Migrer les règles d'exceptions

Publié: 2024-11-04

Vous pouvez migrer les règles d'exceptions depuis un sonde ou console vers un autre via l'API REST. Cela peut être utile si vous avez créé un grand nombre de règles d'exceptions et que vous ne souhaitez pas les recréer manuellement. Dans cette rubrique, nous présentons des méthodes permettant de migrer une règle manuellement via l'explorateur d'API REST et de migrer des règles à l'aide de scripts Python. Un exemple de script migre les règles entre deux machines virtuelles ECA et un exemple de script migre les règles d'une machine virtuelle ECA vers RevealX 360.

Avant de commencer

- Les capteurs ou les consoles doivent exécuter la version 8.4 ou ultérieure du microprogramme.
- Si vous migrez des règles d'exceptions qui font référence à des groupes d'équipements, envisagez de migrer ces groupes d'appareils avec un bundle. Tu peux créer un bundle avec les groupes d'équipements du système source et installer le bundle sur le système cible.
- Pour les capteurs et les machines virtuelles ECA, vous devez disposer d'une clé API valide pour apporter des modifications via l' API REST et suivre les procédures ci-dessous. (Voir Générer une clé API).
- Pour RevealX 360, vous devez disposer d'informations d'identification d'API REST valides pour apporter des modifications via l' API REST et suivre les procédures ci-dessous. (Voir Création d'informations d'identification pour l'API REST).

Migrer une règle de réglage via l'explorateur d'API REST

1. Récupérez les métadonnées des règles de réglage depuis le système source.
   a) 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/`.
   b) Entrez les informations d'identification de votre API REST.

      - Pour les capteurs et les machines virtuelles ECA, cliquez **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 dìidentification API dans le **IDENTIFIANT** et **Secret** champs.

   c) Cliquez **Autoriser** puis cliquez sur **Fermer**.
   d) Cliquez **Détections**.
   e) Cliquez **GET /detections/rules/masquage**.
   f) Cliquez **Essayez-le**.
   g) Cliquez **Envoyer la demande**.
   h) Dans le champ Corps de la réponse, copiez l'objet JSON qui représente la règle de réglage que vous souhaitez copier.
2. Recréez la règle de réglage sur le système cible.
   a) 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/`.
   b) Entrez les informations d'identification de votre API REST.

      - Pour les capteurs et les machines virtuelles ECA, cliquez **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 dìidentification API dans le **IDENTIFIANT** et **Secret** champs.

c) Cliquez **Autoriser** puis cliquez sur **Fermer**.
d) Cliquez **Détections**.
e) Cliquez **POST /détections/règles/masquage**.
f) Cliquez **Essayez-le**.
g) Dans la zone de texte du corps, collez l'objet JSON que vous avez copié depuis la source sonde ou console.

L'entrée doit ressembler au texte suivant :

```
{
    "id": 1,
    "enabled": false,
    "detection_type": "cifs_round_trip_time",
    "offender": {
      "object_type": "device",
      "object_id": 123
    },
    "victim": {
      "object_type": "device",
      "object_id": 321
    },
    "author": "example_user",
    "create_time": 1615588932838,
    "expiration": 1615675096000,
    "detections_hidden": 0
}
```

Note: Si le `description` ou `properties` le champ est défini sur null `null`, vous devez supprimer ces champs du JSON avant d'envoyer la demande.

h) Cliquez **Envoyer la demande**.

3. Optionnel : Désactivez la règle de réglage sur le système cible.

Si la règle de réglage a été désactivée sur le système source, indiqué par le champ activé défini sur `false`, définissez le champ activé sur False sur le système cible.

a) 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/`.

b) Entrez les informations d'identification de votre API REST.

- Pour les capteurs et les machines virtuelles ECA, cliquez **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 dìidentification API dans le **IDENTIFIANT** et **Secret** champs.

c) Cliquez **Autoriser** puis cliquez sur **Fermer**.
d) Cliquez **Détections**.
e) Cliquez **PATCH /détections/règles/masquage**.
f) Cliquez **Essayez-le**.
g) Dans le corps du texte, collez le code JSON suivant :

```
{
    "enabled": false
}
```

h) Cliquez **Envoyer la demande**.

Récupérez et exécutez l'exemple de script Python pour RevealX 360

Le référentiel GitHub ExtraHop contient un exemple de script Python qui migre toutes les règles d'exceptions d'une machine virtuelle ECA vers RevealX 360.

Note: Le script migre uniquement les règles qui sont activées.

1. Accédez au Référentiel GitHub d'exemples de code ExtraHop et téléchargez le `migrate_detection_hiding/migrate_detection_hiding.py` fichier sur votre machine locale.
2. Dans un éditeur de texte, ouvrez `migrate_detection_hiding.py` archivez et remplacez les variables de configuration suivantes par des informations provenant de votre environnement :

   • **HÔTE SOURCE:** Le nom d'hôte de la machine virtuelle ECA à partir de laquelle vous migrez les règles d'exceptions

   • **CLÉ_SOURCE DE L'API:** La clé API de la machine virtuelle ECA à partir de laquelle vous migrez les règles d'exceptions

   • **HÔTE_CIBLE:** Le nom d'hôte de l'API RevealX 360 vers laquelle vous migrez les règles d'exceptions. Ce nom d'hôte est affiché sur la page d'accès à l'API RevealX 360 sous API Endpoint. Le nom d'hôte n'inclut pas `/oauth2/token`.

   • **IDENTIFIANT_CIBLE:** L'ID des informations dcessaires d'identification de l'API REST pour RevealX 360

   • **CIBLE_SECRET:** Le secret des informations dcessaires d'identification de l'API REST pour RevealX 360
3. Exécutez la commande suivante :

   ```
   python3 migrate_detection_hiding.py
   ```

   Si les règles d'exceptions spécifient les appareils participants ou les groupes d'appareils par un ID, le script essaie de trouver les identifiants des participants équivalents sur RevealX 360 en recherchant les adresses IP des appareils et les noms des groupes d'appareils.

   Si le script ne trouve pas les identifiants des participants équivalents sur RevealX 360, il vous invite à migrer les autres règles pour lesquelles des participants équivalents ont été trouvés. Pour continuer, tapez y et appuyez sur ENTER.

   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)
   ```

Récupérez et exécutez l'exemple de script Python pour RevealX Enterprise

Le référentiel GitHub ExtraHop contient un exemple de script Python qui migre toutes les règles d'exceptions d'une machine virtuelle ECA vers une autre machine virtuelle ECA.

Note: Le script migre uniquement les règles qui sont activées.

1. Accédez au Référentiel GitHub d'exemples de code ExtraHop et téléchargez le `migrate_detection_hiding/migrate_detection_hiding_enterprise.py` fichier sur votre machine locale.
2. Dans un éditeur de texte, ouvrez `migrate_detection_hiding_enterprise.py` archivez et remplacez les variables de configuration suivantes par des informations provenant de votre environnement :

   • **HÔTE SOURCE:** Le nom d'hôte de la machine virtuelle ECA à partir de laquelle vous migrez les règles d'exceptions

   • **CLÉ_SOURCE DE L'API:** La clé API de la machine virtuelle ECA à partir de laquelle vous migrez les règles d'exceptions

   • **HÔTE_CIBLE:** Le nom d'hôte de la machine virtuelle ECA vers laquelle vous migrez les règles d'exceptions

   • **CLÉ_API_CIBLE:** La clé d'API de la machine virtuelle ECA vers laquelle vous migrez les règles d'exceptions
3. Exécutez la commande suivante :

   ```
   python3 migrate_detection_hiding_enterprise.py
   ```

   Si les règles d'exceptions spécifient les appareils participants ou les groupes d'appareils par un ID, le script essaie de trouver les ID des participants équivalents sur la machine virtuelle ECA cible en recherchant les adresses IP des équipements et les noms des groupes d'appareils.

   Si le script ne trouve pas les ID des participants équivalents sur la machine virtuelle ECA cible, il vous invite à migrer les autres règles pour lesquelles des participants équivalents ont été trouvés. Pour continuer, tapez y et appuyez sur ENTER.

   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)
   ```