

```
1 <#
2 .SYNOPSIS
3   The purpose of this program is to export mailbox information from the WVU main
   campus Office 365 tenant.
4
5 .DESCRIPTION
6   This file is used to export mailbox information from the WestVirginiaUniversity
   Office 365 tenant.
7   Mailboxes that are selected for export must meet the following requirements.
8   1. The mailbox must be a user mailbox.
9   2. Primary SMTP Address is either @mail.wvu.edu or @wvu.edu.
10  3. Not hidden in the main campus address book
11  4. Contain a first and last name
12  5. Gordon Gee - For some reason, his display name is correct, but he has no
   first/last name in the address book. By rule 3, his account would normally not be
   exported
13  6. Does not contain the word "test" in the name
14
15   There are some other minor clean up to account for discrepancies in the main
   campus address book as well.
16
17 .OUTPUTS
18   This file generates the following output:
19   1. Export main campus email addresses which will be imported (via a separate
   script) the the HSC tenant.
20   2. Create file to import SIP addresses into the HSC tenant.
21
22 .PARAMETER MainCampusMailboxFile
23   This is the main output file which will be imported.
24
25 .PARAMETER SIPFile
26   This is the file containing the SIP addresses which will also be imported.
27
28 .PARAMETER Testing
29   Using this parameter will stop file processing if an error occurs.
30
31 .NOTES
32   Author: Jeff Brusoe
33   Last Updated: August 18, 2020
34 #>
35
36 [CmdletBinding()]
37 [Diagnostics.CodeAnalysis.SuppressMessageAttribute("PSAvoidTrailingWhiteSpace","",Justification = "Not relevant")]
38 [Diagnostics.CodeAnalysis.SuppressMessageAttribute("PSAvoidUsingCmdletAliases","",Justification = "Only MS Aliases are Being Used")]
39 param (
40   [string]$MainCampusMailboxFile = "$PSScriptRoot\Logs\" + (Get-Date -Format yyyy-MM-dd-HH-mm) + "-MainCampusExchangeExport.csv",
41   [string]$SIPFile = "$PSScriptRoot\Logs\" + (Get-Date -Format yyyy-MM-dd-HH-mm) + "-SIP-MC.csv",
42   [switch]$Testing
43 )
44
45 $Error.Clear()
46
47 Set-HSCEnvironment
48 Connect-MainCampusExchangeOnline
49
50 #Initialization of output files
```

```
51 New-Item -path $SIPFile -type "file" -force
52 New-Item -path $MainCampusMailboxFile -type "file" -force
53
54 #Get main campus mailboxes
55 Write-Output "Generating list of main campus mailboxes"
56
57 try
58 {
59     $AdditionalProperties =
60     "FirstName","LastName","Title","Department","Phone","WhenChanged","CustomAttributes8",
61     "HiddenFromAddressListsEnabled"
62     $Mailboxes = Get-EXORecipient -Resultsize Unlimited -ErrorAction Stop -Properties
63     $AdditionalProperties |
64     where {(($_.PrimarySMTPAddress -like "*@mail.wvu.edu") -OR
65     ($_.PrimarySMTPAddress -like "*@wvu.edu")) -AND
66     ($_.HiddenFromAddressListsEnabled -eq $false) -AND
67     ($_.Title -notlike "*Retiree*")} }
68 }
69 catch
70 {
71     Write-Warning "Unable to query main campus mailboxes. Program is exiting."
72     Invoke-HSCEExitCommand
73 }
74
75 Write-Output "`nBeginning to process main campus mailboxes"
76
77 $MailboxCount = 1
78 foreach ($Mailbox in $Mailboxes)
79 {
80     if ($Error.Count -gt 0 -AND $Testing)
81     {
82         Write-Warning "Stopping due to error. Program is exiting."
83         $Error | FL
84
85         Invoke-HSCEExitCommand
86     }
87
88     Write-Output $("`n`nCurrent Mailbox: " + $Mailbox.PrimarySMTPAddress)
89     Write-Output $("Mailbox Count: $MailboxCount")
90
91     if ($Mailbox.PrimarySMTPAddress -eq "Gordon.gee@mail.wvu.edu")
92     {
93         #Make sure to include this address
94         #Currently (April 25, 2016) there is no first or last name entry for him, but
95         his display name is correct.
96         Write-Output "Adding Gordon Gee to export file."
97         $Mailbox | select
98         DisplayName,FirstName,LastName,Title,Department,Phone,PrimarySMTPAddress,Alias,When
99         Changed,customAttribute8,hiddenfromaddresslistsenabled | Export-Csv
100         $MainCampusMailboxFile -Append
101
102         $MailboxCount++
103     }
104     elseif (([string]::IsNullOrEmpty($Mailbox.LastName)) -OR
105     ([string]::IsNullOrEmpty($Mailbox.FirstName)))
106     {
107         if ([string]::IsNullOrEmpty($Mailbox.LastName))
108         {
109             Write-Warning "Skipping mailbox because of missing last name"
110         }
111     }
112 }
```

```
102     else
103     {
104         Write-Warning "Skipping mailbox because of missing first name"
105     }
106 }
107 elseif ($Mailbox.DisplayName.ToLower().IndexOf("test") -ge 0)
108 {
109     Write-Warning "Test account - Skipping"
110 }
111 else
112 {
113     Write-Output "Exporting user information"
114     $Mailbox | select
115     DisplayName,FirstName,LastName,Title,Department,Phone,PrimarySMTPAddress,Alias,
116     WhenChanged,customattribute8,HiddenFromAddressListsEnabled | Export-Csv
117     $MainCampusMailboxFile -Append
118
119     $MailboxCount++
120 }
121
122 #Examine email addresses to see if any are SIP ones.
123 Write-Output "Looking for SIP Addresses"
124
125 $EmailAddresses = $Mailbox.EmailAddresses
126
127 foreach ($EmailAddress in $EmailAddresses)
128 {
129     "EmailAddress: " + $EmailAddress
130
131     if ($EmailAddress.IndexOf("SIP") -ge 0)
132     {
133         Write-Output $("Writing " + $EmailAddress + " to SIP file")
134
135         $Mailbox | select primarySMTPAddress,Alias,@{Label = 'SIPAddress';Expression=
136         {$EmailAddress}} | Export-Csv $SIPFile -Append
137     }
138 }
139 }
140
141 Invoke-HSCEExitCommand
```