

# 0 - Task Manager

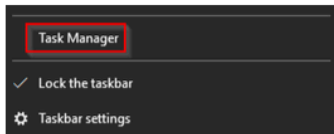
---

## Task 2 Task Manager

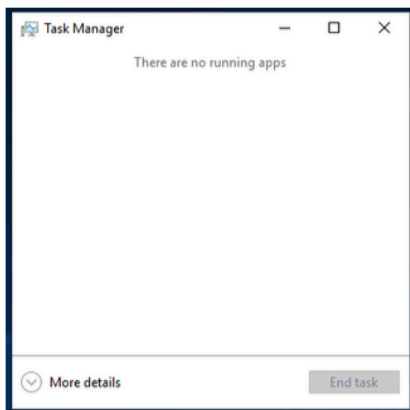
**Task Manager** is a built-in GUI-based Windows utility that allows users to see what is running on the Windows system. It also provides information on resource usage, such as how much each process utilizes CPU and memory. When a program is not responding, Task Manager is used to end (kill) the process.

We'll give a brief overview if you're unfamiliar with Task Manager.

To open Task Manager, right-click the Taskbar. When the new window appears, select **Task Manager** (as shown below).



If you don't have any explicitly opened apps, you should see the same message as shown below.



Weird. Not seeing much, eh? Within a Windows system, many processes are running. Click on **More details**.

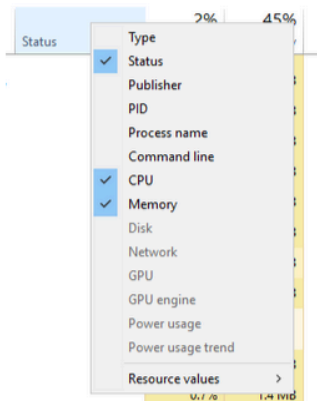
Name	Status	1% CPU	44% Memory
<b>Apps (1)</b>			
Task Manager		0%	11.8 MB
<b>Background processes (19)</b>			
Antimalware Service Executable		0%	39.6 MB

Ok, now we're getting somewhere. Notice the five tabs within Task Manager. By default, the current tab is **Processes**.

**Note:** If you're running Task Manager on your Windows machine, you might see additional tabs.

As shown above, you may notice the processes are categorized as follows: **Apps** and **Background processes**. Another category that is not visible in the above image is **Windows processes**.

The columns are very minimal. The columns **Name**, **Status**, **CPU**, and **Memory** are the only ones visible. To view more columns, right-click on any column header to open more options.



Name	Type	Publisher	PID	Process name	Command line	1% CPU	44% Memory
<b>Apps (1)</b>							
Task Manager	App	Microsoft Corporation	1100	Taskmgr.exe	"C:\Windows\system32\taskmgr.exe" /4	0%	14.5 MB

The view looks a little better. Let's briefly go over each column (excluding **Name**, of course):

- **Type** - Each process falls into 1 of 3 categories (Apps, Background process, or Windows process).
- **Publisher** - Think of this column as the name of the author of the program/file.
- **PID** - This is known as the process identifier number. Windows assigns a unique process identifier each time a program starts. If the same program has multiple running processes, each will have its unique process identifier (PID).
- **Process name** - This is the file name of the process. In the above image, the file name for Task Manager is Taskmgr.exe.
- **Command line** - The full command used to launch the process.
- **CPU** - The amount of CPU (processing power) the process uses.
- **Memory** - The amount of physical working memory utilized by the process.

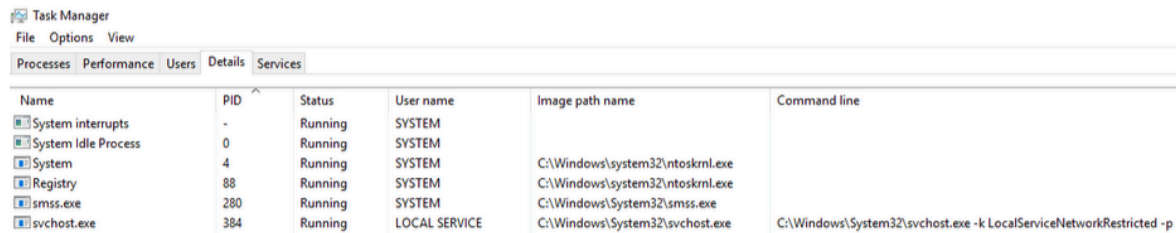
Task Manager is a utility you should be comfortable using, whether you're troubleshooting or performing analysis on the endpoint.

Let's move to the **Details** tab. This view provides some core processes that will be discussed in this room. Sort the **PID** column so that the PIDs are in ascending order.

Name	PID	Status	User name
System interrupts	-	Running	SYSTEM
System Idle Process	0	Running	SYSTEM
System	4	Running	SYSTEM
Registry	88	Running	SYSTEM
smss.exe	280	Running	SYSTEM
svchost.exe	384	Running	LOCAL SERVICE
csrss.exe	392	Running	SYSTEM
svchost.exe	416	Running	LOCAL SERVICE
wininit.exe	496	Running	SYSTEM
csrss.exe	512	Running	SYSTEM
winlogon.exe	592	Running	SYSTEM

Add some additional columns to see more information about these processes. Good columns to add are **Image path name** and **Command line**.

These two columns can quickly alert an analyst of any outliers with a given process. In the below image, PID 384 is paired with a process named svchost.exe, a Windows process, but if the Image path name or Command line is not what it's expected to be, then we can perform a deeper analysis of this process.



Name	PID	Status	User name	Image path name	Command line
System interrupts	-	Running	SYSTEM		
System Idle Process	0	Running	SYSTEM		
System	4	Running	SYSTEM	C:\Windows\system32\ntoskrnl.exe	
Registry	88	Running	SYSTEM	C:\Windows\system32\ntoskrnl.exe	
smss.exe	280	Running	SYSTEM	C:\Windows\System32\smss.exe	
svchost.exe	384	Running	LOCAL SERVICE	C:\Windows\System32\svchost.exe	C:\Windows\System32\svchost.exe -k LocalServiceNetworkRestricted -p

Of course, you can add as many columns as you wish, but adding the columns that would be pertinent to your current task is recommended.

Task Manager is a powerful built-in Windows utility but lacks certain important information when analyzing processes, such as **parent process information**. It is another key column when identifying outliers. Back to svchost.exe, if the parent process for PID 384 is not services.exe, this will warrant further analysis.

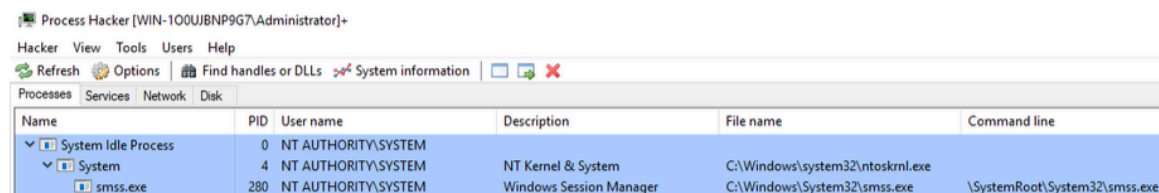
To further prove this point, where is services.exe?



svchost.exe	628	Running
services.exe	632	Running
lsass.exe	640	Running
svchost.exe	748	Running

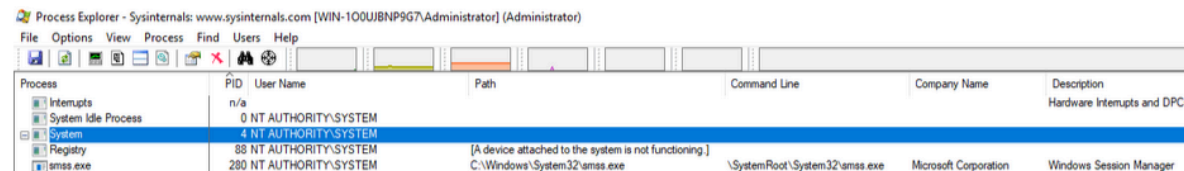
Based on the above image, the PID for services.exe is 632. But wait, one of the svchost.exe processes has a PID of 384. How did svchost.exe start before services.exe? Well, it didn't. Task Manager doesn't show a Parent-Child process view. That is where other utilities, such as **Process Hacker** and **Process Explorer**, come to the rescue.

### Process Hacker



Name	PID	User name	Description	File name	Command line
System Idle Process	0	NT AUTHORITY\SYSTEM			
System	4	NT AUTHORITY\SYSTEM	NT Kernel & System	C:\Windows\system32\ntoskrnl.exe	
smss.exe	280	NT AUTHORITY\SYSTEM	Windows Session Manager	C:\Windows\System32\smss.exe	\SystemRoot\System32\smss.exe

### Process Explorer



Process	PID	User Name	Path	Command Line	Company Name	Description
Interrupts	n/a					Hardware Interrupts and DPCs
System Idle Process	0	NT AUTHORITY\SYSTEM				
System	4	NT AUTHORITY\SYSTEM				
Registry	88	NT AUTHORITY\SYSTEM	[A device attached to the system is not functioning.]			
smss.exe	280	NT AUTHORITY\SYSTEM	C:\Windows\System32\smss.exe	\SystemRoot\System32\smss.exe	Microsoft Corporation	Windows Session Manager

Moving forward, we'll use **Process Hacker** and **Process Explorer** instead of Task Manager to obtain information about each Windows process.

As always, it's encouraged that you inspect and familiarize yourself with all information available within Task Manager. It's a built-in utility that is available in every Windows system. You might find yourself in a situation where you can't bring your tools to the fight and rely on the tools native to the system.

Aside from Task Manager, it would be best if you also familiarize yourself with the command-line equivalent of obtaining information about the running processes on a Windows system:

**tasklist**, **Get-Process** or **ps** (PowerShell), and **wmic**.