



An Introduction to Security in Windows 7

Microsoft has continued its investment in security by adding new technologies to Windows 7 as well as enhancing many of the technologies introduced in Windows Vista. This provides an overview of the new security features and enhancements you'll find in Windows 7.



By Chris Corso

May 15, 2009

This article originally appeared on Microsoft TechNet at [redacted]

Windows Vista introduced a variety of new security technologies that had a significant impact on the Windows ecosystem. User Account Control made it clear that Microsoft wanted to make easy for users to run Windows without being in the Administrators group. BitLocker introduced full volume encryption for the Windows client. Protected Mode Internet Explorer helped to make browsing the Internet a safer experience.

In Windows 7, Microsoft has continued its investment in security by adding new technologies as well as enhancing many of the technologies introduced in Windows Vista. In this article, I will provide an overview of the new security features and enhancements you'll find in Windows 7.

Windows Biometric Framework

Windows Vista included a redesign of the Winlogon experience. This experience removed the GINA (Graphical Identification and Authentication) infrastructure and added the Credential Provider extension model. The Credential Provider infrastructure was a set of interfaces that allowed consistency when third parties extended the user experience around users entering credentials, and it integrates into the common Windows credential dialog.

For Windows 7, Microsoft has added the new Windows Biometric Framework (WBF). With fingerprint readers becoming far more common, it became clear that defining a common framework for exposing, managing, and using these technologies was necessary to drive development and reliability. The WBF is intended to make it easier to support biometric authentication devices. In Windows 7, WBF supports only fingerprint readers, but it can be expanded in the future.

The WBF core platform consists of these main components:

- Windows Biometric Driver Interface (WBDI)
- Windows Biometric Service (WBS)
- WBF API
- WBF User Experience and Integration Points
- WBF Management

The Windows Biometric Driver Interface (WBDI) is meant to provide a common driver interface for biometric devices. It consists of a variety of interfaces that expose the appropriate data structures and IOCTLs (Input/output controls) for biometric devices to integrate into the biometric framework. Drivers can be implemented in any of the common driver frameworks, including Windows Driver Model, Kernel Mode Driver Framework, and User-Mode Driver Framework (UMDF). UMDF, however, is the recommended driver framework for biometric devices because it provides the additional benefit of greater reliability for Windows in case a crash occurs in the biometric device driver.

The Windows Biometric Service (WBS) is the key component that ties together WBF. WBS interfaces with the biometric device drivers and also exposes the Windows Biometric Framework APIs, allowing applications to interact with these devices.

An important feature of WBS is that it never reveals a user's actual biometric data to unprivileged applications. This is important because, unlike a password, it's very difficult for someone to change their biometric signature once it's been compromised. Instead, the WBS exposes a handle (typically a GUID or a SID) that allows applications to work with the biometric data indirectly.

WBS also manages pools of biometric authentication devices. This enables you to control how biometric devices are used. Certain devices can be used with any Credential Dialog, such as the logon prompt or a UAC prompt. For example, you can set up Parental Controls on your home system, and when elevation is required on the system, you can simply swipe your finger to provide elevation. This pool of biometric devices is referred to as the System pool. There are two other pools of devices. There is the Private pool, which allows applications to offer authentication that is not integrated with the Windows authentication infrastructure. And there is the Unassigned pool, which is for devices that, as you might have guessed, fit neither of the previous two pools.

Each device that is part of a device pool is actually abstracted away by the WBS using a data class called a Biometric Unit. The Biometric Unit plugs into the WBS's Biometric Service Provider (BSP), which implements policies and behaviors specific to a set of biometric devices. The Biometric Unit allows the BSP to provide any facilities that a particular device might not support, such as storing fingerprint data or processing fingerprint data after it's been acquired by a device.

The third major component of the WBF is the set of APIs, also known as the WinBio* APIs, that can be used by applications and user mode components to directly interact with the devices. This includes interacting with a device during the original enrollment process for obtaining a user's fingerprint and correlating it with a particular user account, as well as the task of verifying a user for logon or UAC. These APIs also expose data about the specific biometric device and its characteristics. In addition, the WBF APIs can be extended to allow an application to interact with proprietary aspects of a particular device.

The WBF exposes two main ways to configure the use of biometric devices. For end users, there is a Control Panel applet, which is exposed in a few locations. You can find the Biometric Devices Control Panel under Hardware and Sound. From this location, the user can launch a third party fingerprint management application. Windows 7 does not provide a built-in fingerprint management application, so any third party vendor or OEM will have to write its own. (Note that the Windows Biometric Framework supports local and domain logon, as well as fingerprint-based UAC through the built-in Biometrics Credential Provider.)

The Windows Biometric Framework can also be managed through Group Policy. An administrator can enable or disable the entire framework, as well as manage what types of logons can use biometrics (for example, local and domain logons can be configured differently).

Extending Authentication Protocols

Windows 7 enhances the home and small network experience with a feature called Homegroup. Users can share data, such as media files, between computers in a home and use an online ID to authenticate between these computers. Users must explicitly link their Windows user account to an online ID in order for this functionality to work. Authentication is enabled by a new protocol called Public Key-based User to User or PKU2U.

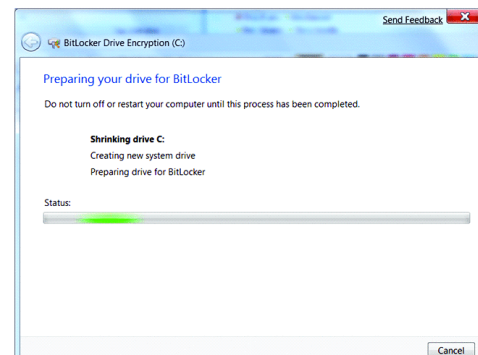
Windows 7 also introduces an extension to the Negotiate authentication package, Spnego.dll. SpNego is the feature that decides which authentication protocol should be used when authenticating. Before Windows 7, it was typically a choice between Kerberos and NTLM (Windows Challenge/Response). The NegoEx extension is treated as an authentication protocol by Windows and it supports two Microsoft security support providers: PKU2U and Live. It's also extensible to allow for development of other security support providers.

Both of these features work when connecting to another computer in the Homegroup using an online ID. When one machine connects to another, the negotiate extension calls the PKU2U security support provider on the logon computer. The PKU2U security support provider obtains a certificate from the certificate authority policy engine and exchanges the policy (along with other metadata) between the peer computers. When validated on the peer computer, the certificate is sent to the logon peer for validation, the user's certificate is mapped to a security token, and the logon process is completed.

BitLocker Core Enhancements

With Windows Vista, Microsoft introduced BitLocker. This is a full volume encryption solution designed to protect the data on laptops and desktop machines, such as branch office servers, even if the machine is lost or falls into the wrong hands. In Windows 7, many enhancements have been made to the management of BitLocker. These include consistent enforcement through all interfaces (the UI, the manage-bde command line tool, and the WMI provider) and separate Group Policy settings for fixed data drives. There are also new Group Policy settings that allow you to update your passwords and integrate with Smart Cards on non-OS drives, and you can also change the behavior related to automatic unlocking.

In Windows Vista, there have been complaints about it being difficult to partition the OS drive to prepare for a BitLocker installation—especially when the operating system is already installed. This problem has been addressed with two enhancements found in Windows 7. First, by default during Windows 7 setup, users will get a separate active system partition, which is required for BitLocker to work on OS drives. This eliminates a second step that was required in many environments. In addition, you can partition a drive for BitLocker as part of BitLocker setup if you do not already have a separate system partition. (see Figure 1).



DOWNLOADS

- Get your hands on Windows 2008 R2 and try it for yourself
- Download Forefront Server Security Management Console
- Windows 7 Enterprise 90-Day Trial
- Download the Windows Server 2008 R2
- Download Microsoft SQL Server 2008 R2
- SharePoint Server 2010 Download
- Download Microsoft Office Professional Plus 2010
- System Center Essentials Download
- Microsoft Business Product Standard Suite

Figure 1. Preparing a Drive for BitLocker

BitLocker To Go

One of the most visible and most important additions is BitLocker To Go, which is designed to protect data on removable data drives. It allows you to configure BitLocker Drive Encryption on USB flash drives and external hard drives. Design goals for BitLocker To Go called for the feature to be easy to use, for it to work on existing drives, to allow for the recovery of data if necessary, and to enable the data to be usable on Windows Vista and Windows XP systems. There are many management enhancements for IT managers to take advantage of with this feature. The most notable is a new Group Policy setting that lets you configure removable drives as Read Only unless they are encrypted with BitLocker To Go. This is an excellent step forward in ensuring that critical corporate data is protected when a USB flash drive is misplaced by an employee.

Also notable is the ability to recover data from any BitLocker To Go device when the data is inaccessible. This technology, called a Data Recovery Agent, was ported from the Encrypted File System (EFS) feature and allows easy recovery of corporate data on a portable drive using the key created by the enterprise.

Getting BitLocker To Go functionality to work on Windows XP and Windows Vista required some reengineering of the core BitLocker feature. To do this, the team refactored the method by which BitLocker protects FAT volumes. BitLocker behavior was modified to overlay a "discovery volume" onto the physical, original volume and virtualize the blocks overwritten. The discovery volume contains the BitLocker To Go Reader as well as a readme file. This is called a Hybrid BitLocker drive. By default, when a FAT drive is encrypted, a hybrid BitLocker drive is created. The discovery drive is visible only on the Windows XP and Windows Vista operating systems.

The reader will also be available on the Microsoft download center after Windows 7 is released. The application provides read-only access to BitLocker drives that utilize the password key protector. Note that smart card authentication is not available when using the BitLocker To Go Reader.

UAC Improvements

User Account Control (UAC) is an often misunderstood technology. First off, it's actually a collection of features rather than just a prompt. These features include File and Registry Redirection, Installer Detection, the UAC prompt, the ActiveX Installer Service, and more. These features are all designed to allow Windows users to run with user accounts that are not members of the Administrators group. These accounts are generally referred to as Standard Users and are broadly described as running with least privilege. The key is that when users run with Standard User accounts, the experience is typically much more secure and reliable. Many developers have started to target their applications to work well for Standard Users. Businesses now have a clearer path toward deploying Standard User accounts, allowing these businesses to reduce support costs and the overall TCO (total cost of ownership) of its computers. In the home, families can use Standard User accounts for children along with Parental Controls to create a safer environment.

Windows 7 includes numerous enhancements to improve the Standard User experience, and new configuration settings provide more control over the User Account Control prompt when run in Administrator Approval Mode. The goal is to improve usability while continuing to make it clear to independent software vendors that the default security context they should be targeting is that of a Standard User. In practice, these changes mean that users are not prompted for common administrative tasks in Windows 7. This is the setting that says "Notify me only when programs try to make changes to my computer."

The way this works is fairly straightforward. During process creation, the policy is checked to see if this setting is enabled. If the process being created is part of Windows, which is verified by checking the Windows catalog files for its signature, the process is created without a prompt. This setting does not prompt when you change Windows settings but instead enables you to focus on administrative changes being requested by non-Windows applications (such as installing new software). For people who want greater control changing Windows settings frequently, without the additional notifications, this setting results in fewer overall prompts and enables users to zero in on the key remaining notifications that they do see.

The other significant change is that several components no longer require Administrator privileges. For example, users can configure whether their desktops should be displayed in High DPI mode, a commonly used feature as computer screens get larger and pixel sizes get smaller. Another example is that Standard Users can now reset their network connection when physically logged into the computer, a common request Microsoft has heard from both home users and enterprises.

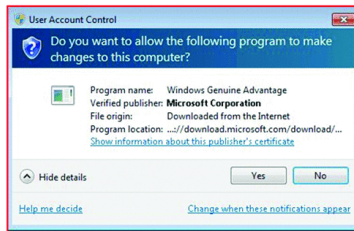


Figure 2. User Account Control When Installing an ActiveX Control

Reducing prompts also means streamlining areas where multiple prompts were encountered for a single user action. On Windows 7, for instance, installing ActiveX controls in Internet Explorer is much smoother. On Windows Vista, Internet Explorer 7 would create the Ielinst.exe process to perform the installation of an ActiveX control. This resulted in a UAC prompt that asked if you wanted "Install an IE Add-On" to run with Administrator privileges. This prompt didn't provide much context about exactly what was being installed and Internet Explorer would immediately prompt you to approve a particular control. On Windows 7 with Internet Explorer 8, the install process has been modified to use the ActiveX Installer Service, which will extract the ActiveX control's publisher information and display it during the installation experience (see Figure 2). The new approach also removes the second prompt during the installation of an ActiveX control.

AppLocker

The ability to control which applications a user, or set of users, can run offers significant increases in the reliability and security of enterprise desktops. Over an application lockdown policy can lower the TCO of computers in an enterprise. Windows 7 adds AppLocker, a new feature that controls application execution and makes it even easier to author an enterprise application lockdown policy.

Durga Prasad Soyama and I discussed application lockdown policies in last year's security issue in an article titled "Application Lockdown Policies in Windows 7". In the article, we detailed a number of challenges that an enterprise must overcome when creating such a policy. Some of these challenges include the following:

- Understanding what software is used in your environment
- Knowing which applications various users should be allowed to run
- Knowing how to author the necessary policy
- Determining whether a policy will work correctly when deployed

To address these hurdles, AppLocker offers a new approach that can audit how an application lockdown policy will work. It provides the ability to control how users run all types of applications—executables, scripts, Windows Installer files, and DLLs. And it offers new application lockdown policy primitives that are more specific and are not subject to break as easily when an application is updated. Windows 7 also includes support for legacy Software Restriction Policy (SRP) rules, but there is no support for the new AppLocker rules on Windows XP and Windows Vista.

The enforcement modes are all implemented on top of AppLocker's underlying enforcement agent, which is implemented in the applocker.sys driver. This driver offers the ability to have kernel mode rule checking for such events as process creation and DLL loading. For applications that implement enforcement in User Mode, the legacy SafeIdentifyLevel API is used to determine whether an application can run. But SafeIdentifyLevel will now hand the enforcement check over to a service to perform the actual verification of the binary and policy. This is a significant architectural enhancement over the legacy Software Restriction Policies feature.

AppLocker is meant to make it easy for IT pros to author a simple set of rules that express all of the applications that are allowed to run and ensure that the rules are resilient to application updates.

To author AppLocker policy, there is a new AppLocker MMC snap-in in the Group Policy Object Editor snap-in in LX, which offers an incredible improvement in the process of creating AppLocker rules. There is a wizard that allows you to create a single rule, and another wizard automatically generates rules for you based on your rule preferences and the folder that you select (see Figure 3).

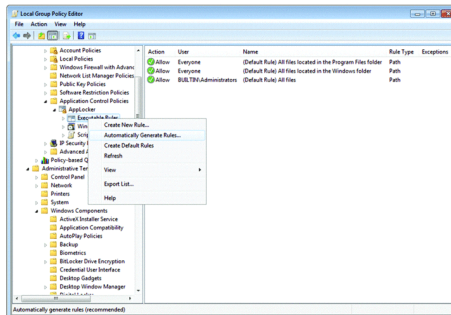


Figure 3. Automatically generate rules for AppLocker policy

You can review the files analyzed and remove them from the list before rules are created for them. You can even get useful statistics about how often a file has been blocked or test AppLocker policy for a given computer.

In the previous Software Restriction Policies, it was particularly difficult to create policies that were secure but also wouldn't

break from software updates. This was due to the lack of granularity of certificate rules and the fragility of hash rules that would break when an application binary was updated. To address this issue, AppLocker lets you author a rule that combines a certificate and a product name, file name, and file version. This makes it easy to specify that anything signed by a particular vendor for a specific product name can run. AppLocker policies in Windows 7 have other benefits, as well, including separation between different types of execution (namely EXEs, DLLs, and MSI or script hosts). These file types are fit into four buckets called rule collections, and enforcement is configured separately for each. For example, administrators can enable AppLocker checks for executables without enabling checks of script files.

The AppLocker policy is stored under the HKLM\Software\Policies\Microsoft\Windows\SrpV2 key. The policy is stored in an XML format and is translated by the Application Identity (AppID) service. When policy is processed, the appid.sys driver is notified about new policy by the service through the IOCTL_SRP_POLICY and the driver will reload the policy. The first task when approaching changes to your IT environment is to assess how the environment is currently functioning. Then you can carefully plan and test any changes to ensure they can be implemented smoothly. This is the purpose of the Audit only enforcement mode.

Auditing the enforcement of the AppLocker policy is extremely important. Not only does this let you test a policy before it's enforced, but it also offers you the ability to watch how the policy performs during its lifetime. You'd definitely want to know if a certain set of users needed an application to work at some point. This can be determined by connecting to a system and reviewing the AppLocker audit information to see whether an application lockdown policy was preventing a particular app from running.

The primary channel for AppLocker events is in the Applications and Service Logs that can be viewed in the Event Viewer (eventvwr.msc) application. In order to view these log entries, look for the EXE and DLL and the MSI and Script logs under the Microsoft\Windows\AppLocker\ event channel. Many different events can be generated, including whether an application was allowed or blocked and whether a policy was applied to a system.

Global SACLs and Granular Auditing
Windows 7 extends previous auditing mechanisms to offer new features that let you manage auditing for users rather than just objects and to provide more information about AccessCheck failures for file objects. This enables new auditing scenarios and provides a significant paradigm change related to auditing.

In other versions of Windows, determining whether to audit object access was based on whether the security descriptor of an object included an access control entry (ACE) in its SACL specifying that it should be audited. This made it very easy to monitor a certain registry key or file to see what access was occurring on that object. Unfortunately, there was no method to watch what a particular user was accessing. If you wanted this scenario, you'd likely need to turn on auditing for every resource that the user could possibly interact with and thus every access by any user of the resource would end up in the audit log.

Enabling auditing on a wide enough data set to capture what a user could access is an incredibly arduous process. Each resource needs to be updated to include the audit policy within the SACL and any changes to this policy would require each SACL to be updated. To overcome this limitation, Windows 7 introduces Global Object Access Auditing, which is managed by auditpol.exe and is configurable using Group Policy.

The Global Object Access Auditing includes a "Global SACL" that is an SDCL string stored in the registry with other data related to auditing. Two new APIs have been added to manage the Global SACL: AuditSetGlobalSacl and AuditQueryGlobalSacl. Updating the Global SACL requires the SeSecurityPrivilege, which protects the Global SACL from being updated by a user without administrator privileges.

Security auditing in Windows 7 also provides the ability to understand why access to an object failed or succeeded. This is important information if you're debugging an application failure or trying to understand whether your security policy is effective. Both the Global Object Access Auditing functionality and the inclusion of additional access audit data are implemented in a new kernel mode security API, SeAccessCheckEx. The two resource managers to consume this API will be NTFS and the Detailed File Share in Windows 7, and when enabled, the API will put information in the audit log about why an access attempt succeeded or failed. Thus these features apply to the file system and file shares for now and can be expanded to other resources managers in future versions of Windows.

Wrapping Up

Windows 7 enables new scenarios and makes using Windows a more secure experience. Many of these features have a strong focus on the user experience (for home users, business users, and IT professionals) and allow Windows 7 systems to work even better.

Chris Corio was a member of the Windows Security team at Microsoft for more than five years. His primary focus at Microsoft was application security technologies and management technologies for securing Windows. You can reach Chris at winsecurity@chriscorio.com.



Rate This Content:
Low High
0 after 0 ratings

DNSSEC Validation

Over the past couple years, DNS-related exploits have become a more common problem on the Internet. There is a better understanding of how to poison DNS servers, and attackers are starting to make use of that information. What this means is that a user can potentially visit a Web site and not be absolutely sure that he isn't visiting a different, malicious Web site.

Windows Server 2008 R2 and Windows 7 introduce support for DNSSEC as per the current standards RFC 4033, RFC 4034, and RFC 4035). Windows Server 2008 R2 will allow the DNS Server to provide origin authority and data integrity artifacts. Basically, a server will be able to attach digital signatures to DNS data in responses as well as validate data received from other DNS servers.

Windows 7 is the first client operating system to include the necessary pieces to allow the client to verify that it is communicating securely with a DNS server and verify that the server has performed DNSSEC validation on its behalf. This technology is currently being tested to ensure the maximum compatibility with current Internet infrastructure and aims to play a continuing role in securing DNS data in the future.

The Network for Technology Professionals

Search:

Copyright 2010 QuinStreet Inc. All Rights Reserved.

Whitepapers and eBooks

Webcasts

Downloads and eKits

Tutorials and Demos

