

XPHONE CONNECT SERVER IMPORT VON TEAMS AUS AD GRUPPEN

STAND 12.10.2020

RECHTLICHE HINWEISE

Alle Rechte sowie Änderungen und Irrtümer vorbehalten. Beschriebene Leistungsmerkmale können in Abhängigkeit von der eingesetzten ITK-Infrastruktur (Soft- & Hardware) nicht oder nur eingeschränkt zur Verfügung stehen. Der Funktionsübersicht erhebt keinen Anspruch auf Vollständigkeit.

XPhone ist ein eingetragenes Warenzeichen der C4B Com For Business AG. Alle verwendeten Hard- und Softwarenamen sind Handelsnamen und / oder Marken der jeweiligen Hersteller.

Stand 12.10.2020

© C4B Com For Business AG

C4B Com For Business AG
Gabriele-Münter-Str. 3-5
82110 Germering
Deutschland

www.c4b.com

INHALT

1. Überblick.....	4
2. Arbeitsgruppen und Monitoringgruppen	4
2.1. Prinzipschema	4
3. Organisationsstruktur und Namensgebung.....	5
4. Vorbereitung.....	5
4.1. AD DS and AD LDS Tools installieren.....	6
4.2. Passwort des Systemadministrators abfragen und als Datei ablegen	6
4.3. Teams erstellen und Beschreibung pflegen.....	7
5. PowerShell Script über Aufgabenplaner ausführen.....	7
6. Das komplette Script.....	8

1. ÜBERBLICK

Die XPhone PowerShell Erweiterung ist ein mächtiges Werkzeug, um viele Einstellungen bequem und schnell für eine große Anzahl Standorte bzw. Konfigurationsgruppen zu setzen. Daraus folgt aber auch die Gefahr, viele Daten fälschlicherweise mit einem Tastendruck zu "zerstören". Lesen Sie daher auch die allgemeinen Hinweise zur Verwendung der PowerShell Erweiterung in der XPhone Connect Server Dokumentation.

Mit dem hier beschriebenen Script werden Teams mit Benutzern aus einer Active Directory-Sicherheitsgruppe importiert.

HINWEISE

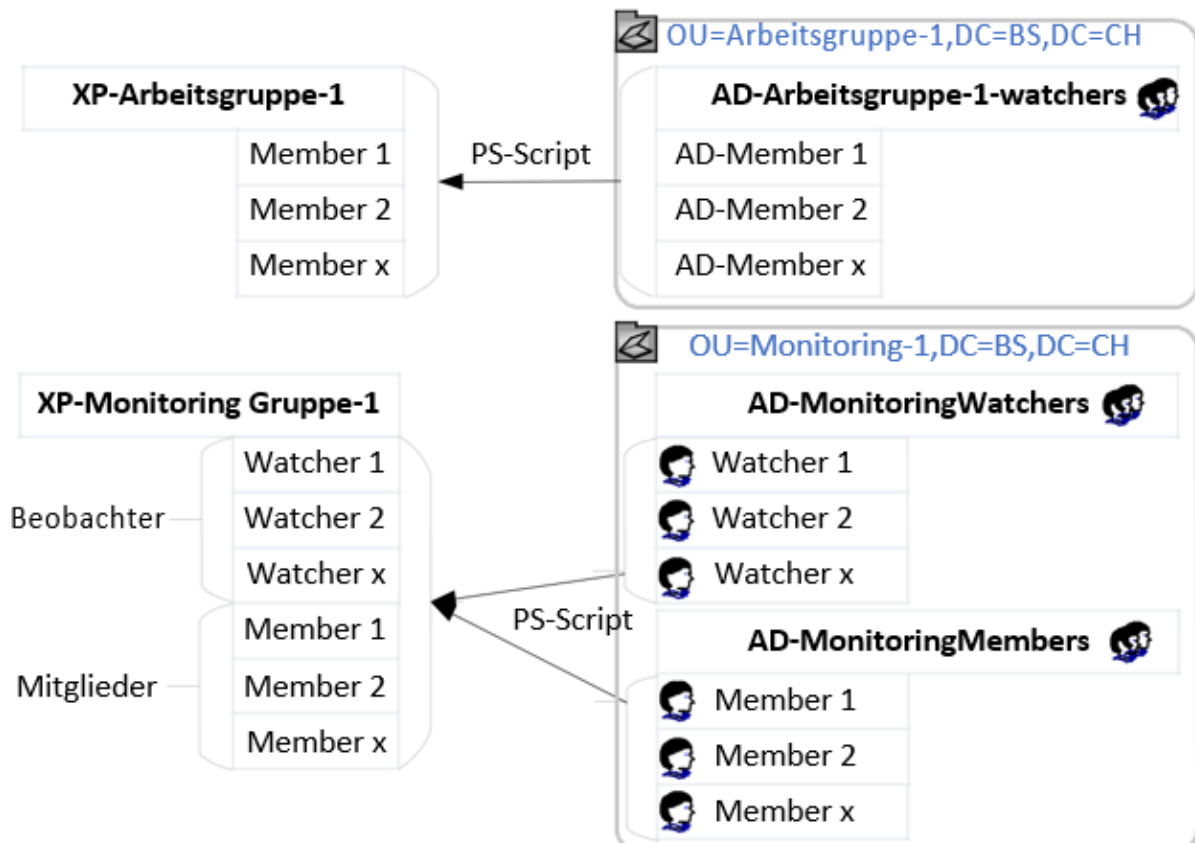
- Mit den PowerShell Erweiterungen in XPhone Connect können keine neuen Teams angelegt werden. Die XPhone Teams müssen also vorab manuell angelegt werden, anschließend kann das Script diese automatisiert befüllen. Ein auf dem XPhone Server neu erstelltes Team mit einer gültigen Beschreibung wird automatisch mit den entsprechenden AD Benutzern befüllt, wenn das PowerShell Script gestartet wurde.
- Es gibt keinen PowerShell Befehl um einen einzelnen Benutzer, der nicht mehr in der AD Gruppe ist zu löschen. Um diese Einschränkung zu umgehen, werden am Anfang des Scripts alle Benutzer aus den Teams gelöscht!

2. ARBEITSGRUPPEN UND MONITORINGGRUPPEN

Jede Arbeitsgruppe ist in einer separaten OU und benötigt eine Sicherheitsgruppe.

Jede Monitoringgruppe ist in einer separaten OU und benötigt zwei Sicherheitsgruppen: Eine für die Beobachter (watchers) und eine für die Gruppenmitglieder (members).

2.1. Prinzipschema



3. ORGANISATIONSSTRUKTUR UND NAMENS GEBUNG

In der Beschreibung der Teams wird die Basis-DN eingetragen, wo sich die entsprechenden Sicherheitsgruppen befinden. Das Script liest die OU aus und durchsucht diese nach Sicherheitsgruppen.

Teams

Bearbeiten	Löschen	Neue Gruppe...	
* NAME ▲		TYP	BESCHREIBUNG
⊕ 01 Arbeitsgruppe PS-Test		Arbeitsgruppe	OU=C4B-QS,OU=Teams;OU=GROUPS,DC=munich,DC=local
⊙ 02 Monitoringgruppe PS-Test		Monitoringgruppe	OU=C4B-Test,OU=Teams;OU=GROUPS,DC=munich,DC=local

Die Monitoringgruppen benötigen **zwei** Sicherheitsgruppen. Eine für die Beobachter (watchers) und eine für die Gruppenmitglieder (members). Das Script durchsucht die OU im Feld *Beschreibung* nach Sicherheitsgruppen, die den Namen "watchers" oder "members" beinhalten.

Beispiele für möglichen Namen der Sicherheitsgruppe der watchers bzw. members:

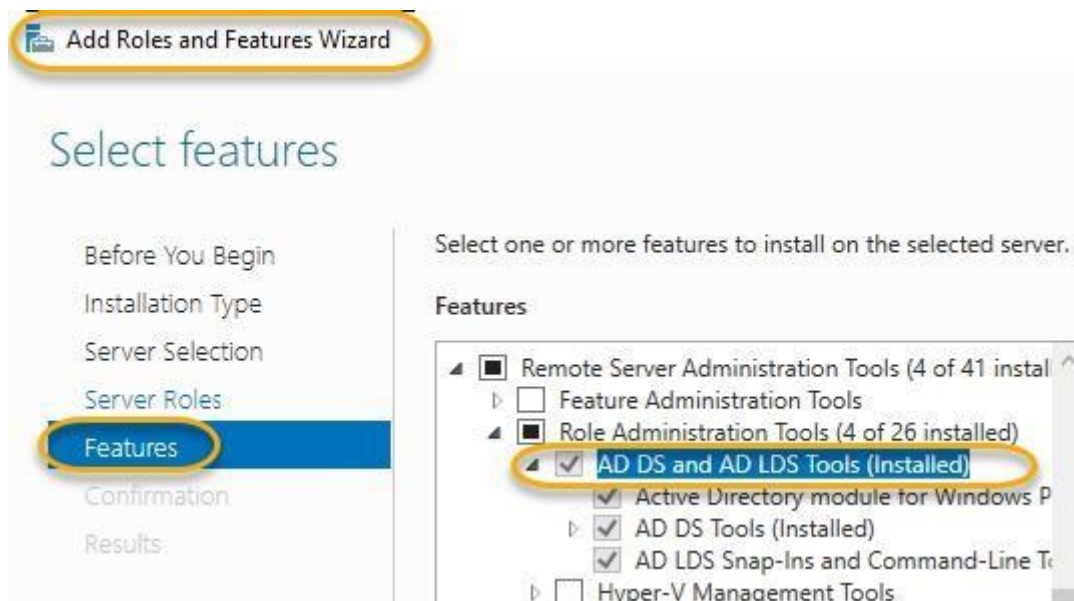
- Vertrieb-Watchers-Gruppe
- Vertrieb-Members-Gruppe
- watchersdergruppevertrieb
- membersdergruppevertrieb
- Vertriebswatchers
- Vertriebsmembers

4. VORBEREITUNG

Folgende Bedingungen müssen erfüllt sein:

- XPhone Server muss Mitglieder der Domäne sein.
- Die *AD DS and AD LDS Tools* müssen als Feature auf dem Server installiert werden.
- Passwort des Systemadministrators abfragen und dann als Datei ablegen.
- Teams eröffnen und in der Beschreibung die entsprechende Base-DN eintragen.
- Die Sicherheitsgruppen müssen im Namen "watchers" oder "members" enthalten.

4.1. AD DS and AD LDS Tools installieren



4.2. Passwort des Systemadministrators abfragen und als Datei ablegen

Im XPhone Server Handbuch ist die Funktionsweise der Methode `get-credential` bereits [beschrieben](#).

Das Script macht eine Abfrage, ob die XML Datei mit dem verschlüsselten Passwort vom XPhone Server schon vorhanden ist oder nicht. Wenn nicht, wird sie zuerst angelegt und man wird einmalig aufgefordert das Passwort einzutragen.

```
# Pfad der Datei mit den Authentifizierungsinformationen
$credentialsFilePath = "c:\_install\Powershell\xpcredentials.xml"

# Test ob es die xml Datei im Pfad gibt
if ( Test-Path $credentialsFilePath ) {
    # Wenn es die Datei gibt, wird sie geladen
    $credentials = Import-CliXml -Path $credentialsFilePath
} else {
    # Wenn es die Datei nicht gibt, wird sie angelegt
    $credentials = get-credential
    $credentials | Export-CliXml -Path $credentialsFilePath
}

Connect-XpServer $credentials

# Danach folgt das Script
Disconnect-XpServer
```

4.3. Teams erstellen und Beschreibung pflegen

Teams

Bearbeiten	Löschen	Neue Gruppe...	
* NAME ▲		TYP	BESCHREIBUNG
⊕ 01 Arbeitsgruppe PS-Test		Arbeitsgruppe	OU=C4B-QS,OU=Teams;OU=GROUPS,DC=munich,DC=local
⊕ 02 Monitoringgruppe PS-Test		Monitoringgruppe	OU=C4B-Test,OU=Teams;OU=GROUPS,DC=munich,DC=local

Hinweis: Um den Schreibaufwand und damit verbundene Schreibfehler zu reduzieren, kann man die OU's mit einem PowerShell Befehl auslesen und "Distinguishedname" auslesen:

```
$DgnameOu = Get-ADOrganizationalUnit -SearchBase "OU=Teams,OU=GROUPS,DC=ucba,DC=local"
-Filter * ` | Select-Object name,distinguishedname; $DgnameOu
```

5. POWERSHELL SCRIPT ÜBER AUFGABENPLANER AUSFÜHREN

- Neue Aufgabe erstellen
- Namen vergeben und Kurzbeschreibung erstellen
- Ausführungsintervall festlegen
- Ausführungszeitpunkt festlegen
- Script soll als Programm ausgeführt werden
- In das Feld "Program/Script" muss der Pfad zu Powershell.exe eingetragen werden.
- Der Pfad vom Script muss im Feld "Add arguments" eingetragen werden.

6. DAS KOMPLETTE SCRIPT

```
# Pfad der Datei mit den Authentifizierungsinformationen
$credentialsFilePath = "c:\_install\Powershell\xpcredentials.xml"

# Test ob es die xml Datei im Pfad gibt
if ( Test-Path $credentialsFilePath ) {

    # Wenn es die Datei gibt, wird sie geladen
    $credentials = Import-Clixml -Path $credentialsFilePath
} else {

    # Wenn es die Datei nicht gibt, wird sie angelegt
    $credentials = get-credential
    $credentials | Export-Clixml -Path $credentialsFilePath
}

Connect-XpServer $credentials

# Alle User aus allen Teams entfernen
Get-XpTeams | Remove-XpTeamMembers -Clear | Set-XpTeams

# XPhone Team Name und die Beschreibung (Description = Basis-DN) werden ermittelt
$Liste = Get-XpTeams | Select-Object Description,Name
$config = @($Liste);

# In der Search-Base wird nach Gruppen gesucht, die im Namen *members* oder *watchers* haben
foreach ($item in $config){
    $ADOU = Get-ADGroup -SearchBase $item.description -Filter * foreach ($ADGROUPS in $ADOU){
        if ($ADGROUPS.SamAccountName -like "*members*"){

            # User aus den Gruppen, die im Namen *members* haben, werden ausgelesen
            # und in das entsprechende Team importiert
            $xpUsers = Get-XpUsersFromADGroups -ADGroupNames $ADGROUPS; $team = Get-XpTeams
            -TeamNames $item.Name;

            # Abfrage ob es den gefundenen User oder das gefundene Team gibt
            if(!$xpUsers) -or (!$team){
                continue;
            }

            $tmp = Add-XpTeamMembers -Teams $team -Users $xpUsers;
            $result = $tmp;
            Set-XpTeams $result;

        }
    }
}

foreach ($ADGROUPS in $ADOU){
    if ($ADGROUPS.SamAccountName -like "*watchers*"){

        # Die User aus den Gruppen, die im Namen *watchers* haben, werden ausgelesen
        # und in das entsprechende Team importiert
        $xpUsers = Get-XpUsersFromADGroups -ADGroupNames $ADGROUPS $team = Get-XpTeams
        -TeamNames $item.Name

        # Abfrage ob es den gefundennn User oder das gefundene Team gibt.
        if(!$xpUsers) -or (!$team){
            continue;
        }

        $tmp = Add-XpTeamMembers -Teams $team -AsObserver $xpUsers $result =
        $tmp;
        Set-XpTeams $result;

    }
}

}

Disconnect-XpServer
```