

Erkennungsverfolgung aktivieren

Veröffentlicht: 2025-03-28

Mit der Erkennungsverfolgung können Sie einem Benutzer eine Erkennung zuweisen, den Status festlegen und Notizen hinzufügen. Sie können Erkennungen direkt im ExtraHop-System, mit einem externen Ticketsystem eines Drittanbieters oder mit beiden Methoden verfolgen.



Hinweis Sie müssen die Ticketverfolgung auf allen angeschlossenen Sensoren aktivieren.

Bevor Sie beginnen

- Sie müssen Zugriff auf ein ExtraHop-System mit einem Benutzerkonto haben, das **Administratorrechte** .
 - Nachdem Sie die externe Ticketverfolgung aktiviert haben, müssen Sie **Ticket-Tracking von Drittanbietern konfigurieren** indem Sie einen Auslöser schreiben, um Tickets in Ihrem Ticketsystem zu erstellen und zu aktualisieren, und dann Ticketaktualisierungen auf Ihrem ExtraHop-System über die REST-API aktivieren.
 - Wenn Sie das externe Ticket-Tracking deaktivieren, werden zuvor gespeicherte Status- und Empfänger-Ticketinformationen in das ExtraHop-Erkennungs-Tracking umgewandelt. Wenn das Erkennungs-Tracking innerhalb des ExtraHop-Systems aktiviert ist, können Sie Tickets einsehen, die bereits existierten, als Sie das externe Ticket-Tracking deaktiviert haben, aber Änderungen an diesem externen Ticket werden nicht im ExtraHop-System angezeigt.
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. Klicken Sie auf der Übersichtsseite auf **Systemeinstellungen**  und klicken Sie dann **Die gesamte Verwaltung**.
 4. Aus dem Einstellungen der Konsole Abschnitt, klicken **Erkennungsverfolgung**.
 5. Wählen Sie eine oder beide der folgenden Methoden für die Nachverfolgung von Erkennungen aus:
 - Wählen **Ermöglichen Sie ExtraHop-Benutzern, Erkennungen aus dem ExtraHop-System heraus zu verfolgen**.
 - Wählen **Ermöglichen Sie externe Integrationen wie SOAR oder Ticket-Tracking-Systeme, um Erkennungen über die ExtraHop Rest API zu verfolgen**.
 6. Optional: Nachdem Sie die Option zum Aktivieren externer Integrationen ausgewählt haben, geben Sie 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/$ticket_id`. Das `$ Ticket_ID` Die Variable wird durch die Ticket-ID ersetzt, die der Erkennung zugeordnet ist.

Nachdem die URL-Vorlage konfiguriert ist, können Sie in einer Erkennung auf die Ticket-ID klicken, um das Ticket in einem neuen Browser-Tab zu öffnen.

Today 14:00
lasting an hour

83
RISK

LATERAL MOVEMENT

Status — **CLOSED**
Ticket ID — **EX-4437**
Assignee — **hopuser**

Suspicious CIFS Client File Share Access on AccountingLaptop

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)

AccountingLaptop Activity Map

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

Nächste Schritte

Wenn Sie externe Ticket-Tracking-Integrationen aktiviert haben, müssen Sie mit der folgenden Aufgabe fortfahren:

- [Ticket-Tracking von Drittanbietern für Erkennungen konfigurieren](#)

Ticket-Tracking von Drittanbietern für Erkennungen konfigurieren

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

Bevor Sie beginnen

- Das musst du haben [hat in den Verwaltungseinstellungen die Option zum Nachverfolgen der Erkennung durch Dritte ausgewählt](#).
- Sie müssen Zugriff auf ein ExtraHop-System mit einem Benutzerkonto haben, das [System- und Zugriffsadministrationsrechte](#).
- Sie müssen mit dem Schreiben von ExtraHop-Triggern vertraut sein. siehe [Trigger](#) 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-Skripten vertraut sein und über einen gültigen API-Schlüssel verfügen, um die folgenden Verfahren ausführen zu können. siehe [Generieren Sie einen API-Schlüssel](#).

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` hinzu:

```
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);
```

Ticketinformationen über die REST-API an Erkennungen senden

Nachdem Sie einen Auslöser konfiguriert haben, um Tickets für Erkennungen in Ihrem Ticket-Tracking-System zu erstellen, können Sie die Ticketinformationen in Ihrem ExtraHop-System über die REST-API aktualisieren.

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

Das folgende Python-Beispielskript entnimmt 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 = '123456789abcdefghijklmnopqrstuvwxyz'
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)

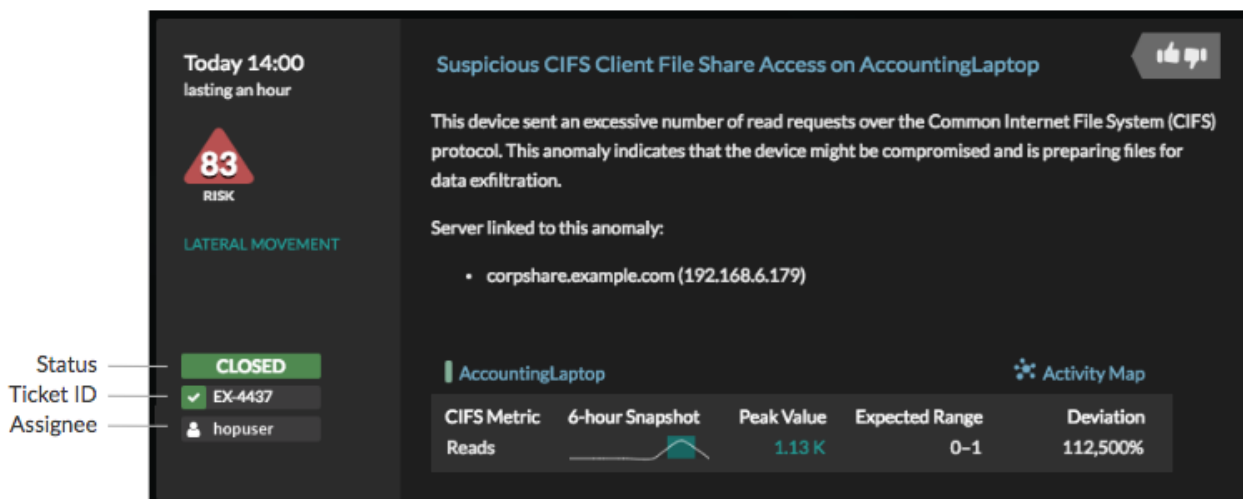
```



Hinweis: Wenn das Skript eine Fehlermeldung zurückgibt, dass die TLS-Zertifikatsüberprüfung 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:



Status

Der Status des Tickets, das mit der Erkennung verknüpft ist. Das Ticket-Tracking unterstützt die folgenden Status:

- Neu
- Im Gange
- 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.

Abtretungsempfänger

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.