Ajouter des propriétés d'instance de cloud d'équipements via l'API REST

Publié: 2025-03-28

Les propriétés du cloud de l'appareil vous permettent d'afficher des informations sur votre environnement cloud dans le système ExtraHop. Vous pouvez identifier le nom, le type et l'ID de l'instance cloud d'un appareil, ainsi que le compte cloud auquel appartient l'appareil et l'ID du cloud privé virtuel dans lequel se trouve l'équipement.

Ce guide fournit des instructions pour ajouter une observation via l' explorateur d'API ExtraHop, un modèle AWS CloudFormation, une fonction AWS Lambda et un script Python pour Microsoft Azure. Si vous mettez à jour automatiquement les propriétés du cloud via l'API REST, vous pouvez récupérer en permanence des informations auprès de votre fournisseur de cloud pour vous assurer que les informations relatives à vos propriétés cloud sont toujours à jour.

Ajoutez des propriétés d'instance cloud via l'explorateur d'API ExtraHop

Avant de commencer

- Pour les capteurs et la console ExtraHop, vous devez disposer d'une clé API valide avec écriture complète privilèges ou supérieur. (Voir Générer une clé API).
- Pour RevealX 360, vous devez disposer d'informations d'identification d'API REST valides avec écriture complète privilèges ou supérieur. (Voir Création d'informations d'identification pour l'API REST).

1. Dans un navigateur, accédez à l'explorateur d'API REST.

   L'URL est le nom d'hôte ou l'adresse IP de votre sonde ou console, suivi par `/api/v1/explore/`. Par exemple, si votre nom d'hôte est seattle-eda, l'URL est `https://seattle-eda/api/v1/explore/`.
2. Entrez les informations dcessitude d'identification de votre API REST.
   - Pour les capteurs et la console ExtraHop, cliquez sur **Entrez la clé API** puis collez ou saisissez votre clé API dans **Clé API** champ.
   - Pour RevealX 360, cliquez sur **Entrez les identifiants de l'API** puis collez ou saisissez l'ID et le code secret de vos informations diciatives d'API dans le **IDENTIFIANT** et **Secret** champs.
3. Cliquez **Autoriser** puis cliquez sur **Fermer**.
4. Trouvez l'ID de l'équipement en recherchant son adresse MAC.
   a) Cliquez **Appareil** puis cliquez sur **POST /dispositifs/search**.
   b) Cliquez **Essayez-le**.
   c) Dans le champ du corps, spécifiez le JSON suivant, en remplaçant MACADDRESS par l'adresse MAC de votre équipement cloud :

   ```
   {
     "filter": {
       "field": "macaddr",
       "operand": "MACADDRESS",
       "operator": "="
     }
   }
   ```

   d) Cliquez **Envoyer la demande**.
   e) Dans la section Corps de la réponse, visualisez et enregistrez la valeur de `id` champ pour chaque équipement renvoyé.
5. Ajoutez les métadonnées de l'équipement cloud.
   a) Cliquez **PATCH /devices/ {id}**.

b) Cliquez **Essayez-le**.
c) Dans le `id` champ, spécifiez un identifiant.
d) Dans le champ du corps, spécifiez le JSON suivant, en remplaçant le `string` valeurs avec des propriétés issues de votre environnement cloud :

```
{
  "cloud_account": "string",
  "cloud_instance_id": "string",
  "cloud_instance_name": "string",
  "cloud_instance_type": "string",
  "vpc_id": "string"
}
```

e) Cliquez **Envoyer la demande**.

Ajoutez des propriétés AWS à RevealX 360 avec CloudFormation

Vous pouvez ajouter des propriétés d'instance de cloud d'équipements AWS à RevealX 360 à l'aide d'un modèle CloudFormation accessible au public sur Amazon S3. Le modèle CloudFormation crée une fonction Lambda qui extrait les propriétés de l'instance AWS EC2 et les envoie à RevealX 360 via l'API REST. La fonction Lambda mappe les interfaces réseau des instances EC2 aux périphériques découverts sur le système ExtraHop par adresse MAC.

Voici quelques considérations importantes concernant la fonction Lambda :

- Le service AWS EventBridge exécute la fonction Lambda toutes les 30 minutes.
- La fonction importe uniquement les propriétés des instances cloud pour les instances EC2.
- Vous devez déployer le modèle CloudFormation dans chaque compte AWS depuis lequel vous souhaitez importer des propriétés.
- Vous ne pouvez déployer la fonction que dans les régions AWS suivantes :
  - Est des États-Unis (Ohio)
  - USA Est (Virginie du Nord)
  - Ouest des États-Unis (Oregon)
  - USA Ouest (Californie du Nord)

  Pour plus d'informations sur l'ajout de propriétés AWS en dehors de ces régions, consultez Ajouter des propriétés AWS à RevealX Enterprise avec Lambda.
- RevealX Enterprise ne prend pas en charge le modèle CloudFormation. Pour plus d'informations sur l'importation de propriétés dans RevealX Enterprise, voir Ajouter des propriétés AWS à RevealX Enterprise avec Lambda.

Avant de commencer
Tu dois avoir informations d'identification valides pour l'API REST avec écriture complète privilèges ou supérieur.

1. Accédez à la page CloudFormation dans AWS.
2. Créez une pile CloudFormation à partir de l'URL Amazon S3 suivante :

   ```
   https://s3.us-east-2.amazonaws.com/ct.s.extrahoplabs/Public/MDS.yml
   ```

3. Configurez les variables suivantes :

   IDENTIFIANT API

   L'ID de vos informations d'identification de l'API REST RevealX 360.

   Secret de l'API

   Le secret de vos informations d'identification de l'API REST RevealX 360.

Nom du locataire

Le sous-domaine de votre console RevealX 360.

Pour plus d'informations sur la configuration d'une pile CloudFormation, consultez Documentation AWS.

Ajouter des propriétés AWS à RevealX Enterprise avec Lambda

Vous pouvez ajouter des propriétés d'instance de cloud d'équipements AWS à RevealX Enterprise à l'aide d' un exemple de script Python. Le script mappe les interfaces réseau des instances EC2 aux périphériques découverts sur le système ExtraHop par adresse MAC.

Note: Pour plus d'informations sur l'importation de propriétés AWS dans RevealX 360, consultez Ajoutez des propriétés AWS à RevealX 360 avec CloudFormation.

Le script est conçu pour être exécuté en tant que fonction Lambda dans AWS. Voici quelques points importants à prendre en compte lors de l'exécution du script dans AWS :

- Le script est conçu pour s'exécuter sur un intervalle de temps défini. Chaque fois que le script est exécuté, il analyse chaque instance du VPC et met à jour les périphériques correspondants dans le système ExtraHop. Pour plus d'informations sur la configuration d'une fonction Lambda pour qu'elle s'exécute périodiquement, consultez le didacticiel AWS ici.
- La fonction Lambda doit pouvoir accéder aux ressources de votre VPC. Pour plus d' informations, consultez le didacticiel AWS ici.
- La fonction Lambda doit disposer d'un accès en liste et en lecture à l' action DescribeInstances pour le service EC2. Pour plus d'informations, consultez le didacticiel AWS ici.

Note: Si le script renvoie un message d'erreur indiquant que la vérification du certificat TLS a échoué, assurez-vous que un certificat fiable a été ajouté à votre sonde ou à votre console. Vous pouvez également ajouter `verify=False` option permettant de contourner la vérification des certificats. Cependant, cette méthode n'est pas sûre et n'est pas recommandée. Le code suivant envoie une requête HTTP GET sans vérification du certificat :

```
requests.get(url, headers=headers, verify=False)
```

Avant de commencer

- Vous devez avoir clé API valide avec écriture complète privilèges ou supérieur.

1. Accédez à l'ExtraHop référentiel GitHub d'exemples de code et téléchargez le `add_cloud_props_lambda/add_cloud_props_lambda.py` fichier sur votre machine locale.
2. Dans un éditeur de texte, ouvrez le `add_cloud_props_lambda.py` archivez et remplacez les variables de configuration suivantes par des informations provenant de votre environnement :
   - **NOM D'HÔTE:** L'adresse IP privée ou le nom d'hôte de la sonde ou de l'instance EC2 de la console.
   - **UN RUCHER:** La clé d'API ExtraHop.
3. Ajoutez le `add_cloud_props_lambda.py` fichier dans un fichier zip avec `requests` Module Python.

   Le script importe le `requests` Module Python, qui n'est pas disponible par défaut pour les fonctions Lambda. Pour plus d'informations sur la création d'un fichier zip pour importer des bibliothèques tierces dans Lambda, consultez Documentation AWS.
4. Dans AWS, créez une fonction Lambda.

   Pour plus d'informations sur la création de fonctions Lambda, consultez Documentation AWS.
5. Sur la page de la fonction Lambda, cliquez sur **Actions** et sélectionnez **Téléchargez un fichier .zip** dossier.
6. Sélectionnez le fichier zip que vous avez créé.

Ajouter des propriétés Azure à ExtraHop avec Python

Le référentiel GitHub ExtraHop contient un exemple de script Python qui importe les propriétés des équipements Azure dans le système ExtraHop. Le script attribue des propriétés d'équipement cloud à chaque équipement découvert par le système ExtraHop avec une adresse MAC appartenant à une interface réseau Azure VM. Le script est conçu pour être exécuté selon un intervalle de temps défini. Chaque fois que le script est exécuté, il analyse chaque machine virtuelle et met à jour les périphériques correspondants dans ExtraHop.

Le script nécessite les modules suivants du SDK Azure Python :

- azure.mgmt.compute
- réseau azure.mgmt.network
- informations d'identification azure.common

Le script nécessite également que vous ayez configuré les identifiants dauthentification Azure dans les variables d'environnement suivantes sur la machine qui exécute le script :

- IDENTIFIANT D'ABONNEMENT AZURE
- IDENTIFIANT_CLIENT AZURE
- AZURE_CLIENT_SECRET
- IDENTIFIANT DU LOCATAIRE AZURE

Pour plus d'informations sur la génération de ces informations dicitives, consultez Documentation Azure.

Important: L'exemple de script python s'authentifie auprès de la sonde ou de la console via une clé API, qui n'est pas compatible avec l'API REST RevealX 360 . Pour exécuter ce script avec RevealX 360, vous devez le modifier pour vous authentifier à l'aide de jetons d'API. Consultez les py_rx360_auth.py script dans le référentiel GitHub d'ExtraHop pour un exemple d'authentification à l'aide de jetons d'API.

1. Accédez au Référentiel GitHub d'exemples de code ExtraHop et téléchargez le `add_cloud_props_azure/add_cloud_props_azure.py` fichier sur votre machine locale.
2. Dans un éditeur de texte, ouvrez `add_cloud_props_azure.py` archivez et remplacez les variables de configuration suivantes par des informations provenant de votre environnement :
   - **NOM D'HÔTE:** L'adresse IP ou le nom d'hôte de la sonde ou de la console.
   - **UN RUCHER:** La clé d'API ExtraHop.
3. Exécutez la commande suivante :

```
python3 add_cloud_props_azure.py
```

Note: Si le script renvoie un message d'erreur indiquant que la vérification du certificat TLS a échoué, assurez-vous que un certificat fiable a été ajouté à votre sonde ou à votre console. Vous pouvez également ajouter `verify=False` option permettant de contourner la vérification des certificats. Cependant, cette méthode n'est pas sûre et n'est pas recommandée. Le code suivant envoie une requête HTTP GET sans vérification du certificat :

```
requests.get(url, headers=headers, verify=False)
```