

PowerShell Integration with Openprovider API

If you are interested in accessing and using the Openprovider API via PowerShell, you can find useful information [here](#).

Authenticating to the Openprovider API via PowerShell

The Openprovider API uses Bearer authentication. You authenticate once with your username and password and are given a bearer code. You then use the bearer code for the rest of your API calls.

Here is an example of what this looks like in PowerShell:

```
function Get-BearereCreds
{
    $EndPoint = "https://api.openprovider.eu/v1beta/auth/login"

    ##Configure Payload
    $Body = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
    $Body.Add("ip","0.0.0.0")
    $Body.Add("username","UsernameHere")
    $Body.Add("password","PasswordPlease")
    $Body = $Body | ConvertTo-Json

    $Bearer = (Invoke-RestMethod -Method Post -Uri $EndPoint -body $Body -ContentType
    'application/json' ).data
    return $Bearer
}

$Bearer = Get-BearereCreds
```

Keep in mind that the bearer code expires after a set time. When you want to perform scheduled tasks or tasks that take a long time, you will need to periodically request a new bearer code.

Checking the status of domain registration via PowerShell

Here is an example of how to request the status of a domain registration:

```
function Get-BearereCreds
{
$EndPoint = "https://api.openprovider.eu/v1beta/auth/login"

##Configure Payload
$Body = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$Body.Add("ip","0.0.0.0")
$Body.Add("username","UsernameHere")
$Body.Add("password","PasswordHerePlease")
$Body = $Body | ConvertTo-Json

$Bearer = (Invoke-RestMethod -Method Post -Uri $EndPoint -body $Body -ContentType 'application/json')
return $Bearer
}

function check-domain
{
Param(
[parameter(Mandatory=$true)]
[String]$Domains
)

$uri = "https://api.openprovider.eu/v1beta/domains/check"
$DomlistArray = @()
$bearcreds = Get-BearereCreds

$IdnScript = [pscustomobject]@{
idn_script = "cyril"}

Foreach ($Domain in $Domains)
{
$TLD = $Domain.Substring($Domain.IndexOf(".") + 1)
$DomName = ($Domain.Split(".")[0])

$Domlist = [pscustomobject]@{
extension = $TLD
name = $DomName}
$DomlistArray = $DomlistArray + $Domlist
}
```

```
}

$DomBody = [pscustomobject]@{
    additional_data = $IdnScript
    application_mode = "preregistration"
    domains = $DomListArray
    with_price = $true }
$DomBodyJson = $DomBody | ConvertTo-Json

$Bearer = Get-BearereCreds

$Domheader = New-Object "System.Collections.Generic.Dictionary[[String],[String]]"
$Domheader.Add("Authorization","Bearer " + $bearcreds.token)

$Result = Invoke-RestMethod -Method Post -Uri $uri -Headers $Domheader -body $DomBodyJson -Content-Type application/json

return $Result.data.results
}
```

Credits

We've drafted this article using the following sources:

- [PowerShell: Authenticating to the Openprovider API](#)
- [PowerShell: Checking the status of domain registration with Openprovider](#)

We thank the author of these sources for sharing their knowledge with us, to help the Openprovider community enjoy all the possibilities of the Openprovider API.

Revision #2

Created 2026-08-13 10:02:11 UTC by Sarath

Updated 2026-08-13 10:08:40 UTC by Sarath