

PROCESS OPERATOR (SERVER EDITION)

1.0.1

DESCRIPTION

The main purpose of **Process Operator** (Server Edition) is to **enhance the responsiveness** of Windows servers during periods of high CPU loads. You will also notice **increased agility and stability** of the server.

Process Operator is a Windows service. It adjusts priorities at which applications get CPU times. As a result the server becomes much more responsive. The priority adjustment process is fully automated, but server administrators can make appropriate settings to fine-tune the Process Operator for their specific needs.

The Server Edition of Process Operator is made specially for Windows servers, not for home PC's. It is designed for Windows 2008 (x86, x64) and Windows 2003 servers and was tested on multiple web servers, mail and DNS servers, and SQL ones.

If some server process (application or service) "eats" too much CPU, the Process Operator sets a lower priority to that process. The process won't be allowed to use 100% of CPU and other applications will be running fine. The server does not stall. All processes get their CPU time slices. The server becomes more stable and agile. If the punished process uses less CPU for certain time, Process Operator restores its priority back to its original level.

Often when a server has about 100% CPU load it becomes unresponsive and even does not allow the use of Terminal Services, you cannot log in remotely to fix the issues. The Process Operator prevents such cases too.

SYSTEM REQUIREMENTS

- Any of the following operating systems: **Windows 2003** (x86 or x64), **Windows 2008** (x86 or x64).
- Microsoft **.NET Framework 3.5** or above (The Process Operator (during its setup) will try to download and install this Framework. You can also download it at Windows Update web site <http://windowsupdate.microsoft.com>).

TRIAL AND REGISTRATION

The evaluation version of Process Operator (Server Edition) works for **around 30 days**.

To register the software please place your order at www.HostsTools.com and a license key will be sent to you. Please create **key.txt** file in Process Operator folder (C:\Program Files\HostsTools.com\Process Operator) and copy/paste the license key to this file.

INSTALLATION

1. Install Microsoft .NET Framework 3.5 (or later) if you do not have it installed on your server yet. **You can skip this step** and allow Process Operator to download the Framework itself.
2. Download Process Operator installation package from <http://www.HostsTools.com> , unzip it and run **setup.exe** file. Follow the installation wizard.
3. Now you have a new Windows service named **ProcessOperator** (You can see it at 'Start – Programs – Administrative Tools – Services' or at 'Services' tab of Windows Task Manager). **Do not start it yet**, you need to configure it first.
4. Open Process Operator folder (*C:\Program Files\HostsTools.com\Process Operator*) and edit **config.ini** file. You can edit it with **Notepad** or other basic text editor (which does not add overhead information to text files). Please read **Configuration** section of this guide below for more information.
5. If you have a license key, open Process Operator folder (*C:\Program Files\HostsTools.com\Process Operator*), create **key.txt** file in this folder and copy/paste the license key to this file.

If no key found, the program works for around 30 days and then stops working. To register the software please place your order at www.HostsTools.com and the license key will be sent to you.

6. Go to 'Start – (Programs) – Administrative Tools – Services' (or open 'Services' tab of Windows Task Manager), select **ProcessOperator** service and start or restart it.

CONFIGURATION

The Process Operator configuration file **config.ini** is placed in *C:\Program Files\HostsTools.com\Process Operator* folder. Please edit it before you start using Process Operator on your server.

It is safe to use Process Operator with its default config.ini file. But we recommend doing some settings relevant to your server purpose.

You must restart ProcessOperator service for changes to take effect (after config.ini editing).

The config.ini file consists of several sections:

[Parameters]

This section contains the only parameter **CPU_Load_Threshold** currently. The Process Operator starts changing process priorities when CPU load is high. Use CPU_Load_Threshold parameter to specify the CPU load (in %) . If you specify

CPU_Load_Threshold=80

it means: "Process Operator should start changing applications priorities when CPU load is 80% or more. If CPU load is lower than 80% the Process Operator should consider it as low CPU load and start setting initial priorities of punished applications after some period."

However CPU_Load_Threshold shows CPU load in percents you must not specify '%' symbol!

Right: CPU_Load_Threshold=75

Wrong: CPU_Load_Threshold=75%

[DoNotChangePriority]

This section contains a list of processes Process Operator is not allowed to change the priority of.

For example, if you specify w3wp.exe in this list, ProcessOperator will never change the priority of w3wp.exe even if this process is taking 100% of CPU time!

You might want to add some system processes to this list. And we also recommend to have ProcessOperator.exe in this list.

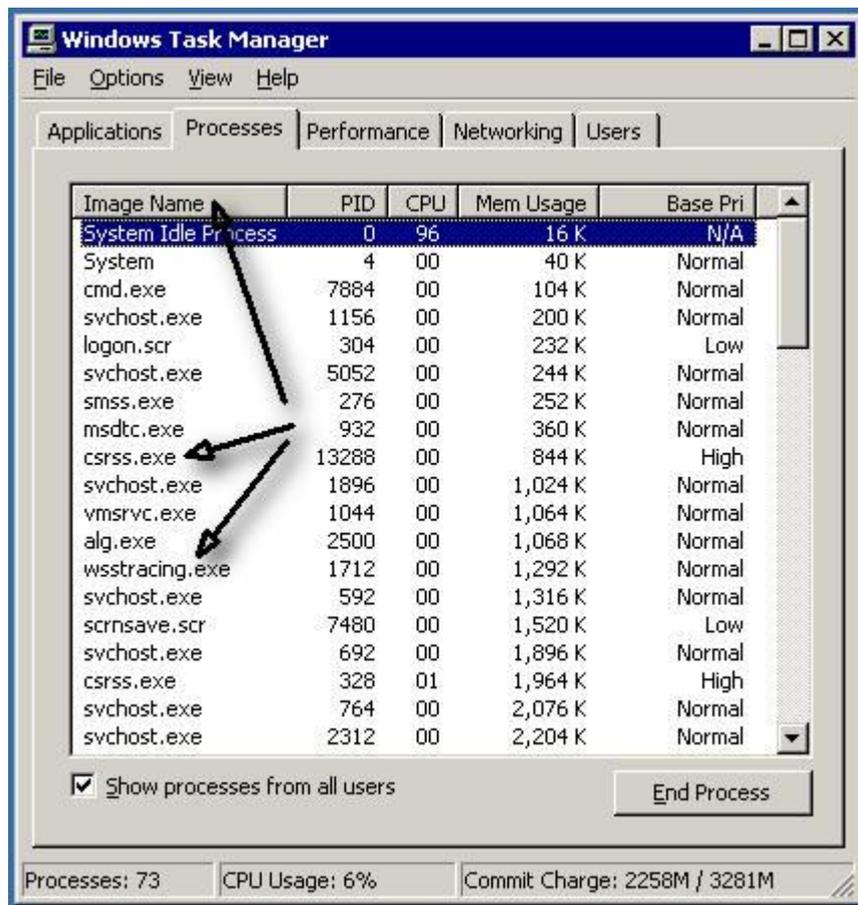
It is not possible to make some universal list. It depends on your server OS, configuration and purpose.

A good start is:

```
[DoNotChangePriority]
ProcessOperator.exe
System
services.exe
taskmgr.exe
winlogon.exe
csrss.exe
lsass.exe
```

You can find process names in Windows Task Manager. Just open its 'Processes' tab and look at 'Image Name' column.

*The name must be specified exactly as you see it at the 'Image Name' column (including '.exe' or other suffix, if any). However the name is **not case-sensitive**.*



[DoNotSetPriorityLowerThan_NORMAL]

Specify process names in this section if you **do not** want Process Operator to lower their priority to any level below **Normal**.

As you know, there are 6 priority levels: **Realtime**, **High**, **Above Normal**, **Normal**, **Below Normal**, **Low** (Idle).

If some process is present in *[DoNotSetPriorityLowerThan_NORMAL]* list, Process Operator will never change its priority to **Below Normal** or **Low**, even if the process is taking 100% of CPU time.

Sample:

[DoNotSetPriorityLowerThan_BELOW_NORMAL]

dns.exe

inetinfo.exe

[DoNotSetPriorityLowerThan_BELOW_NORMAL]

This list is similar to *[DoNotSetPriorityLowerThan_NORMAL]* one (described above), but the lowest allowed priority level is **Below Normal**.

Sample:

```
[DoNotSetPriorityLowerThan_BELOW_NORMAL]
w3wp.exe
mysqld-nt.exe
php-cgi.exe
owstimer.exe
```

PROCESS IDENTIFICATION

In addition to process names the Process Operator differentiates the processes by their unique identifier. You can see the process identifiers in Windows Task Manager ('PID' or 'Process Identifier' column).

While multiple processes can have the same name, they never have the same identifier.

If some particular process is causing high CPU usage, the Process Operator descends priority of this single process only, not of all processes having the same name!

For example, IIS (Internet Information Services) creates a separate process for each application pool. A web server potentially can have hundreds of individual instances of w3wp.exe process running at any given time. Looking for the process name is pointless in such a case, but the process identifiers allow to differentiate between the different worker processes servicing the application pools. If some particular IIS application pool is causing high CPU usage, the Process Operator descends priority of this single process only, not of all w3wp.exe processes!

LOG FILES

You can find Process Operator log files in `C:\Program Files\HostsTools.com\Process Operator\logs` folder.

There are 2 types of log files:

1. Files named `"date_here".log` (e.g. `11.07.2010.log`):
Process Operator writes some of its actions (most important ones) and error messages (if any) to these log files.
2. Files named `"date_here"-CPU-usage.log` (e.g. `11.07.2010-CPU-usage.log`):
These logs contain information about total CPU usage and a list of 3 processes causing the CPU load. Process Operator writes CPU usage data **once a minute**. So you can see the full history of CPU usage and "heavy" processes.

The log format allows importing it to MS Excel, sorting and building graphs.

If the Process Operator writes something about w3wp.exe process to log files (CPU usage, priority etc.), it also writes the name of IIS Application Pool running in that process. It allows to see what pool caused high CPU usage.

TROUBLESHOOTING

In case of any errors, please:

1. Make a screenshot of the error message window (if any).
2. Describe your steps and the case.
3. Find related error messages in Windows Events logs. It may be found in **Application** section of the logs. The source name is **ProcessOperator**.
4. Email the screenshot, description, the message from the Windows Events log to support@hoststools.com

Thank you.