

Planifier un Job Template AWX via l'API

L'interface graphique d'AWX (Tower) peut parfois s'avérer capricieuse ou peu intuitive pour configurer des tâches planifiées complexes. Utiliser l'API REST via une simple commande `curl` permet de créer des planifications précises, rapides et reproductibles.

1. Le Script de Planification

Copiez ce code dans un terminal Linux (ou dans un script bash) pour créer la planification.
Attention à bien modifier les variables avant de l'exécuter.

```
#!/bin/bash

# =====
# Variables de configuration
# =====
AWX_HOST="https://awx.numericare.fr"
TOKEN="ton_token_api_ici"
TEMPLATE_ID="42"

# =====
# Appel API
# =====
curl -X POST "$AWX_HOST/api/v2/job_templates/$TEMPLATE_ID/schedules/" \
  -H "Authorization: Bearer $TOKEN" \
  -H "Content-Type: application/json" \
  -d '{
    "name": "Conversion Media Quotidienne",
    "description": "Exécution automatique 2 fois par jour",
    "rrule": "DTSTART:20240101T030000Z
RRULE:FREQ=DAILY;INTERVAL=1;BYHOUR=3,15;BYMINUTE=0",
    "extra_data": {},
    "inventory": null
  }'
```

2. Explication des Variables

Voici comment récupérer et adapter les valeurs nécessaires pour que la commande fonctionne avec votre environnement :

Variable	Description & Comment la trouver
<code>AWX_HOST</code>	L'URL de base de votre serveur AWX. Exemple : <code>https://awx.mon-homelab.lan</code>
<code>TOKEN</code>	Votre jeton d'accès personnel. Pour le générer dans AWX : allez dans Users > Cliquez sur votre utilisateur > Onglet Tokens > Add (Cochez "Write" ou "All").
<code>TEMPLATE_ID</code>	L'identifiant unique du playbook à lancer. Pour le trouver, ouvrez votre Job Template dans AWX et regardez l'URL : <code>/templates/job_template/42/details</code> . Ici, l'ID est 42.

3. Comprendre la syntaxe `rrule` (La Planification)

Le paramètre le plus complexe de la requête JSON est le `rrule` (Recurrence Rule). Il utilise le standard iCalendar. Voici comment le décoder et le modifier :

```
“ Syntaxe utilisée : DTSTART:20240101T030000Z  
RRULE:FREQ=DAILY;INTERVAL=1;BYHOUR=3,15;BYMINUTE=0
```

- **DTSTART:** Définit la date et l'heure de début de validité de la règle (au format UTC, indiqué par le `Z`). Mettre une date dans le passé fonctionne très bien pour que la règle soit active immédiatement.
- **FREQ=DAILY** : La fréquence de base. Peut être remplacé par `HOURLY` (Toutes les heures), `WEEKLY` (Toutes les semaines), ou `MONTHLY` (Tous les mois).
- **INTERVAL=1** : S'applique tous les X jours (car `FREQ=DAILY`). Si on met 2, le script tournera un jour sur deux.
- **BYHOUR=3,15** : Exécute la tâche à 03h00 et à 15h00 (**Heure UTC**). Attention au décalage horaire selon votre fuseau !
- **BYMINUTE=0** : S'assure que l'exécution se fait à l'heure pile (00 minute).

Autres exemples de `rrule` utiles :

Toutes les 4 heures :	<code>... RRULE:FREQ=HOURLY;INTERVAL=4</code>
Tous les lundis à 2h du matin :	<code>... RRULE:FREQ=WEEKLY;INTERVAL=1;BYDAY=MO;BYHOUR=2;BYMINUTE=0</code>

Revision #2

Created 2026-05-05 09:09:32 UTC by Nico là

Updated 2026-05-05 12:01:33 UTC by Nico là