Table of Contents

[1. Updating From Legacy to Modern Configuration 2](#_Toc198735074)

[1.1 Grant Permissions on Vault Application in Entra Admin Center 2](#_Toc198735075)

[1.2 Update SharePoint Site with PowerShell 2](#_Toc198735076)

[1.3 Authorize Connection between Vault and Office 365 3](#_Toc198735077)

[2. Powershell Script Variables 4](#_Toc198735078)

# Updating From Legacy to Modern Configuration

|  |  |  |
| --- | --- | --- |
| Grant Permissions on Vault Application in Entra Admin Center | | |
| **Step** | **Completed** | **Comments** |
| Navigate to Vault Admin > Settings > Checkout Settings and click **Edit** |  |  |
| In a separate window, login as Admin to Entra Admin Center |  |  |
| Click on **App Registrations** |  |  |
| Navigate to the existing App Registration setup for your Vault |  |  |
| From the Azure app page, click **API permissions** |  |  |
| Click **Add a permission** |  |  |
| Click **Microsoft Graph** and then **Application Permissions** |  |  |
| Under Sites section, check off:   * Sites.Selected   Under Users section, check off:   * User.ReadBasic.All\*   Under Users section, check off:   * User.Invite.All \*\*   Under Directory section, check off:   * Directory.ReadWrite.All \*\*   \*This permission is required if using collaborative authoring workflow actions  \*\*These permissions are for the optional Auto Invite External Users feature |  |  |
| Click **Add permissions** |  |  |
| Click **Grant admin consent for <Tenant Name>** |  |  |
| Update SharePoint Site with PowerShell | | |
| **Step** | **Completed** | **Comments** |
| Login to SharePoint Admin as the administrator user who will be running the Powershell script |  |  |
| Navigate to the site to be updated |  |  |
| Add the SharePoint Admin account that will be running the script as an **Owner** on the site |  |  |
| Install PowerShell 7 (at least version 7) |  |  |
| Open PowerShell 7 and install PnP PowerShell Module with the following command:  Install-Module PnP.PowerShell |  |  |
| Download and open the PowerShell scripts provided in [Vault Help](https://platform.veevavault.help/en/lr/56840/#granting-the-app-access-to-the-sharepoint-team-site) |  |  |
| Prepare the variables in the SitePnP.ps1 script. Each variable must be in a quote. Example: $alias = "Alias"  $adminUrl = “The URL of the SharePoint Admin Center”  $siteBaseUrl = “The base URL for SharePoint sites in the tenant”  $siteTitle = “The title of the new SharePoint site”  $owner = “The email address of the site owner”  $alias = “The unique alias for the SharePoint site”  $tenantUrl = “The tenant domain name used for Azure app registrations”  $appId = “The App ID of the Entra App”  $displayName = “The name of the Entra App created in”  $siteUrl = “The full URL of the SharePoint site constructed using $siteBaseUrl + $alias”  $entraIDApp = “ID of the Azure Admin App that grants Collaborative Authoring Azure App access to the SharePoint site. If left blank, it will be dynamically created upon running this script.  See [Powershell Script Variables](#_Powershell_Script_Variables) for more details. |  |  |
| Run the SitePnP.ps1 script in PowerShell  .\sitepnp.ps1 |  |  |
| Navigate to the newly Created SharePoint site’s **Documents** Library |  |  |
| Copy and paste the URL from SharePoint into Vault Checkout Settings |  | URL:  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Authorize Connection between Vault and Office 365 | | |
| **Step** | **Completed** | **Comments** |
| Check the *Remove Service Account from Collaborative Authoring Configuration* checkbox |  |  |
| Re-enter the **Secret Value** for the app’s Client Secret |  |  |
| Within Vault Checkout Settings, click **Authorize** |  |  |
| Once your Integration Status is verified within Vault Checkout Settings, click **Save** |  |  |

# Powershell Script Variables

*$adminUrl = "https://test-admin.sharepoint.com"*

You can get this URL by navigating to your SharePoint Admin center, and copying the first part of the URL there:

A screenshot of a computer

AI-generated content may be incorrect.

*$siteBaseUrl = "https://test.sharepoint.com/sites/"*

You can get this URL by navigating to any existing SharePoint site and the URL will be up through “/sites/”:

A screenshot of a computer

AI-generated content may be incorrect.

Note: it is important that it end with a forward slash

*$siteTitle = "Site Title"*

This can be defined however is preferred, i.e. “Vault Collaboration”

*$owner = "owner@domain.com"*

This would be the Microsoft username for the administrator who is running the script (i.e. “michaelferrell@devopsveeva.onmicrosoft.com”)

*$tenantUrl = "domain.com"*

You can get this URL by navigating to the Microsoft Entra Admin Center, and the Overview tab. The URL needed is the Primary Domain field:

A screenshot of a computer

AI-generated content may be incorrect.

*$appId = "appId"*

This is the Application (client) ID for the app registration for your Vaults:

A screenshot of a computer

AI-generated content may be incorrect.

*$displayName = "Display Name"*

This is the Display Name for the app registration for your Vaults:

A screenshot of a computer

AI-generated content may be incorrect.

*$siteUrl = “https://test.sharepoint.com/sites/siteAlias”*

If you are setting up a net new site, nothing is needed for this variable (the script will create this using $siteBaseUrl and $alias. If you’re updating an existing site:

This is the URL for your existing SharePoint site, which you can get by navigating to the site in the SharePoint Admin center and copying the Site Address:

A screenshot of a computer

AI-generated content may be incorrect.

*$adminAppName = "Admin App Name"*

If you are not providing $entraIDApp, the script will create an app for you that has the AllSites.FullControl permission using this variable, and you can give this app any name that you prefer. This site is needed to grant your existing application permission to manage your SharePoint site (without a Service Account).

*$entraIDApp = "entraIdApp"*

If you already have an Entra app registration that has the AllSites.FullControl permission, then you can put the Application (client) ID for that app here, rather than using $adminAppName: