Recast

Getting Started with the PowerShell Module

Last Modified on 2026-10-27

To run cmdlets against Application Workspace and perform administration or monitoring tasks, you must first establish a connection to your Application Workspace System. Application Workspace connectivity is stored in a Context object, which makes it possible to manage multiple systems within a single script.

> **Multifactor Authentication**
> Application Workspace does not currently support MFA via PowerShell.

Download the Application Workspace PowerShell Module

You can find a download link for the PowerShell Module at Application Workspace Downloads. Look for it at the bottom of the Product table.

Product	Version		
Application Workspace Server	4.5.4374.7448	Documentation	Download
Application Workspace Agent (Win x64)	4.5.4374.7448	Documentation	Download
Application Workspace Agent (Win x86)	4.5.4374.7448	Documentation	Download
Application Workspace Agent (Windows)	4.5.4374.7448	Documentation	Download
Application Workspace Agent (macOS)	4.5.4374.7448	Documentation	Download
Application Workspace Satellite	4.5.4374.7448	Documentation	Download
Application Workspace PowerShell Module	4.5.4374.7448	Documentation	Download

Use the Application Workspace PowerShell Module

To use the PowerShell module:

1. Start the Application Workspace PowerShell Module using the shortcut created in Windows Start Menu after installation, or by manually loading the PowerShell module.

2. Create a default context object by running Connect-LiquitWorkspace with the Application Workspace URI and valid credentials.

   The username should use the following syntax: `Identity source name\username`

```
$liquitContext = Connect-LiquitWorkspace -URI https://workspace.recastsoftware.com -Credential (Get-Credential)
```

After establishing a connection, all available cmdlet commands will use this default context, unless explicitly specified by the `LiquitContext` parameter on every Application Workspace cmdlet.

If there are spaces in the identity source name, Get-Credential will fail, as username validation doesn't allow spaces. As a workaround, you can use the following code to use spaces within the identity source name:

```
$username = "Microsoft Entra ID\username@recastsoftware.com"
$password = Read-Host "Enter Password" -AsSecureString
$credentials = New-Object -TypeName System.Management.Automation.PSCredential -ArgumentList $username, $password

$liquitContext = Connect-LiquitWorkspace -URI "https://workspace.recastsoftware.com" -Credential $credentials
```

Recast