



# EUROPEAN COLLABORATION SUMMIT 2018

MAINZ, GERMANY / MAY 28-30 2018 - [WWW.COLLABSUMMIT.EU](http://WWW.COLLABSUMMIT.EU)

Tony Redmond

USING POWERSHELL TO MANAGE OFFICE 365 GROUPS AND MICROSOFT TEAMS



# DIAMOND, PLATINUM AND GOLD SPONSORS

WEBCON

aws

skybow®

LightningTools

Alight.

**KWizCom**  
KNOWLEDGE WORKER COMPONENTS

**Rencore**

**SysKit**

Sharegate

**Valo**

**K2**

SOLUTIONS  
2SHARE

**THEOBALD**  
SOFTWARE

**NINTEX**

AvePoint

**Beezy**

dox42

**CONVISTA**  
moving ahead

**deroso**  
solutions GmbH  
an FGND group company

**panagenda**

**implexis**  
winning solutions

**ppedv**



# Using PowerShell to manage Office 365 Groups and Microsoft Teams

Tony Redmond  
@12Knocksinna

# Office 365 for IT Pros

Fourth Edition



## Tony Redmond

- Lead author for "Office 365 for IT Pros" eBook
  - <https://practical365.com/ebooks/office-365-for-it-pros/>
- MVP since 2004
- Columnist for Petri.com
  - <https://www.petri.com/category/office/office-365>



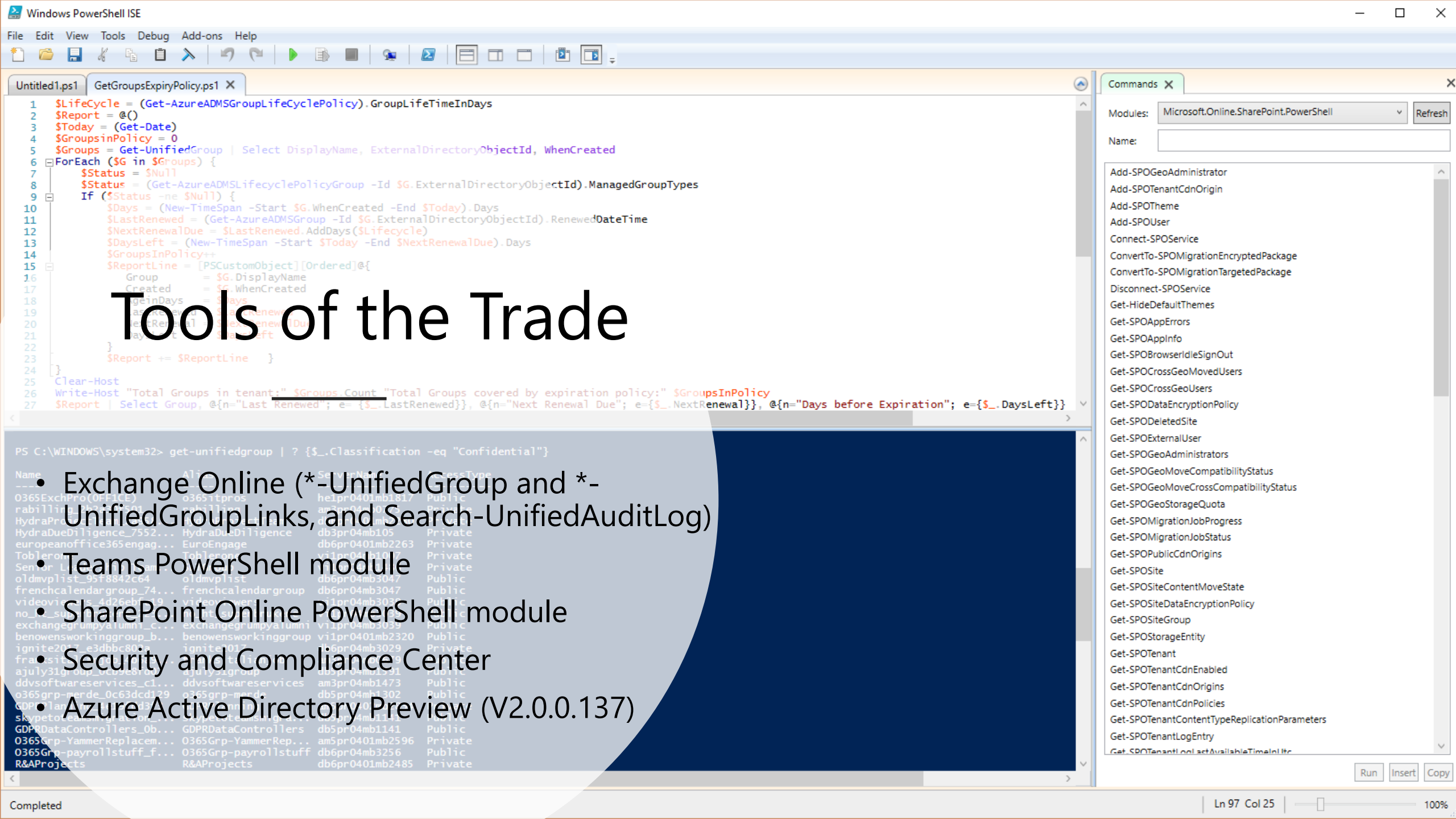
# Context

- Not an Expert class
- **Practical** rather than beautiful
- Focused on getting work done
- You can always improve PowerShell code later

## Office 365 is great, so why do I need PowerShell?

- Admin Tools are imperfect
- Microsoft cannot anticipate every possible administrative task
- PowerShell helps you to understand Office 365
- PowerShell isn't difficult, and it can be fun
- But sometimes, PowerShell is the wrong answer...





# Tools of the Trade

- Exchange Online (\*-UnifiedGroup and \*-UnifiedGroupLinks, and Search-UnifiedAuditLog)
- Teams PowerShell module
- SharePoint Online PowerShell module
- Security and Compliance Center
- Azure Active Directory Preview (V2.0.0.137)

Commands X

Modules: Microsoft.Online.SharePoint.PowerShell Refresh

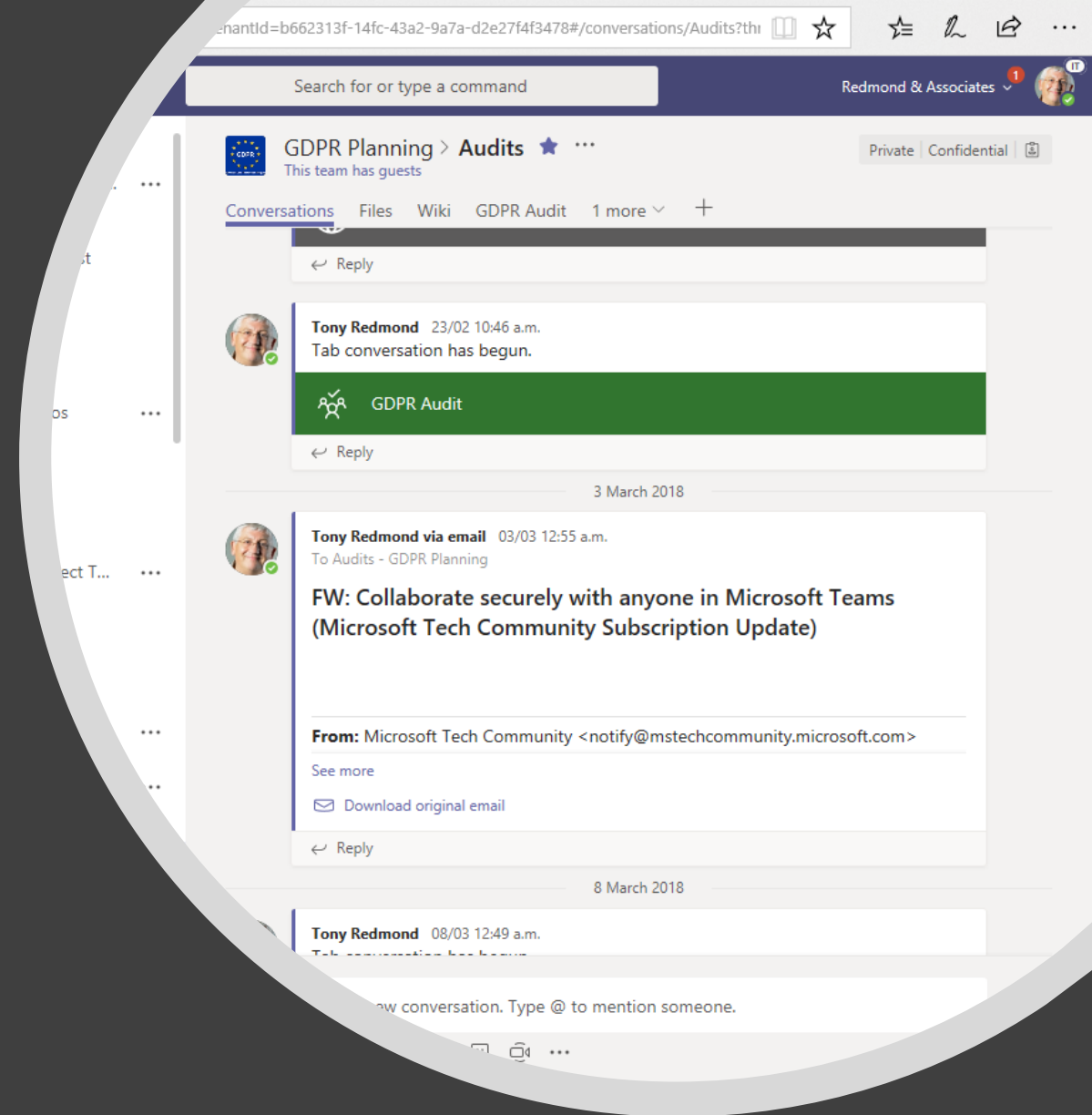
Name:

- Add-SPOGeoAdministrator
- Add-SPOTenantCdnOrigin
- Add-SPOTheme
- Add-SPOUser
- Connect-SPOService
- ConvertTo-SPOMigrationEncryptedPackage
- ConvertTo-SPOMigrationTargetedPackage
- Disconnect-SPOService
- Get-HideDefaultThemes
- Get-SPOAppErrors
- Get-SPOAppInfo
- Get-SPOBrowserIdleSignOut
- Get-SPOCrossGeoMovedUsers
- Get-SPOCrossGeoUsers
- Get-SPODataEncryptionPolicy
- Get-SPODeletedSite
- Get-SPOExternalUser
- Get-SPOGeoAdministrators
- Get-SPOGeoMoveCompatibilityStatus
- Get-SPOGeoMoveCrossCompatibilityStatus
- Get-SPOGeoStorageQuota
- Get-SPOMigrationJobProgress
- Get-SPOMigrationJobStatus
- Get-SPOPublicCdnOrigins
- Get-SPOSite
- Get-SPOSiteContentMoveState
- Get-SPOSiteDataEncryptionPolicy
- Get-SPOSiteGroup
- Get-SPOStorageEntity
- Get-SPOTenant
- Get-SPOTenantCdnEnabled
- Get-SPOTenantCdnOrigins
- Get-SPOTenantCdnPolicies
- Get-SPOTenantContentTypeReplicationParameters
- Get-SPOTenantLogEntry
- Get-SPOTenantLocalLastAvailableTimeInUTC

Run Insert Copy

# A Complex Environment

- Teams depend on Office 365 Groups and share common membership
- Azure Active Directory is the master
- UnifiedGroup cmdlets read EXODS; SharePoint cmdlets read SPODS
- Whenever possible, use Teams cmdlets to update groups used with Teams and Groups cmdlets otherwise
- We can read settings, but not content (conversations)





# Avoid PowerShell Throttling

- Use server-side filters whenever possible
- Select properties to reduce data passing between client and server
- Use Invoke-Command to process data remotely



```
$Groups = (Invoke-Command -Session $Session -ScriptBlock {Get-  
UnifiedGroup -ResultSize Unlimited | Select-Object DisplayName, Alias,  
GroupMemberCount, WhenCreated})
```

```
PS C:\temp> connect-exo
Account Environment TenantId
-----
Administrator@redmondassociates.org AzureCloud b662313f-14fc-43a2-9a7a-d2e27f4f3478

Name : tmp_b3cxcyb.b5q
Path : C:\Users\tonyr\AppData\Local\Temp\tmp_b3cxcyb.b5q
Description : Implicit remoting for https://outlook.office365.com
guid : 6d31bd6d-0dbe-4f5d-b915-f81083b8e459
Version : 1.0
ModuleBase : C:\Users\tonyr\AppData\Local\Temp\tmp_b3cxcyb.b5q
ModuleType : Script
PrivateData : {ImplicitRemoting}
AccessMode : ReadWrite
ExportedAliases : {}
ExportedCmdlets : {}
ExportedFunctions : {[Add-AvailabilityAddressSpace, Add-AvailabilityAdd
Add-DistributionGroupMember], [Add-MailboxFolderPer
Add-MailboxLocation, Add-MailboxLocation]...}
ExportedVariables : {}
NestedModules : {}

Account : Administrator@
Environment : AzureCloud
Tenant : b662313f-14fc-43a2-9a7a-d2e27f4f3478
TenantId : b662313f-14fc-43a2-9a7a-d2e27f4f3478
TenantDomain :
```

# Connecting to PowerShell (otherwise your cmdlets won't work...)

Connect to Exchange Online to use  
\*–UnifiedGroups cmdlets and other Exchange  
cmdlets (like Search-UnifiedAuditLog)

Connect to Teams to use Teams cmdlets

Connect to Azure Active Directory to use Azure AD  
cmdlets

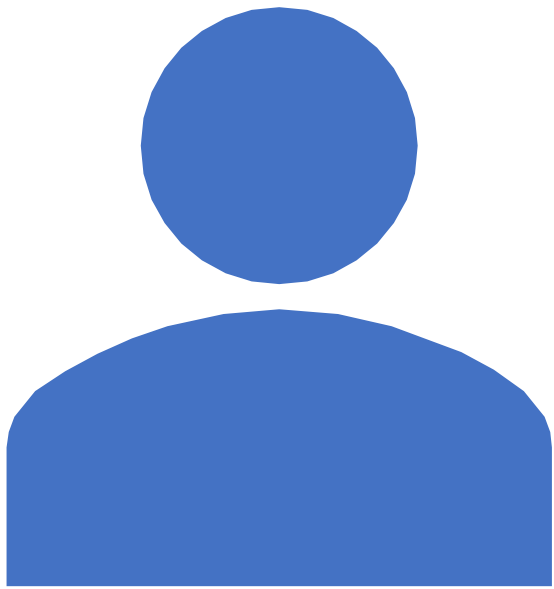
Connect to SharePoint Online to interact with sites

Confused? Use Michel de Rooij's PowerShell profile  
for Office 365

<https://eightwone.com/2015/08/31/connecting-to-office-365exchange/>

# Some Basic Cmdlets

Function	Cmdlets
Create Groups and Teams	New-UnifiedGroup/New-Team
Retrieve settings	Get-UnifiedGroup Get-TeamMemberSettings Get-TeamMessagingSettings Get-TeamGuestSettings Get-TeamFunSettings Get-TeamChannel
Update settings	Set-UnifiedGroup Set-Team*
Access membership	Get-UnifiedGroupLinks [-LinkType Members, Owners] Add-UnifiedGroupLinks Get-TeamUser Add-TeamUser



# Cmdlet Identities

- Groups cmdlets behave like Exchange mailbox cmdlets

```
Get-UnifiedGroup -Identity  
GroupAdmins@Office365ItPros.com
```

- Some Exchange mailbox cmdlets work against group mailboxes
- Teams and Azure AD cmdlets use GUIDs

```
Get-TeamUser -GroupId 72ee570e-3dd8-41d2-bc84-  
7c9eb8024dd4
```

- If you don't know a GUID, ask Groups

```
Get-TeamUser -GroupId (Get-UnifiedGroup -Identity  
ExchangeGoms).ExternalDirectoryObjectId
```

```
Get-AzureADGroupMember -ObjectId (Get-UnifiedGroup  
-Identity ExchangeGoms).ExternalDirectoryObjectId
```

```
Connected to Office 365 as: Administrator@redmondassociates.org Workloads: Azure AD Compliance Center EXO Compliance Center
PS C:\temp> Get-AzureADGroupMember -ObjectId (Get-UnifiedGroup -Identity ExchangeGoms).ExternalDirectoryObjectId
>>
ObjectId                ObjectType
-----
eff4cd58-1bb8-4899-94de-795f656b4a18 User
c814b6e2-d4c2-431d-b82c-e059b152c96c User
53f08764-07d4-418c-8403-a737a8fac7b3 User
c1b58012-68f6-44aa-a838-e8a91f491b1f User
d44088a5-a5ea-47af-b724-cffb12c6ed3e User
7ec3b907-1559-4656-9418-506a2c1bb1c2 User
53f8f8fa-ae0c-49f7-add4-0cb5f1e7b067 User
27fb7fb6-e987-433a-a4d0-3a96072ae384 User
4b478441-3a6b-4b4f-a3c0-68aee6c281c2 User
01b2bcbb-1757-472e-91e2-e24d839eda2d User
fb5dc138-6aa2-48e4-aea7-6718766e0ee9 User
41657dfa-217e-43d1-86c9-c937837348a9 User
c3ac50c4-f328-4177-988b-db10957d5bba User
3f042599-b299-48e2-bca0-dcccd9fbeaa9 User
0366f562-7053-450b-b50b-6ab93568255b User
```

# Demo

Hacking Around with Groups and Teams in PowerShell

# Azure AD Policy for Groups



- Controls whether users can create groups and teams
- Controls whether guest users are supported on a tenant-wide or group-specific basis
- Defines classifications
- Defines naming policy for new groups and teams
- Must be manipulated with PowerShell
- Policy settings in Azure AD settings object

# Creating a new Groups policy

---

```
[PS] C:\> Connect-AzureAD
```

---

```
[PS] C:\> $Policy = Get-  
AzureADDirectorySettingTemplate | ?  
{$_ .DisplayName -eq "Group.Unified"}
```

---

```
[PS] C:\> $Settings =  
$Policy.CreateDirectorySetting()
```

---

```
[PS] C:\> New-  
AzureADDirectorySetting -  
DirectorySetting $Settings
```

# Examining Groups Policy Settings

```
[PS] C:\> Get-AzureADDirectorySetting | ForEach Values
```

Name	Value
-----	-----
CustomBlockedWordsList	
ClassificationDescriptions	General Usage:Anyone can access,External Access:Available outside the company,Internal Only:Must not be shared with external people,Confidential:Can only be disclosed with management permission
DefaultClassification	General Usage
PrefixSuffixNamingRequirement	
AllowGuestsToBeGroupOwner	False
AllowGuestsToAccessGroups	True
GuestUsageGuidelinesUrl	
GroupCreationAllowedGroupId	A3c13e4d-7083-4448-9224-287f10f23e10
AllowToAddGuests	True
UsageGuidelinesUrl	<a href="http://office365itpros.com/GroupGuidelines.html">http://office365itpros.com/GroupGuidelines.html</a>
ClassificationList	General Usage,External Access,Internal Only,Confidential
EnableGroupCreation	False



# Updating Group Policy Setting for Group Creation

GUID for group specifying people who can create groups

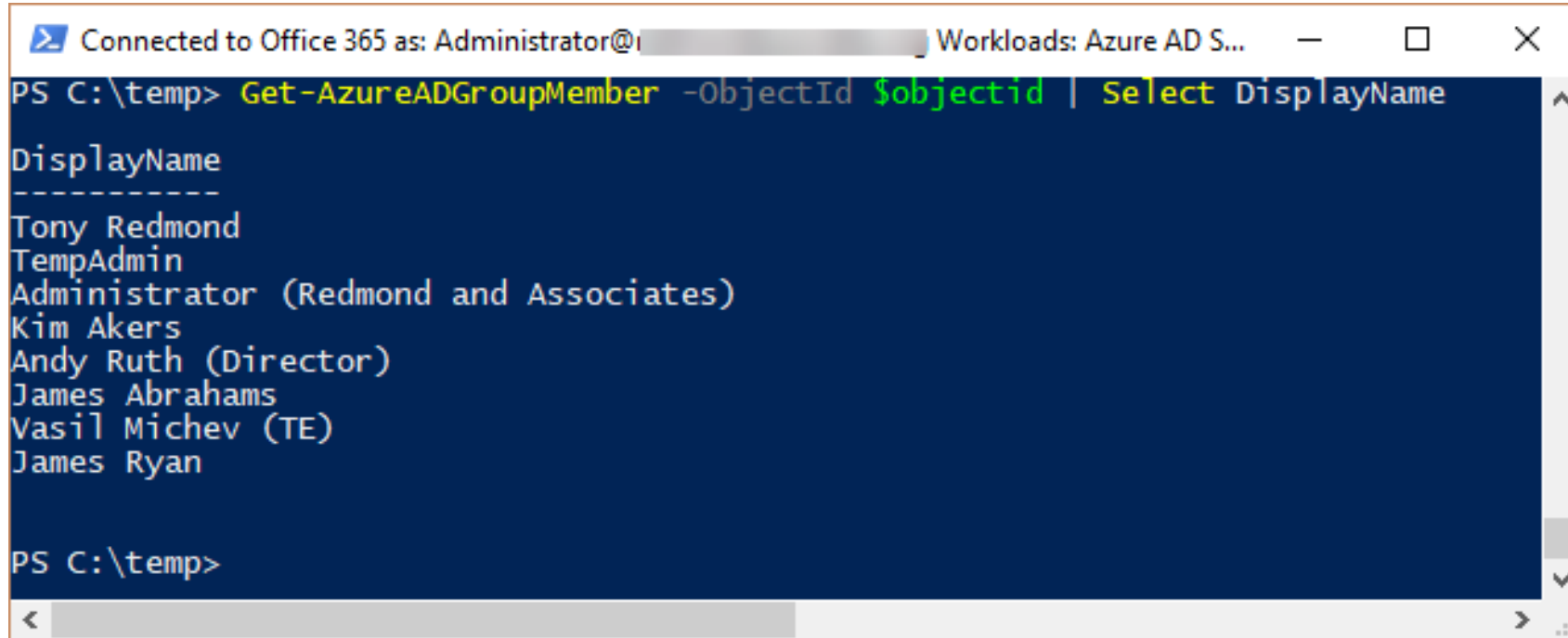
```
[PS] C:\> $ObjectId = (Get-AzureADGroup  
-SearchString  
GroupCreationControl).ObjectId
```

```
[PS] C:\> $Settings = Get-Retrieve current policy settings  
AzureADDirectorySetting | ?  
{$_ .DisplayName -eq "Group.Unified"}
```

```
[PS] C:\>  
$Settings["EnableGroupCreation"] =  
"False"  
Block general group creation
```

```
[PS] C:\>  
$Settings["GroupCreationAllowedGroupId"]  
= $ObjectId  
Except for the group pointed to in $ObjectId
```

```
[PS] C:\> Set-AzureADDirectorySetting  
-Id $Settings.Id -DirectorySetting  
$Settings  
And update the policy...
```



The screenshot shows a PowerShell terminal window with the following content:

```
Connected to Office 365 as: Administrator@i... Workloads: Azure AD S...
PS C:\temp> Get-AzureADGroupMember -ObjectId $Objectid | Select DisplayName
DisplayName
-----
Tony Redmond
TempAdmin
Administrator (Redmond and Associates)
Kim Akers
Andy Ruth (Director)
James Abrahams
Vasil Michev (TE)
James Ryan
PS C:\temp>
```

# Who Can Create Groups and Teams?

```
Get-AzureADGroupMember -ObjectId $ObjectId | Format-Table  
    DisplayName, UserPrincipalName
```

# Some PowerShell Projects with Groups and Teams

Just to get your creative juices going...



# Project: Find Inactive Groups and Teams

- Some groups and teams are created and then go into a state of disuse after a short period
- Goal is to find these groups and teams and then remove or archive them
- No functionality available in standard product
- To detect latest activity:
  - Look for SharePoint activity in Office 365 Audit Log
  - Look for conversations in group inbox
  - Look for Teams compliance records
- Generate nice report

Description	MailboxStatus	TeamEnabled	LastChat	NumberChats	LastConversation	NumberConversations	SPOActivity	SPOStatus	NumberWarnings
6	Low number of conversations found	False	No chats	0	05/01/2018 11:50:44	3	No SPO activity detected in the last 90 days	Normal	2
	Normal	False	No chats	0	12/05/2015 17:42:13	3	No SPO activity detected in the last 90 days	Document library never created	3
A group for Board Members	Normal	False	No chats	0	08/05/2015 23:22:11	2	No SPO activity detected in the last 90 days	Document library never created	3
test	Normal	False	No chats	0	09/05/2015 16:52:55	2	No SPO activity detected in the last 90 days	Document library never created	3
Test Exchange Group	Normal	False	No chats	0	07/05/2016 16:28:36	300	No SPO activity detected in the last 90 days	Document library never created	3
Corporate Banking Team	Normal	True	16/03/2017 18:02:37	1	12/05/2015 16:15:22	8	No SPO activity detected in the last 90 days	Normal	2
We're off to see the Wizard, but we need to climb some mountains first	Normal	False	No chats	0	11/05/2015 21:13:43	4	No SPO activity detected in the last 90 days	Normal	2
	Normal	False	No chats	0	27/01/2016 11:17:42	5	No SPO activity detected in the last 90 days	Normal	2
Interesting places to visit	Normal	False	No chats	0	12/05/2015 17:41:26	2	No SPO activity detected in the last 90 days	Document library never created	3
Patel's best places to visit	Normal	False	No chats	0	14/05/2015 16:21:21	4	No SPO activity detected in the last 90 days	Document library never created	3
Develop corporate plans	Normal	True	06/04/2018 16:58:00	32392	14/02/2018 10:18:22	22	Document library in use	Normal	0
Budget planning co-ordination team for	Low number of conversations found	True	06/04/2018 03:09:36	128	16/11/2017 12:21:44	3	Document library in use	Normal	1
Information about the Exchange track	Normal	False	No chats	0	11/10/2016 13:28:18	38	Document library in use	Normal	1
	Normal	False	No chats	0	14/05/2015 19:34:38	2280	No SPO activity detected in the last 90 days	Normal	2
	Normal	False	No chats	0	14/05/2015 11:14:56	18	No SPO activity detected in the last 90 days	Normal	
			No chats	0	20/06/2017 10:37:55	18	Doc		

```

Write-Host "Archiving" $AGroup.DisplayName -ForegroundColor Yellow
} Else {
Write-Host $CheckGroup "group not found - terminating"
Return }

# Get lists of current owners and members
$CurrentOwners = (Get-UnifiedGroupLinks -Identity $AGroup.Alias -LinkType Owners | Select Name)
$CurrentMembers = (Get-UnifiedGroupLinks -Identity $AGroup.Alias -LinkType Members | Select Name)
# Add a new owner - this is the address of the account that will continue to access the group
$AdminAccount = "Administrator"
Add-UnifiedGroupLinks -Identity $AGroup.Alias -LinkType Members -Links $AdminAccount
Add-UnifiedGroupLinks -Identity $AGroup.Alias -LinkType Owners -Links $AdminAccount
# Remove the other members and owners
ForEach ($O in $CurrentOwners) {
Remove-UnifiedGroupLinks -Identity $AGroup.Alias -LinkType Owners -Links $O.Name -Confirm:$False}
ForEach ($M in $CurrentMembers) {
Remove-UnifiedGroupLinks -Identity $AGroup.Alias -LinkType Members -Links $M.Name -Confirm:$False}
# Create SMTP Address for the archived group
$OldSmtpAddress = $AGroup.PrimarySmtpAddress -Split "@"
$NewSmtpAddress = $OldSmtpAddress[0] + "_archived" + "@" + $OldSmtpAddress[1]
$AddressRemove = "smtp:"+$AGroup.PrimarySmtpAddress
$ArchiveInfo = "Archived on " + (Get-Date) + " by " + $o365cred.username
# Update Group properties
Set-UnifiedGroup -Identity $AGroup.Alias -AccessType Private -RequireSenderAuthenticationEnabled $True -Hidden $True -CustomAttribute1 "Archived" -CustomAttribute2 $ArchiveInfo -PrimarySmtpAddress $NewSmtpAddress
Set-UnifiedGroup -Identity $AGroup.Alias -EmailAddresses @{remove=$AddressRemove}

Write-Host $AGroup.DisplayName "is now archived and" $AdminAccount "is the new group owner"

```

# Project: Archive Obsolete Groups and Teams

- If we find obsolete groups or teams, we might want to keep them in an inactive state, just in case they are needed again
- Remove current membership
- Add new owner
- Replace SMTP address
- Set custom attribute to make it easy to find archived groups

in tenant: 159 Total Groups covered by expiration policy: 22

# Project: Find Groups About to Expire

Last Renewed	Next Renewal Due	Days before Expiration
24/05/2017	14/03/2020	43
12/03/2018	31/03/2020	72
14/03/2018	01/04/2020	72
24/05/2017	19/05/2019	43
16/05/2016	05/06/2018	5
28/09/2016	18/10/2018	19
02/11/2016	22/11/2018	23
29/11/2017	11/12/2019	62
23/01/2017	12/02/2019	31
01/03/2017	02/03/2019	34
14/04/2017	04/05/2019	39
25/05/2017	14/06/2019	43
26/05/2017	15/06/2019	43
20/06/2017	10/07/2019	46
21/06/2017	11/07/2019	46
28/06/2017	18/07/2019	46
03/07/2017	15/07/2019	47
30/07/2017	19/08/2019	49
31/07/2017	20/08/2019	50
31/07/2017	20/08/2019	50
08/08/2017	28/08/2019	50
03/10/2017	23/10/2019	56

- Notifications generated by the Groups expiration policy tell owners when their groups expire
- Tenant admins might like to know too, just in case an important group or team is removed in error
  - Look for groups in scope of expiration policy
  - Calculate next expiration date
  - Sort report by days left before expiry

```
>> If ($R.TargetType -eq "Guest") {  
>> Write-Host "Document" $R.Document "shared with" $R.SharedWith "at" $R.TimeStamp }  
>> }
```

Document Data spillage and GDPR.docx shared with tredmond\_yandex.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-30T16:58:13  
Document BA Expenses JYX88X.pdf shared with tredmond\_yandex.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-30T10:33:26  
Document Compliance Framework\_customer guidance.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T  
Document Compliance Framework\_customer guidance.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T  
Document The Complete Guide to Eradicating PSTs.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T  
Document The Complete Guide to Eradicating PSTs.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T  
Document Migration Datasheet.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:40:31  
Document Migration Datasheet.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:40:31  
Document Radisson Blu.docx shared with flayosc\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:38:30  
Document Radisson Blu.docx shared with flayosc\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:38:30  
Document Radisson Blu.docx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:38:30  
Document Radisson Blu.docx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:38:30  
Document Clearing Up Office 365 Guest Users.docx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:  
Document Complete Guide to Eradicating Public Folders.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:  
Document BA Expenses JYX88X.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:24:18  
Document BA Expenses JYX88X.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:24:18  
Document Complete Guide to Eradicating Public Folders.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-  
Document Complete Guide to Eradicating Public Folders.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-  
Document Fish Bay.docx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:12:19  
Document Fish Bay.docx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T20:12:19  
Document Microsoft Makes Major Changes More Transparent for Office 365.docx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicr  
Document Microsoft Makes Major Changes More Transparent for Office 365.docx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicr  
Document Searching Employee Mailboxes.docx shared with tredmond\_yandex.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T12:18:30  
Document Updated GDPR Deck - Board Meeting 14.03.2018.pptx shared with tredmond\_yandex.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-2  
Document Updated GDPR Deck - Board Meeting 14.03.2018.pptx shared with tredmond\_yandex.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-2  
Document Board Meeting Minutes QT 12.01.18.docx shared with tredmond\_yandex.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T10:31:21  
Document Board Meeting Minutes QT 12.01.18.docx shared with tredmond\_yandex.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T10:31:21  
Document 01-01-2017\_NDA\_9501.pdf shared with tredmond\_yandex.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T10:25:09  
Document 01-01-2017\_NDA\_9501.pdf shared with tredmond\_yandex.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-29T10:25:09  
Document 01-01-2017\_NDA\_9501.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-28T19:07:22  
Document 01-01-2017\_NDA\_9501.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-28T18:10:51  
Document AAD synchronization to Exchange Online.pptx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-28  
Document AAD synchronization to Exchange Online.pptx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-28  
Document Searching Employee Mailboxes.docx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-28T15:4  
Document Board Meeting Minutes QT 12.01.18.docx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-28T15:4  
Document 01-01-2017\_NDA\_9501.pdf shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-28T15:34:28  
Document Searching Employee Mailboxes.docx shared with tredmond\_yandex.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-28T15:25:40  
Document Searching Employee Mailboxes.docx shared with tredmond\_yandex.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-28T15:25:40  
Document Board Meeting Minutes QT 12.01.18.docx shared with draguignan24\_outlook.com#ext#@redmondassociates.onmicrosoft.com at 2018-03-28T15:1  
PS C:\temp>

# Project: Who Shared Documents?

- Groups and Teams make SharePoint document libraries very accessible
- Users can share documents easily, including with users outside the tenant
- Office 365 audit records give us a way to check who shares documents

# Unified Audit Log

---

- Ingests data from multiple Office 365 sources and normalizes audit records
- Retained for 90 days (180 days with Advanced Security Management, or more with ISV products)
- Rich source for investigations about what happens inside Office 365

```
$Records = (Search-UnifiedAuditLog -StartDate 28-Mar-2018 -EndDate 28-Mar-2018 -EventTypes "SharingInvitationCreated", "SharingInvitationAccepted")
If ($Records.Count -eq 0) {
    Write-Host "No SharePoint sharing records found." }
Else {
    Write-Host "Processing" $Records.Count " SharePoint sharing records"
    $Report = @()
    ForEach ($Rec in $Records) {
        $AuditData = ConvertFrom-Json $Rec.Auditdata
        Switch ($AuditData.TargetUserOrGroupType)
        {
            "Member" { $SharedType = "Tenant User" }
            "SharePointGroup" { $SharedType = "SPO Sharing Link" }
            "Guest" { $SharedType = "Guest User" }
        }
        $ReportLine = [PSCustomObject][Ordered]@{
            TimeStamp = $AuditData.CreationTime
            User = $Rec.UserIds
            Action = $AuditData.Operation
            Workload = $AuditData.Workload
            ObjectID = $AuditData.ObjectId
            SourceFileName = $AuditData.SourceFileName
            TargetUserOrGroupName = $AuditData.TargetUserOrGroupName
            SharedType = $SharedType
            ItemType = $AuditData.ItemType
            EventData = $AuditData.EventData
            TargetUserOrGroupType = $AuditData.TargetUserOrGroupType
        }
    }
}
```



```

$Groups = (Get-UnifiedGroup -Filter {GroupExternalMemberCount -gt 0} | Select Alias, DisplayName, SharePointSiteURL, Gr
If ($Groups.Count -gt 0) {
  Write-Host "Processing" $Groups.Count "groups with guest members"
  $Report = @()
  $NumExt = 0
  $LargestGroup = $Null
  $LargestGroupNum = 0
  ForEach ($G in $Groups) {
    Write-Host "Processing" $G.DisplayName
    $Users = Get-UnifiedGroupLinks -Identity $G.Alias -LinkType Members
    ForEach ($U in $Users) {
      If ($U.Name -Match "#EXT#" -and $U.Name -NotLike "*teams.ms*") {
        $NumExt++
        $CheckName = $U.Name + "@redmondassociates.onmicrosoft.com"
        $User = (Get-AzureADUser -ObjectId $CheckName).DisplayName
        $ReportLine = [PSCustomObject][Ordered]@{
          Email      = $U.Name
          User       = $User
          Group      = $G.DisplayName
          Site       = $G.SharePointSiteURL }
        $Report += $ReportLine }
      } |
    If ($G.GroupExternalMemberCount -gt $LargestGroupNum) {
      $LargestGroupNum = $G.GroupExternalMemberCount
      $LargestGroup = $G.DisplayName}
  }
  Write-Host $NumExt "guest user memberships found in" $Groups.Count "groups"
  Write-Host "Largest external group is" $LargestGroup "with" $LargestGroupNum "guests"
}

```

# Project: Report Guest Users

- It's easy for Group Owners to add guests to Teams and Groups, but do you know who the guests are and where they come from?
- It's a bad idea to put very confidential information in a group that has guests, unless you intend to share that way

```
Connected to Office 365 as: Administrator@redmondassociates.org Workloads: Azure AD EXO
Checking Dominik Hoefling (MVP)
Last connection for Dominik Hoefling (MVP) on 05/04/2018 20:24:21 as TeamsSessionStarted
Checking Gustavo Velez (MVP)
No connections found in the last 90 days for Gustavo Velez (MVP) created on 11/03/2018 20:51:20
Email traffic found for Gustavo Velez (MVP) at 06/04/2018 12:58:37
Checking Theresa Miller
Last connection for Theresa Miller on 16/03/2018 23:56:11 as TeamsSessionStarted
Checking Michael Van Horenbeeck
Last connection for Michael Van Horenbeeck on 20/03/2018 11:48:17 as TeamsSessionStarted
Checking Thierry DEMAN (MS MVP)
Last connection for Thierry DEMAN (MS MVP) on 05/04/2018 07:33:52 as TeamsSessionStarted
Checking Andy David (SAS MVP)
No connections found in the last 90 days for Andy David (SAS MVP) created on 25/03/2018 13:47:16
Checking Flayosc Worker
Last connection for Flayosc Worker on 04/04/2018 17:15:51 as UserLoggedIn
Checking jcgonzalezmartin1978@hotmail.com
No connections found in the last 90 days for jcgonzalezmartin1978@hotmail.com created on 02/04/2018 17:04:45
Email traffic found for jcgonzalezmartin1978@hotmail.com at 06/04/2018 17:04:45
Checking Juan Carlos González Martín (MVP)
Last connection for Juan Carlos González Martín (MVP) on 05/04/2018 18:54:23 as TeamsSessionStarted

Statistics
-----
Guest Accounts          124
Active Guests           77
Active on Teams and SPO 37
Active on Email         40
InActive Guests         47
PS C:\temp>
```

# Project: Last Time Guest Users Connected

- Teams, Office 365 Groups, SharePoint Online, and Planner use Azure B2B Collaboration for guest user access
- How do you know whether guests actually connect and use resources in your tenant?

# Things to Remember

Lots of PowerShell examples and advice available on the internet

Never download and run code without checking it first

You don't have to be an expert to screw things up...

Thanks!

