trojan Downloaders & Droppers	–	–	16.8	59.5
Win32/Rimecud	Worms	2.7	2.0	1.3	1.1
Win32/Yeltminky	Worms	–	–	0.2	1.1
Win32/Taterf	Worms	2.4	3.5	3.9	1.1

- Compared to Korea, the other four locations shown in Figure 22 displayed relatively consistent infection rates. Pakistan, the Palestinian territories, Turkey, and Albania were the four locations with the highest infection rates in 2H11, and ranked second through fifth in 1H12.

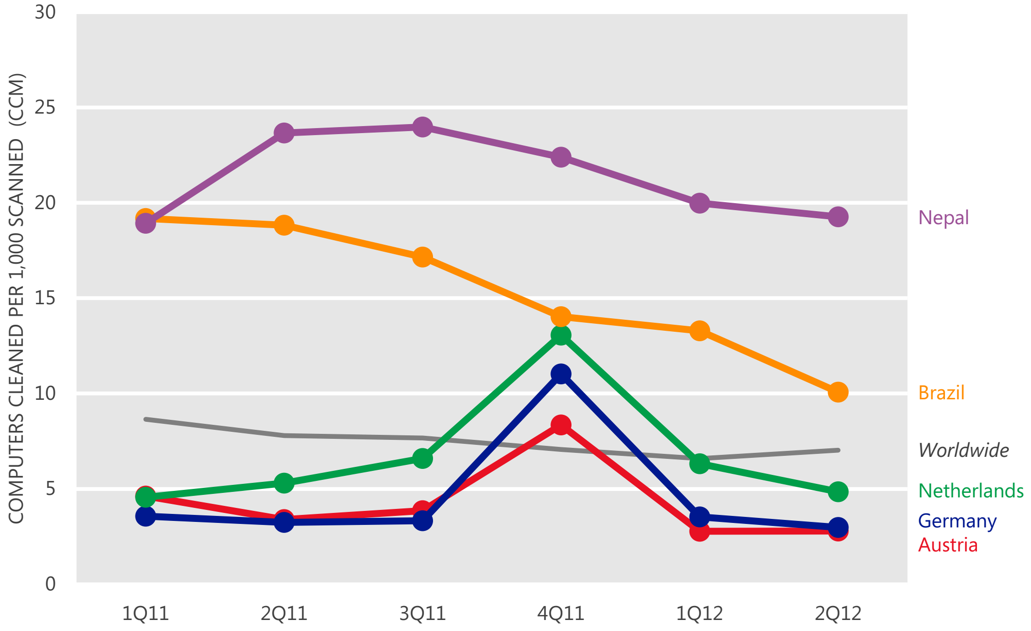
Figure 24. Trends for the five locations with the lowest infection rates in 1H12, by CCM (100,000 MSRT executions minimum per quarter)



- Trends for the locations with the lowest infection rates in 1H12 remained consistent with previous periods. Four of the five locations with the lowest infection rates in 1H12 were also on the list in 2H11, with Switzerland taking the place of Norway. All five had 2Q12 infection rates between 0.6 and 1.7, compared to the worldwide average of 7.0.

- Historically, Nordic countries such as Norway and Finland have had some of the lowest malware infection rates in the world. Japan also usually experiences a low infection rate.
- Although China is one of the locations with the lowest infection rates worldwide as measured by CCM, a number of factors that are unique to China are important to consider when assessing the state of computer security there. The malware ecosystem in China is dominated by a number of Chinese-language threats that are not prevalent anywhere else. The CCM figures are calculated based on telemetry data from the MSRT, which tends to target malware families that are prevalent globally. As a result, many of the more prevalent threats in China are not represented in the data used to calculate CCM. For a more in-depth perspective on the threat landscape in China, see the "Regional Threat Assessment" section of the *Microsoft Security Intelligence Report* website.

Figure 25. Trends for the five locations with the most significant infection rate improvements from 2H11 to 1H12, by CCM (100,000 MSRT executions minimum per quarter)



- Nepal showed consistent improvement between 3Q11 and 2Q12. Nepal's decline was caused by a steady decline in detections of the virus family Win32/Sality, the worm family Win32/Nuqel, and the trojan family Win32/Lethic.

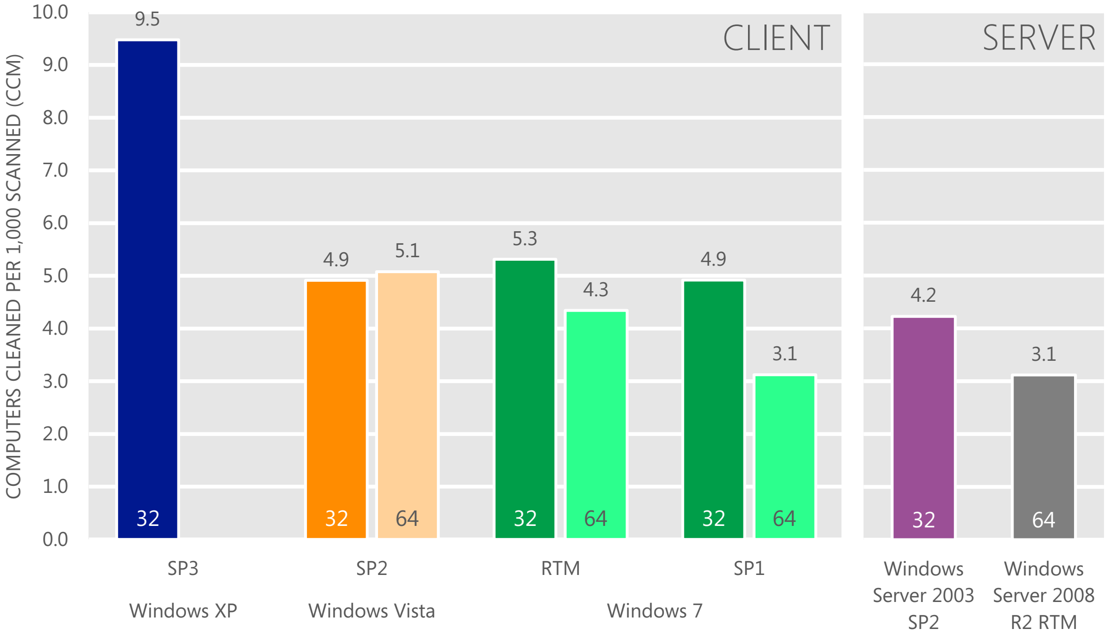
- Brazil, like Nepal, consistently improved each quarter because of a steady decline in detections of Sality, and of three families that are typically used to attack customers of Brazilian banks: the password stealers Win32/Bancos and Win32/Banker, and the trojan downloader Win32/Banload.
- The decreased infection rates in Netherlands, Germany, and Austria all represent returns to typical infection levels after 4Q11 spikes caused by the trojan family Win32/EyeStye. EyeStye is a family of trojans that attempt to steal sensitive data and send it to an attacker. EyeStye variants are created from a malware building kit called SpyEye, which prospective attackers can buy through various malware black market sites. Detection signatures for EyeStye were added to the MSRT in October 2011, which accounted for a large percentage of the EyeStye detections in 4Q11. Germany accounted for about 36 percent of computers reporting detections of EyeStye worldwide in 2H11, with the Netherlands accounting for 13 percent and Austria accounting for 2 percent. With EyeStye detections at much lower levels in 2012, infection rates in all three locations have returned to levels commensurate with those seen during the first three quarters of 2011.

  For more information about this family, see the report "MMPC Threat Report – EyeStye," available from the Microsoft Download Center at www.microsoft.com/download.

Operating system infection rates

The features and updates that are available with different versions of the Windows operating system and the differences in the way people and organizations use each version affect the infection rates for the different versions and service packs. Figure 26 shows the infection rate for each currently supported Windows operating system/service pack combination that accounted for at least 0.1 percent of total MSRT executions in 2Q12.

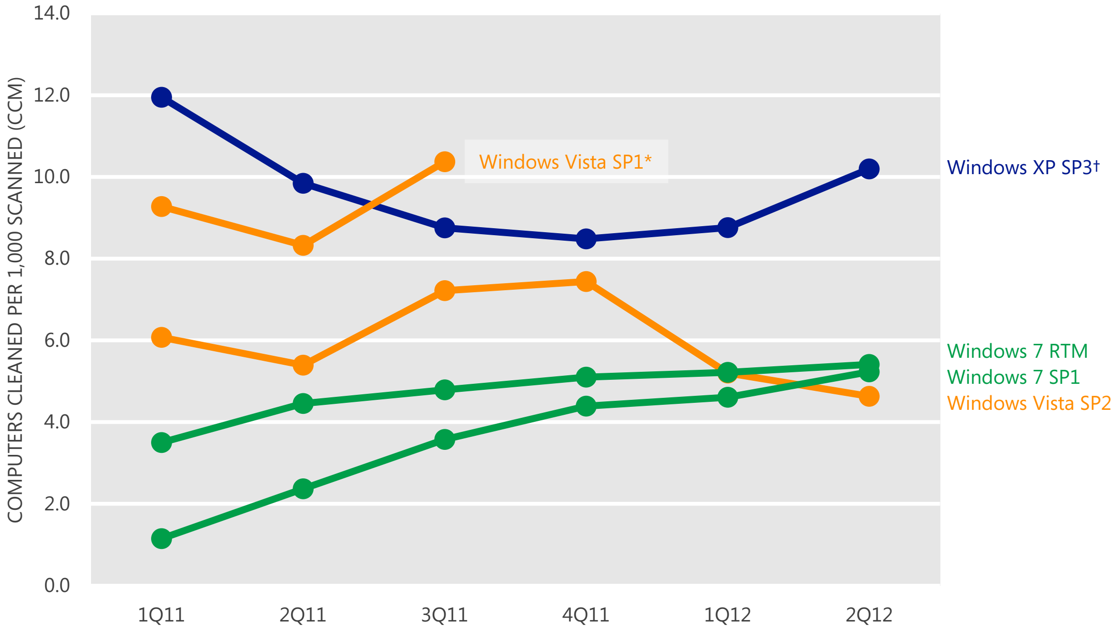
Figure 26. Average infection rate (CCM) by operating system and service pack in 1H12



"32" = 32-bit edition; "64" = 64-bit edition. SP = Service Pack. RTM = release to manufacturing. Operating systems with at least 0.1 percent of total MSRT executions in 2Q12 shown.

- This data is normalized: the infection rate for each version of Windows is calculated by comparing an equal number of computers per version (for example, 1,000 Windows XP SP3 computers to 1,000 Windows 7 SP1 computers).
- As in previous periods, infection rates for more recently released operating systems and service packs tend to be lower than earlier releases, for both client and server platforms. Windows 7 SP1 and Windows Server 2008 R2, the most recently released Windows client and server versions respectively, have the lowest infection rates on the chart, whereas the infection rate for Windows XP SP3 is the highest by a significant margin.
- Infection rates for the 64-bit editions of Windows 7 RTM and SP1 are slightly lower than for the corresponding 32-bit editions, while the infection rates for the 32-bit and 64-bit versions of Windows Vista are nearly identical. In the past, 64-bit computing tended to appeal to a more technically savvy audience than the mainstream, and the infection rates for 64-bit platforms were typically much lower than for their 32-bit counterparts, perhaps because 64-bit users tended to follow safer practices and keep their computers more up-to-date than the average user. Over the past several years, 64-bit computing has become more mainstream, and the infection rate differences between 32-bit and 64-bit platforms have decreased at the same time.

Figure 27. Infection rate (CCM) trends for supported 32-bit version of Windows XP, Windows Vista, and Windows 7, 1Q11–2Q12



* Support ended July 12, 2011. †Extended support for Windows XP ends April 8, 2014.

- Infection rates for Windows 7 increased slightly during the first half of 2012, continuing a trend of small quarter-over-quarter increases since 1Q11. A similar trend of slowly increasing infection rates was observed for Windows Vista between 2007 and 2009, prior to the release of Windows 7. As with the trend for 64-bit computing, this phenomenon may be caused in part by increasing acceptance and usage of the newest consumer version of Windows. Early adopters are often technology enthusiasts who have a higher level of technical expertise than the mainstream computing population. As the Windows 7 install base has grown, new users are likely to possess a lower degree of security awareness than the early adopters and be less aware of safe online practices. (See www.microsoft.com/security/family-safety for tips and guidance about online safety aimed at a non-technical audience.)
- The infection rate for Windows XP SP3 increased in 1H12 after declining for several quarters, primarily because of the worm family Win32/Dorkbot and the trojan downloader Win32/Pluzoks. Most Pluzoks detections affected computers in Korea, where Windows XP remains more widely used than other versions of Windows. (See page 43 for more information about Korea and Pluzoks.) Detection signatures for both families were added to the MSRT

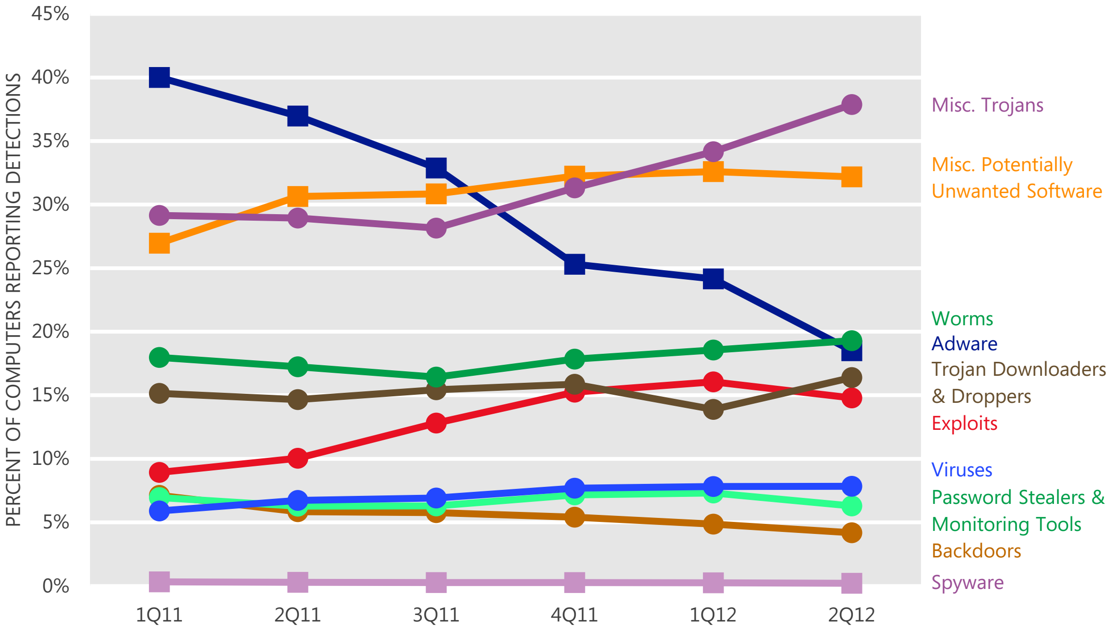
in March 2012 and were responsible for a large portion of the increases observed on Windows XP and Windows 7.

- The infection rate for Windows Vista SP2 decreased significantly in 1Q12, in part because of fewer detections of the trojan family Win32/Tracur. Detection signatures for Tracur were added to the MSRT in July 2011; it was one of the families most commonly detected and removed by the MSRT in the second half of 2011, but was detected in much lower numbers in 1H12.
- For more information about operating system infection rates, see the entry "Operating System Infection Rates - Slight Change in the Trend" (May 17, 2012) in the Microsoft Security blog at blogs.technet.com/security.

Threat categories

The MMPC classifies individual threats into types based on a number of factors, including how the threat spreads and what it is designed to do. To simplify the presentation of this information and make it easier to understand, the *Microsoft Security Intelligence Report* groups these types into 10 categories based on similarities in function and purpose.

Figure 28. Detections by threat category, 1Q11–2Q12, by percentage of all computers reporting detections



Round markers indicate malware categories; square markers indicate potentially unwanted software categories.

- Totals for each time period may exceed 100 percent because some computers report more than one category of threat in each time period.
- The increase in Miscellaneous Trojans, the most commonly detected category during both quarters, was driven by an increase in the generic exploit family JS/IframeRef and the rogue security software family Win32/FakePAV.
- Miscellaneous Potentially Unwanted Software detections remained consistent through both quarters. Win32/Keygen, a generic detection for tools that generate keys for various software products, was the most commonly detected family in this category.
- The Adware category has declined significantly over the past several quarters, from 1st as recently as 3Q11 to 4th in 2Q12. Significantly reduced detections of JS/Pornpop, Win32/Hotbar, and Win32/OpenCandy have been the biggest contributors to the decline.
- The Exploit category, which had been increasing gradually for several quarters, fell slightly in 2Q12. This trend corresponds to the increase and apparent peaking of the Blacole exploit kit.

Threat categories by location

Significant differences exist in the types of threats that affect users in different parts of the world. The spread of malware and its effectiveness are highly dependent on language and cultural factors, in addition to the methods used for distribution. Some threats are spread using techniques that target people who speak a particular language or who use online services that are local to a specific geographic region. Other threats target vulnerabilities or operating system configurations and applications that are unequally distributed around the globe.

Figure 29 shows the relative prevalence of different categories of malware and potentially unwanted software in several locations around the world in 2Q12.

Figure 29. Threat category prevalence worldwide and in the 10 locations with the most detections in 2Q12

Category	World	US	Brazil	Russia	France	Germany	China	Korea	Turkey	UK	Italy
Misc. Trojans	37.9%	43.6%	32.6%	41.8%	28.8%	35.0%	29.9%	23.6%	35.9%	43.2%	31.8%
Misc. Potentially Unwanted Software	32.2%	22.7%	38.1%	57.1%	29.3%	26.6%	45.0%	11.0%	31.2%	23.3%	29.4%
Worms	19.3%	12.3%	23.3%	16.4%	12.6%	8.8%	11.1%	4.9%	34.5%	6.6%	13.6%
Adware	18.5%	19.1%	7.5%	4.9%	31.5%	19.0%	22.4%	38.0%	24.6%	26.1%	24.1%
Trojan Downloaders & Droppers	16.4%	13.1%	22.4%	13.0%	16.1%	10.8%	12.6%	53.8%	13.0%	13.3%	23.2%
Exploits	14.8%	18.7%	5.8%	17.8%	16.2%	28.2%	10.3%	3.5%	6.4%	24.0%	19.7%
Viruses	7.8%	4.4%	9.1%	5.1%	2.2%	2.2%	10.6%	2.0%	15.0%	3.1%	2.5%
Password Stealers & Monitoring Tools	6.3%	4.6%	15.7%	4.1%	4.6%	10.7%	3.2%	2.6%	6.2%	4.8%	7.6%
Backdoors	4.2%	3.4%	3.9%	3.2%	2.8%	3.2%	5.9%	2.0%	4.2%	3.0%	2.9%
Spyware	0.2%	0.3%	0.1%	0.2%	0.1%	0.2%	1.3%	0.1%	0.0%	0.2%	0.1%

Totals for each location may exceed 100 percent because some computers reported threats from more than one category.

- Within each row of Figure 29, a darker color indicates that the category is more prevalent in the specified location than in the others, and a lighter color indicates that the category is less prevalent. As in Figure 20 on page 40, the locations in the table are ordered by number of computers reporting detections in 1H12.
- The United States and the United Kingdom, two predominantly English-speaking locations that also share a number of other cultural similarities, have similar threat mixes in most categories. The Miscellaneous Trojans category is more prevalent in both places than in others primarily because of detections of the English-language rogue security software family Win32/FakePAV, 72.7 percent of which involved computers in the US and UK. Exploits are somewhat more prevalent in the UK than in the US because of the Blacole exploit family, which was detected on proportionally more computers in the UK.

- In Russia, the Miscellaneous Potentially Unwanted Software category is especially prevalent, led by Win32/Pameseg and Win32/Keygen. Pameseg is a family of installers that require the user to send a text message to a premium number to successfully install certain programs, some of which are otherwise available for free. Currently, most variants target Russian speakers.

- Brazil has long had higher-than-average detections of Password Stealers & Monitoring Tools because of the prevalence of malware that targets customers of Brazilian banks, especially Win32/Bancos and Win32/Banker. In 2Q12, Brazil accounted for 69.9 percent of computer reporting Bancos detections worldwide, and 42.4 percent of computers reporting Banker detections.

- Korea had significantly higher-than-average detections of the Trojan Downloaders & Droppers and Adware categories, and significantly lower-than-average detections of all other categories. The high level of Trojan Downloaders & Droppers detection was driven by large numbers of computers infected with Win32/Pluzoks. (See page 43 for more information.) The high level of Adware detections was driven by Win32/Wizpop, which monitors users' Web browsers and diverts requests for certain URLs to a Korean-language site. In 2Q12, 90.4 percent of computers reporting detections of Wizpop were located in Korea.

- Worms were especially prevalent in Turkey in 4Q11 because of Win32/Helompy, a worm that spreads via removable drives and attempts to capture and steal authentication details for a number of different websites or services. The worm contacts a remote host to download arbitrary files and to upload stolen details. In 2Q12, 63.2 percent of computers reporting detections of Helompy were located in Turkey. For more information about Helompy and Turkey, see the blog entry "MSRT December: Win32/Helompy" (December 13, 2011) in the MMPC blog at blogs.technet.com/mmpc.

- Worms and viruses were particularly prevalent in India, driven by detections of the generic worm family Win32/Autorun and the virus families Win32/Sality and Win32/Ramnit.

See "Appendix C: Worldwide infection rates" on page 117 for more information about malware around the world.

Threat families

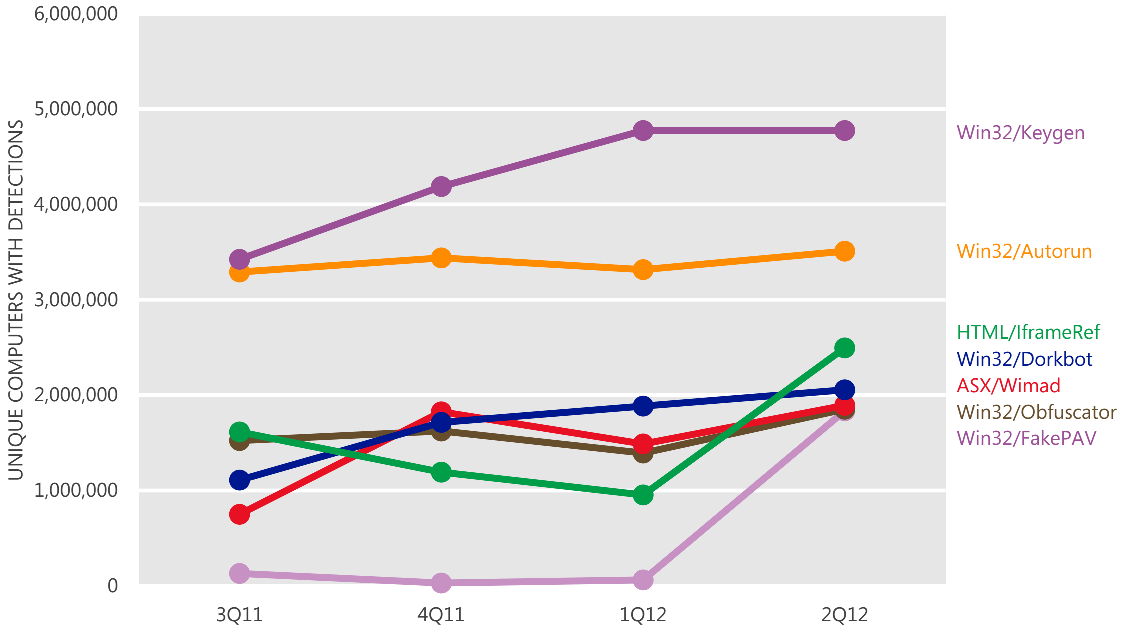
Figure 30 lists the top 10 malware and potentially unwanted software families that were detected on computers by Microsoft antimalware products in the second quarter of 2012, with other quarters included for comparison.

Figure 30. Quarterly trends for the top 10 malware and potentially unwanted software families detected by Microsoft antimalware products in 2Q12, shaded according to relative prevalence

	Family	Category	3Q11	4Q11	1Q12	2Q12
1	Win32/Keygen	Misc. Potentially Unwanted Software	3,424,213	4,187,586	4,775,464	4,775,243
2	Win32/Autorun	Worms	3,292,378	3,438,745	3,316,107	3,510,816
3	JS/Pornpop	Adware	3,944,489	3,906,625	3,994,634	2,838,713
4	JS/IframeRef	Misc. Trojans	1,612,828	1,191,929	952,111	2,493,830
5	Win32/Sality	Viruses	1,728,966	1,951,118	2,101,968	2,097,663
6	Win32/Hotbar	Adware	2,870,465	2,226,173	3,008,677	2,073,789
7	Win32/Dorkbot	Worms	1,107,300	1,713,962	1,883,642	2,055,244
8	ASX/Wimad	Trojan Downloader	748,716	1,825,291	1,487,334	1,890,806
9	Win2/Obfuscator	Misc. Potentially Unwanted Software	1,521,959	1,623,137	1,393,148	1,851,304
10	Win32/FakePAV	Misc. Trojans	128,045	28,694	59,166	1,833,434

For a different perspective on some of the changes that have occurred throughout the year, Figure 31 shows the detection trends for a number of families that increased or decreased significantly over the past four quarters.

Figure 31. Detection trends for a number of notable families, 3Q11–2Q12



- A pair of generic detections, Win32/Keygen and Win32/Autorun, were the first and second most commonly detected families in 1H12. Keygen is a generic detection for tools that generate keys for various software products. See "Deceptive downloads: Software, music, and movies" beginning on page 1 for more information about the role Keygen plays in the distribution of malware.

  Autorun is a generic detection for worms that spread between mounted volumes using the Autorun feature of Windows. Recent changes to the feature in Windows XP and Windows Vista have made this technique less effective, but attackers continue to distribute malware that attempts to target it.

- Detections of the generic family JS/IframeRef more than doubled between 1Q12 and 2Q12 after several quarters of small declines. IframeRef is a generic detection for specially formed HTML inline frame (IFrame) tags that point to remote websites containing malicious content.

- The adware families JS/Pornpop and Win32/Hotbar were the only two families on the list that were detected less often in 1H12 than in 2H11. Detections of both declined considerably between the first and second quarters of the year. Detections of Keygen and the virus family Win32/Sality also decreased slightly between 1Q12 and 2Q12.

- Detections of the rogue security software family Win32/FakePAV increased significantly in 2Q12 and made it the tenth most commonly detected family during that quarter. See “Rogue security software” beginning on page 57 for more information about FakePAV and similar families.

Threat families by platform

Malware does not affect all platforms equally. Some threats are spread by exploits that are ineffective against one or more operating system versions. Some threats are more common in parts of the world where specific platforms are more or less popular than elsewhere. In other cases, differences between platforms may be caused by simple random variation. Figure 32 demonstrates how detections of the most prevalent families in 2Q12 ranked differently on different operating system/service pack combinations.

Figure 32. The malware and potentially unwanted software families most commonly detected by Microsoft antimalware solutions in 2Q12, and how they ranked in prevalence on different platforms

Family	Most significant category	Rank (Windows 7 SP1)	Rank (Windows 7 RTM)	Rank (Windows Vista SP2)	Rank (Windows XP SP3)
Win32/Keygen	Misc. Potentially Unwanted Software	**1**	**1**	11	6
Win32/Autorun	Worms	3	2	15	3
JS/Pornpop	Adware	2	6	2	7
Blacole	Exploits	4	11	**1**	5
JS/IframeRef	Misc. Trojans	10	7	10	**1**
Win32/Sality	Viruses	16	4	35	4
Win32/Hotbar	Adware	6	5	3	21
Win32/Dorkbot	Worms	9	3	21	9
ASX/Wimad	Trojan Downloaders & Droppers	8	9	5	13
Win32/Obfuscator	Misc. Potentially Unwanted Software	5	12	14	11
Win32/FakePAV	Misc. Trojans	7	19	4	10
Win32/Conficker	Worms	15	10	26	8
Win32/Sirefef	Misc. Trojans	12	13	7	15
Java/CVE-2012-0507	Exploits	11	34	6	12
Win32/Pluzoks	Trojan Downloaders & Droppers	40	56	56	2

- Windows 7 is the most widely used consumer operating system worldwide, and the most prevalent families on both Windows 7 RTM and Windows 7 SP1 tended to be the same families that were prevalent overall.

- Blacole was the most commonly detected family on Windows Vista; it ranked lower on other platforms. The exploits in the Blacole kit are deployed on malicious or compromised webpages to infect visitors via the drive-by download method. Internet Explorer 7, which is installed by default with Windows Vista, does not include SmartScreen Filter, the feature that provides malware protection in subsequently released versions of Internet Explorer. This factor may result in some Windows Vista users being more exposed to Blacole. Users should upgrade to a newer version of Internet Explorer with built-in antimalware protection, such as Internet Explorer 9. (See "Malicious websites" beginning on page 75 for more information about SmartScreen Filter and drive-by downloads.)

- The trojan downloader family Win32/Pluzoks was the second most commonly detected family on Windows XP SP3 in 2Q11, but ranked much lower on other platforms. Detections of Pluzoks were highly concentrated in Korea, where use of Windows XP remains relatively higher than in the rest of the world.

Rogue security software

Rogue security software has become one of the most common methods that attackers use to swindle money from victims. Rogue security software, also known as *scareware*, is software that appears to be beneficial from a security perspective but provides limited or no security, generates erroneous or misleading alerts, or attempts to lure users into participating in fraudulent transactions. These programs typically mimic the general look and feel of legitimate security software programs and claim to detect a large number of nonexistent threats while urging users to pay for the "full version" of the software to remove the threats. Attackers typically install rogue security software programs through exploits or other malware, or use social engineering to trick users into believing the programs are legitimate and useful. Some versions emulate the appearance of the Windows Security Center or unlawfully use trademarks and icons to misrepresent themselves. (See www.microsoft.com/security/resources/videos.aspx for an informative series of videos designed to educate general audiences about rogue security software.)

Figure 33. False branding used by a number of commonly detected rogue security software programs

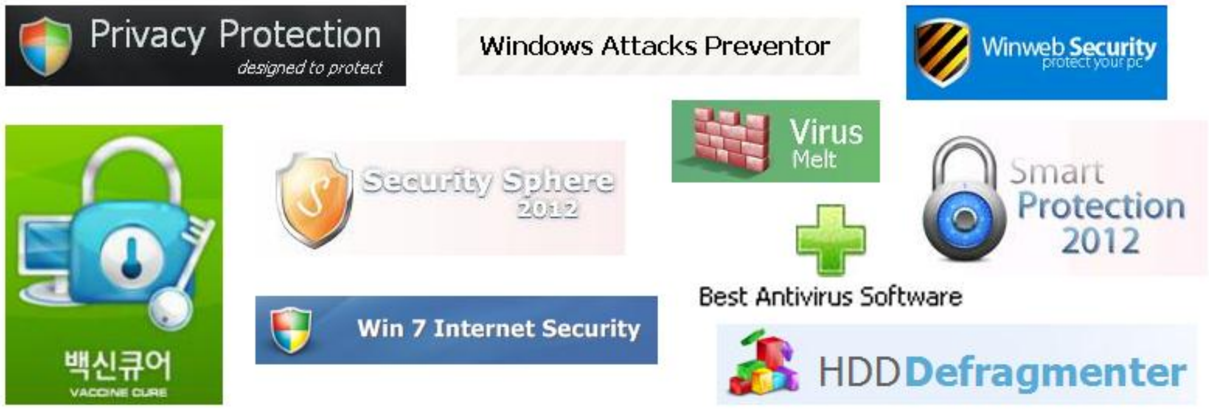
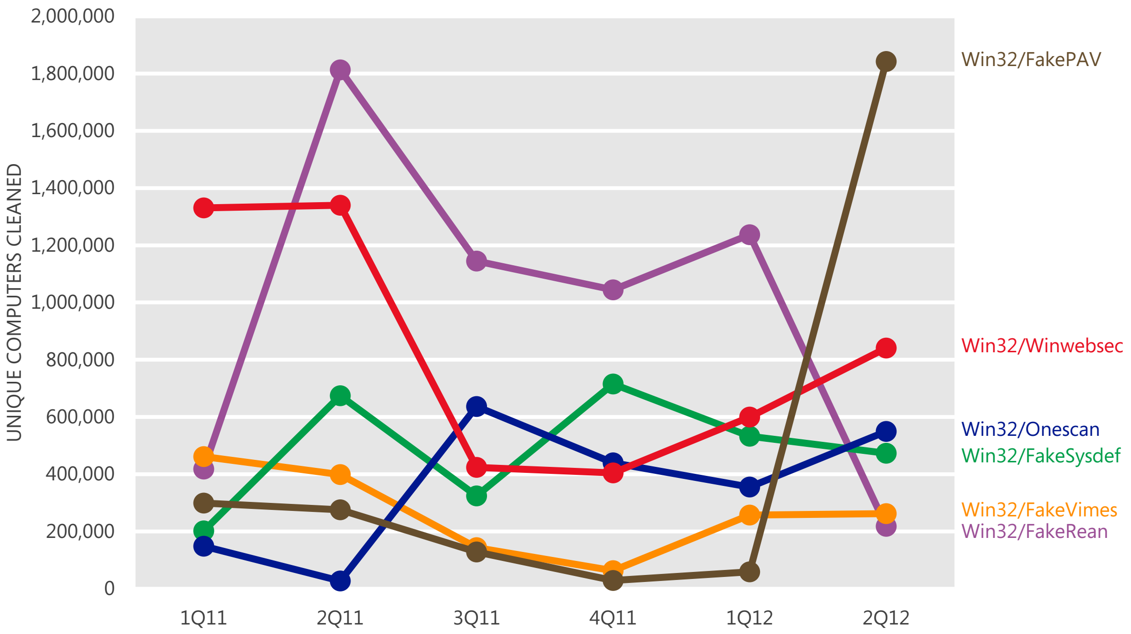


Figure 34 shows detection trends for the most common rogue security software families detected in 1H12.

Figure 34. Trends for the most common rogue security software families detected in 1H12, by quarter



- Worldwide detections of Win32/FakePAV increased by a factor of 30 between 1Q12 and 2Q12, making it the most commonly detected rogue security software family overall during the first half of the year. First detected in 3Q10, FakePAV was the second most commonly detected rogue security software family in 4Q10, when detection signatures for the family were added to the MSRT; detections subsequently declined and remained relatively low each quarter until an enormous increase beginning in April 2012.

FakePAV is distributed under a variety of names, including Windows Threats Destroyer, Windows Firewall Constructor, Windows Attacks Preventor, and Windows Basic Antivirus. FakePAV frequently spreads by masquerading as Microsoft Security Essentials on malicious and compromised webpages, presenting a graphic resembling a genuine Microsoft Security Essentials window and claiming to have discovered several infections on the target computer. Recent variants have included large amounts of irrelevant text, such as excerpts from William Shakespeare's *Romeo and Juliet*, in the installation package in an apparent effort to obfuscate the files and avoid detection by antimalware software.

Visit http://youtu.be/UPY9mJKIagw for an informative video from the *Microsoft Security Intelligence Report* team about FakePAV and how to remove it. For additional in-depth information about FakePAV, see the following blog entries in the MMPC blog (blogs.technet.com/mmpc):

- A Rogue by any other name… (March 1, 2012)
- Knowing you, knowing meme… (July 24, 2012)

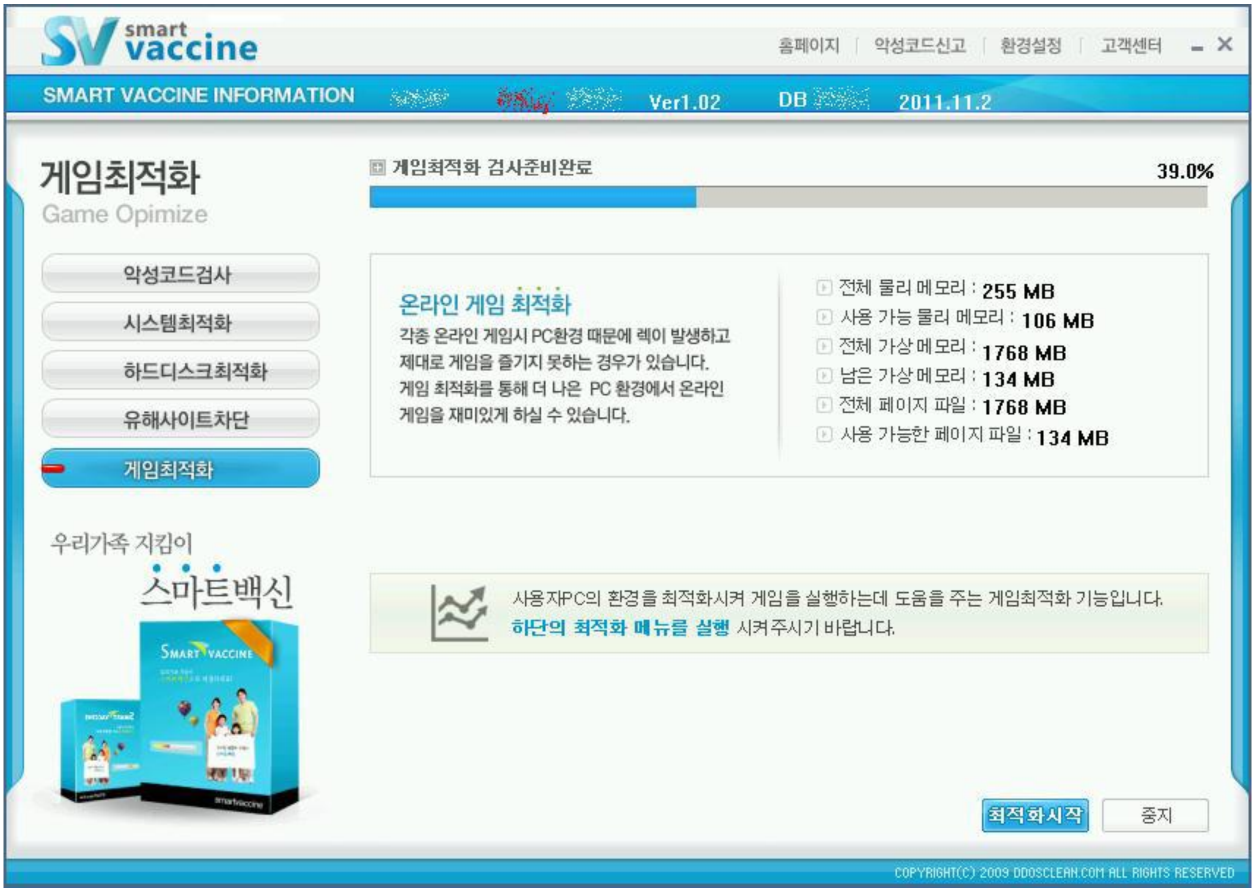
- Detections of Win32/Winwebsec in 2Q12 were more than 40 percent higher than in 1Q12 and more than double the family's 4Q11 total, making it the third most commonly detected rogue security software family in the first half of 2012. Winwebsec has also been distributed under many names, with the user interface and other details varying to reflect each variant's individual branding. These different distributions of the trojan use various installation methods, with file names and system modifications that can differ from one variant to the next. The attackers behind Winwebsec are also believed to be responsible for MacOS_X/FakeMacdef, the highly publicized Mac Defender rogue security software program for Apple Mac OS X that first appeared in May 2011. Detections for Winwebsec were added to the MSRT in May 2009.
- Win32/Onescan was the fourth most commonly detected rogue security software family in 1Q12. Detections increased between the first and second quarters, making it the third most detected family in 2Q12, and it was a major reason that Korea accounted for the second largest number of rogue security software detections by country and region in 1H12.

Figure 35. Countries or regions with the most rogue detections in 1H12

	Country/Region	Rogue detections 1Q12	% of all rogue 1Q12	Rogue detections 2Q12	% of all rogue 2Q12
1	United States	1,858,210	56.8%	2,315,281	52.9%
2	Korea	341,988	10.5%	501,119	11.4%
3	Canada	142,545	4.4%	229,013	5.2%
4	United Kingdom	151,465	4.6%	222,047	5.1%
5	France	67,769	2.1%	107,631	2.5%
6	Australia	59,410	1.8%	90,480	2.1%
7	Germany	51,839	1.6%	71,715	1.6%
8	Brazil	43,049	1.3%	63,067	1.4%
9	Turkey	27,179	0.8%	54,939	1.3%
10	Italy	56,735	1.7%	54,667	1.2%

Onescan is a Korean-language rogue security software distributed under a variety of names, brands, and logos. The installer selects the branding randomly from a defined set, apparently without regard to the operating system version.

Figure 36. A variant of Win32/Onescan, a Korean-language rogue security software program



게임최적화 검사준비완료 39.0%

온라인 게임 최적화

각종 온라인 게임시 PC환경 때문에 렉이 발생하고 제대로 게임을 즐기지 못하는 경우가 있습니다. 게임 최적화를 통해 더 나은 PC 환경에서 온라인 게임을 재미있게 하실 수 있습니다.

전체 물리 메모리 : 255 MB
사용 가능 물리 메모리 : 106 MB
전체 가상 메모리 : 1768 MB
남은 가상 메모리 : 134 MB
전체 페이지 파일 : 1768 MB
사용 가능한 페이지 파일 : 134 MB

사용자PC의 환경을 최적화시켜 게임을 실행하는데 도움을 주는 게임최적화 기능입니다.
하단의 최적화 메뉴를 실행 시켜주시기 바랍니다.

- Win32/FakeSysdef was the third most commonly detected rogue security software family in 1Q12, although it declined to fourth in 2Q12. Unlike most rogue security software families, FakeSysdef does not claim to detect malware infections. Instead, it masquerades as a performance utility that falsely claims to find numerous hardware and software errors such as bad hard disk sectors, disk fragmentation, registry errors, and memory problems. Like other rogue security software families, it claims that the user must purchase additional software to fix the nonexistent problems.

- Detections of Win32/FakeRean, the most commonly detected rogue security software family in 1Q12, declined significantly in 2Q12, to sixth. FakeRean has been distributed under several different names, which are often generated at random based upon the operating system of the affected computer.

- Win32/FakeVimes was the fifth most commonly detected rogue security software family in both 1Q12 and 2Q12. Its variants frequently and unlawfully display the Microsoft Genuine Advantage logo, or copy elements of the Windows Defender and Microsoft Security Essentials user interfaces. Detection signatures for FakeRean and FakeVimes were added to the MSRT in August and November 2009, respectively. See the entry "When imitation isn't a form of flattery" (January 29, 2012) in the MMPC blog for more information about these families.

Home and enterprise threats

The usage patterns of home users and enterprise users tend to be very different. Enterprise users typically use computers to perform business functions while connected to a network, and may have limitations placed on their Internet and email usage. Home users are more likely to connect to the Internet directly or through a home router and to use their computers for entertainment purposes, such as playing games, watching videos, shopping, and communicating with friends. These different usage patterns mean that home users tend to be exposed to a different mix of computer threats than enterprise users.

The infection telemetry data produced by Microsoft antimalware products and tools includes information about whether the infected computer belongs to an Active Directory Domain Services domain. Such domains are used almost exclusively in enterprise environments, and computers that do not belong to a domain are more likely to be used at home or in other non-enterprise contexts. Comparing the threats encountered by domain-joined computers and non-domain computers can provide insights into the different ways attackers target enterprise and home users and which threats are more likely to succeed in each environment.

Figure 37 and Figure 38 list the top 10 families detected on domain-joined and non-domain computers, respectively, in 1H12.

Figure 37. Top 10 families detected on domain-joined computers in 2Q12, by percentage of domain-joined computers reporting detections

	Family	Most significant category	3Q11	4Q11	1Q12	2Q12
1	JS/IframeRef	Misc. Trojans	4.0%	2.6%	2.3%	11.3%
2	Win32/Conficker	Worms	14.7%	13.5%	12.7%	10.8%
3	Win32/Autorun	Worms	9.3%	8.5%	7.5%	7.0%
4	Blacole	Exploits	2.3%	6.4%	7.0%	5.4%
5	Win32/Keygen	Misc. Potentially Unwanted Software	4.6%	5.0%	5.5%	5.3%
6	Win32/FakePAV	Misc. Trojans	3.5%	2.5%	2.0%	4.8%
7	JS/BlacoleRef	Misc. Trojans	0.3%	1.8%	3.3%	4.1%
8	Java/CVE-2012-0507	Exploits	—	—	0.5%	4.7%
9	Win2/Obfuscator	Misc. Potentially Unwanted Software	2.3%	2.7%	2.2%	3.5%
10	Win32/Sirefef	Misc. Trojans	0.5%	2.8%	2.6%	3.5%

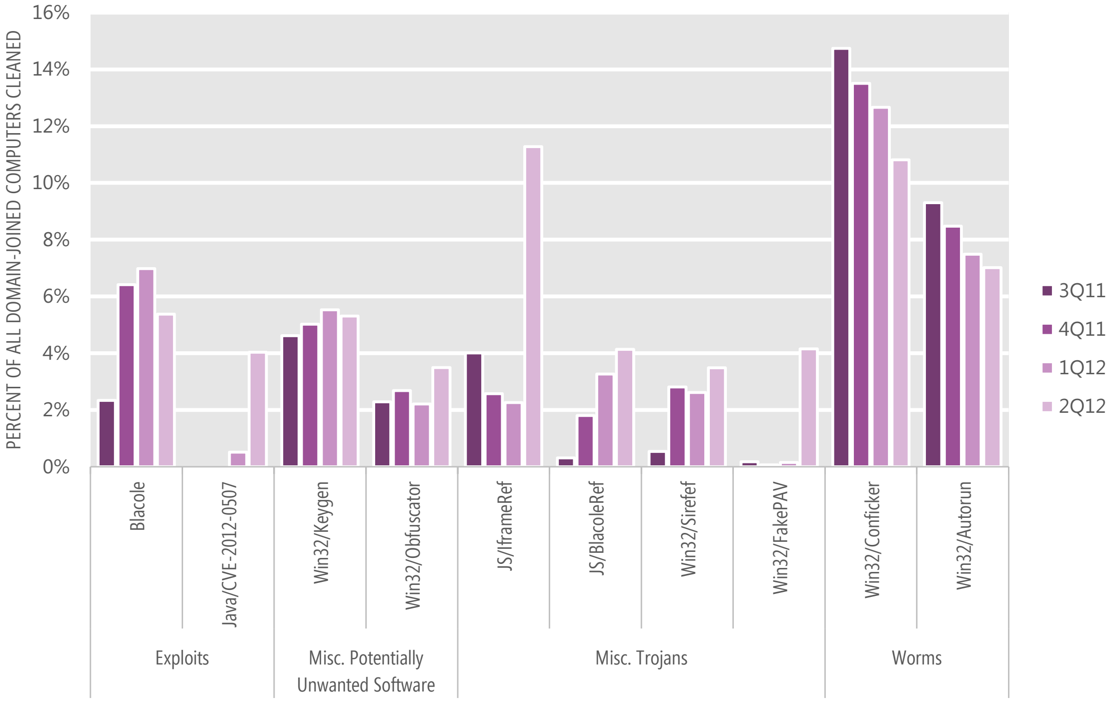
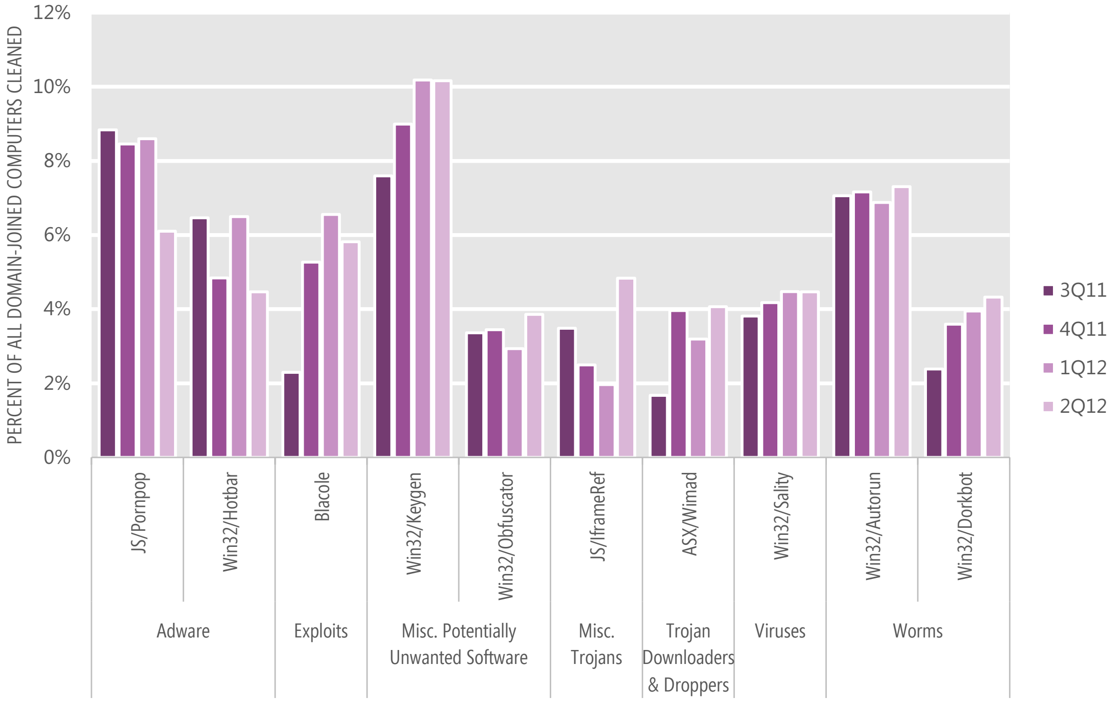


Figure 38. Top 10 families detected on non-domain computers in 2Q12, by percentage of non-domain computers reporting detections

	Family	Most significant category	3Q11	4Q11	1Q12	2Q12
1	Win32/Keygen	Misc. Potentially Unwanted Software	7.6%	9.0%	10.2%	10.2%
2	Win32/Autorun	Worms	7.1%	7.2%	6.9%	7.3%
3	JS/Pornpop	Adware	8.8%	8.5%	8.6%	6.1%
4	Blacole	Exploits	2.3%	5.3%	6.6%	5.8%
5	JS/IframeRef	Misc. Trojans	3.5%	2.5%	2.0%	4.8%
6	Win32/Hotbar	Adware	6.5%	4.8%	6.5%	4.5%
7	Win32/Sality	Viruses	3.8%	4.2%	4.5%	4.5%
8	Win32/Dorkbot	Worms	2.4%	3.6%	4.0%	4.3%
9	ASX/Wimad	Trojan Downloaders & Droppers	1.7%	4.0%	3.2%	4.1%
10	Win2/Obfuscator	Misc. Potentially Unwanted Software	3.4%	3.4%	2.9%	3.9%



- Five families are common to both lists, notably the generic families Win32/Keygen and Win32/Autorun and the exploit family Blacole. (See

"Deceptive downloads: Software, music, and movies" beginning on page 1 for more information about Keygen and similar families.)

- Families that were significantly more prevalent on domain-joined computers during at least one quarter include the generic family JS/IframeRef and the worm family Win32/Conficker. (See "How Conficker continues to propagate" in *Microsoft Security Intelligence Report, Volume 12 (July–December 2011)* for more information about Conficker and Microsoft's efforts to fight it.)
- Families that were significantly more prevalent on non-domain computers include Keygen and the adware families JS/Pornpop and Win32/Hotbar.
- Detections of families in the Miscellaneous Trojans category increased significantly on domain-joined computers in 1H12. Only one such family, the generic detection JS/Redirector, was among the top families on domain-joined computers in 2H11. For this report, four Miscellaneous Trojan families—the generic detections IframeRef and JS/BlacoleRef, the multi-component trojan Win32/Sirefef, and the rogue security software family Win32/FakePAV—were in the top 10.
- Detections on non-domain computers have historically tended to be dominated by adware, but a decline in detections of a number of prevalent adware families has led to a threat mix that more closely resembles that of domain-joined computers, although many of the prevalent families are different.
- Detections of IframeRef increased significantly on both domain-joined and non-domain computers in 2Q12. IframeRef is a generic detection for specially formed HTML inline frame (IFrame) tags that point to remote websites containing malicious content. BlacoleRef, on the top 10 list for domain-joined computers, is a similar family that is used to spread Blacole exploits.
- Java/CVE-2012-0507 is a generic detection for exploits of a vulnerability in the Java JRE. It first appeared in early 2012 and spread rapidly thereafter, becoming the eight most commonly detected family on domain-joined computers in 2Q12. See page 25 for more information about exploits of this vulnerability.

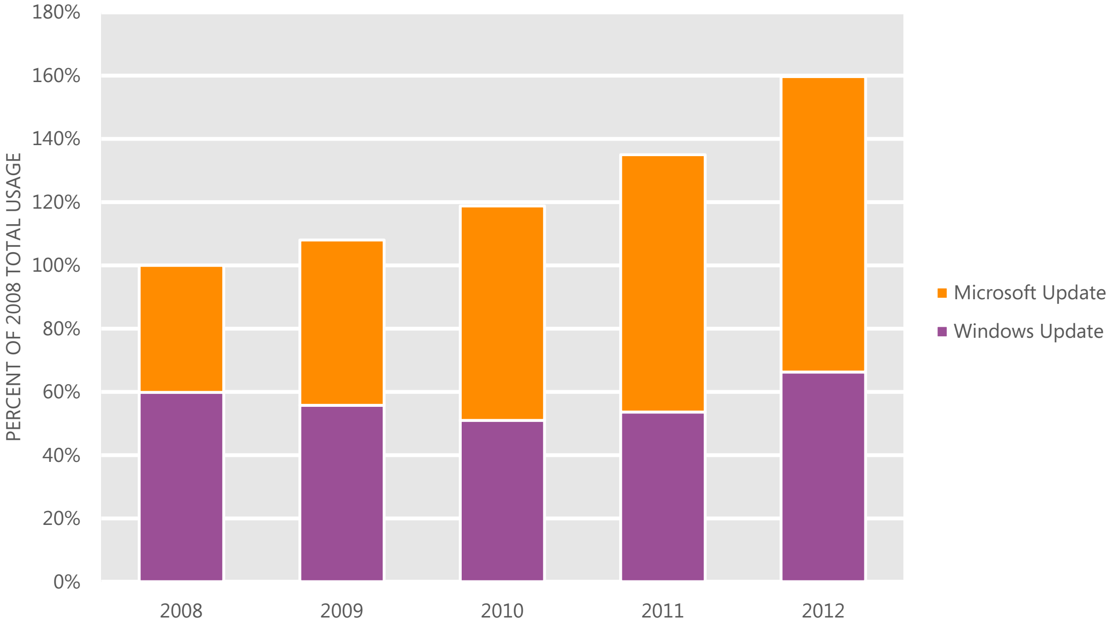
Windows Update and Microsoft Update usage

Microsoft provides several tools and services that enable users to download and install updates directly from Microsoft or, for business customers, from update servers designated by their system administrators. The update client software (called Automatic Updates in Windows XP and Windows Server 2003, and simply Windows Update in subsequently released versions of Windows) connects to an update service for the list of available updates. After the update client determines which updates are applicable to the user's computer, it installs the updates or notifies the user that they are available, depending on how the client is configured and the nature of each update.

For end users, Microsoft provides two update services that the update clients can use:

- **Windows Update** provides updates for Windows components and for device drivers provided by Microsoft and other hardware vendors. Windows Update also distributes signature updates for Microsoft antimalware products and the monthly release of the MSRT. By default, when a user enables automatic updating, the update client connects to the Windows Update service for updates.
- **Microsoft Update** provides all of the updates offered through Windows Update and provides updates for other Microsoft software, such as the Microsoft Office system, Microsoft SQL Server, and Microsoft Exchange Server. Users can opt in to the service when installing software that is serviced through Microsoft Update or at the Microsoft Update website (update.microsoft.com/microsoftupdate). Microsoft recommends that users configure computers to use Microsoft Update instead of Windows Update to help ensure they receive timely security updates for Microsoft products.

Figure 39. Windows computers updated by Windows Update and Microsoft Update worldwide, 2008–2012



- Figure 39 shows the increase in the number of computers updated by Windows Update and Microsoft Update over the last four years, indexed to the total usage for both services in 2008.
- Since 2008, worldwide usage of Windows Update and Microsoft Update has increased by 59.7 percent. Almost all of this growth is because of increased use of Microsoft Update, which went up 53 percentage points between 2008 and 2012, compared to 6 percentage points for Windows Update.

Guidance: Defending against malware

Effectively protecting users from malware requires an active effort on the part of organizations and individuals. For in-depth guidance, see Protecting Against Malicious and Potentially Unwanted Software in the "Mitigating Risk" section of the *Microsoft Security Intelligence Report* website.

Email threats

More than four-fifths of the email messages sent over the Internet are unwanted. Not only does all this unwanted email tax recipients' inboxes and the resources of email providers, but it also creates an environment in which emailed malware attacks and phishing attempts can proliferate. Email providers, social networks, and other online communities have made blocking spam, phishing, and other email threats a top priority.

Spam messages blocked

The information in this section of the *Microsoft Security Intelligence Report* is compiled from telemetry data provided by Exchange Online Protection (formerly Microsoft Forefront Online Protection for Exchange), which provides spam, phishing, and malware filtering services for thousands of Microsoft enterprise customers that process tens of billions of messages each month.

Figure 40. Messages blocked by Exchange Online Protection each month, July 2011–June 2012

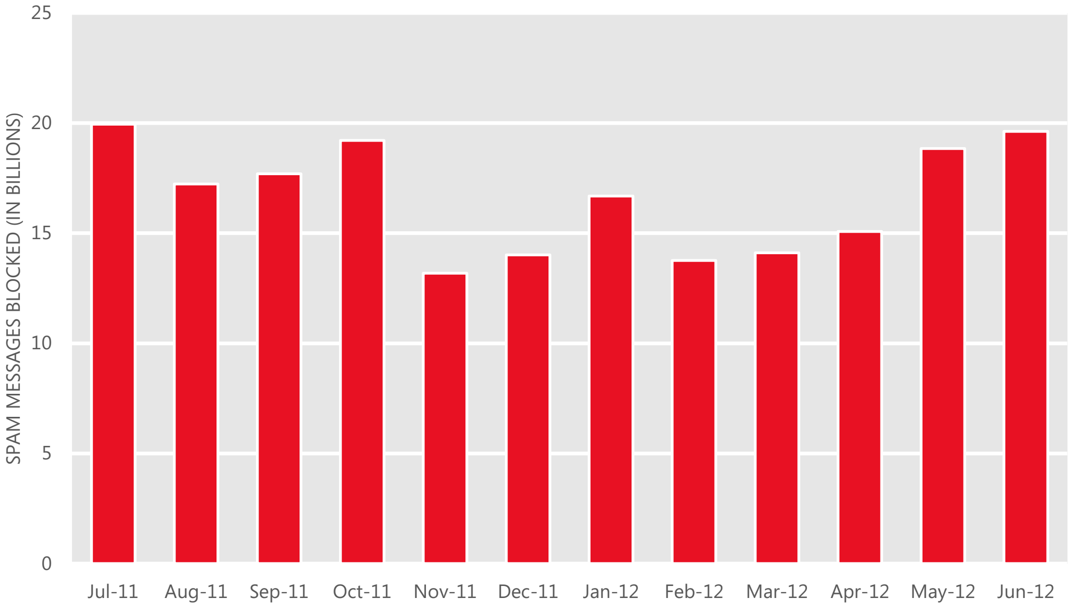
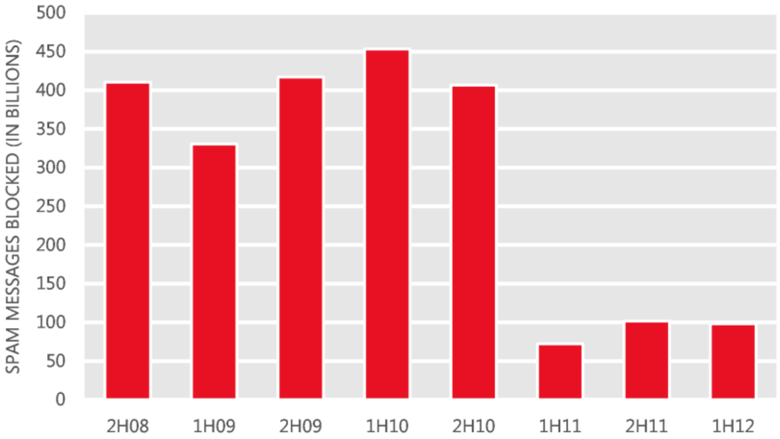


Figure 41. Messages blocked by Exchange Online Protection each half-year period, 2H08–1H12

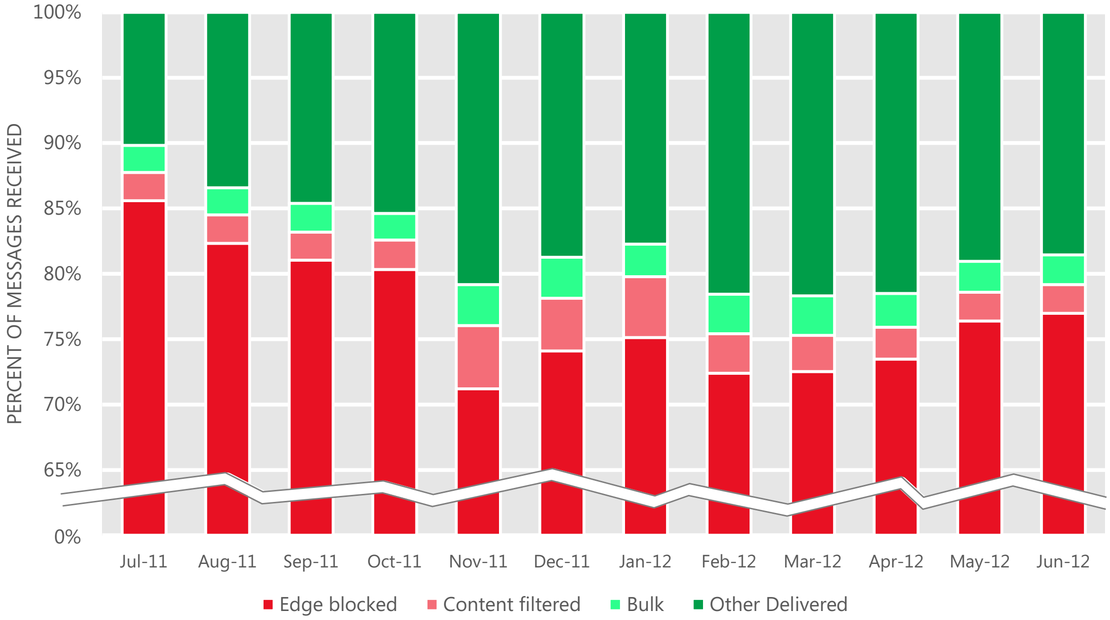


- Blocked mail volumes in 1H12 were consistent with those of 2H11, and remain well below levels seen prior to the end of 2010, as shown in Figure 41. The dramatic decline in spam observed over the past year and a half has occurred in the wake of successful takedowns of a number of large spam-sending botnets, notably Cutwail (August 2010) and Rustock (March 2011).[11] In 1H12, about 1 in 4 email messages were delivered to recipients' inboxes without being blocked or filtered, compared to just 1 in 33 messages two years prior.

Exchange Online Protection performs spam filtering in two stages. Most spam is blocked by servers at the network edge, which use reputation filtering and other non-content-based rules to block spam or other unwanted messages. Messages that are not blocked at the first stage are scanned using content-based rules, which detect and filter many additional email threats, including attachments that contain malware.

[11] For more information about the Cutwail takedown, see *Microsoft Security Intelligence Report, Volume 10 (July-December 2010)*. For more information about the Rustock takedown, see "Battling the Rustock Threat," available from the Microsoft Download Center.

Figure 42. Percentage of incoming messages blocked, categorized as bulk email, and delivered, July 2011–June 2012



- Between 73 and 77 percent of incoming messages were blocked at the network edge each month in 1H12, which means that only 23 to 27 percent of incoming messages had to be subjected to the more resource-intensive content filtering process. Between 9 and 19 percent of the remaining messages (3 to 5 percent of all incoming messages) were filtered as spam each month.
- In general, the volume of mail blocked at the network edge each month is much less than it was prior to the end of 2010, when edge blocks routinely stopped more than 90 percent of incoming messages. Although this reduced volume has caused the share of blocked messages attributed to content filtering to increase, the actual volume of content-filtered messages has also decreased significantly. For example, the number of messages blocked by content filters in June 2012 is less than a third than that of two years prior.
- In May 2011, Exchange Online Protection began identifying bulk email messages that some users consider unwanted, but which aren't categorized as spam by edge blocks or content filters. These typically include email newsletters, advertisements, and marketing messages that users claim they never asked for, or don't remember subscribing to. Exchange Online Protection flags these messages as bulk in an incoming header so customers

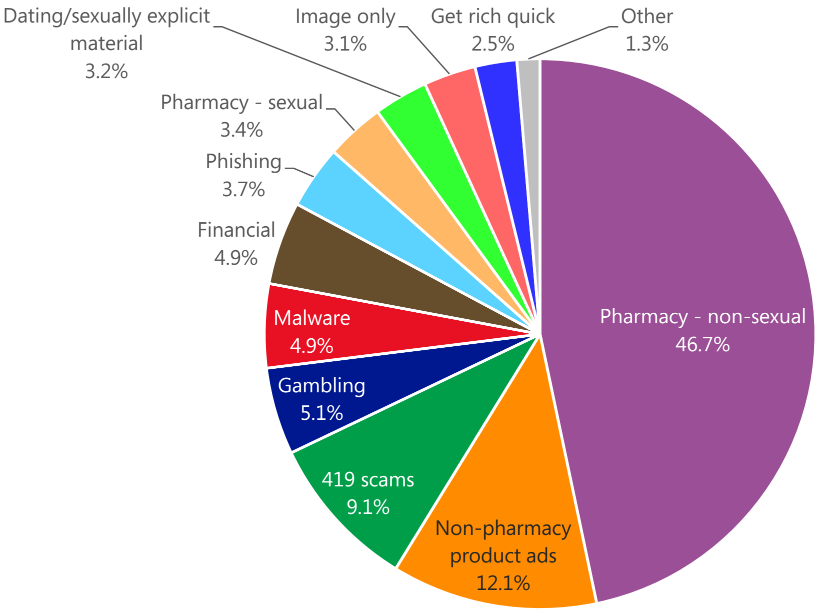
and end users can use rules in Microsoft Outlook or Exchange to filter, move, or deliver them as desired.

Bulk email volumes did not vary significantly from month to month in 1H12. Between 11 and 12 percent of all delivered messages were categorized as bulk each month.

Spam types

The Exchange Online Protection content filters recognize several different common types of spam messages. Figure 43 shows the relative prevalence of the spam types that were detected in 1H12.

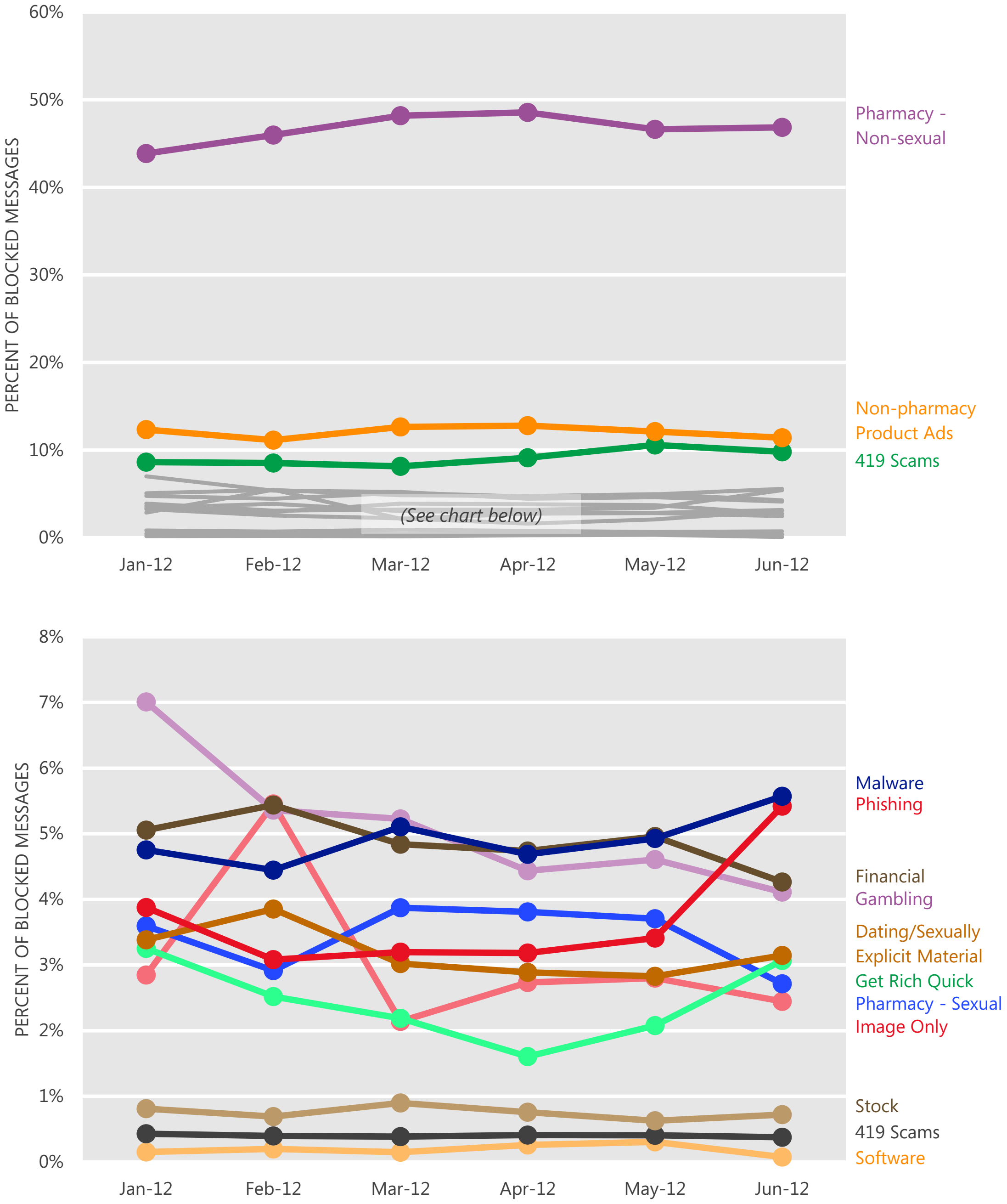
Figure 43. Inbound messages blocked by Exchange Online Protection filters in 1H12, by category



- Advertisements for non-sexual pharmaceutical products accounted for 46.7 percent of the messages blocked by Exchange Online Protection content filters in 1H12, which is nearly unchanged from 46.5 percent in 2H11. Combined with sexually related pharmaceutical ads (3.4 percent of the total), ads for pharmaceutical products accounted for the majority of spam blocked by content filters in 1H12.

- Ads for other types of products accounted for 12.1 percent of messages blocked, a decrease from 13.2 percent in 2H11.
- Spam messages associated with advance-fee fraud (so-called "419 scams") accounted for 9.1 percent of messages blocked, a decrease from 10.7 percent in 1H11. An advance-fee fraud is a common confidence trick in which the sender of a message purports to have a claim on a large sum of money but is unable to access it directly for some reason, typically involving bureaucratic red tape or political corruption. The sender asks the prospective victim for a temporary loan to be used for bribing officials or paying fees to get the full sum released. In exchange, the sender promises the target a share of the fortune amounting to a much larger sum than the original loan, but does not deliver.
- Each of the other spam categories tracked accounted for less than 10 percent of messages blocked in 1H12.

Figure 44. Inbound messages blocked by Exchange Online Protection content filters each month, January–June 2012, by category



- Spammers sometimes engage in campaigns in which large volumes of particular kinds of spam are sent for limited periods of time. Gambling-related messages accounted for 7.0 percent of spam in January before rapidly tapering off; by June, the category accounted for just 4.1 percent of spam.
- Spam messages that contain images and no text, which spammers sometimes send in an attempt to evade content filters, peaked in February before retreating to significantly lower levels through the end of the period.
- Phishing messages, which accounted for between 3.1 and 3.9 percent of messages each month for most of 1H12, rose to 5.4 percent in June. See "Malicious websites" beginning on page 75 for more information about phishing.

Guidance: Defending against threats in email

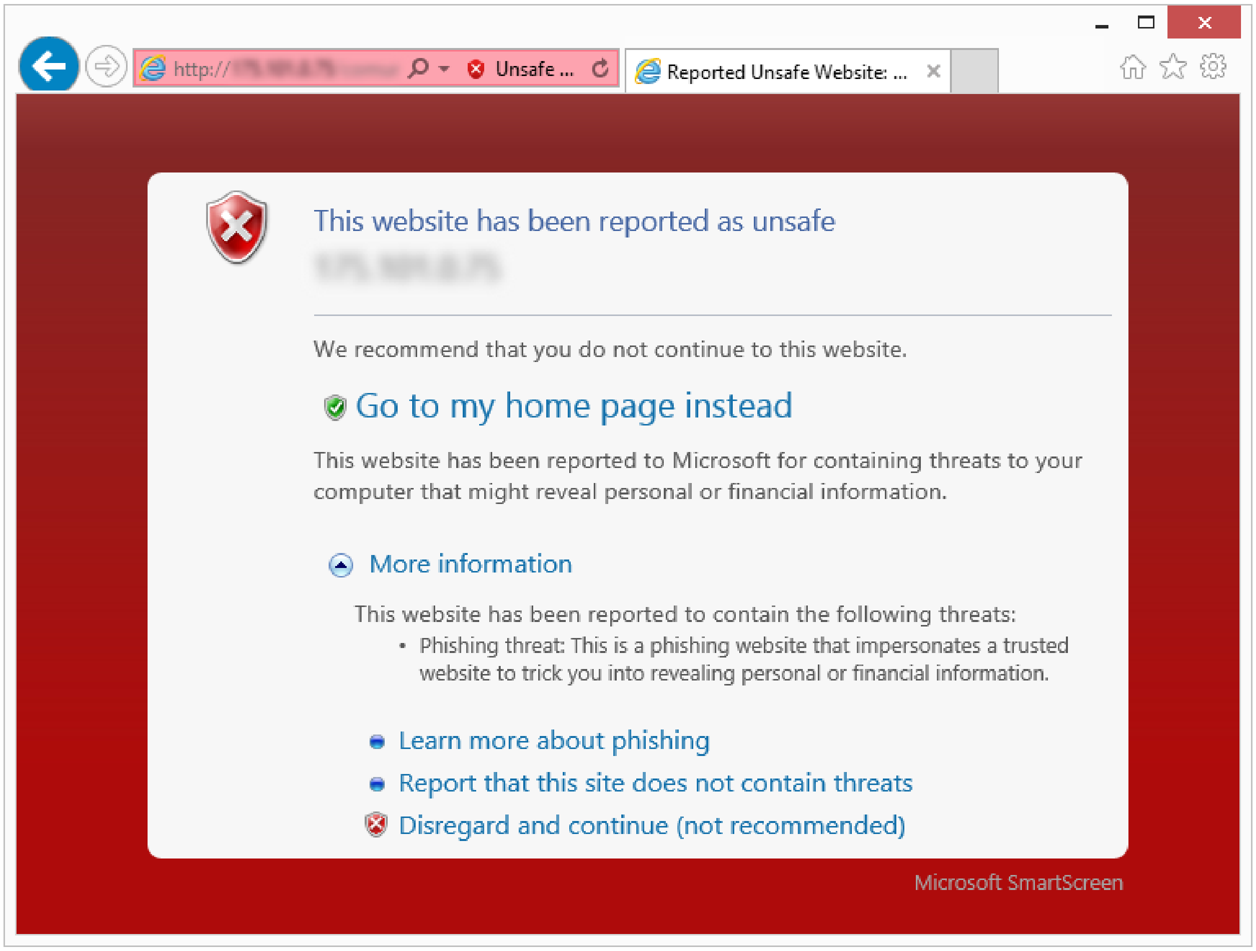
In addition to using a filtering service such as Exchange Online Protection, organizations can take a number of steps to reduce the risks and inconvenience of unwanted email. Such steps include implementing email authentication techniques and observing best practices for sending and receiving email. For in-depth guidance, see Guarding Against Email Threats in the "Managing Risk" section of the *Microsoft Security Intelligence Report* website.

Malicious websites

Attackers often use websites to conduct phishing attacks or distribute malware. Malicious websites typically appear to be completely legitimate and often provide no outward indicators of their malicious nature, even to experienced computer users. In many cases, these sites are legitimate websites that have been compromised by malware, SQL injection, or other techniques in an effort by attackers to take advantage of the trust users have invested in them. To help protect users from malicious webpages, Microsoft and other browser vendors have developed filters that keep track of sites that host malware and phishing attacks and display prominent warnings when users try to navigate to them.

The information in this section is compiled from a variety of internal and external sources, including telemetry data produced by SmartScreen Filter (in Windows Internet Explorer 8 through 10) and the Phishing Filter (in Internet Explorer 7), from a database of known active phishing and malware hosting sites reported by users of Internet Explorer and other Microsoft products and services, and from malware data provided by Microsoft antimalware technologies. (See "Appendix B: Data sources" on page 115 for more information about the products and services that provided data for this report.)

Figure 45. SmartScreen Filter in Internet Explorer blocks reported phishing and malware distribution sites to protect the user



This website has been reported as unsafe

[redacted]

We recommend that you do not continue to this website.

Go to my home page instead

This website has been reported to Microsoft for containing threats to your computer that might reveal personal or financial information.

More information

This website has been reported to contain the following threats:

- Phishing threat: This is a phishing website that impersonates a trusted website to trick you into revealing personal or financial information.

- Learn more about phishing
- Report that this site does not contain threats
- Disregard and continue (not recommended)

Microsoft SmartScreen

Phishing sites

Microsoft gathers information about phishing sites and impressions from *phishing impressions* generated by users who choose to enable the Phishing Filter or SmartScreen Filter in Internet Explorer. A phishing impression is a single instance of a user attempting to visit a known phishing site with Internet Explorer and being blocked, as illustrated in Figure 46.

Figure 46. How Microsoft tracks phishing impressions

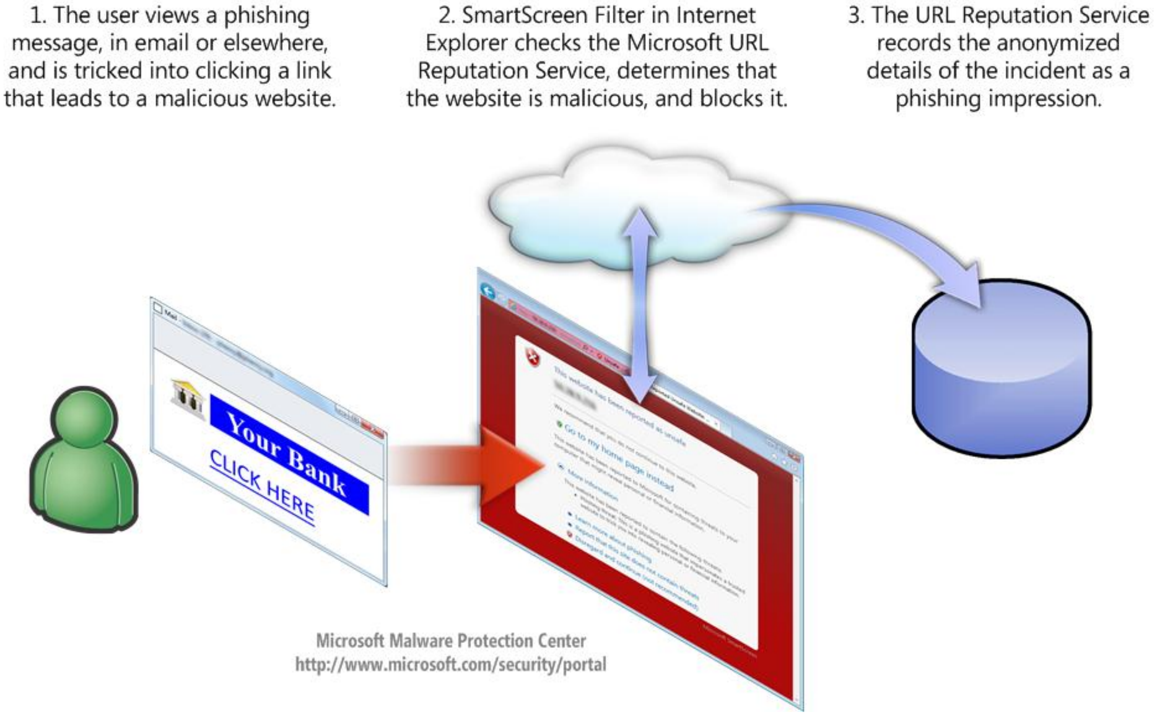
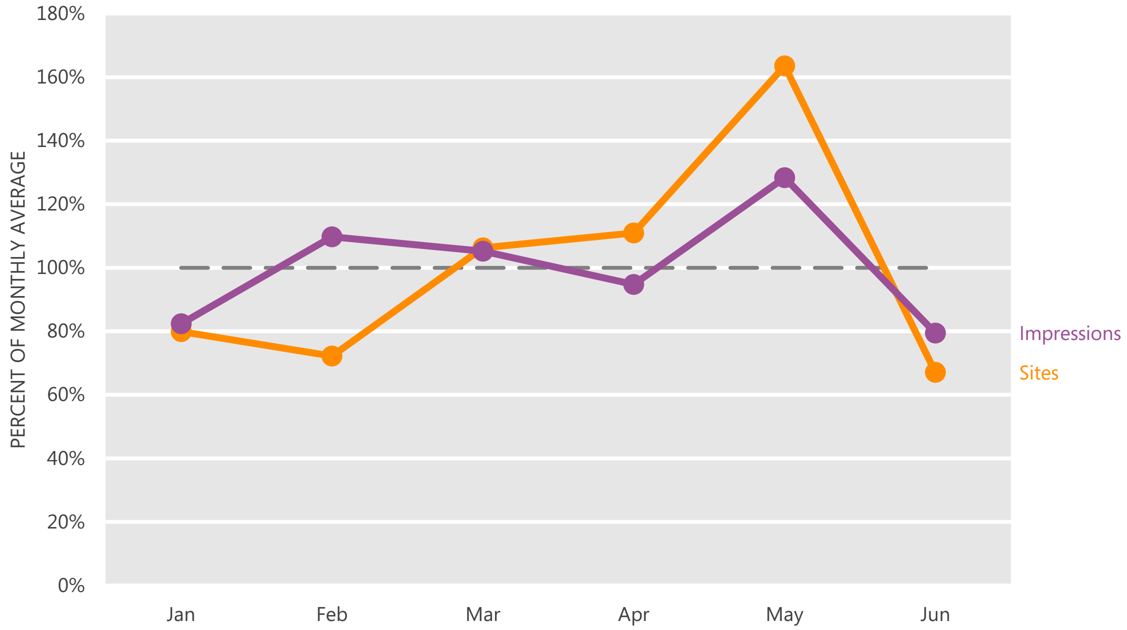

1. The user views a phishing message, in email or elsewhere, and is tricked into clicking a link that leads to a malicious website.
2. SmartScreen Filter in Internet Explorer checks the Microsoft URL Reputation Service, determines that the website is malicious, and blocks it.
3. The URL Reputation Service records the anonymized details of the incident as a phishing impression.
Mail
Your Bank
CLICK HERE
This website has been reported as unsafe
Go to my home page instead
More information
Microsoft Malware Protection Center
http://www.microsoft.com/security/portal

Figure 47 compares the volume of active phishing sites in the Microsoft URL Reputation Service database each month with the volume of phishing impressions tracked by Internet Explorer.

Figure 47. Phishing sites and impressions tracked each month, January–June 2012, relative to the monthly average for each



- Figures for both sites and impression have been mostly stable throughout 1H12. Phishers sometimes engage in temporary campaigns that drive more traffic to each phishing page for a month or two, without necessarily increasing the total number of active phishing pages they maintain at the same time. The SmartScreen data does not display any evidence for significant campaign activity during 1H12, although the number of active sites increased briefly in May before returning to a lower level in June.
- Most phishing sites only last a few days, and attackers create new ones to replace older ones as they are taken offline, so the list of known phishing sites is prone to constant change without significantly affecting overall volume. This phenomenon can cause significant fluctuations in the number of active phishing sites being tracked, like the one seen between April and June.

Target institutions

Figure 48 and Figure 49 show the percentage of phishing impressions and active phishing sites, respectively, recorded by Microsoft during each month from January to June 2012 for the most frequently targeted types of institutions.

Figure 48. Impressions for each type of phishing site each month, January–June 2012, as reported by SmartScreen Filter

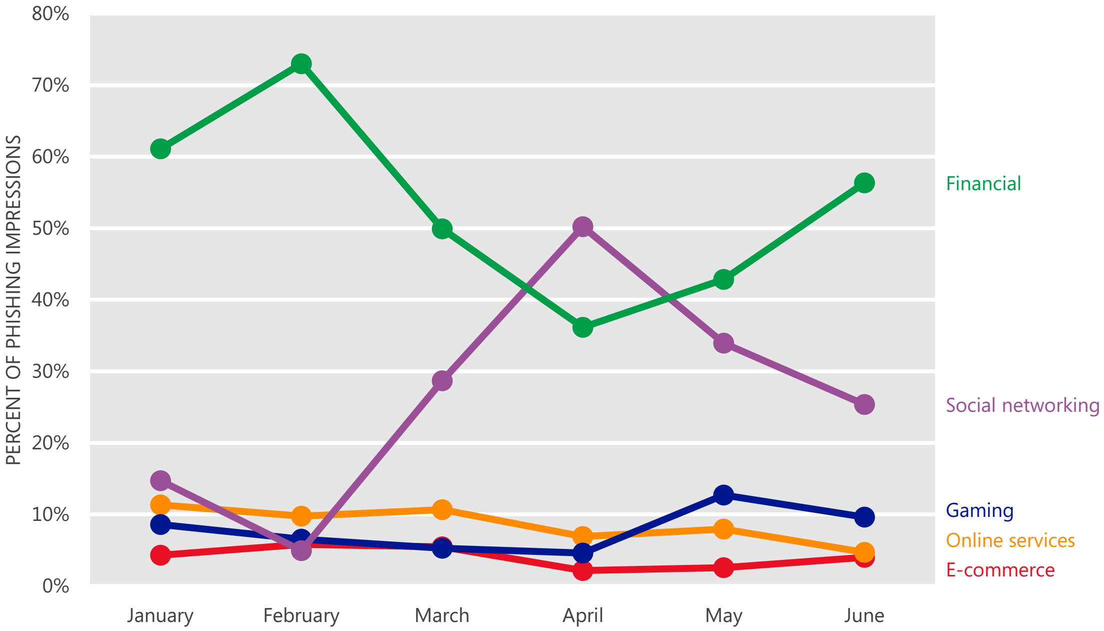
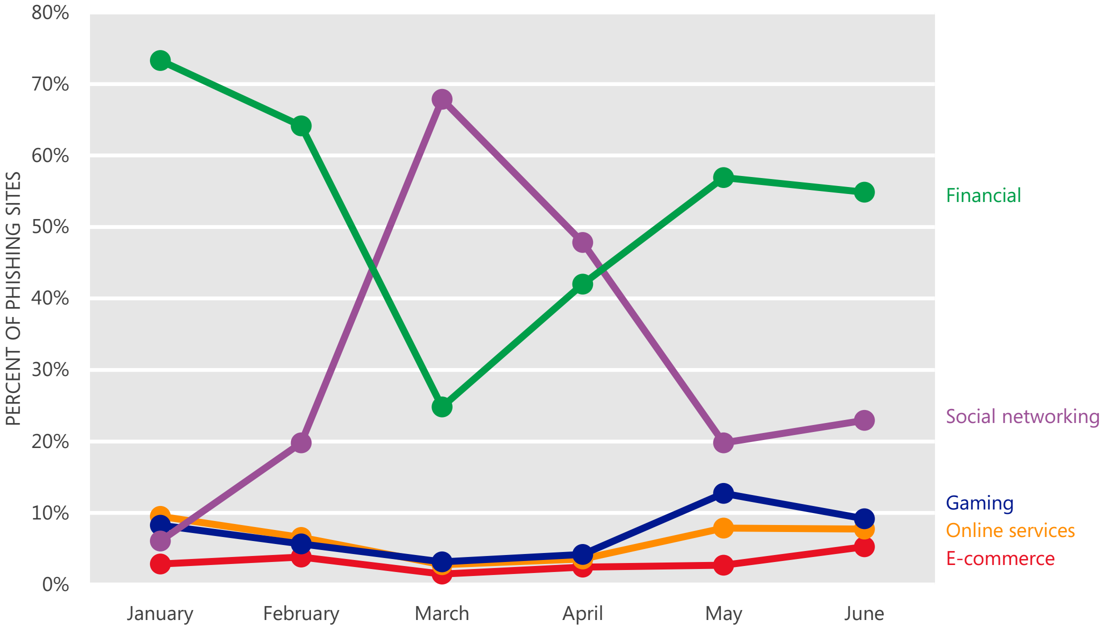


Figure 49. Active phishing sites tracked each month, January–June 2012, by type of target



- Typically, sites that target financial institutions account for most active phishing sites at any given time, often by a wide margin. In 1H12, a significant short-term campaign or campaigns that began in February resulted in sites that targeted social networks outnumbering sites that targeted financial institutions in March and April before returning to a more typical level in May.
- Impressions for phishing sites that target social networks peaked in April, commensurate with the elevated numbers of active social networking phishing sites observed around the same time.

Global distribution of phishing sites

Phishing sites are hosted all over the world on free hosting sites, on compromised web servers, and in numerous other contexts. Performing geographic lookups of IP addresses in the database of reported phishing sites makes it possible to create maps that show the geographic distribution of sites and to analyze patterns.

To provide a more accurate perspective on the phishing and malware landscape, the methodology used to calculate the number of Internet hosts in each country or region has been revised. For this reason, the statistics presented here should not be directly compared to findings in previous volumes.

Figure 50. Phishing sites per 1,000 Internet hosts for locations around the world in 1Q12 (top) and 2Q12 (bottom)

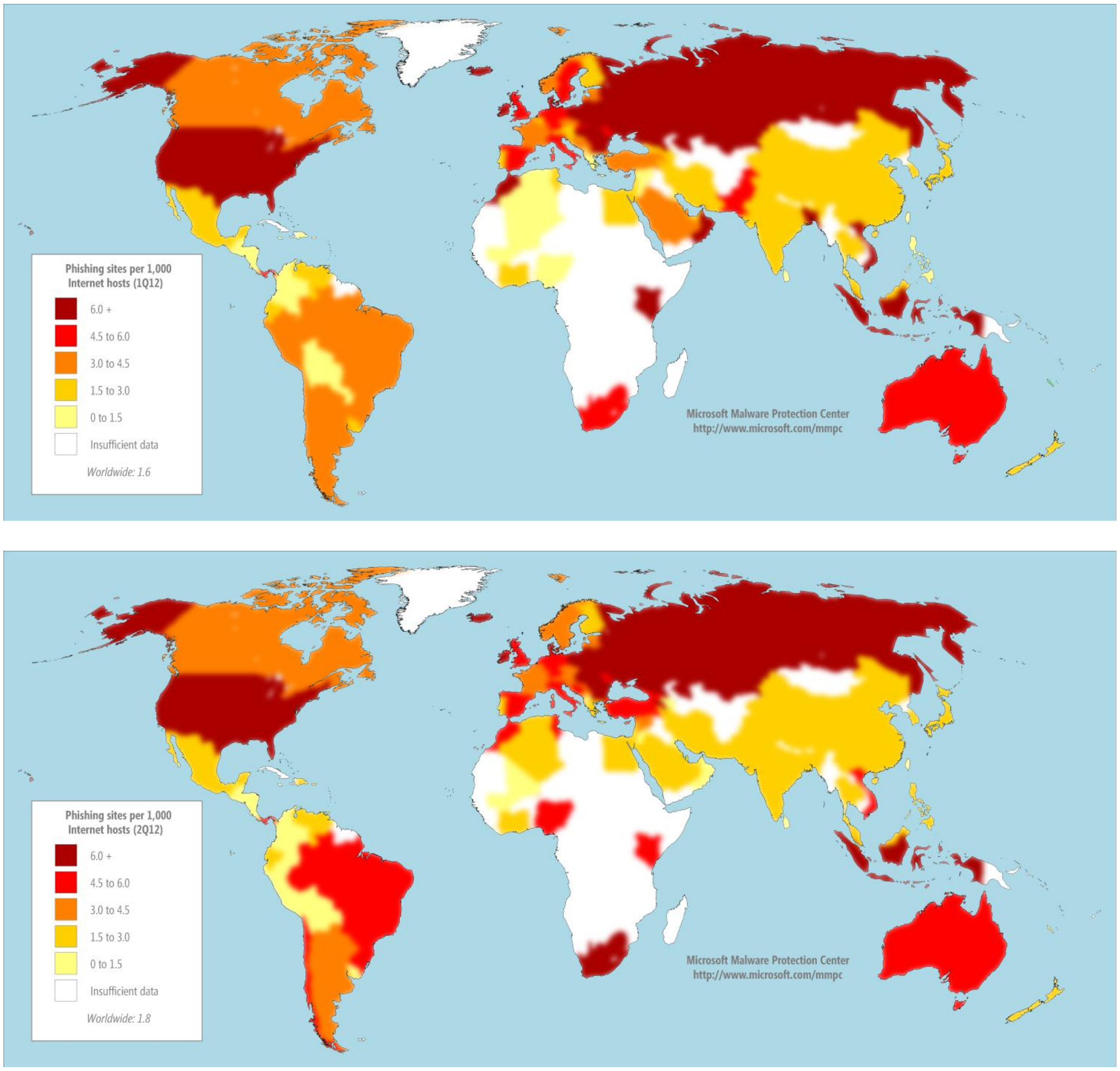
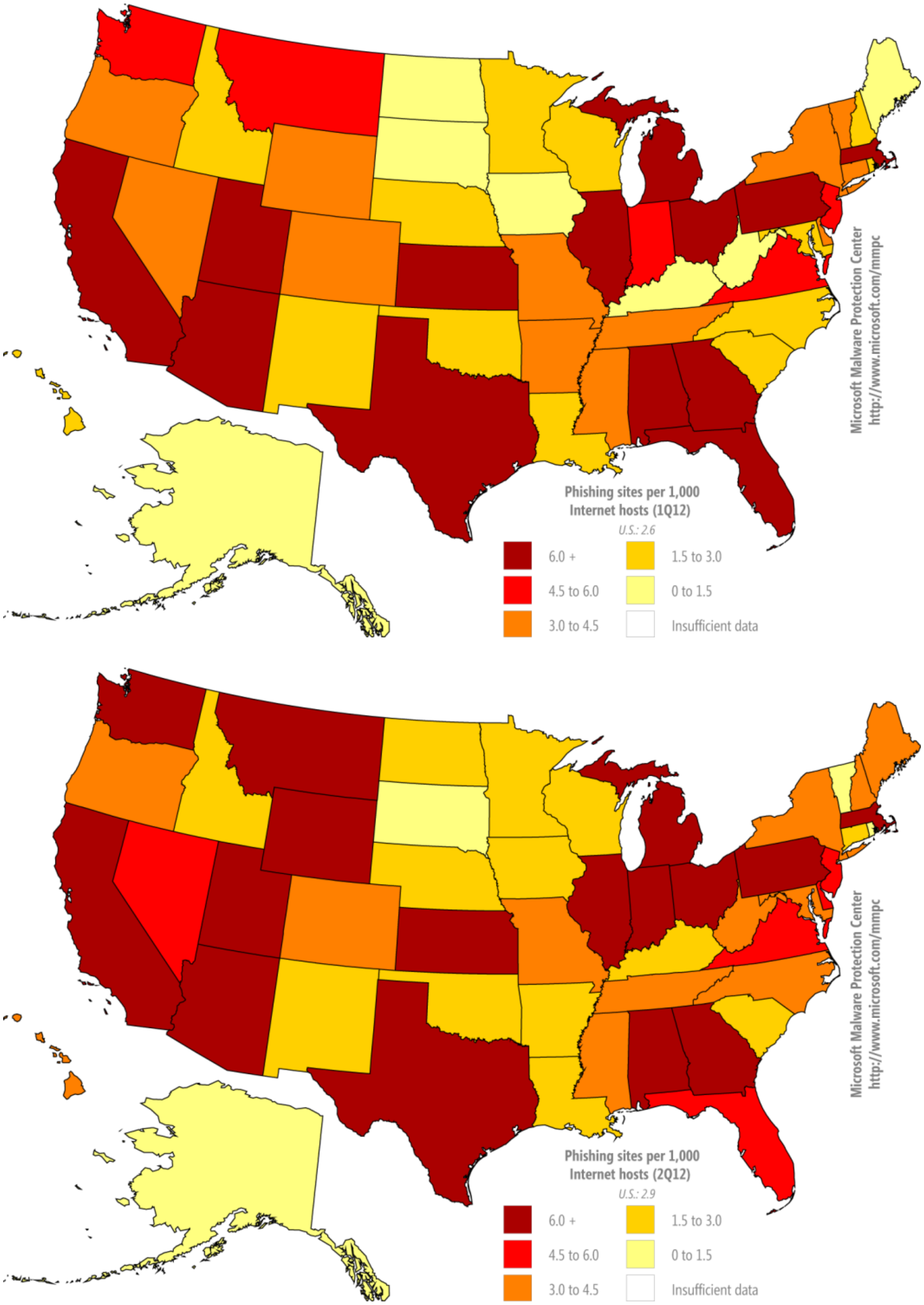

Phishing sites per 1,000 Internet hosts (1Q12)
6.0 +
4.5 to 6.0
3.0 to 4.5
1.5 to 3.0
0 to 1.5
Insufficient data
Worldwide: 1.6
Microsoft Malware Protection Center
http://www.microsoft.com/mmpc
Phishing sites per 1,000 Internet hosts (2Q12)
6.0 +
4.5 to 6.0
3.0 to 4.5
1.5 to 3.0
0 to 1.5
Insufficient data
Worldwide: 1.8
Microsoft Malware Protection Center
http://www.microsoft.com/mmpc

Figure 51. Phishing sites per 1,000 Internet hosts for US states in 1Q12 (top) and 2Q12 (bottom)

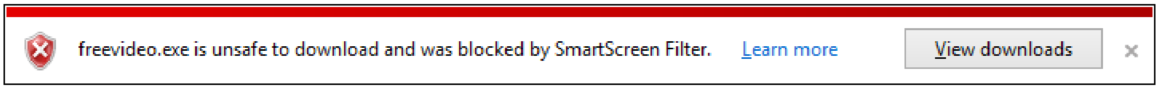


- SmartScreen Filter detected 1.6 phishing sites per 1000 Internet hosts worldwide in 1Q12, and 1.8 per 1000 in 2Q12.
- There is little correlation between the number of Internet hosts in a country or region and the number of phishing sites detected there. The United States, which has the largest number of hosts, also has a large number of phishing sites (2.9 per 1000 Internet hosts in 2Q12); China, with the second largest number of hosts, has a much lower concentration of phishing sites (0.6 per 1000 Internet hosts).
- Locations with high concentrations of phishing sites include Romania (3.8 per 1000 Internet hosts in 2Q12), Russia (3.4), and the United States (2.9). Locations with low concentration of phishing sites include Taiwan (0.4), Colombia (0.4), and China (0.7).
- In the United States, as a general rule, states with more Internet hosts tend to have higher concentrations of phishing sites as well, although there are plenty of exceptions.
- US states with high concentrations of phishing sites include Utah (8.6 per 1000 Internet hosts in 2Q12), Georgia (5.8), and Arizona (5.2). States with low concentrations of phishing sites include Iowa (0.6), Kentucky (1.0), and Minnesota (1.0).

Malware hosting sites

SmartScreen Filter in Internet Explorer helps provide protection against sites that are known to host malware, in addition to phishing sites. SmartScreen Filter uses URL reputation data and Microsoft antimalware technologies to determine whether those sites distribute unsafe content. As with phishing sites, Microsoft keeps track of how many people visit each malware hosting site and uses the information to improve SmartScreen Filter and to better combat malware distribution.

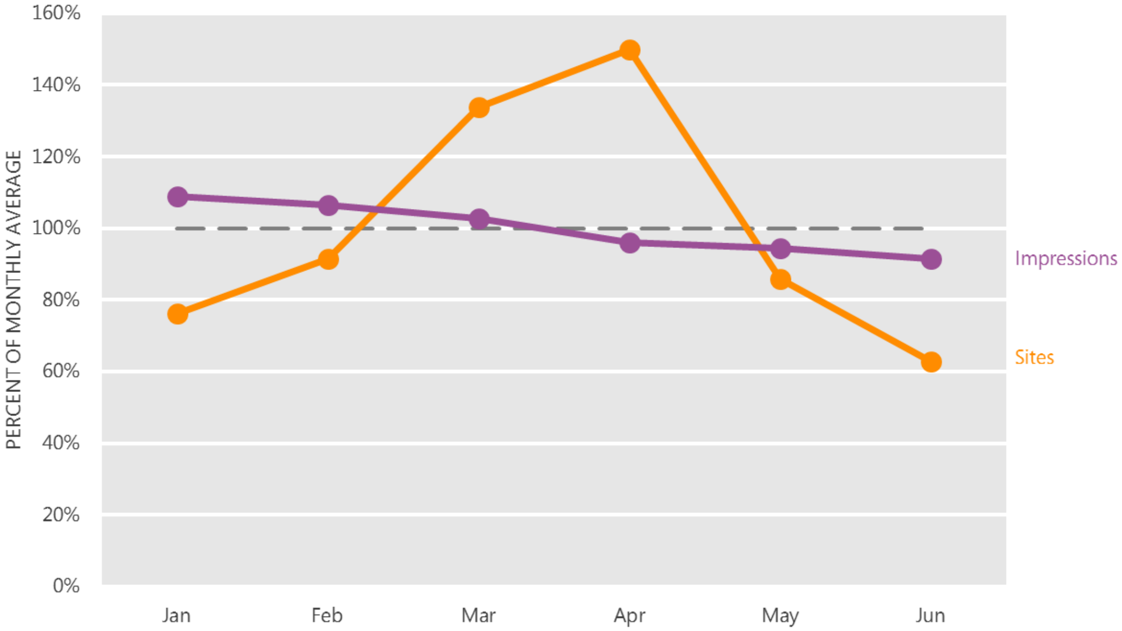
Figure 52. SmartScreen Filter in Internet Explorer displays a warning when a user attempts to download an unsafe file



freevideo.exe is unsafe to download and was blocked by SmartScreen Filter.

Figure 53 compares the volume of active malware hosting sites in the Microsoft URL Reputation Service database each month with the volume of malware impressions tracked by Internet Explorer.

Figure 53. Malware hosting sites and impressions tracked each month in 1H12, relative to the monthly average for each



- As with phishing, malware hosting impressions and active sites rarely correlate strongly with each other, and months with high numbers of sites and low numbers of impressions (or vice versa) are not uncommon.

Malware categories

Figure 54 and Figure 55 show the types of threats hosted at URLs that were blocked by SmartScreen Filter in 1H12.

Figure 54. Categories of malware found at sites blocked by SmartScreen Filter in 1H12, by percent of all malware impressions

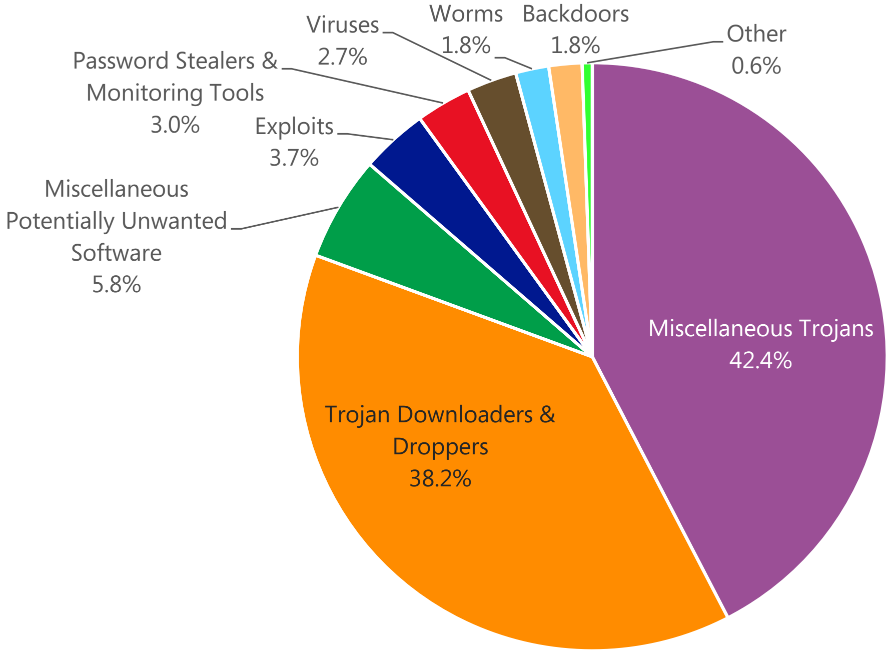


Figure 55. Top families found at sites blocked by SmartScreen Filter in 1H12, by percent of all malware impressions

	Family	Most significant category	Percent of malware impressions
1	Win32/Swisyn	Trojan Downloaders & Droppers	24.1%
2	Win32/Meredrop	Miscellaneous Trojans	9.0%
3	Win32/Bumat	Miscellaneous Trojans	5.9%
4	Win32/Microjoin	Trojan Downloaders & Droppers	4.2%
5	Win32/Obfuscator	Miscellaneous Potentially Unwanted Software	3.4%
6	Win32/Startpage	Miscellaneous Trojans	2.9%
7	Blacole	Exploits	2.8%
8	Win32/Orsam	Miscellaneous Trojans	2.2%
9	Win32/Banload	Trojan Downloaders & Droppers	2.2%
10	Win32/Malagent	Miscellaneous Trojans	1.9%
11	JS/IframeRef	Misc. Trojans	1.9%
12	Win32/VB	Worms	1.8%
13	Win32/Dynamer	Miscellaneous Trojans	1.6%
14	Win32/Sisproc	Miscellaneous Trojans	1.5%
15	JS/BlacoleRef	Miscellaneous Trojans	1.0%

- Most of the families on the list are generic detections for a variety of threats that share certain identifiable characteristics.

- Win32/Swisyn, the family responsible for the most malware impressions in 1H12, is a family of trojans that drops and executes malware on infected computers. These files may be embedded as resource files, and are often bundled with legitimate files in an effort to evade detection. Sites hosting Swisyn accounted for 24.1 percent of malware impressions in 1H12, an increase from 10.4 percent in 2H11.

- Win32/Meredrop, in second place, is a generic detection for trojans that drop and execute multiple forms of malware on local computers. These trojans are usually packed, and may contain multiple trojans, backdoors, or worms. Dropped malware may connect to remote websites and download additional malicious programs. Sites that host Meredrop accounted for 9.0 percent of malware impressions in 1H12, an increase from 1.8 percent in 2H11.

- Win32/Startpage, found on sites that accounted for 15.7 percent of malware impressions in 2H11, decreased to 2.9 percent in 1H12. Startpage is a generic detection for malware that changes the home page of an affected user's web browser without consent.

Global distribution of malware hosting sites

As with phishing sites, Figure 56 and Figure 57 show the geographic distribution of malware hosting sites reported to Microsoft in 1H12.

Figure 56. Malware distribution sites per 1,000 Internet hosts for locations around the world in 1Q12 (top) and 2Q12 (bottom)

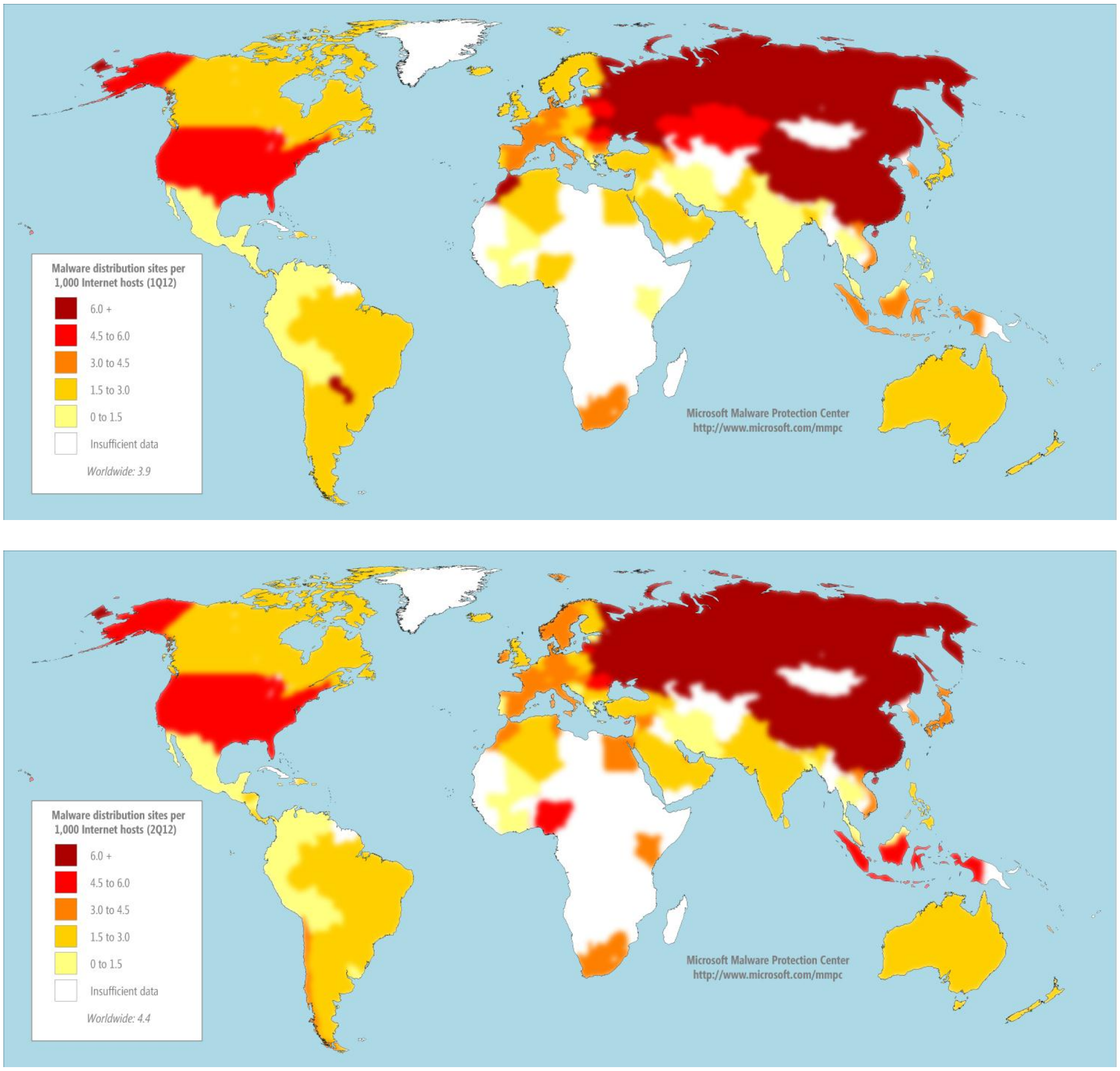
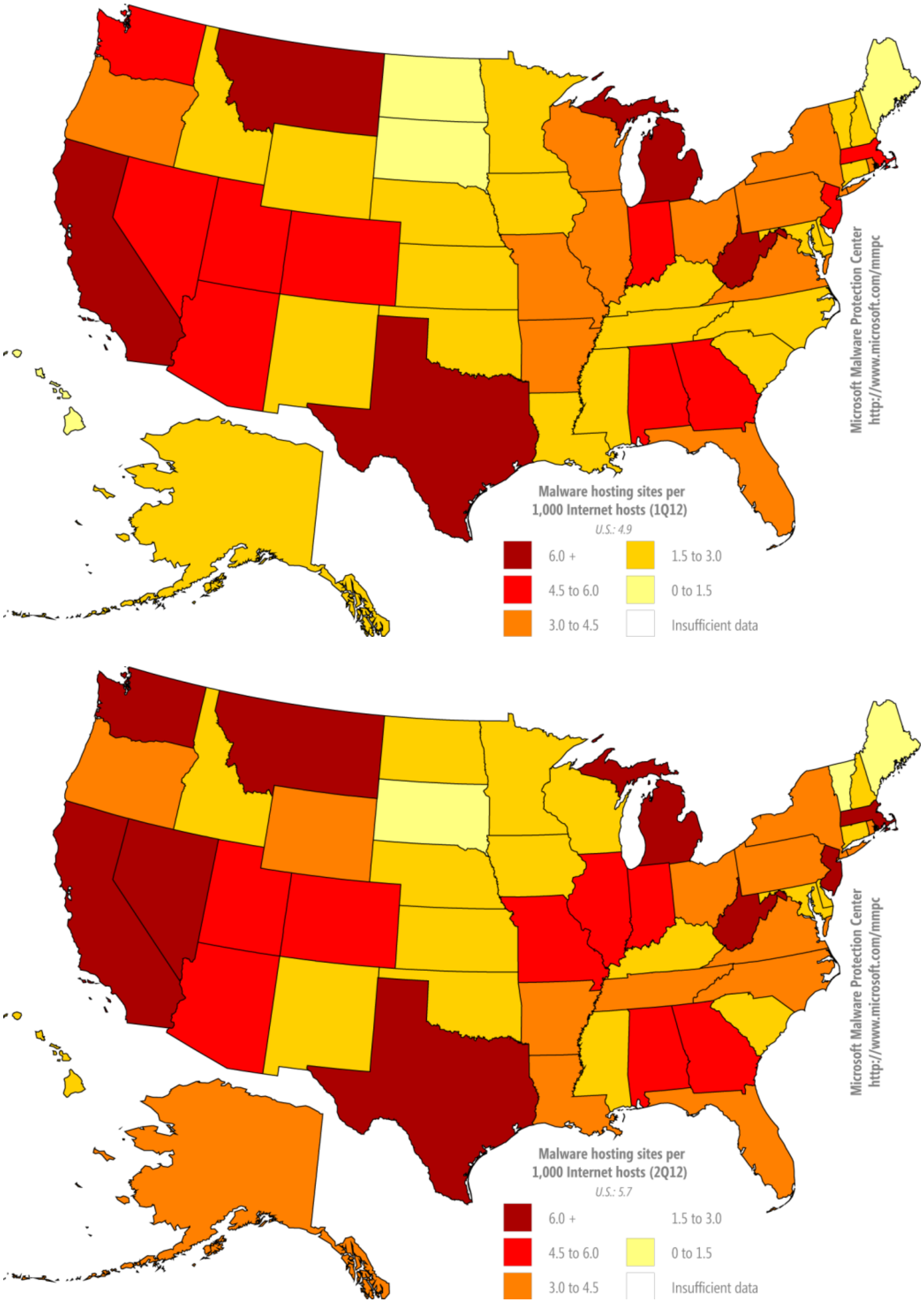

Malware distribution sites per 1,000 Internet hosts (1Q12)
6.0 +
4.5 to 6.0
3.0 to 4.5
1.5 to 3.0
0 to 1.5
Insufficient data
Worldwide: 3.9
Microsoft Malware Protection Center
http://www.microsoft.com/mmpc
Malware distribution sites per 1,000 Internet hosts (2Q12)
6.0 +
4.5 to 6.0
3.0 to 4.5
1.5 to 3.0
0 to 1.5
Insufficient data
Worldwide: 4.4
Microsoft Malware Protection Center
http://www.microsoft.com/mmpc

Figure 57. Malware distribution sites per 1,000 Internet hosts for US states in 1Q12 (top) and 2Q12 (bottom)


Malware hosting sites per 1,000 Internet hosts (1Q12)
U.S.: 4.9
6.0 +
1.5 to 3.0
4.5 to 6.0
0 to 1.5
3.0 to 4.5
Insufficient data
Microsoft Malware Protection Center
http://www.microsoft.com/mmpc
Malware hosting sites per 1,000 Internet hosts (2Q12)
U.S.: 5.7
6.0 +
1.5 to 3.0
4.5 to 6.0
0 to 1.5
3.0 to 4.5
Insufficient data
Microsoft Malware Protection Center
http://www.microsoft.com/mmpc

- Sites that host malware were significantly more common than phishing sites in 1H12. SmartScreen Filter detected 3.9 phishing sites per 1000 Internet hosts worldwide in 1Q12, and 4.4 per 1000 in 2Q12.
- China, with one of the lowest concentrations of phishing sites in the world in 1H12, also had the highest concentration of malware hosting sites (8.1 malware hosting sites per 1000 Internet hosts in 2Q12). Other locations with large concentrations of malware hosting sites included Russia (7.7), the United States (5.6), and Romania (5.5). Locations with low concentrations of malware hosting sites included Thailand (1.2), Malaysia (1.3), and Mexico (1.5).
- As with phishing sites, US states with more Internet hosts tend to have higher concentrations of phishing sites as well, although there are many exceptions.
- US states with high concentrations of malware hosting sites include California (8.4 per 1000 Internet hosts in 2Q12), Michigan (8.2), and Texas (7.0). States with low concentrations of phishing sites include Kentucky (2.0), South Carolina (2.2), and Oklahoma (1.9).

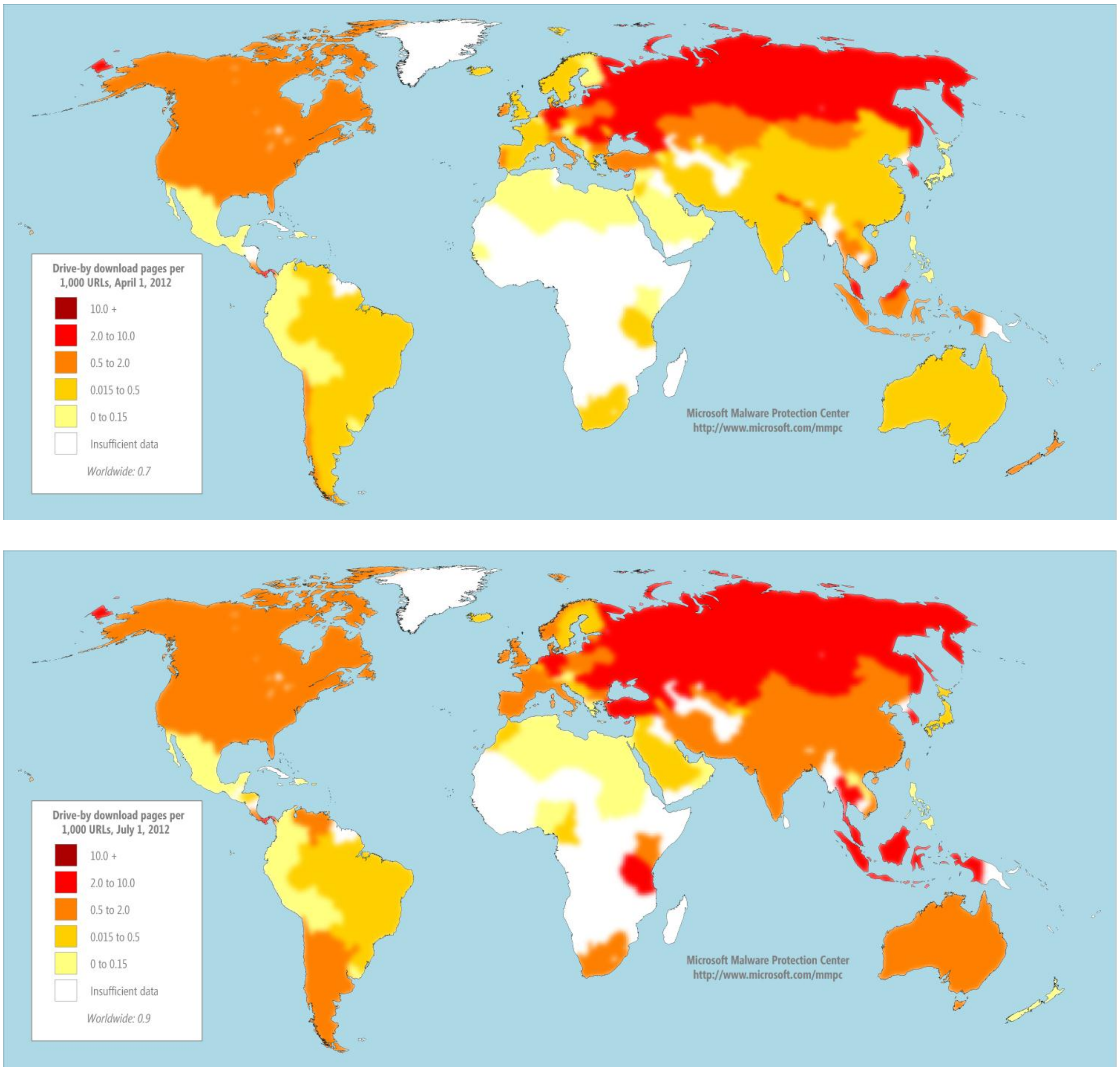
Drive-by download sites

A *drive-by download* site is a website that hosts one or more exploits that target vulnerabilities in web browsers and browser add-ons. Users with vulnerable computers can be infected with malware simply by visiting such a website, even without attempting to download anything.

Search engines such as Bing have taken a number of measures to help protect users from drive-by downloads. Bing analyzes websites for exploits as they are indexed and displays warning messages when listings for drive-by download pages appear in the list of search results. (See Drive-By Download Sites at the *Microsoft Security Intelligence Report* website for more information about how drive-by downloads work and the steps Bing takes to protect users from them.)

Figure 58 shows the concentration of drive-by download pages in countries and regions throughout the world at the end of 1Q12 and 2Q12, respectively.

Figure 58. Drive-by download pages indexed by Bing.com at the end of 1Q12 (top) and 2Q12 (bottom), per 1000 URLs in each country/region


Drive-by download pages per 1,000 URLs, April 1, 2012
10.0 +
2.0 to 10.0
0.5 to 2.0
0.015 to 0.5
0 to 0.15
Insufficient data
Worldwide: 0.7
Microsoft Malware Protection Center
http://www.microsoft.com/mmpc
Drive-by download pages per 1,000 URLs, July 1, 2012
10.0 +
2.0 to 10.0
0.5 to 2.0
0.015 to 0.5
0 to 0.15
Insufficient data
Worldwide: 0.9
Microsoft Malware Protection Center
http://www.microsoft.com/mmpc

- Each map shows the concentration of drive-by download URLs tracked by Bing in each country or region on a reference date at the end of the associated quarter, expressed as the number of drive-by download URLs per every 1,000 URLs hosted in the country/region.
- Significant locations with high concentrations of drive-by download URLs in both quarters include Malaysia, with 5.7 drive-by URLs for every 1,000 URLs

tracked by Bing at the end of 2Q12; Ukraine, with 5.1; Germany, with 3.9; and Korea, with 3.1.

Guidance: Protecting users from unsafe websites

Organizations can best protect their users from malicious and compromised websites by mandating the use of web browsers with appropriate protection features built in and by promoting safe browsing practices. For in-depth guidance, see the following resources in the "Managing Risk" section of the *Microsoft Security Intelligence Report* website:

- Promoting Safe Browsing
- Protecting Your People

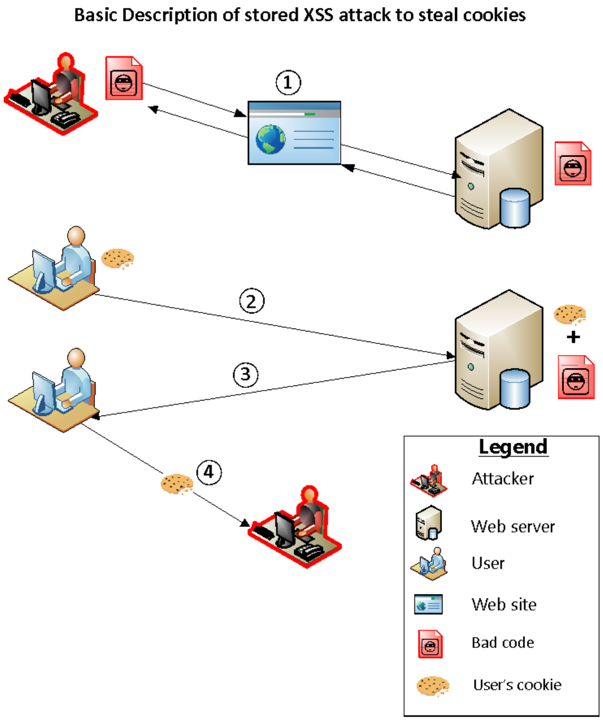
Mitigating risk

Cross-site scripting

Cross-site scripting (XSS) attacks have become the most prevalent and dangerous security issue affecting web applications. XSS vulnerabilities occur whenever an application takes data that originated from a user and sends it to a web browser without first properly validating or encoding it. XSS attacks can be used to hijack user sessions, deface websites, conduct port scans on victims' internal networks, conduct phishing attacks, and take over users' browsers.

In a typical XSS attack, an attacker causes a malicious script to execute in a prospective victim's browser when visiting a legitimate website. In a *reflected* attack, the attacker tricks the victim into submitting the malicious script to the vulnerable site (for example, by visiting a specially crafted URL with the script embedded in the query string). In a *stored* attack, the attacker uploads the malicious script to a vulnerable website in such a way that the script will be exposed to subsequent visitors and execute in their browsers. Figure 59 illustrates how a basic stored XSS attack can be used to steal cookie files from a victim's computer.

Figure 59. Overview of a stored XSS attack

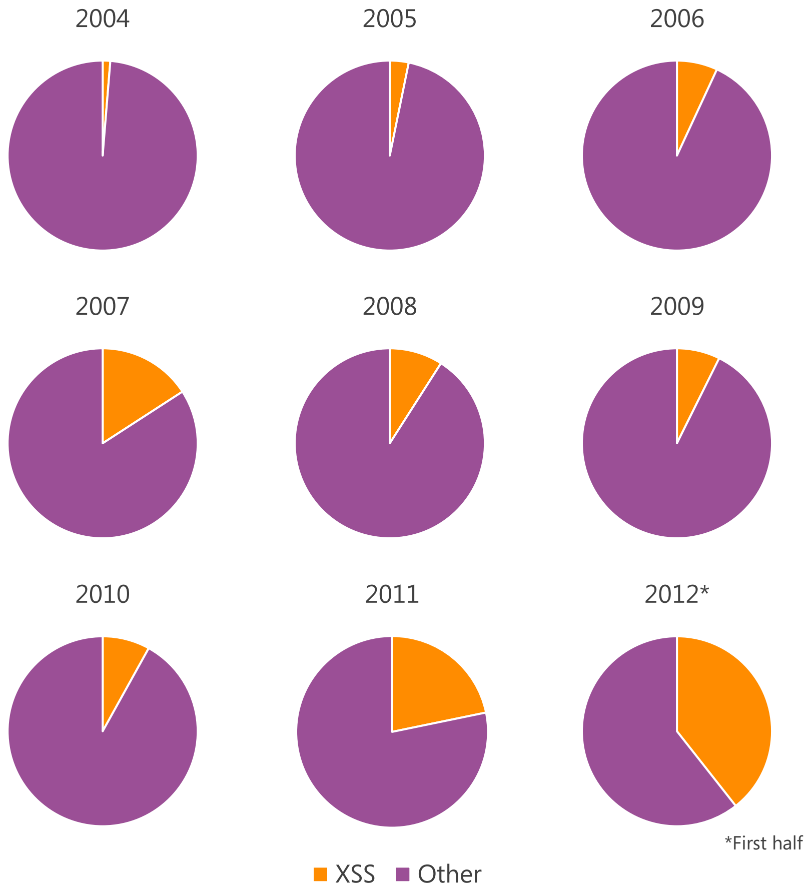


1. Attacker places bad code on a vulnerable Web site.
2. User navigates to the vulnerable Web site and submits a cookie.
3. The Web site allows the user to log on.
4. The malicious code sends the user's cookie to the attacker.

XSS trends

As Figure 60 illustrates, the MSRC has observed a significant increase in reported XSS cases within the past two years, to the point where XSS vulnerabilities have started to displace other types of reported vulnerabilities by percentage.

Figure 60. Portion of MSRC cases identified as involving XSS, 2004–2012, by year



Mitigating XSS with Windows Internet Explorer

Recent versions of Windows Internet Explorer have included a number of XSS mitigations, such as:

- **HttpOnly cookies**. Introduced in Internet Explorer 6 SP1 in 2002, HttpOnly is a flag that a website can set when sending a browser cookie to a client. If the HttpOnly flag is set for a cookie, it cannot be accessed by client-side scripts in a browser that supports the flag. The HttpOnly flag is supported by recent versions of most major desktop and mobile browsers. For more information, see the "HttpOnly" entry on the Open Web Application Security Project website (www.owasp.org).

- **XSS Filter**. Introduced in Windows Internet Explorer 8, the XSS Filter is a component of Internet Explorer that identifies and neutralizes likely XSS attacks. For more information, see the entry "IE8 Security Part IV: The XSS Filter" (July 2, 2008) at the Windows Internet Explorer Engineering Team blog (blogs.msdn.com/ie).

  An analysis of the vulnerabilities reported to the MSRC in 1H12 reveals that 37 percent of all verified vulnerabilities involved XSS techniques that the Internet Explorer XSS Filter can mitigate. For some perspective, another highly reported vulnerability class is *memory safety*, which accounts for 24 percent of vulnerabilities within the same data set. (Memory safety means buffer overflows and so on that may be exploited for code execution or information disclosure.)

- **SmartScreen Filter**. Introduced in Windows Internet Explorer 8, SmartScreen Filter blocks access to known harmful websites, which can include sites compromised by XSS. See "Malicious websites" beginning on page 75 for more information about SmartScreen Filter.

- **Inline frame security and the HTML5 Sandbox:** Website developers can specify a `sandbox` attribute for individual inline frames (IFrames). When a compliant browser loads a sandboxed IFrame with content from a domain other than the one hosting the frame, scripts and other potentially dangerous content are disabled. Microsoft provided a proprietary implementation of IFrame security starting with Internet Explorer 8, and the World Wide Web Consortium (W3C) has included a slightly different implementation as a proposed standard in the HTML5 Working Draft. Several modern browsers support the HTML5 model, including Internet Explorer 10. For more information, see the article "How to Safeguard your Site with HTML5 Sandbox" at the Microsoft Developer Network (msdn.microsoft.com).

Developments such as these, along with similar mitigations implemented by other browsers, can play a significant part in protecting users from XSS attacks. Nevertheless, the data suggests that XSS will remain a prominent threat for the immediate future. As long as it does, web browsers (along with infrastructure components like web application firewalls) will play an increasingly important role in defending against attacks.

For additional information about XSS and how to mitigate it, see the Cross-Site Scripting Quick Security Reference, which is available from the Microsoft Download Center (www.microsoft.com/download).

Defending against Pass-the-Hash attacks

Pass-the-Hash (PtH) attacks have become a staple in targeted attackers' toolkits. Many organizations have reported that PtH attacks have been used in attempted attacks on their organization. This section summarizes what PtH attacks are, the risks they present, and how a Windows environment can defend against them.

How password hashes work

Well-designed authentication systems expend considerable effort to prevent unauthorized disclosure of passwords and other credentials. Storing and transmitting passwords in plaintext puts them at risk of exposure to hackers, eavesdroppers, and malware. To prevent such exposure, strong authentication systems use multiple mechanisms to reduce the likelihood that unencrypted credentials will be exposed, and to ensure that any authentication data that does get stored and transmitted will be of limited use to an attacker.

One of the fundamental security techniques used by authentication systems is the use of cryptographic hash functions to encode credentials for storage and transmission. A cryptographic hash function is an algorithm that transforms an unencrypted message, such as a password string, into an encrypted representation, called the hash value or simply the hash. If the hash function is sufficiently strong, hash values are impossible to decrypt in a reasonable amount of time using conventionally available computing power, and the probability of two different strings producing the same hash value is either zero or extremely small. Using a hash function as part of the authentication process means that the authenticating server never has to store unencrypted passwords, which is a serious security weakness. Instead, the server computes the hash value of the submitted password from a client computer (or accepts the hash from the client directly) and compares it to its own stored hash for the account making the request. If they match, the client is authenticated.

In Microsoft Windows, hashes are stored in one of two places: a local Security Accounts Manager (SAM) database and/or a networked Active Directory database (which is stored as a physical file called NTDS.DIT on each participating domain controller). Password hashes can be stored in one of four forms: LAN Manager (LM), NT, AES key, or Digest.

Figure 61. Hash schemes used in different versions of Windows

Hash scheme	Encryption	Key length	Versions of Windows
LM	DES	56-bit	Windows XP, Windows Server 2003*
NT	MD4	128-bit	All currently supported versions
Digest	MD5	128-bit	All currently supported versions
AES key†	AES	128-bit, 256-bit	Windows Vista, Windows 7, Windows 8, Windows Server 2008, Windows Server 2008 R2, Windows Server 2012

* Present but disabled by default in later versions of Windows. †Used only on domain-joined computers.

- LM is an obsolete hash scheme originally developed for Microsoft LAN Manager, a network operating system product that predated the Windows Server line of products. To provide backward compatibility with legacy resources, some older versions of Windows, including Windows XP and Windows Server 2003, were designed to store password hashes in both LM and NT forms by default. The LM hash is very weak, and Microsoft has long recommended that it be disabled. (See support.microsoft.com/kb/299656 for instructions for preventing Windows from storing LM hashes). The much stronger NT hash scheme is used by all currently supported versions of Windows.

- Windows Vista, Windows Server 2008, and later versions of Windows support the use of AES encryption for password keys in conjunction with the Kerberos authentication protocol. Windows 7, Windows 8, Windows Server 2008 R2, and Windows Server 2012 attempt to use Kerberos and AES for authentication by default.

- Digest hashes are stored in Active Directory only if the appropriate option (*Store passwords using reversible encryption*) is enabled, and can be used for Internet Information Service (IIS) 6.0 (and earlier) digest authentication.

Credential hashes are the ultimate authentication verifier, and an attacker who is able to obtain a hash and successfully present it to the authentication server can assume whatever security identity is associated with the hash. A good authentication implementation attempts to make it difficult for malicious actors to access stored or transmitted hashes. Microsoft has never provided an intentional way for any user to access stored credentials, although several tools have been discovered that are designed to give attackers access to stored hashes, often using Local Security Authority (LSA) process injection. Even then, the tools require Local System, local Administrator, or domain Administrator account level access to be successful—in other words, the attacker needs to have already compromised the computer or network before these tools can be used to harvest and use hashes.[12]

With built-in Windows logon authentication, neither the password nor its representative hash is ever communicated from the originating logon computer to the target computer that validates authentication. LAN Manager (LM), NTLMv1, and NTLMv2 Windows authentication protocols use a challenge-response feature, which uses a cryptographic operation to prove they know the password without ever disclosing the password. Kerberos authentication employs a similar mechanism, but also uses encrypted timestamps to discourage replay attacks (in which an eavesdropper captures traffic to an authentication server, and repeats it to the server later to gain access).

LM and NTLMv1 authentication protocols contain known vulnerabilities, and Microsoft has long recommended that Windows computers use only the NTLMv2 or Kerberos authentication protocols. By default, supported versions of Windows use Kerberos whenever possible.

Even though authentication protocols don't directly communicate the hash, the originating client always has the hash and oftentimes the authenticating or host target must create or have access to the involved hash for successful authentication to occur.

These created or accessed hashes are stored in memory during logon authentication. The hashes often remain in memory after successful authentication, especially during an interactive session, so that future authentication can be done quickly if needed and without requiring the security principal (the entity requesting authentication, such as a user) to reenter the plaintext password. As a result, password hashes can be found in memory during active logon sessions (and sometimes after), as well as stored more permanently within the relevant authentication databases. Hashes are often deleted from

[12] For local SAM databases, the hacking tools typically use local Administrator privileges to gain access to the Local System account, and get into the database that way. Hacking tools that attempt to gain access to the Active Directory database must be run in the domain Administrator or Local System account on the domain controller to successfully access stored hashes.

memory when their associated logon sessions are terminated, but such deletion depends on the particulars of involved applications and does not always occur. Removing all hashes from memory may require a reboot.

Pass-the-hash attacks

PtH attacks involve two primary steps: obtaining the hashes, and using them to create new authenticated network logon sessions. An attacker that has compromised a target computer and obtained administrative permissions can access hashes in the stored authentication database or from memory. Microsoft has not received any verified reports of attackers using unprivileged accounts to access password hashes that belong to other accounts, either in memory or from the authentication databases.

Before the emergence of PtH attacks, attackers had to crack password hashes (recover their plaintext equivalent passwords) to use them. As LM hash use has declined in favor of stronger algorithms, password hash reversing has become more difficult to accomplish and made PtH more attractive to attackers.

Although the feasibility of using stolen password hashes to authenticate has long been known, the first public versions of PtH tools became available early in the year 2000. These tools automate the process of creating new authenticated logon sessions from stolen hashes. They allow attackers to skip the sometimes unsuccessful cracking step and move on more quickly to gaining unauthorized access to systems. Plaintext passwords are still more useful to the attacker than hashes—for example, they can be directly entered into most logon interfaces, which hashes cannot—but PtH techniques can often allow attackers to achieve most of the same outcomes, even when plaintext passwords cannot be obtained.

In a typical PtH scenario, an attacker uses social engineering to trick a user into running a trojan that gives the attacker backdoor access to a domain-joined computer. If the user is a member of the local Administrators group, the attacker can access the local SAM authentication database and dump the account names and password hashes of the local account. The attacker can also run PtH tools to dump password hashes of any locally or interactively logged-in sessions. (Non-network logon sessions, such as remote drive mappings, do not use or leave password hashes in memory on the target computer.)

The attacker uses the credentials and hashes obtained on the local computer to explore and gain access to additional computer systems. The goal in many cases is a hash associated with a domain administrator account, which will give the attacker access to an Active Directory domain controller and allow the attacker to dump the password hashes of the entire domain, or even an entire forest.

It's important to remember that PtH (or similar) attacks are possible in any computer environment, regardless of which operating systems are in use. All computer systems store authentication secrets. If an attacker can obtain the highest privileged access on the host operating system, that attacker can gain access to the stored authentication secrets on that computer and other computers with shared secrets. Susceptibility to PtH attacks are an inherent part of any password-based or tokenized authentication solution. This susceptibility is true of any operating system and almost every authentication system.

The latter point is important. Even though current PtH methods concentrate on password hashes, similar attacks could (and can) work against other authentication mechanisms, including tokens, delegation, and two-factor authentication. If the attacker is able to capture the ultimate secret—be it a password hash, a token, or some other entity—the attack will succeed. Many authentication defenses work by not allowing authentication session reuse (that is, replay), but PtH attacks work by capturing the ultimate authentication secret and creating new sessions. Replay attacks are much easier to defend against than attacks in which the ultimate authentication secret is stolen.

After hashes are stolen, they can be used repeatedly, and their theft and use can be difficult to detect. By the time a customer is aware of PtH attacks, the attackers may have had access to the computer or network for months and may have accessed confidential information. Malicious hackers could have installed backdoor programs, modified operating systems, and ensured that it will be very difficult for the victims to regain an assured healthy state.

Pass-the-hash defenses

So what can be done to reduce the risk of PtH attacks? This section of the article summarizes the official Microsoft recommendations to decrease the risk of PtH attacks. Many mitigation controls are possible, but not all are equally easy to implement by all customers. Implement suggested mitigations only after careful consideration and testing.

Prevent attackers from gaining local administrator access

The real problem faced by victims of PtH attacks is that the attackers were able to gain access to privileged accounts in the first place. Password hashes must be stored in memory on computers in single-sign-on environments, so there is little an IT department can do to prevent an attacker with administrator or system permissions from stealing the hashes and other credentials entrusted to the local computer. In addition, the level of access that is required to steal password hashes from an active session would also allow attackers to capture logon credentials using other mechanisms, such as a keylogger. Therefore, even if PtH attack methods could be completely stopped, attackers could still accomplish the same outcomes through equally powerful means.

More than a decade ago, the MSRC formulated a set of computer security observations called Ten Immutable Laws of Security[13] that covered a number of principles that apply to PtH attacks:

- Law #1: If a bad guy can persuade you to run his program on your computer, it's not solely your computer anymore.
- Law #6: A computer is only as secure as the administrator is trustworthy.
- Law #10: Technology is not a panacea.

The most powerful defense is to prevent attackers from taking advantage of users' permission levels in the first place. If users aren't local administrators, running an attacker's trojan won't automatically give the attacker the level of access required to do password hash dumps. Without privileged access, the attacker may still be able to gain access to the current user's logged on account but they won't be as easily able to capture and use other hashes.

To gain privileged access when an attacker only has the current user's non-privileged access (called an escalation of privilege attack), the attacker would need to use an additional exploit that escalates their privilege. A fully updated computer makes escalation-of-privilege attacks more difficult to accomplish.

Enable User Account Control

The ideal solution for user workstations is to have user accounts run without local administrator permissions. However, this requirement may be something that many customers are unwilling to undertake. The User Account Control (UAC)

[13] Visit technet.microsoft.com/library/hh278941.aspx to read an updated version of this essay.

technology built into Windows 7 and other recent Windows releases can provide some benefits by providing an additional barrier that an attacker must overcome. When UAC is enabled, members of administrator groups run with standard permissions most of the time and must explicitly grant permission for a program or process (such as an attacker's trojan) to run with administrator rights. This functionality can help prevent an attacker from gaining the necessary permission level to access the hashes stored in authentication databases.

Minimize the membership of privileged groups

Minimize the number and type of computers that members of privileged groups are allowed to log on to. For example:

- Prevent members of the Domain Admins group from logging on to non-domain controllers.
- Prevent local Administrators (and other local accounts with elevated permissions) from performing network logons.
- Prevent elevated accounts from logging on to any computers except the ones they need.

Use role-based delegation

Instead of granting permanent privileged group membership to Administrator accounts, use a role-based delegation strategy to grant accounts the specific permissions that they require. The Delegation of Control Wizard in Active Directory Users and Computers provides hundreds of predefined tasks that can be delegated to individual user accounts or groups. Create role-based groups and assign one or more delegated tasks to each role, and use organizational units (OUs) to control the logical areas in which each delegated task can be performed.

For example, if level 1 help desk personnel need the ability to reset user passwords and modify user accounts, adding them to the Domain Admins group would create significant risk by giving them rights to perform hundreds of privileged tasks that their job doesn't require them to do. Instead, assign them to a role-based group that can reset user passwords and modify user accounts for anyone in the domain. Similarly, you can create roles for server operators to administer only specific servers, and so on. By giving every account holder only the minimum permissions they need to perform their duties, you can significantly reduce the risk the organization faces from PtH and similar attacks.

See technet.microsoft.com/library/dd145344 for Delegation of Control Wizard documentation that applies to recent versions of Windows Server, and see "Best Practices for Delegating Active Directory Administration," available from the Microsoft Download Center, for a comprehensive overview of delegation in Active Directory (written for older versions of Windows Server, but containing much valuable information for anyone who is new to the concepts.)

Minimize the use of privileged user rights assignments

Privileged group membership is not always necessary to elevate users' permissions. Microsoft considers the following permissions to be elevated:

- Create token object (SeCreateTokenPrivilege)
- Act as part of the operating system (SeTcbPrivilege)
- Take ownership (SeTakeOwnershipPrivilege)
- Back up files and directories (SeBackupPrivilege)
- Restore files and directories (SeRestorePrivilege)
- Debug programs (SeDebugPrivilege)
- Impersonate client after authentication (SeImpersonate)
- Modify object label (SeRelabelPrivilege)
- Load and unload device drivers (SeLoadDriverPrivilege)

Any security principal account, even without belonging to a privileged group, can perform elevated tasks, up to and including gaining complete control of the domain or forest. Therefore, it's considered a best practice to minimize the number of security principals who are granted permanent elevated permissions. Instead, use delegation as suggested earlier or grant elevated permissions only when needed.

Restrict logons to additional computers

An attacker with sufficient permissions can use stolen hashes to move from computer to computer in an organization. You can mitigate this risk by using various methods to restrict logons, as shown in Figure 62.

Figure 62. Mitigating PtH attacks by restricting logons

Mitigation	Local Accounts	Domain Accounts
Restricting the logon rights for privileged accounts stored in the local SAM: • Denying network logon • Denying RDP logon	•	
Randomizing the passwords for built-in accounts in the local SAM	•	
Disabling the local accounts	•	
Granting permissions to workstation administrators only on a temporary basis		•

Use isolated management computers

One of the greatest risks in any environment occurs when privileged accounts log on to computers that are not under complete control of the privileged account, especially computers that have access to Internet browsing or email. Every time a privileged account logs on to a high-risk computer, it increases the chances that the privileged accounts credentials will be stolen.

To reduce risk, privileged accounts should only be used to log on to highly secure, dedicated management workstations. These computers (sometimes known as jump boxes or trusted computers) should not be used for any other purpose, and should not be able to access the Internet; if Internet access is needed, it should be limited to a few defined websites. Such a management computer can be used to safely run privileged processes, or to run other processes against remote computers that are less susceptible to PtH attacks (discussed later in this section). Many organizations use virtual machines as management computers, resetting them after each and every logout. Access to management computers should be monitored and audited.

Reboot computers after logging on using privileged credentials

Most of the time, password hashes are removed from memory after an active logon session is terminated. However, password hash removal depends on the involved applications and processes, and some malfunctioning applications may not remove the hashes. Therefore, it is possible for password hashes to remain after the privileged user has logged out. It is also common for privileged users to end remote sessions without logging out, thereby leaving the active session open.

Whenever possible, privileged users should reboot computers after they accomplish their tasks to ensure that password hashes do not remain in memory for attackers to access. Rebooting may not be possible on highly used production computers, but it should be done whenever possible.

Minimize elevated remote interactive network logins

One of the most common administration techniques is to use Remote Desktop or Terminal Services to log on to a computer to administer it. Because these types of logons are interactive, the password hashes of the logged on user are kept in memory for at least the duration of the session. Therefore, administrators should minimize the use of remote, interactive logons (as well as batch and service logons, which have the same issues). Instead, use other, non-interactive methods of administration, including the ones in the following list.

- **MMC snap-ins**. Administrators can use Microsoft Management Console (MMC) snap-ins to perform operations on remote computers as they would locally, without leaving password hashes on the target computer for attackers to steal.
- **Windows PowerShell**. Using PowerShell (without CredSSP) is an excellent way to administer targeted computers without leaving password hashes in memory. Many tasks can be completely accomplished using PowerShell cmdlets.
- **Other tools**. Many other remote management tools can be used to administer computers remotely without using a full interactive session. For example, the PsExec tool, available for download from the Windows Sysinternals site at technet.microsoft.com/sysinternals, can be used to perform remote administrative tasks. Although it is theoretically possible for an attacker to steal the logon credentials or token while the task is operating, the credentials or token are removed from memory when the task finishes and exits.

Minimize password reuse

Password reuse across multiple computers creates some of the greatest risk from PtH attacks. Many customers use scripts and third-party tools to either ensure that no password is reused over multiple computers (for example, by requiring a different password for every local Administrator account on each computer), or to require one-time passwords (OTP) or password changes after every session. Any OTP hashes that might be stolen by an attacker cannot be used to create future active sessions, which minimizes the attacker's ability to access other computers. (Note, however, that many third-party OTP management tools use privileged accounts to reset passwords, which can themselves become vulnerable to attack.)

Scan for and eliminate PtH tools

Most antimalware scanners detect many common PtH tools. Make sure your antimalware scanner is configured to detect hacking tools, and to look specifically for the presence of PtH-related software.

Create separate security zones

PtH attacks often compromise all the password hashes in an entire domain or forest. If you are worried about one successful PtH campaign in a weakly secured domain leading to a compromise of a more strongly secured domain, consider creating a separate forest. Forests are security boundaries in Active Directory. Having separate forests reduces the potential spread of a single PtH attack campaign. Maintaining separate forests can impose additional administrative costs and responsibilities, but the additional overhead may be justified in some scenarios.

Check for leftover password hashes in memory

Although no currently supported Microsoft software products leave password hashes in memory after the conclusion of an interactive session, some older or third-party tools might. If you are concerned about this problem, use existing tools to look for password hashes in memory after the active sessions have ended. The LogonSessions tool, available for download from the Windows Sysinternals site at technet.microsoft.com/sysinternals, lists the currently active logon sessions on a computer, which can help you locate problem software that leaves hashes in memory.

Summary

Successful PtH attacks can create a significant degree of risk in any environment. Although they require privileged access to begin the attacks, they can allow an attacker to gain a large amount of control over a domain or forest if administrative practices are not adapted to thwart attacks. Following the guidelines presented in this section can help you mitigate the risk your organization faces from PtH attacks and minimize any potential resulting damage.

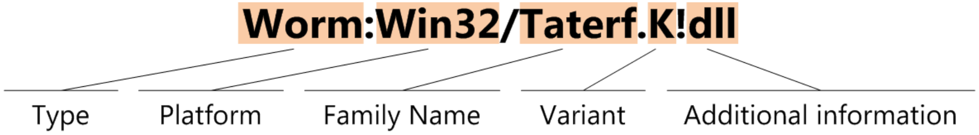
Appendixes

Appendix A: Threat naming conventions

The MMPC malware naming standard is derived from the Computer Antivirus Research Organization (CARO) Malware Naming Scheme, originally published in 1991 and revised in 2002. Most security vendors use naming conventions that are based on the CARO scheme, with minor variations, although family and variant names for the same threat can differ between vendors.

A threat name can contain some or all of the components seen in Figure 63.

Figure 63. The Microsoft malware naming convention



The *type* indicates the primary function or intent of the threat. The MMPC assigns each individual threat to one of a few dozen different types based on a number of factors, including how the threat spreads and what it is designed to do. To simplify the presentation of this information and make it easier to understand, the *Microsoft Security Intelligence Report* groups these types into 10 categories. For example, the TrojanDownloader and TrojanDropper types are combined into a single category, called Trojan Downloaders & Droppers.

The *platform* indicates the operating environment in which the threat is designed to run and spread. For most of the threats described in this report, the platform is listed as “Win32,” for the Win32 API used by 32-bit and 64-bit versions of Windows desktop and server operating systems. (Not all Win32 threats can run on every version of Windows, however.) Platforms can include programming languages and file formats, in addition to operating systems. For example, threats in the ASX/Wimad family are designed for programs that parse the Advanced Stream Redirector (ASX) file format, regardless of operating system. Some families have components that run on multiple platforms, in which case the most significant platform is usually the one given. In some contexts, a different platform might be listed for a family than one given elsewhere, when appropriate. In some rare cases, some predominantly multiplatform families may be listed without a platform, as with the exploit family Blacole.

Groups of closely related threats are organized into *families,* which are given unique names to distinguish them from others. The family name is usually not related to anything the malware author has chosen to call the threat. Researchers use a variety of techniques to name new families, such as excerpting and modifying strings of alphabetic characters found in the malware file. Security vendors usually try to adopt the name used by the first vendor to positively identify a new family, although sometimes different vendors use completely different names for the same threat, which can happen when two or more vendors discover a new family independently. The MMPC Encyclopedia (www.microsoft.com/mmpc) lists the names used by other major security vendors to identify each threat, when known.

Some malware families include multiple components that perform different tasks and are assigned different types. For example, the Win32/Frethog family includes variants designated TrojanDownloader:Win32/Frethog.C and PWS:Win32/Frethog.C, among others. In the *Microsoft Security Intelligence Report*, the category listed for a particular family is the one that Microsoft security analysts have determined to be the most significant category for the family (which, in the case of Frethog, is Password Stealers & Monitoring Tools).

Malware creators often release multiple *variants* for a family, typically in an effort to avoid being detected by security software. Variants are designated by letters, which are assigned in order of discovery—A through Z, then AA through AZ, then BA through BZ, and so on. A variant designation of "gen" indicates that the threat was detected by a generic signature for the family rather than as a specific variant. Any additional characters that appear after the variant provide comments or additional information.

In the *Microsoft Security Intelligence Report,* a threat name that consists of a platform and family name (for example, "Win32/Taterf") is a reference to a family. When a longer threat name is given (for example, "Worm:Win32/Taterf.K!dll"), it is a reference to a more specific signature or to an individual variant. To make the report easier to read, family and variant names have occasionally been abbreviated in contexts where confusion is unlikely. Thus, Win32/Taterf would be referred to simply as "Taterf" on subsequent mention in some places, and Worm:Win32/Taterf.K simply as "Taterf.K."

Appendix B: Data sources

Data included in the *Microsoft Security Intelligence Report* is gathered from a wide range of Microsoft products and services. The scale and scope of this telemetry data allows the report to deliver the most comprehensive and detailed perspective on the threat landscape available in the software industry:

- Bing, the search and decision engine from Microsoft, contains technology that performs billions of webpage scans per year to seek out malicious content. After such content is detected, Bing displays warnings to users about it to help prevent infection.
- Hotmail has hundreds of millions of active email users in more than 30 countries/regions around the world.
- Exchange Online Protection (formerly Forefront Online Protection for Exchange, or FOPE) protects the networks of thousands of enterprise customers worldwide by helping to prevent malware from spreading through email. FOPE scans billions of email messages every year to identify and block spam and malware.
- Microsoft System Center Endpoint Protection (formerly Forefront Endpoint Protection) is a unified product that provides protection from malware and potentially unwanted software for enterprise desktops, laptops, and server operating systems. It uses the Microsoft Malware Protection Engine and the Microsoft antivirus signature database to provide real-time, scheduled, and on-demand protection.
- Windows Defender is a program that is available at no cost to licensed users of Windows that provides real-time protection against pop-ups, slow performance, and security threats caused by spyware and other potentially unwanted software. Windows Defender runs on more than 100 million computers worldwide.
- The Malicious Software Removal Tool (MSRT) is a free tool that Microsoft designed to help identify and remove prevalent malware families from customer computers. The MSRT is primarily released as an important update through Windows Update, Microsoft Update, and Automatic Updates. A

version of the tool is also available from the Microsoft Download Center. The MSRT was downloaded and executed more than 600 million times each month on average in 1H12. The MSRT is not a replacement for an up-to-date antivirus solution because of its lack of real-time protection and because it uses only the portion of the Microsoft antivirus signature database that enables it to target specifically selected, prevalent malicious software.

- Microsoft Security Essentials is a free real-time protection product that combines an antivirus and antispyware scanner with phishing and firewall protection.
- The Microsoft Safety Scanner is a free downloadable security tool that provides on-demand scanning and helps remove malware and other malicious software. The Microsoft Safety Scanner is not a replacement for an up-to-date antivirus solution, because it does not offer real-time protection and cannot prevent a computer from becoming infected.
- SmartScreen Filter, a feature in Internet Explorer 8 and 9, offers users protection against phishing sites and sites that host malware. Microsoft maintains a database of phishing and malware sites reported by users of Internet Explorer and other Microsoft products and services. When a user attempts to visit a site in the database with the filter enabled, Internet Explorer displays a warning and blocks navigation to the page.

Figure 64. US privacy statements for the Microsoft products and services used in this report

Product or Service	Privacy Statement URL
Bing	www.microsoft.com/privacystatement/en-us/bing/default.aspx
Hotmail	privacy.microsoft.com/en-us/fullnotice.mspx
Exchange Online Protection	https://admin.messaging.microsoft.com/legal/privacy/en-us.htm
Malicious Software Removal Tool	www.microsoft.com/security/pc-security/msrt-privacy.aspx
System Center Endpoint Protection	www.microsoft.com/download/en/details.aspx?id=23308
Microsoft Security Essentials	windows.microsoft.com/en-US/windows/products/security-essentials/privacy
Microsoft Safety Scanner	www.microsoft.com/security/scanner/en-us/Privacy.aspx
Windows Internet Explorer 9	windows.microsoft.com/en-US/internet-explorer/products/ie-9/windows-internet-explorer-9-privacy-statement

Appendix C: Worldwide infection rates

"Global infection rates," on page 39, explains how threat patterns differ significantly in different parts of the world. Figure 60 shows the infection rates in locations with at least 100,000 quarterly MSRT executions in 1H12, as determined by geolocation of the IP address of the reporting computer.[14] CCM is the number of computers cleaned for every 1,000 executions of MSRT. See the *Microsoft Security Intelligence Report* website for more information about the CCM metric and how it is calculated.

For a more in-depth perspective on the threat landscape in any of these locations, see the "Regional Threat Assessment" section of the *Microsoft Security Intelligence Report* website.

Figure 65. Infection rates (CCM) for locations around the world, 3Q11–2Q12, by quarter

Country/Region	3Q11	4Q11	1Q12	2Q12
Worldwide	***7.7***	***7.1***	***6.6***	***7.0***
Afghanistan	—	11.7	10.3	10.6
Albania	19.3	25.0	27.5	25.7
Algeria	14.2	17.3	20.1	19.0
Angola	18.6	16.1	15.0	14.8
Argentina	8.3	8.3	8.7	7.2
Armenia	6.9	6.8	6.7	6.5
Australia	5.3	4.6	4.0	2.9
Austria	3.9	8.4	2.8	2.8
Azerbaijan	10.3	11.7	12.8	12.0
Bahamas, The	12.0	10.6	11.6	10.4
Bahrain	18.0	15.6	15.4	14.7

[14] For more information about this process, see the entry "Determining the Geolocation of Systems Infected with Malware" (November 15, 2011) on the Microsoft Security Blog (blogs.technet.com/security).

Country/Region	3Q11	4Q11	1Q12	2Q12
Bangladesh	14.9	17.0	15.6	15.1
Barbados	5.4	4.6	5.3	3.8
Belarus	6.3	5.6	5.4	7.2
Belgium	6.1	4.7	3.7	4.1
Bolivia	13.9	13.0	11.7	10.7
Bosnia and Herzegovina	13.4	15.8	16.6	14.9
Brazil	17.2	14.0	13.3	10.1
Brunei	9.6	9.1	8.0	7.9
Bulgaria	8.3	9.0	9.0	8.0
Burkina Faso	—	—	—	10.1
Cambodia	12.4	11.5	11.8	11.6
Cameroon	11.3	12.8	14.6	13.6
Canada	5.8	4.3	3.8	2.7
Chile	7.9	13.9	13.7	9.4
China	1.5	1.0	0.8	0.6
Colombia	8.7	7.8	8.3	7.3
Costa Rica	6.4	5.8	5.8	4.3
Côte d’Ivoire	12.9	13.3	15.2	12.7
Croatia	8.1	10.0	9.3	8.0
Cyprus	9.6	8.0	7.4	6.3
Czech Republic	2.6	2.3	2.1	1.8
Denmark	2.2	2.0	1.5	1.7
Dominican Republic	14.8	14.0	15.2	13.8
Ecuador	9.0	8.6	11.3	11.1
Egypt	17.5	22.7	24.7	23.4
El Salvador	8.1	6.5	6.8	6.1
Estonia	4.8	4.1	3.6	3.0
Ethiopia	9.8	9.2	9.7	10.5
Finland	1.8	1.6	1.1	1.1
France	4.2	3.8	3.2	2.9
Georgia	20.1	21.6	23.3	25.2
Germany	3.3	11.0	3.5	3.0
Ghana	10.5	11.6	11.9	11.2

Country/Region	3Q11	4Q11	1Q12	2Q12
Greece	9.5	8.5	7.3	6.3
Guadeloupe	9.7	9.1	9.6	9.6
Guatemala	8.8	7.1	8.0	6.9
Haiti	14.6	17.6	16.4	12.1
Honduras	10.2	9.4	9.1	8.5
Hong Kong SAR	5.6	4.4	3.5	2.6
Hungary	5.9	5.1	5.3	5.2
Iceland	4.4	3.7	3.2	2.4
India	15.0	13.8	13.2	12.6
Indonesia	18.7	18.6	17.0	16.6
Iraq	20.5	22.0	23.7	25.3
Ireland	4.8	3.8	4.0	2.9
Israel	9.2	9.5	9.7	8.6
Italy	5.3	9.0	6.5	4.5
Jamaica	9.0	9.1	8.9	8.2
Japan	1.9	1.3	1.0	0.9
Jordan	15.3	16.0	15.8	18.0
Kazakhstan	8.0	10.2	8.8	8.5
Kenya	10.5	9.5	9.5	9.0
Korea	12.0	11.1	27.6	70.4
Kuwait	12.8	12.0	11.8	11.6
Latvia	7.1	6.8	5.1	4.5
Lebanon	12.7	12.3	13.3	13.9
Libya	28.3	29.5	25.4	23.0
Lithuania	7.9	7.7	7.4	6.4
Luxembourg	3.2	3.1	2.8	2.0
Macao SAR	4.6	3.0	3.0	2.2
Macedonia, FYRO	12.5	15.1	16.5	14.3
Malaysia	10.2	9.0	9.3	8.7
Mali	—	—	14.1	—
Malta	5.6	4.5	4.1	3.6
Martinique	8.4	7.8	8.0	8.6
Mauritius	10.8	9.2	9.2	8.2

Country/Region	3Q11	4Q11	1Q12	2Q12
Mexico	9.7	8.8	11.2	10.0
Moldova	6.0	6.5	5.9	6.7
Mongolia	9.2	11.2	12.0	13.8
Morocco	12.0	12.3	15.6	20.1
Mozambique	12.6	12.0	11.9	11.3
Namibia	—	9.0	10.5	9.7
Nepal	24.0	22.4	20.0	19.3
Netherlands	6.6	13.1	6.3	4.9
New Zealand	4.8	3.8	3.5	3.1
Nicaragua	6.7	5.7	6.2	6.3
Nigeria	9.4	8.5	8.1	8.1
Norway	2.5	2.3	1.6	1.9
Oman	14.4	15.5	14.9	16.2
Pakistan	31.9	32.9	32.8	35.3
Palestinian Authority	27.1	29.9	29.1	29.8
Panama	10.8	9.6	9.9	7.6
Paraguay	6.7	6.3	6.1	4.9
Peru	10.3	10.0	10.7	10.3
Philippines	10.4	9.6	10.2	9.8
Poland	8.7	8.9	9.0	8.0
Portugal	8.9	8.9	6.5	5.1
Puerto Rico	8.0	6.9	6.7	5.9
Qatar	12.2	13.5	12.1	11.7
Réunion	7.9	7.4	7.1	7.3
Romania	14.0	13.8	14.9	15.0
Russia	6.1	7.2	6.2	6.7
Saudi Arabia	14.3	14.1	14.0	13.4
Senegal	10.1	10.4	11.5	9.7
Serbia	13.3	14.4	15.1	13.5
Singapore	6.9	5.7	5.6	4.4
Slovakia	4.2	3.6	3.4	3.0
Slovenia	5.0	4.6	4.2	4.0
South Africa	9.4	8.1	7.9	6.9

Country/Region	3Q11	4Q11	1Q12	2Q12
Spain	6.9	7.6	7.3	5.4
Sri Lanka	11.3	10.8	10.5	10.0
Sweden	2.7	2.5	1.8	2.1
Switzerland	2.8	2.3	1.8	1.7
Taiwan	10.4	8.2	6.9	5.3
Tanzania	11.6	10.2	10.1	9.8
Thailand	19.4	17.9	18.9	17.4
Trinidad and Tobago	10.1	8.4	8.5	7.2
Tunisia	11.2	13.2	15.3	14.3
Turkey	22.7	26.6	31.9	26.7
Uganda	12.0	11.6	11.4	11.1
Ukraine	6.3	7.1	6.6	7.0
United Arab Emirates	15.1	16.0	16.1	14.6
United Kingdom	5.5	5.1	3.9	3.2
United States	9.4	5.5	5.0	6.0
Uruguay	5.3	4.0	4.3	4.0
Uzbekistan	—	—	4.3	4.5
Venezuela	7.5	7.1	7.1	6.0
Vietnam	16.3	16.5	17.0	18.1
Yemen	—	20.5	21.8	21.9
Zambia	—	—	12.4	11.7
Zimbabwe	—	—	12.7	13.4
Worldwide	***7.7***	***7.1***	***6.6***	***7.0***

Glossary

For additional information about these and other terms, visit the MMPC glossary at www.microsoft.com/security/portal/Threat/Encyclopedia/Glossary.aspx.

419 scam
See *advance-fee fraud*.

ActiveX control
A software component of Microsoft Windows that can be used to create and distribute small applications through Internet Explorer. ActiveX controls can be developed and used by software to perform functions that would otherwise not be available using typical Internet Explorer capabilities. Because ActiveX controls can be used to perform a wide variety of functions, including downloading and running programs, vulnerabilities discovered in them may be exploited by malware. In addition, cybercriminals may also develop their own ActiveX controls, which can do damage to a computer if a user visits a webpage that contains the malicious ActiveX control.

Address Space Layout Randomization (ASLR)
A security feature in recent versions of Windows that randomizes the memory locations used by system files and other programs, which makes it harder for an attacker to exploit the system by targeting specific memory locations.

advance-fee fraud
A common confidence trick in which the sender of a message purports to have a claim on a large sum of money but is unable to access it directly for some reason, typically involving bureaucratic red tape or political corruption. The sender asks the prospective victim for a temporary loan to be used for bribing officials or for paying fees to get the full sum released. In exchange, the sender promises the target a share of the fortune amounting to a much larger sum than the original loan, but does not deliver. Advance-fee frauds are often called *419 scams*, in reference to the article of the Nigerian Criminal Code that addresses fraud.

adware
A program that displays advertisements. Although some adware can be beneficial by subsidizing a program or service, other adware programs may display advertisements without adequate consent.

ASLR
See *Address Space Layout Randomization (ASLR)*.

backdoor trojan
A type of trojan that provides attackers with remote unauthorized access to and control of infected computers. Bots are a subcategory of backdoor trojans. Also see *botnet*.

botnet
A set of computers controlled by a "command-and-control" (C&C) computer to execute commands as directed. The C&C computer can issue commands directly (often through Internet Relay Chat [IRC]) or by using a decentralized mechanism, such as peer-to-peer (P2P) networking. Computers in a botnet are often called *bots*, *nodes*, or *zombies*.

buffer overflow
An error in an application in which the data written into a buffer exceeds the current capacity of that buffer, thus overwriting adjacent memory. Because memory is overwritten, unreliable program behavior may result and, in certain cases, allow arbitrary code to run.

CCM
Short for *computers cleaned per mille* (thousand). The number of computers cleaned for every 1,000 executions of the MSRT. For example, if the MSRT has 50,000 executions in a particular location in the first quarter of the year and removes infections from 200 computers, the CCM for that location in the first quarter of the year is 4.0 (200 ÷ 50,000 × 1,000).

clean
To remove malware or potentially unwanted software from an infected computer. A single cleaning can involve multiple disinfections.

cross-site scripting
Abbreviated *XSS*. An attack technique in which an attacker inserts malicious HTML and JavaScript into a vulnerable Web page, often in an effort to distribute malware or to steal sensitive information from the Web site or its visitors. Despite

the name, cross-site scripting does not necessarily involve multiple websites. Persistent cross-site scripting involves inserting malicious code into a database used by a web application, potentially causing the code to be displayed for large numbers of visitors.

Data Execution Prevention (DEP)
A security technique designed to prevent buffer overflow attacks. DEP enables the system to mark areas of memory as non-executable, which prevents code in those memory locations from running.

definition
A set of signatures that antivirus, antispyware, or antimalware products can use to identify malware. Other vendors may refer to definitions as DAT files, pattern files, identity files, or antivirus databases.

DEP
See *Data Execution Prevention (DEP)*.

detection
The discovery of malware or potentially unwanted software on a computer by antimalware software. Disinfections and blocked infection attempts are both considered detections.

detection signature
A set of characteristics that can identify a malware family or variant. Signatures are used by antivirus and antispyware products to determine whether a file is malicious or not. Also see *definition*.

disclosure
Revelation of the existence of a vulnerability to a third party.

downloader
See *trojan downloader/dropper*.

exploit
Malicious code that takes advantage of software vulnerabilities to infect a computer or perform other harmful actions.

firewall
A program or device that monitors and regulates traffic between two points, such as a single computer and the network server, or one server to another.

generic
A type of signature that is capable of detecting a variety of malware samples from a specific family, or of a specific type.

IFrame
Short for *inline frame*. An IFrame is an HTML document that is embedded in another HTML document. Because the IFrame loads another webpage, it can be used by criminals to place malicious HTML content, such as a script that downloads and installs spyware, into non-malicious HTML pages that are hosted by trusted websites.

in the wild
Said of malware that is currently detected on active computers connected to the Internet, as compared to those confined to internal test networks, malware research laboratories, or malware sample lists.

keylogger
A program that sends keystrokes or screen shots to an attacker. Also see *password stealer (PWS)*.

malware
Any software that is designed specifically to cause damage to a user's computer, server, or network. Viruses, worms, and trojans are all types of malware.

malware impression
A single instance of a user attempting to visit a page known to host malware and being blocked by SmartScreen Filter in Internet Explorer 8 or 9. Also see *phishing impression*.

monitoring tool
Software that monitors activity, usually by capturing keystrokes or screen images. It may also include network sniffing software. Also see *password stealer (PWS)*.

P2P
See *peer-to-peer (P2P)*.

password stealer (PWS)
Malware that is specifically used to transmit personal information, such as user names and passwords. A PWS often works in conjunction with a keylogger. Also see *monitoring tool*.

payload
The actions conducted by a piece of malware for which it was created. Payloads can include, but are not limited to, downloading files, changing system settings, displaying messages, and logging keystrokes.

peer-to-peer (P2P)
A system of network communication in which individual nodes are able to communicate with each other without the use of a central server.

phishing
A method of credential theft that tricks Internet users into revealing personal or financial information online. Phishers use phony websites or deceptive email messages that mimic trusted businesses and brands to steal personally identifiable information (PII), such as user names, passwords, credit card numbers, and identification numbers.

phishing impression
A single instance of a user attempting to visit a known phishing page with Internet Explorer 7, 8, or 9, and being blocked by the Phishing Filter or SmartScreen Filter. Also see *malware impression*.

polymorphic
A characteristic of malware that can mutate its structure to avoid detection by antimalware programs, without changing its overall algorithm or function.

pop-under
A webpage that opens in a separate window that appears beneath the active browser window. Pop-under windows are commonly used to display advertisements.

potentially unwanted software
A program with potentially unwanted functionality that is brought to the user's attention for review. This functionality may affect the user's privacy, security, or computing experience.

rogue security software
Software that appears to be beneficial from a security perspective but that provides limited or no security capabilities, generates a significant number of erroneous or misleading alerts, or attempts to socially engineer the user into participating in a fraudulent transaction.

rootkit
A program whose main purpose is to perform certain functions that cannot be easily detected or undone by a system administrator, such as hiding itself or other malware.

sandbox
A specially constructed portion of a computing environment in which potentially dangerous programs or processes may run without causing harm to resources outside the sandbox.

signature
See *detection signature*.

spyware
A program that collects information, such as the websites a user visits, without adequate consent. Installation may be without prominent notice or without the user's knowledge.

SQL injection
A technique in which an attacker enters a specially crafted Structured Query Language (SQL) statement into an ordinary web form. If form input is not filtered and validated before being submitted to a database, the malicious SQL statement may be executed, which could cause significant damage or data loss.

tool
Software that may have legitimate purposes but may also be used by malware authors or attackers.

trojan
A generally self-contained program that does not self-replicate but takes malicious action on the computer.

trojan downloader/dropper
A form of trojan that installs other malicious files to a computer that it has infected, either by downloading them from a remote computer or by obtaining them directly from a copy contained in its own code.

virus
Malware that replicates, typically by infecting other files in the computer, to allow the execution of the malware code and its propagation when those files are activated.

vulnerability
A weakness, error, or poor coding technique in a program that may allow an attacker to exploit it for a malicious purpose.

wild
See *in the wild*.

worm
Malware that spreads by spontaneously sending copies of itself through email or by using other communication mechanisms, such as instant messaging (IM) or peer-to-peer (P2P) applications.

XSS
See *cross-site scripting*.

Threat families referenced in this report

The definitions for the threat families referenced in this report are adapted from the Microsoft Malware Protection Center encyclopedia (www.microsoft.com/security/portal), which contains detailed information about a large number of malware and potentially unwanted software families. See the encyclopedia for more in-depth information and guidance for the families listed here and throughout the report.

Win32/Autorun. A family of worms that spreads by copying itself to the mapped drives of an infected computer. The mapped drives may include network or removable drives.

Win32/Bancos. A data-stealing trojan that captures online banking credentials and relays them to the attacker. Most variants target customers of Brazilian banks.

Win32/Banker. A family of data-stealing Trojans that captures banking credentials such as account numbers and passwords from computer users and relays them to the attacker. Most variants target customers of Brazilian banks; some variants target customers of other banks.

Win32/Banload. A family of trojans that download other malware. Banload usually downloads Win32/Banker, which steals banking credentials and other sensitive data and sends it back to a remote attacker.

Blacole. An exploit pack, also known as Blackhole, that is installed on a compromised web server by an attacker and includes a number of exploits that target browser software. If a vulnerable computer browses a compromised website that contains the exploit pack, various malware may be downloaded and run.

JS/BlacoleRef. An obfuscated script, often found inserted into compromised websites, that uses a hidden inline frame to redirect the browser to a Blacole exploit server.

Win32/Bumat. A generic detection for a variety of threats.

Win32/Conficker. A worm that spreads by exploiting a vulnerability addressed by Security Bulletin MS08-067. Some variants also spread via removable drives and by exploiting weak passwords. It disables several important system services and security products, and downloads arbitrary files.

Java/CVE-2012-0507. A detection for a malicious Java applet that exploits the Java Runtime Environment (JRE) vulnerability described in CVE-2012-0507, addressed by an Oracle security update in February 2012.

Win32/Dorkbot. A worm that spreads via instant messaging and removable drives. It also contains backdoor functionality that allows unauthorized access and control of the affected computer. Win32/Dorkbot may be distributed from compromised or malicious websites using PDF or browser exploits.

AndroidOS/DroidDream. A malicious program that affects mobile devices running the Android operating system. It may be bundled with clean applications, and is capable of allowing a remote attacker to gain access to the mobile device.

Win32/Dynamer. A generic detection for a variety of threats.

Win32/EyeStye. A trojan that attempts to steal sensitive data using a method known as form grabbing, and sends it to a remote attacker. It may also download and execute arbitary files and use a rootkit component in order to hide its activities.

MacOS_X/FakeMacdef. A rogue security software family that affects Apple Mac OS X. It has been distributed under the names MacDefender, MacSecurity, MacProtector, and possibly others.

Win32/FakePAV. A rogue security software family that often masquerades as Microsoft Security Essentials or other legitimate antimalware products.

Win32/FakeRean. A rogue security software family distributed under a variety of randomly generated names, including Privacy Protection, Win 7 Internet Security 2010, Vista Antivirus Pro, XP Guardian, and many others.

Win32/FakeSysdef. A rogue security software family that claims to discover nonexistent hardware defects related to system memory, hard drives, and overall system performance, and charges a fee to fix the supposed problems.

Win32/FakeVimes. A rogue security software family distributed under the names Internet Security Guard, Extra Antivirus, Virus Melt, and many others.

MacOS_X/Flashback. A trojan that targets Java JRE vulnerability CVE-2012-0507 on Mac OS X to enroll the infected computer in a botnet.

Win32/Gendows. A tool that attempts to activate Windows 7 and Windows Vista operating system installations.

AndroidOS/GingerBreak. A program that affects mobile devices running the Android operating system. It drops and executes an exploit that, if run successfully, gains administrator privileges on the device.

AndroidOS/GingerMaster. A malicious program that affects mobile devices running the Android operating system. It may be bundled with clean applications, and is capable of allowing a remote attacker to gain access to the mobile device.

Win32/Helompy. A worm that spreads via removable drives and attempts to capture and steal authentication details for a number of different websites or online services.

Win32/Hotbar. Adware that displays a dynamic toolbar and targeted pop-up ads based on its monitoring of web-browsing activity.

JS/IframeRef. A generic detection for specially formed IFrame tags that point to remote websites that contain malicious content.

Win32/Keygen. A generic detection for tools that generate product keys for various software products.

Win32/Lethic. A trojan that connects to remote servers, which may lead to unauthorized access to an affected system.

Unix/Lotoor. A detection for specially crafted Android programs that attempt to exploit vulnerabilities in the Android operating system to gain root privilege.

Win32/Malagent. A generic detection for a variety of threats.

Win32/Meredrop. A generic detection for trojans that drop and execute multiple forms of malware on a local computer. These trojans are usually packed, and may contain multiple trojans, backdoors, or worms. Dropped malware may connect to remote websites and download additional malicious programs.

Win32/Microjoin. A generic detection for tools that bundle malware files with clean files in an effort to deploy malware without being detected by security software.

JS/Mult. A generic detection for various exploits written in the JavaScript language.

Win32/Nuqel. A worm that spreads via mapped drives and certain instant messaging applications. It may modify system settings, connect to certain websites, download arbitrary files, or take other malicious actions.

Win32/Obfuscator. A generic detection for programs that have had their purpose disguised to hinder analysis or detection by antivirus scanners. Such programs commonly employ a combination of methods, including encryption, compression, anti-debugging and anti-emulation techniques.

Win32/Onescan. A Korean-language rogue security software family distributed under the names One Scan, Siren114, EnPrivacy, PC Trouble, Smart Vaccine, and many others.

Win32/OpenCandy. An adware program that may be bundled with certain third-party software installation programs. Some versions may send user-specific information, including a unique machine code, operating system information, locale, and certain other information to a remote server without obtaining adequate user consent.

Win32/Orsam. A generic detection for a variety of threats.

Win32/Pameseg. A fake program installer that requires the user to send SMS messages to a premium number to successfully install certain programs.

Win32/Patch. A family of tools intended to modify, or “patch,” programs that may be evaluation copies, or unregistered versions with limited features for the purpose of removing the limitations.

Win32/Pdfjsc. A family of specially crafted PDF files that exploit Adobe Acrobat and Adobe Reader vulnerabilities. Such files contain malicious JavaScript that executes when the file is opened.

JS/Phoex. A malicious script that exploits the Java Runtime Environment (JRE) vulnerability discussed in CVE-2010-4452. If run in a computer running a vulnerable version of Java, it downloads and executes arbitrary files.

Win32/Pluzoks. A trojan that silently downloads and installs other programs without consent. This could include the installation of additional malware or malware components.

JS/Popupper. A detection for a particular JavaScript script that attempts to display pop-under advertisements.

JS/Pornpop. A generic detection for specially-crafted JavaScript-enabled objects that attempt to display pop-under advertisements, usually with adult content.

Win32/Ramnit. A family of multi-component malware that infects executable files, Microsoft Office files, and HTML files. Win32/Ramnit spreads to removable drives and steals sensitive information such as saved FTP credentials and browser cookies. It may also open a backdoor to await instructions from a remote attacker.

JS/Redirector. A detection for a class of JavaScript trojans that redirect users to unexpected websites, which may contain drive-by downloads.

Win32/Rimecud. A family of worms with multiple components that spread via fixed and removable drives and via instant messaging. It also contains backdoor functionality that allows unauthorized access to an affected system.

Win32/Sality. A family of polymorphic file infectors that target executable files with the extensions .scr or .exe. They may execute a damaging payload that deletes files with certain extensions and terminates security-related processes and services.

Win32/Sirefef. A rogue security software family distributed under the name Antivirus 2010 and others.

Win32/Sisproc. A generic detection for a group of trojans that have been observed to perform a number of various and common malware behaviors.

Win32/Startpage. A detection for various threats that change the configured start page of the affected user's web browser and may also perform other malicious actions.

Win32/Stuxnet. A multi-component family that spreads via removable volumes by exploiting the vulnerability addressed by Microsoft Security Bulletin MS10-046.

Win32/Swisyn. A trojan that drops and executes arbitrary files on an infected computer. The dropped files may be potentially unwanted or malicious programs.

Win32/Taterf. A family of worms that spread through mapped drives to steal login and account details for popular online games.

Win32/Tracur. A trojan that downloads and executes arbitrary files, redirects web search queries to a malicious URL, and may also install other malware.

Win32/VB. A detection for various threats written in the Visual Basic programming language.

Win32/Vobfus. A family of worms that spreads via network drives and removable drives and download/executes arbitrary files. Downloaded files may include additional malware.

ASX/Wimad. A detection for malicious Windows Media files that can be used to encourage users to download and execute arbitrary files on an affected machine.

Win32/Winwebsec. A rogue security software family distributed under the names Winweb Security, Win 8 Security System, System Security, and others.

Win32/Wizpop. Adware that may track user search habits and download executable programs without user consent.

Win32/Wpakill. A family of tools that attempt to disable or bypass WPA (Windows Product Activation), WGA (Windows Genuine Advantage) checks, or WAT (Windows Activation Technologies), by altering Windows operating system files, terminating processes, or stopping services.

Win32/Yeltminky. A family of worms that spreads by making copies of itself on all available drives and creating an autorun.inf file to execute that copy.

TwC Next

Microsoft

One Microsoft Way
Redmond, WA 98052-6399
microsoft.com/security