



Security Implications of Microsoft® Windows Vista™

Security Implications of Microsoft Windows Vista

Contents

Introduction	4
Security technologies in Windows Vista	4
Generic exploit mitigation	5
Kernel integrity	8
System integrity and user-mode defenses	9
Security evolution not revolution	11
Malicious code and what it means to Windows Vista	11
The network threat	12
Summary and conclusions	12
Attackers go where the vulnerabilities are	13
New technology brings opportunities and risks	14
Risks posed by third-party applications	14

Introduction

Windows Vista is the result of over four years of work and the investment of many billions of dollars. It is billed as the most secure version yet of the Microsoft Windows® operating system. This paper discusses not only the security technologies employed by Microsoft that justify this accolade but also how, in combination, these technologies mitigate specific classes of threats. This paper presents a high-level summary of Symantec's research findings into the security of Windows Vista, and a set of conclusions that discuss the exposure that remains even in the face of its new security technologies. The intent of this paper is not to detract from the improvements that Microsoft has made, but rather to provide an objective and balanced view of how Windows Vista will affect the overall threat landscape.

Symantec started researching Windows Vista in 2005 and has monitored its development carefully. The goal of this research has been to understand the technology improvements being made by Microsoft and also to understand the threats facing the new operating system and, in turn, Symantec's customers.

Security technologies in Windows Vista

With the introduction of Windows Vista, Microsoft has leveraged a number of security technologies in order to mitigate several classes of attack that have historically plagued the Windows operating system. These technologies are numerous, and are best depicted visually (see Figure 1).

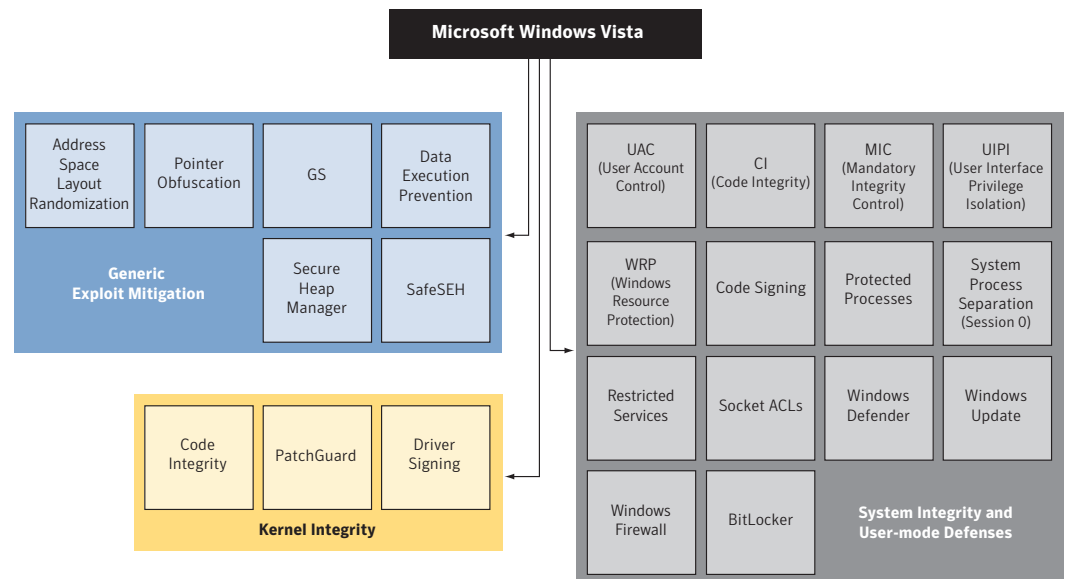


Figure 1. A depiction of Windows Vista security technologies

These technologies can be broken down into three core categories:

- Generic exploit mitigation
- Kernel integrity
- System integrity and user-mode defenses

Generic exploit mitigation

This category of mitigation is designed to prevent attackers from successfully exploiting applications that contain specific classes of code-level vulnerabilities. The technologies employed here fall into two key categories: developer-controlled and operating system improvements. When combined, these techniques successfully inhibit the exploitation of memory corruption and memory manipulation vulnerabilities. This includes the following common classes of software flaws:

- Stack buffer overflow vulnerabilities
- Stack function pointer overwrites
- Structured exception handler overwrites
- Heap overflow and structure manipulation

The technologies introduced in Windows Vista are very effective at protecting the core Windows operating system as well as Microsoft compiled applications. They serve to make the exploitation of traditional vulnerabilities infeasible, including those leveraged by well-known widespread worms observed earlier this decade. As a result, the overall impact of some code-level flaws, even when introduced by a Microsoft software engineer, is greatly diminished.

Developer-controlled technologies

Developer-controlled technologies can be leveraged by software engineers in order to make their applications more robust. These technologies can be incorporated either through the enabling of compiler options or through the introduction of explicit code changes.

The technologies that fall into this category are:

- Pointer obfuscation
- GS
- Safe Structured Exception Handlers (SafeSEH)
- Address Space Layout Randomization (ASLR)
- Terminate on Heap Corruption

Analysis of developer-controlled technologies

One barrier to the success of these technologies is the requirement for third-party software vendors to explicitly leverage them. Software engineers must utilize the latest version of Microsoft's development tools in a specific manner. Only by doing so can they enable the functionality that is designed to inhibit or minimize the impact of the different exploitation techniques.

Only when developers recompile their application or, in certain instances such as pointer obfuscation, make modifications to their application's source code will they benefit from these improvements.

While the majority of newer Microsoft applications are expected to use these technologies, older software and software written by third parties may not. As a result, older Microsoft or third-party applications and drivers will continue to pose a risk, as they will remain largely unprotected. This fact has already been borne out with the recent announcement of vulnerabilities present in the Windows Vista version of a common server application.¹

¹ www.coresecurity.com/index.php5?module=ContentMod&action=item&id=1660

Security Implications of Microsoft Windows Vista

Symantec researchers noted that in some cases even core Windows Vista components failed to adequately leverage these technologies. Specifically, a small percentage of Windows Vista 32-bit has not been compiled with GS technology from Microsoft Visual Studio® 2005.

The reason for the exclusion of these applications from the protection afforded by this technology is unclear. It is acknowledged that these components pose a greater risk than those that are protected.

Consequently, these components of Windows Vista are not protected against the aforementioned class of memory corruption and memory manipulation vulnerabilities. While the exposure to risk resulting from this circumstance is low, it does serve to increase the potential attack surface for Windows Vista. Symantec expects attackers to identify these vulnerable points and investigate their potential.

Operating system improvements

Operating system improvements are technologies that are native to the core operating system. While similar in overall effect to developer-controlled technologies, their function is ultimately implemented by components within the core operating system. The technologies that fall under operating system improvements are:

- Heap manager improvements
- Data Execution Prevention (DEP)
- Safe Structured Exception Handlers (SafeSEH)
- Address Space Layout Randomization (ASLR)
- Terminate on Heap Corruption

Analysis of operating system improvements

Like those discussed in the previous section, the majority of technologies falling into this category also require that software engineers first enable them in their application. Of these four different technologies, only the first (heap manager) applies by default to the operating system as a whole. The second (DEP) is enabled only for Windows Vista core operating system components and not for some common applications such as Internet Explorer. The final three require developers to specifically enable support in their application during development.

As a result, third-party applications, as well as those developed by Microsoft that are not considered part of the core operating system, are not afforded equal protection even with the introduction of these technologies.

Limited scope of Data Execution Prevention

In default installations of Windows Vista, Symantec observed that one technology (DEP) is applied by default only to the core operating system, as shown in Figure 2.

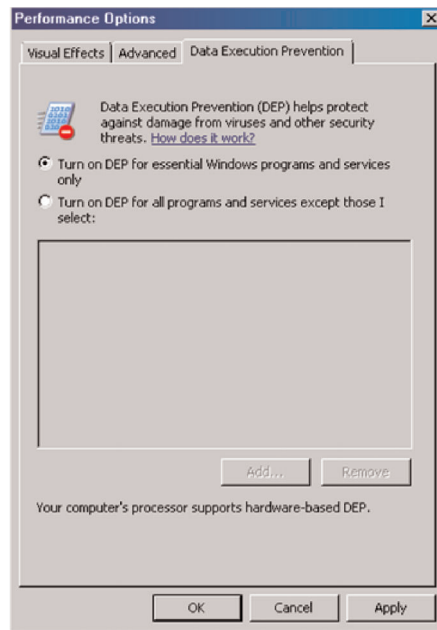


Figure 2. Default DEP configuration

This limitation leaves third-party applications on Windows Vista with less protection than the core Windows Vista operating system and service. This fact increases the likelihood of successful exploitation of vulnerabilities present in these applications. As mentioned previously, even common applications such as Internet Explorer do not leverage the benefits of DEP.

ASLR: Not as random as expected

Symantec performed an in-depth analysis on the effectiveness of Address Space Layout Randomization (ASLR). The purpose of this technology is to randomly locate programs in memory and, by doing so, enhance security. This enhancement comes from the attacker's inability to know exactly what to target during the exploitation of a vulnerable program. When implemented correctly, this technology is extremely effective in mitigating the exploitation of memory corruption and memory manipulation vulnerabilities.

Security Implications of Microsoft Windows Vista

Figure 3 shows the plotting of 11,500 executions of a test harness, the purpose of which was to understand the distribution of memory usage.

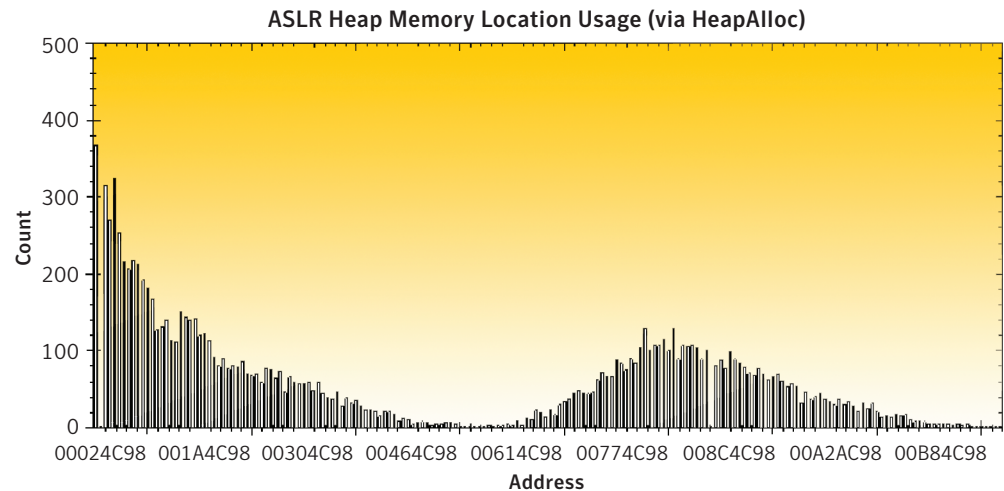


Figure 3. Address selection of HeapAlloc

The results of this analysis show that at least one aspect of ASLR's implementation did not perform as expected. Symantec found that one of the randomized components was not randomized consistently, resulting in a reduced degree of randomness in the layout of an application's memory. While ASLR continues to be effective, this reduction does increase the likelihood that an attacker can guess the correct address to target.

Microsoft has confirmed Symantec's research findings and resolved the issue highlighted. These shortcomings are due to be addressed in Windows Vista SP1.

Kernel integrity

The kernel is the core component of any modern operating system. It is the central building block upon which the security of the system is built. Should the kernel be compromised or subverted in any way, then the underlying foundation can no longer be trusted. Kernel integrity and security have become a hot topic in recent years due to the aggressive evolution of rootkit technologies. These technologies are used by attackers and threats to hide their presence while also providing potential backdoors into the system. In addition the evolution of Digital Rights Management (DRM) provides another, arguably even stronger incentive for securing the kernel to avoid the unauthorized interception of audio and video content.

For this reason Microsoft has invested heavily in technologies that can help improve the reliability and security of the Windows Vista kernel. The three technologies employed by Microsoft to improve kernel security are:

- Driver signing
- Code Integrity
- PatchGuard

Security Implications of Microsoft Windows Vista

Driver signing is designed to ensure that all kernel drivers loaded by the system are signed by a trusted authority. The goal of this technology is to ensure that only code that has been tested by Microsoft or signed by a trusted developer is loaded into the kernel—with the side effect of stopping malicious code from loading into the heart of the operating system.

Code Integrity is designed to ensure that the core operating system has not been tampered with either accidentally or maliciously. Code integrity verifies the digital signature and associated hash on core operating system binaries (in particular kernel components) in order to detect this tampering.

PatchGuard is the most controversial of these technologies. Whereas Code Integrity protects core operating system files on disk and in memory, PatchGuard protects key operating system structures from being patched or extended in kernel memory. Vendors such as Symantec have historically used this patching technique to provide protection at the lowest level possible to ensure the maximum protection against malicious code such as rootkits. However, these same techniques are utilized by rootkit writers to ensure the stealthiest operation possible.

Analysis of kernel integrity technologies

It is important to note that only the 64-bit version of Windows Vista benefits from this category of technology, while 32-bit Windows Vista, expected to be the standard deployment for years to come, does not.

As demonstrated during the development process of Windows Vista and during its release, hackers can and will subvert PatchGuard.² The kernel integrity protection mechanisms that are present on 64-bit Windows Vista can only be described as a bump in the road. That is, while these technologies may slow down an attacker, they may not provide a meaningful defense against a determined one.

Symantec researchers investigated the feasibility of disabling all three key kernel integrity technologies: driver signing, Code Integrity, and PatchGuard. Results have shown that all three technologies can be permanently disabled and removed from Windows Vista after approximately one man-week of effort. A potential victim need make only one mistake to become infected by a threat that does the same. The result: All new security technologies are stripped from Windows Vista in their entirety.

System integrity and user-mode defenses

Microsoft's system integrity and user-mode defenses are numerous, and their purpose is clear. Microsoft's strategy is to run software with the minimum set of privileges required and, where possible, to run applications in a compartmentalized environment. This approach is further strengthened by reliance on signing to provide assurances about the identity of the publisher of software. Such assurances allow the user to make informed decisions about running an application and allowing it to perform actions on the host when prompted.

The goal of these technologies is to encourage users to run programs at a reduced privilege rather than running everything as Administrator, forcing them to consider the consequences of their actions. In addition, these technologies seek to reduce the ability of malicious code to automatically compromise the entire system.

² www.uninformed.org/?v=3&a=3&t=sumry

Security Implications of Microsoft Windows Vista

Analysis of user-mode defenses

The implementation of these protections achieves many of the security goals that Microsoft had envisioned. Despite this increased protection, however, several risks continue to exist. The first risk is the lack of information provided by the many dialog boxes and prompts that appear during normal operating system use. This lack of information will lead to indifference on the part of the user when presented with these prompts.

Although some of these improvements are intended to improve security, some, such as User Account Control (UAC) are not actually considered by Microsoft to be a security boundary. They acknowledge that there are methods to bypass these protections and do not consider these to be security issues. Microsoft actually states that this is the case in their consumer best practice guidance:³

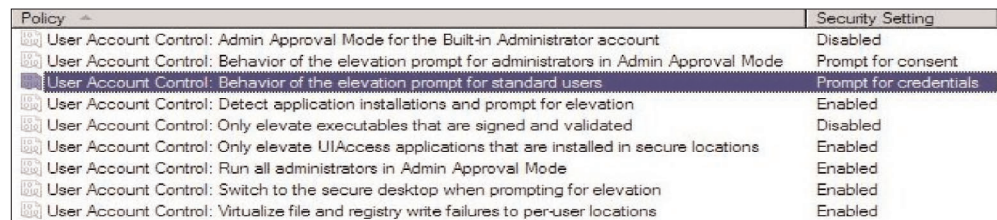
“It’s very important to remember that UAC prompts are not a security boundary—they don’t offer direct protection. They do offer you a chance to verify an action before it happens. Once you allow an action to proceed, there may be no easy way back.”

This message has been echoed by others at Microsoft in response to vulnerabilities being discovered in UAC.^{4,5,6} Microsoft’s message is that UAC vulnerabilities are not considered security issues, as UAC does not provide a security boundary.

Symantec has also discovered issues with certain executables that ship with Windows Vista that can undermine UAC’s reliance on digital signatures. Symantec has demonstrated that it is possible to execute an unsigned arbitrary library even though the user is presented with a dialog box implying that it is Microsoft authored code. This undermines user notifications since users can no longer trust the information they are presented with.⁷

A final and more worrisome issue is that users may ultimately disable these security functions. While these types of risks may be easy to manage in the enterprise environment, managing them in a home environment may be nearly impossible.

Symantec researchers observed that the User Account Control can be easily disabled manually. This action can be performed via the Local Security Policy tool included in Windows Vista.



Policy	Security Setting
User Account Control: Admin Approval Mode for the Built-in Administrator account	Disabled
User Account Control: Behavior of the elevation prompt for administrators in Admin Approval Mode	Prompt for consent
User Account Control: Behavior of the elevation prompt for standard users	Prompt for credentials
User Account Control: Detect application installations and prompt for elevation	Enabled
User Account Control: Only elevate executables that are signed and validated	Disabled
User Account Control: Only elevate UIAccess applications that are installed in secure locations	Enabled
User Account Control: Run all administrators in Admin Approval Mode	Enabled
User Account Control: Switch to the secure desktop when prompting for elevation	Enabled
User Account Control: Virtualize file and registry write failures to per-user locations	Enabled

Figure 4. Configuration of UAC via the Local Security Policy

³ http://download.microsoft.com/download/0/e/9/0e922c03-8537-482f-b57c-aa385b3dee20/Security_Best_Practice_Guidance_for_Consumers.doc

⁴ <http://blogs.technet.com/markrussinovich/archive/2007/02/12/638372.aspx>

⁵ <http://theinvisiblethings.blogspot.com/2007/02/vista-security-model-big-joke.html>

⁶ <http://theinvisiblethings.blogspot.com/2007/02/confusion-about-joke.html>

⁷ http://www.symantec.com/enterprise/security_response/weblog/2007/02/an_example_of_why_uac_prompts.html

Security evolution not revolution

Many of the technologies that Microsoft has employed to bolster the security of Windows Vista are not new. In fact, most are derived from the groundwork originally laid by open source operating systems such as Linux and OpenBSD, the PaX and Stackguard projects, as well as numerous academic publications.^{8,9}

The majority of these technologies first appeared in Windows XP SP2. Windows XP SP2, at the time of its release, was also billed as the most secure version of Windows, with Microsoft describing the benefits of installing Windows XP SP2 in the following way:

“When you install SP2 on your Windows XP-based PC, you can feel confident that you're running the most secure Windows operating system available.”¹⁰

These technologies, which are now integrated in Windows Vista, include driver signing, SafeSEH, DEP, pointer obfuscation, PatchGuard (Windows XP x64), UAC, code signing, Windows Defender, and Windows Update. In addition to these, Socket ACLs first debuted on a version of Windows prior to Windows Vista, in Windows Server® 2003 SP1.

Technologies that in Windows XP were disabled by default, are now instead enabled by default in Windows Vista.

The inclusion of these technologies in earlier versions of Windows, such as Windows XP and Windows Server 2003, has already resulted in a decline in the number of attacks that focused on core operating system components. As a result, Symantec has seen an increase in the number of attacks that focus on the applications that run on top of the operating system, such as office productivity suites and Web browsers. While Microsoft has invested heavily in protecting the core operating system, attackers have already moved on.

Malicious code and what it means to Windows Vista

Symantec researched the exposure of Windows Vista to threats propagating on prior versions of Microsoft Windows. The goal of this research was to determine whether the new security technologies in Windows Vista could mitigate the risks posed by legacy malicious code, even where the code was not written to run on Windows Vista or adapted to its new security model.

The results showed that 3 percent of backdoors can successfully execute and survive a system restart on Windows Vista without modification. Other categories include keyloggers, of which 4 percent can successfully execute and survive a system restart, mass mailers (4 percent), Trojans (2 percent), spyware (2 percent), and adware (2 percent). Symantec believes that these percentages would increase dramatically with only minor code changes to make these threats Windows Vista-aware, in turn allowing them to run successfully within the new Windows Vista security model.

As expected, no kernel-based rootkits were able to successfully install themselves. This can be attributed to the fact that a reduced set of privileges are used to run user applications by default. On 32-bit Windows Vista, a threat can penetrate the Windows Vista kernel unimpeded, if it is able to elevate its privilege level to that of full administrator. In order to do so, a threat must circumvent the new User Account Control technology in Windows Vista. As discussed earlier, several techniques exist that make this scenario much more accessible to threat authors than Microsoft had originally envisioned.

⁸ http://en.wikipedia.org/wiki/Address_space_layout_randomization

⁹ <http://pax.grsecurity.net/docs/aslr.txt>

¹⁰ <http://www.microsoft.com/windowsxp/sp2/overview.msp>

Security Implications of Microsoft Windows Vista

This research demonstrates that, while Windows Vista has made improvements that restrict the exploitation of vulnerabilities and reduce the likelihood of complete system compromise, some legacy threats survive and go unhindered by the improvements. This suggests that authors of existing threats need only update their code with minor changes in order to adapt to Windows Vista and continue to run within its confines. If the new security technologies introduced in Windows Vista were a silver bullet, we would expect that no legacy threats would survive.

The network threat

Microsoft has completely rewritten the network protocol stack for Windows Vista. This was an extremely ambitious project, because network protocol stacks typically require many years in production environments to mature. Full maturation is achieved by giving bugs ample time and opportunity to manifest themselves, even given the extensive testing and security design process implemented by Microsoft. During the development cycle, Symantec researchers discovered three remote denial-of-service conditions and three historic network attacks that worked successfully on public beta versions of the operating system. These issues have since been resolved, proving that Microsoft was making ongoing improvements to the Windows Vista network stack up until its final release. It's highly likely that more will be discovered given the significant volume of new code.

The same security risks that impact the new network protocol stack also exist for the built-in firewall, which has also been newly developed. Symantec researchers have already identified the existence of one unexpected firewall exception caused by an oversight in the core Windows Vista firewall implementation.

Windows Vista introduces a relatively large number of new network protocols. Two of these protocols stand out from the rest due to their impact on the enterprise. The first is IPv6, which is enabled and preferred by default. The second is a protocol called Teredo, which is a transitional technology that allows the tunneling of IPv6 over IPv4. The implication is that the vast majority of Windows Vista hosts are, by default, remotely accessible directly via IPv6 and Teredo. The usage of Teredo has the side effect of bypassing many firewall and NAT configurations. This has significant consequences for enterprises that rely on network-based protection, since perimeter security devices and other network defenses such as IPS and IDS will need to be upgraded in order to understand and decapsulate this new protocol.

Summary and conclusions

Symantec predicts that the new security features in Windows Vista will result in fewer instances of widespread worms that target core Windows operating system vulnerabilities. This class of worm was largely responsible for the majority of high-profile outbreaks in the early part of this century. We expect that worms will continue to thrive; however, their method of propagation will change. This trend has already been observed since the release of Windows XP SP2 and is expected to continue.

Symantec does not believe that Windows Vista security improvements will stifle other classes of malicious code that have historically targeted the Windows operating system.

Attackers go where the vulnerabilities are

Attackers follow security vulnerabilities, as these are a requirement for their success. Over the past several years, these vulnerabilities have increasingly moved up the application stack and away from the core operating system. Threats have and will continue to move into other areas, such as the Web application layer, where over 78 percent of all new security vulnerabilities reside today.¹¹ Windows Vista provides no enhanced security in this space, as the majority of vulnerabilities today are seen within PHP, Python, Perl, ASP, and other languages. In addition, new Web 2.0 technologies such as AJAX provide an entirely new layer on which tomorrow's threats will propagate.

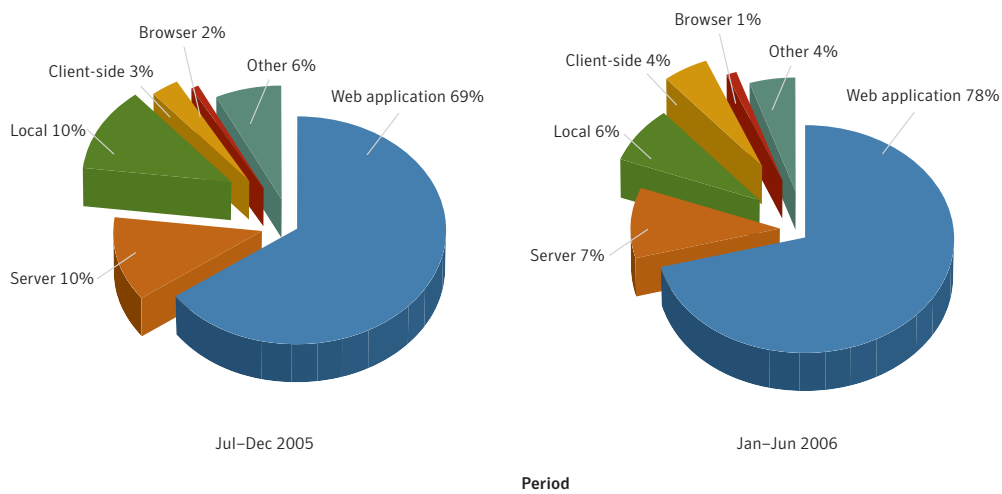


Figure 5. Distribution of security vulnerabilities by class

As mentioned previously, worms have already migrated away from their traditional means of propagation. This movement has resulted in their shift to more available technologies, including email, IM, and the Web; leveraging social engineering and other convincing trickery in order to infect their victims. Consequently, while Microsoft has made a significant number of improvements to quell the traditional network worm, threats have already moved on as seen in Figure 6.¹²

¹¹ Symantec *Internet Security Threat Report*, Volume X
¹² *ibid.*

Security Implications of Microsoft Windows Vista

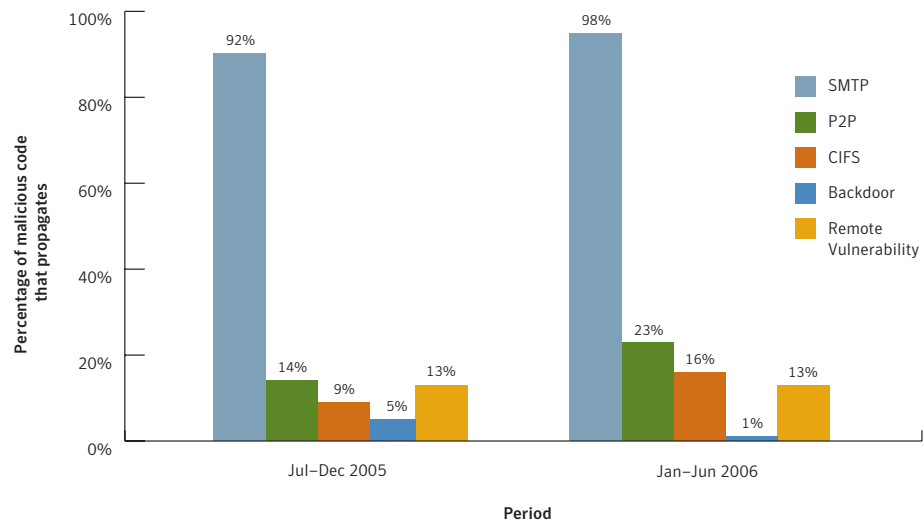


Figure 6. Propagation techniques used by the top 50 malicious code threats

New technology brings opportunities and risks

Windows Vista introduces a variety of new technologies, along with all of the potential risks that typically accompany them. Windows SideBar and gadgets are among the new technologies that may pose an increased risk to users in the future. Gadgets are a mixture of static HTML and scripting, the purpose of which is to allow the quick and easy development of new plug-ins for the Windows desktop. Examples of such applications are a clock, calculator, and RSS feed reader, but the possibilities extend far beyond these.

While gadgets do not automatically execute, Symantec researchers anticipate that they will be quickly adopted by malicious code writers as a novel way to convince users to download and execute arbitrary code. Although these gadgets are bound by the same restrictions as other applications, the fact that they are automatically authorized to communicate via the Web makes them an effective means to introduce arbitrary content, and also to extract sensitive, confidential information from the host.

It is the responsibility of security software providers to quickly identify malicious threats such as these and deliver appropriate antivirus solutions. The introduction of such technologies and the attendant risks underscores the need for the concomitant development of security solutions.

Risks posed by third-party applications

With the advent of Windows Vista and the continued use of the Security Development Lifecycle (SDL), Microsoft-authored code has become more robust and more difficult to exploit. Attackers will predictably respond by turning their focus to common third-party applications that are developed by companies without an SDL process in place. Third-party applications may not use accepted software development best practices, such as secure design, secure development practices, code reviews, or developer tools that enhance security, such as Microsoft Visual Studio 2005. Consequently, third-party applications may be less secure than the platform on which they are deployed. Third-party applications can, as a result, create exposure to an otherwise secure operating system.

Security Implications of Microsoft Windows Vista

These third-party applications could include third-party security software (such as antivirus technology), Web browsers, instant message clients, email clients, document viewers, and office suites. They may include applications that have a significant user base, either globally or locally. Symantec has already observed the emergence of a number of zero-day vulnerabilities, that were discovered in regional office productivity suites and used in targeted attacks.

Due to the security improvements presented in Windows Vista, third-party drivers may be targeted as a means of gaining kernel-level access on compromised hosts. By targeting vulnerable third-party drivers, attackers could potentially bypass the improvements in Windows Vista that are designed to prevent compromise (running applications with non-administrative user privileges).

Only by implementing secure development practices can developers ensure the optimal security of their applications. The failure to employ all available secure development measures increases the probability of the discovery and successful exploitation of vulnerabilities.

No Silver Bullet

Microsoft Windows Vista in and of itself is not a security solution; rather it is a more secure version of Microsoft Windows. Symantec continues to see the user as the weakest link, as social engineering attacks become more elaborate in order to undermine the security technologies within Windows Vista. Symantec also predicts that that the greatest exposure to risk will come from third-party software, which is less likely to employ all the security features available—at least in the short to medium term.

While Microsoft has invested heavily in multiple technologies to mitigate the effect of memory corruption and memory manipulation vulnerabilities, Symantec anticipates the discovery of new techniques for successful exploitation. However these new techniques will not be generic; instead, they will be environment-, configuration-, and vulnerability-specific.

In summary, both enterprises and consumers will continue to face threats that Windows Vista and its built-in security features cannot protect against. This is, in part, due to the slow pace at which operating systems can evolve in relation to today's ever-changing threat landscape.

About Symantec

Symantec is a global leader in infrastructure software, enabling businesses and consumers to have confidence in a connected world.

The company helps customers protect their infrastructure, information, and interactions by delivering software and services that address risks to security, availability, compliance, and performance. Headquartered in Cupertino, Calif., Symantec has operations in 40 countries.

More information is available at www.symantec.com.

For specific country offices and contact numbers, please visit our Web site. For product information in the U.S., call toll-free 1 (800) 745 6054.

Symantec Corporation
World Headquarters
20330 Stevens Creek Boulevard
Cupertino, CA 95014 USA
+1 (408) 517 8000
1 (800) 721 3934
www.symantec.com

Copyright © 2007 Symantec Corporation. All rights reserved. Symantec and the Symantec logo are trademarks or registered trademarks of Symantec Corporation or its affiliates in the U.S. and other countries. Microsoft, Visual Studio, Windows, Windows Server, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. Other names may be trademarks of their respective owners. This document is provided for informational purposes only. All warranties relating to the information in this document, either express or implied, are disclaimed to the maximum extent allowed by law. The information in this document is subject to change without notice.
02/07 12065948