Exportieren von AD Gruppen mit Powershell

Sometimes you want to export Active Directory group members to CSV file. For example, you must export all AD group members, only a particular OU or multiple groups. In this article, you will learn how to export AD group membership to CSV file with PowerShell.

Quelle: https://www.alitajran.com/export-ad-group-members-powershell/

Inhalt

```
# Get year and month for CSV export file
$DateTime = Get-Date -f "yyyyMMddhhmm"

# Set CSV file name
$CSVFile = "C:\temp\ADGroups_" + $DateTime + ".csv"

# Set distinguishedName as searchbase, you can use one DN or multiple DNs
# Or use the root domain like DC=exoip,DC=local
$DNs = @(
    "DC=exoip,DC=local"
)

# Create empty array for CSV data
$CSVOutput = @()

# Create empty array for AD groups
$ADGroups = @()

# Loop through DNs
foreach ($DN in $DNs) {

    # Add every DN to AD groups
    $ADGroups += Get-ADGroup -Filter * -SearchBase $DN
```

```
}

# Set progress bar variables
$i = 0
$tot = $ADGroups.count

foreach ($ADGroup in $ADGroups) {

    # Set up progress bar
    $i++
    $status = "{0:N0}" -f ($i / $tot * 100)
    Write-Progress -Activity "Exporting AD Groups" -status "Processing Group $i of $tot :
$status% Completed" -PercentComplete ($i / $tot * 100)

    # Ensure Members variable is empty
    $Members = ""

    # Get group members which are also groups and add to string
    $MembersArr = (Get-ADGroup -filter { Name -eq $ADGroup.Name } -Properties Members
).Members | Get-ADObject | select Name, objectClass, distinguishedName
    if ($MembersArr) {
        foreach ($Member in $MembersArr) {
            $MemDN = $Member.distinguishedName
            $UserObj = Get-ADUser -filter { DistinguishedName -eq $MemDN }
            if ($UserObj.Enabled -eq $False) {
                continue
            }
            if ($Member.objectClass -eq "user") {
                $Members = $Members + ",U-" + $Member.Name
            }
            elseif ($Member.objectClass -eq "contact") {
                $Members = $Members + ",C-" + $Member.Name
            }
            else {
                $Members = $Members + ",G-" + $Member.Name
            }
        }
        # Check for members to avoid error for empty groups
        if ($Members) {
            $Members = $Members.Substring(1, ($Members.Length) - 1)
        }
```

```
    }

    # Set up hash table and add values
    $HashTab = $null
    $HashTab = [ordered]@{
        "Name"     = $ADGroup.Name
        "Category" = $ADGroup.GroupCategory
        "Scope"    = $ADGroup.GroupScope
        "Members"  = $Members
    }

    # Add hash table to CSV data array
    $CSVOutput += New-Object PSObject -Property $HashTab
}

# Export report to CSV file
$CSVOutput | Sort-Object Name | Export-Csv -Encoding UTF8 -Path $CSVFile -NoTypeInformation #-
Delimiter ";"
```

Revision #1
Created 2024-07-23 18:15:23 UTC by CoreStream Group
Updated 2024-07-23 18:16:12 UTC by CoreStream Group