

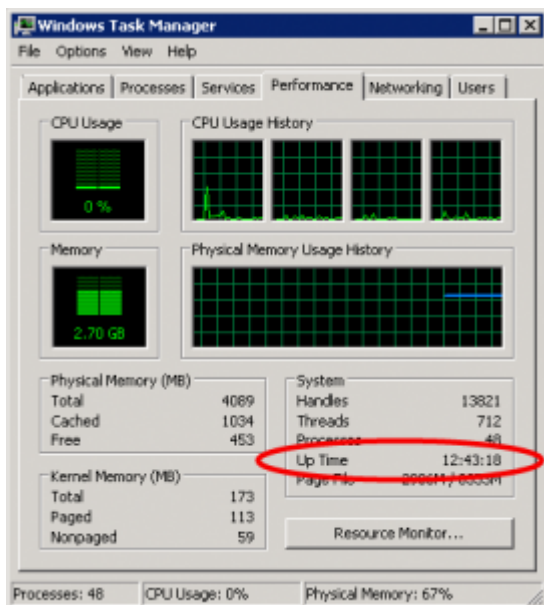
Seven Simple Ways to Find Your Uptime in Windows

Knowing how much time your system has been running is a piece of information useful for administrators. Sometimes you might need it in order to know when the machine has recovered from a possible power failure or any other issue it might have had. While there are plenty of 3rd-party tools that allow one to find the exact uptime of the system, there are several methods that are either built-in Windows Server 2008, or can be easily accomplished.

Method #1: By using the Task Manager

In Windows Vista and Windows Server 2008, the Task Manager has been beefed up to show additional information about the system. One of these pieces of info is the server's running time.

1. Right-click on the Taskbar, and click Task Manager. You can also click CTRL+SHIFT+ESC to get to the Task Manager.
2. In Task Manager, select the Performance tab.
3. The current system uptime is shown under System.



Method #2: By Using the System Information Utility

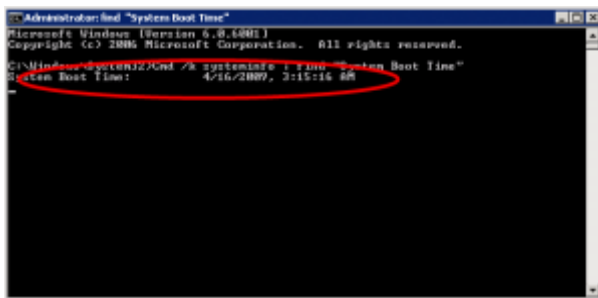
Probably one of the easiest methods to accomplish this task. The Systeminfo command line utility checks and displays various system statistics such as installation date, installed hotfixes and more.

Open a Command Prompt and type the following command:

```
systeminfo
```

You can also narrow down the results to just the line you need:

```
systeminfo | find "System Boot Time:"
```



```
Administrator:cmd "System Boot Time"
Microsoft Windows [Version 6.0.6002]
Copyright (c) 2006 Microsoft Corporation. All rights reserved.

C:\Windows\system32\cmd /k systeminfo | find "System Boot Time"
System Boot Time: 4/10/2009, 3:15:16 PM
```

Method #3: By Using the Uptime Utility

Microsoft have published a tool called Uptime.exe. It is a simple command line tool that analyzes the computer's reliability and availability information. It can work locally or remotely. In its simple form, the tool will display the current system uptime. An advanced option allows you to access more detailed information such as shutdown, reboots, operating system crashes, and Service Pack installation.

Read the following KB for more info and for the download links:

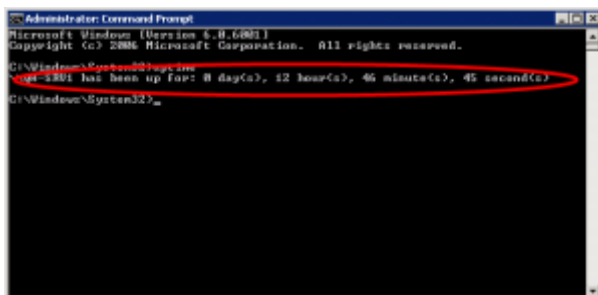
Uptime.exe Tool Allows You to Estimate Server Availability with Windows NT 4.0 SP4 or Higher

<http://support.microsoft.com/kb/232243>

To use it, follow these steps:

1. Download uptime.exe from the above link, and save it to a folder, preferably in one that's in the system's path (such as SYSTEM32).
2. Open an elevated Command Prompt window. To open an elevated Command Prompt, click Start, click All Programs, click Accessories, right-click Command Prompt, and then click Run as administrator. You can also type CMD in the search box of the Start menu, and when you see the Command Prompt icon click on it to select it, hold CTRL+SHIFT and press ENTER.
3. Navigate to where you've placed the uptime.exe utility.
4. Run the uptime.exe utility. You can add a /? to the command in order to get more options.

uptime.exe



```
Administrator: Command Prompt
Microsoft Windows [Version 5.0.6002]
Copyright (c) 2006 Microsoft Corporation. All rights reserved.

C:\Windows\System32>uptime.exe
System has been up for: 0 day(s), 12 hour(s), 45 minute(s), 45 second(s)
C:\Windows\System32>
```

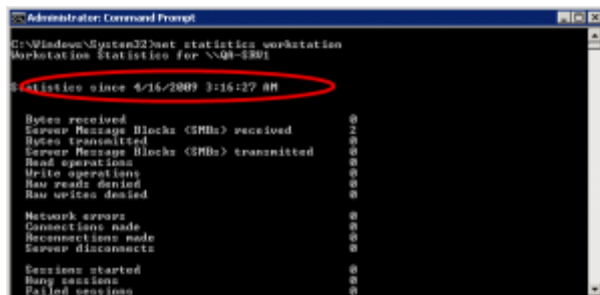
Method #4: By Using the NET STATISTICS Utility

Another easy method, if you can remember it, is to use the approximate information found in the statistics displayed by the NET STATISTICS command.

Open a Command Prompt and type the following command:

```
net statistics workstation
```

The statistics should tell you how long it's been running, although in some cases this information is not as accurate as other methods.



```
Administrator: Command Prompt
C:\Windows\System32>net statistics workstation
Workstation Statistics for \\0-1301
Statistics since 4/16/2009 3:16:27 AM
Bytes received 0
Server Message Blocks (SMBs) received 2
Bytes transmitted 0
Server Message Blocks (SMBs) transmitted 0
Read operations 0
Write operations 0
Raw reads denied 0
Raw writes denied 0
Network errors 0
Connections made 0
Reconnection made 0
Server disconnects 0
Sessions started 0
Hung sessions 0
Failed sessions 0
```

Method #5: By Using the Event Viewer

Probably the most accurate of them all, but it does require some clicking. It does not display an exact day or hour count since the last reboot, but it will display important information regarding why the computer was rebooted and when it did so. We need to look at Event ID 6005, which is an event that tells us that the computer has just finished booting, but you should be aware of the fact that there are virtually hundreds if not thousands of other event types that you could potentially learn from. **Note:** BTW, the 6006 Event ID is what tells us when the server has gone down, so if there's much time difference between the 6006 and 6005 events, the server was down for a long time.

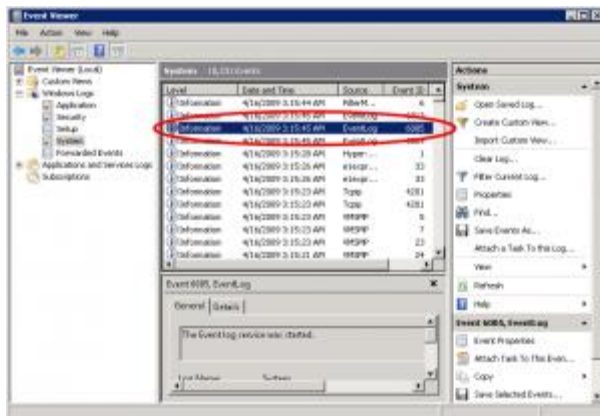
1. Open Server Manager tool by right-clicking the Computer icon on the start menu (or on the Desktop if you have it enabled) and select Manage. Navigate to the Event Viewer.

Note: You can also open the Event Viewer by typing eventvwr.msc in the Run command, and you might as well use the shortcut found in the Administrative tools folder.

2. Click on Event Viewer (Local) in the left navigation pane.

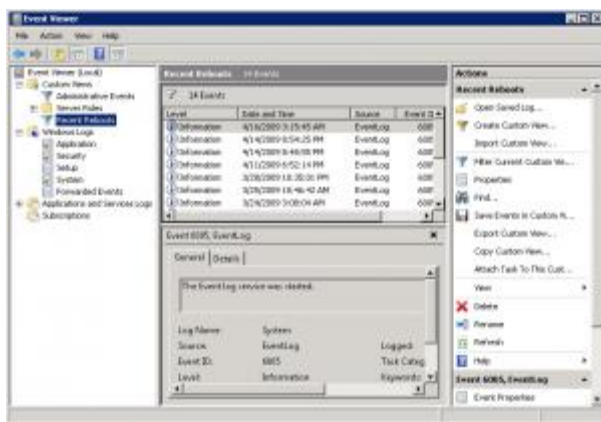
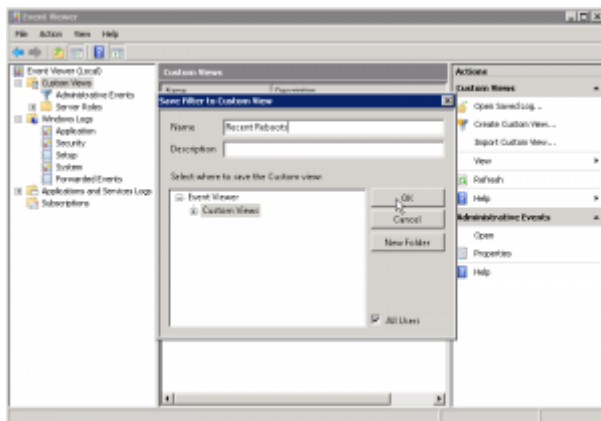
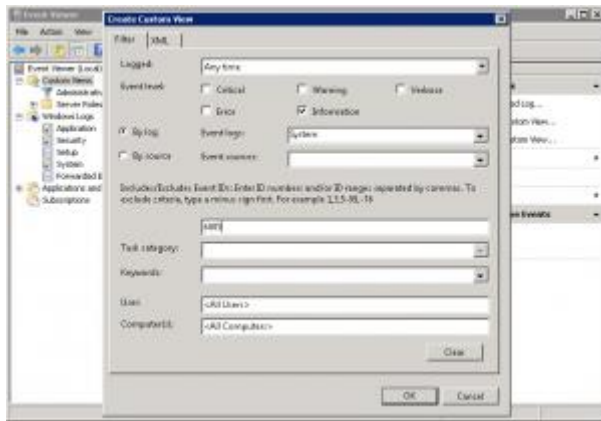
3. In the middle pane, click on the Information event type, and scroll down till you see Event ID 6005. Double-click the 6005 Event ID, or right-click it and select View All Instances of This Event.

4. A list of all instances of the 6005 Event ID will be displayed. You can examine this list, look at the dates and times of each reboot event, and so on.



Note: You can also easily create a Custom View to find all 6005 events. Please read my [“Working with Filtering and Custom Views in the Vista Event Viewer”](#) article for more information.

I will use the System Log and 6005 Event ID as the parameters for the custom view, and bingo, we can see all recent reboots of the system.



Method #6: By Using WMI

I found this nice article by the Microsoft Scripting Guy (read article for more details). I've changed the original suggestion a bit to make it more readable. Copy the following text into a text file and save it with a VBS extension. When done, double click on the file to get the system's running time in minutes.

```
strComputer = "."

Set objWMIService = GetObject("winmgmts:\" & strComputer & "rootcimv2")
Set colOperatingSystems = objWMIService.ExecQuery _
    ("Select * From Win32_PerfFormattedData_PerfOS_System")

For Each objOS in colOperatingSystems
    intSystemUptime = Int(objOS.SystemUpTime / 60)
    strMessage = "System uptime is " & intSystemUptime & " minutes"
    msgBox strMessage, 0, "System Uptime"
Next
```

Hey, Scripting Guy! How Can I Determine the Uptime for a Server?:

<http://www.microsoft.com/technet/scriptcenter/resources/qanda/aug05/hey0802.mspix>

Method #7: By Using PowerShell

I found this interesting article by the Microsoft Scripting guys.

The logic and explanations are too long for me to include here. Read it for more information.

Use PowerShell and WMI to calculate server uptime:

One long string !!

```
Get-WmiObject win32_operatingsystem |  
select csname,@{LABEL='LastBootUpTime';EXPRESSION={$_.ConverttoDateTime($_.lastbootuptime)}}
```

Or:

```
PS C:\> $wmi = Get-WmiObject -Class Win32_OperatingSystem
```

```
PS C:\> $wmi.LocalDateTime - $wmi.LastBootUpTime
```

Or:

```
PS C:\> $wmi = Get-WmiObject -Class Win32_OperatingSystem  
PS C:\> $wmi.ConvertToDateTime($wmi.LocalDateTime) -  
$wmi.ConvertToDateTime($wmi.LastBootUpTime)
```

<http://technet.microsoft.com/en-us/magazine/2008.12.heyscriptingguy.aspx?pr=blog>