

```
# === Configuration initiale ===
$domainName = "printcorp.com"
$domainNetBIOS = "PRINTCORP"
$domainAdminUser = "$domainNetBIOS\Administrateur"
$domainAdminPass = ConvertTo-SecureString "Astaroth01!" -AsPlainText -Force
$domainCred = New-Object System.Management.Automation.PSCredential($domainAdminUser, $domainAdminPass)

# === Configuration IP statique ===
$interfaceAlias = (Get-NetAdapter | Where-Object {$_.Status -eq "Up"}).InterfaceAlias
New-NetIPAddress -InterfaceAlias $interfaceAlias -IPAddress "10.1.100.40" -PrefixLength 24
Set-DnsClientServerAddress -InterfaceAlias $interfaceAlias -ServerAddresses ("127.0.0.1", "10.1.100.1")

# === Renommage et redémarrage pour appliquer le nom ===
Rename-Computer -NewName "PSERV-04" -Force -Restart

# === Rejoindre le domaine (à faire après redémarrage) ===
Add-Computer -DomainName $domainName -Credential $domainCred -Restart

# === Installer les rôles AD, DNS, DHCP ===
Install-WindowsFeature AD-Domain-Services, DNS, DHCP -IncludeManagementTools

# === Promouvoir le serveur en tant que contrôleur de domaine ===
Install-ADDSDomainController `
    -DomainName $domainName `
    -InstallDns `
    -Credential $domainCred `
    -SiteName "Default-First-Site-Name" `
    -SafeModeAdministratorPassword $domainAdminPass `
    -Force

# === Autoriser le serveur DHCP dans l'Active Directory ===
Add-DhcpServerInDC -DnsName "PSERV-04.printcorp.com" -IPAddress "10.1.100.40"

# === Forcer la réinscription DNS et Netlogon
ipconfig /flushdns
ipconfig /registerdns
Restart-Service netlogon
```

## A faire sur PSERV-01 pour le FAILOVER DHCP

```
# === S'assurer que l'ordre DNS est correct (interroge lui-même d'abord)
Set-DnsClientServerAddress -InterfaceAlias "Ethernet0" -ServerAddresses ("127.0.0.1", "10.1.100.1")
```

```
# === Forcer la réinscription DNS et Netlogon
ipconfig /flushdns
ipconfig /registerdns
Restart-Service netlogon

# === Configuration Failover DHCP en HotStandby
$failoverName = "Failover-PSERV"
$partnerServer = "PSERV-04"
$sharedSecret = "FailoverSecret123!"

# Récupération de toutes les étendues DHCP
$scopes = Get-DhcpServerv4Scope

# Création du failover en mode actif/passif (HotStandby)
Add-DhcpServerv4Failover `
    -Name $failoverName `
    -PartnerServer $partnerServer `
    -ScopeId $scopes.ScopeId `
    -MaxClientLeadTime 1:00:00 `
    -ReservePercent 5 `
    -ServerRole "Active" `
    -AutoStateTransition $true `
    -SharedSecret $sharedSecret
```



## ✓ Résumé du projet : Basculement DNS automatique client

### Objectif

Permettre aux postes clients du domaine [PrintCorp](#) de :

- Basculement automatique vers le serveur DNS secondaire (PSERV-04) si PSERV-01 est indisponible
- **Revenir automatiquement** vers PSERV-01 dès qu'il est de nouveau en ligne
- Sans intervention manuelle de l'utilisateur
- Avec **vérification régulière** (au démarrage et toutes les X heures)

---

## ❧ Composants techniques

---

Élément	Description
<a href="#">PSERV-01</a>	DNS/AD/DHCP principal ( <a href="#">10.1.100.10</a> )
<a href="#">PSERV-04</a>	DNS/AD/DHCP secondaire ( <a href="#">10.1.100.40</a> )
<a href="#">PSERV-03</a>	Serveur de fichiers, héberge le script via un partage caché ( <a href="#">\$Script</a> )
<a href="#">Script PowerShell</a>	<a href="#">Dns_Failover.ps1</a> – teste le DNS principal et bascule si injoignable
<a href="#">Script Batch</a>	<a href="#">dns_failover_launcher.bat</a> – lance le <a href="#">.ps1</a> via GPO
<a href="#">GPO de démarrage</a>	Exécute le script au boot de chaque poste avec droits SYSTEM
<a href="#">Tâche planifiée</a>	Relance le script automatiquement toutes les 2 heures

---

## 🔧 Procédure complète : Mise en place du basculement DNS automatique

---

### 1. 📁 Création des scripts

---

#### a. [Dns\\_Failover.ps1](#) (exemple simplifié) :

---

```
# Chemin Ã tester (nom Internet ou interne connu)
```

POWERSHELL

```
$testDomain = "google.com"
$dns1 = "10.1.100.10" # PSERV-01
```

```

$dns2 = "10.1.100.40" # PSERV-04

$interfaces = Get-NetAdapter | Where-Object {$_.Status -eq "Up"}

foreach ($iface in $interfaces) {
    try {
        # Test de résolution avec le DNS principal
        Resolve-DnsName $testDomain -Server $dns1 -QuickTimeout -ErrorAction Stop

        # Si OK : on s'assure que l'ordre est bon
        Set-DnsClientServerAddress -InterfaceAlias $iface.InterfaceAlias -ServerAddresses
        Write-Output "$dns1 OK, DNS remis en ordre normal"

    } catch {
        # Si KO : on bascule vers le DNS secondaire
        Set-DnsClientServerAddress -InterfaceAlias $iface.InterfaceAlias -ServerAddresses
        Write-Output "$dns1 injoignable, DNS basculé sur $dns2"
    }
}

```

## b. dns\_failover\_launcher.bat :

```

@echo off
powershell.exe -ExecutionPolicy Bypass -File "\\PSERV-03\Scripts\DNS_Failover\Dns_

```



## 2. Droits sur le partage \\PSERV-03\Scripts\$

- Dossier local : D:\Scripts
- Partage : Scripts\$ (caché)
- Droits Partage et NTFS :
  - Domain Computers : Lecture
  - Domain Users : Lecture (facultatif)



### 3. GPO de démarrage

---

#### ◆ Emplacement :

---

Configuration de l'ordinateur > Paramètres Windows > Scripts > Démarrage

#### ◆ Script à pointer :

---

```
\\PSERV-03\Scripts$\dns_failover_launcher.bat
```

✓ Ce script est exécuté **en tant que SYSTEM**, donc avec droits élevés.



### 4. GPO : Tâche planifiée pour exécution régulière

---

#### ◆ Emplacement :

---

Configuration de l'ordinateur > Préférences > Paramètres du panneau de configuration > Tâches planifiées

#### ◆ Paramètres :

---

- **Nom** : DNS\_Failover\_Check
- **Compte** : SYSTEM avec privilèges élevés
- **Déclencheur** : toutes les 2 heures
- **Action** :

```
powershell.exe -ExecutionPolicy Bypass -File "\\PSERV-03\Scripts$\Dns_Failover
```

