



# IT Automation Omnishell



m:  
<https://pandorafms.com/manual/!775/>  
Permanent link:  
[https://pandorafms.com/manual/!775/en/documentation/pandorafms/management\\_and\\_operation/16\\_omnishell](https://pandorafms.com/manual/!775/en/documentation/pandorafms/management_and_operation/16_omnishell)  
2024/03/18 21:03



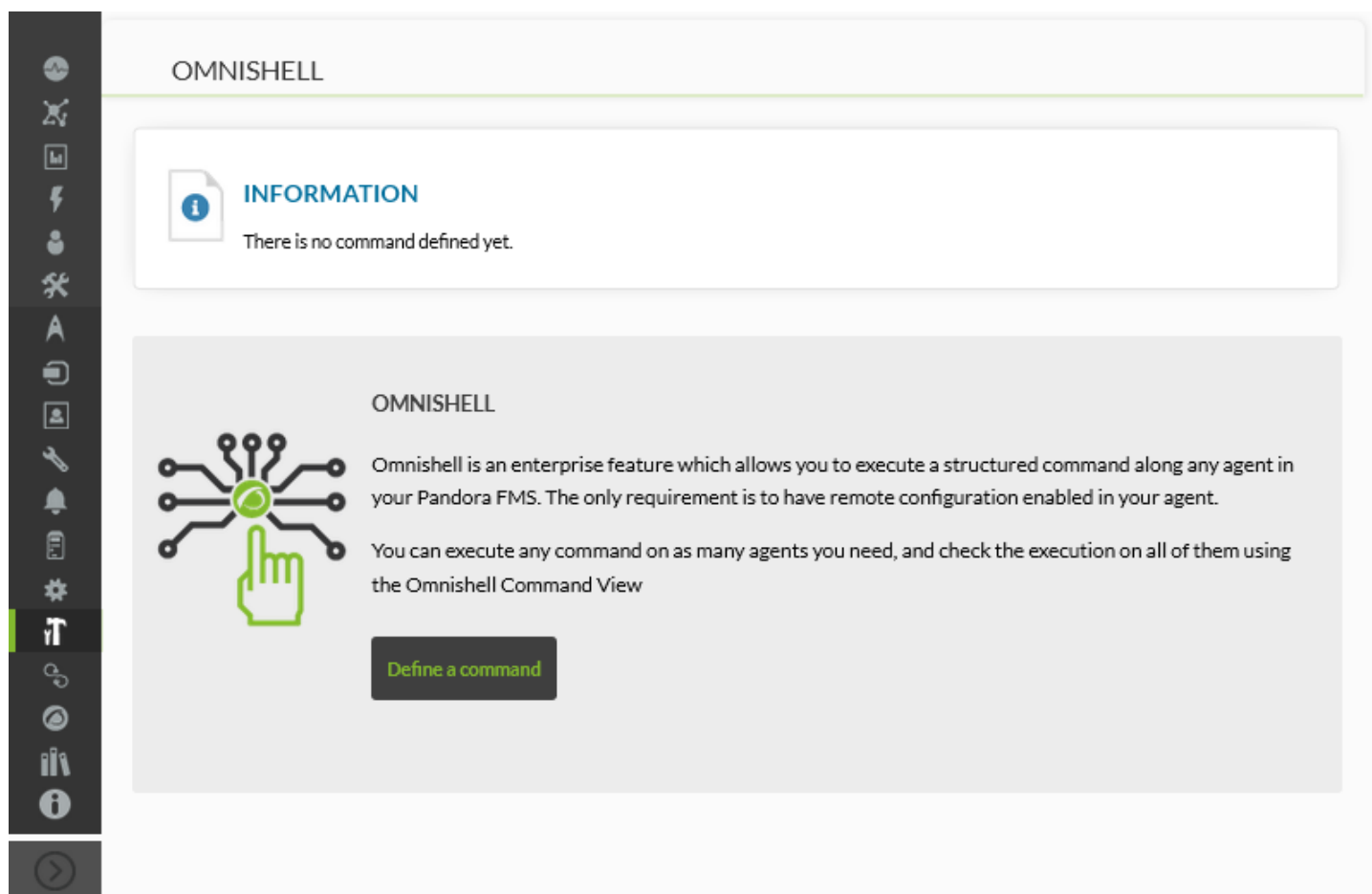


# IT Automation Omnishell

## Introduction to Omnishell

**E** Version NG 741 or higher.

Omnishell is a Pandora FMS Enterprise feature used for IT orchestration/automation. It is a fully native and Pandora FMS console and agent integrated console that allows defining commands or blocks of commands and selecting targets where they can be executed. Learn more about it in our video tutorial «[IT automation in Pandora FMS: Omnishell](#)».



OMNISHELL

**INFORMATION**  
There is no command defined yet.

**OMNISHELL**

Omnishell is an enterprise feature which allows you to execute a structured command along any agent in your Pandora FMS. The only requirement is to have remote configuration enabled in your agent.

You can execute any command on as many agents you need, and check the execution on all of them using the Omnishell Command View

[Define a command](#)

Omnishell is based on communication between the different components of Pandora FMS through [Tentacle](#), to safely coordinate executions and responses of the commands you wish to program.

You must have installed in Pandora FMS console the dependency `php-yaml` and in Software GNU/Linux agents, the dependency `perl-YAML-Tiny`.

## Additional Tentacle server settings

Only for version NG 742 or previous version.

- Tentacle Server needs a change in configuration to be able to redirect files with extension `.rcmd`.
- These files contain command definition, which each **Software agent** downloads and evaluates.
- Tentacle Server file filter definition must be updated through:
- Modifying the file `/etc/tentacle/tentacle_server.conf` :

```
# Filters (regexp:dir,regexp:dir...)
filters
.*\.conf:conf;.*\.md5:md5;.*\.zip:collections;.*\.lock:trans**;.*\.rcmd:commands
**
```

- Copy of the file `tentacle_server.conf.new`, which is distributed when installing Pandora FMS Server.
- `.rcmd` files are published in a new directory:

```
/var/spool/pandora/data_in/commands
```

- This folder must have permissions for users that manage the Pandora FMS server and the Apache that run the console:

```
chown pandora:apache -R /var/spool/pandora/data_in/commands
chmod g+s /var/spool/pandora/data_in/commands
```

- Note: A *hash* is used based in certain parameters of the command's definition.
- Ensure the installation of the Tentacle protocol, in **GNU/Linux®** or in **MS Windows®**, according to your case.

## Performance

An Omnishell task is defined by three elements:

### Precondition

Through one or several commands with each sequential execution that must be satisfying for all of them (`result errorlevel 0`).

### Commands

To be executed in the target software agent, in a sequential list.

### Postcondition

Through one or several commands with each sequential execution that must be satisfying for all of them (result error level 0).

If any of the commands defined in Precondition, Commands or Postconditions goes into interactive mode, blocking the execution flow, the task will be interrupted. The Software agent itself will destroy the process after the *timeout* configured for that task.

## Usage example

### Example in GNU Linux

An example shall be explained to better understand the working process of the new Omnishell tool.

Problem: The objective is configuring the auto-start of the HTTPD service on CentOS 6 systems in a massive way.

- What is the precondition? It is for the service to appear as installed, and install it in case it is not:

For example:

```
[ `rpm -qa | grep -i httpd | wc -l` -lt 1 ] || yum install -y httpd
```

This line will verify whether the service is installed and will install it in case it is not.

- What is the main goal? To start and configure the auto-start of the HTTPD service.

To that end, the following commands must be used:

```
/etc/init.d/httpd restart  
chkconfig httpd on
```

This will start the service on the remote machine and will configure the system to start it up.

- How to validate what has been done? Verify that the system has activated the service and it is listening.

```
[ `netstat -lptn | grep 80 | grep LISTEN | wc -l` -gt 0 ]
```

Once the action the agents must perform is clearly set, create the command in Omnishell:


Command definition 


**Name**


**Group**


**Time out**


**Retries**

**Preconditions** 

**Execute commands** 

**Postconditions** 

Next 

Go back 

After creating the command, assign the needed agents.

## Select targets

### Available agents

DESKTOP-1JQQE4J  
pandorafms  
Sample\_1  
Sample\_2  
Sample\_3  
Sample\_4  
Sample\_5  
Sample\_6  
Sample\_7  
Sample\_8



### Selected agents

None


Finish >

Go back ✕

The progress of the command execution can be seen in the selected agents:

## OMNISHELL


[> Filter](#)Show  entries

▲ Identifier	Group	Timestamp	Progress	Options	<input type="checkbox"/>
httpd service in CentOS 7		10 seconds	<div style="width: 100%; height: 10px; background-color: #ccc;"></div>	   	<input type="checkbox"/>

Showing 1 to 1 of 1 entries [Create command >](#)[Delete selected items !\[\]\(e8fb589d58dad1692debababa5e928b6\_img.jpg\)](#)[Refresh !\[\]\(f95dab70c751fda7d824b8b03650f7aa\_img.jpg\)](#)

## AGENT LIST

Show  entries

▲ Agent name	Status	Last contact	Options	<input type="checkbox"/>
pandorafms agent	<div style="width: 100%; height: 15px; background-color: #ccc;"></div>	4 minutes 01 seconds		<input type="checkbox"/>

Showing 1 to 1 of 1 entries [Delete selected items !\[\]\(4146d17f71dced09c6ad789cacceaa6d\_img.jpg\)](#)

Accessing the view, you will see the estimated time left for command execution:



Omnishell / Command viewer

## Detailed view: Start httpd in Centos

262s

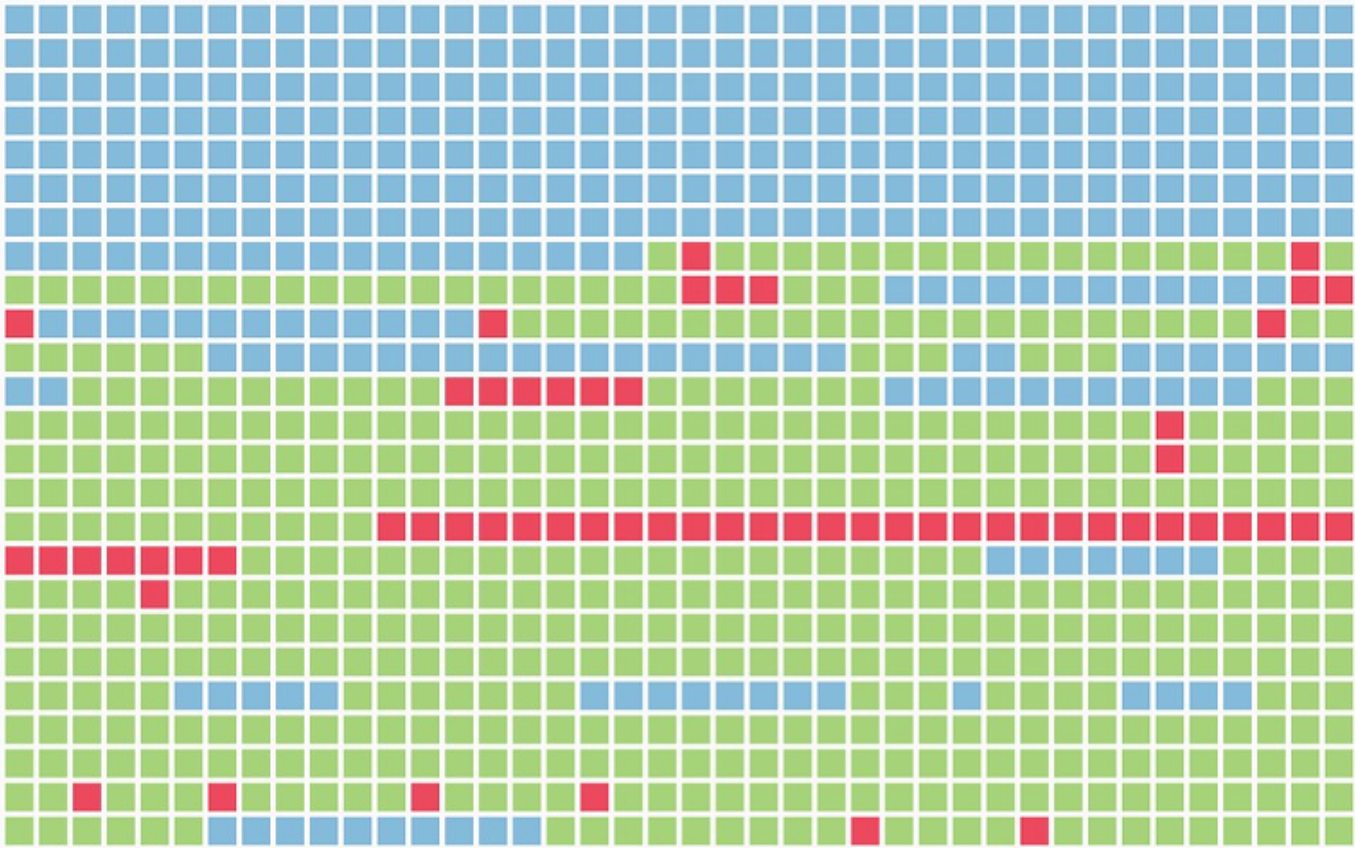
Refresh ↻

Go back ✕

When mass command execution is applied, the view will be as the following one from the example:

Omnishell / Command viewer

## Detailed view: Install HTTP



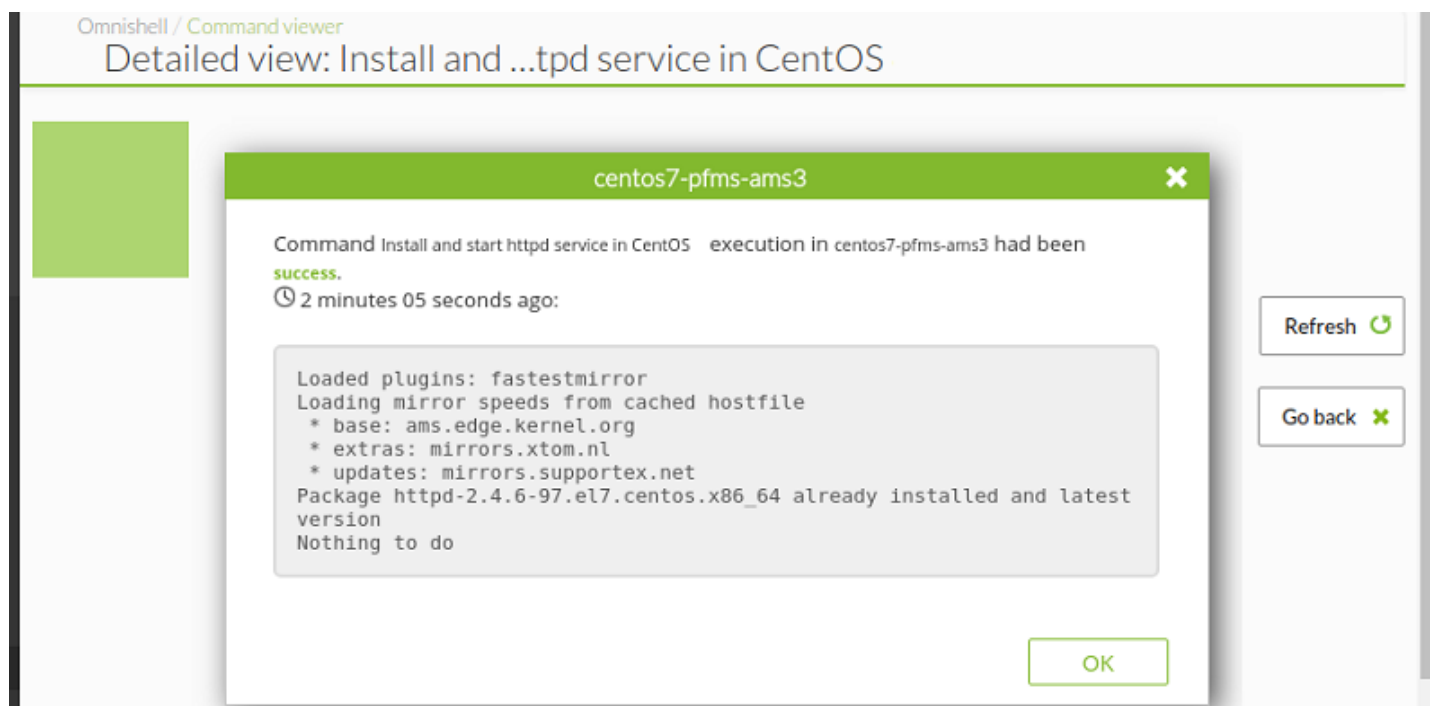
The grid consists of 20 rows and 100 columns of small squares. The colors represent different execution states: blue for successful, green for pending, and red for failed. The top 10 rows are almost entirely blue. The middle 10 rows show a mix of green and blue, with some red squares appearing. The bottom 10 rows are mostly green, with scattered red and blue squares.

Refresh ↻

Go back ✕

Once the agent's response is received, execution results will be displayed. By clicking on its box, a

modal can be obtained with extra information:



### Example in MS Windows

For this environment, Omnishell is still in process, so some of its features are still subject to changes or are not available yet. This is only temporarily and we apologize for that.

Regarding the configuration process in Pandora FMS console, repeat exactly the same steps carried out with [the previous example in GNU/Linux](#) and only the code to be executed will change.

Due to the simplicity of this example, only the precondition and the main task stages are used, respectively:

```
powershell.exe -command "Start-Sleep -Seconds 1"
```

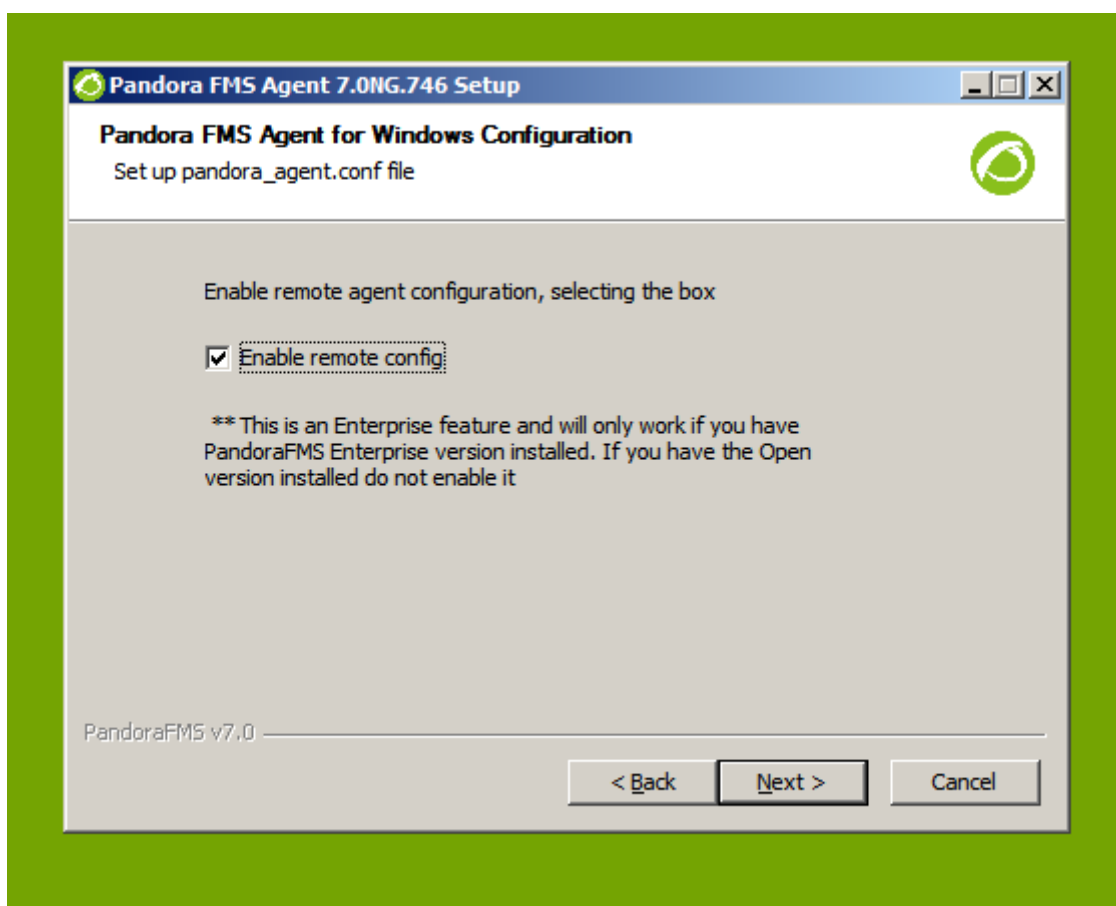
```
powershell.exe -command "$mode = Get-Executionpolicy ; if ( $mode -like
'restricted' ) {Write-Error -Message '0'}"
```

PowerShell is a new Windows *shell* that has been specifically conceived for system admins. This includes an interactive system symbol and a scripting environment that can be used together or separately. It comes installed with said operating system, although you may install PowerShell Core from its repository in [GitHub](#) to get the latest updated version. You may get more interesting information about PowerShell in [Pandora FMS blog](#).

In the precondition it is checked that PowerShell is installed and can be used, otherwise it will return an error and it will *not be able to be executed in the next step*.

In the main task, it is investigated about the possibilities of executing *scripts* written in PowerShell. If its properly executed you may add code at the checking stage to use one and carry out longer and more complex tasks. To learn more, check the documentation about [how to execute scripts in a module](#) or [how tu use a template to create a module that uses PowerShell](#).

During the Software agent installation check the option Enable remote config. If you already have that agent installed, modify the file %PROGRAMFILES%\pandora\_agent\pandora\_agent.conf with admin rights and add (or modify) the following line: remote\_config 1. In the group of the Pandora FMS program from the home menu there is a direct access to the script %PROGRAMFILES%\pandora\_agent\scripts\edit\_config\_file.bat to make things easy.



## Comments

- The agent will execute each order of each block.
- If the evaluation of each order is not 0 ( `error_level != 0` ), it will stop evaluating and the command will be considered as failed.
- The agent will take `retry` and `timeout` parameter values for the number of retries and the

maximum amount of execution time, accordingly.

- Regardless of the result, the agent will save an execution mark. Each command is executed only once.
- If you update the name of the command, it will cause the agent to execute it again, since the reference will change.
- You can assign as many commands as desired to the same agent.
- Agent command execution may affect the monitoring. Do not use a timeout higher than the agent interval.

[Go back to Pandora FMS documentation index](#)