RECEVOIR VOTRE PISTE D'AUDIT DANS

CrowdStrike Falcon

Recevez la piste d'audit REVIEWIT dans CrowdStrike Falcon Next-Gen SIEM (et Falcon LogScale) par son connecteur de données HEC : la clé d'API du connecteur, confiée une fois, aucun agent, aucun composant intermédiaire.

VERSION
1.0 · Octobre 2026

DURÉE
Trente minutes, en trois étapes

DIFFUSION
Confidentiel · Clients REVIEWIT

CE QU'IL VOUS FAUT

01 **Un administrateur Falcon** avec le droit de créer un connecteur de données dans Next-Gen SIEM (Data onboarding › Data connectors), et le dépôt (repository) qui recevra la piste.

02 **Un administrateur de la souscription** REVIEWIT. Créer, activer et remplacer un identifiant exigent une authentification forte de moins de dix minutes : la console vous envoie chez votre fournisseur d'identité et vous ramène.

03 **L'ordre de ce guide** : d'abord la cible chez vous, puis la destination dans REVIEWIT, puis le test. Rien n'est livré avant qu'une écriture réelle sous vos identifiants ait réussi.

LE PARCOURS, EN TROIS ÉTAPES

ÉTAPE 1

Préparer la cible

Créer le point de réception et l'identifiant que REVIEWIT utilisera — droits minimaux, rien de plus.

ÉTAPE 2

Déclarer la destination

Quatre écrans dans le Subscription Manager : cible, résidence, contenu, récapitulatif.

ÉTAPE 3

Tester, puis activer

Une écriture de sonde sous vos identifiants ; un refus dit pourquoi. La livraison commence dans la minute.

DANS CE GUIDE

1 Comment BEACON livre votre piste d'audit

2 Ce que vous recevez

3 Préparer la cible chez CrowdStrike Falcon

4 Déclarer la destination dans REVIEWIT

5 Tester et activer

6 Vérifier hors ligne avec beacon-verify

7 Exploiter

8 Dépannage et liste de contrôle

01 Comment BEACON livre votre piste d'audit

BEACON est le canal de diffusion de la piste d'audit REVIEWIT. Il ne produit pas la preuve et ne possède pas les événements : **la piste d'audit prouve, REVIEWIT SEAL scelle, BEACON livre.** Chaque événement d'audit devient un maillon — un rang strictement croissant, le SHA-256 de l'enregistrement, l'empreinte du maillon précédent et l'enregistrement original verbatim — et la tête de chaque chaîne est signée par l'autorité de scellement. Ce qui arrive chez vous est une chaîne que vous recalculez et un sceau que vous contrôlez, sans jamais rappeler REVIEWIT.

VOTRE RÉGION

Piste d'audit

Événements chaînés par domaine et classe de rétention, scellés toutes les heures par REVIEWIT SEAL.

→

VOTRE RÉGION

BEACON

Une passe par minute : des lots de 500 événements ou 5 Mo au plus, un manifeste par lot, un battement toutes les 15 minutes.

→

CHEZ VOUS

CrowdStrike Falcon

Le connecteur de données HEC de votre Falcon Next-Gen SIEM, sous la clé d'API du connecteur.

PRINCIPE DIRECTEUR

Rien n'est perdu et rien n'est prétendu. La livraison avance chaîne par chaîne, sur un acquittement strict seulement ; un échec reprend exactement où il s'est arrêté ; après douze tentatives, la destination est mise en pause et une alerte est levée. Une purge de rétention est livrée comme une coupe signée, un filtre laisse des jalons qui gardent la chaîne vérifiable, et le silence est couvert par le battement : jamais de trou non signé.

PRIS EN CHARGE PAR

La carte **CrowdStrike** de l'assistant de destination — un préréglage du profil HTTP Event Collector : même enveloppe, même transport, même acquittement strict (`{"code":0}`). Authentification : BEACON envoie `Authorization: Splunk <clé>` ; si le connecteur répond 401, il rejoue une fois en `Bearer` et retient le schéma pour la passe. Rien à configurer. Le profil HEC est prouvé de bout en bout sur Splunk ; la preuve sur un tenant Falcon est programmée avec l'accès d'essai de REVIEWIT — voir le dépannage si votre connecteur acquitte autrement.

02 Ce que vous recevez

Trois sortes d'enregistrements voyagent vers votre cible, toutes en JSON UTF-8 dont l'ordre des propriétés est le contrat. Un **maillon** est un événement d'audit : `metadata.uid` (l'identifiant stable de l'événement), `reviewit.chain` (clé, genèse), `reviewit.integrity` (`sequence`, `previous_hash`, `hash`, `raw_data`). Un **manifeste** clôt chaque lot : chaîne, plage, empreintes d'entrée et de sortie, décomptes, `batch_id`, et la référence du dernier sceau de la chaîne. Un **battement** porte, toutes les 15 minutes, la tête et la position acquittée de chaque chaîne.

ENREGISTREMENT	OÙ IL ARRIVE	À QUOI IL SERT
`reviewit:audit`	Un événement par maillon dans le dépôt du connecteur ; la charge de l'événement est le maillon	L'événement d'audit, avec ses champs de preuve et `raw_data` verbatim
`reviewit:audit:manifest`	Un événement par lot, manifeste nu	Plage, empreintes, décomptes, `batch_id`, référence de sceau
`reviewit:audit:heartbeat`	Un événement toutes les 15 minutes	L'état de chaque chaîne — alertez sur son absence

AU MOINS UNE FOIS, DÉDOUBLONNÉ PAR VOUS

Après un doute réseau, le même lot peut être présenté de nouveau, avec le même `batch_id` et les mêmes identifiants d'événements. Dédoublonnez sur `metadata.uid`, et ordonnez entre chaînes par l'heure de l'occurrence. Un enregistrement qui n'arrive jamais est un trou que votre SIEM peut voir : chaque maillon est numéroté, et le battement dit où se tient la tête.

LE PULL, MÊME CONTRAT

Votre SIEM peut aussi lire la piste lui-même par l'API publique REVIEWIT, sous la portée `audit.read` : `GET /api/v1/audit/chains` liste les chaînes de votre souscription avec leur curseur de départ, `GET /api/v1/audit/stream?cursor=…&limit=500&manifest=true` rend les mêmes maillons et les mêmes manifestes que le push. Push et pull se vérifient avec le même outil.

03 Préparer la cible chez CrowdStrike Falcon

Un administrateur Falcon crée **un connecteur de données HEC dédié à REVIEWIT**. Falcon affiche l'URL du connecteur et la clé d'API une seule fois ; les deux vont dans l'assistant REVIEWIT et nulle part ailleurs.

01 **Next-Gen SIEM › Data onboarding › Data connectors › Add connection**

Cherchez **HEC / HTTP Event Collector** et ouvrez-le.

02 **Réglages du connecteur**

Name : **REVIEWIT BEACON**. Parser : **none** (ou un parser JSON) : chaque événement est un objet JSON ; les champs du maillon s'interrogent comme `reviewit.chain.key`, `reviewit.integrity.sequence`, et `raw_data` porte l'enregistrement original verbatim. Repository : celui qui tiendra la piste, avec la rétention voulue.

03 **Enregistrez, puis copiez l'URL d'API et la clé d'API**

Falcon affiche l'**URL d'API** du connecteur — `https://<instance>.ingest.<région>.crowdstrike.com/api/ingest/hec/<id>/v1/services/collector` — et la **clé d'API**, une fois. La clé se saisit dans l'assistant REVIEWIT (section 4) et n'est plus jamais affichée d'un côté ni de l'autre.

04 **Falcon LogScale (auto-géré ou cloud)**

Même chemin depuis le dépôt : Settings › Ingest tokens › New token (parser : none). L'URL du collecteur est `https://<votre hôte logscale>/api/v1/ingest/hec` — saisissez-la telle quelle, ou son origine : BEACON complète une URL qui se termine par `/services/collector` et utilise une URL complète sans y toucher.

CE QUE VOUS SAISIREZ DANS L'ASSISTANT

CHAMP DE L'ASSISTANT	VALEUR
Transport	**CrowdStrike** (Falcon Next-Gen SIEM — connecteur de données HEC)
Nom	Comment la destination se lira dans votre liste — « Falcon — SOC ».
URL de la cible	L'URL de connecteur affichée par Falcon (`…/api/ingest/hec/<id>/v1/services/collector`), utilisée telle quelle. Pas de requête, pas d'identifiant dans l'URL.
Clé d'API	La clé que Falcon a montrée une fois à la création.
Géographie de la cible	La région de votre instance Falcon (US-1, US-2, EU-1, US-GOV-1…). Déclarative.

04 Déclarer la destination dans REVIEWIT

Subscription Manager › **Paramètres** › **Intégrations & IA** › **BEACON** › **Nouvelle destination**. L'assistant a quatre écrans ; rien n'est créé avant le dernier, et la cible est contrôlée par la garde d'egress avant que la destination existe.

01 Cible

Nommez la destination, choisissez la carte **CrowdStrike**, saisissez l'URL de la cible et l'identifiant de la section 3. L'identifiant est ouvert en mémoire pour ce seul geste, puis scellé sous garde REVIEWIT SEAL : l'écran ne le montre plus jamais.

02 Résidence

Déclarez la **région de votre instance Falcon** — l'hôte de l'URL du connecteur la nomme (`ingest.us-1`, `ingest.eu-1`…). Pour un point HTTP elle est déclarative. Si elle diffère de la géographie de votre souscription REVIEWIT, une justification écrite est exigée et consignée dans la piste d'audit.

03 Contenu

Départ : depuis le début de chaque chaîne (historique complet) ou depuis l'activation (seulement ce qui suit). **Événements envoyés en clair** : laissez vide pour le flux complet. Un filtre remplace chaque événement écarté par un jalon — rang et empreintes, aucune donnée métier — et la preuve scellée s'arrête au premier jalon ; l'écran le dit avant que vous confirmiez.

04 Récapitulatif

Relisez, puis cliquez **Créer**. La garde d'egress juge d'abord la cible : HTTPS seulement, aucun identifiant dans l'URL, port 443 ou 8088, un hôte routable publiquement. Une cible refusée nomme le contrôle fautif ; rien n'est créé.

LA GARDE D'EGRESS, AVANT CHAQUE ENVOI

Chaque cible passe les mêmes sept contrôles à la création, à chaque test et avant chaque livraison : URL valide · HTTPS · aucun identifiant dans l'URL · port 443 ou 8088 · aucune zone interne REVIEWIT · l'hôte se résout · l'hôte n'est pas une adresse privée ou réservée. Les redirections ne sont jamais suivies, et la vérification TLS n'est jamais désactivée.

05 Tester et activer

Une destination n'est activée qu'après une écriture réelle sous vos identifiants. Ouvrez le panneau de la destination et cliquez **Tester la cible** : REVIEWIT écrit un enregistrement de sonde (un événement de test sous la clé du connecteur). « La cible passe la garde d'egress » signifie que l'hôte, l'identifiant et ses droits sont prouvés. Un refus dit sa cause — jamais le secret.

01 Tester la cible

REVIEWIT poste un événement à l'URL du connecteur sous la clé — schéma `Splunk` d'abord, `Bearer` sur un 401 — et attend une réponse 2xx avec `{"code":0}`. Un refus nomme sa cause : **HEC HTTP 401/403** — clé invalide ou connecteur désactivé ; **HEC HTTP 404** — mauvaise URL de connecteur (recopiez-la depuis Falcon) ; **HTTP 2xx sans acquittement** — le connecteur a répondu sans champ `code` (voir le dépannage).

02 Activer

Depuis le panneau, **Activer** (ou cochez « Activer dès la création » dans l'assistant). La livraison commence à la passe suivante, dans la minute ; le premier battement suit.

03 Lire le panneau

Par chaîne : le point de reprise acquitté, l'âge du plus ancien événement non livré, la marge avant la prochaine coupe de rétention, le dernier sceau et ce qui est au-delà. L'historique des lots liste chaque manifeste ; le journal du canal liste chaque geste d'administration avec son code d'audit.

06 Vérifier hors ligne avec beacon-verify

`beacon-verify` est un outil en ligne de commande, distribué avec sa spécification publique. Il lit ce que votre cible a reçu et le trousseau que REVIEWIT publie à `/.well-known/seal-keyring.json`, ancré sur l'empreinte de la clé racine imprimée dans l'annexe sécurité de votre contrat, et rend un verdict par chaîne et par plage — sans réseau et sans appel à REVIEWIT.

THE COMMAND

```
beacon-verify <export> --keys seal-keyring.json --root <empreinte racine> --namespace <v
otre espace de scellement>
```

Votre **espace de scellement** est affiché dans le panneau de la destination (`t-` suivi de 26 caractères). Passez-le avec `--namespace` : un sceau d'un autre espace est, pour vous, une rupture. Sans `--root` ni `--namespace`, les plages sont rendues scellées *sous réserve*.

01 **Chercher dans le dépôt sur une fenêtre d'ingestion**

`#repo=<dépôt> #sourcetype=reviewit:audit*` avec le sélecteur de temps sur l'horodatage d'**ingestion** (`@ingesttimestamp`), assez large pour contenir des lots entiers. Les maillons portent l'heure de l'occurrence dans `@timestamp` ; manifestes et jalons portent l'heure d'ingestion : ne bornez jamais l'export par `@timestamp`.

02 **Exporter les événements bruts**

Export › **JSON** ou **NDJSON** du champ `@rawstring` — un objet JSON par ligne, exactement tel que reçu. N'exportez pas une vue tableau : une valeur à la fois brute et enveloppée est refusée par le vérificateur (code de sortie 2).

03 **Lancer le vérificateur**

`beacon-verify export.ndjson --keys seal-keyring.json --root <racine> --namespace <t-…>`. Code 0 quand tout ce qui précède le dernier sceau est scellé ; 3 quand une chaîne n'a pas encore de sceau sous votre espace (attendu juste après l'activation — le prochain sceau horaire la couvre).

VERDICT	SIGNIFICATION
`SCELLÉ`	Chaîné jusqu'à une tête dont la signature est valide, trousseau ancré, espace épinglé.
`SCELLÉ SOUS RÉSERVE`	Idem, mais la racine ou l'espace n'ont pas été épinglés sur la ligne de commande.
`CHAÎNAGE SEUL`	Intact, au-delà du dernier sceau ou derrière un jalon — le prochain sceau le couvrira.
`NON VÉRIFIABLE`	Un sceau sans espace, un algorithme inconnu ou une clé absente du trousseau — jamais dit faux.
`ROMPU`	Une empreinte fausse, un trou, un doublon divergent, une signature invalide ou un espace étranger — avec la cause et le rang.

Codes de sortie : **0** rien de rompu et tout ce qui précède le dernier sceau est scellé · **1** une rupture ou un espace étranger · **2** entrée inexploitable (une propriété JSON en double, une valeur à la fois brute et enveloppée, un enregistrement tronqué à l'indexation) · **3** rien de rompu, preuve incomplète.

07 Exploiter

Chaque geste sur une destination est posé par un administrateur de la souscription, tracé dans la piste d'audit sous son nom, et les gestes sensibles exigent une authentification forte de moins de dix minutes. Le canal qui exporte la piste est lui-même sous la piste.

Faire tourner la clé d'API	Créez une nouvelle clé de connecteur dans Falcon (ou un nouveau connecteur), puis panneau › **Remplacer la clé d'API**. Le remplacement est sec : l'ancienne clé cesse d'être utilisée aussitôt. Révoquez-la ensuite dans Falcon.
Pause pour échecs	Un refus persistant — identifiant révoqué, droit retiré, cible disparue — met la destination en pause après douze tentatives et lève une alerte dans le Centre d'alertes. Rien n'est perdu : la reprise repart du point de reprise acquitté de chaque chaîne, automatiquement quand la sonde horaire revoit la cible, ou par **Reprendre**.
Changer de cible	**Modifier** › nouvelle URL, même forme. La clé est conservée.
Retirer	**Désactiver**, puis supprimer ou archiver. Une destination qui a livré est archivée, jamais supprimée : ses manifestes sont des preuves de livraison. Révoquez ensuite l'identifiant de votre côté : il ne sert plus à rien.

08 Dépannage

CE QUE VOUS VOYEZ	CAUSE	CE QU'IL FAUT FAIRE
Chaque envoi refusé en HEC HTTP 401	La clé a été révoquée, ou le connecteur désactivé	Créez une nouvelle clé ; remplacez-la dans le panneau ; **Reprendre**.
HEC HTTP 404 à chaque lot	L'URL de la cible n'est pas l'URL de connecteur que Falcon affiche	**Modifier** › recollez l'URL du connecteur, inchangée ; **Tester la cible**.
« HEC HTTP 200 sans acquittement (code absent) »	Le connecteur a acquitté sans le corps standard `{"code":0}`	Envoyez au support REVIEWIT le corps exact de la réponse : le profil HEC n'acquitte que sur `code: 0`, par construction, et l'adaptation à la réponse de votre connecteur est une évolution du produit, pas un réglage.
Destination en pause, « après échecs »	Douze refus consécutifs (clé, URL, réseau)	Lisez la cause dans le panneau, corrigez, **Tester la cible**, puis **Reprendre**. Rien n'a été perdu.
beacon-verify rend ROMPU sur un export qui semble complet	L'export a été borné par `@timestamp`, ou tronqué à l'ingestion	Exportez par fenêtre d'ingestion, événements bruts seulement ; relancez. Si la rupture tient, contactez le support REVIEWIT avec la clé de chaîne et le rang.
La garde d'egress refuse la cible	Une requête ou un identifiant dans l'URL, ou un hôte non public	Collez l'URL du connecteur exactement comme Falcon l'affiche ; les hôtes d'ingestion Falcon sont publics en 443.

Liste de contrôle avant de commencer

6 POINTS

- [ ] Un connecteur de données HEC dédié à REVIEWIT, parser none ou JSON
- [ ] Son URL de connecteur et sa clé d'API, copiées une fois depuis Falcon
- [ ] Un dépôt avec la rétention voulue
- [ ] La région de votre instance Falcon connue, et sa justification si elle diffère de celle de votre souscription
- [ ] Un administrateur de la souscription avec son second facteur prêt
- [ ] L'empreinte racine de l'annexe de votre contrat, pour beacon-verify

Ce guide est fourni aux clients REVIEWIT qui raccordent leur SIEM ou leur stockage à BEACON. Les chemins de menus sont ceux des consoles des éditeurs à la date de publication ; ils peuvent évoluer. En cas d'écart, le support REVIEWIT vous accompagne.