

技術ノート KGTN 2016102501

現象

[Windows/PowerShell] 複数の GG サーバー（ワークグループ）に複数のアカウントを一括登録したい。

説明

以下にスクリプトのサンプルを示します。

ファイル: hostList.ps1

```
# 処理の対象とするサーバー
$hostList =
@(
    "192.168.102.60",
    "192.168.102.72",
    "192.168.102.110"
)
```

ファイル: userList.ps1

```
# 追加するアカウント
$userList =
@(
    @"(\"wguser9025\", \"Pass. Word\"),
    @"(\"wguser9026\", \"Pass. Word\"),
    @"(\"wguser9027\", \"Pass. Word\")
)
```

ファイル: addUserList.ps1

```
# ホスト情報およびユーザー情報の設定
Try
{
    . ("¥hostList.ps1")
    . ("¥userList.ps1")
}
Catch
{
    Write-Host "インクルードファイルが見つかりません。"
    Exit 1001
}

# アカウントの追加 (本体)
Function addUser ([string]$user, [string]$pass)
{
    # 共通
    [bool]$myError = $false
    $hostname = hostname

    # アカウント作成
    try
    {
        [ADSI]$Computer = "WinNT://$hostname,computer"
        $NewUser = $Computer.Create("User", $user)
        [void]$NewUser.SetPassword($pass)
        [void]$NewUser.SetInfo()
        Write-Host "* addUser: Create = OK (" $hostname ")"
    }
    catch
    {
        $myError = $true
        Write-Host "* addUser: Create = NG (" $hostname ")"
    }

    # RDS の利用設定
    if($myError -eq $false)
    {
        try
        {
            $Group = $Computer.GetObject("group", "Remote Desktop Users")
            [void]$Group.Add("WinNT://$hostname/$user")
            Write-Host "* addUser: RDS = OK (" $hostname ")"
        }
        catch
        {
            $myError = $true
            Write-Host "* addUser: RDS = NG (" $hostname ")"
        }
    }
}
```

```

# パスワード無期限設定
if($myError -eq $false)
{
    try
    {
        [ADSI]$UpdateUser = "WinNT://$hostname/$user, User"
        $UserFlags = $UpdateUser.Get("UserFlags")
        $UserFlags = $UserFlags -bor 0x10000
        [void]$UpdateUser.Put("UserFlags", $UserFlags)
        [void]$UpdateUser.SetInfo()
        Write-Host "* addUser: Password/DontExpire = OK (" $hostname ")"
    }
    catch
    {
        $myError = $true
        Write-Host "* addUser: Password/DontExpire = NG (" $hostname ")"
    }
}
# return [string]($hostname + ":" + $user + ":" + $pass + ":" + !$myError)
return "[0,-16], [1,-16], [2,-16], [3]" -f $hostname, $user, $pass, !$myError
}

# アカウントの追加（呼出し）
$i = 0
ForEach($targetU In $userList)
{
    ++$i
    $j = 0
    ForEach($targetH In $hostList)
    {
        $ret = Invoke-Command $targetH -ScriptBlock ${function:addUser} -ArgumentList
        $targetU[0], $targetU[1]
        ++$j
        "[{0}-{1}] {2}" -f $i, $j, $ret
    }
}

```