

Exchange Online

- [Verteilergruppen per PowerShell in Exchange Online importieren](#)

Verteilergruppen per PowerShell in Exchange Online importieren

Mit dem PowerShell-Skript **Import-Verteilergruppen.ps1** können mehrere E-Mail-Verteilergruppen automatisiert in Microsoft 365 beziehungsweise Exchange Online angelegt und mit Mitgliedern befüllt werden.

Das Skript verwendet eine CSV-Datei als Datenquelle. Es überprüft vor dem Anlegen einer Verteilergruppe, ob diese bereits vorhanden ist, und verhindert doppelte Mitgliedschaften.

Funktionen

- Automatisches Erstellen von Verteilergruppen
- Zuweisen der primären E-Mail-Adresse
- Hinzufügen mehrerer Mitglieder pro Verteilergruppe
- Überprüfung bereits vorhandener Gruppen
- Vermeidung doppelter Mitgliedschaften
- Ausgabe von Statusmeldungen und Fehlern in PowerShell

Hinweis: Das Skript erstellt klassische Exchange-Verteilergruppen, keine Microsoft-365-Gruppen oder dynamischen Verteilergruppen.

2. Voraussetzungen

Für die Ausführung werden folgende Komponenten benötigt:

- Windows PowerShell 5.1 oder PowerShell 7
- Internetverbindung zu Microsoft 365
- Installiertes ExchangeOnlineManagement-Modul
- Ein Microsoft-365-Konto mit ausreichenden Exchange-Berechtigungen zur Verwaltung von Verteilergruppen
- CSV-Datei mit den zu importierenden Gruppen und Mitgliedern
- PowerShell-Skript `Import-Verteilergruppen.ps1`

Die Benutzer beziehungsweise Empfänger, die als Mitglieder hinzugefügt werden sollen, müssen bereits in Exchange Online bekannt sein.

3. Exchange Online PowerShell vorbereiten

Schritt 1: PowerShell starten

Öffnen Sie PowerShell auf dem Administrationscomputer.

Schritt 2: ExchangeOnlineManagement installieren

Falls das Modul noch nicht installiert ist:

```
Install-Module ExchangeOnlineManagement -Scope CurrentUser
```

Schritt 3: Verbindung zu Exchange Online herstellen

```
Import-Module ExchangeOnlineManagement
```

```
Connect-ExchangeOnline -UserPrincipalName admin@firma.de
```

Ersetzen Sie `admin@firma.de` durch das entsprechende Administratorkonto.

Die Anmeldung erfolgt über die Microsoft-Anmeldeseite, gegebenenfalls einschließlich Multifaktor-Authentifizierung (MFA).

4. CSV-Datei vorbereiten

Das Skript erwartet eine Datei mit dem Namen:

```
verteilergruppen.csv
```

Diese muss folgende Spalten enthalten:

Spaltenname	Beschreibung
GroupName	Name der Verteilergruppe
PrimarySmtpAddress	Primäre E-Mail-Adresse der Gruppe
Member	E-Mail-Adresse beziehungsweise Empfängererkennung des Mitglieds

Beispiel einer CSV-Datei

```

GroupName,PrimarySmtpAddress,Member
IT-Team,it-team@firma.de,max@firma.de
IT-Team,it-team@firma.de,lisa@firma.de
Vertrieb,vertrieb@firma.de,tom@firma.de
Vertrieb,vertrieb@firma.de,anna@firma.de
Support,support@firma.de,service@firma.de

```

Im Beispiel werden drei Verteilergruppen angelegt:

- IT-Team mit zwei Mitgliedern
- Vertrieb mit zwei Mitgliedern
- Support mit einem Mitglied

Wichtig: Die Spaltennamen müssen exakt mit den Angaben im Beispiel übereinstimmen.

Das Skript verwendet standardmäßig ein Komma als CSV-Trennzeichen. Falls die Datei aus deutschem Excel exportiert wurde und Semikolons verwendet, muss die Importzeile angepasst werden:

```
$data = Import-Csv ".\verteilergruppen.csv" -Delimiter ";"
```

Verwenden Sie für Gruppennamen möglichst einfache, eindeutige Bezeichnungen ohne problematische Sonderzeichen, da das Skript den Gruppennamen gleichzeitig als Exchange-Alias verwendet.

5. Dateien vorbereiten

Legen Sie auf dem Administrationscomputer beispielsweise folgenden Ordner an:

```
C:\Scripts\Verteilergruppen
```

Speichern Sie beide Dateien dort:

```
C:\Scripts\Verteilerguppen\  
|  
├─ Import-Verteilerguppen.ps1  
└─ verteilerguppen.csv
```

Das Skript liest die CSV-Datei aus dem aktuellen PowerShell-Arbeitsverzeichnis. Deshalb sollte PowerShell vor der Ausführung in diesen Ordner wechseln.

6. Skript ausführen

Schritt 1: Zum Verzeichnis wechseln

```
cd C:\Scripts\Verteilerguppen
```

Schritt 2: Verbindung überprüfen

```
Get-DistributionGroup -ResultSize 5
```

Wenn vorhandene Verteilergruppen angezeigt werden, funktioniert die Exchange-Online-Verbindung grundsätzlich.

Schritt 3: CSV-Datei überprüfen

```
Import-Csv ".\verteilergruppen.csv" |  
Format-Table
```

Kontrollieren Sie, ob Gruppenname, Gruppenadresse und Mitglieder korrekt eingelesen werden.

Schritt 4: Import starten

```
.\Import-Verteilerguppen.ps1
```

Falls die lokale Ausführungsrichtlinie das Skript blockiert, können Sie für die aktuelle PowerShell-Sitzung bei Bedarf folgende Einstellung verwenden, sofern Ihre Organisationsrichtlinien dies erlauben:

```
Set-ExecutionPolicy -Scope Process -ExecutionPolicy RemoteSigned
```

Danach das Skript erneut starten.

7. Was passiert während des Imports?

Das Skript verarbeitet die CSV-Datei gruppenweise.

Schritt 1 - Gruppe überprüfen

Zunächst wird geprüft, ob eine Verteilergruppe mit der angegebenen primären E-Mail-Adresse bereits existiert.

Schritt 2 - Gruppe anlegen

Existiert die Gruppe nicht, wird sie angelegt. Bereits vorhandene Gruppen werden nicht erneut erstellt.

Schritt 3 - Mitglieder überprüfen

Für jedes Mitglied wird geprüft, ob es bereits in der entsprechenden Gruppe vorhanden ist.

Schritt 4 - Mitglieder hinzufügen

Fehlende Mitglieder werden zur Verteilergruppe hinzugefügt.

Schritt 5 - Status ausgeben

Während des Imports erscheinen farbige Statusmeldungen.

Farbe	Bedeutung
Grün	Verteilergruppe erfolgreich erstellt
Gelb	Gruppe oder Mitglied bereits vorhanden
Cyan	Mitglied erfolgreich hinzugefügt
Warnmeldung	Fehler beim Hinzufügen eines Mitglieds

Beispielausgabe

```
Gruppe erstellt: it-team@firma.de
Mitglied hinzugefügt: max@firma.de
Mitglied hinzugefügt: lisa@firma.de

Gruppe existiert bereits: vertrieb@firma.de
Mitglied bereits vorhanden: tom@firma.de
Mitglied hinzugefügt: anna@firma.de
```

Import abgeschlossen.

Die Abschlussmeldung bestätigt das Ende der Skriptausführung, garantiert jedoch nicht, dass alle Gruppen und Mitgliedschaften erfolgreich verarbeitet wurden. Eventuelle Fehlermeldungen müssen separat geprüft werden.

8. Importergebnis kontrollieren

Alle Verteilergruppen anzeigen

```
Get-DistributionGroup |  
Select-Object DisplayName, PrimarySmtpAddress
```

Mitglieder einer bestimmten Verteilergruppe anzeigen

```
Get-DistributionGroupMember -Identity "it-team@firma.de"
```

Gruppe gezielt überprüfen

```
Get-DistributionGroup -Identity "it-team@firma.de" |  
Format-List Name, Alias, PrimarySmtpAddress
```

Alternativ kann die Prüfung im Exchange Admin Center erfolgen:

<https://admin.exchange.microsoft.com>

Navigieren Sie dort zu **Empfänger → Gruppen**.

9. Häufige Fehler und Lösungen

Fehler: Get-DistributionGroup wird nicht erkannt

Ursache: Keine aktive Exchange-Online-Verbindung oder fehlendes PowerShell-Modul.

Lösung:

```
Import-Module ExchangeOnlineManagement  
Connect-ExchangeOnline
```

Fehler: Die CSV-Datei wird nicht gefunden

Ursache: Die Datei `verteilergruppen.csv` liegt nicht im aktuellen Arbeitsverzeichnis.

Lösung:

```
Get-Location  
Get-ChildItem *.csv
```

Prüfen Sie Dateinamen und Speicherort.

Fehler: Mitglied kann nicht hinzugefügt werden

Mögliche Ursachen:

- Der Empfänger existiert nicht.
- Die E-Mail-Adresse ist falsch.
- Es fehlen Berechtigungen.
- Die angegebene Gruppe kann nicht geändert werden.

Prüfen Sie den Empfänger:

```
Get-Recipient -Identity "max@firma.de"
```

Fehler: Verteilergruppe kann nicht erstellt werden

Mögliche Ursachen:

- Der Gruppenname beziehungsweise Alias ist ungültig.
- Die E-Mail-Adresse wird bereits verwendet.
- Der Administrator besitzt keine ausreichenden Berechtigungen.
- Die Gruppe wird bereits anderweitig verwaltet.

Prüfen Sie vor einem erneuten Import die Fehlermeldung und den Empfängerbestand.

10. Wichtige Hinweise

- Das Skript löscht keine vorhandenen Verteilergruppen.
- Bestehende Mitglieder werden nicht automatisch entfernt.
- In der CSV aufgeführte Mitglieder werden ergänzt.
- Bereits vorhandene Mitgliedschaften werden übersprungen.
- Bei wiederholter Ausführung werden vorhandene Einträge grundsätzlich nicht doppelt angelegt.
- Das Skript gleicht nicht den vollständigen Mitgliederbestand mit der CSV-Datei ab.
- Fehler beim Erstellen einer Verteilergruppe können die Ausführung unterbrechen.

Es empfiehlt sich, vor einem größeren Import zunächst zwei bis drei Testgruppen anzulegen und die Ergebnisse zu kontrollieren.

11. Verbindung beenden

Nach Abschluss aller Arbeiten kann die Exchange-Online-Sitzung beendet werden:

```
Disconnect-ExchangeOnline -Confirm:$false
```

Dokumentation: Import von Exchange-Online-Verteilergruppen

Skript: `Import-Verteilergruppen.ps1`

Plattform: Microsoft 365 / Exchange Online

Verfahren: CSV-basierter PowerShell-Import

Das komplette Script findet man hier

```
# CSV-Datei einlesen
$data = Import-Csv ".\verteilergruppen.csv"

foreach ($group in ($data | Group-Object GroupName)) {

    $firstRow = $group.Group[0]
    $groupAddress = $firstRow.PrimarySmtpAddress

    # Gruppe prüfen
    $existingGroup = Get-DistributionGroup `
        -Identity $groupAddress `
        -ErrorAction SilentlyContinue

    # Gruppe erstellen, falls sie nicht existiert
    if (-not $existingGroup) {
        New-DistributionGroup `
            -Name $firstRow.GroupName `
            -Alias $firstRow.GroupName `
            -PrimarySmtpAddress $groupAddress `
```

```
-Type Distribution `
-ErrorAction Stop
```

```
Write-Host "Gruppe erstellt: $groupAddress" -ForegroundColor Green
}
else {
    Write-Host "Gruppe existiert bereits: $groupAddress" -ForegroundColor Yellow
}
```

```
# Mitglieder hinzufügen
```

```
foreach ($member in $group.Group) {
    try {
        $alreadyMember = Get-DistributionGroupMember `
            -Identity $groupAddress `
            -ResultSize Unlimited |
            Where-Object {
                $_.PrimarySmtpAddress -eq $member.Member
            }

        if (-not $alreadyMember) {
            Add-DistributionGroupMember `
                -Identity $groupAddress `
                -Member $member.Member `
                -ErrorAction Stop

            Write-Host "Mitglied hinzugefügt: $($member.Member)" `
                -ForegroundColor Cyan
        }
        else {
            Write-Host "Mitglied bereits vorhanden: $($member.Member)" `
                -ForegroundColor Yellow
        }
    }
    catch {
        Write-Warning "Fehler bei $($member.Member): $($_.Exception.Message)"
    }
}
```

Write-Host "Import abgeschlossen." -ForegroundColor Green