# PowerShell による SharePoint Online の設定スクリプト集

2015 年 5 月時点の SharePoint Online の先行リリースを

有効化してある 1 テナントにおいて動作したスクリプト例を紹介します本スクリプトは、あくまでサンプルであり、

すべての環境において、完全な動作を将来に渡って保証するものではありません

本書に記載した情報は、本書各項目に関する発行日現在の Microsoft の見解を表明するものです

Microsoft は絶えず変化する市場に対応しなければならないため、ここに記載した情報に対していかなる責務を負うものではなく、提示された情報の信憑性については保証できません本書は情報提供のみを目的としています

Microsoft は、明示的または暗示的を問わず、本書にいかなる保証も与えるものではありません

SPO スクリプト集 - 1 ページ

# #SPO 管理シェルに接続する

2015年5月26日 14:15

#PowerShell の利用に当たって、Windows 8 は、PowerShell 3.0 を標準搭載しているが、

#Windows 7 SP1 の場合には、最新の 3.0 をダウンロードしてインストールのこと

#Windows Management Framework 3.0 (PowerShell 3.0 含む)

# <http://www.microsoft.com/en-us/download/details.aspx?id=34595>

#SharePoint Point Online 管理シェルを利用する場合、以下をインストールのこと

#SharePoint Online Management Shell

# <http://www.microsoft.com/ja-JP/download/details.aspx?id=35588>

#共通の接続設定 SharePoint Online 管理サイトに、テナント管理者で接続すること

#スクリプトでは以下接続用関数を利用する

function Connect-SPO(){

$adminUrl ="[https://xxxx-admin.sharepoint.com"](https://xxxx-admin.sharepoint.com/)

$admin ="xxxx@xxxx.onmicrosoft.com"

$password ="xxxx"

$secpass = ConvertTo-SecureString $password -asplaintext -force

$cred = New-Object System.Management.Automation.PSCredential($admin, $secpass)

Connect-SPOService -Url $adminUrl -credential $cred

}

SPO スクリプト集 - 2 ページ

## #CSOM を使って接続する

2015年5月26日 14:13

#PowerShell の利用に当たって、Windows 8 は、PowerShell 3.0 を標準搭載しているが、

#Windows 7 SP1 の場合には、最新の 3.0 をダウンロードしてインストールのこと

#Windows Management Framework 3.0 (PowerShell 3.0 含む)

# <http://www.microsoft.com/en-us/download/details.aspx?id=34595>

#PowerShell から CSOM を利用するには、以下のモジュールをインストールのこと

#SharePoint Online Client Components SDK

# <http://www.microsoft.com/en-us/download/details.aspx?id=42038&44F86079-8679-400C-BFF2-9CA5F2BCBDFC=1>

#共通の接続設定 設定を行いたいサイトに、サイト コレクションの管理者の ID で接続する

#スクリプトでは以下接続用関数を利用する

[void][System.Reflection.Assembly]::LoadWithPartialName("Microsoft.SharePoint.Client")

function Connect-CSOM($site){

$admin ="xxxx@xxxx.onmicrosoft.com"

$password ="xxxx"

$global:ctx = New-Object Microsoft.SharePoint.Client.ClientContext($site)

$secpass = ConvertTo-SecureString $password -asplaintext -force

$global:ctx.Credentials = New-Object Microsoft.SharePoint.Client.SharePointOnlineCredentials($admin, $secpass) }

SPO スクリプト集 - 3 ページ

# #POST を使って設定する

2015年5月27日 10:34

CSOM で操作ができないプロパティ設定の自動化を行いたい場合、

PowerShell の Invoke-WebRequest を利用して、

ブラウザと同等の設定ページの GET およびフォーム内容の POST を行って、スクリプトから設定を行う

CSOM で取得した Cookie をセッション情報として持たせることで、認証を通すことが可能他フォームで利用したい場合には、Input フィールドの name や Value などを分析して、

POST する内容に反映する必要がある

SPO スクリプト集 - 4 ページ

# #Web サービスを利用する

2015年5月27日 10:36

ユーザー プロファイルの全件取得など、CSOM から操作できない設定を自動化する場合、

SOAP の Web サービスを通じた、操作が可能

CSOM で取得した Cookie をセッション情報として持たせることで、認証を通すことが可能

## サイト コレクションを作成する (管理シェル)

2015年5月26日 14:09

#SPO 管理シェル接続

Connect-SPO

#チーム サイトを作成

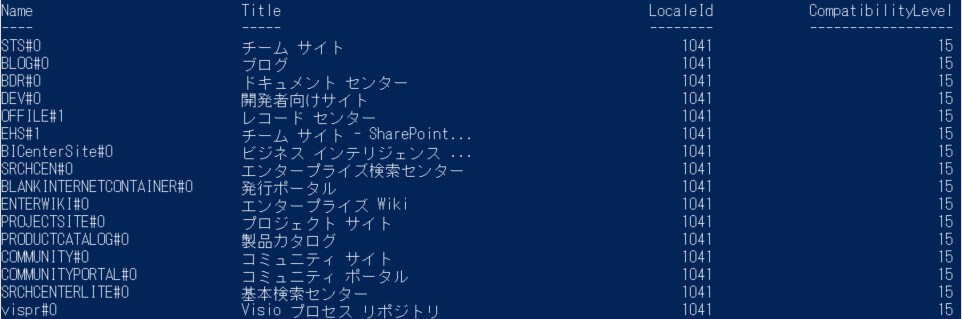
$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share)

$owner = "xxxx@xxxx.onmicrosoft.com"

New-SPOSite -url $siteUrl -owner $owner -StorageQuota 1000 -LocaleId 1041 -ResourceQuota 300 -Template "STS#0" TimeZoneId 20 -Title "チーム サイト"

#参考テンプレート一覧

Get-SPOWebTemplate



SPO スクリプト集 - 6 ページ

# コレクション管理者を追加する (管理シェル)

2015年5月26日 17:37

#SPO 管理シェル接続

Connect-SPO

#既存のサイト コレクション管理者は変更せず、サイト コレクション管理者を追加する

$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share)

$secondaryOwner = "xxxx@xxxx.onmicrosoft.com"

Set-SPOUser -Site $siteUrl -LoginName $secondaryOwner -IsSiteCollectionAdmin $true

#OneDrive for Business のサイトに対しても URL を指定することで、同等の操作が可能

コレクションで外部共有を有効化する (管理シェル)

2015年5月26日 17:47

#SPO 管理シェル接続

Connect-SPO

#サイト コレクションで外部共有を有効化

$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share)

Set-SPOSite -Identity $siteUrl -SharingCapability ExternalUserSharingOnly

#-SharingCapabilit では以下のいずれかが設定可能

Disabled

ExternalUserSharingOnly

ExternalUserAndGuestSharing

# コレクションで容量を制限する (管理シェル)

2015年6月30日 10:17

#SPO 管理シェル接続

Connect-SPO

#サイト コレクションで外部共有を有効化

$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share)

#MB 単位で容量制限を設定

Set-SPOSite -Identity $siteUrl -StorageQuota 1000 -StorageQuotaWarningLevel 800

#OneDrive for Business のサイトに対しても URL を指定することで、同等の操作が可能

$user="user@xxx.onmicrosoft.com"

$user=$user.replace("@","\_").replace(".","\_")

$odUrl="https://xxxx-my.sharepoint.com/personal/"+$user

Set-SPOSite -Identity $odUrl -StorageQuota 1000 -StorageQuotaWarningLevel 800

サイト コレクションでメンバー権限での共有を変更する(POST)

2015年5月26日 14:09

#以下の共有設定を PowerShell からの POST 要求で書き換え、サイト メンバーによる共有を禁止する

#CSOM 接続

$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share)

$mailto ="admin@xxxx.onmicrosoft.com" Connect-CSOM $siteUrl

#認証 Cookie をセッション変数として持たせる

$cookie =$ctx.Credentials.GetAuthenticationCookie($siteUrl)

$ses=new-object Microsoft.PowerShell.Commands.WebRequestSession $ses.Cookies.SetCookies($siteUrl,$cookie)

#権限設定の Web ページ

$target = $siteUrl+"/\_layouts/15/setrqacc.aspx?type=web&isDlg=1"

#まずは、Web ページを開き、フォームの内容を読み取る

$r=Invoke-WebRequest $target -WebSession $ses -usebasicParsing

$fields=$r|select -ExpandProperty inputfields | select name, value, id

#POST すべき内容を抽出する(POST の変数名はID ではなく Name )

$par = New-Object "System.Collections.Generic.Dictionary``2[System.String,System.String]" foreach($f in $fields){

if($f.name -ne $null -and !$f.name.StartsWith('ctl00$ctl'))

{$par.Add($f.Name,$f.value)}

}

#変更して送信する内容を POST の変数にセットする

#$ によって PowerShell の変数として認識されないよう ' で囲む

$par['ctl00$PlaceHolderMain$ctl00$chkMembersCanShare']=''

$par['ctl00$PlaceHolderMain$ctl00$chkMembersCanAddToGroup']=''

$par['ctl00$PlaceHolderMain$ctl00$chkRequestAccess']='on'

$par['ctl00$PlaceHolderMain$ctl00$txtEmail']=$mailto

$par[’\_\_EVENTTARGET’]='ctl00$PlaceHolderMain$ctl01$RptControls$btnSubmit'

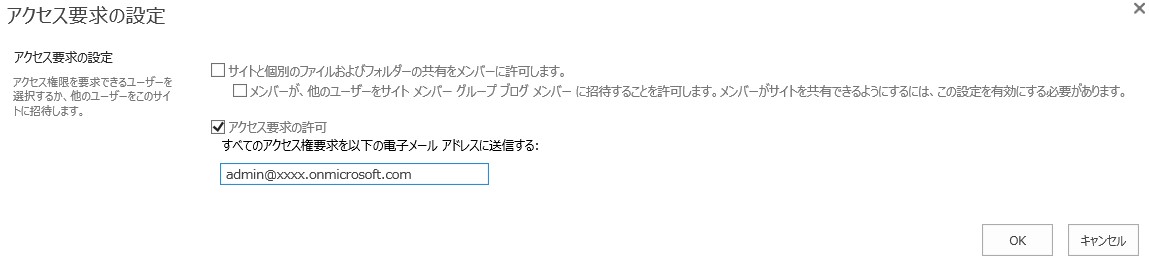
SPO スクリプト集 - 10 ページ

$par.Remove('ctl00$PlaceHolderMain$ctl01$BtnCancel')

#POST する

$r2=Invoke-WebRequest $target -WebSession $ses -Method POST -Body $par -usebasicParsing

#上記実行後の結果



SPO スクリプト集 - 11 ページ

サイト コレクションの監査設定をする (CSOM)

2015年5月26日 14:41

#CSOM 接続

$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share) Connect-CSOM $siteUrl

#サイト コレクションで記録する監査対象のイベントを指定する

$site=$ctx.site

$site.Audit.AuditFlags=@([Microsoft.SharePoint.Client.AuditMaskType]::Update,[Microsoft.SharePoint.Client.AuditMaskType]::Mov e,[Microsoft.SharePoint.Client.AuditMaskType]::Copy,[Microsoft.SharePoint.Client.AuditMaskType]::ObjectDelete,[Microsoft.ShareP oint.Client.AuditMaskType]::UnDelete,[Microsoft.SharePoint.Client.AuditMaskType]::SecurityChange) $site.Audit.Update()

$ctx.ExecuteQuery()

#30日を超えたデータは、トップサイトの、サイトのリソース ファイルのライブラリに、Excel形式で保存しておくよう設置する

$ctx.load($site)

$ctx.ExecuteQuery()

$web=$site.rootweb

$site.TrimAuditLog=$true

$site.AuditLogTrimmingRetention=30

$target=($site.ServerRelativeUrl +"/SiteAssets").Replace("//","/")

$web.AllProperties["\_auditlogreportstoragelocation"]=$target

$web.Update()

$ctx.ExecuteQuery()

#設定後



#なお監査項目の指定は、以下の種類がある

[Microsoft.SharePoint.Client.AuditMaskType]

All

SPO スクリプト集 - 12 ページ

CheckIn

CheckOut

ChildDelete

Copy

Move

None

ObjectDelete

ProfileChange

SchemaChange

Search

SecurityChange

Undelete

Update View

Workflow

SPO スクリプト集 - 13 ページ

# サイトのタイムゾーンと第二言語を設定する (CSOM)

2015年5月26日 15:03

#CSOM 接続

$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share)

Connect-CSOM $siteUrl

$web=$ctx.web

#タイムゾーン一覧から、日本時間を選択する

$tz=$web.RegionalSettings.TimeZones.GetById(20) $ctx.load($tz)

#取得したタイムゾーンをサイトに設定する

$web.RegionalSettings.TimeZone=$tz

$web.Update()

$ctx.ExecuteQuery()

#日本語サイトに、第二言語として英語の追加をする

$web.IsMultilingual=$true

$web.AddSupportedUILanguage(1033)

$web.Update()

$ctx.ExecuteQuery()

#タイムゾーン設定一覧

Id Description

-- -----------

39 (UTC-12:00) 国際日付変更線 西側

95 (UTC-11:00) 協定世界時 - 11

15 (UTC-10:00) ハワイ

14 (UTC-09:00) アラスカ

78 (UTC-08:00) バハカリフォルニア

13 (UTC-08:00) 太平洋標準時 (米国およびカナダ)

38 (UTC-07:00) アリゾナ

77 (UTC-07:00) チワワ、ラパス、マサトラン

12 (UTC-07:00) 山地標準時 (米国およびカナダ)

55 (UTC-06:00) 中央アメリカ

11 (UTC-06:00) 中部標準時 (米国およびカナダ)

37 (UTC-06:00) グアダラハラ、メキシコシティ、モンテレー

36 (UTC-06:00) サスカチュワン

35 (UTC-05:00) ボゴタ、リマ、キト

10 (UTC-05:00) 東部標準時 (米国およびカナダ)

34 (UTC-05:00) インディアナ東部

88 (UTC-04:30) カラカス

91 (UTC-04:00) アスンシオン

9 (UTC-04:00) 大西洋標準時 (カナダ)

81 (UTC-04:00) クイアバ

33 (UTC-04:00) ジョージタウン、ラパス、マナウス、サンフアン

65 (UTC-04:00) サンティアゴ

28 (UTC-03:30) ニューファンドランド

8 (UTC-03:00) ブラジリア

85 (UTC-03:00) ブエノスアイレス

32 (UTC-03:00) カイエンヌ、フォルタレザ

60 (UTC-03:00) グリーンランド

90 (UTC-03:00) モンテビデオ

103 (UTC-03:00) サルバドル

96 (UTC-02:00) 協定世界時 - 02

30 (UTC-02:00) 中央大西洋

29 (UTC-01:00) アゾレス諸島

53 (UTC-01:00) カーボベルデ

86 (UTC) カサブランカ

93 (UTC) 協定世界時

2 (UTC) ダブリン、エジンバラ、リスボン、ロンドン

31 (UTC) モンロビア、レイキャビク

4 (UTC+01:00) アムステルダム、ベルリン､ベルン、ローマ､スト... 6 (UTC+01:00) ベオグラード、ブラチスラバ、ブダペスト、リュ... 3 (UTC+01:00) ブリュッセル、コペンハーゲン、マドリード、パリ

57 (UTC+01:00) サラエボ、スコピエ、ワルシャワ、ザグレブ

69 (UTC+01:00) 西中央アフリカ

83 (UTC+01:00) ウィントフック

79 (UTC+02:00) アンマン

5 (UTC+02:00) アテネ、ブカレスト、イスタンブール

80 (UTC+02:00) ベイルート

49 (UTC+02:00) カイロ

98 (UTC+02:00) ダマスカス

50 (UTC+02:00) ハラーレ、プレトリア

59 (UTC+02:00) ヘルシンキ、キエフ、リガ、ソフィア、タリン、...

101 (UTC+02:00) イスタンブール

27 (UTC+02:00) エルサレム

7 (UTC+02:00) ミンスク (旧)

104 (UTC+02:00) 東ヨーロッパ

100 (UTC+02:00) カリーニングラード (RTZ 1)

26 (UTC+03:00) バグダッド

74 (UTC+03:00) クウェート、リヤド

109 (UTC+03:00) ミンスク

51 (UTC+03:00) モスクワ、サンクトペテルブルク、ボルゴグラー...

56 (UTC+03:00) ナイロビ 25 (UTC+03:30) テヘラン

24 (UTC+04:00) アブダビ、マスカット

54 (UTC+04:00) バク

106 (UTC+04:00) イジェフスク、サマーラ (RTZ 3)

89 (UTC+04:00) ポートルイス

82 (UTC+04:00) トビリシ

84 (UTC+04:00) エレバン

48 (UTC+04:30) カブール

58 (UTC+05:00) エカテリンバーグ (RTZ 4)

87 (UTC+05:00) イスラマバード、カラチ

47 (UTC+05:00) タシケント

23 (UTC+05:30) チェンナイ、コルカタ、ムンバイ、ニューデリー

66 (UTC+05:30) スリ・ジャヤワルダナプラ

62 (UTC+05:45) カトマンズ

71 (UTC+06:00) アスタナ

102 (UTC+06:00) ダッカ

46 (UTC+06:00) ノボシビルスク (RTZ 5)

61 (UTC+06:30) ヤンゴン (ラングーン)

22 (UTC+07:00) バンコク、ハノイ、ジャカルタ

64 (UTC+07:00) クラスノヤルスク (RTZ 6)

45 (UTC+08:00) 北京、重慶、香港、ウルムチ

63 (UTC+08:00) イルクーツク (RTZ 7)

21 (UTC+08:00) クアラルンプール、シンガポール

73 (UTC+08:00) パース

75 (UTC+08:00) 台北

94 (UTC+08:00) ウランバートル

20 (UTC+09:00) 大阪、札幌、東京

72 (UTC+09:00) ソウル

70 (UTC+09:00) ヤクーツク (RTZ 8)

19 (UTC+09:30) アデレード

44 (UTC+09:30) ダーウィン

18 (UTC+10:00) ブリスベン

76 (UTC+10:00) キャンベラ、メルボルン、シドニー

43 (UTC+10:00) グアム、ポートモレスビー

42 (UTC+10:00) ホバート

99 (UTC+10:00) マガダン

68 (UTC+10:00) ウラジオストク、マガダン (RTZ 9)

107 (UTC+11:00) チョクルダフ (RTZ 10)

41 (UTC+11:00) ソロモン諸島、ニューカレドニア

108 (UTC+12:00) アナディリ、ペトロパブロフスク-カムチャツキ...

17 (UTC+12:00) オークランド、ウェリントン

97 (UTC+12:00) 協定世界時 + 12

40 (UTC+12:00) フィジー

92 (UTC+12:00) ペトロパブロフスク-カムチャツキー - 旧

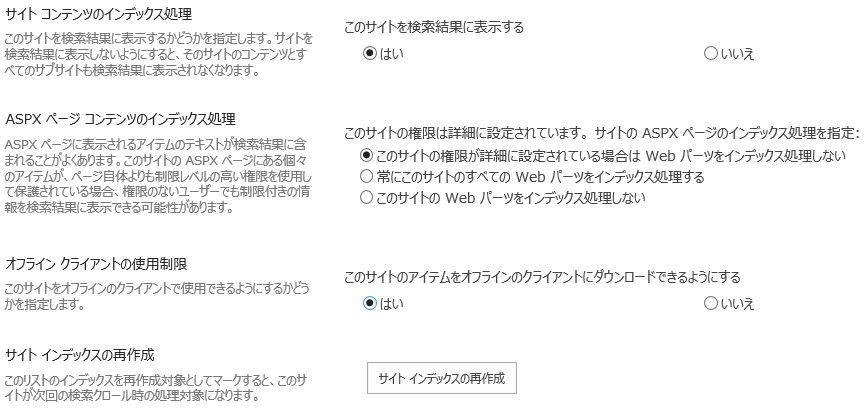
67 (UTC+13:00) ヌクアロファ

16 (UTC+13:00) サモア

# サイトのオフライン同期と検索結果表示を禁止する(POST)

2015年5月27日 10:01

#以下のサイト設定である「検索とオフラインでの使用制限 」にある、検索結果表示と、オフライン同期を禁止する



#CSOM 接続

$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share) Connect-CSOM $siteUrl

#認証 Cookie をセッション変数として持たせる

$cookie =$ctx.Credentials.GetAuthenticationCookie($siteUrl)

$ses=new-object Microsoft.PowerShell.Commands.WebRequestSession $ses.Cookies.SetCookies($siteUrl,$cookie)

#権限設定の Web ページ

$target = $siteUrl+"/\_layouts/15/srchvis.aspx?isDlg=1"

#まずは、Web ページを開き、フォームの内容を読み取る

$r=Invoke-WebRequest $target -WebSession $ses -usebasicParsing

$fields=$r|select -ExpandProperty inputfields | select name, value, id

#POST すべき内容を抽出する(POST の変数名はID ではなく Name )

$par = New-Object "System.Collections.Generic.Dictionary``2[System.String,System.String]"

foreach($f in $fields){

if($f.name -ne $null -and !$par.Keys.Contains($f.name) -and !$f.name.StartsWith('ctl00$ctl'))

{$par.Add($f.Name,$f.value)}

}

#変更して送信する内容を POST の変数にセットする

SPO スクリプト集 - ページ

#$ によって PowerShell の変数として認識されないよう ' で囲む

$par['ctl00$PlaceHolderMain$IndexSiteContent$ctl01$RadIndexSiteContent']='radIndexSiteContentNo'

$par['ctl00$PlaceHolderMain$IndexAspxContent$IndexAspxContentControl

$RadIndexAspxContent']='radIndexAspxContentAuto'

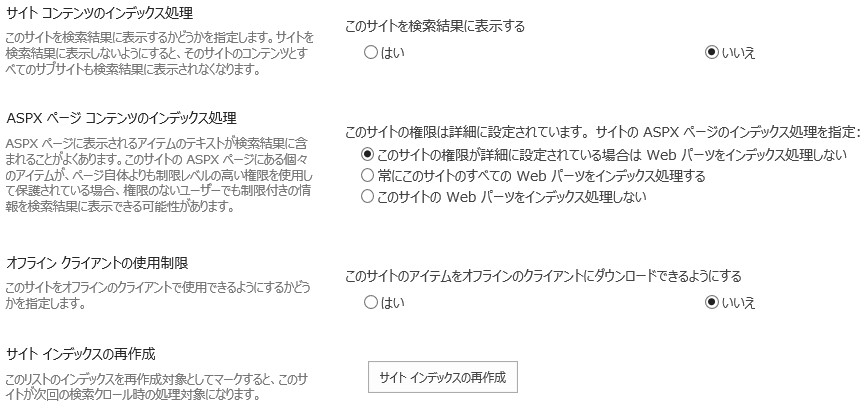
$par['ctl00$PlaceHolderMain$AllowSyncSection$ctl01$AllowSync']='RadAllowSyncNo'

$par[’\_\_EVENTTARGET’]='ctl00$PlaceHolderMain$ctl00$RptControls$btnSubmit' $par.Remove('ctl00$PlaceHolderMain$ctl00$BtnCancel')

#POST する

$r2=Invoke-WebRequest $target -WebSession $ses -Method POST -Body $par -usebasicParsing

#上記実行後の結果



SPO スクリプト集 - ページ

## サイトで外部ユーザーに招待を出す(CSOM)

2015年5月26日 14:08

#CSOM 接続

$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share)

Connect-CSOM $siteUrl

$web=$ctx.web

#招待先のユーザーと、付与する権限 None, View, Edit, Owner を指定する

$ura=new-object Microsoft.SharePoint.Client.Sharing.UserRoleAssignment

$ura.UserId="imikiyasu@xxxx.com"

$ura.Role=[Microsoft.SharePoint.Client.Sharing.Role]::View

#ジェネリック リストに、招待ユーザー、権限の設定を追加する(複数可)

$uras= new-object System.Collections.Generic.List[Microsoft.SharePoint.Client.Sharing.UserRoleAssignment] $uras.Add($ura)

#招待を出す

[Microsoft.SharePoint.Client.Sharing.WebSharingManager]::UpdateWebSharingInformation($ctx,$web, $uras,$true,"ようこそ",$false,$true)

$ctx.ExecuteQuery()

#招待を出す パラメータは下記URL 参照

# [https://msdn.microsoft.com/en-](https://msdn.microsoft.com/en-us/library/office/microsoft.sharepoint.client.sharing.websharingmanager.updatewebsharinginformation.aspx)

[us/library/office/microsoft.sharepoint.client.sharing.websharingmanager.updatewebsharinginformation.aspx](https://msdn.microsoft.com/en-us/library/office/microsoft.sharepoint.client.sharing.websharingmanager.updatewebsharinginformation.aspx) #public static IList<UserSharingResult> UpdateWebSharingInformation(

# ClientRuntimeContext context,

# Web web,

# IList<UserRoleAssignment> userRoleAssignments,

# bool sendServerManagedNotification,

# string customMessage,

# bool additivePermission,

# bool allowExternalSharing

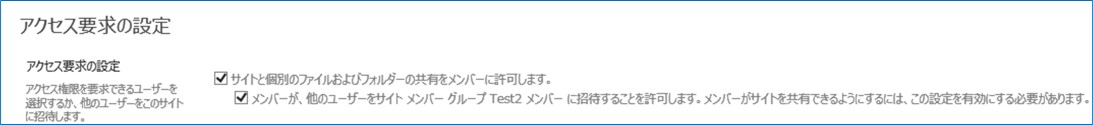
#)

SPO スクリプト集 - 20 ページ

# 参考外部招待時の設定の違い

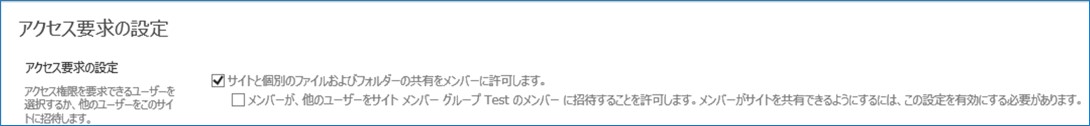
2015年5月26日 14:23

(1) 既定の設定

権限による動作の違い

|  |  |  |  |
| --- | --- | --- | --- |
|  | 社内・外部ユーザーのサイトへの招待 | フォルダ・ファイルの共有 | 匿名ゲストリンク |
| 管理者 | グループによる権限を指定した即時の共有 | 編集か表示かを指定した社内・外部への即時の共有 | 編集か表示かを指定した  即時のゲスト リンクの生成とメールでの共有 |
| メンバー | 社内・外部ユーザーの  メンバー グループへの即時の招待 | 編集か表示かを指定した社内・外部への即時の共有 | 編集か表示かを指定した  即時のゲスト リンクの生成とメールでの共有 |
| 参照者 | 社内ユーザーへは  管理者承認依頼での招待社外ユーザーへは NG | 社内ユーザーへは  管理者承認依頼での招待社外ユーザーへは NG | ×  ゲストリンクの作成不可作成済みリンクの参照不可 |

(2)メンバーによるメンバーの招待を禁止



権限による動作の違い

|  |  |  |  |
| --- | --- | --- | --- |
|  | 社内・外部ユーザーのサイトへの招待 | フォルダ・ファイルの共有 | 匿名ゲストリンク |
| 管理者 | グループによる権限を指定した即時の共有 | 編集か表示かを指定した社内・外部への即時の共有 | 編集か表示かを指定した  即時のゲスト リンクの生成とメールでの共有 |
| メンバー | 社内ユーザーへは  管理者承認依頼での招待社外ユーザーへは NG | 編集か表示かを指定した社内・外部への即時の共有 | 編集か表示かを指定した  即時のゲスト リンクの生成とメールでの共有 |
| 参照者 | 社内ユーザーへは管理者承認依頼での招待 | 社内ユーザーへは管理者承認依頼での招待 | ×  ゲストリンクの作成不可 |

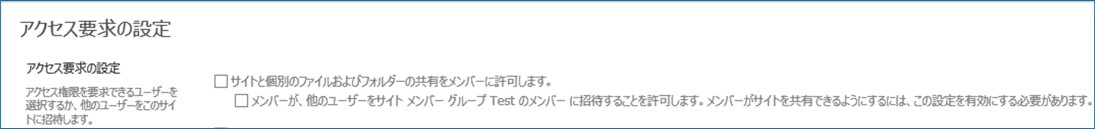
作成済みリンクの参照不可

SPO スクリプト集 - 21 ページ

作成済みリンクの参照不可

|  |  |  |  |
| --- | --- | --- | --- |
|  | 社外ユーザーへは NG | 社外ユーザーへは NG |  |

(3)ファイルとフォルダの共有も禁止

権限による動作の違い

|  |  |  |  |
| --- | --- | --- | --- |
|  | 社内・外部ユーザーのサイトへの招待 | フォルダ・ファイルの共有 | 匿名ゲストリンク |
| 管理者 | グループによる権限を指定した即時の共有 | 編集か表示かを指定した社内・外部への即時の共有 | 編集か表示かを指定した  即時のゲスト リンクの生成とメールでの共有 |
| メンバー | 社内ユーザーへは  管理者承認依頼での招待社外ユーザーへは NG | 社内ユーザーへは  管理者承認依頼での招待社外ユーザーへは NG | ×  ゲストリンクの作成不可作成済みリンクの参照不可 |
| 参照者 | 社内ユーザーへは  管理者承認依頼での招待社外ユーザーへは NG | 社内ユーザーへは  管理者承認依頼での招待社外ユーザーへは NG | ×  ゲストリンクの作成不可作成済みリンクの参照不可 |

SPO スクリプト集 - 22 ページ

## 外部ユーザーを一覧で取得する・削除する (管理シェル)

2015年5月26日 14:27

#SPO 管理シェル接続

Connect-SPO

#外部ユーザーの取得 最大 50 件ごと

$max=50

$userslist=@() try{

$users=Get-SPOExternalUser -SiteUrl <https://xxxx.sharepoint.com/sites/partner/> -PageSize $max -position $userslist.count

while($users.count -eq $max){ $userslist+=$users

$users=Get-SPOExternalUser -SiteUrl <https://xxxx.sharepoint.com/sites/partner/> -PageSize $max -position $userslist.count

}

}

catch [Exception]

{

}

$userslist

#外部ユーザーの削除

$user = Get-SPOExternalUser -Filter someone@example.com

Remove-SPOExternalUser -UniqueIDs @($user.UniqueId)

SPO スクリプト集 - 23 ページ

## IRM ライブラリを設定する (CSOM)

2015年5月26日 14:44

function SetIRM($l){

$p=$l.InformationRightsManagementSettings

#「ダウンロード時にこのライブラリの権限を制限する」の設定

$l.IrmEnabled=$true

#「アクセス許可ポリシーのタイトルを入力してください」の設定

$p.PolicyTitle="社外秘"

#「アクセス許可ポリシーの説明を入力してください」の設定

$p.PolicyDescription="このコンテンツは、社外秘のため、印刷、コピー&ペーストが禁止されており、暗号化して管理されています。またダウンロード後10日間のみ参照可能です。"

#「IRM をサポートしないドキュメントのアップロードをユーザーに許可しない」の設定

$l.IrmReject=$true

#「このライブラリに対するアクセス権の制限を解除する日」の設定

$l.IrmExpire=$false

$p.DocumentLibraryProtectionExpireDate="2015/01/16 13:23:02"

#「このドキュメント ライブラリではドキュメントをブラウザーで開かないようにする」の設定 (Office Online での表示を許可するかどうかの設定)

$p.DisableDocumentBrowserView=$false

# Office Online の 印刷用ページを許可するかどうかの設定 (UI での設定項目なし、過去 Excel Online で印刷用ページがあったが、現在は存在せず)

$p.EnableDocumentBrowserPublishingView=$false

#「閲覧者に印刷を許可する」の設定

$p.AllowPrint=$false

#「ダウンロードしたドキュメント上でスクリプトの実行およびスクリーン リーダーの使用を閲覧者に許可する」の設定

$p.AllowScript=$false

#「ダウンロードしたドキュメントのコピーに対する書き込みを閲覧者に許可する」の設定

$p.AllowWriteCopy=$false

#「ダウンロード後、ドキュメントのアクセス権はこの日数 (1 ～ 365) 後に失効する」の設定

$p.EnableDocumentAccessExpire=$true

$p.DocumentAccessExpireDays=10

#「ユーザーはこの期間 (日数) を使用して自分の資格情報を確認する必要がある」の設定

$p.EnableLicenseCacheExpire=$true

$p.LicenseCacheExpireDays=2

#「グループの保護を許可します。既定のグループ」の設定

$p.EnableGroupProtection=$false

$p.GroupName=""

#更新

$p.Update()

$l.Update()

}

#CSOM 接続

$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share)

SPO スクリプト集 - 24 ページ

Connect-CSOM $siteUrl

$list=$ctx.Web.Lists.GetByTitle("ドキュメント")

SetIRM $list

$ctx.ExecuteQuery()

SPO スクリプト集 - 25 ページ

## OneDrive for Business のサイト URL を取得する (Web サービス)

2015年5月26日 14:53

#OneDrive for Business のベースを設定する $baseUrl ="[https://xxxx-my.sharepoint.com"](https://xxxx-my.sharepoint.com/)

#CSOM で管理サイトにテナント管理者で接続する

$adminUrl = "[https://xxxx-admin.sharepoint.com"](https://xxxx-admin.sharepoint.com/)

Connect-CSOM $adminUrl

$proxyaddr = "$adminUrl/\_vti\_bin/UserProfileService.asmx?wsdl"

$ups= New-WebServiceProxy -Uri $proxyaddr -UseDefaultCredential False

$cookie = $ctx.Credentials.GetAuthenticationCookie($adminUrl)

$container = New-Object System.Net.CookieContainer

$container.SetCookies($adminUrl,$cookie)

$ups.CookieContainer = $container

$PersonalSiteUrlList=@()

$UserProfileResult = $ups.GetUserProfileByIndex(-1)

$NumProfiles = $ups.GetUserProfileCount()

$i = 1

While ($UserProfileResult.NextValue -ne -1) {

$Prop = $UserProfileResult.UserProfile | Where-Object { $\_.Name -eq "PersonalSpace" }

$Url= $Prop.Values[0].Value

if ($Url) {$PersonalSiteUrlList+=$baseUrl+$Url}

$UserProfileResult = $ups.GetUserProfileByIndex($UserProfileResult.NextValue) $i++

}

$PersonalSiteUrlList

#出力例

<https://kkiw-my.sharepoint.com/personal/yoshi_kkstu_jp/><https://kkiw-my.sharepoint.com/personal/demo2_kkiw_onmicrosoft_com/><https://kkiw-my.sharepoint.com/personal/demo1_kkiw_onmicrosoft_com/><https://kkiw-my.sharepoint.com/personal/admin_kkiw_onmicrosoft_com/>

## ユーザー プロファイルの上司の値を書き込む (CSOM)

2015年5月27日 10:33

$targetUser="xxxx@xxxx.onmicrosoft.com" $manager="xxxx@xxxx.onmicrosoft.com"

#CSOM で管理サイトにテナント管理者で接続する

$adminUrl = "[https://xxxx-admin.sharepoint.com"](https://xxxx-admin.sharepoint.com/) Connect-CSOM $adminUrl

#対象ユーザーのプロパティを取得

[void][System.Reflection.Assembly]::LoadWithPartialName("Microsoft.SharePoint.Client.UserProfiles")

$pm=new-object Microsoft.SharePoint.Client.UserProfiles.PeopleManager($ctx)

$prop=$pm.GetPropertiesFor("i:0#.f|membership|"+$targetUser)

$ctx.load($prop)

$ctx.ExecuteQuery()

#対象ユーザーの上司属性に上司を設定

$pm.SetSingleValueProfileProperty($prop.AccountName, "Manager", "i:0#.f|membership|"+$manager) $ctx.ExecuteQuery()

## ユーザー プロファイルの上司の値を書き込む (Web サービス)

2015年5月27日 10:33

$targetUser="xxxx@xxxx.onmicrosoft.com" $manager="xxxx@xxxx.onmicrosoft.com"

#CSOM で管理サイトにテナント管理者で接続する

$adminUrl = "[https://xxxx-admin.sharepoint.com"](https://xxxx-admin.sharepoint.com/)

Connect-CSOM $adminUrl

$proxyaddr = "$adminUrl/\_vti\_bin/UserProfileService.asmx?wsdl"

$ups= New-WebServiceProxy -Uri $proxyaddr -UseDefaultCredential False

$cookie = $ctx.Credentials.GetAuthenticationCookie($adminUrl)

$container = New-Object System.Net.CookieContainer

$container.SetCookies($adminUrl,$cookie)

$ups.CookieContainer = $container

#Webサービスの型の名称

$up="Microsoft.PowerShell.Commands.NewWebserviceProxy.AutogeneratedTypes.WebServ iceProxy1n\_UserProfileService\_asmx\_wsdl"

#プロパティ設定用の配列を準備

$newdata = new-object ($up+".PropertyData[]")1 $newdata[0] = new-object ($up+".PropertyData")

$newdata[0].Values =new-object ($up+".ValueData[]")1 $newdata[0].Values[0] =new-object ($up+".ValueData")

#上司を admin に設定

$newdata[0].Name = "Manager";

$newdata[0].IsValueChanged = $true

$newdata[0].Values[0].Value = "i:0#.f|membership|"+$manager

#指定したユーザーのプロファイルを書き換え

$ups.ModifyUserPropertyByAccountName("i:0#.f|membership|"+$targetUser, $newdata)

## 対象のユーザーの Delve を無効化・有効化する (CSOM)

2015年5月26日 17:22

$targetUser="xxxx@xxxx.onmicrosoft.com"

#CSOM で管理サイトにテナント管理者で接続する

$adminUrl = "[https://xxxx-admin.sharepoint.com"](https://xxxx-admin.sharepoint.com/) Connect-CSOM $adminUrl

#対象ユーザーのプロパティを取得

[void]

[System.Reflection.Assembly]::LoadWithPartialName("Microsoft.SharePoint.Client.UserProfil es")

$pm=new-object Microsoft.SharePoint.Client.UserProfiles.PeopleManager($ctx)

$prop=$pm.GetPropertiesFor("i:0#.f|membership|"+$targetUser)

$ctx.load($prop)

$ctx.ExecuteQuery()

#Office Graph を有効化する場合

$pm.SetSingleValueProfileProperty($prop.AccountName, "OfficeGraphEnabled", $true)

$pm.SetSingleValueProfileProperty($prop.AccountName, "DelveFlags", 4166)

$ctx.ExecuteQuery();

#Office Graph を無効化する場合

$pm.SetSingleValueProfileProperty($prop.AccountName, "OfficeGraphEnabled", $false)

$pm.SetSingleValueProfileProperty($prop.AccountName, "DelveFlags", 70)

$ctx.ExecuteQuery();

## 対象のユーザーの Delve を無効化・有効化する (Web サービス)

2015年5月26日 17:22

$targetUser="xxxx@xxxx.onmicrosoft.com"

#CSOM で管理サイトにテナント管理者で接続する

$adminUrl = "[https://xxxx-admin.sharepoint.com"](https://xxxx-admin.sharepoint.com/)

Connect-CSOM $adminUrl

$proxyaddr = "$adminUrl/\_vti\_bin/UserProfileService.asmx?wsdl"

$ups= New-WebServiceProxy -Uri $proxyaddr -UseDefaultCredential False

$cookie = $ctx.Credentials.GetAuthenticationCookie($adminUrl)

$container = New-Object System.Net.CookieContainer

$container.SetCookies($adminUrl,$cookie)

$ups.CookieContainer = $container

#Webサービスの型の名称

$up="Microsoft.PowerShell.Commands.NewWebserviceProxy.AutogeneratedTypes.WebServ iceProxy1n\_UserProfileService\_asmx\_wsdl"

#プロパティ設定用の配列を準備

$newdata = new-object ($up+".PropertyData[]")2 $newdata[0] = new-object ($up+".PropertyData")

$newdata[0].Values =new-object ($up+".ValueData[]")1 $newdata[0].Values[0] =new-object ($up+".ValueData")

$newdata[1] = new-object ($up+".PropertyData")

$newdata[1].Values =new-object ($up+".ValueData[]")1 $newdata[1].Values[0] =new-object ($up+".ValueData")

#Office Graph を無効化

$newdata[0].Name = "OfficeGraphEnabled"

$newdata[0].IsValueChanged = $true

$newdata[0].Values[0].Value = $false

#フラグを設定

$newdata[1].Name = "DelveFlags"

$newdata[1].IsValueChanged = $true

$newdata[1].Values[0].Value = 4166

#指定したユーザーのプロファイルを書き換え

$ups.ModifyUserPropertyByAccountName("i:0#.f|membership|"+$targetUser, $newdata) #Office Graph を有効化する場合

$newdata[0].Name = "OfficeGraphEnabled"

$newdata[0].IsValueChanged = $true

$newdata[0].Values[0].Value = $true

#フラグを設定

$newdata[1].Name = "DelveFlags"

$newdata[1].IsValueChanged = $true

$newdata[1].Values[0].Value = 70

#指定したユーザーのプロファイルを書き換え

$ups.ModifyUserPropertyByAccountName("i:0#.f|membership|"+$targetUser, $newdata)

## サイトのグループにユーザーを追加する (CSOM)

2015年5月26日 14:41

#CSOM 接続

$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share)

Connect-CSOM $siteUrl

$user ="xxxx@xxxx.onmicrosoft.com"

$web=$ctx.web

#サイトの既定のメンバー グループを取得する

$member=$web.AssociatedMemberGroup

#ユーザー ID を解決して、追加する

$u=$web.EnsureUser($user)

$u=$member.Users.AddUser($u) $ctx.ExecuteQuery()

#グループは以下のバリエーションもある

#サイトの既定の所有者グループを取得する場合

$member=$web.AssociatedOwnerGroup

#サイトの既定の閲覧者グループを取得する場合

$member=$web.AssociatedVisitorGroup

### サイト コレクションを作成してローカルに保存されたサイト テンプレートをルートサイトに適用する (管理シェル & CSOM)

2015年6月30日 12:44

#SPO 管理シェル接続

Connect-SPO

#新規に作成するサイトの URL

$siteUrl = "[https://xxx.sharepoint.com/sites/fromtemp"](https://xxx.sharepoint.com/sites/fromtemp)

#ローカルに保存してあるサイト テンプレート

$localfile ="C:\Users\Demo1\Desktop\CustomTempalte01.wsp"

#サイトをテンプレート保存した際の名称

$templateName="CustomTemplate01"

#サイトのURL から想定されるテンプレートの保存先

$targeturl ="/sites/fromtemp/\_catalogs/solutions/CustomTemplate01.wsp"

$owner = "xxxx@xxx.onmicrosoft.com"

#テンプレートを適用しないサイトを作成

New-SPOSite -url $siteUrl -owner $owner -StorageQuota 1000 -ResourceQuota 300 -TimeZoneId 20 -Title "チーム サイト"

#ソリューション ギャラリーに、テンプレートをアップロードする

Connect-CSOM $siteUrl

$fs = new-object System.IO.FileStream($localfile ,[System.IO.FileMode]::Open,[System.IO.FileAccess]::Read)

[Microsoft.SharePoint.Client.File]::SaveBinaryDirect($ctx,$targeturl , $fs, $true) $fs.Close()

#アップロードしたテンプレートを元に、ソリューションを有効化する

[void][System.Reflection.Assembly]::LoadWithPartialName("Microsoft.SharePoint.Client.Publishing")

$pkg=New-Object Microsoft.SharePoint.Client.Publishing.DesignPackageInfo

$pkg.PackageGuid=[GUID]::Empty

$pkg.PackageName=$templateName

$pkg.MajorVersion=1

$pkg.MinorVersion=0

[Microsoft.SharePoint.Client.Publishing.DesignPackage]::Install($ctx,$ctx.site,$pkg, $targeturl) $ctx.ExecuteQuery()

#前の処理で新しい WSP ファイルが生成されているので、元ファイルを削除する

$web=$ctx.web

$oldfile=$web.GetFileByServerRelativeUrl($targeturl)

$oldfile.DeleteObject() $ctx.ExecuteQuery()

#有効化されたテンプレートを読み込む

$templates=$ctx.site.GetWebTemplates(1041,0)

SPO スクリプト集 - 33 ページ

$ctx.Load($templates)

$ctx.ExecuteQuery()

$t=$templates|?{$\_.Title -eq $templateName}

#テンプレートをサイトに適用する

#テンプレートに指定する名称は以下のような文字列 "{D3E4C48D-1364-40DA-BDF3-7BD88393E719}#CustomTemplate01"

$ctx.web.ApplyWebTemplate($t.Name) $ctx.ExecuteQuery()

#既定のグループを設定しておく

$groups=$web.SiteGroups

$ctx.Load($groups)

$ctx.ExecuteQuery()

$web.AssociatedOwnerGroup=$groups|? Id -eq 3

$web.AssociatedVisitorGroup=$groups|? Id -eq 4

$web.AssociatedMemberGroup=$groups|? Id -eq 5

$web.Update()

$ctx.ExecuteQuery()

SPO スクリプト集 - 34 ページ

## OneDrive for Business 同期クライアントをテナント全体で無効化する (管理シェル)

2015年7月1日 16:59

#SPO 管理シェル接続

Connect-SPO

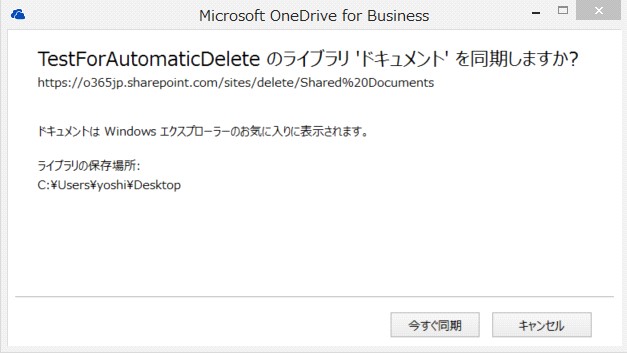
#存在しない空の GUID を持つドメインからの同期のみを許可することで、

#同期クライアントの利用を禁止する 最大 24 時間反映には時間がかかる

$guid=new-object guid

Set-SPOTenantSyncClientRestriction -Enable -DomainGuids $guid

#各ライブラリでの同期設定時



#エラーとなり同期できない



#解除する場合は以下のコマンドを実行 最大 24 時間反映には時間がかかる

Remove-SPOTenantSyncClientRestriction

SPO スクリプト集 - 35 ページ

## 特定のサイト コレクションでカスタム スクリプトの実行を禁止する (管理シェル)

2015年7月7日 13:39

#SPO 管理シェル接続

Connect-SPO

$siteUrl = "[https://xxxx.sharepoint.com/sites/share"](https://xxxx.sharepoint.com/sites/share)

Set-SPOSite $siteUrl -DenyAddAndCustomizePages $true

#カスタム スクリプトの実行を禁止した際の動作については、

#以下を参照のこと

# スクリプト機能のオン/オフの切り替え

# <https://support.office.com/ja-jp/article/HA104182969>

SPO スクリプト集 - 36 ページ