

Kill Windows a process with Tskill and Taskkill

<https://4sysops.com/archives/kill-windows-process-with-tskill-and-taskkill/>

In my previous post, [Terminate Windows processes with PsKill](#), I explained how to use PsKill to kill Windows processes on local or remote systems. As mentioned in the post, PsKill is a rather old tool with just a few options. Today, I will explain how to use two built-in tools: Tskill and Taskkill.

Contents

1. Kill a Windows process with Tskill
2. Kill a Windows process with Taskkill
3. Final words

Both Tskill and Taskkill have been available since Windows XP as built-in tools. As with PsKill, they can both kill a Windows process locally or remotely. Administrative privileges are required if you want to terminate a process from another user or with a higher security context. Let's have a closer look at how you can kill Windows processes with these tools.

Kill a Windows process with Tskill

Tskill is a simple tool with only a few options. It can terminate a process based on process ID (PID) or process name. Wildcard characters are accepted as well. Tskill can be used locally or remotely.



```
Windows PowerShell
PS C:\Users\...> tskill.exe
Invalid parameter(s)
Ends a process.

TSKILL processid | processname [/SERVER:servername] [/ID:sessionid | /A] [/V]

processid      Process ID for the process to be terminated.
processname    Process name to be terminated.
/SERVER:servername Server containing processID (default is current).
                /ID or /A must be specified when using processname
                and /SERVER
/ID:sessionid  End process running under the specified session.
/A            End process running under ALL sessions.
/V            Display information about actions being performed.

PS C:\Users\...>
```

Tskill command line options

The basic syntax is as follows:

```
tskill 5564
tskill mspaint
tskill mspa*
```

The Tskill [documentation](#) on the Microsoft website tells you that the tool ends a process running in a session on a Remote Desktop Services (RDS) host server. This can be useful when terminating processes by wildcard or from remote servers.

You can specify the session in which you want to kill the remote process. Use **/ID:sessionid** to specify a session or **/A** to kill the Windows process under all sessions. These options are only required when you kill a process by its name or a wildcard, as multiple users may run the same process name on an RDS server. Terminating a remote process by its PID does not require these options, as the PID is always unique. By default, Tskill does not produce any output. This can be changed using the **/V** switch. I recommend using this switch for such operations so that you can see the output.

```
Windows PowerShell
PS C:\Users\ > taskkill.exe mspaint /SERVER: /ID:2 /V
End Process (2360)
PS C:\Users\ > taskkill.exe mspaint /SERVER: /A /V
End Process (76)
PS C:\Users\ >
```

Terminating remote processes with Tskill

Kill a Windows process with Taskkill

Compared to Tskill, Taskkill has many more features, as can be seen in the help message. It allows you to kill a Windows process from a local or remote system, use different credentials, filter to select a set of tasks, or terminate a process tree.

```
Windows PowerShell
PS C:\Users\ > taskkill.exe /?
TASKKILL [/S system [/U username [/P [password]]]
{ [/FI filter] [/PID processid | /IM imagename] } [/T] [/F]
Description:
  This tool is used to terminate tasks by process id (PID) or image name.
Parameter List:
  /S system           Specifies the remote system to connect to.
  /U [domain\]user    Specifies the user context under which the
                      command should execute.
  /P [password]       Specifies the password for the given user
                      context. Prompts for input if omitted.
  /FI filter          Applies a filter to select a set of tasks.
                      Allows "*" to be used. ex. imagename eq acme*
  /PID processid      Specifies the PID of the process to be terminated.
                      Use TaskList to get the PID.
  /IM imagename       Specifies the image name of the process
                      to be terminated. Wildcard '*' can be used
                      to specify all tasks or image names.
  /T                  Terminates the specified process and any
                      child processes which were started by it.
  /F                  Specifies to forcefully terminate the process(es).
  /?                  Displays this help message.
Filters:
  Filter Name  Valid Operators  Valid Value(s)
  -----
  STATUS       eq, ne           RUNNING |
              NOT RESPONDING | UNKNOWN
  IMAGENAME    eq, ne           Image name
  PID          eq, ne, gt, lt, ge, le  PID value
  SESSION      eq, ne, gt, lt, ge, le  Session number.
  CPUTIME      eq, ne, gt, lt, ge, le  CPU time in the format
                      of hh:mm:ss.
                      hh - hours,
                      mm - minutes, ss - seconds
  MEMUSAGE     eq, ne, gt, lt, ge, le  Memory usage in KB
  USERNAME     eq, ne           User name in [domain\]user
                      format
  MODULES      eq, ne           DLL name
  SERVICES     eq, ne           Service name
  WINDOWTITLE  eq, ne           Window title
NOTE
----
1) Wildcard '*' for /IM switch is accepted only when a filter is applied.
2) Termination of remote processes will always be done forcefully (/F).
3) "WINDOWTITLE" and "STATUS" filters are not considered when a remote
   machine is specified.
```

Taskkill command line options

The basic syntax is as follows:

```
taskkill /IM mspaint.exe
taskkill /IM mspa*
taskkill /PID 1258
taskkill /PID 5589 /T
```

Note that the .exe suffix is required if you don't use a wildcard. Typing only mspaint will return "ERROR: The process mspaint not found." The help message also says that wildcards in the /IM option can only be used together with the filter (/FI option), which is not true. The /T option is used to kill the entire process tree.

Taskkill can also kill a Windows process softly. It sends a kill signal to the application, allowing it to save its data and end properly. For example, if you use Taskkill to terminate a Word process that has unsaved changes, Word will ask you if you want to save the data. To forcefully kill a Windows process, use the **/F** option. Note that terminating a process on a remote system is always done forcefully.

Killing a process on a remote system requires administrative rights on the target system. Taskkill allows you to specify alternate credentials for such actions using the **/U** and **/P** options. If you don't add the **/P** option, you will be prompted for the password. The syntax is as follows:

```
taskkill /S myserver /IM mspaint.exe
taskkill /S myserver /U LAB\admin /P Passw0rd /PID 1234
```

Taskkill also allows you to use filters to specify a set of processes to be terminated. For example, you may want to kill all processes that have higher memory usage than 100 MB. Or you may want to kill all processes running under a specific user. This is done using the following syntax:

```
taskkill /F /FI "MEMUSAGE gt 102400"
taskkill /F /FI "USERNAME eq LAB\Admin"
```

You can also combine both conditions and kill all Windows processes for the Admin user that use more than 100 MB of memory. Simply specify the **/FI** option twice.

```
taskkill /F /FI "MEMUSAGE gt 102400"/FI "USERNAME eq LAB\Admin"
```

Use the help message to see all filtering options. Note that using filters incorrectly may result in the termination of critical system processes or processes you don't want terminated.

Subscribe to 4sysops newsletter!

```
Filters:
Filter Name      Valid Operators      Valid Value(s)
-----
STATUS           eq, ne               RUNNING |
                  NOT RESPONDING | UNKNOWN
IMAGENAME        eq, ne               Image name
PID              eq, ne, gt, lt, ge, le PID value
SESSION          eq, ne, gt, lt, ge, le Session number
CPU TIME         eq, ne, gt, lt, ge, le CPU time in the format
                  of hh:mm:ss,
                  hh - hours,
                  mm - minutes, ss - seconds
MEMUSAGE         eq, ne, gt, lt, ge, le Memory usage in KB
USERNAME         eq, ne               User name in [domain]user
                  format
MODULES          eq, ne               DLL name
SERVICES         eq, ne               Service name
WINDOWTITLE      eq, ne               Window title

NOTE
----
1) Wildcard '*' for /IM switch is accepted only when a filter is applied.
2) Termination of remote processes will always be done forcefully (/F).
3) "WINDOWTITLE" and "STATUS" filters are not considered when a remote
   machine is specified.

Examples:
TASKKILL /IM notepad.exe
TASKKILL /PID 1230 /PID 1241 /PID 1253 /T
TASKKILL /F /IM cmd.exe /T
TASKKILL /F /FI "PID ge 1000" /FI "WINDOWTITLE ne untitled*"
TASKKILL /F /FI "USERNAME eq NT AUTHORITY\SYSTEM" /IM notepad.exe
TASKKILL /S system /U domain\username /FI "USERNAME ne NT*" /IM *
TASKKILL /S system /U username /P password /FI "IMAGENAME eq note*"
```

Taskkill filtering options

Final words

In this post, you have learned how to use Tskill and Taskkill to kill a Windows process on local or remote systems. As you have seen, both tools offer more features than Sysinternals PsKill. In addition, they are built into Windows, so no download is required. As always, be careful with the process you are killing. Inappropriate actions might lead to data loss or system crash.