

LOSTE TRADI-FRANCE

Sécurisation des Scripts deSauvegarde PowerShell (DPAPI)

Guide de déploiement des tâches planifiées

Rédacteur	Paul Barrault-Gautier
Entreprise	Loste Tradi-France
Poste	Administrateur Réseau & Systèmes
Date	Avril 2026
Version	1.0
Classification	Usage interne — Confidentiel

1. Objectif

Ce document décrit la procédure de création et de gestion des tâches planifiées Windows qui exécutent les scripts de sauvegarde PowerShell sécurisés. Une attention particulière est portée à la création des tâches planifiées sans nécessiter de droits administrateur à l'exécution, en utilisant Register-ScheduledTask avec patching XML du déclencheur.

1.1 Contraintes techniques

- Les scripts doivent s'exécuter sous le compte de service dédié au site (ex. LOSTE\svc-backup-site1)
- Le compte de service ne dispose pas des droits administrateur locaux
- La tâche doit s'exécuter même si l'utilisateur n'est pas connecté
- Le déclencheur doit être configurable précisément (heure, jours) sans passer par l'interface graphique
- La tâche doit journaliser son résultat dans l'Observateur d'événements

2. Création de la tâche planifiée via PowerShell

2.1 Script de création — Register-ScheduledTask

i Ce script doit être exécuté UNE SEULE FOIS par site, sous un compte administrateur local du poste. La tâche créée s'exécutera ensuite sous le compte de service sans droits admin.

```
# =====
# New-BackupScheduledTask.ps1
# Création de la tâche planifiée de sauvegarde
# A exécuter sous compte admin – une seule fois par site
# =====

# ——— PARAMÈTRES À ADAPTER PAR SITE ———
$TaskName      = 'Loste-Backup-Site-Production-1'
$ScriptPath    = 'C:\Scripts\Backup\Backup-Site.ps1'
$ServiceAccount = 'LOSTE\svc-backup-site1'
$TriggerTime   = '22:00'      # Heure d'exécution quotidienne
$WorkingDir    = 'C:\Scripts\Backup'

# ——— ACTION : exécuter PowerShell avec le script ———
$Action = New-ScheduledTaskAction `
    -Execute 'powershell.exe' `
    -Argument "-NonInteractive -NoProfile -ExecutionPolicy Bypass -File `
    `"$ScriptPath`"" `
    -WorkingDirectory $WorkingDir

# ——— DÉCLENCHEUR : tous les jours à l'heure définie ———
$Trigger = New-ScheduledTaskTrigger -Daily -At $TriggerTime

# ——— PARAMÈTRES D'EXÉCUTION ———
$Settings = New-ScheduledTaskSettingsSet `
    -ExecutionTimeLimit (New-TimeSpan -Hours 4) `    # Timeout 4h max
    -RestartCount 2 `                                # 2 relances si échec
    -RestartInterval (New-TimeSpan -Minutes 30) `    # Attendre 30 min entre relances
    -StartWhenAvailable `                             # Lancer si l'heure est passée
    -RunOnlyIfNetworkAvailable `                       # Nécessite réseau (NAS)
    -MultipleInstances IgnoreNew                       # Ne pas lancer 2 instances

# ——— PRINCIPAL : compte de service ———
# Demande le mot de passe du compte de service pour l'enregistrement
$Principal = New-ScheduledTaskPrincipal `
    -UserId $ServiceAccount `
    -LogonType Password `                             # S'exécute même si non connecté
```

```
-RunLevel Limited # PAS de droits élevés (non admin)

# — ENREGISTREMENT DE LA TÂCHE —
$Password = Read-Host -Prompt "Mot de passe de $ServiceAccount" -AsSecureString
$PlainPassword = [Runtime.InteropServices.Marshal]::PtrToStringAuto(
    [Runtime.InteropServices.Marshal]::SecureStringToBSTR($Password))

Register-ScheduledTask `
    -TaskName $TaskName `
    -Action $Action `
    -Trigger $Trigger `
    -Settings $Settings `
    -Principal $Principal `
    -Password $PlainPassword `
    -Description "Sauvegarde Robocopy sécurisée DPAPI – Loste Tradi-France" `
    -Force

Write-Host "Tache planifiee '$TaskName' creee avec succes." -ForegroundColor Green
```

2.2 Patch XML du déclencheur (jours spécifiques)

Pour affiner le déclencheur (ex. : lundi au vendredi uniquement, ou exclure certains jours), il est nécessaire de modifier le XML de la tâche directement, car Register-ScheduledTask ne supporte pas tous les paramètres de déclencheur avancés.

```
# Exporter la tâche en XML
Export-ScheduledTask -TaskName $TaskName | Out-File 'C:\Temp\task.xml' -Encoding UTF8

# Ouvrir task.xml et modifier la section <Triggers>
# Exemple : exécution du lundi au vendredi uniquement
<Triggers>
  <CalendarTrigger>
    <StartBoundary>2026-04-01T22:00:00</StartBoundary>
    <Enabled>true</Enabled>
    <ScheduleByWeek>
      <WeeksInterval>1</WeeksInterval>
      <DaysOfWeek>
        <Monday />
        <Tuesday />
        <Wednesday />
        <Thursday />
        <Friday />
      </DaysOfWeek>
    </ScheduleByWeek>
  </CalendarTrigger>
</Triggers>

# Réimporter la tâche modifiée
Unregister-ScheduledTask -TaskName $TaskName -Confirm:$false
Register-ScheduledTask -TaskName $TaskName -Xml (Get-Content 'C:\Temp\task.xml' -Raw)
    -User $ServiceAccount -Password $PlainPassword -Force
```

3. Vérification et gestion des tâches

3.1 Vérifier l'état d'une tâche

```
# Lister toutes les tâches de sauvegarde Loste
Get-ScheduledTask | Where-Object { $_.TaskName -like 'Loste-Backup*' } |
    Select-Object TaskName, State, LastRunTime, LastTaskResult

# Codes LastTaskResult courants :
```

```
# 0 = Succès
# 0x1 = Erreur générique
# 0x41301 = Tâche en cours d'exécution
# 0x41303 = Tâche non encore exécutée
# 0x8004131F = Aucune instance disponible (timeout ou conflit)
```

3.2 Lancer une exécution manuelle

```
# Démarrer la tâche manuellement (test ou rattrapage)
Start-ScheduledTask -TaskName 'Loste-Backup-Site-Production-1'

# Suivre l'exécution en temps réel
Get-ScheduledTask -TaskName 'Loste-Backup-Site-Production-1' | Get-ScheduledTaskInfo
```

3.3 Consulter les logs d'exécution

```
# Logs texte du script
Get-Content 'C:\Scripts\Backup\logs\backup-Site-Production-1-2026-04-09.log'

# Événements Windows liés aux sauvegardes
Get-WinEvent -LogName Application | Where-Object { $_.ProviderName -eq 'Loste-Backup' } | Select -First 20

# Historique des tâches planifiées (Observateur d'événements)
Get-WinEvent -LogName 'Microsoft-Windows-TaskScheduler/Operational' |
    Where-Object { $_.Message -like '*Loste-Backup*' } | Select -First 10
```

4. Dashboard de supervision centralisé

4.1 Principe

Chaque script de sauvegarde met à jour un fichier JSON partagé sur le NAS (\\nas-lostelsupervision\status.json). Un script de génération de dashboard HTML lit ce fichier et produit une page web de synthèse, accessible depuis n'importe quel poste du réseau.

4.2 Script de génération du dashboard HTML

```
# Generate-BackupDashboard.ps1
# Génère un dashboard HTML de supervision des sauvegardes

$StatusFile = '\\nas-loste\supervision\status.json'
$DashboardFile = '\\nas-loste\supervision\dashboard.html'

$AllStatus = Get-Content $StatusFile -Raw | ConvertFrom-Json

$Rows = foreach ($Site in $AllStatus.PSObject.Properties) {
    $S = $Site.Value
    $Color = if ($S.Status -eq 'OK') { '#d4edda' } else { '#f8d7da' }
    $Icon = if ($S.Status -eq 'OK') { 'OK' } else { 'ERREUR' }
    "<tr style='background:$Color'><td>${$S.Site}</td><td>${$S.Date}</td>" +
    "<td><strong>$Icon</strong></td><td>${$S.RoboCode}</td></tr>"
}

$HTML = @"
<!DOCTYPE html><html lang='fr'><head><meta charset='UTF-8'>
<title>Supervision Sauvegardes – Loste</title>
<style>body{font-family:Arial;margin:2em} table{border-collapse:collapse;width:100%}
th{background:#1F4E79;color:white;padding:8px} td{padding:8px;border:1px solid #ccc}</style>
</head><body>
<h1>Supervision des Sauvegardes – Loste Tradi-France</h1>
<p>Mis à jour le : $(Get-Date -Format 'dd/MM/yyyy HH:mm')</p>
"@
```

```
<table><tr><th>Site</th><th>Dernière exécution</th><th>Statut</th><th>Code
Robocopy</th></tr>
$(($Rows -join ' ')</table></body></html>
"@

$HTML | Set-Content $DashboardFile -Encoding UTF8
Write-Host "Dashboard généré : $DashboardFile"
```

5. Tableau de bord opérationnel — Vérifications hebdomadaires

N°	Vérification	Fréquence	Statut
1	Dashboard HTML : tous les sites en statut OK	Quotidien	[] OK / [] KO
2	Logs texte : absence d'erreurs Robocopy (code < 8)	Quotidien	[] OK / [] KO
3	Tâches planifiées : LastTaskResult = 0 sur tous les sites	Hebdo	[] OK / [] KO
4	Espace disque NAS : alerte si < 20 % libre	Hebdo	[] OK / [] KO
5	Intégrité DPAPI : test Import-Clixml sur chaque site	Mensuel	[] OK / [] KO
6	Test de restauration Robocopy depuis le NAS	Trimestriel	[] OK / [] KO