



Installer le connecteur d'attributs dynamiques Cisco Secure

Ce chapitre explique comment installer le connecteur d'attributs dynamiques Cisco sur tous les systèmes d'exploitation pris en charge.

- [Systèmes d'exploitation et logiciels tiers pris en charge, à la page 1](#)
- [Installer les logiciels prérequis, à la page 2](#)
- [Installer le Connecteur d'attributs dynamiques Cisco Secure, à la page 6](#)
- [Mettre à niveau le Connecteur d'attributs dynamiques Cisco Secure, à la page 9](#)

Systemes d'exploitation et logiciels tiers pris en charge

Les conditions requises par le connecteur d'attributs dynamiques sont les suivantes :

- Ubuntu 18.04 ou 20.04
- CentOS 7 Linux
- Red Hat Enterprise Linux (RHEL) 7 ou 8
- Python 3.6.x
- Ansible 2.9 ou version ultérieure

Exigences minimales pour tous les systèmes d'exploitation :

- 4 unités centrales (CPU)
- 8 Go DE RAM
- 100 Go d'espace disque disponible

Si vous souhaitez utiliser les attributs vCenter, nous avons également besoin de :

- vCenter 6.7
- Les outils VMware doivent être installés sur la machine virtuelle.

Dimensionnement de la machine virtuelle

Nous vous recommandons de dimensionner vos machines virtuelles comme suit :

- 50 connecteurs, en supposant 5 filtres par connecteur et 20 000 charges de travail : 4 CPU; 8 Go de RAM; 100 Go d'espace disque disponible
- 125 connecteurs, en supposant 5 filtres par connecteur et 50 000 charges de travail : 8 CPU; 16 Go de RAM; 100 Go d'espace disque disponible



Remarque

Si vous ne dimensionnez pas correctement vos machines virtuelles, le connecteur d'attributs dynamiques risque de ne pas fonctionner ou de ne pas démarrer.

Installer les logiciels prérequis

Avant de commencer

Assurez-vous que vous disposez d'une configuration physique ou virtuelle et que le système qui peut communiquer avec votre le On-Prem Firewall Management Center ou Cloud-Delivered Firewall Management Center (centre de gestion de pare-feu en nuage).

Étape 1

(Facultatif) Utilisez un éditeur de texte pour modifier le fichier `/etc/environment` afin d'exporter les variables suivantes afin de permettre la communication avec Internet si votre machine Ubuntu se trouve derrière un proxy Internet.

Variable	Valeur
<code>export http_proxy</code>	À utiliser avec un proxy HTTP. <i>utilisateur:mot de passe@hôte ou ip:port</i>
<code>export https_proxy</code>	À utiliser avec un proxy HTTPS. <i>utilisateur:mot de passe@hôte ou ip:port</i>
<code>export no_proxy</code>	Supprimer la configuration du proxy. <code>export no_proxy="localhost,127.0.0.1"</code>

Exemples :

Proxy HTTP sans authentification :

```
vi /etc/environment
export http_proxy="myproxy.example.com:8181"
```

Proxy HTTPS avec authentification :

```
vi /etc/environment
export https_proxy="ben.smith:bens-password@myproxy.example.com:8181"
```

Étape 2

Utilisez une autre fenêtre de commande pour confirmer les paramètres :

```
env grep | proxy
```

Exemple de résultat :

```
http_proxy=myproxy.example.com:8181
```

Étape 3 Poursuivre avec l'une des sections suivantes.

Sujets connexes

[Installer les logiciels prérequis : Ubuntu](#), à la page 5

[Installer les logiciels prérequis : CentOS](#), à la page 3

[Installer les logiciels prérequis : RHEL](#), à la page 4

Installer les logiciels prérequis : CentOS

Avant de commencer

Effectuez toutes les opérations suivantes :

- Assurez-vous que votre système remplit les conditions préalables décrites dans la section [Systèmes d'exploitation et logiciels tiers pris en charge](#), à la page 1.
- (Facultatif) Si vous avez besoin d'un accès proxy au connecteur d'attributs dynamiques, consultez [Installer les logiciels prérequis](#), à la page 2.

Étape 1 Assurez-vous que Docker n'est pas installé et désinstallez-le s'il l'est.

```
docker --version
```

Si Docker est installé, désinstallez-le comme indiqué dans [Désinstallation du moteur Docker sur Ubuntu](#).

Étape 2 Mettez à jour et mettez à niveau vos référentiels.

CentOS 7 :

```
sudo yum -y update && sudo yum -y upgrade
```

Étape 3 Installez le référentiel epel.

CentOS 7 :

```
sudo yum -y install epel-release
```

Étape 4 (CentOS 7 uniquement.) Installez Python 3.

```
sudo yum install -y python3 libselenium-python3
```

Étape 5 Installez Ansible.

CentOS 7:

```
sudo yum install -y ansible
```

Étape 6 Vérifiez que la version d'Ansible est 2.9 ou ultérieure.

CentOS 7 :

```
ansible --version
ansible 2.9.24
config file = /etc/ansible/ansible.cfg
```

```
configured module search path = [u'/home/admin/.ansible/plugins/modules',
u'/usr/share/ansible/plugins/modules']
ansible python module location = /usr/lib/python2.7/site-packages/ansible
executable location = /usr/bin/ansible
python version = 2.7.5 (default, Apr  2 2020, 13:16:51) [GCC 4.8.5 20150623 (Red Hat 4.8.5-39)]
```

Remarque Il est normal qu'Ansible fasse référence à Python 2.x, comme le montre l'exemple précédent. Le connecteur utilisera toujours Python 3.

Prochaine étape

Installez le connecteur comme mentionné dans la section [Installer le Connecteur d'attributs dynamiques Cisco Secure, à la page 6](#).

Pour arrêter d'utiliser un proxy avec le connecteur d'attributs dynamiques, modifiez le fichier `/etc/environment` et supprimez la configuration du proxy.

Installer les logiciels prérequis : RHEL

Avant de commencer

Effectuez toutes les opérations suivantes :

- Assurez-vous que votre système remplit les conditions préalables décrites dans la section [Systèmes d'exploitation et logiciels tiers pris en charge, à la page 1](#).
- (Facultatif) Si vous avez besoin d'un accès proxy au connecteur d'attributs dynamiques, consultez [Installer les logiciels prérequis, à la page 2](#).

Étape 1

Assurez-vous que Docker n'est pas installé et désinstallez-le s'il l'est.

```
docker --version
```

Si Docker est installé, désinstallez-le comme indiqué dans [Désinstallation du moteur Docker sur Ubuntu](#).

Étape 2

Mettez à jour vos référentiels.

RHEL 7 :

```
sudo yum -y update && sudo yum -y upgrade
```

RHEL 8 :

```
sudo dnf -y update && sudo dnf -y upgrade
```

Étape 3

Installez le référentiel epel.

RHEL 7 :

```
sudo yum -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

RHEL 8 :

```
sudo dnf -y install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

Étape 4

(RHEL 7 uniquement.) Installez Python 3.

```
sudo yum install -y python3 libselenium-python3
```

Étape 5

Installez Ansible.

RHEL 7 :

```
sudo yum -y install ansible
```

RHEL 8 :

```
sudo dnf install -y ansible
```

Étape 6

Vérifiez la version d'Ansible.

```
ansible --version
```

Voici un exemple.

RHEL 7 :

```
ansible 2.9.24
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/stevej/.ansible/plugins/modules',
u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.5 (default, Mar 20 2020, 17:08:22) [GCC 4.8.5 20150623 (Red Hat 4.8.5-39)]
```

Remarque Il est normal qu'Ansible fasse référence à Python 2.x, comme le montre l'exemple précédent. Le connecteur utilisera toujours Python 3.

RHEL 8 :

```
ansible 2.9.24
  config file = /etc/ansible/ansible.cfg
  configured module search path = ['/home/stevej/.ansible/plugins/modules',
'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python3.6/site-packages/ansible
  executable location = /usr/bin/ansible
  python version = 3.6.8 (default, Mar 18 2021, 08:58:41) [GCC 8.4.1 20200928 (Red Hat 8.4.1-1)]
```

Prochaine étape

Installez le connecteur comme mentionné dans la section [Installer le Connecteur d'attributs dynamiques Cisco Secure, à la page 6](#).

Pour arrêter d'utiliser un proxy avec le connecteur d'attributs dynamiques, modifiez le fichier `/etc/environment` et supprimez la configuration du proxy.

Installer les logiciels prérequis : Ubuntu

Cette tâche explique comment installer les logiciels prérequis sur Ubuntu.

Étape 1

Assurez-vous que Docker n'est pas installé et désinstallez-le s'il l'est.

```
docker --version
```

Si Docker est installé, désinstallez-le comme indiqué dans [Désinstallation du moteur Docker sur Ubuntu](#).

Étape 2 Mettez à jour vos référentiels.

```
sudo apt -y update && sudo apt -y upgrade
```

Étape 3 Vérifiez votre version de Python.

```
/usr/bin/python3 --version
```

Si la version est antérieure à la version 3.6, vous devez installer la version 3.6.x.

Étape 4 Installez Python 3.6.

```
sudo apt -y install python3.6
```

Étape 5 Installez les bibliothèques communes.

```
sudo apt -y install software-properties-common
```

Étape 6 Installez Ansible.

```
sudo apt-add-repository -y -u ppa:ansible/ansible && sudo apt -y install ansible
```

Étape 7 Vérifiez la version d'Ansible.

```
ansible --version
```

Voici un exemple.

```
ansible --version
ansible 2.9.19
  config file = /etc/ansible/ansible.cfg
  configured module search path = [u'/home/admin/.ansible/plugins/modules',
u'/usr/share/ansible/plugins/modules']
  ansible python module location = /usr/lib/python2.7/dist-packages/ansible
  executable location = /usr/bin/ansible
  python version = 2.7.17 (default, Feb 27 2021, 15:10:58) [GCC 7.5.0]
```

Remarque Il est normal qu'Ansible fasse référence à Python 2.x, comme le montre l'exemple précédent. Le connecteur utilisera toujours Python 3.6.

Prochaine étape

Installez le connecteur comme mentionné dans la section [Installer le Connecteur d'attributs dynamiques Cisco Secure](#), à la page 6.

Pour arrêter d'utiliser un proxy avec le connecteur d'attributs dynamiques, modifiez le fichier `/etc/environment` et supprimez la configuration du proxy.

Installer le Connecteur d'attributs dynamiques Cisco Secure

À propos de l'installation

Cette rubrique traite de l'installation du Connecteur d'attributs dynamiques Cisco Secure. Vous devez installer le connecteur en tant qu'utilisateur dotés de privilèges `sudo` mais vous pouvez exécuter le connecteur en tant qu'utilisateur sans privilèges.

Avant de commencer

Assurez-vous que votre système dispose des logiciels prérequis suivants :

- Ubuntu 18.04 ou 20.04
- CentOS 7 Linux
- Red Hat Enterprise Linux (RHEL) 7 ou 8
- Python 3.6.x
- Ansible 2.9 ou version ultérieure

Exigences minimales pour tous les systèmes d'exploitation :

- 4 unités centrales (CPU)
- 8 Go DE RAM
- 100 Go d'espace disque disponible

Nous vous recommandons de dimensionner vos machines virtuelles comme suit :

- 50 connecteurs, en supposant 5 filtres par connecteur et 20 000 charges de travail : 4 CPU; 8 Go de RAM; 100 Go d'espace disque disponible
- 125 connecteurs, en supposant 5 filtres par connecteur et 50 000 charges de travail : 8 CPU; 16 Go de RAM; 100 Go d'espace disque disponible



Remarque Si vous ne dimensionnez pas correctement vos machines virtuelles, le connecteur d'attributs dynamiques risque de ne pas fonctionner ou de ne pas démarrer.

Si vous souhaitez utiliser les attributs vCenter, nous avons également besoin de :

- vCenter 6.7
- Les outils VMware doivent être installés sur la machine virtuelle.

Pour installer les logiciels prérequis, reportez-vous à la section [Installer les logiciels prérequis](#), à la page 2.

Consulter le fichier Lisez-moi et les notes de mise à jour

Pour obtenir les dernières informations sur l'installation, consultez le site suivant :

Lisez-moi : <https://galaxy.ansible.com/cisco/csdac>

Notes de mise à jour : [Connecteur d'attributs dynamiques Cisco Secure Notes de mise à jour](#)

Obtenir le logiciel du connecteur d'attributs dynamiques

Pour obtenir la dernière version du logiciel connecteur d'attributs dynamiques, exécutez la commande suivante :

```
ansible-galaxy collection install cisco.csdac
```

Installer le service d'appel (muster)

Le service d'appel est un autre nom pour le connecteur d'attributs dynamiques.

Exécutez la commande suivante à partir du répertoire

```
~/ansible/collections/ansible_collections/cisco/csdac .
```

```
ansible-playbook default_playbook.yml [--ask-become-pass] [--extra-vars " vars " ]
```

Description de la syntaxe

--ask-become-pass Vous êtes invité à saisir le mot de passe **sudo**. Obligatoire si sudo est activé sur votre machine.

--extra-vars Les variables optionnelles suivantes permettent au connecteur d'attributs dynamiques d'utiliser un proxy. La valeur utilisée doit correspondre à la valeur du fichier `/etc/environment`, que vous avez configuré comme indiqué dans la section [Installer les logiciels prérequis](#), à la page 2.

- **csdac_proxy_enabled=true**
- **csdac_http_proxy_url=http://PROXY_URL**
csdac_https_proxy_url=PROXY_URL

Les variables supplémentaires facultatives suivantes permettent de créer un certificat signé automatiquement que vous pouvez utiliser pour vous connecter en toute sécurité à l'application connecteur d'attributs dynamiques. Si vous omettez ces paramètres, connecteur d'attributs dynamiques utilise un certificat par défaut.

- **csdac_certificate_domain**
nom de domaine pour le certificat autogénéré. La valeur par défaut est le nom d'hôte auto détecté de l'hôte (détecté par ansible).
 - **csdac_certificate_country_name**
Code pays à deux lettres. (la valeur par défaut est `US`)
 - **csdac_certificate_organization_name**
Nom de l'entreprise. (La valeur par défaut est `Cisco`)
 - **csdac_certificate_organization_unit_name**
Nom de l'unité organisationnelle (la valeur par défaut est `Cisco`)
-

Exemple d'installation avec un certificat par défaut

Par exemple, pour installer le logiciel avec les options par défaut :

```
ansible-galaxy collection install cisco.csdac
cd ~/ansible/collections/ansible_collections/cisco/csdac
ansible-playbook default_playbook.yml --ask-become-pass
```

Exemple d'installation avec un certificat facultatif

Par exemple, pour installer le logiciel avec un certificat facultatif :


```
ansible-galaxy collection install cisco.csdac
cd ~/.ansible/collections/ansible_collections/cisco/csdac
ansible-playbook default_playbook.yml --ask-become-pass --extra-vars
"csdac_certificate_domain=domain.example.com csdac_certificate_country_name=US
csdac_certificate_organization_name=Cisco
csdac_certificate_organization_unit_name=Engineering"
```

Après avoir créé le certificat, importez-le dans le navigateur Web que vous utiliserez pour accéder au connecteur. Le certificat est créé dans le répertoire `~/csdac/app/config/certs`.

Afficher le journal d'installation

Le journal d'installation se trouve à l'emplacement suivant :

```
~/.ansible/collections/ansible_collections/cisco/csdac/logs/csdac.log
```

Utilisez votre certificat pour vous connecter à connecteur d'attributs dynamiques

Si vous disposez d'un certificat et d'une clé, placez-les dans le répertoire `~/csdac/app/config/certs` sur votre machine virtuelle.

Après avoir effectué la tâche précédente, redémarrez le conteneur Docker de connecteur d'attributs dynamiques en entrant la commande suivante :

```
docker restart muster-ui
```

Connectez-vous au connecteur

1. Accédez à connecteur d'attributs dynamiques à l'adresse `https://ip-address`
2. Connexion.

Utilisez pour la connexion initiale le nom d'utilisateur `admin`, le mot de passe `admin`. Vous devez modifier le mot de passe lors de votre première connexion.

Mettre à niveau le Connecteur d'attributs dynamiques Cisco Secure

Cette rubrique explique comment passer d'une version antérieure de Connecteur d'attributs dynamiques Cisco Secure à la version actuelle. Ces tâches peuvent être effectuées indépendamment de la version de Connecteur d'attributs dynamiques Cisco Secure ou du système d'exploitation.

Étape 1 Connectez-vous à la machine que vous souhaitez mettre à niveau.

Étape 2 Saisissez les commandes suivantes :

```
ansible-galaxy collection install cisco.csdac --force
ansible-playbook default_playbook.yml --ask-become-pass
ansible-playbook default_playbook.yml --ask-become-pass [--extra-vars vars]
```

Description de la syntaxe `--ask-become-pass` Vous êtes invité à saisir le mot de passe `sudo`. Obligatoire si `sudo` est activé sur votre machine.

--extra-vars Les variables optionnelles suivantes permettent au connecteur d'attributs dynamiques d'utiliser un proxy. La valeur utilisée doit correspondre à la valeur du fichier `/etc/environment`, que vous avez configuré comme indiqué dans la section [Installer les logiciels prérequis](#), à la page 2.

- **csdac_proxy_enabled=true**
- **csdac_http_proxy_url=http://PROXY_URL**
csdac_https_proxy_url=PROXY_URL

Les variables supplémentaires facultatives suivantes permettent de créer un certificat signé automatiquement que vous pouvez utiliser pour vous connecter en toute sécurité à l'application connecteur d'attributs dynamiques. Si vous omettez ces paramètres, connecteur d'attributs dynamiques utilise un certificat par défaut.

- **csdac_certificate_domain**
nom de domaine pour le certificat autogénéré. La valeur par défaut est le nom d'hôte auto détecté de l'hôte (détecté par ansible).
- **csdac_certificate_country_name**
Code pays à deux lettres. (la valeur par défaut est `us`)
- **csdac_certificate_organization_name**
Nom de l'entreprise. (La valeur par défaut est `Cisco`)
- **csdac_certificate_organization_unit_name**
• Nom de l'unité organisationnelle (la valeur par défaut est `Cisco`)

Étape 3 Attendez que la mise à niveau soit terminée.

Étape 4 Les journaux de mise à niveau sont disponibles à l'emplacement suivant :

```
~/ansible/collections/ansible_collections/cisco/csdac/logs/csdac.log
```

Prochaine étape

Consultez [Créer un connecteur](#).

À propos de la traduction

Cisco peut fournir des traductions du présent contenu dans la langue locale pour certains endroits. Veuillez noter que des traductions sont fournies à titre informatif seulement et, en cas d'incohérence, la version anglaise du présent contenu prévaudra.