



# AUTOMAÇÃO E RELATÓRIOS COM POWERSHELL

Responsável: João Pedro Toledo Gonçalves

Data: 26/01/2026

Código: ITGENG 0030/26 | Classificação: CONFIDENCIAL

Responsável: João Pedro Toledo Gonçalves | Data: 26/01/2026

## 1. HISTÓRICO DE REVISÃO

| Data       | Versão | Descrição       | Autor                       |
|------------|--------|-----------------|-----------------------------|
| 26/01/2026 | 1.0    | Criação Inicial | João Pedro Toledo Gonçalves |

## 2. OBJETIVO

Centralizar scripts de manutenção, relatórios de auditoria e operações em massa para administradores de sistemas. **Elimine o trabalho manual repetitivo.**

## 3. RELATÓRIOS DE AUDITORIA

### 1. Usuários Inativos (Stale Users)

Encontra usuários que não logam há mais de 90 dias.

```
$Dias = 90
Search-ADAccount -AccountInactive -TimeSpan ([timespan]::$Dias.00:00:00) -UsersOnly |
Select-Object Name, SamAccountName, LastLogonDate, Enabled |
Export-Csv "C:\Relatorios\Inativos.csv" -NoTypeInformation
```

### 2. Senhas Prestes a Expirar

Avise os usuários antes do bloqueio.

```
Get-ADUser -filter {Enabled -eq $True -and PasswordNeverExpires -eq $False} -Properties
msDS-UserPasswordExpiryTimeComputed | Select-Object Name, SamAccountName, msDS-UserPasswordExpiryTimeComputed
```

### 3. Membros do Grupo Domain Admins (Segurança)

Verifique quem tem a chave do reino.

```
Get-ADGroupMember "Domain Admins" | Select-Object Name, SamAccountName, ObjectClass
```

---

## 4. OPERAÇÕES EM MASSA (BULK)

### Criação de Usuários via CSV

Arquivo `novos.csv` deve ter colunas: `Nome,Login,Cargo`.

```
Import-Csv "C:\Temp\novos.csv" | ForEach-Object {
    New-ADUser -Name $_.Nome `
        -SamAccountName $_.Login `
        -UserPrincipalName "$($_.Login)@empresa.local" `
        -Title $_.Cargo `
        -Path "OU=Usuarios,DC=empresa,DC=local" `
        -AccountPassword (ConvertTo-SecureString "Mudar@123" -AsPlainText -Force) `
        -ChangePasswordAtLogon $true `
        -Enabled $true
}
```

---

## 5. MANUTENÇÃO DE SERVIDOR

### Limpeza de Disco (Logs Antigos)

Script para rodar no Task Scheduler semanalmente.

```
$Path = "C:\inetpub\logs\LogFiles"
$Daysback = "-30"
Get-ChildItem $Path -Recurse -Force | Where-Object { $_.LastWriteTime -lt (Get-Date).AddDays($Daysback) } | Remove-Item -Force
```

### Verificar Serviços Parados (Automático)

```
$ServicosCriticos = "DNS","Dhcp","LanmanServer"
foreach ($S in $ServicosCriticos) {
    if ((Get-Service $S).Status -ne "Running") {
        # Aqui voce poderia enviar um e-mail ou tentar iniciar
        Start-Service $S
        Write-Warning "Serviço $S foi reiniciado."
    }
}
```

## 6. VALIDAÇÃO FINAL

- ☐ Os scripts rodam sem erros vermelhos?
- ☐ O CSV de saída é gerado corretamente?
- ☐ A execução em massa criou os usuários na OU certa?