

# Création d'un Token API AWX et Commandes Utiles

L'API REST d'AWX permet d'automatiser et de piloter l'intégralité de votre infrastructure sans passer par l'interface web. Pour interagir avec cette API en toute sécurité, il est nécessaire de générer un **Personal Access Token (PAT)**.

## 1. Créer un Token d'Accès dans AWX

La génération du token se fait une seule fois depuis l'interface web d'AWX :

1. Connectez-vous à l'interface web d'AWX avec votre compte administrateur.
2. Dans le menu latéral de gauche, allez dans **Access > Users**.
3. Cliquez sur votre nom d'utilisateur (ex: *admin*).
4. Allez dans l'onglet **Tokens** en haut de la page.
5. Cliquez sur le bouton **Add** (Ajouter).
6. Remplissez le formulaire :
  - **Description** : Donnez un nom clair (ex: *Token Script Bash Homelab*).
  - **Scope (Portée)** : Sélectionnez **Write** (si vous avez besoin de lancer des jobs) ou **All**.
7. Cliquez sur **Save**.

**⚠ ATTENTION** : Le Token généré (une longue chaîne de caractères) ne s'affichera **qu'une seule fois**. Copiez-le immédiatement et conservez-le dans un endroit sûr (comme un gestionnaire de mots de passe ou Vaultwarden).

## 2. Comment utiliser le Token

Pour chaque requête API (souvent via `curl`), vous devez passer ce token dans les en-têtes (Headers) HTTP pour prouver votre identité.

```
# Structure de base de l'authentification
curl -H "Authorization: Bearer VOTRE_TOKEN_ICI" https://awx.domaine.lan/api/v2/...
```

# 3. Antisèche (Cheat Sheet) des requêtes API utiles

Voici quelques commandes Bash prêtes à l'emploi. Pensez à adapter `$AWX_HOST` et `$TOKEN` avec vos valeurs.

## A. Trouver l'ID d'un Job Template

Avant de lancer un playbook, vous avez besoin de connaître son ID. Cette commande liste tous les templates avec leur nom et leur ID.

```
curl -s -k -H "Authorization: Bearer $TOKEN" \  
  -X GET "$AWX_HOST/api/v2/job_templates/" \  
  | jq '.results[] | "ID: \(.id) - Nom: \(.name)''
```

Note : L'outil `jq` est utilisé ici pour formater la sortie JSON de manière lisible.

## B. Lancer un Job Template immédiatement

Permet de déclencher un playbook à la demande (très utile pour l'intégrer dans d'autres scripts ou pipelines).

```
TEMPLATE_ID="42" # Remplacez par votre ID  
  
curl -s -k -X POST "$AWX_HOST/api/v2/job_templates/$TEMPLATE_ID/launch/" \  
  -H "Authorization: Bearer $TOKEN" \  
  -H "Content-Type: application/json" \  
  -d '{}'
```

## C. Lancer un Job Template avec une Limit spécifique

Idéal si vous voulez exécuter un playbook générique (comme "Setup SSH") sur une seule machine spécifique via l'API.

```
TEMPLATE_ID="42"  
MACHINE_CIBLE="bookstack"
```

```
curl -s -k -X POST "$AWX_HOST/api/v2/job_templates/$TEMPLATE_ID/launch/" \  
  -H "Authorization: Bearer $TOKEN" \  
  -H "Content-Type: application/json" \  
  -d '{  
    "limit": ""$MACHINE_CIBLE""  
  }'
```

## D. Vérifier le statut du dernier Job exécuté

Permet de savoir si le job s'est bien passé (successful) ou s'il a échoué (failed).

```
TEMPLATE_ID="42"
```

```
curl -s -k -H "Authorization: Bearer $TOKEN" \  
  -X GET "$AWX_HOST/api/v2/job_templates/$TEMPLATE_ID/jobs/?order_by=-id&page_size=1" \  
  | jq '.results[0] | "Job ID: \(.id) - Statut: \(.status)''
```

---

Revision #1

Created 2026-05-05 11:56:45 UTC by Nico là

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