

PowerShell i Harmonogram Zadań – opis techniczny

1. Wprowadzenie

Harmonogram zadań umożliwia automatyczne uruchamianie poleceń, skryptów lub aplikacji. PowerShell zapewnia kontrolę nad tworzeniem, konfiguracją i uruchamianiem zadań poprzez moduł ScheduledTasks.

2. Kluczowe komponenty zadania

- Action – co ma być wykonane.
- Trigger – kiedy ma być wykonane.
- Principal – w jakim kontekście użytkownika.
- Settings – dodatkowe opcje działania.

3. Moduł ScheduledTasks – najważniejsze cmdlety

- Register-ScheduledTask
- Set-ScheduledTask
- Unregister-ScheduledTask
- Get-ScheduledTask
- Start-ScheduledTask

4. Przykłady

Wybrane przykłady użycia modułu ScheduledTasks.

Codzienne uruchamianie Backup.ps1 o 02:00:

```
$Action = New-ScheduledTaskAction -Execute "PowerShell.exe" -  
Argument "-NoProfile -File C:\Scripts\CreateTestFile.ps1"  
$Trigger = New-ScheduledTaskTrigger -Daily -At 02:00  
Register-ScheduledTask -TaskName "DailyBackup" -Action $Action -  
Trigger $Trigger -RunLevel Highest -User "SYSTEM"
```

Dodatkowe informacje – Kontekst wykonywania zadań

1. Kontekst konta wykonywania

Zadanie działa w kontekście konta wskazanego podczas jego tworzenia – konto domenowe, lokalne lub SYSTEM.

2. Wylogowanie nie przerywa działania

Zadania działają niezależnie od zalogowanej sesji użytkownika.

3. Logowanie innego użytkownika

Nie wpływa na wykonywanie zadania – działa ono w kontekście przypisanego konta.

4. Brak aktywnej sesji

Zadania wykonują się nawet na ekranie logowania, jeśli konto ma prawo logon as batch.

5. Restart systemu

Zadanie wykona się po restarcie, o ile czas wykonania jest przyszły. Jeśli restart przypada na czas wykonania – wykona się tylko przy opcji StartWhenAvailable.

6. Zadania jako SYSTEM

Nie wymagają hasła, mają pełne uprawnienia i są odporne na brak sesji użytkownika.

7. Przechowywanie poświadczeń

Tryb „Run whether user is logged on or not” umożliwia uruchamianie zadań bez aktywnej sesji poprzez zapis zaszyfrowanych poświadczeń.