

Ticket-Tracking für Erkennungen konfigurieren

Veröffentlicht: 2023-10-24

Mit der Ticketverfolgung können Sie Tickets, Alarme oder Fälle in Ihrem Work-Tracking-System mit ExtraHop-Erkennungen verbinden. Jedes Ticketsystem von Drittanbietern, das Open Data Stream (ODS) -Anfragen akzeptieren kann, wie Jira oder Salesforce, kann mit ExtraHop-Erkennungen verknüpft werden.

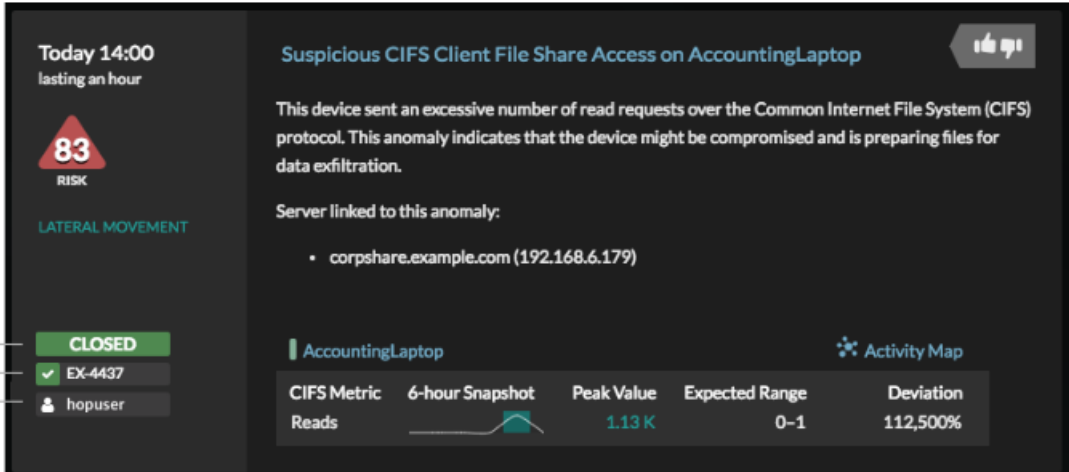
Bevor Sie beginnen

- Sie benötigen Zugriff auf ein ExtraHop-System mit einem Benutzerkonto, das [Rechte für die System- und Zugriffsadministration](#).
- Sie müssen mit dem Schreiben von ExtraHop-Triggern vertraut sein. siehe [Auslöser](#) und die Verfahren in [Einen Auslöser erstellen](#).
- Sie müssen ein ODS-Ziel für Ihren Ticket-Tracking-Server erstellen. Weitere Informationen zur Konfiguration von ODS-Zielen finden Sie in den folgenden Themen : [HTTP](#), [Kafka](#), [MongoDB](#), [Syslog](#), oder [Rohdaten](#).
- Sie müssen mit dem Schreiben von REST-API-Skripts vertraut sein und über einen gültigen API-Schlüssel verfügen, um die folgenden Verfahren ausführen zu können. siehe [Generieren Sie einen API-Schlüssel](#).

Aktivieren Sie die Ticketverfolgung und geben Sie eine URL-Vorlage an

Sie müssen die Ticketverfolgung aktivieren, bevor REST-API-Skripts Ticketinformationen auf dem ExtraHop-System aktualisieren können. Geben Sie optional eine URL-Vorlage an, die dem Ticket in Ihrem Ticketsystem einen HTML-Link auf der Erkennungskarte hinzufügt.

1. Loggen Sie sich in die Administrationseinstellungen des ExtraHop-Systems ein über `https://<extrahop-hostname-or-IP-address>/admin`.
2. In der Konfiguration des Systems Abschnitt, klicken **Erkennungsverfolgung**.
3. Wählen **Verfolgen Sie Erkennungen von einem externen Ticketsystem**.
4. Optional: Geben Sie im Feld URL die URL-Vorlage für Ihr Ticketsystem an und fügen Sie die `$ticket_id` variabel an der entsprechenden Stelle. Geben Sie beispielsweise eine vollständige URL ein, z. B. `https://jira.example.com/browse/$ticket_id`. Die `$ticket_id` Die Variable wird durch die der Erkennung zugeordnete Ticket-ID ersetzt.



The screenshot shows an alert card with the following details:

- Time:** Today 14:00, lasting an hour.
- Risk Level:** 83 RISK (indicated by a red triangle icon).
- Category:** LATERAL MOVEMENT.
- Title:** Suspicious CIFS Client File Share Access on AccountingLaptop.
- Description:** This device sent an excessive number of read requests over the Common Internet File System (CIFS) protocol. This anomaly indicates that the device might be compromised and is preparing files for data exfiltration.
- Server linked to this anomaly:**
 - corpshare.example.com (192.168.6.179)
- Activity Map:** AccountingLaptop.
- Table:**


CIFS Metric	6-hour Snapshot	Peak Value	Expected Range	Deviation
Reads		1.13 K	0-1	112,500%
- Metadata:**
 - Status: CLOSED
 - Ticket ID: EX-4437
 - Assignee: hopuser

Schreiben Sie einen Auslöser, um Tickets zu Erkennungen in Ihrem Ticketsystem zu erstellen und zu aktualisieren


Dieses Beispiel zeigt Ihnen, wie Sie einen Auslöser erstellen, der die folgenden Aktionen ausführt:

- Erstellen Sie jedes Mal, wenn eine neue Erkennung im ExtraHop-System erscheint, ein neues Ticket im Ticketsystem.
- Weisen Sie einem Benutzer mit dem Namen neue Tickets zu `escalations_team` im Ticketsystem.
- Wird jedes Mal ausgeführt, wenn eine Erkennung auf dem ExtraHop-System aktualisiert wird.
- Senden Sie Erkennungsaktualisierungen über einen HTTP Open Data Stream (ODS) an das Ticketsystem.

Das vollständige Beispielskript ist am Ende dieses Themas verfügbar.

1. Loggen Sie sich in das ExtraHop-System ein über `https://<extrahop-hostname-or-IP-address>`.
2. Klicken Sie auf das Symbol Systemeinstellungen  und dann klicken **Auslöser**.
3. klicken **Neu**.
4. Geben Sie einen Namen und eine optionale Beschreibung für den Auslöser an.
5. Wählen Sie in der Liste Ereignisse **ERKENNUNGSUPDATE**.

Das Ereignis `DETECTION_UPDATE` wird jedes Mal ausgeführt, wenn eine Erkennung im ExtraHop-System erstellt oder aktualisiert wird.

6. Geben Sie im rechten Bereich Folgendes an **Erkennungsklasse**  Parameter in einem JavaScript-Objekt. Diese Parameter bestimmen die Informationen, die an Ihr Ticketsystem gesendet werden.

Der folgende Beispielcode fügt die Erkennungs-ID, die Beschreibung, den Titel, die Kategorien, die MITRE-Techniken und -Taktiken sowie die Risikoscore zu einem JavaScript-Objekt mit dem Namen `payload`:

```
const summary = "ExtraHop Detection: " + Detection.id + ": " +
  Detection.title;
const description = "ExtraHop has detected the following event on your
  network: " + Detection.description
const payload = {
  "fields": {
    "summary": summary,
    "assignee": {
      "name": "escalations_team"
    },
    "reporter": {
      "name": "ExtraHop"
    },
    "priority": {
      "id": Detection.riskScore
    },
    "labels": Detection.categories,
    "mitreCategories": Detection.mitreCategories,
    "description": description
  }
};
```

7. Definieren Sie als Nächstes die HTTP-Anforderungsparameter in einem JavaScript-Objekt unter dem vorherigen JavaScript-Objekt.

Der folgende Beispielcode definiert eine HTTP-Anfrage für die im vorherigen Beispiel beschriebene Nutzlast: definiert eine Anfrage mit einer JSON-Payload:

```
const req = {
  'path': '/rest/api/issue',
```

```
'headers': {
    'Content-Type': 'application/json'
  },
  'payload': JSON.stringify(payload)
};
```

Weitere Hinweise zu ODS-Anforderungsobjekten finden Sie unter [Offene Datenstromklassen](#).

8. Geben Sie abschließend die HTTP-POST-Anfrage an, die die Informationen an das ODS-Ziel sendet. Der folgende Beispielcode sendet die im vorherigen Beispiel beschriebene HTTP-Anfrage an ein ODS-Ziel namens Ticket-Server:

```
Remote.HTTP('ticket-server').post(req);
```

Der vollständige Triggercode sollte dem folgenden Beispiel ähneln:

```
const summary = "ExtraHop Detection: " + Detection.id + ": " +
  Detection.title;
const description = "ExtraHop has detected the following event on your
  network: " + Detection.description
const payload = {
  "fields": {
    "summary": summary,
    "assignee": {
      "name": "escalations_team"
    },
    "reporter": {
      "name": "ExtraHop"
    },
    "priority": {
      "id": Detection.riskScore
    },
    "labels": Detection.categories,
    "mitreCategories": Detection.mitreCategories,
    "description": description
  }
};

const req = {
  'path': '/rest/api/issue',
  'headers': {
    'Content-Type': 'application/json'
  },
  'payload': JSON.stringify(payload)
};

Remote.HTTP('ticket-server').post(req);
```

Senden Sie Ticketinformationen über die REST-API an Erkennungen

Nachdem Sie in Ihrem Ticket-Tracking-System einen Auslöser zum Erstellen von Tickets für Erkennungen konfiguriert haben, können Sie die Ticketinformationen auf Ihrem ExtraHop-System über die REST-API aktualisieren.

Ticketinformationen werden unter Erkennungen auf der Seite Erkennungen im ExtraHop-System angezeigt. Weitere Informationen finden Sie in der [Erkennungen](#) Thema.

Das folgende Python-Beispielskript verwendet Ticketinformationen aus einem Python-Array und aktualisiert die zugehörigen Erkennungen auf dem ExtraHop-System.

```
#!/usr/bin/python3
```

```

import json
import requests
import csv

API_KEY = '123456789abcdefghijklmnop'
HOST = 'https://extrahop.example.com/'

# Method that updates detections on an ExtraHop system
def updateDetection(detection):
    url = HOST + 'api/v1/detections/' + detection['detection_id']
    del detection['detection_id']
    data = json.dumps(detection)
    headers = {'Content-Type': 'application/json',
              'Accept': 'application/json',
              'Authorization': 'ExtraHop apikey=%s' % API_KEY}
    r = requests.patch(url, data=data, headers=headers)
    print(r.status_code)
    print(r.text)

# Array of detection information
detections = [
    {
        "detection_id": "1",
        "ticket_id": "TK-16982",
        "status": "new",
        "assignee": "sally",
        "resolution": None,
    },
    {
        "detection_id": "2",
        "ticket_id": "TK-2078",
        "status": None,
        "assignee": "jim",
        "resolution": None,
    },
    {
        "detection_id": "3",
        "ticket_id": "TK-3452",
        "status": None,
        "assignee": "alex",
        "resolution": None,
    }
]

for detection in detections:
    updateDetection(detection)

```



Hinweis Wenn das Skript eine Fehlermeldung zurückgibt, dass die Überprüfung des SSL-Zertifikats fehlgeschlagen ist, stellen Sie sicher, dass **Ihrem Sensor oder Ihrer Konsole wurde ein vertrauenswürdiges Zertifikat hinzugefügt**. Alternativ können Sie das hinzufügen `verify=False` Option zur Umgehung der Zertifikatsüberprüfung. Diese Methode ist jedoch nicht sicher und wird nicht empfohlen. Der folgende Code sendet eine HTTP GET-Anfrage ohne Zertifikatsüberprüfung:

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

Nachdem die Ticketverfolgung konfiguriert wurde, werden Ticketdetails im linken Bereich der Erkennungsdetails angezeigt, ähnlich der folgenden Abbildung:

The screenshot displays a ticket detail view for an anomaly titled "Suspicious CIFS Client File Share Access on AccountingLaptop". On the left, a sidebar shows the ticket's status as "CLOSED", its ID as "EX-4437", and the assignee as "hopuser". The main content area includes a risk score of 83, a "LATERAL MOVEMENT" indicator, and a description of the anomaly: "This device sent an excessive number of read requests over the Common Internet File System (CIFS) protocol. This anomaly indicates that the device might be compromised and is preparing files for data exfiltration." Below this, it lists the server linked to the anomaly: "corpshare.example.com (192.168.6.179)". At the bottom, a table provides CIFS metrics for "AccountingLaptop":

CIFS Metric	6-hour Snapshot	Peak Value	Expected Range	Deviation
Reads		1.13 K	0-1	112,500%

Status

Der Status des Tickets, das mit der Erkennung verknüpft ist. Die Ticketverfolgung unterstützt die folgenden Status:

- Neu
- In Bearbeitung
- geschlossen
- Mit ergriffenen Maßnahmen geschlossen
- Geschlossen, ohne dass Maßnahmen ergriffen wurden

Ticket-ID

Die ID des Tickets in Ihrem Work-Tracking-System, das mit der Erkennung verknüpft ist. Wenn Sie eine Vorlagen-URL konfiguriert haben, können Sie auf die Ticket-ID klicken, um das Ticket in Ihrem Work-Tracking-System zu öffnen.

Bevollmächtigter

Der Benutzername, der dem Ticket zugewiesen wurde, das mit der Erkennung verknüpft ist. Graue Benutzernamen weisen auf ein Konto hin, das kein ExtraHop-Konto ist.