



Automatiser votre réseau avec Red Hat

Un manuel technique pour mettre en œuvre les principales
tâches d'automatisation des réseaux avec la solution
Red Hat Ansible Automation Platform

Sommaire

Accélérer l'exploitation avec l'automatisation des réseaux

Chapitre 1

Installer et configurer Red Hat Ansible Automation Platform

Chapitre 2

Exécuter une première commande et un premier playbook

Chapitre 3

Créer votre inventaire

Chapitre 4

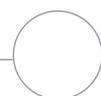
Mettre en œuvre des cas d'utilisation fréquents

Chapitre 5

Accéder au contenu utilisable avec Red Hat Ansible Automation Platform

En savoir plus

Prêt à vous lancer ?



Accélérer l'exploitation avec l'automatisation des réseaux

Les approches traditionnelles et manuelles adoptées jusqu'ici pour configurer et mettre à jour les réseaux sont lentes et propices aux erreurs. Elles ne permettent donc pas de satisfaire efficacement les exigences en matière de transferts des données et d'applications qui évoluent en permanence. Avec des technologies d'automatisation logicielles et programmables, votre équipe sera en mesure de mieux soutenir les initiatives numériques de votre entreprise.

Grâce à l'automatisation des réseaux, les équipes d'exploitation des réseaux (NetOps) peuvent rapidement répondre à l'évolution des besoins en matière de capacités, de sécurité des applications, d'équilibrage de charge et d'intégrations multicloud. Elles peuvent aussi mettre en œuvre des activités réseau en libre-service et à la demande.

Par conséquent, les équipes NetOps ont la capacité de devenir aussi agiles et flexibles que les équipes responsables des infrastructures et des applications afin de prendre en charge les nouvelles demandes métier.

Accélérer l'exploitation avec Red Hat Ansible Automation Platform

Avec la **solution Red Hat Ansible Automation Platform**, nous avons adapté le célèbre projet communautaire Ansible aux entreprises, en y ajoutant les fonctions et fonctionnalités nécessaires aux équipes pour automatiser à grande échelle. Cette plateforme d'automatisation informatique puissante associe un langage d'automatisation simple et facile à lire à un environnement d'exécution modulable et éprouvé ainsi qu'à des capacités de partage et de collaboration sécurisées. Comme elle n'exige aucune compétence en programmation, tous vos salariés, quel que soit leur rôle, peuvent utiliser immédiatement Ansible Automation Platform.

La solution Ansible Automation Platform permet de rationaliser et gérer des environnements de datacenters complexes, des serveurs et réseaux aux applications et pratiques DevOps. Parce qu'elle assure la prise en charge des périphériques d'infrastructure réseau ouverte existants dans des environnements virtuels et physiques à plusieurs fournisseurs, elle facilite l'automatisation de l'ensemble du réseau au moyen d'un seul outil.

Ce livre numérique explique comment effectuer les tâches courantes d'automatisation réseau.

Ressources

Consultez les ressources suivantes pour en savoir plus sur les bases de Red Hat Ansible Automation Platform :

- ▶ Formation en ligne : **Ansible Essentials**
- ▶ Livre numérique : **L'automatisation des réseaux à la portée de tous**



Remarque : les lignes de commandes présentées dans ce livre numérique sont écrites pour Ansible Automation Platform 1.x et ne sont pas applicables à Ansible Automation Platform 2.0, ni aux versions ultérieures.



Installer et configurer Red Hat Ansible Automation Platform

Installer Red Hat Ansible Automation Platform

La solution Red Hat Ansible Automation Platform est facile et rapide à installer et configurer.

Étape

1a

Installer l'interface en ligne de commande d'Ansible avec yum

Exécutez la commande suivante :

```
$ sudo yum install ansible
```

Lisez le [Guide d'installation Ansible](#) pour des instructions complètes.

Étape

1b

Installer Ansible Automation Platform à l'aide de l'outil d'installation

1. Assurez-vous d'avoir la **dernière version** ou téléchargez une version d'essai gratuite sur red.ht/try_ansible.
2. Désarchivez le fichier .tar (la version et le nom peuvent être différents) :

```
$ tar xvzf ansible-automation-platform-setup-bundle-1.2.1-1.tar.gz
```

3. Paramétrez vos mots de passe :

- ▶ `admin_password` pour l'administration
- ▶ `rabbitmq_password` pour la messagerie
- ▶ `pg_password` pour la base de données

4. Exécutez le script de configuration. Une fois que l'installation est terminée, accédez à votre hôte Ansible Automation Platform à l'aide de Google Chrome ou Mozilla Firefox en utilisant le nom d'hôte ou l'adresse IP.

Lisez le [Guide d'installation rapide d'Ansible Automation Platform](#) pour des instructions complètes.

Étape

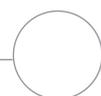
2

Installer les collections de contenu Ansible Automation Platform pour la mise en réseau

Red Hat propose des **collections de contenu** compatibles et certifiées pour un grand nombre de périphériques, outils et infrastructures réseau via [Ansible Automation Hub](#). Chaque collection existe au sein d'un espace de nommage contenant une ou plusieurs collections. Utilisez la commande `ansible-galaxy` pour installer ces collections :

```
$ ansible-galaxy collection install namespace.collection_name
```

Suivez les instructions de la [documentation Red Hat Ansible Automation Platform](#) pour configurer Ansible Automation Hub afin d'accéder à vos collections et de les gérer. Vous trouverez des collections communautaires sur [Ansible Galaxy](#).



Configurer votre environnement réseau

Nous vous recommandons de configurer votre environnement réseau pour Red Hat Ansible Automation Platform selon les meilleures pratiques suivantes :



Assurer la connectivité à votre environnement réseau

Installez un compte de service Ansible Automation Platform sur vos routeurs et commutateurs pour la connexion et l'authentification. La solution Ansible Automation Platform prend en charge les méthodes d'authentification d'entreprise comme le système TACACS+ (Terminal Access Controller Access-Control System Plus) et le service RADIUS (Remote Access Dial-In User Service). Pour en savoir plus, lisez la [section Configurer l'authentification d'entreprise](#) de la documentation.



Créer votre référentiel de playbooks

Connectez Ansible Automation Platform à votre outil de gestion de code source (SCM) en [paramétrant un projet dans l'interface web](#) afin d'accéder à tous les playbooks dans ce référentiel de projet.



Configurer votre inventaire

Créez un [inventaire](#) des périphériques réseau que vous souhaitez automatiser. Ansible Automation Platform peut gérer plusieurs inventaires. Vous pouvez charger des inventaires de manière dynamique depuis des outils courants tels qu'Amazon Web Services EC2, Microsoft Azure Resource Center et VMware vCenter à l'aide des [plug-ins d'inventaire](#). Vous pouvez également [charger des inventaires](#) à partir d'un projet Ansible Automation Platform. Le [Chapitre 3](#) de ce livre numérique explique plus en détail la création et l'utilisation des inventaires.



Paramétrer les règles de pare-feu de votre réseau

Configurez les règles de votre pare-feu pour permettre à la solution Ansible Automation Platform de se connecter aux routeurs et commutateurs via le port Secure Shell (SSH) 22 par défaut. Si vous le souhaitez, vous pouvez modifier le numéro de ce port à l'aide de la [variable d'hôte](#) `ansible_port`.



Paramétrer vos mots de passe pour Ansible Automation Platform

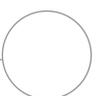
Créez un [identifiant](#) pour retenir vos mots de passe. Vous pouvez autoriser vos utilisateurs et vos équipes à utiliser les identifiants sans les leur divulguer.



Créer un modèle de tâche Ansible

Créez un [modèle de tâche](#) pour connecter votre inventaire, votre identifiant et votre projet. Les modèles de tâche définissent des ensembles de paramètres pour exécuter les tâches d'automatisation, ce qui permet à plusieurs de vos équipes d'exécuter les mêmes ensembles de tâches plusieurs fois et d'en réutiliser le contenu. Chaque modèle de tâche inclut :

- ▶ Un [projet](#) à partir duquel charger les playbooks Ansible.
- ▶ Un [inventaire](#) ou une liste de cibles d'automatisation telles que des commutateurs réseau.
- ▶ Un [identifiant](#) pour vous connecter aux appareils de votre inventaire et les automatiser.

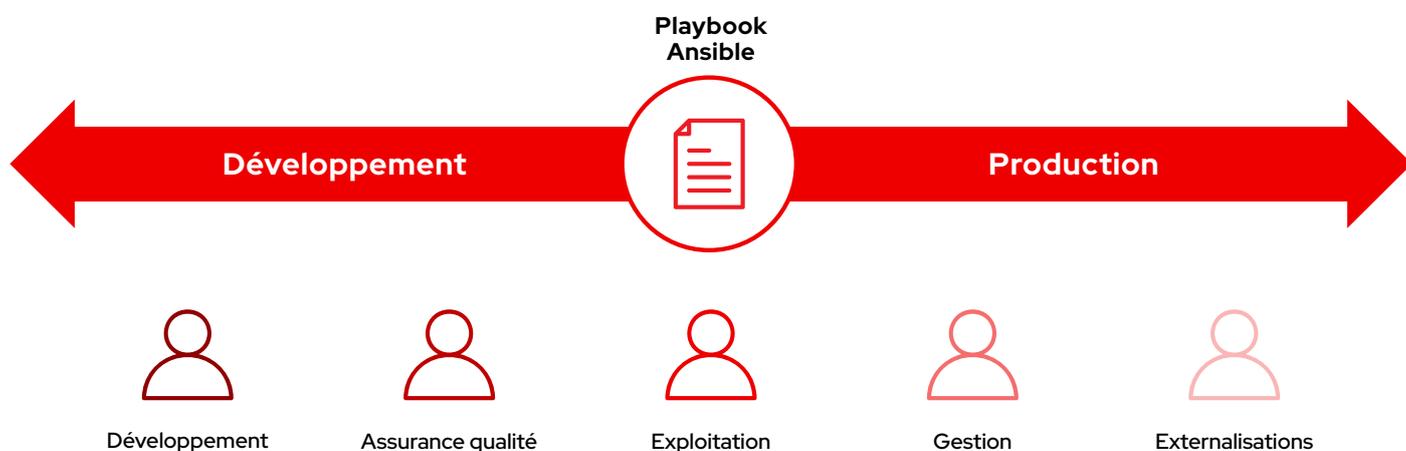


Exécuter une première commande et un premier playbook

Se familiariser avec les playbooks

Les playbooks sont le langage utilisé par Ansible pour la configuration, le déploiement et l'orchestration. Ils se composent d'un ensemble d'instructions en langage naturel appelées « plays » qui définissent l'automatisation pour tout un inventaire d'hôtes. Chaque play comprend une ou plusieurs tâches qui ciblent un, plusieurs ou tous les hôtes d'un inventaire. Chacune des tâches appelle un module Ansible qui se charge d'une fonction spécifique, comme recueillir des informations utiles, sauvegarder des fichiers réseau, gérer la configuration des réseaux ou valider la connectivité.

Les playbooks peuvent être partagés et réutilisés par plusieurs équipes afin de créer un schéma d'automatisation reproductible.



Anatomie d'un playbook

L'exemple suivant montre les parties principales d'un playbook Ansible.

```
1 ---
2 - name: add vlans
3   hosts: arista
4   gather_facts: false
5
6   vars:
7     vlans:
8       - name: desktops
9         vlan_id: 20
10      - name: servers
11        vlan_id: 30
12      - name: DMZ
13        vlan_id: 50
14
15  tasks:
16    - name: add VLAN configuration
17      arista.eos.eos_vlans:
18        state: merged
19        config: "{{ vlans }}"
```

Indique le début d'un playbook

Appelle un périphérique ou groupe de périphériques nommé `arista`

Paramètre facultatif pour récupérer des faits

Définition de la variable

Dans ce playbook, la valeur des variables est définie directement.

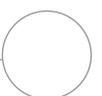
Si vous utilisez l'interface web Red Hat Ansible Automation Platform, vous pouvez aussi **créer un formulaire** pour inviter vos utilisateurs à saisir la valeur des variables lorsqu'ils exécutent votre playbook. Dans ce cas, il faut remplacer les lignes 9, 11 et 13 par :

```
# {{variable_name}} input
```

Nous reparlerons des formulaires dans la section **Créer un formulaire Red Hat Ansible** à la page 6.

Tâches

Les tâches et les modules Ansible ont une corrélation 1:1. Cette section appelle des modules pour configurer des réseaux locaux virtuels (VLAN) pour chacune des trois variables définies dans la section `vars`.



Créer un formulaire Red Hat Ansible

Les formulaires définissent des variables supplémentaires pour votre playbook dans un format question/réponse pratique pour les utilisateurs. Pour créer un formulaire :

1. Cliquez sur le bouton **Add survey** (Ajouter un formulaire) dans l'interface web de Red Hat Ansible Automation Platform.
2. Pour chaque question, indiquez les informations suivantes :
 - ▶ **Nom** : la question à poser à l'utilisateur
 - ▶ **Description (facultative)** : une description de la réponse attendue
 - ▶ **Nom de la variable de réponse** : le nom de la variable Ansible dans laquelle la réponse sera stockée
 - ▶ **Type de réponse** : le format de la réponse (saisie de texte, choix multiples, nombre)
 - ▶ **Réponse par défaut** : la valeur par défaut de la variable
 - ▶ **Obligatoire** : indique si la question est facultative ou non
3. Cliquez sur le bouton **+** pour ajouter la question à votre formulaire.
4. Répétez l'étape 3 pour ajouter d'autres questions à votre formulaire.
5. Cliquez sur le bouton **Save** (Enregistrer) pour enregistrer le formulaire lorsque vous avez terminé.

Pour en savoir plus, lisez la [section sur les formulaires](#) dans la documentation d'Ansible Automation Platform.

Exécuter votre playbook

Exécuter un playbook, ce n'est pas compliqué. Toutefois, la procédure est différente entre l'interface en ligne de commande d'Ansible et l'interface web d'Ansible Automation Platform.

Interface en ligne de commande d'Ansible

Exécutez la commande suivante :

```
ansible-playbook <playbook name> -i <inventory file>
```

Interface web de Red Hat Ansible Automation Platform

Cliquez sur le bouton de lancement de la tâche (fusée) à côté de votre modèle sur l'interface web d'Ansible Automation Platform.



Créer votre inventaire

Comprendre les inventaires

Un **inventaire** est une collection d'hôtes sur lesquels vous pouvez agir à l'aide de commandes et de playbooks Ansible. Les fichiers d'inventaire classent les hôtes en groupes et peuvent servir de source fiable pour votre réseau. Ces fichiers peuvent avoir des formats simples, tels que INI ou YAML. De nombreuses entreprises choisissent d'écrire leurs inventaires en YAML pour qu'ils soient cohérents avec leurs playbooks. À l'aide d'un fichier d'inventaire, un seul playbook peut assurer le bon fonctionnement de centaines de périphériques réseau avec une seule commande.

Ce chapitre explique comment créer un fichier d'inventaire.

Créer un inventaire de base au format INI

Commencez par créer des groupes logiques dans votre inventaire. Les meilleures pratiques conseillent de regrouper les serveurs et périphériques réseau selon leur **fonction (quoi)** (application, pile ou microservice), leur **localisation (où)** (datacenter ou région) et l'**étape (quand)** (étape de développement).

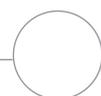
- ▶ **Quoi**: db, web, leaf, spine
- ▶ **Où**: east, west, floor_19, building_A
- ▶ **Quand**: dev, test, staging, prod

Cet exemple de code, au format INI, illustre une structure de groupe de base pour un très petit datacenter. Vous pouvez rassembler les groupes à l'aide de la syntaxe `[metagroupname:children]`, en listant les groupes comme membres du « metagroup ».

Ici, le groupe `network` inclut toutes les « leafs » et les « spines ». Le groupe `datacenter` inclut tous les périphériques réseau et tous les serveurs web.

Pour en savoir plus, lisez la [section Créer votre inventaire](#) de la documentation d'Ansible. Vous trouverez également un [exemple de playbook de rapport d'inventaire](#) sur GitHub.

```
1  [leafs]
2  leaf01
3  leaf02
4
5  [spines]
6  spine01
7  spine02
8
9  [network:children]
10 leafs
11 spines
12
13 [webserver]
14 webserver01
15 webserver02
16
17 [datacenter:children]
18 network
19 webserver
```



Anatomie d'un inventaire au format YAML

```
1 ---
2 all:
3   vars:
4     ansible_user: admin
5     ansible_password: password123
6     ansible_become_pass: password123
7     ansible_become: True
8     ansible_become_method: enable
9     ansible_network_cli_ssh_type: libssh
10  children:
11    routers:
12      children:
13        arista:
14        cisco:
15        juniper:
16    arista:
17      hosts:
18        rtr2:
19          ansible_host: 172.16.100.2
20        rtr4:
21          ansible_host: 172.16.100.4
22      vars:
23        ansible_network_os: arista.eos.eos
24        ansible_connection: ansible.netcommon.network_cli
25    cisco:
26      hosts:
27        rtr1:
28          ansible_host: 172.16.100.1
29      vars:
30        ansible_network_os: cisco.ios.ios
31        ansible_connection: ansible.netcommon.network_cli
32    juniper:
33      hosts:
34        rtr3:
35          ansible_host: 172.16.100.3
36      vars:
37        ansible_network_os: junipernetworks.junos.junos
38        ansible_connection: ansible.netcommon.netconf
```

Indique le début d'un playbook

Définit les variables applicables à tous les hôtes de l'inventaire, quel que soit le groupe

Hiérarchie du groupe

Les lignes 10 à 15 identifient les groupes d'hôtes dans cet inventaire. Dans ce cas, le groupe `routers` contient trois sous-groupes : `arista`, `cisco` et `juniper`.

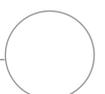
Définition du groupe

La commande `hosts` indique quels hôtes appartiennent à chaque groupe. Dans cet exemple, le groupe `arista` contient deux hôtes identifiés par leur adresse IP.

Variables du groupe

Chaque groupe peut avoir son propre ensemble de variables. Cet inventaire définit le système d'exploitation et le type de connexion pour chaque groupe. Ces deux variables pointent vers des éléments contenus dans des collections de contenu.

Pour en savoir plus sur ces variables, consultez la [section Grouper vos variables](#) à la page 10.

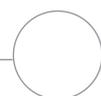


Créer des groupes par plateforme

Si votre inventaire se développe, il est conseillé de regrouper les périphériques par plateforme. Ainsi, vous pourrez définir facilement des variables spécifiques à une plateforme et applicables à tous les périphériques de cette plateforme. Les lignes 10 à 15 de l'exemple d'inventaire identifient la hiérarchie des groupes d'hôtes pour cet inventaire. Le groupe `routers` contient trois enfants ou sous-groupes classés par plateforme : `arista`, `cisco` et `juniper`. Chacun de ces sous-groupes contient un ou plusieurs hôtes, définis par leur adresse IP aux lignes 16-21, 25-28 et 32-35.

```
10  children:
11    routers:
12      children:
13        arista:
14        cisco:
15        juniper:
16  arista:
17    hosts:
18      rtr2:
19        ansible_host: 172.16.100.2
20      rtr4:
21        ansible_host: 172.16.100.4
```

Pour en savoir plus, lisez la [section Créer des groupes par plateforme](#) de la documentation d'Ansible.



Paramétrer vos variables

Dans l'inventaire, vous pouvez définir les valeurs de nombreuses variables dont vous avez eu besoin pour votre première commande et ainsi les omettre dans la commande `ansible-playbook`. Les lignes 2 à 9 de l'exemple d'inventaire définissent des variables qui s'appliquent à tous les hôtes listés dans l'inventaire, quel que soit leur groupe.

```
2  all:
3    vars:
4      ansible_user: admin
5      ansible_password: password123
6      ansible_become_pass: password123
7      ansible_become: True
8      ansible_become_method: enable
9      ansible_network_cli_ssh_type: libssh
```

Vous pouvez paramétrer et stocker des variables dans plusieurs fichiers différents. Les meilleures pratiques conseillent de paramétrer les variables utilisées pour se connecter aux périphériques, notamment des informations de connexion ou une adresse IP, dans des fichiers d'inventaires ou des playbooks. Configurez les variables en lien avec la configuration d'un périphérique dans des fichiers séparés stockés dans le répertoire `group_vars`. Pour en savoir plus, lisez la [section Organiser les variables de groupe et d'hôte](#) de la documentation d'Ansible.

Grouper vos variables

Lorsque des périphériques dans un groupe partagent les mêmes valeurs de variables, notamment le système d'exploitation (OS) ou l'utilisateur SSH, vous pouvez réduire les doublons et simplifier la maintenance en consolidant ces valeurs dans des variables de groupe. Les variables de groupe sont paramétrées au sein de leurs définitions de groupes respectives. Les lignes 22-24, 29-31 et 36-38 de l'exemple d'inventaire configurent des valeurs de variables de groupe pour chacun des trois groupes hôtes.

```
22    vars:
23      ansible_network_os: arista.eos.eos
24      ansible_connection: ansible.netcommon.network_cli
```

Cet exemple définit les variables de système d'exploitation réseau (NOS) et de type de connexion pour chacun des trois sous-groupes. Dans ce cas, ces variables pointent vers des éléments contenus dans les collections de contenu de Red Hat Ansible Automation Platform. Les éléments des collections de contenu ont le format suivant : `namespace.collection_name.item`. Par exemple, `arista.eos.eos` pointe vers le plug-in de système d'exploitation EOS dans la collection EOS servie par l'espace de nommage Arista, tandis que `ansible.netcommon.network_cli` pointe vers le plug-in de la CLI réseau dans la collection Netcommon servie par l'espace de nommage Ansible.



Syntaxe des variables

La syntaxe des valeurs de variables est différente dans l'inventaire, les playbooks et les fichiers group_vars. Même si les playbooks et les fichiers group_vars sont tous deux écrits en YAML, les variables s'utilisent différemment.

Fichiers d'inventaire de style INI

Utilisez la syntaxe `key=value` pour les valeurs des variables :

```
ansible_network_os=cisco.ios.ios
```

Fichiers avec extensions .YML et .YAML

Utilisez la syntaxe YAML :

```
key: value
```

Fichiers group_vars et playbook

Utilisez le nom complet de la clé :

```
ansible_network_os: cisco.ios.ios
```

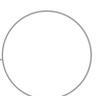
Pour en savoir plus, lisez la [section Syntaxe des variables](#) de la documentation d'Ansible.

Protéger les variables sensibles

Les meilleures pratiques conseillent d'utiliser une protection supplémentaire pour les variables sensibles, telles que les mots de passe.

La solution Red Hat Ansible Automation Platform fournit un gestionnaire d'identifiants pour les mots de passe et les informations de clé. Sur la page **Credentials** (Identifiants) de l'interface web, vous pouvez autoriser vos utilisateurs et vos équipes à utiliser les identifiants sans les leur divulguer. Pour en savoir plus, lisez la [section sur les identifiants](#) de la documentation d'Ansible.

Notez que la plateforme Ansible Automation Platform peut être exécutée sur des systèmes où le **mode FIPS (Federal Information Processing Standards)** est activé.



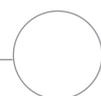
Mettre en œuvre des cas d'utilisation fréquents

Ce chapitre montre des exemples de playbooks pour les cas d'utilisation d'automatisation des réseaux les plus fréquents.

Ajouter un VLAN

La configuration d'un VLAN sur plusieurs périphériques réseau est une activité récurrente pour l'équipe NetOps. Ansible facilite la création d'un VLAN et sa propagation sur votre réseau.

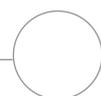
```
1 ---
2 - name: add vlans
3   hosts: arista
4   gather_facts: false
5   vars:
6     vlans:
7       - name: desktops
8         vlan_id: 20
9       - name: servers
10        vlan_id: 30
11       - name: DMZ
12        vlan_id: 50
13
14   tasks:
15     - name: add VLAN configuration
16       arista.eos.eos_vlans:
17         state: merged
18         config: "{{ vlans }}"
```



Regrouper des informations

La plupart des réseaux comprennent un grand nombre de plateformes et de périphériques différents. Ansible peut interroger les données de votre réseau (versions logicielles, informations sur les interfaces), les stocker et générer des rapports.

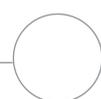
```
1 ---
2 - name: use facts module
3   hosts: cisco
4   gather_facts: false
5
6   tasks:
7     - name: retrieve facts
8       cisco.ios.ios_facts:
9
10    - name: display version
11      debug:
12        msg: "{{ ansible_net_version }}"
13
14    - name: display serial number
15      debug:
16        msg: "{{ ansible_net_serialnum }}"
```



Récupérer des informations sur une ressource

Les **modules de ressource réseau** Ansible simplifient et standardisent la gestion de plusieurs périphériques réseau. Tous les modules de ressource peuvent utiliser `state: gathered` pour récupérer des informations à propos de ressources réseau.

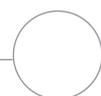
```
1 ---
2 - name: retrieve interface information
3   hosts: cisco
4   gather_facts: false
5
6   tasks:
7     - name: use state gathered
8       cisco.ios.ios_interfaces:
9         state: gathered
10        register: interfaces_info
11
12    - name: print out interfaces information
13      debug:
14        msg: "{{ interfaces_info }}"
```



Sauvegarder des configurations

Le stockage des sauvegardes des configurations est une activité essentielle pour les équipes NetOps. Ansible Automation Platform facilite l'extraction des données relatives à l'ensemble ou à certaines parties seulement d'une configuration depuis un périphérique réseau.

```
1 ---
2 - hosts: cisco
3   gather_facts: false
4
5   tasks:
6     - name: back up config
7       cisco.ios.ios_config:
8         backup: yes
```



Accéder au contenu utilisable avec Red Hat Ansible Automation Platform

Vous pouvez accéder à du contenu d'automatisation prêt à l'emploi afin d'adopter Red Hat Ansible Automation Platform encore plus rapidement et facilement.



Collections de contenu Ansible

Une **collection de contenu** représente un format standardisé de distribution de contenu Ansible qui peut inclure des exemples de playbooks, rôles, modules, plug-ins et bien plus encore. Ce nouveau format sépare l'exécutable Ansible de la plupart du contenu d'automatisation pour vous offrir plus de flexibilité et de portabilité. Vous pouvez installer des collections communautaires issues d'**Ansible Galaxy** et des **collections certifiées** entièrement prises en charge issues d'**Ansible Automation Hub**.



Rôles Ansible

Les rôles Ansible regroupent le contenu d'automatisation pour le rendre réutilisable. Au lieu de créer de longs playbooks contenant des centaines de tâches, vous pouvez classer et diviser les tâches en unités de travail distinctes plus petites en utilisant les rôles. Un rôle inclut toutes les tâches, variables et gestionnaires nécessaires pour former une unité de travail complète. Les rôles sont distribués comme des entités autonomes ou des parties des collections de contenu.



Ansible Automation Hub

Ansible Automation Hub donne accès aux titulaires de souscription Red Hat Ansible Automation Platform à des collections de contenu entièrement prises en charge et certifiées, développées, testées et mises à jour par Red Hat et ses partenaires technologiques. Il vous fournit un portail sécurisé vers les collections de contenu, ainsi qu'un réseau privé pour le contenu d'automatisation interne et tiers. Automation Hub est le référentiel standard pour le contenu utilisé dans les environnements d'automatisation en production.



Ansible Galaxy

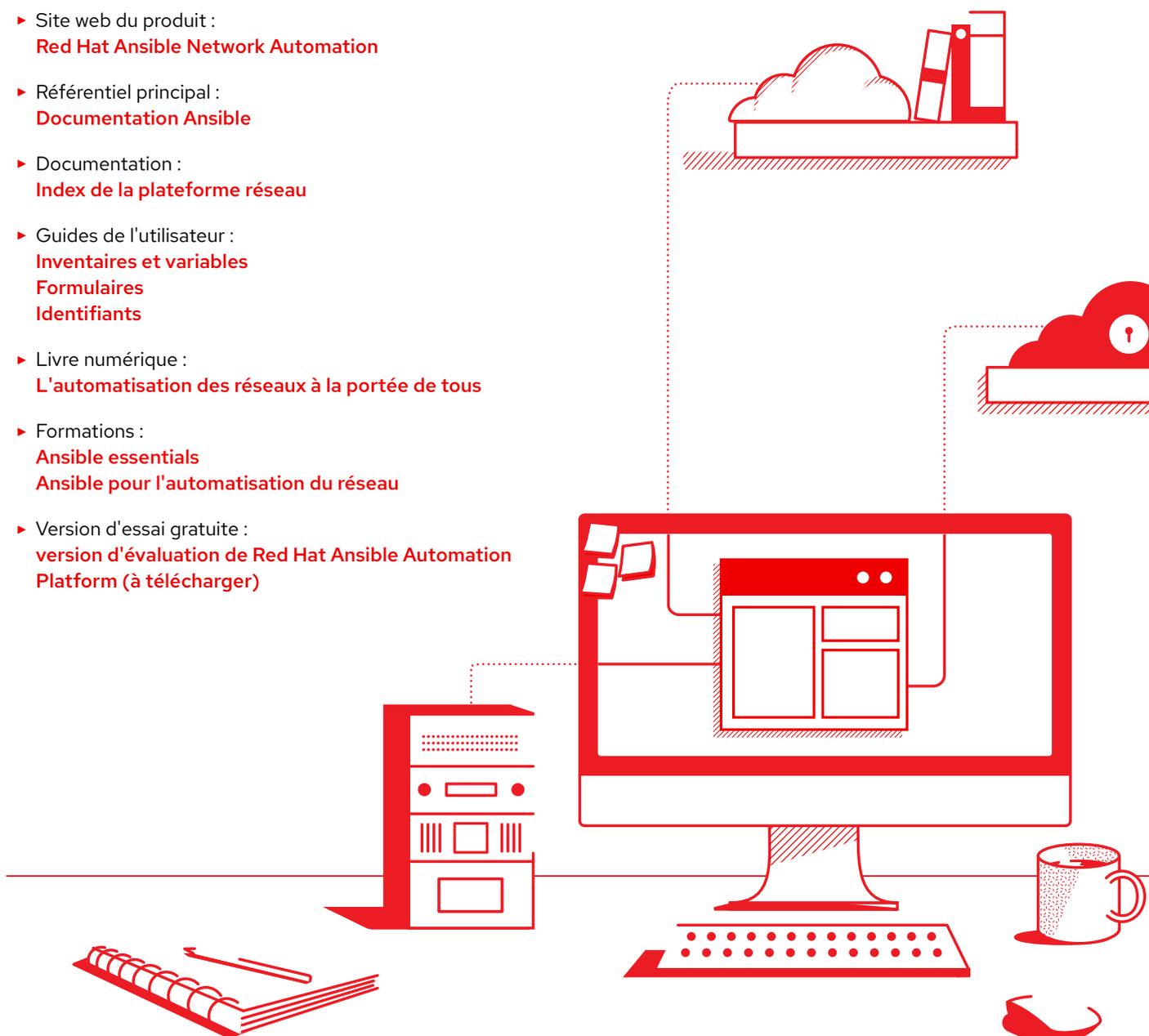
Ansible Galaxy abrite toutes les collections Ansible communautaires, ainsi que les rôles autonomes. Vous pouvez aussi contribuer à la communauté en déposant dans Ansible Galaxy les collections et rôles que vous créez.



En savoir plus

Red Hat met à votre disposition de nombreuses ressources sur la solution Red Hat Ansible Automation Platform, notamment de la documentation détaillée, des articles, des vidéos et des forums de discussion. La plupart de ces ressources sont disponibles à l'adresse [ansible.com](https://www.ansible.com) et sur le [portail client Red Hat](#).

- ▶ Site web du produit :
[Red Hat Ansible Network Automation](#)
- ▶ Référentiel principal :
[Documentation Ansible](#)
- ▶ Documentation :
[Index de la plateforme réseau](#)
- ▶ Guides de l'utilisateur :
[Inventaires et variables](#)
[Formulaires](#)
[Identifiants](#)
- ▶ Livre numérique :
[L'automatisation des réseaux à la portée de tous](#)
- ▶ Formations :
[Ansible essentials](#)
[Ansible pour l'automatisation du réseau](#)
- ▶ Version d'essai gratuite :
[version d'évaluation de Red Hat Ansible Automation Platform \(à télécharger\)](#)



Prêt à automatiser votre réseau ?

À l'aide d'un langage naturel compréhensible et intuitif, la solution Red Hat Ansible Automation Platform vous offre une manière simple et puissante de moderniser l'exploitation de votre réseau, tout en prenant en charge vos processus et votre infrastructure actuels. Avec son framework flexible et évolutif, vous pouvez plus facilement améliorer le niveau de disponibilité de l'infrastructure, de productivité de vos équipes, de sécurité de vos réseaux et de conformité de la configuration.

Essayer gratuitement Red Hat Ansible Automation Platform :
red.ht/try_ansible

Accélérez le déploiement avec l'aide des spécialistes Red Hat.

L'automatisation du réseau peut sembler difficile, mais le service de consulting Red Hat peut vous aider. Tous les contrats de consulting Red Hat débutent par une session de découverte gratuite d'une demi-journée sur site. Au cours de cette session, les spécialistes Red Hat vous aident à identifier vos défis les plus urgents, les meilleures approches pour les relever et les résultats souhaités pour la mise en œuvre de l'automatisation de votre réseau.

Pour organiser une session de découverte gratuite :
redhat.com/consulting