Microsoft Security Intelligence Report

Volume 12

July through December, 2011

HOW CONFICKER CONTINUES TO PROPAGATE

www.microsoft.com/sir

Microsoft Security Intelligence Report

Authors

Dennis Batchelder
Microsoft Protection Technologies

Shah Bawany
Microsoft Windows Safety Platform

Joe Blackbird
Microsoft Malware Protection Center

Eve Blakemore
Microsoft Trustworthy Computing

Joe Faulhaber
Microsoft Malware Protection Center

Sarmad Fayyaz
Bing

David Felstead
Bing

Paul Henry
Wadeware LLC

Nitin Kumar Goel
Microsoft Security Response Center

Jeff Jones
Microsoft Trustworthy Computing

Jimmy Kuo
Microsoft Malware Protection Center

Marc Lauricella
Microsoft Trustworthy Computing

Ken Malcolmson
Microsoft Trustworthy Computing

Nam Ng
Microsoft Trustworthy Computing

Mark Oram
Microsoft Trustworthy Computing

Daryl Pecelj
Microsoft IT Information Security and Risk Management

Dave Probert
Microsoft Security Engineering Center

Tim Rains
Microsoft Trustworthy Computing

Frank Simorjay
Microsoft Trustworthy Computing

Holly Stewart
Microsoft Malware Protection Center

Matt Thomlinson
Microsoft Trustworthy Computing

Scott Wu
Microsoft Malware Protection Center

Terry Zink
Microsoft Forefront Online Protection for Exchange

Contributors

Doug Cavit
Microsoft Trustworthy Computing

Chris Compton
Microsoft Trustworthy Computing

Mike Convertino
Microsoft Trustworthy Computing

Enrique Gonzalez
Microsoft Malware Protection Center

Heather Goudey
Microsoft Malware Protection Center

Roger Grimes
Microsoft IT Information Security and Risk Management

Satomi Hayakawa
CSS Japan Security Response Team

Jenn LeMond
Microsoft IT Information Security and Risk Management

Le Li
Microsoft Windows Safety Platform

Jenner Mandel
Microsoft Trustworthy Computing

Hideya Matsuda
CSS Japan Security Response Team

Patrick Nolan
Microsoft Malware Protection Center

Takumi Onodera
Microsoft Premier Field Engineering, Japan

Anthony Penta
Microsoft Windows Safety Platform

Kathy Phillips
Microsoft Legal and Corporate Affairs

Hilda Larina Ragragio
Microsoft Malware Protection Center

Laura A. Robinson
Microsoft IT Information Security and Risk Management

Richard Saunders
Microsoft Trustworthy Computing

Jasmine Sesso
Microsoft Malware Protection Center

Adam Shostack
Microsoft Trustworthy Computing

Maarten Van Horenbeeck
Microsoft Trustworthy Computing

Henk van Roest
CSS Security EMEA

Patrik Vicol
Microsoft Malware Protection Center

Steve Wacker
Wadeware LLC

Dan Wolff
Microsoft Malware Protection Center

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 third and fourth quarters of 2011, respectively, 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, as in previous volumes of the report.

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, 2H11 represents the second half of 2011 (July 1 through December 31), 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.

How Conficker continues to propagate

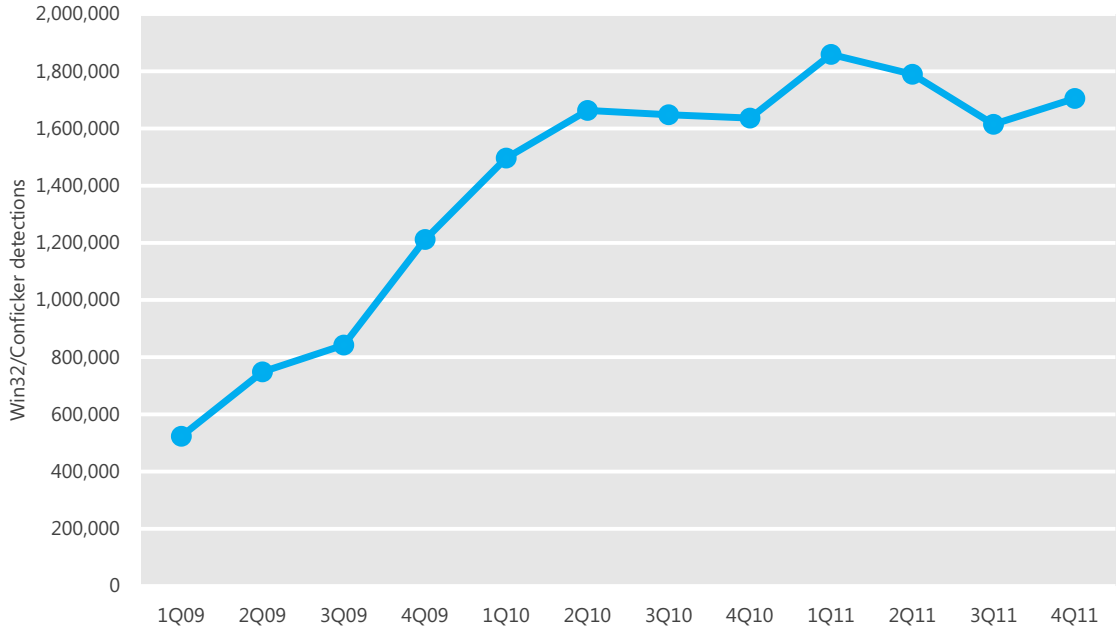
Background

In October 2008, Microsoft® released a security update (MS08-067) that addressed a software vulnerability in some versions of the Windows operating system. At that time, Microsoft recommended that customers install the update as soon as possible and warned that attackers could potentially create a worm that would affect vulnerable computers. Over the next few weeks, hundreds of millions of computers around the world received the MS08-067 update.

In November 2008, the Microsoft Malware Protection Center (MMPC) detected the emergence of the first version of Win32/Conficker, an aggressive and technically complex new family of worms. Win32/Conficker targeted the vulnerability addressed by MS08-067. Although the first version of this new threat did not spread widely, it seriously challenged security responders and others charged with ensuring the safety of the world's computer systems and data. In late December 2008—a full two months after Microsoft released the security update—a second version of Conficker was detected. This version includes additional attack vectors that help the worm to spread quickly.

Microsoft created and distributed antimalware signatures for the new threats. In addition, Microsoft worked with other members of the international security community to contain much of the damage that was caused by Conficker, and in the process established a potentially groundbreaking template for future cooperative response efforts.

Figure 1. Win32/Conficker detections by Microsoft antimalware products, 1Q09–4Q11



This section of the *Microsoft Security Intelligence Report, Volume 12* establishes that Conficker remains a threat, provides background information on why it is a serious threat, and what organizations can do to protect themselves. (For more information and deep technical details on Conficker, see the "Win32/Conficker Update" section in *Microsoft Security Intelligence Report, Volume 7 (January through June 2009)*, available at www.microsoft.com/sir.)

At its peak, Conficker infected an estimated seven million computers worldwide, according to the Conficker Working Group. Conficker was immediately recognized as dangerous because it attempts to exploit a vulnerability on Windows XP®-based systems that allows remote code execution when file sharing is enabled (CVE-2008-4250, which Microsoft had addressed in October 2008 with critical update MS08-067). In addition, Conficker disables several important system services and security products, and also downloads arbitrary files. The initial version (labeled Worm:Win32/Conficker.A by the MMPC) was not very successful at propagating, mostly because the MS08-067 security update had already been distributed and widely installed. However, the next variant, Worm:Win32/Conficker.B, uses two new propagation methods—abusing the Autorun feature on Windows XP and Windows Vista®-based computers, and

guessing administrator passwords on network shares with weak or shared passwords—to quickly propagate through the Internet.

In addition to quick propagation, the newer variants of Conficker use a larger array of attack techniques than most malware families. In addition to a suite of self-defense mechanisms such as blocking access to security-related websites and disabling security software on infected computers, Conficker uses encryption and a method called *HTTP rendezvous* to protect its payload channel.[1]

Because of the way Conficker uses multiple attack vectors to maximize its reach, there was a global effort to thwart its use and to determine who would try to make use of it. Worm:Win32/Conficker.E was reported to perform some downloads of the Win32/Waledac spambot and the rogue security software family Win32/FakeSpypro (which identified itself as "SpyProtect 2009"). This variant was programmed to delete itself in May 2009.

Propagation mechanisms

Although the efforts of the Conficker Working Group and associated organizations restricted Conficker's potential for damage, the MMPC received telemetry reports of the worm infecting or attacking 1.7 million computers in 4Q11, about 100,000 computers more than in 3Q11. A detailed analysis of the MMPC telemetry can help organizations defend against Conficker variants by understanding the relative success rates of the different propagation methods that the worm uses.

Information about the propagation vectors is directly observable through data reported by Microsoft security products running on computers whose administrators or users choose to opt in to data collection. The MMPC used this data to deduce the following information about Conficker's propagation mechanisms:

- **Credential-based attacks**. This type of attack uses the credentials of the logged-in user to access local or network resources, or else attacks password-protected resources using a built-in list of common or weak passwords.[2] When the worm successfully infects a computer using this type of attack, it

[1] See page 96 of *Microsoft Security Intelligence Report, Volume 7 (January through June 2009)* for more information about this technique.
[2] See the entry for Worm:Win32/Conficker.C in the MMPC encyclopedia (www.microsoft.com/security/portal) for the list of weak passwords used by Conficker.

creates a scheduled task on the infected computer that attempts to re-infect the computer at regular intervals. Credential-based attacks can therefore be identified through the presence of such a scheduled task.

- **Autorun feature abuse attempt**. Conficker can attempt to spread to a computer by abusing the Autorun feature in Windows, through the use of a malicious autorun.ini file that links to a Conficker executable. Microsoft security software detects and blocks this file, even on computers running versions of Windows that are not at risk from this form of attack. Detection of the malicious autorun.ini file is therefore not an indication of an infected computer, but indicates that an attack has been attempted.
- **MS08-067 exploitation**. It is possible to determine this type of attack because of a detail of the worm's implementation. After successful exploitation, Conficker calls a Windows API that in turn calls the Microsoft **IOfficeAntivirus** provider, which detects and blocks the transfer of the worm's code. The telemetry includes an indicator of whether the worm was active or not, which allows excluding partially removed or broken infection attempts.
- **Preexisting infection**. Microsoft antimalware software also reports details about Conficker infections that were present on the computer before the antimalware software was installed. These pre-existing infections are indicated by the presence of a Windows service created by Conficker.

Results

Figure 2 shows an analysis of three weeks of telemetry data of active Conficker installations or installation attempts.[3]

[3] This data was collected after the February 2011 release (through Windows Update and Microsoft Update) of a security update that addressed the Autorun feature abuse technique used by Conficker, as mentioned earlier. See blogs.technet.com/b/security/archive/2011/06/27/defending-against-autorun-attacks.aspx for more information.

Figure 2. Propagation methods used by Win32/Conficker variants, by percent of all attempted attacks detected

Worm Variant	Credential-based attack	Preexisting infection	Exploit	Autorun abuse attempt
Worm:Win32/Conficker.A	—	58%	42%	—
Worm:Win32/Conficker.B	61%	14%	17%	8%
Worm:Win32/Conficker.C	61%	15%	24%	*
Worm:Win32/Conficker.D	—	100%	—	—
Overall	60%	15%	20%	6%

* Autorun files for variants B and C are identical, and accordingly are all grouped with Conficker.B in this chart.

Most of the analyzed incidents (60 percent) involved credential-based attacks, with the remaining 40 percent including all other known propagation methods. The second-greatest number of incidents in the specified timeframe (20 percent) exploited the CVE-2008-4250 vulnerability on computers that had not yet been updated with Security Bulletin MS08-067, despite the fact that the update had been released more than two years before. The third-greatest number of analyzed incidents (15 percent) involved infections that were present on the computer before the installation of the antimalware product that detected and removed the infection. Finally, only 6 percent of incidents that were observed in the specified timeframe involved abuse of the Autorun feature in Windows. The release of an update that hardened the Autorun feature in Windows XP and Windows Vista may have helped achieve this relatively low percentage.

This attack pattern suggests that improving credential policies and practices is one of the most important steps computer administrators can take to effectively combat the spread of Conficker. Domain administrators can use Active Directory® Domain Services (AD DS) to define and enforce Group Policy Objects (GPOs) that require users to create complex passwords.[4] If local passwords are used for some resources in an organization, resource owners should be required or encouraged to use strong passwords for them as well.

When considered from the perspective of the affected operating system, it becomes clearer that credential-based attacks on file shares are the primary mechanism Conficker uses to compromise computers running recent versions of the Windows operating system, as shown in Figure 3.

[4] See "Enforcing Strong Password Usage Throughout Your Organization" on Microsoft TechNet for more information and instructions.

Figure 3. Blocked Conficker infection attempts by operating system

Operating System	Credential-based attack	Exploit	Autorun abuse attempt
Windows 2003	81%	19%	1%
Windows XP	54%	43%	2%
Windows Vista	84%	—	16%
Windows 7	89%	—	11%

Windows 7 was never vulnerable to CVE-2008-4250 exploits, and although Windows Vista was vulnerable, no exploit attempts were observed in the measurement period. Network Inspection System (NIS), a feature of Microsoft Security Essentials and Microsoft Forefront® Threat Management Gateway, blocks exploit attempts on vulnerable computers running Windows Vista and other recent versions of Windows, which prevents the Conficker worm from exploiting the CVE-2008-4250 vulnerability.[5] Windows 7 was also far more difficult to attack through Autorun feature abuse, and although autorun abuse attempts were observed and blocked on 11 percent of Windows 7 systems, they would not have been successful because of the restricted Autorun policy on that platform.

The Conficker worm may or may not have had as great an effect as its creators expected, but it continues to search for new victims. Although installing all relevant security updates and hardening the Autorun feature in Windows can close off several Conficker attack vectors, this analysis of the worm's attacks shows that using weak passwords for network and local resources can still leave computers at significant risk of infection. To effectively defend against Conficker and similar malware families, responsible computer administrators should develop a multifaceted strategy that includes strong passwords, quick deployment of security updates, and the use of regularly updated, real-time antimalware software.

[5] See go.microsoft.com/fwlink/?LinkId=248183 for more information about the Network Inspection System.

Figure 4. Blocked Conficker infection attempts on enterprise computers, as detected by Microsoft Forefront Endpoint Protection

Operating System	Credential-based attack	Exploit	Autorun abuse attempt
Windows 2003	91%	9%	—
Windows XP	88%	12%	—
Windows Vista	100%	—	—
Windows 7	100%	—	—

Figure 5. Blocked Conficker infection attempts on consumer computers, as detected by Microsoft Security Essentials

Operating system	Credential-based attack	Exploit	Autorun abuse attempt
Windows 2003	77%	22%	1%
Windows XP	46%	51%	3%
Windows Vista	77%	—	23%
Windows 7	85%	—	15%

Tips to help clean up an environment in which Conficker is present

Malware such as Conficker can still pose a challenge for IT administrators, despite the fact that it is a well-known threat. Even a conscientious IT department that follows responsible practices for quickly installing security updates, installing and monitoring antimalware and intrusion detection systems, and controlling access to file shares can still encounter outbreaks of a threat such as Conficker.

Malware that uses common network protocols such as Server Message Block (SMB) to replicate can pose a threat to locked-down file shares, because an infected computer that has write privileges to the file share can pass the infection on to it. A common scenario is one in which a file share is disinfected by server-side antimalware software, but is quickly reinfected when an infected client computer connects to it. This potential for repeated reinfection gives malware that leverages open file shares, such as Conficker, staying power in data centers. Identifying the original source of the infection within the organization is therefore essential for eradicating such malware. Finding it can require a bit of agility and creativity on the part of server administrators.

Microsoft provides information to help IT administrators deal with Conficker infections at www.microsoft.com/conficker. The following list provides some additional tips that may help advanced users who possess a good understanding of computer security and Windows administration find computers that are infected with Conficker in order to minimize their attack surface.

- Create a "rogue" file share, populate it with various executable files and share the directory for full control to all. However, before sharing the folder, turn on Windows monitoring to identify computers that successfully write to the share.[6] The events captured in Windows Event Viewer with share monitoring enabled will capture enough information to identify the original source of the infection. Use this practice on several shares and systems in the environment and monitor as needed.
- On infected computers, check the device log; by default, the Windows installation places this log in *C:\Windows\inf\setupapi.dev*. The log will contain information about devices such as memory sticks or other USB hardware that has been installed on the system and will help find the original source of the infection if this method was used to install Conficker or other malware that propagates through Autorun.[7]
- The original source of the infection is often determined to be a computer inside the organization's backup infrastructure. Because of performance and other related factors, many organizations relax security controls for backup systems, which is a big mistake. It is important for the organization's IT staff to ensure that basic security practices are in place, especially for an environment in which Conficker is problematic. It isn't uncommon for malware to be stored on backup servers, because the files are usually encrypted and continuously copied back down to clean servers.
- Inside the data center, implement a server administrator file share change control process that reviews and approves file share configurations; such an approach will help minimize the attack surface for malware that uses network shares to replicate. Depending on the size of the organization, it could be a daunting task to implement such a process throughout an entire data center, but at a minimum it should be required for servers that have been identified as repeat offenders or other systems that have been deemed critical to the organization's service.

---

[6] For details on auditing user access, see Microsoft Knowledge Base article 310399 at support.microsoft.com.
[7] For more information about the device log, see "Troubleshooting Device Installation with the SetupAPI Log File" at the Microsoft Developer Network website (msdn.microsoft.com).

Microsoft

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