Global Information Assurance Certification Paper

SANS GIAC Windows NT Security Practical Assignment

VERSION 1.0

January 16, 2005

JIM LANGSTER

Table of Contents

BACKGROUND INFORMATION

Much of the Windows NT policy recommendation in this document is taken from the books "Windows NT 4.0 Guidelines for Security, Audit and Control", Microsoft's web site: http://www.microsoft.com/security, and SANS Windows NT Security Step by Step guide.

One of the primary objectives of this network and computer security policy is to simplify the effort of the network user. It is the goal of this policy that users will be able to access all network resources after a single network logon authentication. Currently most networked server applications each have their own security/authentication mechanisms.

With Windows NT Server the infrastructure is now in place to begin achieving this goal of single network logon authentication.

WINDOWS NT SERVER SECURITY POLICY

NT Server Password Policy

Policy	Setting/value	Explanation
Password maximum age	45 days	you must change your password at least every 'n' days
Password minimum age	7 days	you must use a new password for at least 'n'7 days
Password minimum length	6 characters	use alphanumeric password for greatest security
Password uniqueness	8	prevent user from using last 'n' passwords
Password never expires		only enabled for "service" accounts, these will need to change periodically for security reasons
Bad logins before lockout	3	you will be locked out of the network after 'n'3 bad attempts
Length of lockout	30 minutes	you will be locked out of the network for 'n' minutes

Enforce policy using passfilt.dll (See attached instructions to use .DLL file) requiring characters from at least 3 of the following four characteristics:

Description	Example
English upper case letters	A, B, C, …Z
English lower case letters	a, b, c, …z
numerals	0, 1, 2, 3, …9
non-alphanumeric (special characters)	({}[],.<>;:'"?/|\`~!@#$%^&*()_-+=)

NT Server Access Policy

Policy	Setting/value	Explanation
After hours disconnect	enable	force users to log off after logon hours, currently no effect since logon hours are 24X7

NT Server Account Policy

Policy	Setting/value	Explanation
Guest account	disabled	only domain or trusted domain users can access server resources
"machine" accounts	no longer allowed	"machine" accounts will only be created for specialized application servers
ftp logon		anonymous or user account

NT Server File System Policy

Policy	Setting/value	Explanation
Server file system	NTFS	allows specifying security attributes at the file/directory levels

INSTRUCTIONS TO IMPLEMENT PASSFILT.DLL

To ensure Strong Password functionality occurs throughout your domain structure, make the following changes on all primary domain controllers (or stand-alone servers, where needed).

PASSFILT.DLL is not necessary on backup domain controllers since the PDC is the only machine where changes to the domain accounts database are made. However, it should be installed on all BDCs because they can be promoted to PDC. If a BDC without PASSFILT.DLL is promoted to PDC, then strong password enforcement will be lost but there will be no other adverse effects.

WARNING: Using Registry Editor incorrectly can cause serious, system-wide problems that may require you to reinstall Windows NT to correct them. Microsoft cannot guarantee that any problems resulting from the use of Registry Editor can be solved. Use this tool at your own risk.

1. Install the latest Windows NT 4.0 service pack.

2. Copy Passfilt.dll to the %SYSTEMROOT%\SYSTEM32 folder.

3. Use Registry Editor (Regedt32.exe) to add the value "Notification Packages", of type REG_MULTI_SZ, under the LSA key.

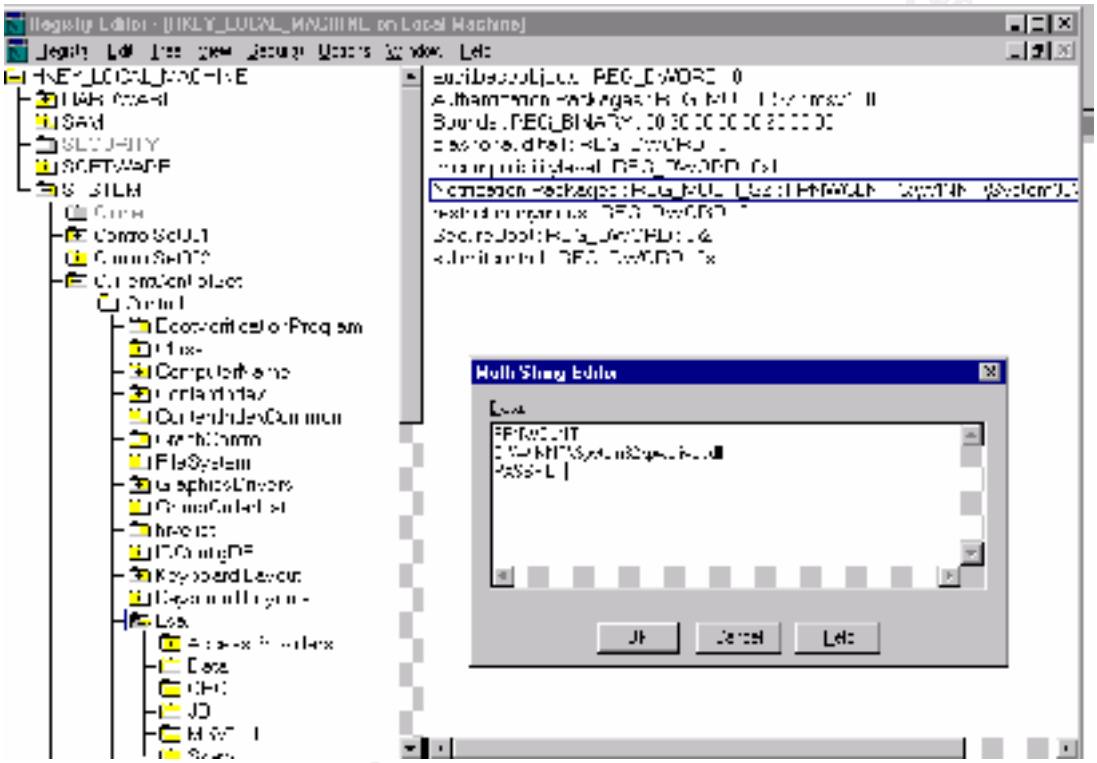
   NOTE: If this key already exists, go to Step 4.

   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa

4. Double-click the "Notification Packages" key and add the following value:

   NOTE: If the value FPNWCLNT is already present, place the following entry beneath the FPNWCLNT entry:

   PASSFILT



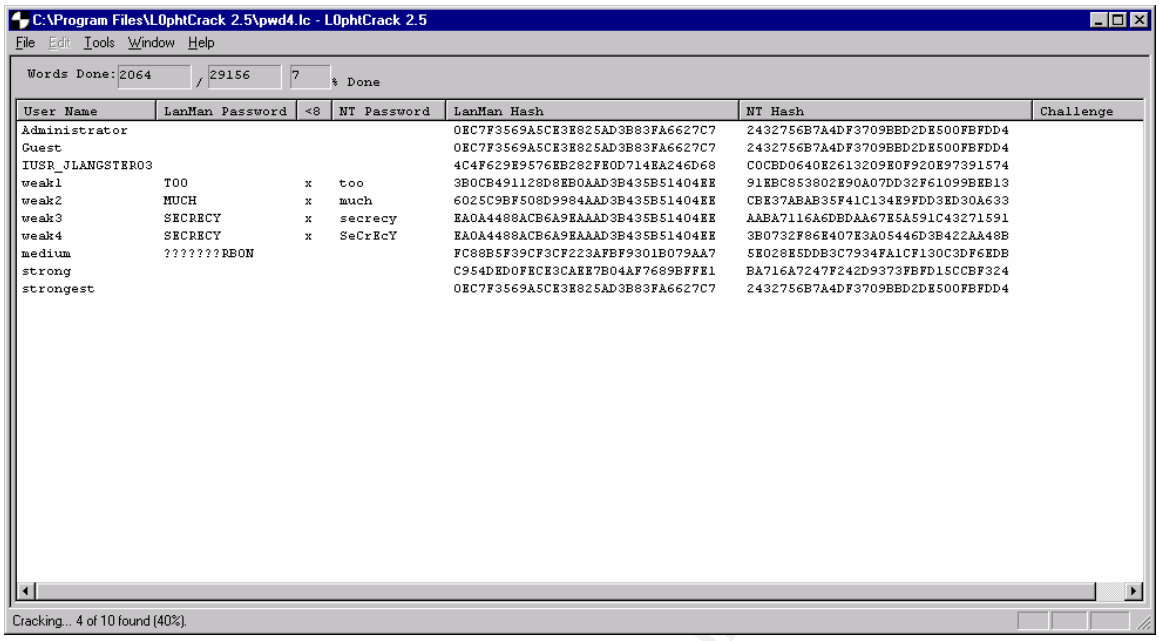
Passfilt Implementation

5. Click OK and then exit Registry Editor.

6. Shut down and restart the computer running Windows NT Server.

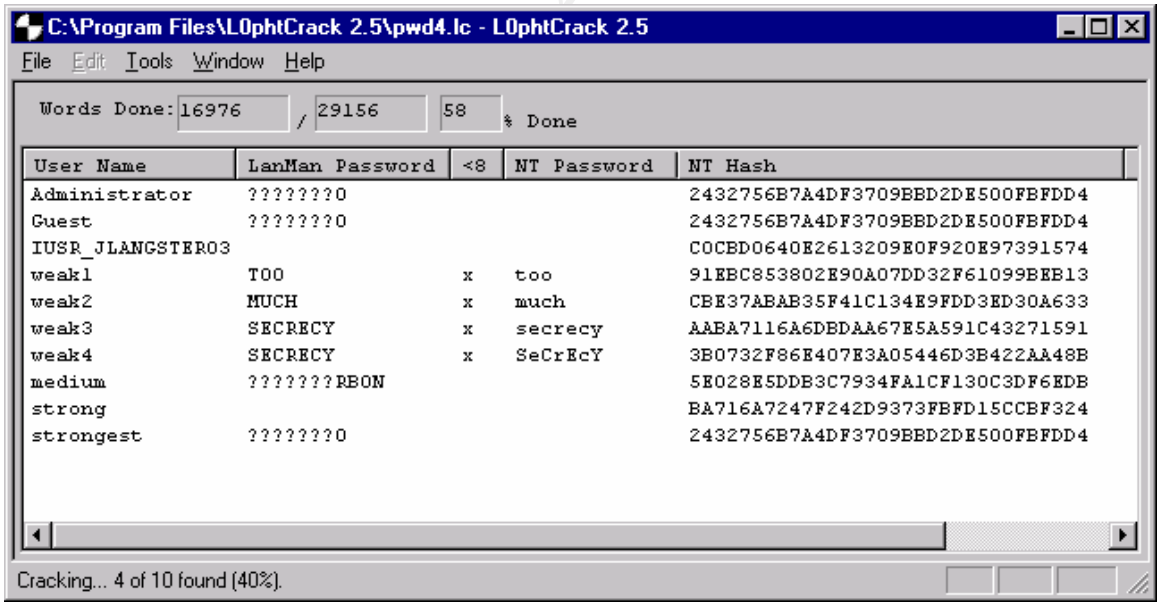
USING L0PHTCRACK TO SHOW DIFFERENCE BETWEEN LANMAN AND NT PASSWORD HASH

Microsoft Windows NT Server 4.0 and Microsoft Windows NT Workstation 4.0 store both an NT and LAN Manager version of the password for user accounts. These password versions are encrypted using technology referred to as a "HASH". The basic difference between the two types is the LAN Manager password does not recognize case-sensitive characters. The main reason for this is for legacy applications and operating systems such as Microsoft Windows 3.x and Windows 95. The LAN Manager password hash is created by converting the password to uppercase. Then the password is truncated to 14 characters and split into 2 parts. If the password is 7 or less characters, the last 7 characters of the LanMan Hash are the same. An illustration is demonstrating that below.



L0phtCrack Screen Shot 1

Note that Users “weak1”, “weak2”, “weak3” and “weak4” all have the last 16 bytes of the 32 byte hash the same AAD3B435B51404EE. This has enabled L0phCrack to correctly identify these accounts as ones having a password of 7 characters or fewer.



L0phtCrack Screen Shot 2

For comparison, here are the NT Hash passwords displayed:
The NT passwords are case sensitive, but can lose this attribute. If the client computer is not running NT, the password will be treated as non case sensitive. If a non-NT client

computer issues the NET PASSWORD command to change the password, the resulting password becomes case insensitive for any client computer Windows operating system

MICROSOFT WINDOWS 2000

Strong Password Functionality Included with Microsoft Windows 2000

The functionality described above for the Passfilt.dll file for Windows NT 4.0 has been included in the operating system security components for Windows 2000. You can enable strong password enforcement in Windows 2000 by starting the Local Computer Policy snap-in and enabling the **Passwords must meet complexity requirements** setting in Computer Configuration\Software Settings\Account Policy\Password Policy.

SQL SERVER SECURITY POLICY

The objective of Windows NT Server security is to provide single logon validation to any and all services on the network. Therefore the preferred access validation mechanism is through the SQL Server integrated (integrated with NT Server) security mechanism. Since some client applications will not be able to use this mechanism, the interim solution is to use the SQL Server mixed security mechanism. In this scenario the preferred logon is through SQL Server integrated security and the alternative mechanism is through SQL Server standard security.

SQL Server Security Policy

Policy	Setting/value	Explanation
SQL security	mixed with migration to integrated	will attempt to validate user against network account
sa password		• single password for production servers • single password for development servers

You can use SQL Server facilities to implement many useful data access security strategies:

- **Remove the Guest Login** Drop the guest login from database access permissions.
- **Identify Common Work Groups** Identify common work groups with common database permission requirements and create a group-based security profile.
- **Provide an Anonymous Logon Account for the Domain** If IIS is on a stand-alone server and other remote resources will be accessed from that IIS machine, it is generally the best approach to make the anonymous logon account a domain

account. This means that whenever authentication is required, the account will always be available to other network servers for validation. The most obvious advantage here is that it satisfies SQL Server's Named Pipes authentication.

- **Use Standard Security for IIS Applications** If your IIS application uses SQL Server, use the standard security mode. This will give you maximum flexibility in deploying your application to network workstations.

- **Use Views** Restrict data access by developing unique views that limit the amount of data a user can see or modify. For example, you could allow access to some columns of a table but restrict access to other columns containing sensitive data. Access to these views is regulated through SQL server permission schemes. Some corporate development sites prefer this approach because it provides the highest degree of security and is the easiest to implement.

- **Use Stored Procedures** If data access must be very secure, consider using only stored procedures for all data viewing and modification. You can simplify the administration of data access permissions by granting EXECUTE permission to run certain stored procedures. This avoids assigning discrete permissions to all of the tables and views referenced within the stored procedure or embedded SQL statements. If you protect data access by using only stored procedures, remove all SELECT, UPDATE, INSERT, and DELETE privileges from every table and view in the database.

- **Restrict Query Tools** If some users will be using query tools, consider assigning each user two logon accounts: a primary logon account for queries, and an application logon account for using your application. The primary query logon account should be their Windows NT user identification and restricted to READ ONLY. Thus, when the user runs the query tool, the query tool will receive the Windows NT logon account permissions and be limited to READ ONLY for the permitted databases and objects. The second logon account is for your application, where you control data access with the application's built-in methods and processes.

- **Construct an Audit Trail** You can easily audit data access by creating triggers that automatically execute each time a table is modified. User permissions are not required to execute a trigger.

RENAMING THE ADMINISTRATOR ACCOUNT

Because every NT system has an Administrator account—and because that account has the most access and privileges—an intruder will probably start there when trying to gain unauthorized access. The Administrator account can't be locked out, it can't be disabled, and you can't remove its privileges—but you can rename it.
When you rename the Administrator account, you can call it anything you want. Making the name fairly normal is probably most appropriate. Also, be sure you remove the comment indicating that this is the built-in Administrator account.

Here's one more little trick. Rename the Administrator account, giving it a less obvious name. Then, create a new user account called Administrator, give it restricted privileges, and don't make it a member of a group.

USE OF TWO ACCOUNTS

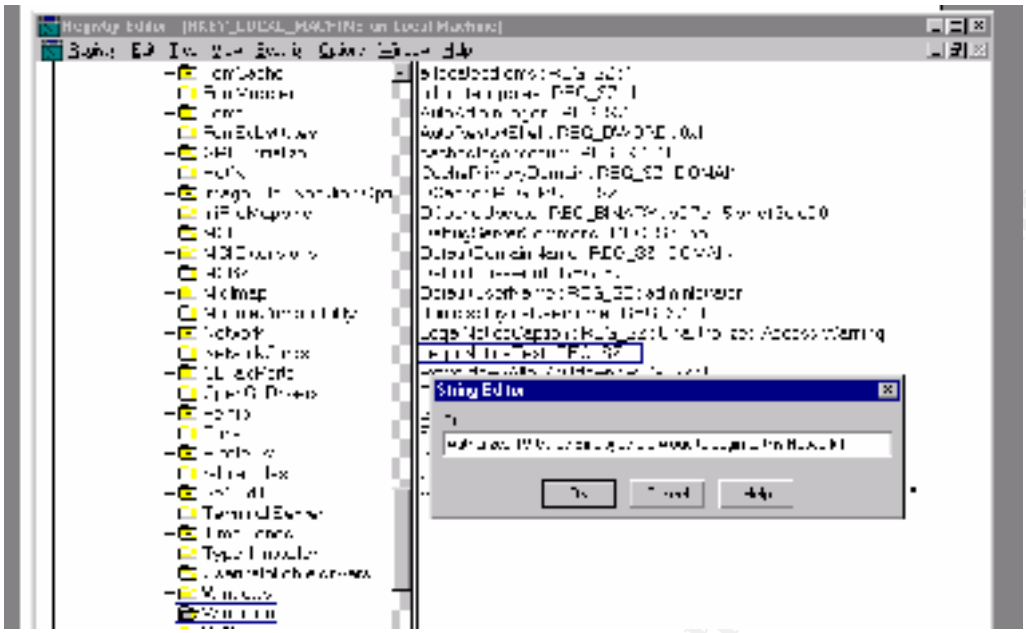
Ideally, the computer administrative staff should use the two-account method. This method requires users with Domain Admin privileges to only use the Domain Admin account when administrative duties require it. All other activities, users will use a standard Domain User account.

DISABLING THE GUEST ACCOUNT

Because the guest account is a built-in account, this will be one of the first attacks a hacker will attempt. On a computer with Microsoft Windows NT Workstation, the guest account is enabled by default. It is the opposite for Microsoft NT Server. Similarly to the Administrator account, the guest account cannot be deleted, but can be renamed and should be disabled on all computers that it is not required.

ADDING A SECURITY BANNER

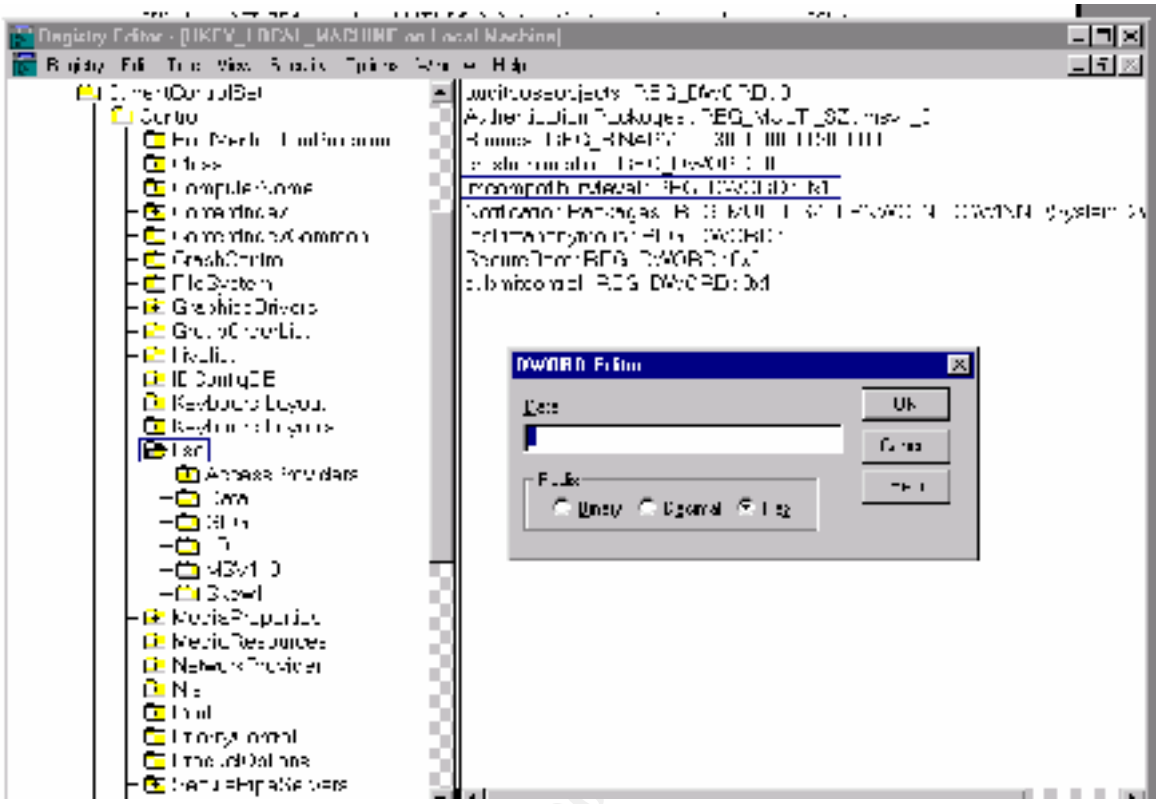
In some situations you may want to add a security warning to Windows NT that users will see before they login. Doing so is easy, but requires you to edit the registry. As always, you should exercise extreme caution when editing the registry, and make a backup before you begin because making a mistake in the registry can destroy Windows. To create a security warning, open the registry editor and navigate to \HKEY_LOCAL_MACHINE \SOFTWARE \Microsoft \Windows NT\CurrentVersion\WinLogon. Next, double-click on the LegalNoticeCaption value, and enter the following text into the data field: Unauthorized Access Warning. Now, double-click on the value named LegalNoticeText and enter your security warning in the data field. Log out. When you do, you should see your security warning displayed before you are even given the chance to log back in.



Security Banner

NTLM SETTINGS

Windows NT SP4 introduced NTLMv2 Authentication which implements 128bit encrypted keys and provides for a method to eliminate LANMAN hashes for NT clients. LANMAN Password authentication is easy to attack since it uses upper-case letters (reducing the set from 52 to 26 letters) and limiting password length to 7 characters (effectively from a dictionary attack viewpoint). To modify LANMAN values:
Hive: HKEY_LOCAL_MACHINE
Key: SYSTEM\CurrentControlSet\Control\LSA
Name: LMCompatibilityLevel
Type: REG_DWORD
Value: 5 : DC refuses LM and NTLM responses (accepts only NTLMv2)
Value: 4 : DC refuses LM responses
Value: 3 : Send NTLMv2 response only
Value: 2 : Send NTLM response only
Value: 1 : Use NTLMv2 session security if negotiated
Value: 0 : default - Send LM response and NTLM response; never use NTLMv2 **session** security



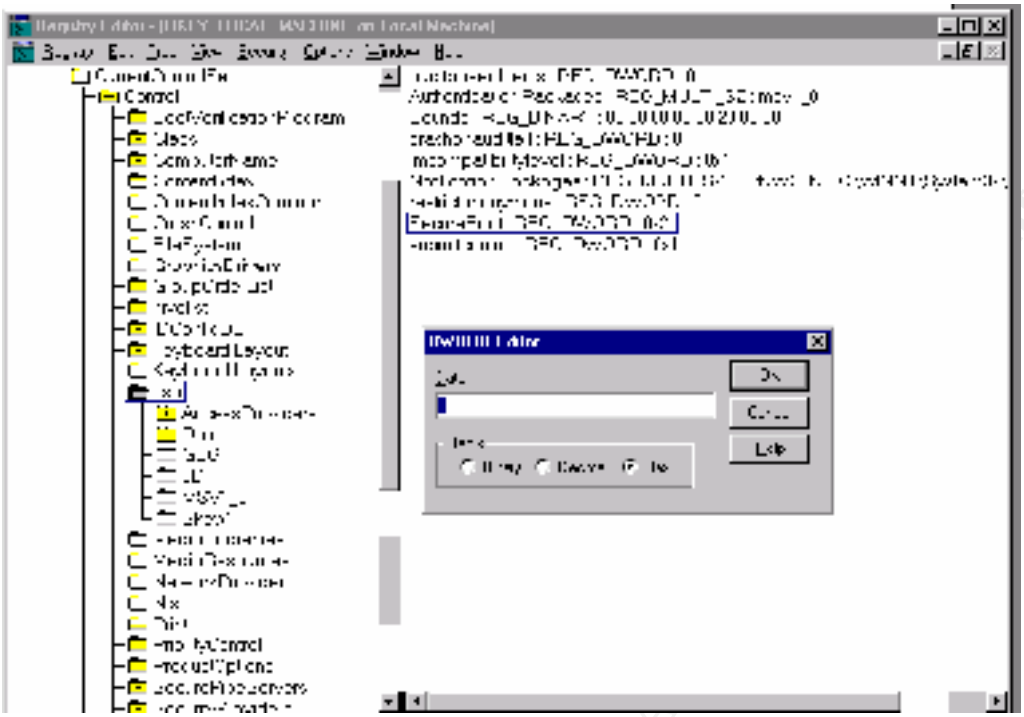
NTLM Registry Settings

You MUST read KB Q147706 - How to Disable LM Authentication on Windows NT to understand compatibility issues. It lists gotchas and implementation suggestions. SP4 added levels 3-5 and added considerable complexity. Also see Q175641 - LMCompatibilityLevel and Its Effects
For commercial networks, I suggest setting LMCompatibilityLevel to 1 on all NT workstations and servers. NTLMv2 will be used when possible and allow LANMAN compatibility for Win9x and Mac clients. In high-risk networks, set LMCompatibilityLevel to 5 - eliminates Win9x and its weak authentication requirements.

ENCRYPTING THE SAM

Syskey strongly encrypts the password hashes in the Windows NT SAM. Syskey will help to protect the passwords stored in ERDs and backup tapes. The system will not boot without the encryption key. For background: Encrypt hashes in SAM with 128-bit encryption using SYSKEY.
How can you determine whether Syskey has or has not been applied to enhance NT's security? You can set down at the console of each NT and issue the Syskey command. The Syskey command will tell you whether it is in place, and if it is, whether the startup key is stored locally on the hard drive; startup key must be entered at the console at boot; or the startup key is stored on a floppy disk which must be inserted in the floppy drive when the system prompts for the diskette. Not a realistic solution if you have hundreds of systems spread around the country.

How does NT know that Syskey has been applied to a system? The presence of the SecureBoot value means Syskey has been applied. Its value reveals the method Startup Key must be accessed:
Hive: HKEY_LOCAL_MACHINE
Key: SYSTEM\CurrentControlSet\Control\Lsa
Name: SecureBoot
Type: REG_DWORD
Value: 0x1 Startup Key stored on local hard drive
Value: 0x2 password Startup Key
Value: 0x3 Startup Key stored on floppy disk



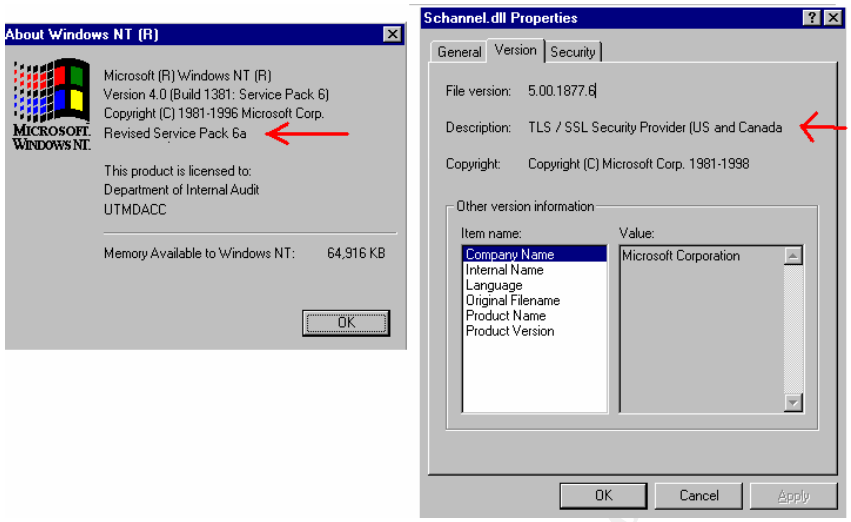
Security using the Syskey Command

INQUIRING THE SERVICE PACK

1. Through inquiry, determine network policy for monitoring availability and installation of service packs and hot fixes. The best remedy to reduce your vulnerabilities is to stay up to date with Service Packs and Hotfixes. To determine what Service Pack you are currently running, do the following steps:

2. For each domain controller:
   1. Click Start, click Run.
   2. Type "Winver" and press enter.
   3. Note the Service Pack installed: _Revised Service Pack 6a___.
   4. Click Start, point to Programs, and then click Windows NT Explorer.
   5. Click the Winnt folder and then click the System 32 folder.
   6. Right click on Schannel.dll, click Properties, click version tab and then view description.
   7. Note encryption level: __U.S. and Canada (128 bit)_
      "Export version" is 40-bit. "U.S. domestic version" is 128-bit.

Attach Screenprint:



Server Name	Service Pack Installed	Tickmark	Compensating Controls / Notes
TVGuide	6a	4 γ	No additional hot fixes applied

If you are using System Management Server (SMS), you can easily package Service Packs and distribute them. The following steps for installing SP6a with Systems Management Server are illustrated using the Distribute Software Wizard. It is not necessary to use the wizard; the package, program, and advertisement can be manually created one at a time in the Systems Management Server Administrator console.
To Install SP6a Using Systems Management Server 2.0

1. Download the SP6I386.EXE from the following Web site: http://www.microsoft.com/ntserver/nts/downloads/recommended/SP6/128bitX86/default.asp
    a. Extract the files to a TEMP directory.
    b. Create a folder named **NT4SP6**a.
    c. Copy all the files from the TEMP folder into **NT4SP6a** folder.
    d. In SMS, setup the package with **NT4SP6a** as my source folder.
    e. Create the following command line: **update\update.exe/q**
2. From the SMS Administrator console, start the Distribute Software Wizard.
3. Click **Create a new package from a definition**.

4. Browse to the folder you created in step 1 and select the Nt4sp6.pdf file. SMS converts this automatically to SMS 2.0 PDF (.sms) format in the SMS\Scripts\00000409\Pdfstore folder.
5. Specify how SMS should manage the source files. In this example, click **Create a compressed version of the source**. This compresses the files in the source folder you select in the next step and moves them to selected SMS distribution server shares.
6. Select the source folder. This should be the path to the folder you created in the first step. Ensure that the folder is specified correctly; SMS copies all the files from the specified folder and all subfolders, compresses them into a package file, then decompresses them to the distribution servers you select in the next step.
7. Select distribution points, advertise the program, and select the collection to which you will be advertising.
8. If the service pack installation is to run automatically, assign the program.
9. After you quit the wizard, select packages from the SMS Administrator console, edit the program properties, and view the **General** tab. Examine the command line for the program. In this case, the program is: "Update x86 Windows NT version 4.0." The command line should be:

   i386\update\update.exe /q

   Ensure that the folder structure of the package share on the distribution server is correct and that the command line will run properly from the SMS package share. You can also manually edit the command line to use any of the Update.exe command-line switches listed later in this article.

10. On the **Program Properties\Environment** tab, ensure that the program is set to **Run with administrative rights** or is using a Windows NT client software installation account with proper administrative rights (if the Windows NT user does not have local administrative rights or no user will be logged on). You should use a Windows NT client software installation account only if the installation source files are not on an SMS Distribution Point or the installation needs to access files not on an SMS Distribution Point.

IIS HOTFIX UTILITY

Overview

HFCHECK.WSF is a tool that compares the local IIS installation against a list of hotfixes provided by Microsoft. If the tool finds that a hotfix has not been installed on the machine, it reports an error and writes a warning to the event log.

Implementation

Hotfix Check tools are distributed pursuant to a license agreement that is packaged as part of the installer. Please read the license agreement. By downloading, installing, accessing or using this Hotfix Check tool, you acknowledge that you have read and fully understand this agreement and you agree to be bound by this license agreement.

The Hotfix Check tool consists of two Windows Script Host files HFCHECK.WSF and NOTIFY.JS. HFCHECK.WSF is the actual tool, NOTIFY.JS can be used to customize and extend the functionality.

HFCHECK.WSF
HFCHECK.WSF consults an XML file list – either hosted on the Microsoft site or downloaded to the local machine – for the list of hotfixes available for IIS, then compares this list to the hotfixes installed on the local system. If a hotfix is missing, the tool calls the Notify function in NOTIFY.JS. The current implementation of Notify reports an error on the command-line and writes a warning message to the Application Eventlog, but it is possible to customize it to perform other actions such as stopping the server or sending an e-mail to the administrator. The Notify function is in a separate file (NOTIFY.JS), so that you can easily rewrite the Notify function for your own needs.

Customization
Instead of writing to the event log you might want to start a different action. For instance, you might want to send a mail to your administrators if a missing hotfix is detected:

Sample:

```
Set objMsg = CreateObject("CDONTS.NewMail")
'Set the properties of the Message
objMsg.From = "HotfixNotification@YourCompany.com"
objMsg.To = "Administrators@YourCompany.com"
objMsg.Subject = "Missing Hotfix Detected!"
objMsg.Body = "Microsoft Security Bulletin _
(" + sBulletin + ") " + sTitle + " Link:
_http://www.microsoft.com" + sLink
objMsg.Send
```

Continuous checking
You don't have to invoke HFCHECK.WSF manually. The check can be done continuously, once a day, once a week, once a month, etc, using the Windows Task Scheduler. Here are three samples:

Starting HFCHECK.WSF once a day:
AT.EXE 7:00am /INTERACTIVE /every:M,T,W,Th,F c:\tools\hfcheck.wsf
Starting HFCHECK.WSF once a week:
AT.EXE 5:00pm /INTERACTIVE /every:Wednesday c:\tools\hfcheck.wsf
Starting HFCHECK.WSF once a month:
AT.EXT 5:00pm /INTERACTIVE /every: c:\tools\hfcheck.wsf

Command-line parameters
If you call HFCHECK.WSF without any command-line parameters, it checks the local machine for missing hotfixes. The following command-line parameters extend the functionality of HFCHECK.WSF.

/B <path bulletins file>
If you want to host the bulletins file in your Intranet (or if your servers don't have access to the Microsoft web site), you can host the bulletins file on your own server. The file is available at http://www.microsoft.com/technet/security/search/bulletins.xml; just save it to your local storage. (Be sure to periodically update your copy of the file).
Sample
HFCHECK.WSF /B c:\tools\bulletins.xml

/M <machine1,machine2>
HFCHECK.WSF can also check the hotfix status on remote machines. This is done via WMI. You can specify multiple machine names, separated by a comma. Don't use spaces between machine names.
Sample
HFCHECK.WSF /M iislive,iistest,iisdev

/U <Domain\Username OR Computername\Username> /P <Password>
If you don't specify a username and a password, WMI connects with the current logon credentials to the remote machine. You can however specify a username and a password if the local user isn't Administrator on the remote machine.
Sample
HFCHECK.WSF /M iislive,iistest,iistest /U iisdomain\Administrator /P adminpassword

/?
Displays command-line options.

HFCHECK Limitations
IIS 5.0 Server only
Currently, HFCHECK only works with IIS 5.0 on Windows 2000. We hope to improve the tool soon and make it available for more Microsoft products.

Reinstallation
Reinstalling IIS overwrites the files installed with a hotfix, but doesn't delete the hotfix entries in the registry database. HFCHECK looks for the registry entries and wouldn't necessarily detect a missing hotfix, because the hotfix might still be mentioned in the registry.

Solution:
Delete the IIS hotfix entries under
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\Hotfix
if you reinstall IIS.

CONCLUSION

There are many tools to assist in making Microsoft Windows environment an easier operating system to manage. You must be careful when evaluating these tools that they do not allow additional vulnerabilities to the inherent operating system or enable users without working knowledge of the operating system to make detrimental modifications. These areas mentioned within this guideline are areas that I feel are important and cause little havoc to most corporate environments. I hope that this guideline can be implemented in other environments and can be used to serve companies with a working model of how, at a minimum, a Microsoft Windows environment can be configured and be relatively secure.

REFERENCES

Microsoft Windows NT 4.0 Security, Audit, and Control
James G. Jumes, Neil F. Cooper, Paula Chamoun, and Todd M. Feinman
Microsoft Press

Hacking Exposed – Network Security Secrets & Solutions
Stuart McClure, Joel Scambray, & George Kurtz
Osborne / McGraw-Hill

Microsoft Security Website
http://www.microsoft.com/security

SANS GIAC Windows NT Security Workbook (October,2000) Version 3.7
Jason Fossen & Jennifer Kolde
SANS Institute

Wayne Maples
http://is-it-true.org/nt/atips/

Michael Howard
Microsoft Corporation
mikehow@microsoft.com
(425) 703-1402

Reviewed and incorporated others works
Alex Parks (Enhanced further with L0phtcrack)
Sherri Heckendorn (Enhanced using SMS)

Last Updated: July 25th, 2017

Upcoming Training

SANS Virginia Beach 2017	**Virginia Beach, VA**	**Aug 21, 2017 - Sep 01, 2017**	**Live Event**
SANS Network Security 2017	**Las Vegas, NV**	**Sep 10, 2017 - Sep 17, 2017**	**Live Event**
SANS vLive - SEC505: Securing Windows and PowerShell Automation	**SEC505 - 201709,**	**Sep 18, 2017 - Nov 13, 2017**	**vLive**
Secure DevOps Summit & Training	**Denver, CO**	**Oct 10, 2017 - Oct 17, 2017**	**Live Event**
San Francisco Winter 2017 - SEC505: Securing Windows and PowerShell Automation	**San Francisco, CA**	**Nov 27, 2017 - Dec 02, 2017**	**vLive**
SANS San Francisco Winter 2017	**San Francisco, CA**	**Nov 27, 2017 - Dec 02, 2017**	**Live Event**
SANS Cyber Defense Initiative 2017	**Washington, DC**	**Dec 12, 2017 - Dec 19, 2017**	**Live Event**
SANS Southern California- Anaheim 2018	**Anaheim, CA**	**Feb 12, 2018 - Feb 17, 2018**	**Live Event**
SANS OnDemand	**Online**	**Anytime**	**Self Paced**
SANS SelfStudy	**Books & MP3s Only**	**Anytime**	**Self Paced**