

Tasks

Last Modified on 04.16.26

Get-LiquitTask

Synopsis

This command displays all the tasks defined in the zone or server. Alternatively, you can use this command to select a specific task.

Syntax

```
Get-LiquitTask
[[-ID] <guid[>]
[-Search <string>]
[-Name <string>]
[-State {Queued | InProgress | Success |
Failed | Cancelled | Suspended | Continue}]
[-LiquitContext <LiquitContext>]
[<CommonParameters>]
```

```
Get-LiquitTask
[-Server] <ServerPrimary>
[[-ID] <guid[>]
[-Search <string>]
[-Name <string>]
[-State {Queued |
InProgress | Success | Failed | Cancelled | Suspended | Continue}]
[-LiquitContext <LiquitContext>]
[<CommonParameters>]
```

```
Get-LiquitTask
[-ScheduledTask] <ScheduledTask>
[[-ID] <guid[>]
[-Search <string>]
[-Name <string>]
[-State {Queued | InProgress | Success | Failed | Cancelled | Suspended | Continue}]
[-LiquitContext <LiquitContext>]
[<CommonParameters>]
```

Examples

This command generates a list of all the failed tasks in the current system:

```
$taskList = Get-LiquitTask -State Failed
```

The following script retrieves all the tasks running on the specified server:

```
$server = Get-LiquitServer -id 00000000-0000-0000-0000-00000000
Get-LiquitTask -Server $server
```



You cannot get a list of tasks running on a satellite server by this method.

Parameters

Name	Value	Description	Required	Default value
ID	<guid[]>	The GUID belonging to the task.	No	
Search	<string>	This parameter can be used to search for a scheduled task.	No	
Name	<string>	The name of the scheduled task.	No	
State	{Queued InProgress Success Failed Cancelled Suspended Continue}	Select tasks by their state. You can for example select only the failed scheduled tasks.	No	
Server	<guid[]>	The GUID belonging to an Application Workspace Server	No	
ScheduledTask	<ScheduledTask>	Select all the tasks spawned by a specific scheduled task.	No	
LiquitContext	<LiquitContext>	Determines the selected zone.	No	Default

Wait-LiquitTask

Synopsis

This command waits until a task is finished.

Syntax

```
Wait-LiquitTask  
[-ID] <guid>  
[-Timeout] <int>  
[-LiquitContext <LiquitContext>]  
[<CommonParameters>]
```

```
Wait-LiquitTask  
[-Task] <Task>  
[-Timeout] <int>  
[-LiquitContext <LiquitContext>]  
[<CommonParameters>]
```

Examples

The following script waits until a specific install is complete:

```
$Zone = Get-LiquitZone -ID 00000000-0000-0000-0000-000000000000  
$Zone | Install-LiquitZoneLicense -Key "00000-00000-00000-00000-0000" | Wait-LiquitTask
```

The following script waits until a specific install is complete based on a task:

Recast

```
$Zone = Get-LiquitZone -ID 2deef6c0-c302-201b-003b-35b7ad2faa1b  
$Task = Install-LiquitZoneLicense -Zone $Zone -Key "00000-00000-00000-00000-0000"  
Wait-LiquitTask -Task $Task
```

This command waits until a specific install is complete based on an ID:

```
Get-LiquitZone -ID 00000000-0000-0000-0000-000000000000
```
