ExtraHop 9.8
RevealX 360 REST-API-Leitfaden

Weitere Informationen finden Sie unter https://docs.extrahop.com.

Veröffentlicht: 2024-10-26

ExtraHop Networks
Seattle, WA 98101
877-333-9872 (US)
+44 (0)203 7016850 (EMEA)
+65-31585513 (APAC)
www.extrahop.com

Inhaltsübersicht

RevealX 360 REST-API-Leitfaden

Mit der RevealX 360-REST-API können Sie Konfigurationsaufgaben automatisieren und Metriken, Pakete und Erkennungen von RevealX 360 abrufen. Sie können Anfragen über eine REST-Schnittstelle (Representational State Transfer) an die API senden, auf die über Ressourcen-URIs und HTTP-Standardmethoden zugegriffen wird.

Bevor Sie eine REST-API-Anfrage an RevealX 360 senden können, müssen Sie das System für den REST-API-Zugriff aktivieren und Anmeldedaten generieren. Anschließend müssen Sie ein temporäres Zugriffstoken abrufen, indem Sie die ID und das Geheimnis Ihrer REST-API-Anmeldeinformationen an RevealX 360 senden. Fügen Sie abschließend das Zugriffstoken in den Header Ihrer Authentifizierungsanfrage ein. REST-API-Anmeldeinformationen laufen nicht automatisch ab und müssen manuell gelöscht werden, bevor sie ungültig werden.

Hinweis: Dieses Handbuch richtet sich an ein Publikum, das über grundlegende Kenntnisse in der Softwareentwicklung und dem ExtraHop-System verfügt.

Aktiviere die REST-API für RevealX 360

Bevor Sie REST-API-Anfragen an RevealX 360 senden können, müssen Sie den REST-API-Zugriff aktivieren.

Bevor Sie beginnen

- Sie müssen über System- und Zugriffsadministrationsrechte verfügen.

1. Loggen Sie sich in RevealX 360 ein.
2. Klicken Sie auf das Symbol Systemeinstellungen oben rechts auf der Seite und klicken Sie dann auf **Die gesamte Verwaltung**.
3. Klicken Sie **API-Zugriff**.
4. In der API-Zugriff verwalten Abschnitt, klicken **Aktiviere**.

   Wenn Sie die REST-API deaktivieren und dann erneut aktivieren, ist die REST-API aufgrund der DNS-Propagierung möglicherweise für etwa 15 Minuten nicht verfügbar, auch wenn im Abschnitt Status angegeben ist, dass der Zugriff aktiviert ist. Wir empfehlen, die REST-API nicht häufig zu deaktivieren und erneut zu aktivieren.

REST-API-Anmeldeinformationen erstellen

RevealX 360 authentifiziert REST-API-Anfragen mit dem OpenID Connect (OIDC) Protokoll. OIDC verlangt von Benutzern, temporäre Zugriffstoken bereitzustellen, wenn sie eine Anfrage an die API stellen. Bevor Sie Zugriffstoken generieren können, müssen Sie REST-API-Anmeldeinformationen erstellen, die auch als Client-Anmeldeinformationen bezeichnet werden.

Hinweis: REST-API-Anmeldeinformationen laufen nicht automatisch ab. Die von einem Benutzer erstellten Anmeldedaten werden nicht gelöscht, wenn der Benutzer aus dem System entfernt wird. Die Anmeldedaten bleiben gültig, bis sie gelöscht werden. Jeder Administrator kann alle Anmeldedaten löschen, unabhängig davon, welcher Benutzer die Anmeldedaten erstellt hat.

Die RevealX 360 REST-API unterstützt Cross-Origin Resource Sharing (CORS) nicht.

Bevor Sie beginnen

- Sie müssen über System- und Zugriffsadministrationsrechte verfügen.

1. Loggen Sie sich in RevealX 360 ein.
2. Klicken Sie auf das Symbol Systemeinstellungen oben rechts auf der Seite und klicken Sie dann auf **Die gesamte Verwaltung**.
3. Klicken Sie **API-Zugriff**.
4. Klicken Sie **Anmeldeinformationen erstellen**.
5. In der **Name** Feld, geben Sie einen Namen für die Anmeldedaten ein.
6. In der **Privilegien** Feld, geben Sie eine Berechtigungsstufe für die Anmeldedaten.

   Die Berechtigungsstufe bestimmt, welche Aktionen mit den Anmeldeinformationen ausgeführt werden können. Gewähren Sie REST-API-Anmeldeinformationen nicht mehr Rechte als nötig , da dies ein Sicherheitsrisiko darstellen kann. Beispielsweise sollten Anwendungen, die nur Metriken abrufen, keine Anmeldedaten erhalten, die Administratorrechte gewähren. Weitere Hinweise zu den einzelnen Berechtigungsstufen finden Sie unter Benutzerrechte.

   Hinweis: Die System- und Zugriffsadministrationsrechte ähneln den vollen Schreibrechten und ermöglichen es den Anmeldedaten, selbstverwaltete Sensoren und Trace-Appliances mit RevealX 360 zu verbinden.
7. In der **Paketzugriff** Feld, geben Sie an, ob Sie Pakete und Sitzungsschlüssel mit den Anmeldedaten abrufen können.
8. Klicken Sie **Speichern**.
   Das REST-API-Anmeldeinformationen kopieren Fenster wird angezeigt.
9. Unter ID, klicken **In die Zwischenablage kopieren** und speichern Sie die ID auf Ihrem lokalen Computer.
10. Unter Geheim, klicken **In die Zwischenablage kopieren** und speichern Sie das Geheimnis auf Ihrem lokalen Computer.

    Wichtig: Das Geheimnis kann später nicht eingesehen oder abgerufen werden.
11. Klicken Sie **Erledigt**.

Generieren Sie ein REST-API-Token

Ein temporäres API-Zugriffstoken muss in allen REST-API-Anfragen an RevealX 360 enthalten sein. Nachdem Sie REST-API-Anmeldeinformationen erstellt haben, können Sie Skripts schreiben, die temporäre API-Zugriffstoken mit den Anmeldedaten generieren. Die Skripte können dann REST-API-Anfragen an RevealX 360 mit den Tokens authentifizieren. Tokens sind nach der Generierung 10 Minuten lang gültig.

Die HTTPS-Token-Anfrage muss die folgenden Anforderungen erfüllen:

- Das Token wird in einer POST-Anforderung an den API-Token-Endpunkt gesendet, der auf der API-Zugriff Seite unter API-Endpunkt in RevealX 360.
- Fügen Sie die folgenden Header ein:
  - `Authorization: Basic <auth>`

    Wo `<auth>` ist eine Base64-kodierte Zeichenfolge aus ID und Geheimnis, die durch einen Doppelpunkt verbunden sind.
  - `Content-Type: application/x-www-form-urlencoded`
- Fügen Sie die folgende Nutzlast hinzu:

```
grant_type=client_credentials
```

Hinweis: Die temporären API-Zugriffstoken, die von den Beispielskripten erstellt wurden, sind nur 10 Minuten gültig. Wenn die Ausführung eines Skripts länger als 10 Minuten dauert, muss das Skript alle 10 Minuten ein neues Token generieren, um sicherzustellen, dass es kein abgelaufenes Token sendet. Wenn ein Skript ein abgelaufenes Token sendet, reagiert das System mit einem 401-HTTP-Fehlercode und der folgenden Fehlermeldung:

```
The incoming token has expired
```

Nächste Schritte
Nachdem Sie ein Token generiert haben, können Sie es als Bearer-Token in den HTTP-Autorisierungsheader aufnehmen, um Anfragen zu authentifizieren. Zum Beispiel, wenn Ihr Token "`abcdefghijklmnop0123456789`„, füge die folgende Zeichenfolge in den Header ein:

```
"Authorization": "Bearer abcdefghijklmnop0123456789"
```

Rufen Sie das Python-Beispielskript ab und führen Sie es aus

Das ExtraHop GitHub-Repository enthält ein Python-Skript, das ein temporäres API-Zugriffstoken generiert und dann zwei einfache Anfragen mit dem Token authentifiziert, die Geräte und Gerätegruppen von RevealX 360 abrufen.

1. Gehe zum ExtraHop Codebeispiele GitHub-Repository und laden Sie das herunter `py_rx360_auth/py_rx360_auth.py` Datei auf Ihrem lokalen Computer.
2. Öffnen Sie in einem Texteditor den `py_rx360_auth.py` archivieren und ersetzen Sie die folgenden Konfigurationsvariablen durch Informationen aus Ihrer Umgebung:

   • **GASTGEBER:** Der Hostname der RevealX 360-API. Dieser Hostname wird im RevealX 360 angezeigt API-Zugriff Seite unter API-Endpunkt. Der Hostname enthält nicht `/oauth2/token`.

   • **ID:** Die ID der REST-API-Anmeldeinformationen.

   • **GEHEIM:** Das Geheimnis der REST-API-Anmeldeinformationen.

Führen Sie den folgenden Befehl aus:

```
python3 py_rx360_auth.py
```

Bash- und cURL-Beispiel

Das ExtraHop GitHub-Repository enthält ein Beispiel-Bash-Skript, das mit dem cURL-Befehl ein REST-API-Token generiert und dann zwei einfache Anfragen mit dem Token authentifiziert , die Geräte und Gerätegruppen von der RevealX 360-REST-API abrufen.

Bevor Sie beginnen

- Das cURL-Tool muss auf Ihrem Computer installiert sein.
- Der JQ-Parser muss auf Ihrem Computer installiert sein. Weitere Informationen finden Sie unter https://stedolan.github.io/jq/.

1. Gehe zum ExtraHop Codebeispiele GitHub-Repository und laden Sie das herunter `bash_rx360_auth/bash_rx360_auth.sh` Datei auf Ihrem lokalen Computer.
2. Öffnen Sie in einem Texteditor den `bash_rx360_auth.sh` archivieren und ersetzen Sie die folgenden Konfigurationsvariablen durch Informationen aus Ihrer Umgebung:

   • **GASTGEBER:** Der Hostname der RevealX 360-API. Dieser Hostname wird im RevealX 360 angezeigt API-Zugriff Seite unter API-Endpunkt. Der Hostname beinhaltet nicht `/oauth2/token`.

   • **ID:** Die ID der REST-API-Anmeldeinformationen.

   • **GEHEIM:** Das Geheimnis der REST-API-Anmeldeinformationen.
3. Führen Sie den folgenden Befehl aus:

```
./bash_auth.sh
```

Erfahren Sie mehr über den REST API Explorer

Der REST API Explorer ist ein webbasiertes Tool, mit dem Sie detaillierte Informationen zu den ExtraHop REST API-Ressourcen, Methoden, Parametern, Eigenschaften und Fehlercodes anzeigen können. Codebeispiele sind in Python, cURL und Ruby für jede Ressource verfügbar. Sie können Operationen auch direkt über das Tool ausführen.

Öffnen Sie den REST API Explorer

Sie können den REST API Explorer über die Administrationseinstellungen oder über die folgende URL öffnen:

```
https://<revealx-360-hostname-or-ip-address>/api/v1/explore/
```

1. Loggen Sie sich in RevealX 360 ein.
2. Klicken Sie auf das Symbol Systemeinstellungen oben rechts auf der Seite und klicken Sie dann auf **Die gesamte Verwaltung**.
3. Klicken Sie **API-Zugriff**.
4. Auf dem API-Zugriff Seite, klicken **Öffnen Sie den ExtraHop REST API Explorer**.
   Der REST API Explorer wird in Ihrem Browser geöffnet.

Betriebsinformationen anzeigen

Im REST API Explorer können Sie auf einen beliebigen Vorgang klicken, um die Konfigurationsinformationen für die Ressource anzuzeigen.

Die folgende Tabelle enthält Informationen zu den Abschnitten, die für Ressourcen im REST API Explorer verfügbar sind. Die Verfügbarkeit von Abschnitten variiert je nach HTTP-Methode. Nicht bei allen Methoden sind alle Abschnitte in der Tabelle aufgeführt.

Abschnitt	Beschreibung
Körperparameter	Stellt alle Felder für den Anforderungstext und unterstützte Werte für jedes Feld bereit.
Parameter	Stellt Informationen zu den verfügbaren Abfrageparametern bereit.
Antworten	Informiert über die möglichen HTTP Statuscodes für die Ressource. Wenn du klickst **Anfrage senden**, dieser Abschnitt enthält auch die Antwort des Server und die cURL-, Python- und Ruby-Syntax, die zum Senden der angegebenen Anfrage erforderlich ist. **Hinweis:** Klicken **Modell** um Beschreibungen der Felder anzuzeigen, die in einer Antwort zurückgegeben wurden.

Identifizieren Sie Objekte auf dem ExtraHop-System

Um API-Operationen für ein bestimmtes Objekt auszuführen, müssen Sie die Objekt-ID ermitteln. Sie können die Objekt-ID mithilfe der folgenden Methoden im REST API Explorer leicht finden.

- Die Objekt-ID wird in den Headern bereitgestellt, die von einer POST-Anfrage zurückgegeben werden. Wenn Sie beispielsweise eine POST-Anfrage senden, um eine Seite zu erstellen, zeigen die Antwortheader eine Standort-URL an.

  Die folgende Anfrage gab den Speicherort für das neu erstellte Tag als zurück `/api/v1/tags/1` und die ID für das Tag als `1`.

  ```
  {
   "date": "Tue, 09 Nov 2021 18:21:00 GMT ",
   "via": "1.1 localhost",
   "server": "Apache",
   "content-type": "text/plain; charset=utf-8",
   "location": "/api/v1/tags/1",
   "cache-control": "private, max-age=0",
   "connection": "Keep-Alive",
   "keep-alive": "timeout=90, max=100",
   "content-length": "0"
  }
  ```

- Die Objekt-ID wird für alle Objekte bereitgestellt, die von einer GET-Anfrage zurückgegeben werden. Wenn Sie beispielsweise eine GET-Anfrage auf allen Geräten ausführen, enthält der Antworttext Informationen für jedes Gerät, einschließlich der ID.

  Der folgende Antworttext zeigt einen Eintrag für ein einzelnes Gerät mit der ID 10212 an:

  ```
  {
   "mod_time": 1448474346504,
   "node_id": null,
   "id": 10212,
   "extrahop_id": "test0001",
   "description": null,
   "user_mod_time": 1448474253809,
   "discover_time": 1448474250000,
   "vlanid": 0,
   "parent_id": 9352,
   "macaddr": "00:05:G3:FF:FC:28",
   "vendor": "Cisco",
   "is_l3": true,
   "ipaddr4": "10.10.10.5",
   "ipaddr6": null,
   "device_class": "node",
   "default_name": "Cisco5",
   "custom_name": null,
   "cdp_name": "",
   "dhcp_name": "",
   "netbios_name": "",
   "dns_name": "",
   "custom_type": "",
   "analysis_level": 1
  },
  ```

RevealX 360-Ressourcen

Sie können Operationen an den folgenden Ressourcen über die RevealX 360 REST-API ausführen. Sie können auch detailliertere Informationen zu diesen Ressourcen anzeigen, z. B. verfügbare HTTP Methoden, Abfrageparameter und Objekteigenschaften.

Hinweis API-Endpunkte befinden sich unter `<host>/api/v1/<endpoint>`, wo `host` ist der Hostname der RevealX 360-API. Zum Beispiel, wenn der Hostname der API ist `https://example.com`, der Endpunkt für Aktivitätskarten wäre die folgende URL:

```
https://example.com/api/v1/activitymaps
```

Sie können den Hostnamen vom API-Token-Endpunkt ableiten, indem Sie Folgendes entfernen `/oauth2/token` aus der Endpunktzeichenfolge, die auf der RevealX 360 API Access-Seite unter erscheint API-Endpunkt.

Karte der Aktivitäten

Eine Aktivitätsdiagramm ist eine dynamische visuelle Darstellung der L4-L7-Protokollaktivität zwischen Geräten in Ihrem Netzwerk. Erstellen Sie in Echtzeit ein 2D- oder 3D-Layout von Geräteverbindungen, um mehr über den Verkehrsfluss und die Beziehungen zwischen Geräten zu erfahren.

Hier sind einige wichtige Überlegungen zu Activity Maps:

- In Standard Analysis und Erweiterte Analyse Analysis können Sie nur Aktivitätskarten für Geräte erstellen. Geräte im Entdeckungsmodus sind nicht in Activity Maps enthalten. Weitere Informationen finden Sie unter Analysestufen.
- Wenn Sie eine Aktivitätsdiagramm für ein Gerät, eine Aktivitätsgruppe oder eine Gerätegruppe ohne Protokollaktivität im ausgewählten Zeitintervall erstellen, wird die Map ohne Daten angezeigt. Ändern Sie das Zeitintervall oder Ihre Herkunftsauswahl und versuchen Sie es erneut.
- Sie können eine Aktivitätsdiagramm in einem erstellen Konsole um die Geräteverbindungen all Ihrer Sensoren zu sehen.

Weitere Informationen zum Konfigurieren und Navigieren in Activity Maps finden Sie unter Karten der Aktivitäten.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
GET /activitymaps	Ruft alle Aktivitätskarten ab.
POST /activitymaps	Erstellen Sie eine neue Aktivitätsdiagramm.
POST /activitymaps/query	Führen Sie eine Netzwerktopologieabfrage durch, die Aktivitätsdiagramm Map-Daten als Flatfile-Inhalt zurückgibt.
/activitymaps/ {id} LÖSCHEN	Löscht eine bestimmte Aktivitätsdiagramm.
GET /activitymaps/ {id}	Rufen Sie eine bestimmte Aktivitätsdiagramm ab.
PATCH /activitymaps/ {id}	Aktualisieren Sie eine bestimmte Aktivitätsdiagramm.
POST /activitymaps/ {id} /query	Führen Sie eine Topologieabfrage für eine bestimmte Aktivitätsdiagramm durch, die Aktivitätsdiagramm Map-Daten als Flatfile-Inhalt zurückgibt.

Betrieb	Beschreibung
GET /activitymaps/ {id} /sharing	Rufen Sie die Benutzer und ihre Freigabeberechtigungen für eine bestimmte Aktivitätsdiagramm ab.
PATCH /activitymaps/ {id} /sharing	Aktualisieren Sie die Benutzer und ihre Freigabeberechtigungen für eine bestimmte Aktivitätsdiagramm.
PUT /activitymaps/ {id} /sharing	Ersetzen Sie die Benutzer und ihre Freigabeberechtigungen für eine bestimmte Aktivitätsdiagramm.

Einzelheiten der Operation

`POST /activitymaps`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Eigenschaften der Aktivitätsdiagramm.

`name:` ***Schnur***

Der freundliche Name für die Aktivitätsdiagramm.

`short_code:` ***Schnur***

(Optional) Der eindeutige Kurzcode, der global für alle Activity Maps gilt.

`description:` ***Schnur***

Die Beschreibung für die Aktivitätsdiagramm.

`weighting:` ***Schnur***

(Optional) Der Metrikwert, der bestimmt, wie Aktivitäten zwischen Geräten gewichtet werden. Unterstützte Elementwerte sind „Bytes“, „Verbindungen“ und „Turns“.

`mode:` ***Schnur***

(Optional) Das Layout der Aktivitätsdiagramm. Unterstützte Werte sind „2dforce“ und „3dforce“.

`show_alert_status:` ***Boolescher Wert***

(Optional) Gibt an, ob der Alarmstatus für Geräte auf der Aktivitätsdiagramm werden soll. Wenn diese Option aktiviert ist, steht die Farbe jedes Geräts auf der Karte für die schwerwiegendste Warnstufe, die dem Gerät zugeordnet ist.

`walks:` ***Reihe von Objekten***

Die Liste von einem oder mehreren Wanderobjekten. Ein Spaziergang ist ein Verkehrsweg, der aus einer oder mehreren Stufen besteht. Jeder Walk beginnt mit einem oder mehreren Ursprungsgeräten und erweitert sich auf Verbindungen zu Peer-Geräten, die auf Protokollaktivitäten basieren. Jede Erweiterung vom Ursprung aus ist ein Schritt. Der Inhalt des Objekts wird im Abschnitt „Gehen“ unten definiert.

`origins:` ***Reihe von Objekten***

Die Liste eines oder mehrerer Ursprungsgeräte des ersten Schritts innerhalb des Spaziergangs. Der Objektinhalt wird im Abschnitt „source_object“ unten definiert.

`object_type:` ***Schnur***

Der Quelltyp der Metrik.

Die folgenden Werte sind gültig:

- `device`

- `device_group`

`object_id:` ***Zahl***

Der eindeutige Bezeichner für das Quellobjekt.

`steps:` ***Reihe von Objekten***

Die Liste von einem oder mehreren Schritten innerhalb des Spaziergangs. Jeder Schritt wird durch die Protokollaktivität zwischen Geräten des vorherigen Schritts und einer neuen Gruppe von Peer-Geräten definiert. Der Objektinhalt wird im Abschnitt „Schritt“ unten definiert.

`relationships:` ***Reihe von Objekten***

(Optional) Die Liste mit einem oder mehreren Filtern, die die Beziehung zwischen zwei Geräten definieren. Die Filter geben an, nach welchen Rollen und Protokollen gesucht werden soll, wenn Peer-Geräte in diesem Schritt gefunden werden. Beziehungen werden in der Aktivitätsdiagramm als Rand dargestellt. Objektinhalte werden im Abschnitt „Beziehung“ weiter unten definiert. Wenn kein Wert angegeben ist, sucht der Vorgang nach allen Peers.

`protocol:` ***Schnur***

(Optional) Das mit der Beziehung verknüpfte Metrikprotokoll, z. B. „HTTP“ oder „DNS“. Der Vorgang sucht nur nach Verbindungen zwischen Geräten über das angegebene Protokoll.

`role:` ***Schnur***

(Optional) Die Geräterolle, die dem Metrik Protokoll der Beziehung zugeordnet ist. Der Vorgang sucht nur nach Verbindungen zwischen Geräten über das zugehörige Protokoll in der angegebenen Rolle. Unterstützte Rollenwerte sind „Client“, „Server“ oder „Any“. Auf „any“ setzen, um alle Client-, Server- und Peer-Gerätebeziehungen zu finden, die dem angegebenen Protokoll zugeordnet sind.

`peer_in:` ***Reihe von Objekten***

(Optional) Die Liste von einem oder mehreren Peer-Geräteobjekten, die in die Activity Map aufgenommen werden sollen. Nur Beziehungen zu Peers des angegebenen Quellobjekts sind enthalten. Der Objektinhalt wird im Abschnitt „source_object“ unten definiert.

`object_type:` ***Schnur***

Der Quelltyp der Metrik.

Die folgenden Werte sind gültig:

- `device`
- `device_group`

`object_id:` ***Zahl***

Der eindeutige Bezeichner für das Quellobjekt.

`peer_not_in:` ***Reihe von Objekten***

(Optional) Die Liste von einem oder mehreren Peer-Geräteobjekten, die von der Aktivitätsdiagramm ausgeschlossen werden sollen. Beziehungen zu Peers des angegebenen Quellobjekts sind ausgeschlossen. Der Objektinhalt wird im Abschnitt „source_object“ unten definiert.

`object_type:` ***Schnur***

Der Quelltyp der Metrik.

Die folgenden Werte sind gültig:

- `device`
- `device_group`

`object_id:` ***Zahl***

Der eindeutige Bezeichner für das Quellobjekt.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "description": "string",
    "mode": "string",
    "name": "string",
    "short_code": "string",
    "show_alert_status": true,
    "walks": {
        "origins": {
            "object_type": "string",
            "object_id": 0
        },
        "steps": {
            "relationships": {
                "protocol": "string",
                "role": "string"
            },
            "peer_in": {
                "object_type": "string",
                "object_id": 0
            },
            "peer_not_in": {
                "object_type": "string",
                "object_id": 0
            }
        }
    },
    "weighting": "string"
}
```

`POST /activitymaps/query`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Eigenschaften der Topologieabfrage.

`from:` ***Zahl***

Der Anfangszeitstempel des Zeitbereichs, den die Abfrage durchsucht, ausgedrückt in Millisekunden seit der Epoche.

`until:` ***Zahl***

(Optional) Der letzte Zeitstempel des Zeitbereichs, den die Abfrage durchsucht, ausgedrückt in Millisekunden seit der Epoche. Wenn kein Wert gesetzt ist, wird das Abfrageende standardmäßig auf „now“ gesetzt.

`weighting:` ***Schnur***

(Optional) Der Metrikwert, der bestimmt, wie Aktivitäten zwischen Geräten gewichtet werden.

Die folgenden Werte sind gültig:

- `bytes`
- `connections`
- `turns`

`edge_annotations:` ***Reihe von Zeichenketten***

(Optional) Die Liste mit einer oder mehreren Kantenanmerkungen, die in die Topologieabfrage aufgenommen werden sollen.

Die folgenden Werte sind gültig:

- `protocols`
- `appearances`

`walks:` ***Reihe von Objekten***

Die Liste von einem oder mehreren Walk-Objekten, die in die Topologieabfrage aufgenommen werden sollen. Ein Spaziergang ist ein Verkehrsweg, der aus einer oder mehreren Stufen besteht. Jeder Walk beginnt mit einem oder mehreren Ursprungsgeräten und erweitert sich auf Verbindungen zu Peer-Geräten, die auf Protokollaktivitäten basieren. Jede Erweiterung vom Ursprung aus ist ein Schritt. Der Objektinhalt wird im Abschnitt „topology_walk“ unten definiert.

`origins:` ***Reihe von Objekten***

Die Liste eines oder mehrerer Ursprungsgeräte des ersten Schritts innerhalb des Spaziergangs. Der Objektinhalt wird im Abschnitt „topology_source“ unten definiert.

`object_type:` ***Schnur***

Der Typ des Quellobjekts.

Die folgenden Werte sind gültig:

- `all_devices`
- `device_group`
- `device`

`object_id:` ***Zahl***

Der eindeutige Bezeichner für das Quellobjekt. Auf 0 setzen, wenn der Wert des Parameter „object_type“ „all_devices“ ist.

`steps:` ***Reihe von Objekten***

Die Liste von einem oder mehreren Schritten innerhalb des Spaziergangs. Jeder Schritt wird durch die Protokollaktivität zwischen Geräten des vorherigen Schritts und einer neuen Gruppe von Peer-Geräten definiert. Objektinhalte werden im Abschnitt „topology_step“ unten definiert.

`relationships:` ***Reihe von Objekten***

(Optional) Die Liste mit einem oder mehreren Filtern, die die Beziehung zwischen zwei Geräten definieren. Die Filter geben an, nach welchen Rollen und Protokollen gesucht werden soll, wenn Peer-Geräte in diesem Schritt gefunden werden. Beziehungen werden in der Aktivitätsdiagramm als Rand dargestellt. Wenn kein Wert festgelegt ist, umfasst die Operation alle Peers. Der Objektinhalt wird im Abschnitt „topology_relationship“ weiter unten definiert.

`role:` ***Schnur***

(Optional) Die Rolle des Peer-Geräts im Verhältnis zum Ursprungsgerät.

Die folgenden Werte sind gültig:

- `client`
- `server`
- `any`

`protocol:` ***Schnur***

(Optional) Das Protokoll, über das das Ursprungsgerät kommuniziert, z. B. „HTTP“. Wenn kein Wert festgelegt ist, enthält das Objekt ein beliebiges Protokoll.

`peer_in:` ***Reihe von Objekten***

(Optional) Die Liste von einem oder mehreren Peer-Geräten, die in das Topologiediagramm aufgenommen werden sollen. Nur Beziehungen zu Peers des angegebenen Quellobjekts sind enthalten. Der Objektinhalt wird im Abschnitt „topology_source“ unten definiert.

`object_type:` ***Schnur***

Der Typ des Quellobjekts.

Die folgenden Werte sind gültig:

- `all_devices`
- `device_group`
- `device`

`object_id:` ***Zahl***

Der eindeutige Bezeichner für das Quellobjekt. Auf 0 setzen, wenn der Wert des Parameter „object_type“ „all_devices“ ist.

`peer_not_in:` ***Reihe von Objekten***

(Optional) Die Liste von einem oder mehreren Peer-Geräten, die aus dem Topologiediagramm ausgeschlossen werden sollen. Beziehungen zu Peer-Geräten des angegebenen Quellobjekts sind ausgeschlossen. Der Objektinhalt wird im Abschnitt „topology_source“ unten definiert.

`object_type:` ***Schnur***

Der Typ des Quellobjekts.

Die folgenden Werte sind gültig:

- `all_devices`
- `device_group`
- `device`

`object_id:` ***Zahl***

Der eindeutige Bezeichner für das Quellobjekt. Auf 0 setzen, wenn der Wert des Parameter „object_type“ „all_devices“ ist.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "edge_annotations": [],
    "from": 0,
    "until": 0,
    "walks": {
        "origins": {
            "object_type": "string",
            "object_id": 0
        },
        "steps": {
            "relationships": {
                "role": "string",
                "protocol": "string"
            },
            "peer_in": {
                "object_type": "string",
                "object_id": 0
            },
            "peer_not_in": {
                "object_type": "string",
                "object_id": 0
            }
        }
```

```
    },
    "weighting": "string"
}
```

`GET /activitymaps`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "description": "string",
    "id": 0,
    "mod_time": 0,
    "mode": "string",
    "name": "string",
    "owner": "string",
    "rights": [
        "string"
    ],
    "short_code": "string",
    "show_alert_status": true,
    "walks": [],
    "weighting": "string"
}
```

`GET /activitymaps/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Aktivitätsdiagramm.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "description": "string",
    "id": 0,
    "mod_time": 0,
    "mode": "string",
    "name": "string",
    "owner": "string",
    "rights": [
        "string"
    ],
    "short_code": "string",
    "show_alert_status": true,
    "walks": [],
    "weighting": "string"
}
```

`POST /activitymaps/{id}/query`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Aktivitätsdiagramm.

`body:` ***Objekt***

Die Eigenschaften der Topologieabfrage.

`from:` ***Zahl***

Der Anfangszeitstempel des Zeitbereichs, den die Abfrage durchsucht, ausgedrückt in Millisekunden seit der Epoche.

`until:` ***Zahl***

(Optional) Der letzte Zeitstempel des Zeitbereichs, den die Abfrage durchsucht, ausgedrückt in Millisekunden seit der Epoche. Wenn kein Wert gesetzt ist, wird das Abfrageende standardmäßig auf „now“ gesetzt.

`edge_annotations:` ***Reihe von Zeichenketten***

(Optional) Die Liste mit einer oder mehreren Kantenanmerkungen, die in die Topologieabfrage aufgenommen werden sollen.

Die folgenden Werte sind gültig:

- `protocols`
- `appearances`

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "edge_annotations": [],
    "from": 0,
    "until": 0
}
```

`DELETE /activitymaps/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Aktivitätsdiagramm.

`PATCH /activitymaps/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Aktivitätsdiagramm.

`body:` ***Objekt***

Die Eigenschaften der Aktivitätsdiagramm, die aktualisiert werden sollen.

`GET /activitymaps/{id}/sharing`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Aktivitätsdiagramm.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "anyone": "string",
    "groups": {},
    "users": {}
}
```

`PUT /activitymaps/{id}/sharing`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Die Benutzer und ihre Berechtigungsstufen.

`id:` ***Zahl***
Die eindeutige Kennung für die Aktivitätsdiagramm.

`PATCH /activitymaps/{id}/sharing`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Die Benutzer und ihre Berechtigungsstufen.

`id:` ***Zahl***
Die eindeutige Kennung für die Aktivitätsdiagramm.

Warnung

Alerts sind Systembenachrichtigungen, die nach bestimmten Warnungskriterien generiert werden. Standardwarnungen sind im System verfügbar, oder Sie können eine benutzerdefinierte Alarm erstellen.

Erkennungen und Schwellenwertwarnungen können so eingestellt werden, dass Sie Alarm werden, wenn eine Metrik den in der Warnungskonfiguration definierten Wert überschreitet. Trendwarnungen können nicht über die REST-API konfiguriert werden. Weitere Informationen finden Sie unter Warnmeldungen.

Hinweis: Erkennungen durch maschinelles Lernen erfordern eine Verbindung zu ExtraHop Cloud Services.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
GET /alerts	Rufen Sie alle Benachrichtigungen ab.
POST/Benachrichtigungen	Erstellen Sie eine neue Alarm mit bestimmten Werten.
/alerts/ {id} LÖSCHEN	Löschen Sie eine bestimmte Alarm.
GET /alerts/ {id}	Rufen Sie eine bestimmte Alarm ab.
PATCH /alerts/ {id}	Wenden Sie Aktualisierungen auf eine bestimmte Alarm an.
GET /alerts/ {id} /applications	Rufen Sie alle Anwendungen ab, denen eine bestimmte Alarm zugewiesen wurde.
POST /alerts/ {id} /Anwendungen	Weisen Sie Anwendungen eine bestimmte Alarm zu und heben Sie deren Zuweisung auf.
LÖSCHEN Sie /alerts/ {id} /applications/ {child-id}	Heben Sie die Zuweisung einer Anwendung zu einer bestimmten Alarm auf.
POST /alerts/ {id} /Anwendungen/ {Child-ID}	Weisen Sie eine Anwendung einer bestimmten Alarm zu.
GET /alerts/ {id} /devicegroups	Alles abrufen Gerätegruppen denen eine bestimmte Alarm zugewiesen wurde.

Betrieb	Beschreibung
POST /alerts/ {id} /devicegroups	Weisen Sie Gerätegruppen eine bestimmte Alarm zu und heben Sie deren Zuweisung auf.
LÖSCHEN Sie /alerts/ {id} /devicegroups/ {child-id}	Heben Sie die Zuweisung einer Gerätegruppe zu einer bestimmten Alarm auf.
POST /alerts/ {id} /devicegroups/ {child-id}	Weisen Sie einer bestimmten Alarm eine Gerätegruppe zu.
GET /alerts/ {id} /devices	Ruft alle Geräte ab, denen eine bestimmte Alarm zugewiesen wurde.
POST /alerts/ {id} /Geräte	Weisen Sie Geräten eine bestimmte Alarm zu und heben Sie deren Zuweisung auf.
LÖSCHEN /alerts/ {id} /devices/ {child-id}	Heben Sie die Zuweisung eines Gerät zu einer bestimmten Alarm auf.
POST /alerts/ {id} /devices/ {child-id}	Weisen Sie einem bestimmten Alarm ein Gerät zu.
GET /alerts/ {id} /emailgroups	Ruft alle E-Mail-Gruppen ab, denen eine bestimmte Alarm zugewiesen wurde.
POST /alerts/ {id} /emailgroups	Weisen Sie E-Mail-Gruppen eine bestimmte Alarm zu und heben Sie deren Zuweisung auf.
LÖSCHEN Sie /alerts/ {id} /emailgroups/ {child-id}	Heben Sie die Zuweisung einer E-Mail-Gruppe zu einer bestimmten Alarm auf.
POST /alerts/ {id} /emailgroups/ {child-id}	Weisen Sie einer bestimmten Alarm eine E-Mail-Gruppe zu.
GET /alerts/ {id} /exclusionintervals	Ruft alle Ausschlussintervalle ab, denen eine bestimmte Alarm zugewiesen wurde.
POST /alerts/ {id} /exclusionintervals	Weisen Sie Ausschlussintervallen eine bestimmte Alarm zu und heben Sie deren Zuweisung auf.
LÖSCHEN /alerts/ {id} /exclusionintervals/ {child-id}	Heben Sie die Zuweisung eines Ausschlussintervalls zu einer bestimmten Alarm auf.
POST /alerts/ {id} /exclusionintervals/ {child-id}	Weisen Sie einer bestimmten Alarm ein Ausschlussintervall zu.
GET /alerts/ {id} /networks	Ruft alle Netzwerke ab, denen eine bestimmte Alarm zugewiesen wurde.
POST /alerts/ {id} /Netzwerke	Weisen Sie Netzwerken eine bestimmte Alarm zu und heben Sie deren Zuweisung auf.
LÖSCHEN /alerts/ {id} /networks/ {child-id}	Heben Sie die Zuweisung eines Netzwerk zu einer bestimmten Alarm auf.
POST /alerts/ {id} /networks/ {child-id}	Weisen Sie einer bestimmten Alarm ein Netzwerk zu.
GET /alerts/ {id} /stats	Rufen Sie alle zusätzlichen Statistiken für eine bestimmte Alarm ab.

Einzelheiten der Operation

`GET /alerts`

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "apply_all": true,
    "author": "string",
    "categories": [
        "string"
    ],
    "cc": [],
    "description": "string",
    "disabled": true,
    "field_name": "string",
    "field_name2": "string",
    "field_op": "string",
    "id": 0,
    "interval_length": 0,
    "mod_time": 0,
    "name": "string",
    "notify_snmp": true,
    "object_type": "string",
    "operand": "string",
    "operator": "string",
    "param": {},
    "param2": {},
    "protocols": [
        "string"
    ],
    "refire_interval": 0,
    "severity": 0,
    "stat_name": "string",
    "type": "string",
    "units": "string"
}
```

`POST /alerts`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Wendet die angegebenen Eigenschaftswerte auf die neue Alarm an.

`description:` ***Schnur***

Eine optionale Beschreibung für die Alarm.

`notify_snmp:` ***Boolesch***

(Optional) Gibt an, ob ein SNMP-Trap gesendet werden soll, wenn eine Alarm generiert wird.

`field_op:` ***Schnur***

Die Art des Vergleichs zwischen den Feldern field_name und field_name2 beim Anwenden eines Verhältnisses. Gilt nur für Schwellenwert-Alarme.

Die folgenden Werte sind gültig:

- `/`
- `null`

`stat_name:` ***Schnur***

Der Statistikname für die Alarm. Gilt nur für Schwellenwert-Alarme.

`disabled:` ***Boolesch***

(Optional) Gibt an, ob die Alarm deaktiviert ist.

`operator:` ***Schnur***

Der logische Operator, der angewendet wird, wenn der Wert des Operandenfeldes mit den Warnbedingungen verglichen wird. Gilt nur für Schwellenwert-Alarme.

Die folgenden Werte sind gültig:

- `==`
- `>`
- `<`
- `>=`
- `<=`

`operand:` ***Schnur***

Der Wert, der mit den Alarmbedingungen verglichen werden soll. Die Vergleichsmethode wird durch den Wert des Operatorfeldes spezifiziert. Gilt nur für Schwellenwert-Alarme.

`field_name:` ***Schnur***

Der Name der überwachten Metrik. Gilt nur für Schwellenwert-Alarme.

`name:` ***Schnur***

Der eindeutige, freundliche Name für die Alarm.

`cc:` ***Reihe von Zeichenketten***

Die Liste der E-Mail-Adressen, die nicht in einer E-Mail-Gruppe enthalten sind und die Benachrichtigungen erhalten sollen.

`apply_all:` ***Boolesch***

Gibt an, ob die Alarm allen verfügbaren Datenquellen zugewiesen ist.

`severity:` ***Zahl***

(Optional) Der Schweregrad der Alarm, der im Warnungsverlauf, in E-Mail-Benachrichtigungen und SNMP-Traps angezeigt wird. Die Schweregrade 0-2 erfordern sofortige Aufmerksamkeit. Die Schweregrade sind beschrieben in der REST-API-Leitfaden.

Die folgenden Werte sind gültig:

- `0`
- `1`
- `2`
- `3`
- `4`
- `5`
- `6`
- `7`

`author:` ***Schnur***

Der Name des Benutzers, der die Alarm erstellt hat.

`param:` ***Objekt***

Der erste Warnparameter, der entweder ein Schlüsselmuster oder ein Datenpunkt ist. Gilt nur für Schwellenwert-Alarme.

`interval_length:` ***Zahl***

Die Länge des Warnintervalls, ausgedrückt in Sekunden. Gilt nur für Schwellenwert-Alarme.

Die folgenden Werte sind gültig:

- `30`
- `60`

- `120`
- `300`
- `600`
- `900`
- `1200`
- `1800`

`param2:` ***Objekt***

Der zweite Warnparameter, der entweder ein Schlüsselmuster oder ein Datenpunkt ist. Gilt nur für Schwellenwert-Alarme.

`units:` ***Schnur***

Das Intervall, in dem der Alarmzustand bewertet werden soll. Gilt nur für Schwellenwert-Alarme.

Die folgenden Werte sind gültig:

- `none`
- `period`
- `1 sec`
- `1 min`
- `1 hr`

`field_name2:` ***Schnur***

Die zweite überwachte Metrik bei der Anwendung eines Verhältnisses. Gilt nur für Schwellenwert-Alarme.

`refire_interval:` ***Zahl***

(Optional) Das Zeitintervall, in dem die Alarmbedingungen überwacht werden, ausgedrückt in Sekunden.

Die folgenden Werte sind gültig:

- `300`
- `600`
- `900`
- `1800`
- `3600`
- `7200`
- `14400`

`type:` ***Schnur***

Die Art der Alarm.

Die folgenden Werte sind gültig:

- `threshold`

`object_type:` ***Schnur***

Der Typ der Metrikquelle, die von der Alert-Konfiguration überwacht wird. Gilt nur für Erkennungswarnungen.

Die folgenden Werte sind gültig:

- `application`
- `device`

`protocols:` ***Reihe von Zeichenketten***

(Optional) Die Liste der überwachten Protokolle. Gilt nur für Erkennungswarnungen.

`categories:` ***Reihe von Zeichenketten***

(Optional) Die Liste einer oder mehrerer Erkennungskategorien. Eine Alarm wird nur generiert, wenn eine Erkennung in den angegebenen Kategorien identifiziert wird. Gilt nur für Erkennungswarnungen.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "apply_all": true,
    "author": "string",
    "categories": [
        "string"
    ],
    "cc": [],
    "description": "string",
    "disabled": true,
    "field_name": "string",
    "field_name2": "string",
    "field_op": "string",
    "interval_length": 0,
    "name": "string",
    "notify_snmp": true,
    "object_type": "string",
    "operand": "string",
    "operator": "string",
    "param": {},
    "param2": {},
    "protocols": [
        "string"
    ],
    "refire_interval": 0,
    "severity": 0,
    "stat_name": "string",
    "type": "string",
    "units": "string"
}
```

`GET /alerts/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "apply_all": true,
    "author": "string",
    "categories": [
        "string"
    ],
    "cc": [],
    "description": "string",
    "disabled": true,
    "field_name": "string",
    "field_name2": "string",
    "field_op": "string",
    "id": 0,
    "interval_length": 0,
    "mod_time": 0,
    "name": "string",
```

```
    "notify_snmp": true,
    "object_type": "string",
    "operand": "string",
    "operator": "string",
    "param": {},
    "param2": {},
    "protocols": [
        "string"
    ],
    "refire_interval": 0,
    "severity": 0,
    "stat_name": "string",
    "type": "string",
    "units": "string"
}
```

`DELETE /alerts/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`PATCH /alerts/{id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Wenden Sie die angegebenen Eigenschaftswertaktualisierungen auf die Alarm an.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`GET /alerts/{id}/stats`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "alert_id": 0,
    "field_name": "string",
    "id": 0,
    "param": "string",
    "stat_name": "string"
}
```

`GET /alerts/{id}/devicegroups`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/devicegroups`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Liste der eindeutigen Identifikatoren für Gerätegruppen, die der Alarm zugewiesen sind oder nicht.

`assign:` ***Reihe von Zahlen***

IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***

IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/devicegroups/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für die Gerätegruppe.

`id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

`DELETE /alerts/{id}/devicegroups/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für die Gerätegruppe.

`id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

`GET /alerts/{id}/emailgroups`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/emailgroups`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Liste der eindeutigen Identifikatoren für E-Mail-Gruppen, die der Warnung zugewiesen sind oder nicht zugewiesen wurden.

`assign:` ***Reihe von Zahlen***

IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***

IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/emailgroups/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für die E-Mail-Gruppe.

`id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

`DELETE /alerts/{id}/emailgroups/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für die E-Mail-Gruppe.

`id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

`GET /alerts/{id}/exclusionintervals`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/exclusionintervals`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Liste der eindeutigen Identifikatoren für Ausschlussintervalle, die der Alarm zugewiesen ist oder nicht.

`assign:` ***Reihe von Zahlen***

IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***

IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/exclusionintervals/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für das Ausschlussintervall.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`DELETE /alerts/{id}/exclusionintervals/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für das Ausschlussintervall.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`GET /alerts/{id}/devices`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/devices`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Die Liste der eindeutigen Identifikatoren für Geräte, die der Alarm zugewiesen sind oder nicht.

`assign:` ***Reihe von Zahlen***
IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***
IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/devices/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für das Gerät.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`DELETE /alerts/{id}/devices/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für das Gerät.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`GET /alerts/{id}/networks`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/networks`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Die Liste der eindeutigen Identifikatoren für Netzwerke, die der Alarm zugewiesen sind oder nicht.

`assign:` ***Reihe von Zahlen***
IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***
IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/networks/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für das Netzwerk.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`DELETE /alerts/{id}/networks/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für das Netzwerk.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`GET /alerts/{id}/applications`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/applications`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Die Liste der eindeutigen Identifikatoren für Anwendungen, die der Alarm zugewiesen sind oder nicht.

`assign:` ***Reihe von Zahlen***
IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***
IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`POST /alerts/{id}/applications/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für die Anwendung.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`DELETE /alerts/{id}/applications/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für die Anwendung.

`id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

Priorität der Analyse

Das ExtraHop-System analysiert und klassifiziert den Traffic für jedes Gerät, das es entdeckt. Ihre Lizenz reserviert Kapazität für das ExtraHop-System, um Metriken für hoher Wert Geräte zu sammeln. Diese Kapazität ist mit zwei Analyseebenen verknüpft: Fortgeschrittene Analyse und Standardanalyse.

Sie können angeben, welche Geräte die Stufen Erweiterte Analyse und Standard Analysis erhalten, indem Sie Folgendes konfigurieren Regeln für die Analysepriorität. Analyseprioritäten helfen dabei, das ExtraHop-System darüber zu informieren, welche Geräte in Ihrer Umgebung wichtig sind. Eine dritte

Analyseebene, der Entdeckungsmodus, ist für Geräte verfügbar, die sich nicht in Advanced oder Standard Analysis befinden.

Hinweis Standardmäßig verwaltet jeder Sensor seine eigenen Analyseprioritäten. Wenn der Sensor an eine Konsole angeschlossen ist, können Sie diese zentral verwalten gemeinsame Systemeinstellungen von der Konsole aus.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Bedienung	Beschreibung
HOLEN SIE SICH /analysispriority/config/ {sensor_id}	Rufen Sie die Analyseprioritätsregeln für eine bestimmte Sensor.
PUT /analysispriority/config/ {sensor_id}	Ersetzen Sie die Analyseprioritätsregeln für eine bestimmte Sensor.
GET /analysispriority/ {sensor_id} /manager	Rufen Sie das System ab, das für die Verwaltung der Analyseprioritätsregeln für das konfiguriert ist Sensor.
PATCH /analysispriority/ {sensor_id} /manager	Aktualisieren Sie das System, das die Analyseprioritätsregeln für eine bestimmte Gruppe verwaltet Sensor.

Einzelheiten der Operation

`GET /analysispriority/{appliance_id}/manager`

Geben Sie die folgenden Parameter an.

`appliance_id`: ***Zahl***
Die Kennung für den lokalen Sensor. Dieser Wert muss auf 0 gesetzt werden.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "manager": {}
}
```

`GET /analysispriority/config/{appliance_id}`

Geben Sie die folgenden Parameter an.

`appliance_id`: ***Zahl***
Die Kennung für einen Sensor. Setzen Sie diesen Wert auf 0, wenn Sie einen Sensor aufrufen.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "advanced_rules": [],
    "autofill_advanced": true,
    "autofill_standard": true,
    "is_in_effect": true,
    "standard_rules": []
}
```

`PUT /analysispriority/config/{appliance_id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Eigenschaften der Regeln für die Prioritätsanalyse.

`autofill_advanced:` ***Boolescher Wert***

Gibt an, ob Geräte automatisch in Erweiterte Analyse platziert werden sollen, bis die Kapazität erreicht ist. Geräte in der Liste advanced_rules werden priorisiert, gefolgt von Geräten in der Liste standard_rules und dann nach der Erkennungszeit für das Gerät. Die Kapazität für Erweiterte Analyse wird durch die ExtraHop-Systemlizenz bestimmt.

`advanced_rules:` ***Reihe von Objekten***

(Optional) Die Erweiterte Analyse Analysis-Prioritätsregeln für eine Gerätegruppe.

`type:` ***Schnur***

Der Gruppentyp, für den die Prioritätsregeln für die Analyse gelten.

Die folgenden Werte sind gültig:

- `device_group`

`object_id:` ***Zahl***

Die eindeutige Kennung für die Gruppe.

`description:` ***Schnur***

(Optional) Die Beschreibung der Prioritätsregeln für Analysen.

`autofill_standard:` ***Boolescher Wert***

Gibt an, ob Geräte automatisch in die Standardanalyse aufgenommen werden sollen, bis die Gesamtkapazität erreicht ist. Geräte in der Liste standard_rules werden priorisiert, gefolgt von der Erkennungszeit für das Gerät. Die Gesamtkapazität wird durch die ExtraHop-Systemlizenz bestimmt.

`standard_rules:` ***Reihe von Objekten***

(Optional) Die Standardanalyse-Prioritätsregeln für eine Gerätegruppe.

`type:` ***Schnur***

Der Gruppentyp, für den die Prioritätsregeln für die Analyse gelten.

Die folgenden Werte sind gültig:

- `device_group`

`object_id:` ***Zahl***

Die eindeutige Kennung für die Gruppe.

`description:` ***Schnur***

(Optional) Die Beschreibung der Prioritätsregeln für Analysen.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "advanced_rules": {
        "type": "string",
        "object_id": 0,
        "description": "string"
    },
    "autofill_advanced": true,
    "autofill_standard": true,
    "standard_rules": {
        "type": "string",
        "object_id": 0,
        "description": "string"
    }
}
```

`appliance_id:` ***Zahl***

Die Kennung für einen Sensor. Setzen Sie diesen Wert auf 0, wenn Sie einen Sensor aufrufen.

`PATCH /analysispriority/{appliance_id}/manager`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die ID des Sensor oder der Konsole, die die Prioritätsregeln für die Analyse des lokalen Sensor verwaltet. Setzen Sie diesen Wert auf 0, um die Verwaltung des lokalen Sensor wiederherzustellen.

`manager:` ***Zahl***

Die eindeutige Kennung für den verwaltenden Sensor oder die verwaltende Konsole.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "manager": 0
}
```

`appliance_id:` ***Zahl***

Die Kennung für den lokalen Sensor. Dieser Wert muss auf 0 gesetzt werden.

Gerät

Das ExtraHop-System besteht aus einem Netzwerk verbundener Geräte, die Aufgaben wie die Überwachung des Verkehrs, die Analyse von Daten, das Speichern von Daten und das Identifizieren von Erkennungen ausführen.

Sie können Informationen über ExtraHop-Appliances abrufen, die mit der lokalen Appliance verbunden sind, und neue Verbindungen zu Remote-ExtraHop-Appliances herstellen.

Hinweis: Sie können nur eine Verbindung zu einer Remote-ExtraHop-Appliance herstellen, die für dieselbe Edition wie die lokale ExtraHop-Appliance lizenziert ist.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Bedienung	Beschreibung
GET /appliances	Ruft alle Remote-ExtraHop-Appliances ab, die mit der lokalen Appliance verbunden sind.
GET /Appliances/ {id}	Ruft eine bestimmte Remote-ExtraHop-Appliance ab, die mit der lokalen Appliance verbunden ist.
GET /Appliances/firmware/next	Rufen Sie Firmware-Versionen ab, auf die Remote-ExtraHop-Systeme aktualisiert werden können.
POST /Appliances/Firmware/Upgrade	Aktualisieren Sie die Firmware auf externen ExtraHop-Systemen, die mit dem lokalen System verbunden sind. Firmware-Images werden von ExtraHop Cloud Services heruntergeladen.
GET /Appliances/Sensortags	Ruft alle Sensor-Tags ab.
POST /Appliances/Sensortags	Erstellen Sie ein Sensor-Tag.
POST /Appliances/Sensortags/delete	Löschen Sie mehrere Sensor-Tags.
LÖSCHE /appliances/sensortags/ {id}	Löscht ein bestimmtes Sensor-Tag.
GET /Appliances/Sensortags/ {id}	Ruft ein bestimmtes Sensor-Tag ab.

Bedienung	Beschreibung
PATCH /Appliances/Sensortags/ {id}	Aktualisieren Sie ein bestimmtes Sensor-Tag.
GET /Appliances/ {id} /sensortags	Ruft alle Tags ab, die einem bestimmten Sensor zugewiesen sind.
PATCH /Appliances/ {id} /sensortags	Aktualisieren Sie die einem Sensor zugewiesenen Tags.

Einzelheiten der Operation

`GET /appliances`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "add_time": 0,
    "advanced_analysis_capacity": 0,
    "analysis_levels_managed": true,
    "connection_type": "string",
    "data_access": true,
    "display_name": "string",
    "fingerprint": "string",
    "firmware_version": "string",
    "hostname": "string",
    "id": 0,
    "license_platform": "string",
    "license_status": "string",
    "licensed_features": {},
    "licensed_modules": [
        "string"
    ],
    "managed_by_local": true,
    "manages_local": true,
    "nickname": "string",
    "platform": "string",
    "status_message": "string",
    "sync_time": 0,
    "total_capacity": 0,
    "uuid": "string"
}
```

`GET /appliances/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Geben Sie die eindeutige Kennung für die Appliance an. Geben Sie 0 an, um die lokale Appliance auszuwählen.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "add_time": 0,
    "advanced_analysis_capacity": 0,
    "analysis_levels_managed": true,
    "connection_type": "string",
    "data_access": true,
```

```
    "display_name": "string",
    "fingerprint": "string",
    "firmware_version": "string",
    "hostname": "string",
    "id": 0,
    "license_platform": "string",
    "license_status": "string",
    "licensed_features": {},
    "licensed_modules": [
        "string"
    ],
    "managed_by_local": true,
    "manages_local": true,
    "nickname": "string",
    "platform": "string",
    "status_message": "string",
    "sync_time": 0,
    "total_capacity": 0,
    "uuid": "string"
}
```

`GET /appliances/{ids_id}/association`

Geben Sie die folgenden Parameter an.

`ids_id:` ***Zahl***
Geben Sie die ID des IDS-Sensors an.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "associated_sensor_id": 0
}
```

`POST /appliances/{ids_id}/association`

Geben Sie die folgenden Parameter an.

`ids_id:` ***Zahl***
Geben Sie die ID des IDS-Sensors an.

`body:` ***Objekt***
Geben Sie die ID des Paketsensor an.

`associated_sensor_id:` ***Zahl***
Die ID des Paketsensor.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "associated_sensor_id": 0
}
```

`GET /appliances/firmware/next`

Geben Sie die folgenden Parameter an.

`ids:` ***Schnur***
(Optional) Eine CSV-Liste mit eindeutigen Identifikatoren für die Remote-Appliances. Wenn dieser Parameter angegeben ist, gibt der Vorgang Firmware-Versionen zurück, auf die jedes der

angegebenen Remote-Geräte aktualisiert werden kann. Wenn dieser Parameter nicht angegeben ist, gibt der Vorgang Firmware-Versionen zurück, auf die jedes Remote-Gerät aktualisiert werden kann.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "release": "string",
    "versions": []
}
```

`POST /appliances/firmware/upgrade`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Firmware-Upgrade-Optionen.

`version:` ***Schnur***

Die Firmware-Version, auf die Appliances aktualisiert werden sollen. Sie können eine Liste der gültigen Versionen mit der Operation GET /api/v1/appliances/firmware/next abrufen.

`system_ids:` ***Reihe von Zahlen***

Eine Liste eindeutiger Identifikatoren für die Remote-Appliances. Sie können Appliance-IDs mit der Operation GET /api/v1/appliances abrufen; Appliance-IDs werden in den ID-Feldern der Antwort zurückgegeben.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "system_ids": [],
    "version": "string"
}
```

Bewerbung

Anwendungen sind benutzerdefinierte Gruppen, die Metriken sammeln, die durch Trigger für verschiedene Arten von Traffic identifiziert wurden. Die Standardanwendung All Activity enthält alle gesammelten Metriken.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit der Anwendungsressource ausführen können:

Betrieb	Beschreibung
GET /anwendungen	Rufen Sie alle Anwendungen ab, die innerhalb eines bestimmten Zeitraums aktiv waren.
POST /Anwendungen	Erstellen Sie eine neue Anwendung.
GET /Anwendungen/ {id}	Rufen Sie eine bestimmte Anwendung ab.
PATCH /Anwendungen/ {id}	Aktualisieren Sie eine bestimmte Anwendung.
GET /Anwendungen/ {id} /Aktivität	Ruft alle Aktivitäten für eine bestimmte Anwendung ab.
GET /applications/ {id} /alerts	Alles abrufen Warnungen die einer bestimmten Anwendung zugewiesen sind.

Betrieb	Beschreibung
POST /Anwendungen/ {id} /Benachrichtigungen	Weisen Sie einer bestimmten Anwendung Warnmeldungen zu und heben Sie deren Zuweisung auf.
LÖSCHEN Sie /applications/ {id} /alerts/ {child-id}	Heben Sie die Zuweisung einer Alarm zu einer bestimmten Anwendung auf.
POST /Anwendungen/ {id} /alerts/ {Child-ID}	Weisen Sie einer bestimmten Anwendung eine Alarm zu.
GET /Anwendungen/ {id} /dashboards	Rufen Sie alle Dashboards ab, die sich auf eine bestimmte Anwendung beziehen.

Einzelheiten der Operation

`GET /applications/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Anwendung.

`include_criteria:` ***Boolescher Wert***

(Optional) Gibt an, ob die mit der Anwendung verknüpften Kriterien in die Antwort aufgenommen werden sollen.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "criteria": [],
    "description": "string",
    "discovery_id": "string",
    "display_name": "string",
    "extrahop_id": "string",
    "id": 0,
    "mod_time": 0,
    "node_id": 0,
    "user_mod_time": 0
}
```

`POST /applications`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Eigenschaften der Anwendung.

`node_id:` ***Zahl***

(Optional) Die eindeutige Kennung für den Sensor, dem diese Anwendung zugeordnet ist. Der Bezeichner kann über den Vorgang GET /appliances abgerufen werden. Dieses Feld ist nur auf einer Konsole gültig.

`discovery_id:` ***Schnur***

Die eindeutige Kennung für die Anwendung, die auf der Anwendungsseite im ExtraHop-System angezeigt wird.

`display_name:` ***Schnur***

Der benutzerfreundliche Name für die Anwendung.

`description:` ***Schnur***

(Optional) Eine optionale Beschreibung der Anwendung.

`criteria:` ***Reihe von Objekten***

(Optional) Eine Reihe von Protokoll- und Quellkriterien, die mit der Anwendung verknüpft sind. Der Inhalt dieses Arrays wird im Abschnitt „Kriterien“ unten definiert.

`protocol_default:` ***Schnur***

Die von der Anwendung überwachten Standardprotokolle. Unterstützte Werte sind „any“ und „none“.

`sources:` ***Reihe von Objekten***

Ein Array, das eine oder mehrere der Anwendung zugeordnete Metrik Quellen enthält. Die Anwendung sammelt nur Metriken aus den angegebenen Quellen. Der Inhalt dieses Arrays ist im Abschnitt „Quelle“ unten definiert.

`type:` ***Schnur***

Der Typ der Metrikquelle, die der Anwendung zugeordnet ist. Unterstützte Quelltypwerte sind 'Gerät' und 'device_group'.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät oder die Gerätegruppe, die der Anwendung zugeordnet ist.

`protocols:` ***Objekt***

(Optional) Die Liste mit einer oder mehreren Protokoll- und Rollenzuordnungen, die der Anwendung zugeordnet sind. Die Anwendung sammelt nur Metriken aus den angegebenen Protokollen. Das Format jedes Protokoll ist {'Protokoll': 'role'}. Beispiel: {'http': 'server'}. Unterstützte Rollenwerte sind „Client“, „Server“, „any“ oder „none“.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "criteria": {
        "protocol_default": "string",
        "sources": {
            "type": "string",
            "id": 0
        },
        "protocols": {}
    },
    "description": "string",
    "discovery_id": "string",
    "display_name": "string",
    "node_id": 0
}
```

`PATCH /applications/{id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Wendet die angegebenen Eigenschaftenupdates auf die Anwendung an.

`id:` ***Zahl***

Die eindeutige Kennung für die Anwendung.

`GET /applications`

Geben Sie die folgenden Parameter an.

`active_from:` ***Zahl***

(Optional) Gibt nur Anwendungen zurück, die nach der angegebenen Zeit aktiv sind. Positive Werte geben die Zeit in Millisekunden seit der Epoche an. Negative Werte geben die Zeit in Millisekunden vor der aktuellen Uhrzeit an.

`active_until:` ***Zahl***

(Optional) Gibt nur Anwendungen zurück, die vor dem angegebenen Zeitpunkt aktiv waren. Positive Werte geben die Zeit in Millisekunden seit der Epoche an. Negative Werte geben die Zeit in Millisekunden vor der aktuellen Uhrzeit an.

`limit:` ***Zahl***

(Optional) Beschränken Sie die Anzahl der Anwendungen, die zurückgegeben werden, auf die angegebene Höchstzahl.

`offset:` ***Zahl***

(Optional) Überspringen Sie die ersten n Anwendungsergebnisse. Dieser Parameter wird häufig mit dem Grenzparameter kombiniert.

`search_type:` ***Schnur***

Der Objekttyp, nach dem gesucht werden soll.

Die folgenden Werte sind gültig:

- `any`
- `name`
- `node`
- `discovery_id`
- `extrahop-id`

`value:` ***Schnur***

(Optional) Die Suchkriterien. Fügen Sie vor und nach den Kriterien einen Schrägstrich hinzu, um den RegEx-Abgleich anzuwenden.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "criteria": [],
    "description": "string",
    "discovery_id": "string",
    "display_name": "string",
    "extrahop_id": "string",
    "id": 0,
    "mod_time": 0,
    "node_id": 0,
    "user_mod_time": 0
}
```

`GET /applications/{id}/activity`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Anwendung.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "application_id": 0,
    "from_time": 0,
    "id": 0,
    "mod_time": 0,
```

```
    "stat_name": "string",
    "until_time": 0
}
```

`GET /applications/{id}/alerts`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Rufen Sie den eindeutigen Bezeichner für die Anwendung ab.

`direct_assignments_only:` ***Boolescher Wert***
(Optional) Gibt an, ob die Ergebnisse auf Warnungen beschränkt sind, die der Anwendung direkt zugewiesen sind.

`POST /applications/{id}/alerts`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Weist die angegebene Liste eindeutiger Kennungen für Warnmeldungen zu oder hebt deren Zuweisung auf.

`assign:` ***Reihe von Zahlen***
IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***
IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***
Geben Sie eine eindeutige Kennung für die Anwendung ein.

`POST /applications/{id}/alerts/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`id:` ***Zahl***
Die eindeutige Kennung für die Anwendung.

`DELETE /applications/{id}/alerts/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`id:` ***Zahl***
Die eindeutige Kennung für die Anwendung.

`GET /applications/{id}/dashboards`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Anwendung.

Audit-Protokoll

Das Audit-Log zeigt eine Datensatz aller aufgezeichneten Systemadministrations- und Konfigurationsaktivitäten an, z. B. die Uhrzeit der Aktivität, den Benutzer, der die Aktivität ausgeführt hat, den Vorgang, die Betriebsdetails und die Systemkomponente.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
GET /auditlog	Ruft alle Audit-Log-Meldungen ab.

Einzelheiten der Operation

`GET /auditlog`

Geben Sie die folgenden Parameter an.

`limit:` ***Zahl***
(Optional) Die maximale Anzahl von Protokollnachrichten, die zurückgegeben werden sollen.

`offset:` ***Zahl***
(Optional) Die Anzahl der Protokollnachrichten, die in den Ergebnissen übersprungen werden sollen. Gibt Logmeldungen ab dem Offset-Wert zurück.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "body": {},
    "id": 0,
    "occur_time": 0,
    "time": 0
}
```

Bündel

Bundles sind Dokumente im JSON-Format, die Informationen zur ausgewählten Systemkonfiguration enthalten, z. B. Trigger, Dashboards, Anwendungen oder Warnungen.

Sie können ein Paket erstellen und diese Konfigurationen dann auf ein anderes ExtraHop-System übertragen oder das Paket als Backup speichern. Bundles können auch heruntergeladen werden von ExtraHop Lösungspakete und über die REST-API angewendet. Weitere Informationen finden Sie unter Bündel.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
HOLEN SIE SICH /bundles	Rufen Sie Metadaten zu allen Bundles auf dem ExtraHop-System ab.

Betrieb	Beschreibung
POST /Bundles	Laden Sie ein neues Paket in das ExtraHop-System hoch.
/bundles/ {id} LÖSCHEN	Löscht ein bestimmtes Paket.
ERHALTE /bundles/ {id}	Rufen Sie einen bestimmten Bundle-Export ab.
POST /bundles/ {id} /apply	Wenden Sie ein gespeichertes Paket auf das ExtraHop-System an.

Einzelheiten der Operation

`GET /bundles`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "built_in": true,
    "created_time": 0,
    "description": "string",
    "id": 0,
    "mod_time": 0,
    "name": "string"
}
```

`POST /bundles`

Geben Sie die folgenden Parameter an.

`body:` ***Schnur***

Ein JSON-formatierter Bundle-Export.

`name:` ***Schnur***

Der freundliche Name für das Paket.

`description:` ***Schnur***

(Optional) Eine optionale Beschreibung für das Paket.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "description": "string",
    "name": "string"
}
```

`GET /bundles/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Paket.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "built_in": true,
    "created_time": 0,
```

```
    "description": "string",
    "id": 0,
    "mod_time": 0,
    "name": "string"
}
```

`DELETE /bundles/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für das Paket.

`POST /bundles/{id}/apply`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für das Paket.

`body:` ***Objekt***
Die Konfigurationsoptionen für die Anwendung des Paket.

`policy:` ***Schnur***
Gibt an, ob widersprüchliche Objekte überschrieben oder übersprungen werden sollen.

Die folgenden Werte sind gültig:

- `overwrite`
- `skip`

`include_assignments:` ***Boolescher Wert***
Gibt an, ob Objektzuweisungen mit dem Paket wiederhergestellt werden sollen.

`node_ids:` ***Reihe von Zahlen***
Eine Liste mit eindeutigen Kennungen für die Sensoren, auf die das Paket angewendet werden soll. Dieses Feld ist nur auf einer Konsole gültig.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "include_assignments": true,
    "node_ids": [],
    "policy": "string"
}
```

Armaturenbretter

Dashboards sind integrierte oder benutzerdefinierte Ansichten Ihrer ExtraHop-Metrikinformationen. Weitere Informationen finden Sie unter Dashboards.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
HOLEN SIE SICH /dashboards	Rufen Sie alle Dashboards ab.
/dashboards/ {id} LÖSCHEN	Löschen Sie ein bestimmtes Dashboard.
GET /dashboards/ {id}	Rufen Sie ein bestimmtes Dashboard ab.

Betrieb	Beschreibung
PATCH /dashboards/ {id}	Aktualisieren Sie den Besitz eines bestimmten Dashboard.
GET /dashboards/ {id} /reports	Rufen Sie Dashboard-Berichte ab, die ein bestimmtes Dashboard enthalten. **Hinweis:** Dieser Vorgang ist nur von einer Konsole aus verfügbar.
GET /dashboards/ {id} /sharing	Rufen Sie die Benutzer und ihre Freigabeberechtigungen für ein bestimmtes Dashboard ab.
PATCH /dashboards/ {id} /sharing	Aktualisieren Sie die Benutzer und ihre Freigabeberechtigungen für ein bestimmtes Dashboard.
PUT /dashboards/ {id} /sharing	Ersetzen Sie die Benutzer und ihre Freigabeberechtigungen für ein bestimmtes Dashboard.

Einzelheiten der Operation

`GET /dashboards`

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "author": "string",
    "comment": "string",
    "id": 0,
    "mod_time": 0,
    "name": "string",
    "owner": "string",
    "rights": [
        "string"
    ],
    "short_code": "string",
    "type": "string"
}
```

`GET /dashboards/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Dashboard.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "author": "string",
    "comment": "string",
    "id": 0,
    "mod_time": 0,
    "name": "string",
    "owner": "string",
    "rights": [
        "string"
```

```
    ],
    "short_code": "string",
    "type": "string"
}
```

`DELETE /dashboards/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für das Dashboard.

`PATCH /dashboards/{id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Der Benutzername des Dashboard-Besitzers.
`id:` ***Zahl***
Die eindeutige Kennung für das Dashboard.

`GET /dashboards/{id}/sharing`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für das Dashboard.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "anyone": "string",
    "groups": {},
    "users": {}
}
```

`PUT /dashboards/{id}/sharing`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Die Benutzer und ihre Berechtigungsstufen.
`id:` ***Zahl***
Die eindeutige Kennung für das Dashboard.

`PATCH /dashboards/{id}/sharing`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Die Benutzer und ihre Berechtigungsstufen.
`id:` ***Zahl***
Die eindeutige Kennung für das Dashboard.

`GET /dashboards/{id}/reports`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Dashboard.

Erkennungen

Mit der Ressource Erkennungen können Sie Erkennungen abrufen, die vom ExtraHop-System identifiziert wurden.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Bedienung	Beschreibung
GET /Erkennungen	Ruft alle Funde ab.
GET /Erkennungen/Formate	Ruft alle Erkennungstypen ab.
GET /detections/formats/ {id}	Ruft einen bestimmten Erkennungstyp ab.
POST /Erkennungen/Formate	Erstellen Sie einen neuen benutzerdefinierten Erkennungstyp.
LÖSCHE /detections/formats/ {id}	Löscht einen bestimmten benutzerdefinierten Erkennungstyp.
PATCH /Erkennungen/Formate/ {id}	Aktualisieren Sie einen bestimmten benutzerdefinierten Erkennungstyp.
GET /Erkennungen/Regeln/Verbergen	Ruft alle Tuning-Regeln ab.
GET /detections/rules/hiding/ {id}	Ruft eine bestimmte Tuning-Regel ab.
POST /Erkennungen/Regeln/Verbergen	Erstellen Sie eine Optimierungsregel.
LÖSCHEN /detections/rules/hiding/ {id}	Löschen Sie eine Tuning-Regel.
PATCH /Erkennungen/Regeln/Ausblenden/ {id}	Aktualisieren Sie eine Tuning-Regel.
POST /Erkennungen/Suche	Ruft Erkennungen ab, die den angegebenen Suchkriterien entsprechen.
PATCH /Erkennungen/Tickets	Aktualisieren Sie ein Ticket, das mit Erkennungen verknüpft ist.
GET /Erkennungen/ {id}	Ruft eine bestimmte Erkennung ab.
GET /Erkennungen/ {id} /untersuchungen	Ruft alle Untersuchungen ab, in denen sich eine bestimmte Erkennung befindet
PATCH /Erkennungen/ {id}	Aktualisieren Sie eine Erkennung.
/detections/ {id} /notes LÖSCHEN	Löscht die Notizen für eine bestimmte Erkennung.
GET /detections/ {id} /notes	Ruft die Notizen für eine bestimmte Erkennung ab.
PUT /Erkennungen/ {id} /notes	Erstellen oder ersetzen Sie Notizen für eine bestimmte Erkennung.
GET /detections/ {id} /related	Ruft alle Funde ab, die sich auf eine bestimmte Erkennung beziehen.

Einzelheiten der Operation

`GET /detections/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Erkennung.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "appliance_id": 0,
    "assignee": "string",
    "categories": [
        "string"
    ],
    "create_time": 0,
    "description": "string",
    "end_time": 0,
    "id": 0,
    "is_user_created": true,
    "mitre_tactics": [],
    "mitre_techniques": [],
    "mod_time": 0,
    "participants": [],
    "properties": {},
    "recommended": true,
    "recommended_factors": [],
    "resolution": "string",
    "risk_score": 0,
    "start_time": 0,
    "status": "string",
    "ticket_id": "string",
    "ticket_url": "string",
    "title": "string",
    "type": "string",
    "update_time": 0,
    "url": "string"
}
```

`GET /detections`

Geben Sie die folgenden Parameter an.

`limit:` ***Zahl***

(Optional) Beschränken Sie die Anzahl der zurückgegebenen Erkennungen auf die angegebene Höchstzahl. Eine zufällige Auswahl von Erkennungen wird zurückgegeben.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "appliance_id": 0,
    "assignee": "string",
    "categories": [
        "string"
    ],
    "create_time": 0,
    "description": "string",
    "end_time": 0,
    "id": 0,
    "is_user_created": true,
```

```
    "mitre_tactics": [],
    "mitre_techniques": [],
    "mod_time": 0,
    "participants": [],
    "properties": {},
    "recommended": true,
    "recommended_factors": [],
    "resolution": "string",
    "risk_score": 0,
    "start_time": 0,
    "status": "string",
    "ticket_id": "string",
    "ticket_url": "string",
    "title": "string",
    "type": "string",
    "update_time": 0,
    "url": "string"
}
```

`POST /detections/search`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Suchparameter für die Erkennung.

`filter:` ***Objekt***

Erkennungsspezifische Filter.

`category:` ***Schnur***

Veraltet. Ersetzt durch das Kategorienfeld.

`categories:` ***Reihe von Zeichenketten***

Gibt Erkennungen aus den angegebenen Kategorien zurück.

`assignee:` ***Reihe von Zeichenketten***

Gibt Erkennungen zurück, die dem angegebenen Benutzer zugewiesen sind. Geben Sie „.none“ an, um nach nicht zugewiesenen Funden zu suchen, oder geben Sie „.me“ an, um nach Funden zu suchen, die dem authentifizierten Benutzer zugewiesen sind.

`ticket_id:` ***Reihe von Zeichenketten***

Gibt Erkennungen zurück, die mit den angegebenen Tickets verknüpft sind. Geben Sie „.none“ an, um nach Entdeckungen zu suchen, die nicht mit Tickets verknüpft sind.

`status:` ***Reihe von Zeichenketten***

Gibt Erkennungen mit dem angegebenen Status zurück. Um nach Erkennungen mit einem Nullstatus zu suchen, der im ExtraHop-System als Offen angezeigt wird, geben Sie „.none“ an. Sie können den Status einer Erkennung nur dann über die REST-API auf „neu“ ändern, wenn Ticket-Tracking durch Dritte ist aktiviert.

Die folgenden Werte sind gültig:

- `new`
- `in_progress`
- `closed`
- `acknowledged`

`resolution:` ***Reihe von Zeichenketten***

Gibt Erkennungen für Tickets mit der angegebenen Auflösung zurück. Geben Sie „.none“ an, um nach Erkennungen ohne Auflösung zu suchen.

Die folgenden Werte sind gültig:

- `action_taken`
- `no_action_taken`

`types:` ***Reihe von Zeichenketten***

Gibt Erkennungen mit den angegebenen Typen zurück.

`risk_score_min:` ***Zahl***

Gibt Erkennungen mit Risikoeinstufungen zurück, die größer oder gleich dem angegebenen Wert sind.

`recommended:` ***Boolesch***

Gibt Erkennungen zurück, die für die Triage empfohlen werden, auch bekannt als Smart Triage. Dieses Feld ist nur auf einer Konsole gültig.

`from:` ***Zahl***

Gibt Erkennungen zurück, die nach dem angegebenen Datum aufgetreten sind, ausgedrückt in Millisekunden seit der Epoche. Erkennungen, die vor dem angegebenen Datum begonnen haben, werden zurückgegeben, wenn die Erkennung zu diesem Zeitpunkt noch nicht abgeschlossen war.

`limit:` ***Zahl***

Gibt nicht mehr als die angegebene Anzahl von Erkennungen zurück.

`offset:` ***Zahl***

Die Anzahl der Erkennungen, die bei der Paginierung übersprungen werden sollen.

`sort:` ***Reihe von Objekten***

Sortiert die zurückgegebenen Erkennungen nach den angegebenen Feldern. Standardmäßig werden Erkennungen nach dem Zeitpunkt der letzten Aktualisierung und dann nach der ID in aufsteigender Reihenfolge sortiert.

`direction:` ***Schnur***

Die Reihenfolge, in der zurückgegebene Erkennungen sortiert werden.

Die folgenden Werte sind gültig:

- `asc`
- `desc`

`field:` ***Schnur***

Das Feld, nach dem Erkennungen sortiert werden sollen.

`until:` ***Zahl***

Gibt Erkennungen zurück, die vor dem angegebenen Datum endeten, ausgedrückt in Millisekunden seit der Epoche.

`update_time:` ***Zahl***

Gibt Erkennungen zurück, die sich auf Ereignisse beziehen, die nach dem angegebenen Datum eingetreten sind, ausgedrückt in Millisekunden seit der Epoche. Beachten Sie, dass der ExtraHop Machine Learning Service historische Daten analysiert, um Erkennungen zu generieren. Daher gibt es eine Zeitverzögerung zwischen dem Auftreten der Ereignisse, die diese Erkennungen verursachen, und dem Zeitpunkt, an dem die Erkennungen generiert werden. Wenn Sie mehrmals im gleichen update_time-Fenster nach Entdeckungen suchen, werden bei der späteren Suche möglicherweise Erkennungen zurückgegeben, die bei der vorherigen Suche nicht gefunden wurden.

`mod_time:` ***Zahl***

Gibt Erkennungen zurück, die nach dem angegebenen Datum aktualisiert wurden, ausgedrückt in Millisekunden seit der Epoche.

`create_time:` ***Zahl***

Gibt Erkennungen zurück, die nach dem angegebenen Datum erstellt wurden, ausgedrückt in Millisekunden seit der Epoche. Für Sensoren gibt dies Erkennungen zurück, die nach dem

angegebenen Datum generiert wurden. Bei Konsolen gibt dies Erkennungen zurück, die nach dem angegebenen Datum zum ersten Mal mit der Konsole synchronisiert wurden.

`id_only:` ***Boolesch***

(Optional) Gibt nur die IDs der Funde zurück.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "create_time": 0,
    "filter": {
        "category": "string",
        "categories": [],
        "assignee": [],
        "ticket_id": [],
        "status": [],
        "resolution": [],
        "types": [],
        "risk_score_min": 0,
        "recommended": true
    },
    "from": 0,
    "id_only": true,
    "limit": 0,
    "mod_time": 0,
    "offset": 0,
    "sort": {
        "direction": "string",
        "field": "string"
    },
    "until": 0,
    "update_time": 0
}
```

`PATCH /detections/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Erkennung.

`body:` ***Objekt***

Die zu aktualisierenden Erkennungsparameter.

`ticket_id:` ***Schnur***

Die ID des Tickets, das mit der Erkennung verknüpft ist.

`assignee:` ***Schnur***

Der Empfänger der Erkennung oder des Tickets, das mit der Erkennung verknüpft ist.

`status:` ***Schnur***

Der Status der Erkennung oder des Tickets, das mit der Erkennung verknüpft ist. Wenn der Wert Null ist, lautet der im ExtraHop-System angezeigte Status Offen. Der Wert „new“ kann nur über die REST-API angegeben werden, wenn Ticket-Tracking durch Dritte ist aktiviert.

Die folgenden Werte sind gültig:

- `new`
- `in_progress`
- `closed`
- `acknowledged`

`resolution:` ***Schnur***

Die Auflösung der Erkennung oder des mit der Erkennung verknüpften Tickets.

Die folgenden Werte sind gültig:

- `action_taken`
- `no_action_taken`

`participants:` ***Reihe von Objekten***

Eine Liste der Geräte und Anwendungen, die mit der Erkennung verknüpft sind. Sie können bestimmte Felder für einen Teilnehmer ändern, aber Sie können einer Erkennung keine neuen Teilnehmer hinzufügen.

`id:` ***Zahl***

Die ID des Teilnehmer, der mit der Erkennung verknüpft ist.

`usernames:` ***Reihe von Zeichenketten***

Die Benutzernamen, die dem Teilnehmer über die REST-API zugeordnet sind.

`origins:` ***Reihe von Zeichenketten***

Die Quell-IP-Adressen, die dem Teilnehmer über die REST-API zugeordnet sind.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assignee": "string",
    "participants": {
        "id": 0,
        "usernames": [],
        "origins": []
    },
    "resolution": "string",
    "status": "string",
    "ticket_id": "string"
}
```

`PATCH /detections/tickets`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die zu aktualisierenden Erkennungsticketwerte.

`ticket_id:` ***Schnur***

Die ID des Tickets, das mit der Erkennung verknüpft ist.

`assignee:` ***Schnur***

Der Empfänger des Tickets, das mit der Erkennung verknüpft ist.

`status:` ***Schnur***

Der Status des Tickets, das mit der Erkennung verknüpft ist.

Die folgenden Werte sind gültig:

- `new`
- `in_progress`
- `closed`
- `acknowledged`

`resolution:` ***Schnur***

Die Auflösung des Tickets, das mit der Erkennung verknüpft ist.

Die folgenden Werte sind gültig:

- `action_taken`
- `no_action_taken`

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assignee": "string",
    "resolution": "string",
    "status": "string",
    "ticket_id": "string"
}
```

`GET /detections/{id}/related`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die ID der Erkennung, für die verwandte Erkennungen abgerufen werden sollen.

`from:` ***Zahl***

Gibt Erkennungen zurück, die nach dem angegebenen Datum aufgetreten sind, ausgedrückt in Millisekunden seit der Epoche. Erkennungen, die vor dem angegebenen Datum begonnen haben, werden zurückgegeben, wenn die Erkennung zu diesem Zeitpunkt noch nicht abgeschlossen war.

`until:` ***Zahl***

Gibt Erkennungen zurück, die vor dem angegebenen Datum endeten, ausgedrückt in Millisekunden seit der Epoche.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "appliance_id": 0,
    "assignee": "string",
    "categories": [
        "string"
    ],
    "create_time": 0,
    "description": "string",
    "end_time": 0,
    "id": 0,
    "is_user_created": true,
    "mitre_tactics": [],
    "mitre_techniques": [],
    "mod_time": 0,
    "participants": [],
    "properties": {},
    "recommended": true,
    "recommended_factors": [],
    "resolution": "string",
    "risk_score": 0,
    "start_time": 0,
    "status": "string",
    "ticket_id": "string",
    "ticket_url": "string",
    "title": "string",
    "type": "string",
    "update_time": 0,
    "url": "string"
}
```

`GET /detections/{id}/investigations`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die ID der Erkennung, für die verwandte Untersuchungen abgerufen werden sollen.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "appliance_id": 0,
    "assignee": "string",
    "categories": [
        "string"
    ],
    "create_time": 0,
    "description": "string",
    "end_time": 0,
    "id": 0,
    "is_user_created": true,
    "mitre_tactics": [],
    "mitre_techniques": [],
    "mod_time": 0,
    "participants": [],
    "properties": {},
    "recommended": true,
    "recommended_factors": [],
    "resolution": "string",
    "risk_score": 0,
    "start_time": 0,
    "status": "string",
    "ticket_id": "string",
    "ticket_url": "string",
    "title": "string",
    "type": "string",
    "update_time": 0,
    "url": "string"
}
```

`GET /detections/formats`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "author": "string",
    "categories": [],
    "display_name": "string",
    "is_user_created": true,
    "last_updated": 0,
    "mitre_categories": [],
    "properties": {},
    "released": 0,
    "status": "string",
    "type": "string"
}
```

`GET /detections/formats/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***

Der Zeichenkettenbezeichner des Erkennungsformats.

`built_in_only:` ***Boolesch***

(Optional) Wenn dieses Feld den Wert true hat, werden nur integrierte Erkennungsformate zurückgegeben. Wenn dieses Feld falsch ist und sowohl ein benutzerdefiniertes Format als auch ein integriertes Format dieselbe ID haben, wird das benutzerdefinierte Format zurückgegeben. Der Standardwert ist falsch.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "author": "string",
    "categories": [],
    "display_name": "string",
    "is_user_created": true,
    "last_updated": 0,
    "mitre_categories": [],
    "properties": {},
    "released": 0,
    "status": "string",
    "type": "string"
}
```

`POST /detections/formats`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Parameter des Erkennungsformats.

`type:` ***Schnur***

Ein Zeichenkettenbezeichner für den Erkennungstyp. Die Zeichenfolge darf nur Buchstaben, Zahlen und Unterstriche enthalten. Obwohl Erkennungstypen in integrierten Formaten einzigartig sind und Erkennungstypen in benutzerdefinierten Formaten eindeutig sind, können ein integriertes und ein benutzerdefiniertes Format denselben Erkennungstyp gemeinsam haben.

`display_name:` ***Schnur***

Der Anzeigename des Erkennungstyps, der auf der Seite „Erkennungen“ im ExtraHop-System angezeigt wird.

`mitre_categories:` ***Reihe von Zeichenketten***

(Optional) Die IDs der MITRE-Techniken, die mit der Erkennung verknüpft sind.

`author:` ***Schnur***

(Optional) Der Autor des Erkennungsformats.

`categories:` ***Reihe von Zeichenketten***

(Optional) Die Liste der Kategorien, zu denen die Erkennung gehört. Geben Sie für POST- und PATCH-Operationen eine Liste mit einer einzigen Zeichenfolge an. Sie können nicht mehr als eine Kategorie für benutzerdefinierte Erkennungsformate angeben. Die Kategorie „Perf“ oder „Sek“ wird automatisch zu allen Erkennungsformaten hinzugefügt.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "author": "string",
    "categories": [],
    "display_name": "string",
    "mitre_categories": [],
```

```
    "type": "string"
}
```

`DELETE /detections/formats/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***
Der Zeichenkettenbezeichner des Erkennungsformats.

`PATCH /detections/formats/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***
Der Zeichenkettenbezeichner des Erkennungsformats.

`body:` ***Objekt***
Die Parameter des Erkennungsformats.

`GET /detections/rules/hiding`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "author": "string",
    "create_time": 0,
    "description": "string",
    "detection_type": "string",
    "detections_hidden": 0,
    "enabled": true,
    "expiration": 0,
    "hide_past_detections": true,
    "id": 0,
    "offender": {},
    "participants_hidden": 0,
    "properties": [],
    "victim": {}
}
```

`GET /detections/rules/hiding/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Der eindeutige Bezeichner für die Tuning-Regel.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "author": "string",
    "create_time": 0,
    "description": "string",
    "detection_type": "string",
    "detections_hidden": 0,
    "enabled": true,
    "expiration": 0,
    "hide_past_detections": true,
    "id": 0,
```

```
    "offender": {},
    "participants_hidden": 0,
    "properties": [],
    "victim": {}
}
```

`POST /detections/rules/hiding`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Parameter der Tuning-Regel.

`offender:` ***Objekt***

Der Täter, für den diese Tuning-Regel gilt. Geben Sie ein detection_hiding_participant-Objekt an, um die Regel auf ein bestimmtes Opfer anzuwenden, oder geben Sie „Any“ an, um die Regel auf einen beliebigen Täter anzuwenden.

`object_type:` ***Schnur***

Die Art des Teilnehmer.

Die folgenden Werte sind gültig:

- `device`
- `device_group`
- `ipaddr`
- `locality_type`
- `network_locality`
- `hostname`
- `scanner_service`

`object_id:` ***Zahl***

Die ID für das Gerät, die Gerätegruppe oder den Netzwerkstandort. Diese Option ist nur gültig, wenn der Objekttyp „Gerät“, „device_group“ oder „network_locality“ ist.

`object_value:` ***Array oder String***

Die IP-Adresse oder der CIDR-Block des Teilnehmer. Sie können eine einzelne Adresse oder einen Block in einer Zeichenfolge oder mehrere Adressen oder Blöcke in einem Array angeben. Diese Option ist nur gültig, wenn der Objekttyp „ipaddr“ ist.

`object_locality:` ***Schnur***

Der Netzwerklokalitätstyp des Teilnehmer. Geben Sie entweder „extern“ oder „intern“ an. Diese Option ist nur gültig, wenn der Objekttyp „locality_type“ ist.

Die folgenden Werte sind gültig:

- `internal`
- `external`

`object_scanner:` ***Array oder String***

Der Name eines externen Scandienstes. Sie können einen einzelnen Dienst in einer Zeichenfolge oder mehrere Werte in einem Array angeben. Sie können auch „Beliebig“ angeben, um einen beliebigen Scandienst auszuwählen. Diese Option ist nur gültig, wenn der Objekttyp „scanner_service“ ist.

`object_hostname:` ***Array oder String***

Der Hostname eines Teilnehmer. Sie können einen einzelnen Hostnamen in einer Zeichenfolge oder mehrere Hostnamen in einem Array angeben. Diese Option ist nur gültig, wenn der Objekttyp „hostname“ ist.

`victim:` ***Objekt***

Das Opfer, für das diese Tuning-Regel gilt. Geben Sie ein detection_hiding_participant-Objekt an, um die Regel auf ein bestimmtes Opfer anzuwenden, oder geben Sie „Any“ an, um die Regel auf ein beliebiges Opfer anzuwenden.

`object_type:` ***Schnur***

Die Art des Teilnehmer.

Die folgenden Werte sind gültig:

- `device`
- `device_group`
- `ipaddr`
- `locality_type`
- `network_locality`
- `hostname`
- `scanner_service`

`object_id:` ***Zahl***

Die ID für das Gerät, die Gerätegruppe oder den Netzwerkstandort. Diese Option ist nur gültig, wenn der Objekttyp „Gerät“, „device_group“ oder „network_locality“ ist.

`object_value:` ***Array oder String***

Die IP-Adresse oder der CIDR-Block des Teilnehmer. Sie können eine einzelne Adresse oder einen Block in einer Zeichenfolge oder mehrere Adressen oder Blöcke in einem Array angeben. Diese Option ist nur gültig, wenn der Objekttyp „ipaddr“ ist.

`object_locality:` ***Schnur***

Der Netzwerklokalitätstyp des Teilnehmer. Geben Sie entweder „extern“ oder „intern“ an. Diese Option ist nur gültig, wenn der Objekttyp „locality_type“ ist.

Die folgenden Werte sind gültig:

- `internal`
- `external`

`object_scanner:` ***Array oder String***

Der Name eines externen Scandienstes. Sie können einen einzelnen Dienst in einer Zeichenfolge oder mehrere Werte in einem Array angeben. Sie können auch „Beliebig“ angeben, um einen beliebigen Scandienst auszuwählen. Diese Option ist nur gültig, wenn der Objekttyp „scanner_service“ ist.

`object_hostname:` ***Array oder String***

Der Hostname eines Teilnehmer. Sie können einen einzelnen Hostnamen in einer Zeichenfolge oder mehrere Hostnamen in einem Array angeben. Diese Option ist nur gültig, wenn der Objekttyp „hostname“ ist.

`expiration:` ***Zahl***

Die Zeit, in der die Tuning-Regel abläuft, ausgedrückt in Millisekunden seit der Epoche. Ein Wert von Null oder 0 gibt an, dass die Regel nicht abläuft.

`description:` ***Schnur***

(Optional) Die Beschreibung der Tuning-Regel.

`detection_type:` ***Schnur***

Der Erkennungstyp, für den diese Optimierungsregel gilt. Zeigen Sie eine Liste der gültigen Felder für „type“ an, indem Sie die Operation GET /detections/formats ausführen. Geben Sie „all_performance“ oder „all_security“ an, um die Regel auf alle Leistungs- oder Sicherheitserkennungen anzuwenden.

`properties:` ***Reihe von Objekten***

(Optional) Die Filterkriterien für Erkennungseigenschaften.

`property:` ***Schnur***

Der Name der Eigenschaft, die gefiltert werden soll.

`operator:` ***Schnur***

Die Vergleichsmethode wird angewendet, wenn der Operandenwert mit dem Wert der Erkennungseigenschaft verglichen wird.

Die folgenden Werte sind gültig:

- `=`
- `!=`
- `~`
- `!~`
- `in`

`operand:` ***Zeichenfolge oder Zahl oder Objekt***

Der Wert, den der Filter abzugleichen versucht. Der Filter vergleicht den Wert des Operanden mit dem Wert der Erkennungseigenschaft und wendet die im Operatorparameter angegebene Vergleichsmethode an. Sie können den Operanden als Zeichenfolge, Ganzzahl oder Objekt angeben. Weitere Informationen finden Sie in der REST-API-Leitfaden.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "description": "string",
    "detection_type": "string",
    "expiration": 0,
    "offender": {
        "object_type": "string",
        "object_id": 0,
        "object_value": "array",
        "object_locality": "string",
        "object_scanner": "array",
        "object_hostname": "array"
    },
    "properties": {
        "property": "string",
        "operator": "string",
        "operand": "string"
    },
    "victim": {
        "object_type": "string",
        "object_id": 0,
        "object_value": "array",
        "object_locality": "string",
        "object_scanner": "array",
        "object_hostname": "array"
    }
}
```

`PATCH /detections/rules/hiding/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Der eindeutige Bezeichner für die Tuning-Regel.

`body:` ***Objekt***

Die zu aktualisierenden Tuning-Regelfelder.

`enabled:` ***Boolesch***

Gibt an, ob die Optimierungsregel aktiviert ist.

`expiration:` ***Zahl***

Die Zeit, in der die Tuning-Regel abläuft, ausgedrückt in Millisekunden seit der Epoche. Ein Wert von Null oder 0 gibt an, dass die Regel nicht abläuft.

`description:` ***Schnur***

Die Beschreibung der Tuning-Regel.

`offender:` ***Objekt***

Der Täter, für den diese Tuning-Regel gilt. Geben Sie ein detection_hiding_participant-Objekt an, um die Regel auf ein bestimmtes Opfer anzuwenden, oder geben Sie „Any“ an, um die Regel auf einen beliebigen Täter anzuwenden.

`object_type:` ***Schnur***

Die Art des Teilnehmer.

Die folgenden Werte sind gültig:

- `device`
- `device_group`
- `ipaddr`
- `locality_type`
- `network_locality`
- `hostname`
- `scanner_service`

`object_id:` ***Zahl***

Die ID für das Gerät, die Gerätegruppe oder den Netzwerkstandort. Diese Option ist nur gültig, wenn der Objekttyp „Gerät“, „device_group“ oder „network_locality“ ist.

`object_value:` ***Array oder String***

Die IP-Adresse oder der CIDR-Block des Teilnehmer. Sie können eine einzelne Adresse oder einen Block in einer Zeichenfolge oder mehrere Adressen oder Blöcke in einem Array angeben. Diese Option ist nur gültig, wenn der Objekttyp „ipaddr“ ist.

`object_locality:` ***Schnur***

Der Netzwerklokalitätstyp des Teilnehmer. Geben Sie entweder „extern“ oder „intern“ an. Diese Option ist nur gültig, wenn der Objekttyp „locality_type“ ist.

Die folgenden Werte sind gültig:

- `internal`
- `external`

`object_scanner:` ***Array oder String***

Der Name eines externen Scandienstes. Sie können einen einzelnen Dienst in einer Zeichenfolge oder mehrere Werte in einem Array angeben. Sie können auch „Beliebig“ angeben, um einen beliebigen Scandienst auszuwählen. Diese Option ist nur gültig, wenn der Objekttyp „scanner_service“ ist.

`object_hostname:` ***Array oder String***

Der Hostname eines Teilnehmer. Sie können einen einzelnen Hostnamen in einer Zeichenfolge oder mehrere Hostnamen in einem Array angeben. Diese Option ist nur gültig, wenn der Objekttyp „hostname“ ist.

`victim:` ***Objekt***

Das Opfer, für das diese Tuning-Regel gilt. Geben Sie ein detection_hiding_participant-Objekt an, um die Regel auf ein bestimmtes Opfer anzuwenden, oder geben Sie „Any“ an, um die Regel auf ein beliebiges Opfer anzuwenden.

`object_type:` ***Schnur***

Die Art des Teilnehmer.

Die folgenden Werte sind gültig:

- `device`
- `device_group`
- `ipaddr`
- `locality_type`
- `network_locality`
- `hostname`
- `scanner_service`

`object_id:` ***Zahl***

Die ID für das Gerät, die Gerätegruppe oder den Netzwerkstandort. Diese Option ist nur gültig, wenn der Objekttyp „Gerät“, „device_group“ oder „network_locality“ ist.

`object_value:` ***Array oder String***

Die IP-Adresse oder der CIDR-Block des Teilnehmer. Sie können eine einzelne Adresse oder einen Block in einer Zeichenfolge oder mehrere Adressen oder Blöcke in einem Array angeben. Diese Option ist nur gültig, wenn der Objekttyp „ipaddr“ ist.

`object_locality:` ***Schnur***

Der Netzwerklokalitätstyp des Teilnehmer. Geben Sie entweder „extern“ oder „intern“ an. Diese Option ist nur gültig, wenn der Objekttyp „locality_type“ ist.

Die folgenden Werte sind gültig:

- `internal`
- `external`

`object_scanner:` ***Array oder String***

Der Name eines externen Scandienstes. Sie können einen einzelnen Dienst in einer Zeichenfolge oder mehrere Werte in einem Array angeben. Sie können auch „Beliebig“ angeben, um einen beliebigen Scandienst auszuwählen. Diese Option ist nur gültig, wenn der Objekttyp „scanner_service“ ist.

`object_hostname:` ***Array oder String***

Der Hostname eines Teilnehmer. Sie können einen einzelnen Hostnamen in einer Zeichenfolge oder mehrere Hostnamen in einem Array angeben. Diese Option ist nur gültig, wenn der Objekttyp „hostname“ ist.

`properties:` ***Reihe von Objekten***

Die Filterkriterien für Erkennungseigenschaften.

`property:` ***Schnur***

Der Name der Eigenschaft, die gefiltert werden soll.

`operator:` ***Schnur***

Die Vergleichsmethode wird angewendet, wenn der Operandenwert mit dem Wert der Erkennungseigenschaft verglichen wird.

Die folgenden Werte sind gültig:

- `=`
- `!=`
- `~`
- `!~`
- `in`

`operand:` ***Zeichenfolge oder Zahl oder Objekt***

Der Wert, den der Filter abzugleichen versucht. Der Filter vergleicht den Wert des Operanden mit dem Wert der Erkennungseigenschaft und wendet die im Operatorparameter angegebene Vergleichsmethode an. Sie können den Operanden als Zeichenfolge, Ganzzahl oder Objekt angeben. Weitere Informationen finden Sie in der REST-API-Leitfaden.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "description": "string",
    "enabled": true,
    "expiration": 0,
    "offender": {
        "object_type": "string",
        "object_id": 0,
        "object_value": "array",
        "object_locality": "string",
        "object_scanner": "array",
        "object_hostname": "array"
    },
    "properties": {
        "property": "string",
        "operator": "string",
        "operand": "string"
    },
    "victim": {
        "object_type": "string",
        "object_id": 0,
        "object_value": "array",
        "object_locality": "string",
        "object_scanner": "array",
        "object_hostname": "array"
    }
}
```

`DELETE /detections/rules/hiding/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Der eindeutige Bezeichner für die Tuning-Regel.

`GET /detections/{id}/notes`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Erkennung.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "author": "string",
    "note": "string",
    "update_time": 0
}
```

`DELETE /detections/{id}/notes`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Erkennung.

`PUT /detections/{id}/notes`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Erkennung.

`body:` ***Objekt***
Die Parameter der Erkennungsnotiz.

Operandenwerte für Regeln zur Abstimmung von Erkennungseigenschaften

Die `POST /detections/rules/hiding` Mithilfe dieses Vorgangs können Sie Optimierungsregeln erstellen, die Erkennungen auf der Grundlage von Erkennungseigenschaften filtern. Sie können Filterkriterien für Erkennungseigenschaften in Objekten angeben. Jedes Objekt sollte einen eindeutigen Wert für die enthalten `operand` Feld, das für das angegebene Feld gültig ist `property` Wert.

Hinweis: Sie können gültige Eigenschaftswerte abrufen über `GET /detections/formats` Betrieb. Sehen Sie die Schlüssel des `properties` Objekt in der Antwort. Im folgenden Beispiel ist der `property` Wert ist `s3_bucket`:

```
"properties": {
  "s3_bucket": {
    "is_optional": true,
    "status": "active",
    "is_tunable": true,
    "data_type": "string"
  }
}
```

Die `is_tunable` Feld gibt an, ob Sie eine Optimierungsregel auf der Grundlage der Eigenschaft erstellen können.

`registered_domain_name`

Um Regeln für einen registrierten Domänenname auszublenden, geben Sie den `property` Wert als `registered_domain_name` und der `operand` Wert als Domänenname.

Die folgende Beispielregel verbirgt DNS-Tunnelerkennungen für `example.com`.

```
{
    "detection_type": "dns_tunnel",
    "expiration": null,
    "offender": "Any",
    "victim": "Any",
    "properties": [
        {
            "operand": "example.com",
            "operator": "=",
            "property": "registered_domain_name"
        }
    ]
}
```

`uris`

Um Regeln anhand eines URI auszublenden, geben Sie den `property` Wert als `uris` und der `operand` Wert als URI.

Die folgende Beispielregel verbirgt Erkennungen von SQL-Injection-Angriffen (SQLi) für `http://example.com/test`.

```
{
    "detection_type": "sqli_attack",
    "expiration": null,
    "offender": "Any",
    "victim": "Any",
    "properties": [
        {
            "operand": "http://example.com/test",
            "operator": "=",
            "property": "uris"
        }
      ]
}
```

`top_level_domain`

Um Regeln für einen Top-Level-Domainnamen auszublenden, geben Sie den `property` Wert als `top_level_domain` und der `operand` Wert als Top-Level-Domainname.

Die folgende Beispielregel verbirgt Erkennungen verdächtiger Top-Level-Domains für `org` Top-Level-Domain.

```
{
    "detection_type": "suspicious_tld",
    "expiration": null,
    "offender": "Any",
    "victim": "Any",
    "properties": [
        {
            "operand": "org",
            "operator": "=",
            "property": "top_level_domain"
        }
    ]
}
```

Suche mit regulären Ausdrücken (Regex)

Mit Sicherheit `property` Werte, die Zeichenfolge kann in Regex-Syntax sein. Spezifizieren Sie die `operand` Wert als Objekt, das eine `value` Parameter mit der Regex-Syntax, die Sie abgleichen möchten, und einem `is_regex` Parameter, der auf gesetzt ist `true`. Die folgende Regel filtert DNS-Tunnelerkennungen mit Domainnamen, die mit enden `example.com`.

```
{
    "detection_type": "dns_tunnel",
    "expiration": null,
    "offender": "Any",
    "victim": "Any",
    "properties": [
        {
            "operand": {
                "value": ".*?example.com",
                "is_regex": true
```

```
            },
            "operator": "=",
            "property": "registered_domain_name"
        }
    ]
}
```

Groß- und Kleinschreibung deaktivieren

Sucht standardmäßig nach einer Zeichenfolge `property` Bei Werten wird zwischen Groß- und Kleinschreibung unterschieden. Sie können jedoch die Berücksichtigung von Groß- und Kleinschreibung deaktivieren, indem Sie den Operandenwert als Objekt angeben, das eine `case_sensitive` Parameter, der auf gesetzt ist `false`.

Die folgende Regel verbirgt Erkennungen von Hacking-Tool-Domänenzugriffen mit dem ArchStrike-Hacking-Tool.

```
{
    "detection_type": "hacking_tools",
    "expiration": null,
    "offender": "Any",
    "victim": "Any",
    "properties": [
        {
            "operand": {
                "value": "archstrike",
                "case_sensitive": false
            },
            "operator": "=",
            "property": "hacking_tool"
        }
    ]
}
```

Erkennungskategorien

Das Feld Kategorien ist ein Array, das in Antworten für zurückgegeben wird `GET /detections` und `POST /detections/search` Operationen. In der folgenden Tabelle sind gültige Einträge im Array aufgeführt:

Wert	Kategorie
`sec`	Sicherheit
`sec.action`	Zielgerichtete Maßnahmen
`sec.attack`	Attacke
`sec.botnet`	Botnetz
`sec.caution`	Vorsicht
`sec.command`	Befehl und Steuerung
`sec.cryptomining`	Krypto-Mining
`sec.dos`	Diensteverweigerung
`sec.exfil`	Exfiltration
`sec.exploit`	Ausbeutung
`sec.hardening`	Aushärten

Wert	Kategorie
`sec.lateral`	Seitliche Bewegung
`sec.ransomware`	Ransomware
`sec.recon`	Aufklärung
`perf`	Aufführung
`perf.auth`	Autorisierung und Zugriffskontrolle
`perf.db`	Datenbank
`perf.network`	Netzwerk-Infrastruktur
`perf.service`	Verschlechterung des Dienstes
`perf.storage`	Aufbewahrung
`perf.virtual`	Desktop- und Anwendungsvirtualisierung
`perf.web`	Web-Applikation

Gerätegruppe

Gerätegruppen kann entweder statisch oder dynamisch sein.

Eine statische Gerätegruppe ist benutzerdefiniert. Sie erstellen eine Gerätegruppe und identifizieren dann jedes Gerät manuell und weisen es dieser Gruppe zu. Eine dynamische Gerätegruppe wird durch eine Reihe von konfigurierten Regeln definiert und automatisch verwaltet.

Sie können beispielsweise eine Gerätegruppe erstellen und dann eine Regel festlegen, um alle Geräte innerhalb eines bestimmten IP-Adressbereichs zu klassifizieren, sodass sie dieser Gruppe automatisch hinzugefügt werden. Weitere Informationen finden Sie unter Gerätegruppen.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
GET /devicegroups	Ruft alle Gerätegruppen ab, die innerhalb eines bestimmten Zeitraums aktiv waren.
POST /Gerätegruppen	Erstellen Sie eine neue Gerätegruppe.
/devicegroups/ {id} LÖSCHEN	Löscht eine Gerätegruppe.
GET /devicegroups/ {id}	Rufen Sie eine bestimmte Gerätegruppe ab.
PATCH /Gerätegruppen/ {id}	Aktualisieren Sie eine bestimmte Gerätegruppe.
GET /devicegroups/ {id} /alerts	Alles abrufen Warnungen die einer bestimmten Gerätegruppe zugewiesen sind.
POST /devicegroups/ {id} /alerts	Weisen Sie Benachrichtigungen eine bestimmte Gerätegruppe zu und heben Sie deren Zuweisung auf.
LÖSCHEN Sie /devicegroups/ {id} /alerts/ {child-id}	Heben Sie die Zuweisung einer Alarm zu einer bestimmten Gerätegruppe auf.
POST /devicegroups/ {id} /alerts/ {child-id}	Weisen Sie einer bestimmten Gerätegruppe eine Alarm zu.

Betrieb	Beschreibung
GET /devicegroups/ {id} /dashboards	Rufen Sie alle Dashboards ab, die sich auf eine bestimmte Gerätegruppe beziehen.
GET /devicegroups/ {id} /devices	Ruft alle Geräte in der Gerätegruppe ab, die innerhalb eines bestimmten Zeitfensters aktiv sind. **Hinweis:** Ein Gerät gilt als inaktiv, wenn fünf Minuten lang keine Pakete gesendet oder empfangen wurden. Wenn ein Gerät jedoch nach einem Zeitraum der Inaktivität von weniger als fünf Tagen wieder Pakete sendet oder empfängt, wird davon ausgegangen, dass das Gerät kontinuierlich aktiv war, auch während des Zeitraums der Inaktivität.
POST /devicegroups/ {id} /devices	Weisen Sie Geräte einer bestimmten statischen Gerätegruppe zu und heben Sie deren Zuweisung auf.
LÖSCHEN /devicegroups/ {id} /devices/ {child-id}	Heben Sie die Zuweisung eines Gerät zu einer bestimmten statischen Gerätegruppe auf.
POST /devicegroups/ {id} /devices/ {child-id}	Weisen Sie ein Gerät einer bestimmten statischen Gerätegruppe zu.
GET /devicegroups/ {id} /triggers	Ruft alle Trigger ab, die einer bestimmten Gerätegruppe zugewiesen sind.
POST /devicegroups/ {id} /triggers	Weisen Sie Triggern eine bestimmte Gerätegruppe zu und heben Sie deren Zuweisung auf.
LÖSCHEN /devicegroups/ {id} /triggers/ {child-id}	Heben Sie die Zuweisung eines Auslöser zu einer bestimmten Gerätegruppe auf.
POST /devicegroups/ {id} /triggers/ {Child-ID}	Weisen Sie einer bestimmten Gerätegruppe einen Auslöser zu.

Einzelheiten der Operation

`GET /devicegroups`

Geben Sie die folgenden Parameter an.

`since:` ***Zahl***
(Optional) Gibt nur Gerätegruppen zurück, die nach dieser Zeit geändert wurden, ausgedrückt in Millisekunden seit der Epoche.

`all:` ***Boolesch***
(Optional) Veraltet. Ersetzt durch den Typparameter.

`name:` ***Schnur***
(Optional) Der Regex-Suchwert zum Filtern der Gerätegruppen nach Namen.

`type:` ***Schnur***
(Optional) Gibt nur Gerätegruppen des angegebenen Typs zurück.

Die folgenden Werte sind gültig:

- `user_created`

- `built_in`
- `all`

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "built_in": true,
    "description": "string",
    "dynamic": true,
    "editors": [],
    "field": "string",
    "filter": {},
    "id": 0,
    "include_custom_devices": true,
    "mod_time": 0,
    "name": "string",
    "value": "string"
}
```

`GET /devicegroups/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Gerätegruppe.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "built_in": true,
    "description": "string",
    "dynamic": true,
    "editors": [],
    "field": "string",
    "filter": {},
    "id": 0,
    "include_custom_devices": true,
    "mod_time": 0,
    "name": "string",
    "value": "string"
}
```

`POST /devicegroups`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Wendet die angegebenen Eigenschaftswerte auf die neue Gerätegruppe an.

`description:` ***Schnur***

Eine optionale Beschreibung der Gerätegruppe.

`name:` ***Schnur***

Der benutzerfreundliche Name für die Gerätegruppe.

`include_custom_devices:` ***Boolesch***

(Optional) Veraltet. Ersetzt durch den Filterparameter.

`dynamic:` ***Boolesch***

(Optional) Gibt an, ob die Gerätegruppe dynamisch ist.

`field:` ***Schnur***

Veraltet. Ersetzt durch den Filterparameter.

Die folgenden Werte sind gültig:

- `any`
- `name`
- `ip address`
- `mac address`
- `vendor`
- `type`
- `tag`
- `vlan`
- `activity`
- `node`
- `discover time`

`value:` ***Objekt***

(Optional) Veraltet. Ersetzt durch den Filterparameter.

`filter:` ***Objekt***

(Optional) Geben Sie die Filterkriterien für Suchergebnisse an.

`field:` ***Schnur***

Der Name des Feldes, nach dem die Ergebnisse gefiltert werden sollen. Die Suche vergleicht den Inhalt des Feldparameters mit dem Wert des Operandenparameters.

Die folgenden Werte sind gültig:

- `name`
- `ipaddr`
- `macaddr`
- `vendor`
- `tag`
- `activity`
- `node`
- `vlan`
- `discover_time`
- `role`
- `dns_name`
- `dhcp_name`
- `netbios_name`
- `cdp_name`
- `custom_name`
- `software`
- `model`
- `is_critical`
- `instance_id`
- `instance_name`
- `instance_type`
- `cloud_account`
- `vpc_id`
- `subnet_id`
- `is_active`
- `network_locality_type`

- `network_locality_id`
- `id`

`operator:` ***Schnur***

Die Vergleichsmethode, die angewendet wird, wenn der Operandenwert mit dem Feldinhalt verglichen wird. Alle Filterobjekte benötigen einen Operator.

Die folgenden Werte sind gültig:

- `>`
- `<`
- `<=`
- `>=`
- `=`
- `!=`
- `startswith`
- `and`
- `or`
- `not`
- `exists`
- `not_exists`
- `~`
- `!~`

`operand:` ***Zeichenfolge oder Zahl oder Objekt***

Der Wert, den die Abfrage abzugleichen versucht. Die Abfrage vergleicht den Wert des Operanden mit dem Inhalt des Feldparameters und wendet die durch den Operatorparameter angegebene Vergleichsmethode an. Sie können den Operanden als Zeichenfolge, Ganzzahl oder Objekt angeben. Informationen zu Objektwerten finden Sie in der REST-API-Leitfaden.

`rules:` ***Reihe von Objekten***

Ein Array aus einem oder mehreren Filterobjekten, die rekursiv eingebettet werden können. Für diesen Parameter sind nur die Operatoren „und“, „oder“ oder „nicht“ zulässig.

`editors:` ***Reihe von Zeichenketten***

(Optional) Die Liste der Benutzer, die die Gerätegruppe bearbeiten können.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "description": "string",
    "dynamic": true,
    "editors": [],
    "field": "string",
    "filter": {
        "field": "string",
        "operator": "string",
        "operand": "string",
        "rules": []
    },
    "include_custom_devices": true,
    "name": "string",
    "value": "string"
}
```

```
DELETE /devicegroups/{id}
```

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

```
PATCH /devicegroups/{id}
```

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Wendet die Aktualisierungen der angegebenen Eigenschaftswerte auf eine bestimmte Gerätegruppe an.

`description:` ***Schnur***
Eine optionale Beschreibung der Gerätegruppe.

`name:` ***Schnur***
Der benutzerfreundliche Name für die Gerätegruppe.

`include_custom_devices:` ***Boolesch***
(Optional) Veraltet. Ersetzt durch den Filterparameter.

`field:` ***Schnur***
Veraltet. Ersetzt durch den Filterparameter.

Die folgenden Werte sind gültig:

- `any`
- `name`
- `ip address`
- `mac address`
- `vendor`
- `type`
- `tag`
- `vlan`
- `activity`
- `node`
- `discover time`

`value:` ***Objekt***
(Optional) Veraltet. Ersetzt durch den Filterparameter.

`filter:` ***Objekt***
(Optional) Geben Sie die Filterkriterien für Suchergebnisse an.

`editors:` ***Reihe von Zeichenketten***
(Optional) Die Liste der Benutzer, die die Gerätegruppe bearbeiten können.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "description": "string",
    "editors": [],
    "field": "string",
    "filter": {},
    "include_custom_devices": true,
    "name": "string",
    "value": "string"
}
```

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`GET /devicegroups/{id}/alerts`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`direct_assignments_only:` ***Boolesch***
(Optional) Beschränken Sie die Ergebnisse auf Warnungen, die direkt der Gerätegruppe zugewiesen sind.

`POST /devicegroups/{id}/alerts/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`DELETE /devicegroups/{id}/alerts/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für die Alarm.

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`POST /devicegroups/{id}/alerts`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Die Liste der eindeutigen Identifikatoren für Warnmeldungen, die der Gerätegruppe zugewiesen sind oder nicht.

`assign:` ***Reihe von Zahlen***
IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***
IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`GET /devicegroups/{id}/triggers`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`direct_assignments_only:` ***Boolesch***
(Optional) Beschränken Sie die Ergebnisse auf Trigger, die direkt der Gerätegruppe zugewiesen sind.

`POST /devicegroups/{id}/triggers/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für den Auslöser.

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`DELETE /devicegroups/{id}/triggers/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für den Auslöser.

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`POST /devicegroups/{id}/triggers`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Die Liste der eindeutigen Identifikatoren für Trigger, die der Gerätegruppe zugewiesen oder nicht zugewiesen sind.

`assign:` ***Reihe von Zahlen***
IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***
IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`POST /devicegroups/{id}/devices/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für ein Gerät.

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`DELETE /devicegroups/{id}/devices/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für ein Gerät.

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`POST /devicegroups/{id}/devices`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Die Liste der eindeutigen Identifikatoren für Geräte, die der Gerätegruppe zugewiesen sind oder nicht.

`assign:` ***Reihe von Zahlen***
IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***
IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`GET /devicegroups/{id}/devices`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

`active_from:` ***Zahl***
(Optional) Der Anfangszeitstempel für die Anfrage. Gibt nur Geräte zurück, die nach dieser Zeit aktiv sind. Die Zeit wird in Millisekunden seit der Epoche ausgedrückt. 0 gibt den Zeitpunkt der Anfrage an. Ein negativer Wert wird relativ zur aktuellen Uhrzeit ausgewertet. Die Standardeinheit für einen negativen Wert ist Millisekunden, aber andere Einheiten können mit einem Einheitensuffix angegeben werden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`active_until:` ***Zahl***
(Optional) Der Endzeitstempel für die Anfrage. Nur Gerät zurückgeben, die vor diesem Zeitpunkt aktiv waren. Folgt den gleichen Zeitwertrichtlinien wie der active_from−Parameter.

`limit:` ***Zahl***
(Optional) Begrenzen Sie die Anzahl der zurückgegebenen Geräte.

`offset:` ***Zahl***
(Optional) Überspringen Sie die ersten n Geräteergebnisse. Dieser Parameter wird häufig mit dem Grenzwertparameter kombiniert.

```
GET /devicegroups/{id}/dashboards
```

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für die Gerätegruppe.

Gerät

Geräte sind Objekte in Ihrem Netzwerk, die von Ihrem ExtraHop-System identifiziert und klassifiziert wurden. Weitere Informationen finden Sie unter Geräte.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
GET /Geräte	Rufen Sie alle Geräte ab, die innerhalb eines bestimmten Zeitraums aktiv waren. Weitere Informationen finden Sie unter Extrahieren Sie die Geräteliste über die REST-API. **Hinweis:** Ein Gerät gilt als inaktiv, wenn fünf Minuten lang keine Pakete gesendet oder empfangen wurden. Wenn ein Gerät jedoch nach einem Zeitraum der Inaktivität von weniger als fünf Tagen wieder Pakete sendet oder empfängt, wird davon ausgegangen, dass das Gerät kontinuierlich aktiv war, auch während des Zeitraums der Inaktivität.
POST /Geräte/Suche	Rufen Sie alle Geräte ab, die bestimmten Kriterien entsprechen. Weitere Informationen finden Sie unter Suchen Sie über die REST-API nach einem Gerät. **Hinweis:** Ein Gerät gilt als inaktiv, wenn fünf Minuten lang keine Pakete gesendet oder empfangen wurden. Wenn ein Gerät jedoch nach einem Zeitraum der Inaktivität von weniger als fünf Tagen wieder Pakete sendet oder empfängt, wird davon ausgegangen, dass das Gerät kontinuierlich aktiv war, auch während des Zeitraums der Inaktivität.
GET /devices/ {id}	Rufen Sie ein bestimmtes Gerät ab.
PATCH /Geräte/ {id}	Aktualisieren Sie ein bestimmtes Gerät.
GET /devices/ {id} /activity	Ruft alle Aktivitäten für ein Gerät ab.
GET /devices/ {id} /alerts	Alles abrufen Warnungen die einem bestimmten Gerät zugewiesen sind.
POST /devices/ {id} /alerts	Weisen Sie Warnmeldungen ein bestimmtes Gerät zu und heben Sie die Zuweisung auf.
LÖSCHEN Sie /devices/ {id} /alerts/ {child-id}	Heben Sie die Zuweisung einer Alarm zu einem bestimmten Gerät auf.

Betrieb	Beschreibung
POST /devices/ {id} /alerts/ {child-id}	Weisen Sie einem bestimmten Gerät eine Alarm zu.
GET /devices/ {id} /dashboards	Rufen Sie alle Dashboards ab, die sich auf ein bestimmtes Gerät beziehen.
GET /devices/ {id} /devicegroups	Alles abrufen Gerätegruppen die einem bestimmten Gerät zugewiesen sind.
POST /devices/ {id} /devicegroups	Weisen Sie Gerätegruppen ein bestimmtes Gerät zu und heben Sie die Zuweisung auf.
LÖSCHEN Sie /devices/ {id} /devicegroups/ {child-id}	Heben Sie die Zuweisung einer Gerätegruppe zu einem bestimmten Gerät auf.
POST /devices/ {id} /devicegroups/ {child-id}	Weisen Sie einem bestimmten Gerät eine Gerätegruppe zu.
GET /devices/ {id} /dnsnames	Ruft alle DNS-Namen ab, die einem bestimmten Gerät zugeordnet sind.
GET /devices/ {id} /ipaddrs	Ruft alle IP-Adressen ab, die innerhalb eines bestimmten Zeitraums mit einem bestimmten Gerät verknüpft waren.
GET /devices/ {id} /software	Ruft eine Liste der Software ab, die auf dem angegebenen Gerät ausgeführt wird.
GET /devices/ {id} /tags	Ruft alle Tags ab, die einem bestimmten Gerät zugewiesen sind.
POST /Geräte/ {id} /tags	Weisen Sie Tags ein bestimmtes Gerät zu und heben Sie die Zuweisung auf.
LÖSCHEN Sie /devices/ {id} /tags/ {child-id}	Heben Sie die Zuweisung eines Tags zu einem bestimmten Gerät auf.
POST /Geräte/ {id} /tags/ {Kinder-ID}	Weisen Sie einem bestimmten Gerät ein Tag zu.
GET /devices/ {id} /triggers	Ruft alle Trigger ab, die einem bestimmten Gerät zugewiesen sind.
POST /Geräte/ {id} /Trigger	Weisen Sie Triggern ein bestimmtes Gerät zu und heben Sie die Zuweisung auf.
LÖSCHEN /devices/ {id} /triggers/ {child-id}	Heben Sie die Zuweisung eines Auslöser zu einem bestimmten Gerät auf.
POST /Geräte/ {id} /triggers/ {Kinder-ID}	Weisen Sie einem bestimmten Gerät einen Auslöser zu.

Einzelheiten der Operation

`GET /devices`

Geben Sie die folgenden Parameter an.

`active_from:` ***Zahl***

(Optional) Der Anfangszeitstempel für die Anfrage. Gibt nur Geräte zurück, die nach dieser Zeit aktiv sind. Die Zeit wird in Millisekunden seit der Epoche ausgedrückt. 0 gibt den Zeitpunkt der Anfrage an. Ein negativer Wert wird relativ zur aktuellen Uhrzeit ausgewertet. Die Standardeinheit

für einen negativen Wert ist Millisekunden, aber andere Einheiten können mit einem Einheitensuffix angegeben werden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`active_until:` ***Zahl***

(Optional) Der Endzeitstempel für die Anfrage. Nur Gerät zurückgeben, die vor diesem Zeitpunkt aktiv waren. Folgt den gleichen Zeitwertrichtlinien wie der active_from Parameter.

`limit:` ***Zahl***

(Optional) Beschränken Sie die Anzahl der zurückgegebenen Geräte auf die angegebene Höchstzahl.

`offset:` ***Zahl***

(Optional) Überspringen Sie die ersten n Geräteergebnisse. Dieser Parameter wird häufig mit dem Grenzwertparameter kombiniert.

`search_type:` ***Schnur***

Gibt das zu durchsuchende Feld an.

Die folgenden Werte sind gültig:

- `any`
- `name`
- `discovery_id`
- `ip address`
- `mac address`
- `vendor`
- `type`
- `tag`
- `activity`
- `node`
- `vlan`
- `discover time`

`value:` ***Schnur***

(Optional) Gibt die Suchkriterien an.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "activity": [],
    "analysis": "string",
    "analysis_level": 0,
    "auto_role": "string",
    "cdp_name": "string",
    "cloud_account": "string",
    "cloud_instance_description": "string",
    "cloud_instance_id": "string",
    "cloud_instance_name": "string",
    "cloud_instance_type": "string",
    "critical": true,
    "custom_criticality": "string",
    "custom_make": "string",
    "custom_model": "string",
    "custom_name": "string",
    "custom_type": "string",
    "default_name": "string",
    "description": "string",
    "device_class": "string",
    "dhcp_name": "string",
    "discover_time": 0,
    "discovery_id": "string",
    "display_name": "string",
    "dns_name": "string",
```

```
    "extrahop_id": "string",
    "id": 0,
    "ipaddr4": "string",
    "ipaddr6": "string",
    "is_l3": true,
    "last_seen_time": 0,
    "macaddr": "string",
    "mod_time": 0,
    "model": "string",
    "model_override": "string",
    "netbios_name": "string",
    "node_id": 0,
    "on_watchlist": true,
    "parent_id": 0,
    "role": "string",
    "subnet_id": "string",
    "user_mod_time": 0,
    "vendor": "string",
    "vlanid": 0,
    "vpc_id": "string"
}
```


`POST /devices/search`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Gerätekriterien.

`active_from:` ***Zahl***

(Optional) Der Anfangszeitstempel für die Anfrage. Gibt nur Geräte zurück, die nach dieser Zeit aktiv sind. Die Zeit wird in Millisekunden seit der Epoche ausgedrückt. 0 gibt den Zeitpunkt der Anfrage an. Ein negativer Wert wird relativ zur aktuellen Uhrzeit ausgewertet. Die Standardeinheit für einen negativen Wert ist Millisekunden, aber andere Einheiten können mit einem Einheitensuffix angegeben werden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`active_until:` ***Zahl***

(Optional) Der Endzeitstempel für die Anfrage. Gibt nur Geräte zurück, die vor diesem Zeitpunkt aktiv waren. Folgt den gleichen Zeitwertrichtlinien wie der active_from Parameter.

`limit:` ***Zahl***

(Optional) Beschränken Sie die Anzahl der zurückgegebenen Geräte auf die angegebene Höchstzahl.

`offset:` ***Zahl***

(Optional) Überspringen Sie die angegebene Anzahl von Geräten. Dieser Parameter wird häufig mit dem Grenzwertparameter kombiniert, um Ergebnismengen zu paginieren.

`filter:` ***Objekt***

(Optional) Geben Sie die Filterkriterien für Suchergebnisse an.

`field:` ***Schnur***

Der Name des Feldes, nach dem die Ergebnisse gefiltert werden sollen. Die Suche vergleicht den Inhalt des Feldparameters mit dem Wert des Operandenparameters.

Die folgenden Werte sind gültig:

- `name`
- `discovery_id`
- `ipaddr`
- `macaddr`

- vendor
- tag
- activity
- node
- vlan
- discover_time
- role
- dns_name
- dhcp_name
- netbios_name
- cdp_name
- custom_name
- software
- model
- is_critical
- instance_id
- instance_name
- instance_type
- cloud_account
- vpc_id
- subnet_id
- is_active
- analysis
- network_locality_type
- network_locality_id
- id

`operator:` ***Schnur***

Die Vergleichsmethode, die angewendet wird, wenn der Operandenwert mit dem Feldinhalt verglichen wird. Alle Filterobjekte benötigen einen Operator.

Die folgenden Werte sind gültig:

- >
- <
- <=
- >=
- =
- !=
- startswith
- and
- or
- not
- exists
- not_exists
- ~
- !~
- in
- not_in

`operand:` ***Zeichenfolge oder Zahl oder Objekt oder Array***

Der Wert, den die Abfrage abzugleichen versucht. Die Abfrage vergleicht den Wert des Operanden mit dem Inhalt des Feldparameters und wendet die durch den

Operatorparameter angegebene Vergleichsmethode an. Sie können den Operanden als Zeichenfolge, Ganzzahl oder Objekt angeben. Informationen zu Objektwerten finden Sie in der REST-API-Leitfaden.

`rules:` ***Reihe von Objekten***

Ein Array aus einem oder mehreren Filterobjekten, die rekursiv eingebettet werden können. Für diesen Parameter sind nur die Operatoren „und“, „oder“ oder „nicht“ zulässig.

`result_fields:` ***Reihe von Zeichenketten***

(Optional) Gibt die angegebenen Felder und die Geräte-ID zurück. Wenn diese Option nicht angegeben ist, werden alle Felder zurückgegeben.

Die folgenden Werte sind gültig:

- `mod_time`
- `node_id`
- `id`
- `extrahop_id`
- `discovery_id`
- `display_name`
- `description`
- `user_mod_time`
- `discover_time`
- `vlanid`
- `parent_id`
- `macaddr`
- `vendor`
- `is_l3`
- `ipaddr4`
- `ipaddr6`
- `device_class`
- `default_name`
- `custom_name`
- `cdp_name`
- `dhcp_name`
- `netbios_name`
- `dns_name`
- `custom_type`
- `auto_role`
- `analysis_level`
- `analysis`
- `role`
- `on_watchlist`
- `last_seen_time`
- `activity`
- `model`
- `model_override`
- `custom_make`
- `custom_model`
- `critical`
- `custom_criticality`
- `cloud_instance_id`

- `cloud_instance_type`
- `cloud_instance_description`
- `cloud_instance_name`
- `cloud_account`
- `vpc_id`
- `subnet_id`

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "active_from": 0,
    "active_until": 0,
    "filter": {
        "field": "string",
        "operator": "string",
        "operand": "string",
        "rules": []
    },
    "limit": 0,
    "offset": 0,
    "result_fields": []
}
```

`GET /devices/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "activity": [],
    "analysis": "string",
    "analysis_level": 0,
    "auto_role": "string",
    "cdp_name": "string",
    "cloud_account": "string",
    "cloud_instance_description": "string",
    "cloud_instance_id": "string",
    "cloud_instance_name": "string",
    "cloud_instance_type": "string",
    "critical": true,
    "custom_criticality": "string",
    "custom_make": "string",
    "custom_model": "string",
    "custom_name": "string",
    "custom_type": "string",
    "default_name": "string",
    "description": "string",
    "device_class": "string",
    "dhcp_name": "string",
    "discover_time": 0,
    "discovery_id": "string",
    "display_name": "string",
    "dns_name": "string",
    "extrahop_id": "string",
    "id": 0,
    "ipaddr4": "string",
```

```
    "ipaddr6": "string",
    "is_l3": true,
    "last_seen_time": 0,
    "macaddr": "string",
    "mod_time": 0,
    "model": "string",
    "model_override": "string",
    "netbios_name": "string",
    "node_id": 0,
    "on_watchlist": true,
    "parent_id": 0,
    "role": "string",
    "subnet_id": "string",
    "user_mod_time": 0,
    "vendor": "string",
    "vlanid": 0,
    "vpc_id": "string"
}
```

`PATCH /devices/{id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Wenden Sie die angegebenen Eigenschaftswertaktualisierungen auf das Gerät an.

`id:` ***Zahl***
Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`GET /devices/{id}/activity`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "device_id": 0,
    "from_time": 0,
    "id": 0,
    "mod_time": 0,
    "stat_name": "string",
    "until_time": 0
}
```

`GET /devices/{id}/ipaddrs`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`from:` ***Zahl***
(Optional) Ruft IP-Adressen ab, die dem Gerät nach dem angegebenen Datum zugeordnet wurden, ausgedrückt in Millisekunden seit der Epoche.

`until:` ***Zahl***

(Optional) Ruft IP-Adressen ab, die dem Gerät vor dem angegebenen Datum zugeordnet wurden, ausgedrückt in Millisekunden seit der Epoche.

`GET /devices/{id}/dnsnames`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`from:` ***Zahl***

(Optional) Ruft DNS-Namen ab, die dem Gerät nach dem angegebenen Datum zugeordnet wurden, ausgedrückt in Millisekunden seit der Epoche.

`until:` ***Zahl***

(Optional) Ruft DNS-Namen ab, die dem Gerät vor dem angegebenen Datum zugeordnet wurden, ausgedrückt in Millisekunden seit der Epoche.

`GET /devices/{id}/triggers`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`direct_assignments_only:` ***Boolesch***

(Optional) Beschränken Sie die Ergebnisse auf Trigger, die dem Gerät direkt zugewiesen sind.

`POST /devices/{id}/triggers`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Eine Liste eindeutiger Identifikatoren für Trigger, die dem Gerät zugewiesen sind oder nicht.

`assign:` ***Reihe von Zahlen***

IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***

IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`POST /devices/{id}/triggers/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für den Auslöser.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`DELETE /devices/{id}/triggers/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für den Auslöser.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`GET /devices/{id}/dashboards`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`GET /devices/{id}/devicegroups`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät.

`active_from:` ***Zahl***

(Optional) Der Anfangszeitstempel für die Anfrage. Gibt nur dynamische Gerätegruppen zurück, zu denen das Gerät nach dieser Zeit gehörte. Die Zeit wird in Millisekunden seit der Epoche ausgedrückt. 0 gibt den Zeitpunkt der Anfrage an. Ein negativer Wert wird relativ zur aktuellen Uhrzeit ausgewertet. Die Standardeinheit für einen negativen Wert ist Millisekunden, aber andere Einheiten können mit einem Einheitensuffix angegeben werden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`active_until:` ***Zahl***

(Optional) Der Endzeitstempel für die Anfrage. Gibt nur dynamische Gerätegruppen zurück, zu denen das Gerät vor diesem Zeitpunkt gehörte. Folgt den gleichen Zeitwertrichtlinien wie der active_from Parameter.

`POST /devices/{id}/devicegroups`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Liste der eindeutigen Identifikatoren für Gerätegruppen, die dem Gerät zugewiesen sind oder nicht.

`assign:` ***Reihe von Zahlen***

IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***

IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`POST /devices/{id}/devicegroups/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für die Gerätegruppe.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`DELETE /devices/{id}/devicegroups/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für die Gerätegruppe.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`GET /devices/{id}/tags`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`POST /devices/{id}/tags`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Eine Liste eindeutiger Identifikatoren für Tags, die dem Gerät zugewiesen sind oder nicht.

`assign:` ***Reihe von Zahlen***

IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***

IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`POST /devices/{id}/tags/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für das Tag.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`DELETE /devices/{id}/tags/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für das Tag.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`GET /devices/{id}/alerts`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`direct_assignments_only:` ***Boolesch***

(Optional) Beschränken Sie die Ergebnisse auf Warnungen, die dem Gerät direkt zugewiesen sind.

`POST /devices/{id}/alerts`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Liste der eindeutigen Identifikatoren für Warnmeldungen, die dem Gerät zugewiesen sind oder nicht.

`assign:` ***Reihe von Zahlen***

IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***

IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

```
POST /devices/{id}/alerts/{child-id}
```

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

```
DELETE /devices/{id}/alerts/{child-id}
```

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für die Alarm.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

```
GET /devices/{id}/software
```

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Gerät, die als API-ID auf der Geräteseite im ExtraHop-System angezeigt wird.

`from:` ***Zahl***

(Optional) Gibt Software zurück, die nach dem angegebenen Datum auf dem Gerät beobachtet wurde, ausgedrückt in Millisekunden seit der Epoche.

`until:` ***Zahl***

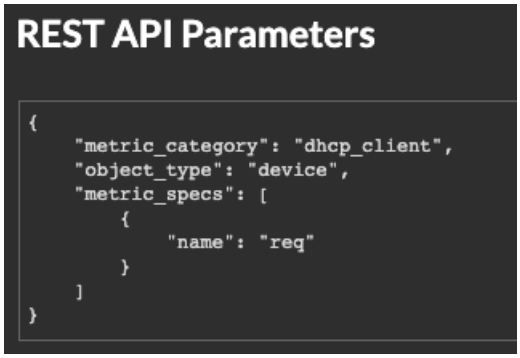
(Optional) Gibt Software zurück, die vor dem angegebenen Datum auf dem Gerät beobachtet wurde, ausgedrückt in Millisekunden seit der Epoche.

Operandenwerte für die Gerätesuche

Mit dem POST /devices/search-Vorgang können Sie anhand der in Filterobjekten angegebenen Kriterien nach Geräten suchen. Jedes Objekt sollte einen eindeutigen Wert für den enthalten `operand` Feld, das für das angegebene Feld gültig ist `field` Wert.

```
activity
```

Um nach Metrik Aktivität zu suchen, geben Sie den `field` Wert als `activity` und die `operand` Wert als `metric_category`. Du kannst finden `metric_category` Werte im Abschnitt REST-API-Parameter des Metrikkatalogs.



Das folgende Beispiel gibt Ergebnisse für Geräte zurück, die allen für einen DHCP-Client klassifizierten Metrikaktivitäten entsprechen, z. B. der Anzahl der gesendeten DHCP-Anfragen .

```
{
    "filter":  {
        "field": "activity",
        "operand": "dhcp_client",
        "operator": "="
    }
}
```

Hinweis: Rufen Sie programmgesteuert eine Liste aller Metrik Aktivitäten für ein Gerät über die `GET /devices/{id}/activity` Betrieb. Das `stat_name` Wert entspricht dem `metric_category` Wert in der `metric_catalog`, nach dem letzten Punkt.

In der folgenden Beispielantwort ist die `stat_name` Wert ist `extrahop.device.dhcp_client`. Entfernen Sie den Text vor dem letzten Punkt, um den zu identifizieren `metric_catalog` Wert von `dhcp_client`.

```
{
    "id": 198606,
    "from_time": 1581537120000,
    "until_time": 1581542520000,
    "mod_time": 1581542533963,
    "device_id": 30096,
    "stat_name": "extrahop.device.dhcp_client"
}
```

Analyse

Um nach Geräteanalyseebene zu suchen, geben Sie den `field` Wert als `analysis` und die `operand` Wert als eine der folgenden Zeichenketten:

Standard
Geräte in Standardanalyse.

fortgeschrittene
Geräte in Erweiterte Analyse.

Entdeckung
Geräte in Entdeckungsmodus.

l2_exempt
Geräte in L2 Parent Analysis.

flow_log
Geräte in der Strömungsanalyse.

`discover_time`

Um nach einem Zeitraum zu suchen, geben Sie den `field` Wert als `discover_time` und ein `operand` Wert mit `from` und `until` Parameter, wobei es sich bei den Werten um Daten handelt, ausgedrückt in Millisekunden seit der Epoche.

Das folgende Beispiel gibt Ergebnisse für alle Geräteaktivitäten zurück, die am 21. August 2019 zwischen 13:00 Uhr und 15:00 Uhr stattfanden.

```
{
    "filter":  {
        "field": "discover_time",
        "operand": {
            "from": "1566392400000",
            "until": "1566399600000"
        },
        "operator": "="
    }
}
```

`discovery_id`

Um nach der eindeutigen ID für das Gerät zu suchen, geben Sie die `field` Wert als `discovery_id` und die `operand` Wert als Discovery-ID.

```
{
  "filter": {
    "field": "discovery_id",
    "operand": "c12vf90qpg290000",
    "operator": "="
  }
}
```

id

Um mehrere Geräte abzurufen, geben Sie den Feldwert als an `id`, das `operator` Wert als `in`, und die `operand` Wert als Array von IDs.

```
{
  "filter": {
    "field": "id",
    "operand": [5388,5387],
    "operator": "in"
  }
}
```

Um Geräte aus den Suchergebnissen auszuschließen, geben Sie einen Filter mit mehreren Regeln an und geben Sie eine Regel mit dem Feldwert als an `id`, das `operator` Wert als `not_in`, und die `operand` Wert als Array von IDs.

```
{
  "filter": {
    "operator": "and",
    "rules": [
      {
        "field": "id",
        "operand": [5388,5387],
        "operator": "not_in"
      },
      {
        "field": "discover_time",
```

```
        "operand": {
          "from": "1692984750000",
          "until": "1693416750000"
        },
        "operator": "="
      }
    ]
  }
}
```

ist_aktiv

Um nach Geräten zu suchen, die in den letzten 30 Minuten aktiv waren, geben Sie den Feldwert als an `is_active` und die `operand` Wert als boolescher Wert.

```
{
  "filter": {
    "field": "is_active",
    "operand": true,
    "operator": "="
  }
}
```

`ipaddr`

Um nach der IP-Adresse zu suchen, geben Sie den `field` Wert als `ipaddr` und die `operand` Wert als IP-Adresse oder CIDR-Block.

```
{
  "filter": {
    "field": "ipaddr",
    "operand": "192.168.12.0/28",
    "operator": "="
  }
}
```

`node`

Um nach der eindeutigen ID eines zu suchen Sensor, spezifizieren Sie die `field` Wert als `node` und die `operand` Wert als Sensor UUID.

```
{
  "filter": {
    "field": "node",
    "operand": "qqvsplfa-zxsk-32l0-19g1-076vfr42pw31",
    "operator": "="
  }
}
```

`macaddr`

Um nach der MAC-Adresse eines Gerät zu suchen, geben Sie den Feldwert als an `macaddr` und der Operandenwert als MAC-Adresse des Gerät. Das folgende Beispiel gibt Ergebnisse für Geräte mit einer MAC-Adresse von `C1:1C:N2:0Q:PJ:10` oder `C1:1C:N2:0Q:PJ:11`.

```
{
  "filter": {
    "operator": "or",
    "rules": [
```

```
      {
        "field": "macaddr",
        "operand": "C1:1C:N2:0Q:PJ:10",
        "operator": "="
      },
      {
        "field": "macaddr",
        "operand": "C1:1C:N2:0Q:PJ:11",
        "operator": "="
      }
    ]
  }
}
```

`model`

Um nach dem Gerätemodell zu suchen, geben Sie den `field` Wert als `model`. Wenn der Betreiber `=`, `!=`, `exists`, oder `not_exists`, geben Sie den Operanden als Modell-ID an, die Sie in der `model` Feld von `POST /device/search` Antworten.

```
{
  "filter": {
    "field": "model",
    "operand": "apple_ipad_pro_12_9_inch_wifi_cellular_5th_gen",
    "operator": "="
  }
}
```

Wenn der Betreiber `~` oder `!~`, geben Sie den Operanden als Namen der Marke und des Modells an, die Sie bei der Suche nach einem Gerät im ExtraHop-System einsehen können.

```
{
  "filter": {
    "field": "model",
    "operand": "Apple iPad Pro",
    "operator": "~"
  }
}
```

`name`

Um nach dem Anzeigenamen des Gerät zu suchen, geben Sie den `field` Wert als Name und der `operand` Wert als Gerätename oder als Regex-Zeichenfolge.

```
{
  "filter": {
    "field": "name",
    "operand": "VMware B2CEB6",
    "operator": "="
  }
}
```

Netzwerk_Lokalitäts-ID

Um nach Netzwerklokalität zu suchen, geben Sie den `field` Wert als `network_locality_id` und der Operandenwert als Netzwerklokalitäts-ID.

```
{
  "filter": {
    "field": "network_locality_id",
```

```
    "operand": 123,
    "operator": "="
  }
}
```

`role`

Um nach der Geräterolle zu suchen, geben Sie die `field` Wert als `role` und die `operand` Wert als Geräterolle.

```
{
  "filter": {
    "field": "role",
    "operand": "voip_phone",
    "operator": "="
  }
}
```

`software`

Um nach der auf dem Gerät ausgeführten Software zu suchen, geben Sie die `field` Wert als `software` und die `operand` Wert als die ID, die dieser Software auf dem ExtraHop-System zugeordnet ist.

```
{
  "filter": {
    "field": "software",
    "operand": "windows_10",
    "operator": "="
  }
}
```

Hinweis: Rufen Sie programmgesteuert eine Liste aller Software-IDs ab, die einem Gerät zugeordnet sind, über den `GET /devices/{id}/software` Betrieb.

In der folgenden Beispielantwort ist die `id` Wert für die Software ist `windows_10`.

```
[
  {
    "software_type": "OS",
    "name": "Windows",
    "version": "10",
    "description": null,
    "id": "windows_10"
  }
]
```

`software_type`

Um nach der Art der auf dem Gerät ausgeführten Software zu suchen, geben Sie die `field` Wert als `software_type` und die `operand` Wert als Softwaretyp-ID.

```
{
  "filter": {
    "field": "software_type",
    "operand": "OS",
    "operator": "="
  }
}
```

Hinweis: Rufen Sie programmgesteuert eine Liste aller Softwaretyp-IDs ab, die einem Gerät zugeordnet sind, über die `GET /devices/{id}/software` Betrieb.

In der folgenden Beispielantwort lautet der ID-Wert für den Softwaretyp `OS`.

```
[
  {
    "software_type": "OS",
    "name": "Windows",
    "version": "10",
    "description": null,
    "id": "windows_10"
  }
]
```

`tag`

Um nach einem Geräte-Tag zu suchen, geben Sie das `field` Wert als `tag` und die `operand` Wert als Tag-Name oder als Regex-Zeichenfolge.

```
{
  "filter": {
    "field": "tag",
    "operand": "Custom Tag",
    "operator": "="
  }
}
```

Hinweis: Rufen Sie programmgesteuert eine Liste aller Geräte-Tags über die `GET /devices/{id}/tags` Betrieb.

In der folgenden Beispielantwort ist die `name` Der Wert für das Tag ist `Custom Tag`.

```
[
  {
    "mod_time": 1521577040934,
    "id": 19,
    "name": "Custom Tag"
  }
]
```

vlan

Um nach der ID eines VLANs zu suchen, geben Sie den `field` Wert als `vlan` und die `operand` Wert als ID des VLAN.

```
{
  "filter": {
    "field": "vlan",
    "operand": "0",
    "operator": "="
  }
}
```

Suche mit regulären Ausdrücken (Regex)

Mit Sicherheit `field` Werte, die Zeichenfolge kann in Regex-Syntax vorliegen. Spezifizieren Sie die `operand` Wert als Objekt, das einen `value` Parameter mit der Regex-Syntax, die Sie abgleichen möchten,

und einem `is_regex` Parameter, der auf gesetzt ist `true`. Das folgende Beispiel gibt Ergebnisse für alle DNS-Namen zurück, die mit enden `com`.

```
{
    "filter":  {
        "field": "dns_name",
        "operand": {
            "value": ".*?com",
            "is_regex": true
        },
        "operator": "="
    }
}
```

Ein `operand` Feld mit Regex-Syntax ist gültig für Folgendes `field` Werte:

- CDP_Name
- benutzerdefinierter_Name
- DNS-Name
- dhcp_name
- Modell
- Name
- netbios_name
- Software
- Tag
- Lieferant

Unterstützte Zeiteinheiten

Für die meisten Parameter ist die Standardeinheit für die Zeitmessung Millisekunden. Die folgenden Parameter geben jedoch alternative Zeiteinheiten wie Minuten und Stunden zurück oder akzeptieren diese:

- Gerät
  - aktive_von
  - aktiv_bis
- Gerätegruppe
  - aktive_von
  - aktiv_bis
- Metriken
  - von
  - bis
- Protokoll aufzeichnen
  - von
  - bis
  - kontext_ttl

Die folgende Tabelle zeigt die unterstützten Zeiteinheiten:

Zeiteinheit	Einheitensuffix
Jahr	`y`
Monat	`M`
Woche	`w`
Tag	`d`

Zeiteinheit	Einheitensuffix
Stunde	h
Minute	m
Zweiter	s
Millisekunde	ms

Um für einen Parameter eine andere Zeiteinheit als Millisekunden anzugeben, hängen Sie das Einheitensuffix an den Wert an. Um beispielsweise Geräte anzufordern, die in den letzten 30 Minuten aktiv waren, geben Sie den folgenden Parameterwert an:

```
GET /api/v1/devices?active_from=-30m
```

Das folgende Beispiel spezifiziert eine Suche nach HTTP Datensätze, die vor 1 bis 2 Stunden erstellt wurden:

```
{
    "from": "-2h",
    "until": "-1h",
    "types": ["~http"]
}
```

Ausschlussintervalle

Ein Ausschlussintervall kann erstellt werden, um einen Zeitraum für die Unterdrückung eines Alarm.

Wenn Sie beispielsweise außerhalb der Geschäftszeiten oder am Wochenende nicht über Benachrichtigungen informiert werden möchten, kann ein Ausschlussintervall eine Regel erstellen, um die Alarm während dieses Zeitraums zu unterdrücken. Weitere Informationen finden Sie unter Warnmeldungen.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
GET /exclusioninterval	Ruft alle Ausschlussintervalle ab.
POST/Ausschlussintervalle	Erstellen Sie ein neues Ausschlussintervall.
LÖSCHEN /exclusionintervals/ {id}	Löscht ein bestimmtes Ausschlussintervall.
GET /exclusionintervals/ {id}	Rufen Sie ein bestimmtes Ausschlussintervall ab.
PATCH /exclusionintervals/ {id}	Wenden Sie Updates für ein bestimmtes Ausschlussintervall an.

Einzelheiten der Operation

`GET /exclusionintervals`

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "alert_apply_all": true,
    "author": "string",
    "description": "string",
```

```
    "end": 0,
    "id": 0,
    "interval_type": "string",
    "mod_time": 0,
    "name": "string",
    "start": 0,
    "trend_apply_all": true
}
```

`POST /exclusionintervals`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Legt die angegebenen Eigenschaftswerte für das neue Ausschlussintervall fest.

`name:` ***Schnur***

Der freundliche Name für das Ausschlussintervall.

`author:` ***Schnur***

(Optional) Der Name des Erstellers des Ausschlussintervalls.

`description:` ***Schnur***

(Optional) Eine optionale Beschreibung des Ausschlussintervalls.

`interval_type:` ***Schnur***

Das Zeitfenster, in dem das Ausschlussintervall ausgewertet wurde.

Die folgenden Werte sind gültig:

- `onetime`
- `weekly`
- `daily`

`start:` ***Zahl***

Der Beginn des Zeitbereichs für das Ausschlussintervall, ausgedrückt in Sekunden. Dieser Wert bezieht sich bei einmaligen Ausschlüssen auf die Epoche, bei täglichen Ausschlüssen auf Mitternacht und bei wöchentlichen Ausschlüssen auf Montag um Mitternacht.

`end:` ***Zahl***

Das Ende des Zeitbereichs für das Ausschlussintervall, ausgedrückt in Sekunden. Dieser Wert bezieht sich bei einmaligen Ausschlüssen auf die Epoche, bei täglichen Ausschlüssen auf Mitternacht und bei wöchentlichen Ausschlüssen auf Montag um Mitternacht.

`alert_apply_all:` ***Boolescher Wert***

Gibt an, ob dieses Ausschlussintervall auf alle Warnungen angewendet werden soll.

`trend_apply_all:` ***Boolescher Wert***

Gibt an, ob dieses Ausschlussintervall auf alle Trends angewendet werden soll.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "alert_apply_all": true,
    "author": "string",
    "description": "string",
    "end": 0,
    "interval_type": "string",
    "name": "string",
    "start": 0,
    "trend_apply_all": true
}
```

`GET /exclusionintervals/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung des Ausschlussintervalls.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "alert_apply_all": true,
    "author": "string",
    "description": "string",
    "end": 0,
    "id": 0,
    "interval_type": "string",
    "mod_time": 0,
    "name": "string",
    "start": 0,
    "trend_apply_all": true
}
```

`DELETE /exclusionintervals/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung des Ausschlussintervalls.

`PATCH /exclusionintervals/{id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Wendet die angegebenen Eigenschaftswertaktualisierungen auf das Ausschlussintervall an.
`id:` ***Zahl***
Die eindeutige Kennung für das Ausschlussintervall.

Ermittlungen

Mithilfe von Untersuchungen können Sie mehrere Funde in einer einzigen Zeitleiste und Karte hinzufügen und anzeigen. Weitere Informationen finden Sie unter Ermittlungen.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Bedienung	Beschreibung
GET /untersuchungen	Rufen Sie alle Untersuchungen ab.
POST /Ermittlungen	Erstelle eine Untersuchung.
POST /investigations/search	Suchen Sie nach Ermittlungen.
LÖSCHE /investigations/ {id}	Löschen Sie eine bestimmte Untersuchung.
GET /investigations/ {id}	Rufen Sie eine bestimmte Untersuchung ab.
PATCH /investigations/ {id}	Aktualisiere eine Untersuchung.

Einzelheiten der Operation

`GET /investigations/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für die Untersuchung.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "assessment": "string",
    "assignee": "string",
    "created_by": "string",
    "creation_time": 0,
    "description": "string",
    "detections": [
        "string"
    ],
    "end_time": 0,
    "id": 0,
    "investigation_types": [
        "string"
    ],
    "is_user_created": true,
    "last_interaction_by": "string",
    "name": "string",
    "notes": "string",
    "start_time": 0,
    "status": "string",
    "update_time": 0,
    "url": "string"
}
```

`GET /investigations`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "assessment": "string",
    "assignee": "string",
    "created_by": "string",
    "creation_time": 0,
    "description": "string",
    "detections": [
        "string"
    ],
    "end_time": 0,
    "id": 0,
    "investigation_types": [
        "string"
    ],
    "is_user_created": true,
    "last_interaction_by": "string",
    "name": "string",
    "notes": "string",
    "start_time": 0,
    "status": "string",
    "update_time": 0,
```

```
    "url": "string"
}
```

`POST /investigations/search`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Parameter für die Untersuchung.

`update_time:` ***Zahl***

Gibt Untersuchungen zurück, die nach dem angegebenen Datum aktualisiert wurden, ausgedrückt in Millisekunden seit der Epoche.

`creation_time:` ***Zahl***

Gibt Untersuchungen zurück, die nach dem angegebenen Datum erstellt wurden, ausgedrückt in Millisekunden seit der Epoche.

`is_user_created:` ***Boolesch***

(Optional) Gibt nur Untersuchungen zurück, die manuell von einem Benutzer erstellt wurden.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "creation_time": 0,
    "is_user_created": true,
    "update_time": 0
}
```

`PATCH /investigations/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die ID der Untersuchung, die aktualisiert werden soll.

`body:` ***Objekt***

Die zu aktualisierenden Untersuchungsfelder.

`name:` ***Schnur***

(Optional) Der Name der Untersuchung.

`status:` ***Schnur***

(Optional) Der Status der Untersuchung.

Die folgenden Werte sind gültig:

- `open`
- `in_progress`
- `closed`

`notes:` ***Schnur***

(Fakultativ) Optionale Hinweise zur Untersuchung.

`event_ids:` ***Reihe von Zahlen***

(Optional) Die Liste der IDs für Erkennungen in der Untersuchung. Wenn Sie dieses Feld angeben, ersetzt die neue Liste von IDs die bestehende Liste.

`assignee:` ***Schnur***

(Optional) Der Benutzername des mit der Untersuchung beauftragten Mitarbeiters.

`assessment:` ***Schnur***

(Fakultativ) Die Bewertung der Untersuchung.

Die folgenden Werte sind gültig:

- `malicious_true_positive`
- `benign_true_positive`
- `false_positive`
- `undecided`

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assessment": "string",
    "assignee": "string",
    "event_ids": [],
    "name": "string",
    "notes": "string",
    "status": "string"
}
```

`POST /investigations`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Bereiche der neuen Untersuchung.

`name:` ***Schnur***

Der Name der Untersuchung.

`status:` ***Schnur***

(Optional) Der Status der Untersuchung.

Die folgenden Werte sind gültig:

- `open`
- `in_progress`
- `closed`

`notes:` ***Schnur***

(Fakultativ) Optionale Hinweise zur Untersuchung.

`event_ids:` ***Reihe von Zahlen***

(Optional) Die Liste der IDs für Erkennungen in der Untersuchung.

`assignee:` ***Schnur***

(Optional) Der Benutzername des mit der Untersuchung beauftragten Mitarbeiters.

`assessment:` ***Schnur***

(Fakultativ) Die Bewertung der Untersuchung.

Die folgenden Werte sind gültig:

- `malicious_true_positive`
- `benign_true_positive`
- `false_positive`
- `undecided`

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assessment": "string",
    "assignee": "string",
    "event_ids": [],
    "name": "string",
```

```
    "notes": "string",
    "status": "string"
}
```

`DELETE /investigations/{id}`

Geben Sie die folgenden Parameter an.

`id`: ***Zahl***
Die ID der zu löschenden Untersuchung.

Metriken

Zu jedem Objekt, das vom ExtraHop-System identifiziert wird, werden Metrikinformationen gesammelt.

Beachten Sie, dass Metriken über die POST-Methode abgerufen werden, die eine Abfrage erstellt, um die angeforderten Informationen über die API zu sammeln. Weitere Informationen finden Sie unter Extrahieren Sie Metriken über die REST-API.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Bedienung	Beschreibung
POST /Metriken	Ruft Metriken für jedes angegebene Objekt ab.
GET /metrics/next/ {xid}	Wenn Sie Metriken von einem anfordern Konsole mit dem `POST /metrics`, `POST /metrics/total`, oder `POST /metrics/totalbyobject` Operation, und Sie geben Objekte an, die von mehreren Sensoren beobachtet wurden, die Antwort enthält `xid` Feld statt Metrik Daten. Sie können Metrikdaten abrufen, indem Sie Folgendes angeben `xid` Feld in der `GET /metrics/next/{xid}` Operation, die Metriken von einem der an die Konsole angeschlossenen Sensoren zurückgibt. Wiederhole das `GET /metrics/next/{xid}` Betrieb, um Metriken von zusätzlichen Sensoren zurückzugeben. Nachdem alle Metriken abgerufen wurden, gibt der Vorgang Null zurück. Wenn vom Sensor noch keine Metriken verfügbar sind, wird die Zeichenfolge `again` wird zurückgegeben. Warten Sie ein paar Sekunden und versuchen Sie es dann erneut. **Hinweis:** Die Antwort könnte eine enthalten `xid` Feld, auch wenn Sie nur Messwerte für eine einzelne Gerätegruppe angefordert haben, da Gerätegruppen Geräte von mehreren Sensoren enthalten können.
POST /Metriken/insgesamt	Ruft kombinierte Metriksummen für alle angegebenen Objekte ab.
POST /metrics/totalbyobject	Ruft Metriksummen für jedes angegebene Objekt ab.

Der folgende Anforderungstext ruft beispielsweise HTTP-Antworten ab, die zwei Geräte in den letzten 30 Minuten gesendet haben.

```
{
  "cycle": "auto",
  "from": -1800000,
  "metric_category": "http_server",
  "metric_specs": [
    {
      "name": "rsp"
    }
  ],
  "object_ids": [
    180, 177
  ],
  "object_type": "device",
  "until": 0
}
```

Für die `POST /metrics` Operation, der vorherige Beispielanforderungstext gibt die Anzahl der HTTP-Antworten zurück, die in jedem Zeitintervall aufgetreten sind. Sie sind mit der Uhrzeit jedes Ereignis und der ID des Gerät, das die Antworten gesendet hat, beschriftet, ähnlich der folgenden Beispielantwort:

```
{
  "cycle": "30sec",
  "node_id": 0,
  "clock": 1709659320000,
  "from": 1709657520000,
  "until": 1709659320000,
  "stats": [
    {
      "oid": 177,
      "time": 1709657520000,
      "duration": 30000,
      "values": [
        4
      ]
    },
    {
      "oid": 177,
      "time": 1709657550000,
      "duration": 30000,
      "values": [
        4
      ]
    },
    {
      "oid": 180,
      "time": 1709657520000,
      "duration": 30000,
      "values": [
        4
      ]
    },
    {
      "oid": 180,
      "time": 1709657550000,
      "duration": 30000,
      "values": [
        4
      ]
    }
```

```
  ]
}
```

Für die `POST /metrics/totalbyobject` Operation, derselbe vorherige Beispielanforderungstext ruft die Gesamtsumme für jedes Gerät über den gesamten Zeitraum ab, ähnlich der folgenden Beispielantwort:

```
{
  "cycle": "30sec",
  "node_id": 0,
  "clock": 1709659620000,
  "from": 1709657820000,
  "until": 1709659620000,
  "stats": [
    {
      "oid": 180,
      "time": 1709659620000,
      "duration": 1830000,
      "values": [
        8
      ]
    },
    {
      "oid": 177,
      "time": 1709659620000,
      "duration": 1830000,
      "values": [
        8
      ]
    }
  ]
}
```

Für die `POST /metrics/total` Operation, derselbe vorherige Beispiel-Anforderungstext ruft die Gesamtsumme beider Geräte über den gesamten Zeitraum ab, ähnlich der folgenden Beispielantwort:

```
{
  "cycle": "30sec",
  "node_id": 0,
  "clock": 1709659830000,
  "from": 1709658030000,
  "until": 1709659830000,
  "stats": [
    {
      "oid": -1,
      "time": 1709659830000,
      "duration": 1830000,
      "values": [
        16
      ]
    }
  ]
}
```

Beachten Sie, dass das Verhalten des `/metrics/total` und `/metrics/totalbyobject` Endpunkte hängen vom Typ der Metrik ab. Für Zählmetriken ist der `values` Das Feld enthält eine Gesamtsumme der Werte über das angegebene Zeitintervall, wie im obigen Beispiel gezeigt. Für Datensatzmetriken ist jedoch `values` Das Feld enthält eine Liste von Werten und die Häufigkeit, mit der diese Werte auftauchten. Zum Beispiel eine Abfrage nach Serververarbeitungszeiten mit dem `POST /metrics/total` operation gibt eine Antwort zurück, die dem folgenden Beispiel ähnelt:

```
{
```

```
  "cycle": "30sec",
  "node_id": 0,
  "clock": 1494541440000,
  "from": 1494539640000,
  "until": 1494541440000,
  "stats": [
    {
      "oid": -1,
      "time": 1494541380000,
      "duration": 1800000,
      "values": [
        [
          {
            "value": 2.271,
            "freq": 5
          },
          {
            "value": 48.903,
            "freq": 1
          }
        ]
      ]
    }
  ]
}
```


Wenn im angegebenen Zeitraum mehr als 1.000 unterschiedliche Datensatzwerte vorliegen, werden ähnliche Werte konsolidiert, um die Antwortvariablen auf 1.000 Werte zu reduzieren. Wenn es beispielsweise weniger als 1.000 Werte gibt, kann die Antwort die folgenden Einträge enthalten:


```
{
  "value": 2.571,
  "freq": 4
},
{
  "value": 2.912,
  "freq": 2
}
```


Wenn die Antwort jedoch mehr als 1.000 Werte enthält, können diese Einträge zu dem folgenden Eintrag konsolidiert werden:


```
{
  "value": 2.571,
  "freq": 6
}
```


Wenn der `calc_type` Ein Feld ist angegeben und die Antwort enthält mehr als 1.000 Werte. Das Perzentil oder der Mittelwert wird anhand des konsolidierten Datensatzes berechnet.

Einzelheiten der Operation

`POST /metrics`

Geben Sie die folgenden Parameter an.

`body`: ***Objekt***

Die Beschreibung der Metrikanforderung.

`from:` ***Zahl***

Der Anfangszeitstempel für die Anfrage. Gibt nur Metriken zurück, die nach dieser Zeit erfasst wurden. Die Zeit wird in Millisekunden seit der Epoche ausgedrückt. 0 gibt den Zeitpunkt der Anfrage an. Ein negativer Wert wird relativ zur aktuellen Uhrzeit ausgewertet. Die Standardeinheit für einen negativen Wert ist Millisekunden, aber andere Einheiten können mit einem Einheitensuffix angegeben werden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`until:` ***Zahl***

Der Endzeitstempel für die Anfrage. Gibt nur Metriken zurück, die vor diesem Zeitpunkt erfasst wurden. Folgt den gleichen Zeitwertrichtlinien wie der From Parameter.

`cycle:` ***Schnur***

Der Aggregationszeitraum für Metriken.

Die folgenden Werte sind gültig:

- `auto`
- `1sec`
- `30sec`
- `5min`
- `1hr`
- `24hr`

`object_type:` ***Schnur***

Gibt den Objekttyp der eindeutigen Bezeichner an, die in der Eigenschaft object_ids angegeben sind.

Die folgenden Werte sind gültig:

- `network`
- `device`
- `application`
- `vlan`
- `device_group`
- `system`

`object_ids:` ***Reihe von Zahlen***

Die Liste der numerischen Werte, die eindeutige Identifikatoren darstellen. Eindeutige Identifikatoren können über die Ressourcen /networks, /devices, /applications, /vlans, /devicegroups, /activitygroups und /appliances abgerufen werden. Geben Sie für Systemintegritätsmetriken die ID des Sensor oder der Konsole an und setzen Sie den Parameter object_type auf „system“.

`metric_category:` ***Schnur***

Die Gruppe von Metriken, die im Metrikkatalog durchsucht werden können.

`metric_specs:` ***Reihe von Objekten***

Ein Array von Metrik Spezifikationsobjekten.

`name:` ***Schnur***

Der Feldname für die Metrik. Wenn Sie im Metrikkatalog nach einer metric_category filtern, ist jedes Ergebnis ein potenzieller metric_spec-Name. Wenn ein Ergebnis aus dem Katalog ausgewählt wird, ist der Feldwert „Metrik“ eine gültige Option für dieses Feld.

`key1:` ***Schnur***

(Optional) Filtern Sie Detailmetriken. Detailmetriken unterteilen Daten anhand von Schlüsseln, bei denen es sich um Zeichenketten oder IP-Adressen handelt. Beispielsweise akzeptiert die Metrik „HTTP Requests by Method“ den key1-Wert

„GET“. Schlüssel können auch reguläre Ausdrücke sein, die durch Schrägstriche („/GET/“) getrennt sind.

`key2:` ***Schnur***

(Optional) Aktivieren Sie zusätzliche Filterung für Detailmetriken.

`calc_type:` ***Schnur***

(Optional) Die Art der auszuführenden Berechnung.

Die folgenden Werte sind gültig:

- `mean`
- `percentiles`

`percentiles:` ***Reihe von Zahlen***

(Optional) Die in aufsteigender Reihenfolge sortierte Liste der Perzentile, die zurückgegeben werden sollen. Dieser Parameter ist nur erforderlich, wenn der Parameter calc_type auf „Perzentile“ gesetzt ist. Wenn der Parameter calc_type auf mean gesetzt ist, kann die Percentile-Eigenschaft nicht festgelegt werden.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "cycle": "string",
    "from": 0,
    "metric_category": "string",
    "metric_specs": {
        "name": "string",
        "key1": "string",
        "key2": "string",
        "calc_type": "string",
        "percentiles": []
    },
    "object_ids": [],
    "object_type": "string",
    "until": 0
}
```

`POST /metrics/total`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Beschreibung der Metrikanforderung.

`from:` ***Zahl***

Der Anfangszeitstempel für die Anfrage. Gibt nur Metriken zurück, die nach dieser Zeit erfasst wurden. Die Zeit wird in Millisekunden seit der Epoche ausgedrückt. 0 gibt den Zeitpunkt der Anfrage an. Ein negativer Wert wird relativ zur aktuellen Uhrzeit ausgewertet. Die Standardeinheit für einen negativen Wert ist Millisekunden, aber andere Einheiten können mit einem Einheitensuffix angegeben werden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`until:` ***Zahl***

Der Endzeitstempel für die Anfrage. Gibt nur Metriken zurück, die vor diesem Zeitpunkt erfasst wurden. Folgt den gleichen Zeitwertrichtlinien wie der From Parameter.

`cycle:` ***Schnur***

Der Aggregationszeitraum für Metriken.

Die folgenden Werte sind gültig:

- `auto`

- `1sec`
- `30sec`
- `5min`
- `1hr`
- `24hr`

`object_type:` ***Schnur***

Gibt den Objekttyp der eindeutigen Bezeichner an, die in der Eigenschaft object_ids angegeben sind.

Die folgenden Werte sind gültig:

- `network`
- `device`
- `application`
- `vlan`
- `device_group`
- `system`

`object_ids:` ***Reihe von Zahlen***

Die Liste der numerischen Werte, die eindeutige Identifikatoren darstellen. Eindeutige Identifikatoren können über die Ressourcen /networks, /devices, /applications, /vlans, /devicegroups, /activitygroups und /appliances abgerufen werden. Geben Sie für Systemintegritätsmetriken die ID des Sensor oder der Konsole an und setzen Sie den Parameter object_type auf „system“.

`metric_category:` ***Schnur***

Die Gruppe von Metriken, die im Metrikkatalog durchsucht werden können.

`metric_specs:` ***Reihe von Objekten***

Ein Array von Metrik Spezifikationsobjekten.

`name:` ***Schnur***

Der Feldname für die Metrik. Wenn Sie im Metrikkatalog nach einer metric_category filtern, ist jedes Ergebnis ein potenzieller metric_spec-Name. Wenn ein Ergebnis aus dem Katalog ausgewählt wird, ist der Feldwert „Metrik“ eine gültige Option für dieses Feld.

`key1:` ***Schnur***

(Optional) Filtern Sie Detailmetriken. Detailmetriken unterteilen Daten anhand von Schlüsseln, bei denen es sich um Zeichenketten oder IP-Adressen handelt. Beispielsweise akzeptiert die Metrik „HTTP Requests by Method“ den key1-Wert „GET“. Schlüssel können auch reguläre Ausdrücke sein, die durch Schrägstriche („/GET/“) getrennt sind.

`key2:` ***Schnur***

(Optional) Aktivieren Sie zusätzliche Filterung für Detailmetriken.

`calc_type:` ***Schnur***

(Optional) Die Art der auszuführenden Berechnung.

Die folgenden Werte sind gültig:

- `mean`
- `percentiles`

`percentiles:` ***Reihe von Zahlen***

(Optional) Die in aufsteigender Reihenfolge sortierte Liste der Perzentile, die zurückgegeben werden sollen. Dieser Parameter ist nur erforderlich, wenn der Parameter calc_type auf „Perzentile“ gesetzt ist. Wenn der Parameter calc_type auf mean gesetzt ist, kann die Percentile-Eigenschaft nicht festgelegt werden.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "cycle": "string",
    "from": 0,
    "metric_category": "string",
    "metric_specs": {
        "name": "string",
        "key1": "string",
        "key2": "string",
        "calc_type": "string",
        "percentiles": []
    },
    "object_ids": [],
    "object_type": "string",
    "until": 0
}
```

`POST /metrics/totalbyobject`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Beschreibung der Metrikanforderung.

`from:` ***Zahl***

Der Anfangszeitstempel für die Anfrage. Gibt nur Metriken zurück, die nach dieser Zeit erfasst wurden. Die Zeit wird in Millisekunden seit der Epoche ausgedrückt. 0 gibt den Zeitpunkt der Anfrage an. Ein negativer Wert wird relativ zur aktuellen Uhrzeit ausgewertet. Die Standardeinheit für einen negativen Wert ist Millisekunden, aber andere Einheiten können mit einem Einheitensuffix angegeben werden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`until:` ***Zahl***

Der Endzeitstempel für die Anfrage. Gibt nur Metriken zurück, die vor diesem Zeitpunkt erfasst wurden. Folgt den gleichen Zeitwertrichtlinien wie der From Parameter.

`cycle:` ***Schnur***

Der Aggregationszeitraum für Metriken.

Die folgenden Werte sind gültig:

- `auto`
- `1sec`
- `30sec`
- `5min`
- `1hr`
- `24hr`

`object_type:` ***Schnur***

Gibt den Objekttyp der eindeutigen Bezeichner an, die in der Eigenschaft object_ids angegeben sind.

Die folgenden Werte sind gültig:

- `network`
- `device`
- `application`
- `vlan`
- `device_group`

- `system`

`object_ids:` ***Reihe von Zahlen***

Die Liste der numerischen Werte, die eindeutige Identifikatoren darstellen. Eindeutige Identifikatoren können über die Ressourcen /networks, /devices, /applications, /vlans, /devicegroups, /activitygroups und /appliances abgerufen werden. Geben Sie für Systemintegritätsmetriken die ID des Sensor oder der Konsole an und setzen Sie den Parameter object_type auf „system“.

`metric_category:` ***Schnur***

Die Gruppe von Metriken, die im Metrikkatalog durchsucht werden können.

`metric_specs:` ***Reihe von Objekten***

Ein Array von Metrik Spezifikationsobjekten.

`name:` ***Schnur***

Der Feldname für die Metrik. Wenn Sie im Metrikkatalog nach einer metric_category filtern, ist jedes Ergebnis ein potenzieller metric_spec-Name. Wenn ein Ergebnis aus dem Katalog ausgewählt wird, ist der Feldwert „Metrik“ eine gültige Option für dieses Feld.

`key1:` ***Schnur***

(Optional) Filtern Sie Detailmetriken. Detailmetriken unterteilen Daten anhand von Schlüsseln, bei denen es sich um Zeichenketten oder IP-Adressen handelt. Beispielsweise akzeptiert die Metrik „HTTP Requests by Method“ den key1-Wert „GET“. Schlüssel können auch reguläre Ausdrücke sein, die durch Schrägstriche („/GET/“) getrennt sind.

`key2:` ***Schnur***

(Optional) Aktivieren Sie zusätzliche Filterung für Detailmetriken.

`calc_type:` ***Schnur***

(Optional) Die Art der auszuführenden Berechnung.

Die folgenden Werte sind gültig:

- `mean`
- `percentiles`

`percentiles:` ***Reihe von Zahlen***

(Optional) Die in aufsteigender Reihenfolge sortierte Liste der Perzentile, die zurückgegeben werden sollen. Dieser Parameter ist nur erforderlich, wenn der Parameter calc_type auf „Perzentile“ gesetzt ist. Wenn der Parameter calc_type auf mean gesetzt ist, kann die Percentile-Eigenschaft nicht festgelegt werden.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "cycle": "string",
    "from": 0,
    "metric_category": "string",
    "metric_specs": {
        "name": "string",
        "key1": "string",
        "key2": "string",
        "calc_type": "string",
        "percentiles": []
    },
    "object_ids": [],
    "object_type": "string",
    "until": 0
}
```

```
GET /metrics/next/{xid}
```

Geben Sie die folgenden Parameter an.

`xid:` ***Zahl***
Der eindeutige Bezeichner, der von einer Metrikabfrage zurückgegeben wird.

Unterstützte Zeiteinheiten

Für die meisten Parameter ist die Standardeinheit für die Zeitmessung Millisekunden. Die folgenden Parameter geben jedoch alternative Zeiteinheiten wie Minuten und Stunden zurück oder akzeptieren diese:

- Gerät
  - aktive_von
  - aktiv_bis
- Gerätegruppe
  - aktive_von
  - aktiv_bis
- Metriken
  - von
  - bis
- Protokoll aufzeichnen
  - von
  - bis
  - kontext_ttl

Die folgende Tabelle zeigt die unterstützten Zeiteinheiten:

Zeiteinheit	Einheitensuffix
Jahr	`y`
Monat	`M`
Woche	`w`
Tag	`d`
Stunde	`h`
Minute	`m`
Zweiter	`s`
Millisekunde	`ms`

Um für einen Parameter eine andere Zeiteinheit als Millisekunden anzugeben, hängen Sie das Einheitensuffix an den Wert an. Um beispielsweise Geräte anzufordern, die in den letzten 30 Minuten aktiv waren, geben Sie den folgenden Parameterwert an:

```
GET /api/v1/devices?active_from=-30m
```

Das folgende Beispiel spezifiziert eine Suche nach HTTP Datensätze, die vor 1 bis 2 Stunden erstellt wurden:

```
{
    "from": "-2h",
    "until": "-1h",
```

```
    "types": ["~http"]
}
```

Eingabe der Netzwerklokalität

Sie können eine Liste verwalten, die die Netzwerklokalität von IP-Adressen angibt.

Sie können beispielsweise einen Eintrag in der Netzwerklokalisierungsliste erstellen, der angibt, dass eine IP-Adresse oder ein CIDR-Block intern oder extern ist.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
GET /networkalities	Ruft alle Netzwerk-Lokalitätseinträge ab.
POST /Netzwerklocations	Erstellen Sie einen Eintrag für die Netzwerklokalität.
/networklocalities/ {id} LÖSCHEN	Löscht einen Eintrag für die Netzwerklokalität.
GET /networklocalities/ {id}	Ruft einen bestimmten Eintrag für die Netzwerklokalität ab.
PATCH /networklocalities/ {id}	Wenden Sie Aktualisierungen auf einen bestimmten Netzwerklokalitätseintrag an.

Einzelheiten der Operation

`GET /networklocalities`

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "description": "string",
    "external": true,
    "id": 0,
    "mod_time": 0,
    "name": "string",
    "network": "string",
    "networks": []
}
```

`POST /networklocalities`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Wendet die angegebenen Eigenschaftswerte auf den neuen Eintrag für die Netzwerklokalität an.

`name:` ***Schnur***

(Optional) Der Name der Netzwerklokalität. Wenn dieses Feld nicht angegeben ist, wird die Netzwerklokalität im folgenden Format benannt: „Locality_ID“, wobei ID die eindeutige Kennung der Netzwerklokalität ist.

`network:` ***Schnur***

(Optional) Veraltet. Geben Sie CIDR-Blöcke oder IP-Adressen im Feld Netzwerke an.

`networks:` ***Reihe von Zeichenketten***

(Optional) Eine Reihe von CIDR-Blöcken oder IP-Adressen, die die Netzwerklokalität definieren.

`external:` ***Boolescher Wert***

Gibt an, ob das Netzwerk intern oder extern ist.

`description:` ***Schnur***

(Optional) Eine optionale Beschreibung des Eintrags zur Netzwerklokalität.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "description": "string",
    "external": true,
    "name": "string",
    "network": "string",
    "networks": []
}
```

`GET /networklocalities/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für den Eintrag zur Netzwerklokalität.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "description": "string",
    "external": true,
    "id": 0,
    "mod_time": 0,
    "name": "string",
    "network": "string",
    "networks": []
}
```

`DELETE /networklocalities/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für den Eintrag zur Netzwerklokalität.

`PATCH /networklocalities/{id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Wendet die angegebenen Eigenschaftswertaktualisierungen auf den Eintrag für die Netzwerklokalität an.

`network:` ***Schnur***

(Optional) Veraltet. Geben Sie CIDR-Blöcke oder IP-Adressen im Feld Netzwerke an.

`networks:` ***Reihe von Zeichenketten***

(Optional) Eine Reihe von CIDR-Blöcken oder IP-Adressen, die die Netzwerklokalität definieren.

`name:` ***Schnur***

(Optional) Der Name der Netzwerklokalität.

`external:` ***Boolescher Wert***

(Optional) Gibt an, ob das Netzwerk intern oder extern ist.

`description:` ***Schnur***

(Optional) Eine optionale Beschreibung des Eintrags zur Netzwerklokalität.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "description": "string",
    "external": true,
    "name": "string",
    "network": "string",
    "networks": []
}
```

`id:` ***Zahl***

Die eindeutige Kennung für den Eintrag zur Netzwerklokalität.

Netzwerk

Netzwerke sind mit der Netzwerkschnittstellenkarte korreliert, die Eingaben von allen vom ExtraHop-System identifizierten Objekten empfängt.

Auf einem Konsole, jeder angeschlossene Sensor wird als Netzwerkaufnahme identifiziert. Weitere Informationen finden Sie unter Netzwerke.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
GET /netzwerke	Ruft alle Netzwerke ab.
GET /networks/ {id}	Ruft ein bestimmtes Netzwerk anhand der ID ab.
PATCH /Netzwerke/ {id}	Aktualisieren Sie ein bestimmtes Netzwerk anhand der ID.
GET /networks/ {id} /alerts	Alles abrufen Warnungen die einem bestimmten Netzwerk zugewiesen sind.
POST /networks/ {id} /alerts	Weisen Sie Alerts einem bestimmten Netzwerk zu und heben Sie die Zuweisung auf.
LÖSCHEN Sie /networks/ {id} /alerts/ {child-id}	Heben Sie die Zuweisung einer Alarm zu einem bestimmten Netzwerk auf.
POST /networks/ {id} /alerts/ {child-id}	Weisen Sie einem bestimmten Netzwerk eine Alarm zu.
GET /networks/ {id} /vlans	Ruft alle VLANS ab, die einem bestimmten Netzwerk zugewiesen sind.

Einzelheiten der Operation

`GET /networks`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "appliance_uuid": "string",
    "description": "string",
    "id": 0,
    "idle": true,
    "mod_time": 0,
    "name": "string",
    "node_id": 0
}
```

`PATCH /networks/{id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Eigenschaftswertaktualisierungen, die auf das Netzwerk angewendet werden sollen.

`id:` ***Zahl***
Eindeutige Kennung des Netzwerk.

`GET /networks/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Eindeutige Kennung des Netzwerk.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "appliance_uuid": "string",
    "description": "string",
    "id": 0,
    "idle": true,
    "mod_time": 0,
    "name": "string",
    "node_id": 0
}
```

`GET /networks/{id}/alerts`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Eindeutige Kennung des Netzwerk.

`direct_assignments_only:` ***Boolescher Wert***
(Optional) Beschränken Sie die Ergebnisse auf Warnmeldungen, die dem Netzwerk direkt zugewiesen sind.

`POST /networks/{id}/alerts`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Listen von Alert-IDs, die zugewiesen und/oder aufgehoben werden sollen.

`assign:` ***Reihe von Zahlen***

IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***

IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***

Eindeutige Kennung des Netzwerk.

`POST /networks/{id}/alerts/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Eindeutige Kennung der Alarm.

`id:` ***Zahl***

Eindeutige Kennung des Netzwerk.

`DELETE /networks/{id}/alerts/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Eindeutige Kennung der Alarm.

`id:` ***Zahl***

Eindeutige Kennung des Netzwerk.

`GET /networks/{id}/vlans`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Eindeutige Kennung des Netzwerk.

Beobachtungen

Eine Beobachtung verknüpft die IP-Adresse eines Gerät auf dem ExtraHop-System mit einer IP-Adresse außerhalb Ihres Netzwerk. Sie können beispielsweise die Aktivität eines VPN-Benutzers verfolgen, indem Sie die IP-Adresse des VPN-Clients in Ihrem internen Netzwerk mit der externen IP-Adresse verknüpfen, die dem Benutzer im öffentlichen Internet zugewiesen wurde.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
BEITRAG /observations/associatedipaddrs	Fügen Sie eine Beobachtung hinzu, um eine Zuordnung zwischen Geräte-IP-Adressen herzustellen.

Einzelheiten der Operation

`POST /observations/associatedipaddrs`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Beobachtungsparameter.

`observations:` ***Reihe von Objekten***

Eine Reihe von Beobachtungen.

`ipaddr:` ***Schnur***

Die vom Sensor oder der Konsole beobachtete Geräte-IP-Adresse.

`associated_ipaddr:` ***Schnur***

Die zugehörige IP-Adresse.

`timestamp:` ***Zahl***

Die Zeit, in der die Beobachtung von der Quelle erstellt wurde, ausgedrückt in Millisekunden seit der Epoche.

`source:` ***Schnur***

Die Quelle der Beobachtungen.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "observations": {
        "ipaddr": "string",
        "associated_ipaddr": "string",
        "timestamp": 0
    },
    "source": "string"
}
```

Paketsuche

Sie können nach Paketen suchen und diese herunterladen, die auf dem ExtraHop-System gespeichert sind. Das heruntergeladene Pakete können dann mit einem Drittanbieter-Tool wie Wireshark analysiert werden.

Weitere Informationen zu Paketen finden Sie unter Pakete.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Bedienung	Beschreibung
GET /Pakete/Suche	Suchen Sie nach Paketen, indem Sie Parameter in einer URL angeben.
POST /Pakete/Suche	Suchen Sie nach Paketen, indem Sie Parameter in einer JSON-Zeichenfolge angeben.

Einzelheiten der Operation

`GET /packets/search`

Geben Sie die folgenden Parameter an.

`output:` ***Schnur***

(Optional) Das Ausgabeformat. * `pcap` - Eine PCAP-Datei, die Pakete enthält. * `keylog_txt` - Eine Keylog-Textdatei, die Geheimnisse für die Entschlüsselung enthält. * `pcapng` - Eine PCAPNG-Datei, die sowohl Pakete als auch Geheimnisse für die Entschlüsselung enthalten kann. * `zip` - Eine ZIP-Datei, die sowohl eine PCAP- als auch eine Keylog-Textdatei enthält. * `extract` - Eine ZIP-Datei, die Dateien enthält, die aus Paketen extrahiert wurden, die der Abfrage entsprachen. Diese Option ist nur gültig, wenn Sie vollen Zugriff auf das NDR-Modul haben.

Die folgenden Werte sind gültig:

- `pcap`
- `keylog_txt`
- `pcapng`
- `zip`
- `extract`

`include_secrets:` ***Boolesch***

(Optional) Gibt an, ob Geheimnisse in die PCAPNG-Datei aufgenommen werden sollen. Diese Option ist nur gültig, wenn `output` auf `pcapng` gesetzt ist.

`decrypt_files:` ***Boolesch***

(Optional) Gibt an, ob entpackte Dateien mit gespeicherten Geheimnissen entschlüsselt werden sollen. Diese Option ist nur gültig, wenn der `output`-Parameter `extract` ist.

`limit_bytes:` ***Schnur***

(Optional) Die ungefähre maximale Anzahl von Byte, die zurückgegeben werden sollen. Nachdem das ExtraHop-System Pakete gefunden hat, die der in den Suchkriterien angegebenen Größe entsprechen, stoppt das System die Suche nach weiteren Paketen. Da das System jedoch mehrere Pakete gleichzeitig analysiert, kann die Gesamtgröße der zurückgegebenen Pakete größer als die angegebene Größe sein. Die Standardeinheit ist Byte, aber Sie können auch andere Einheiten mit einem Einheitensuffix angeben. Der Standardwert ist „100 MB“. **Hinweis**: Wenn die Ausgabe „extract“ lautet, gibt es einen Maximalwert für dieses Feld. Das Standardmaximum ist „100 MB“, aber das Maximum kann in der laufenden Konfiguration geändert werden. Wenn die Ausgabe nicht „Extrahieren“ ist, gibt es keinen Maximalwert.

`limit_search_duration:` ***Schnur***

(Optional) Die ungefähre maximale Zeit für die Durchführung der Paketsuche. Nach Ablauf der angegebenen Zeit hört das ExtraHop-System auf, nach weiteren Paketen zu suchen. Das System verlängert jedoch die angegebene Zeit, um die Analyse von Paketen abzuschließen, die vor Ablauf der Zeit durchsucht wurden, und das System analysiert mehrere Pakete gleichzeitig. Daher kann die Suche länger als die angegebene Zeit dauern. Die Standardeinheit ist Millisekunden, aber andere Einheiten können mit einem Einheitensuffix angegeben werden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe. Der Standardwert ist „5m“. **Hinweis**: Wenn die Ausgabe „extract“ lautet, gibt es einen Maximalwert für dieses Feld. Das Standardmaximum ist „5 m“, aber das Maximum kann in der laufenden Konfiguration geändert werden. Wenn die Ausgabe nicht „Extrahieren“ ist, gibt es keinen Maximalwert.

`always_return_body:` ***Boolesch***

(Optional) Gibt das Verhalten an, wenn die Abfrage keinen Paketen entspricht oder wenn die Pakete, auf die die Abfrage zutrifft, keine Dateien enthalten. Wenn der Wert wahr ist, gibt das System eine leere Datei und einen 200-Statuscode zurück. Wenn der Wert falsch ist, gibt das System einen 204-Statuscode zurück, aber keine Datei.

`from:` ***Schnur***

Der Anfangszeitstempel des Zeitbereichs, den die Suche umfassen wird, ausgedrückt in Millisekunden seit der Epoche. Ein negativer Wert gibt an, dass die Suche mit Paketen beginnt, die zu einem Zeitpunkt in der Vergangenheit erfasst wurden. Geben Sie beispielsweise -10m an, um die Suche mit Paketen zu beginnen, die 10 Minuten vor dem Zeitpunkt der Anfrage erfasst wurden. Negative Werte können mit einer anderen Zeiteinheit als Millisekunden angegeben werden, z. B.

Sekunden oder Stunden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`until:` ***Schnur***

(Optional) Der Endzeitstempel des Zeitbereichs, den die Suche einschließen wird, ausgedrückt in Millisekunden seit der Epoche. Ein Wert 0 gibt an, dass die Suche mit Paketen endet, die zum Zeitpunkt der Suche erfasst wurden. Ein negativer Wert gibt an, dass die Suche mit Paketen endet, die zu einem Zeitpunkt in der Vergangenheit erfasst wurden. Geben Sie beispielsweise -5m an, um die Suche mit Paketen zu beenden, die 5 Minuten vor dem Zeitpunkt der Anfrage erfasst wurden. Negative Werte können mit einer anderen Zeiteinheit als Millisekunden angegeben werden, z. B. Sekunden oder Stunden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`bpf:` ***Schnur***

(Optional) Die Berkeley Paket Filter (BPF) -Syntax für die Paketsuche. Weitere Informationen zur BPF-Syntax finden Sie in der REST-API-Leitfaden.

`ip1:` ***Schnur***

(Optional) Gibt Pakete zurück, die an die angegebene IP-Adresse gesendet oder von dieser empfangen wurden.

`port1:` ***Schnur***

(Optional) Gibt Pakete zurück, die vom angegebenen Port gesendet oder empfangen wurden.

`ip2:` ***Schnur***

(Optional) Gibt Pakete zurück, die an die angegebene IP-Adresse gesendet oder von dieser empfangen wurden.

`port2:` ***Schnur***

(Optional) Gibt Pakete zurück, die vom angegebenen Port gesendet oder empfangen wurden.

`POST /packets/search`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Parameter der Paketsuche.

`output:` ***Schnur***

(Optional) Das Ausgabeformat.

Die folgenden Werte sind gültig:

- `pcap`
- `keylog_txt`
- `pcapng`
- `zip`
- `extract`

`include_secrets:` ***Boolesch***

(Optional) Ob TLS-Geheimnisse zusammen mit Paketdaten in .pcapng-Dateien aufgenommen werden sollen oder nicht. Nur gültig, wenn „output“ „pcapng“ ist.

`decrypt_files:` ***Boolesch***

(Optional) Gibt an, ob entpackte Dateien mit gespeicherten Geheimnissen entschlüsselt werden sollen. Diese Option ist nur gültig, wenn der `output`-Parameter `extract` ist.

`limit_bytes:` ***Schnur***

(Optional) Die ungefähre maximale Anzahl von Byte, die zurückgegeben werden sollen. Nachdem das ExtraHop-System Pakete gefunden hat, die der in den Suchkriterien angegebenen Größe entsprechen, stoppt das System die Suche nach weiteren Paketen. Da das System jedoch mehrere Pakete gleichzeitig analysiert, kann die Gesamtgröße der

zurückgegebenen Pakete größer als die angegebene Größe sein. Die Standardeinheit ist Byte, aber Sie können auch andere Einheiten mit einem Einheitensuffix angeben. Der Standardwert ist „100 MB“. **Hinweis**: Wenn die Ausgabe „extract“ lautet, gibt es einen Maximalwert für dieses Feld. Das Standardmaximum ist „100 MB“, aber das Maximum kann in der laufenden Konfiguration geändert werden. Wenn die Ausgabe nicht „Extrahieren“ ist, gibt es keinen Maximalwert.

`limit_search_duration:` ***Schnur***

(Optional) Die ungefähre maximale Zeit für die Durchführung der Paketsuche. Nach Ablauf der angegebenen Zeit hört das ExtraHop-System auf, nach weiteren Paketen zu suchen. Das System verlängert jedoch die angegebene Zeit, um die Analyse von Paketen abzuschließen, die vor Ablauf der Zeit durchsucht wurden, und das System analysiert mehrere Pakete gleichzeitig. Daher kann die Suche länger als die angegebene Zeit dauern. Die Standardeinheit ist Millisekunden, aber andere Einheiten können mit einem Einheitensuffix angegeben werden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe. Der Standardwert ist „5m“. **Hinweis**: Wenn die Ausgabe „extract“ lautet, gibt es einen Maximalwert für dieses Feld. Das Standardmaximum ist „5 m“, aber das Maximum kann in der laufenden Konfiguration geändert werden. Wenn die Ausgabe nicht „Extrahieren“ ist, gibt es keinen Maximalwert.

`always_return_body:` ***Boolesch***

(Optional) Gibt das Verhalten an, wenn die Abfrage keinen Paketen entspricht oder wenn die Pakete, auf die die Abfrage zutrifft, keine Dateien enthalten. Wenn der Wert wahr ist, gibt das System eine leere Datei und einen 200-Statuscode zurück. Wenn der Wert falsch ist, gibt das System einen 204-Statuscode zurück, aber keine Datei.

`from:` ***Schnur***

Der Anfangszeitstempel des Zeitbereichs, den die Suche umfassen wird, ausgedrückt in Millisekunden seit der Epoche. Ein negativer Wert gibt an, dass die Suche mit Paketen beginnt, die zu einem Zeitpunkt in der Vergangenheit erfasst wurden. Geben Sie beispielsweise -10m an, um die Suche mit Paketen zu beginnen, die 10 Minuten vor dem Zeitpunkt der Anfrage erfasst wurden. Negative Werte können mit einer anderen Zeiteinheit als Millisekunden angegeben werden, z. B. Sekunden oder Stunden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`until:` ***Schnur***

(Optional) Der Endzeitstempel des Zeitbereichs, den die Suche einschließen wird, ausgedrückt in Millisekunden seit der Epoche. Ein Wert 0 gibt an, dass die Suche mit Paketen endet, die zum Zeitpunkt der Suche erfasst wurden. Ein negativer Wert gibt an, dass die Suche mit Paketen endet, die zu einem Zeitpunkt in der Vergangenheit erfasst wurden. Geben Sie beispielsweise -5m an, um die Suche mit Paketen zu beenden, die 5 Minuten vor dem Zeitpunkt der Anfrage erfasst wurden. Negative Werte können mit einer anderen Zeiteinheit als Millisekunden angegeben werden, z. B. Sekunden oder Stunden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`bpf:` ***Schnur***

(Optional) Die Berkeley Paket Filter (BPF) -Syntax für die Paketsuche. Weitere Hinweise zur BPF-Syntax finden Sie unter Filtert Pakete mit der Berkeley-Paketfilter-Syntax.

`ip1:` ***Schnur***

(Optional) Gibt Pakete zurück, die an die angegebene IP-Adresse gesendet oder von dieser empfangen wurden.

`port1:` ***Schnur***

(Optional) Gibt Pakete zurück, die vom angegebenen Port gesendet oder empfangen wurden.

`ip2:` ***Schnur***

(Optional) Gibt Pakete zurück, die an die angegebene IP-Adresse gesendet oder von dieser empfangen wurden.

`port2:` ***Schnur***

(Optional) Gibt Pakete zurück, die vom angegebenen Port gesendet oder empfangen wurden.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "always_return_body": true,
    "bpf": "string",
    "decrypt_files": true,
    "from": "string",
    "include_secrets": true,
    "ip1": "string",
    "ip2": "string",
    "limit_bytes": "string",
    "limit_search_duration": "string",
    "output": "string",
    "port1": "string",
    "port2": "string",
    "until": "string"
}
```

Paarung

Mit dieser Ressource können Sie ein Token generieren, das für die Verbindung mit einem erforderlich ist Sensor zu einem Konsole.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
POST /pairing/token	Generieren Sie ein Token, das für die Verbindung mit dem erforderlich ist Sensor zu einem Konsole.

Einzelheiten der Operation

`POST /pairing/token`

Für diesen Vorgang gibt es keine Parameter.

Protokoll aufzeichnen

Aufzeichnungen sind strukturierte Fluss- und Transaktionsinformationen über Ereignisse in Ihrem Netzwerk.

Bevor Sie beginnen
Sie können nur auf diese REST-API-Ressource zugreifen, wenn Ihr RevealX 360-System über einen Cloud-basierten Recordstore mit Premium Investigation verfügt.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Bedienung	Beschreibung
`GET /records/cursor/{cursor}`	Veraltet. Ersetzt durch `POST /records/cursor`.
`POST /records/cursor`	Ruft Datensätze ab einem bestimmten Cursor ab.
`POST /records/search`	Führen Sie eine Datensatzprotokollabfrage durch.

Einzelheiten der Operation

`POST /records/search`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Datensatzprotokollabfrage.

`from:` ***Zahl***

Der Anfangszeitstempel des Zeitbereichs, den die Abfrage durchsucht, ausgedrückt in Millisekunden seit der Epoche. Ein negativer Wert gibt an, dass die Suche mit Datensätzen beginnt, die zu einem Zeitpunkt in der Vergangenheit erstellt wurden. Geben Sie beispielsweise -600000ms an, um die Suche mit Datensätzen zu beginnen, die 10 Minuten vor dem Zeitpunkt der Anfrage erstellt wurden. Negative Werte können mit einer anderen Zeiteinheit als Millisekunden angegeben werden, z. B. Sekunden oder Stunden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`until:` ***Zahl***

Der Endzeitstempel des Zeitbereichs, den die Abfrage durchsucht, ausgedrückt in Millisekunden seit der Epoche. Ein Wert 0 gibt an, dass die Suche mit Datensätzen endet, die zum Zeitpunkt der Anfrage erstellt wurden. Ein negativer Wert gibt an, dass die Suche mit Datensätzen endet, die zu einem Zeitpunkt in der Vergangenheit erstellt wurden. Geben Sie beispielsweise -300000ms an, um die Suche mit Datensätzen zu beenden, die 5 Minuten vor dem Zeitpunkt der Anfrage erstellt wurden. Negative Werte können mit einer anderen Zeiteinheit als Millisekunden angegeben werden, z. B. Sekunden oder Stunden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe.

`types:` ***Reihe von Zeichenketten***

(Optional) Ein Array mit einem oder mehreren Datensatzformaten. Die Abfrage gibt nur Datensätze zurück, die den angegebenen Formaten entsprechen. Wenn kein Wert angegeben ist, gibt die Abfrage Datensätze eines beliebigen Typs zurück. Gültige Werte für dieses Feld werden im Feld Datensatztyp auf der Seite Datensatzformate angezeigt. Zum Beispiel: „~cifs“.

`limit:` ***Zahl***

Die maximale Anzahl von Datensätzen, die von der Abfrage zurückgegeben wurden. Der Höchstwert darf 10000 nicht überschreiten. Der Standardwert ist 100.

`offset:` ***Zahl***

Die Anzahl der Datensätze, die in den Abfrageergebnissen übersprungen werden sollen. Die Abfrage gibt Datensätze zurück, die mit dem Offsetwert beginnen. Dieser Parameter wird häufig mit den Grenzwert- und Sortierparametern kombiniert. Der Standardwert ist 0. Für ExtraHop-Recordstores ist der Höchstwert 10.000; Informationen zum Abrufen von Datensätzen, die nach den ersten 10.000 zurückgegeben wurden, finden Sie unter POST /records/cursor/. Für Recordstores von Drittanbietern gibt es keinen Maximalwert.

`sort:` ***Reihe von Objekten***

Die Liste von einem oder mehreren Sortierobjekten, die Sortierprioritäten angeben. Die zurückgegebenen Datensätze werden in der Reihenfolge sortiert, in der die Objekte aufgelistet sind. Die Parameter sind im Abschnitt sort_item unten definiert. Wenn keine sort_item-Werte angegeben werden, werden die Datensätze in absteigender Reihenfolge nach Zeitstempel sortiert.

`field:` ***Zeichenfolge***

Der Feldname, nach dem Datensätze zurückgegeben wurden, wird sortiert.

`direction:` ***Zeichenfolge***

Die Reihenfolge, in der die zurückgegebenen Datensätze sortiert werden. Die Standardreihenfolge ist absteigend. Nachdem alle anderen Sortierkriterien

angewendet wurden oder wenn keine Sortierkriterien angegeben wurden, ist die Standardreihenfolge nach Zeitstempel absteigend.

Die folgenden Werte sind gültig:

- `asc`
- `desc`

`filter:` ***Objekt***

Das Objekt, das die Parameter enthält, die die Filterkriterien angeben. Die Parameter werden im Filterabschnitt unten definiert. Wenn keine Filterwerte angegeben werden, gibt die Abfrage alle Datensätze zurück, die dem Zeitbereich und allen angegebenen Datensatzformaten entsprechen.

`field:` ***Zeichenfolge***

Der Name des Feldes in dem Datensatz, der gefiltert werden soll. Die Abfrage vergleicht den Inhalt des Feldparameters mit dem Wert des Operandenparameters. Wenn der angegebene Feldname „.any“ ist, wird die Vereinigung aller Feldwerte durchsucht. Wenn der angegebene Feldname „.ipaddr“ oder „.port“ lautet, werden die Client-, Server-, Sender- und Empfängerrollen in die Suche einbezogen. Feldnamen befinden sich in Datensatzformaten, die im ExtraHop-System eingesehen werden können.

`operator:` ***Zeichenfolge***

Die Vergleichsmethode, die angewendet wird, wenn der Operandenwert mit dem Feldinhalt verglichen wird. Alle Filterobjekte benötigen einen Operator.

Die folgenden Werte sind gültig:

- `>`
- `<`
- `<=`
- `>=`
- `=`
- `!=`
- `startswith`
- `~`
- `!~`
- `and`
- `or`
- `not`
- `exists`
- `not_exists`
- `in`
- `not_in`

`operand:` ***Zeichenfolge oder Zahl oder Objekt***

Der Wert, den die Abfrage abzugleichen versucht. Die Abfrage vergleicht den Wert des Operanden mit dem Inhalt des Feldparameters und wendet die durch den Operatorparameter angegebene Vergleichsmethode an. Sie können den Operanden-Datentyp explizit angeben, wie in der REST-API-Leitfaden.

`rules:` ***Reihe von Objekten***

Die Liste eines oder mehrerer Filterobjekte innerhalb eines einzelnen Filterobjekts. Filterobjekte können rekursiv eingebettet werden. Für diesen Parameter sind nur die Operatoren „und“, „oder“ oder „nicht“ zulässig.

`context_ttl:` ***Zahl***

Die Zeitspanne, in der der Suchkontext aktiv bleibt. Die Standardeinheit ist Millisekunden, aber andere Einheiten können mit einem Einheitensuffix angegeben werden. Sehen Sie die REST-API-Leitfaden für unterstützte Zeiteinheiten und Suffixe. In RevealX Enterprise ist dieses Feld nur gültig, wenn Datensätze in einem ExtraHop-Recordstore (z. B. einem EXA 5300) oder auf CrowdStrike LogScale gespeichert sind. In RevealX 360 ist dieses Feld nur für Systeme gültig, die über einen Cloud-basierten Recordstore mit Premium Investigation verfügen. Sowohl in RevealX Enterprise mit CrowdStrike LogScale als auch in RevealX 360 mit Premium Investigation ist dieses Feld ungültig, wenn die Sortier- oder Offsetfelder angegeben sind.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "context_ttl": 0,
    "filter": {
        "field": "string",
        "operator": "string",
        "operand": "string",
        "rules": []
    },
    "from": 0,
    "limit": 0,
    "offset": 0,
    "sort": {
        "field": "string",
        "direction": "string"
    },
    "types": [],
    "until": 0
}
```

`POST /records/cursor`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Cursor-ID, die die nächste Seite mit Ergebnissen in der Abfrage angibt.

`cursor:` ***Zeichenfolge***

Der eindeutige Bezeichner des Cursors, der die nächste Seite mit Ergebnissen in der Abfrage angibt.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "cursor": "string"
}
```

`context_ttl:` ***Zahl***

(Optional) Die Zeitspanne, in der der Suchkontext aktiv bleibt, ausgedrückt in Millisekunden. Nach Ablauf der angegebenen Zeit wird der Cursor ungültig und Sie können keine weiteren Datensätze mehr aus der Suche abrufen. Geben Sie diesen Parameter an, um den zuvor angegebenen Suchkontext zu erweitern.

`GET /records/cursor/{cursor}`

Geben Sie die folgenden Parameter an.

`cursor:` ***Schnur***

Die Cursor-ID.

`context_ttl:` ***Zahl***

(Optional) Die Zeitspanne, in der der Suchkontext aktiv bleibt, ausgedrückt in Millisekunden.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "cursor": "string",
    "from": 0,
    "lookback_exceeded": true,
    "lookback_truncated": true,
    "records": {},
    "total": 0,
    "until": 0,
    "warnings": {}
}
```

Operandenwerte in Datensatzabfragen

Die `operand` Feld in der `POST /records/search` Methode gibt den Wert an, den eine Datensatzabfrage zu finden versucht. Sie können entweder nur den Wert oder sowohl den Datentyp als auch den Wert angeben. Wenn Sie nur den Wert angeben, bezieht sich die Abfrage auf das Datensatzformat, das mit dem verknüpft ist `field` Parameter zur Bestimmung des Datentyps des Werts.

Wenn Sie beispielsweise nach einer IP-Adresse suchen möchten, können Sie einen IP-Adressdatentyp angeben und dann die tatsächliche Adresse als Wert angeben.

Das folgende Beispiel spezifiziert explizit den Datentyp und den Wert des Operanden:

```
{
    "from": -1000,
    "filter": {
        "field" : "senderAddr",
        "operator": "=",
        "operand" : { "type" : "ipaddr4", "value": "1.2.3.4" }
    }
}
```

Das folgende Beispiel spezifiziert nur den Wert des Operanden:

```
{
    "from": -1000,
    "filter": {
        "field" : "senderAddr",
        "operator": "=",
        "operand" : "1.2.3.4"
    }
}
```

Sie können die folgenden Datentypen explizit angeben in der `operand` Feld:

- Anwendung
- boolesch
- Gerät

  Hinweis: Sie müssen die Discovery-ID des Gerät im Wertfeld angeben. Sie finden die Discovery-ID eines Gerät über `POST /devices/search` Betrieb.
- Gerätefilter
- Gerätegruppe

- Flow-Schnittstelle
- Flow-Netzwerk
- iPad dr4
- iPad dr6
- Nummer
- Netzwerk_Lokalität
- Objekt
- Schnur

Die `operand` Feld unterstützt die CIDR-Notation beim Filtern nach IP-Adressen; das `operator` Feld muss auf „=“ oder „! =“.

Sie können mehrere Filter angeben, indem Sie den `rules` Option, wie im folgenden Beispiel gezeigt:

```
{
  "filter": {
    "operator": "and",
    "rules": [
      {
        "field": "method",
        "operand": "SMB2_READ",
        "operator": "="
      },
      {
        "field": "reqL2Bytes",
        "operand": "100",
        "operator": ">"
      }
    ]
  },
  "types": [
    "~cifs"
  ],
    "from": "-30m"
}
```

Datensätze mit einem Gerätegruppenfilter abfragen

Um Datensätze in der REST-API nach Gerätegruppe zu filtern, müssen Sie eine senden `POST` Anfrage an den `/records/search` Endpunkt mit einem Datensatzabfragefilter, der die folgenden Kriterien erfüllt:

- Die `field` muss Geräte angeben, wie `client`, `server`, `sender`, oder `receiver`.
- Die `operator` muss entweder sein `in` oder `not_in`.
- Die `operand` `type` muss sein `device_group`.
- Die `operand` `value` muss eine Zeichenkettendarstellung der numerischen Gerätegruppen-ID sein. Sie können Gerätegruppen-IDs abrufen, indem Sie den Vorgang GET /devicegroup ausführen und den Inhalt des `id` Feld in der Antwort.

Die folgende Abfrage sucht beispielsweise nach Datensätzen, in denen das Client-Gerät Mitglied einer Gerätegruppe mit der ID 200 war:

```
{
    "from": "-30m",
    "filter": {
        "field": "client",
        "operator": "in",
        "operand": {
            "type": "device_group",
            "value": "200"
        }
    }
```

```
}
```

Sie können Datensätze auch nach Gerätegruppenkriterien filtern, ohne eine Gerätegruppe zu erstellen, indem Sie den Operandentyp angeben als `device_filter`. Mit der folgenden Abfrage wird beispielsweise nach Datensätzen gesucht, in denen auf dem Client-Gerät Windows 10 ausgeführt wird:

```
{
    "from": "-30m",
    "filter": {
        "field": "client",
        "operator": "in",
        "operand": {
            "type": "device_filter",
            "value": {
                "field": "software",
                "operand": "windows_10",
                "operator": "="
            }
        }
    }
}
```

Hinweis: Operandenwerte mit Typ `device_filter` für die Datensatzsuche sind genauso formatiert wie Gerätesuchfilter. Weitere Informationen finden Sie unter Operandenwerte für Gerätegruppen.

Datensätze mit einem Netzwerk-Lokalitätsfilter abfragen

Um Datensätze in der REST-API nach Gerätegruppe zu filtern, müssen Sie eine POST-Anfrage an die `/records/search` Endpunkt mit einem Datensatzabfragefilter, der die folgenden Kriterien erfüllt:

- Das Feld muss ein Datensatzfeld sein, das eine IP-Adresse angibt, z. B. `clientAddr`, `serverAddr`, `senderAddr`, oder `receiverAddr`.
- Der Betreiber muss entweder `in` oder `not_in`.
- Der Operandentyp muss `network_locality`.
- Der Operandenwert muss eine Zeichenkettendarstellung einer numerischen Netzwerk-Lokalitäts-ID sein. Sie können Lokalitäts-IDs mit dem `GET /networklocalities` Betrieb.

Die folgende Abfrage sucht beispielsweise nach Datensätzen, bei denen sich das Client-Gerät in einer Netzwerklokalität mit der ID von befindet `123`:

```
{
    "from": "-30m",
    "filter": {
        "field": "clientAddr",
        "operand": {
            "type": "network_locality",
            "value": "123"
        },
        "operator": "in"
    }
}
```

Unterstützte Zeiteinheiten

Für die meisten Parameter ist die Standardeinheit für die Zeitmessung Millisekunden. Die folgenden Parameter geben jedoch alternative Zeiteinheiten wie Minuten und Stunden zurück oder akzeptieren diese:

- Gerät
  - aktive_von

    - aktiv_bis
- Gerätegruppe
    - aktive_von
    - aktiv_bis
- Metriken
    - von
    - bis
- Protokoll aufzeichnen
    - von
    - bis
    - kontext_ttl

Die folgende Tabelle zeigt die unterstützten Zeiteinheiten:

Zeiteinheit	Einheitensuffix
Jahr	`y`
Monat	`M`
Woche	`w`
Tag	`d`
Stunde	`h`
Minute	`m`
Zweiter	`s`
Millisekunde	`ms`

Um für einen Parameter eine andere Zeiteinheit als Millisekunden anzugeben, hängen Sie das Einheitensuffix an den Wert an. Um beispielsweise Geräte anzufordern, die in den letzten 30 Minuten aktiv waren, geben Sie den folgenden Parameterwert an:

```
GET /api/v1/devices?active_from=-30m
```

Das folgende Beispiel spezifiziert eine Suche nach HTTP Datensätze, die vor 1 bis 2 Stunden erstellt wurden:

```
{
    "from": "-2h",
    "until": "-1h",
    "types": ["~http"]
}
```

Bericht

Ein Bericht ist eine PDF-Datei mit einem Dashboard, das Sie für die E-Mail-Zustellung an einen oder mehrere Empfänger planen können. Sie können angeben, wie oft die Berichts-E-Mail zugestellt wird und in welchem Zeitintervall die in der PDF-Datei enthaltenen Dashboard-Daten angezeigt werden.

Wichtig: Sie können nur Berichte von einer ECA-VM aus planen.

Hier sind einige wichtige Überlegungen zu Dashboard-Berichten:

- Sie können nur einen Bericht für Dashboards erstellen, die Ihnen gehören oder die mit Ihnen geteilt wurden. Ihre Fähigkeit, einen Bericht zu erstellen, hängt von Ihren Benutzerrechten ab. Wenden Sie sich an Ihren ExtraHop-Administrator, um Hilfe zu erhalten.
- Jeder Bericht kann nur mit einem Dashboard verknüpft werden.
- Wenn Sie einen Bericht für ein Dashboard erstellt haben, das später gelöscht wurde oder auf das Sie nicht mehr zugreifen konnten, wird die geplante E-Mail weiterhin an die Empfänger gesendet. Die E-Mail wird die PDF-Datei jedoch nicht enthalten und stattdessen die Empfänger darüber informieren, dass das Dashboard für den Berichtsbesitzer nicht verfügbar ist.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
GET /reports	Rufen Sie alle Berichte ab.
POST/Berichte	Erstellen Sie einen Bericht.
/reports/ {id} LÖSCHEN	Löschen Sie einen bestimmten Bericht.
GET /reports/ {id}	Rufen Sie einen bestimmten Bericht ab.
PATCH /reports/ {id}	Aktualisieren Sie einen bestimmten Bericht.
GET /reports/ {id} /contents	Rufen Sie den Inhalt eines bestimmten Berichts ab.
PUT /reports/ {id} /contents	Ersetzt den Inhalt eines bestimmten Berichts.
GET /reports/ {id} /herunterladen	Rufen Sie das PDF eines Berichts ab.
POST /reports/ {id} /Warteschlange	Generieren und senden Sie sofort einen bestimmten Bericht.

Einzelheiten der Operation

`GET /reports`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "cc": [],
    "description": "string",
    "email_message": "string",
    "email_subject": "string",
    "enabled": true,
    "from": "string",
    "id": 0,
    "include_links": "string",
    "name": "string",
    "output": {},
    "owner": "string",
    "schedule": {},
    "until": "string"
}
```

`POST /reports`

Geben Sie die folgenden Parameter an.

`body`: ***Objekt***
Der Inhalt des Berichts.

`name:` ***Schnur***

Der Name des Berichts.

`description:` ***Schnur***

(Optional) Die Beschreibung des Berichts.

`owner:` ***Schnur***

Der Benutzername des Berichtsbesitzers.

`cc:` ***Reihe von Zeichenketten***

Die Liste der E-Mail-Adressen, die nicht in einer E-Mail-Gruppe enthalten sind, für den Empfang von Berichten.

`enabled:` ***Boolesch***

(Optional) Gibt an, ob der Bericht aktiviert ist.

`from:` ***Schnur***

Der Anfangszeitstempel des Zeitintervalls für den Berichtsinhalt, relativ zur aktuellen Uhrzeit, ausgedrückt in Millisekunden.

`until:` ***Schnur***

(Optional) Der Endzeitstempel des Zeitintervalls für den Berichtsinhalt, relativ zur aktuellen Uhrzeit, ausgedrückt in Millisekunden.

`email_subject:` ***Schnur***

(Optional) Der Inhalt der Betreffzeile für die Berichts-E-Mail.

`schedule:` ***Objekt***

(Optional) Das Objekt, das die Parameter enthält, die den geplanten Zeitraum für die Generierung und das Senden des Berichts angeben. Die Parameter sind im Abschnitt schedule_type unten definiert.

`type:` ***Schnur***

Die Art des Lieferplans für den Bericht.

Die folgenden Werte sind gültig:

- `hourly`
- `daily`
- `weekly`
- `monthly`

`at:` ***Reihe von Objekten***

(Optional) Die Liste der Objekte, die die Übermittlungsparameter für den Bericht angeben. Die Parameter sind im Abschnitt at_type unten definiert.

`by_day:` ***Reihe von Zeichenketten***

(Optional) Die Wochentage, an denen der Bericht gesendet werden soll.

Die folgenden Werte sind gültig:

- `mo`
- `tu`
- `we`
- `th`
- `fr`
- `sa`
- `su`

`on_day:` ***Zahl***

(Optional) Der Tag des Monats, an dem der Bericht ausgeführt werden soll.

`tz:` ***Schnur***

(Optional) Die Zeitzone, in der der Bericht gesendet werden soll.

`hour:` ***Zahl***

(Optional) Die Stunde, zu der der Bericht gesendet werden soll.

`minute:` ***Zahl***

(Optional) Die Minute, in der der Bericht gesendet werden soll.

`interval:` ***Schnur***

(Optional) Das Intervall kann previous_week, previous_month oder nichts sein.

Die folgenden Werte sind gültig:

- `previous_week`
- `previous_month`

`output:` ***Objekt***

Das Objekt, das die Parameter enthält, die das Ausgabeformat für den Bericht angeben. Die Parameter sind im Abschnitt format_type unten definiert.

`type:` ***Schnur***

Das Ausgabeformat für den Bericht.

Die folgenden Werte sind gültig:

- `pdf`

`width:` ***Schnur***

(Optional) Die Breitenoption für die Berichtsausgabe.

Die folgenden Werte sind gültig:

- `narrow`
- `medium`
- `wide`

`pagination:` ***Schnur***

(Optional) Das Paginierungsschema für die Berichtsausgabe.

Die folgenden Werte sind gültig:

- `per_region`

`theme:` ***Schnur***

(Optional) Das Anzeigedesign für die Berichtsausgabe.

Die folgenden Werte sind gültig:

- `light`
- `dark`
- `space`
- `contrast`

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "cc": [],
    "description": "string",
    "email_subject": "string",
    "enabled": true,
    "from": "string",
    "name": "string",
    "output": {
        "type": "string",
```

```
        "width": "string",
        "pagination": "string",
        "theme": "string"
    },
    "owner": "string",
    "schedule": {
        "type": "string",
        "at": {
            "by_day": [],
            "on_day": 0,
            "tz": "string",
            "hour": 0,
            "minute": 0
        },
        "interval": "string"
    },
    "until": "string"
}
```

`POST /reports/{id}/queue`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für den Bericht.

`PATCH /reports/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für den Bericht.

`body:` ***Objekt***
Der Inhalt des Berichts.

`name:` ***Schnur***
Der Name des Berichts.

`description:` ***Schnur***
(Optional) Die Beschreibung des Berichts.

`owner:` ***Schnur***
Der Benutzername des Berichtsbesitzers.

`cc:` ***Reihe von Zeichenketten***
Die Liste der E-Mail-Adressen, die nicht in einer E-Mail-Gruppe enthalten sind, für den Empfang von Berichten.

`enabled:` ***Boolesch***
(Optional) Gibt an, ob der Bericht aktiviert ist.

`from:` ***Schnur***
Der Anfangszeitstempel des Zeitintervalls für den Berichtsinhalt, relativ zur aktuellen Uhrzeit, ausgedrückt in Millisekunden.

`until:` ***Schnur***
(Optional) Der Endzeitstempel des Zeitintervalls für den Berichtsinhalt, relativ zur aktuellen Uhrzeit, ausgedrückt in Millisekunden.

`email_subject:` ***Schnur***
(Optional) Der Inhalt der Betreffzeile für die Berichts-E-Mail.

`schedule:` ***Objekt***

(Optional) Das Objekt, das die Parameter enthält, die den geplanten Zeitraum für die Generierung und das Senden des Berichts angeben. Die Parameter sind im Abschnitt schedule_type unten definiert.

`type:` ***Schnur***

Die Art des Lieferplans für den Bericht.

Die folgenden Werte sind gültig:

- `hourly`
- `daily`
- `weekly`
- `monthly`

`at:` ***Reihe von Objekten***

(Optional) Die Liste der Objekte, die die Übermittlungsparameter für den Bericht angeben. Die Parameter sind im Abschnitt at_type unten definiert.

`by_day:` ***Reihe von Zeichenketten***

(Optional) Die Wochentage, an denen der Bericht gesendet werden soll.

Die folgenden Werte sind gültig:

- `mo`
- `tu`
- `we`
- `th`
- `fr`
- `sa`
- `su`

`on_day:` ***Zahl***

(Optional) Der Tag des Monats, an dem der Bericht ausgeführt werden soll.

`tz:` ***Schnur***

(Optional) Die Zeitzone, in der der Bericht gesendet werden soll.

`hour:` ***Zahl***

(Optional) Die Stunde, zu der der Bericht gesendet werden soll.

`minute:` ***Zahl***

(Optional) Die Minute, in der der Bericht gesendet werden soll.

`interval:` ***Schnur***

(Optional) Das Intervall kann previous_week, previous_month oder nichts sein.

Die folgenden Werte sind gültig:

- `previous_week`
- `previous_month`

`output:` ***Objekt***

Das Objekt, das die Parameter enthält, die das Ausgabeformat für den Bericht angeben. Die Parameter sind im Abschnitt format_type unten definiert.

`type:` ***Schnur***

Das Ausgabeformat für den Bericht.

Die folgenden Werte sind gültig:

- `pdf`

`width:` ***Schnur***

(Optional) Die Breitenoption für die Berichtsausgabe.

Die folgenden Werte sind gültig:

- `narrow`
- `medium`
- `wide`

`pagination:` ***Schnur***

(Optional) Das Paginierungsschema für die Berichtsausgabe.

Die folgenden Werte sind gültig:

- `per_region`

`theme:` ***Schnur***

(Optional) Das Anzeigedesign für die Berichtsausgabe.

Die folgenden Werte sind gültig:

- `light`
- `dark`
- `space`
- `contrast`

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "cc": [],
    "description": "string",
    "email_subject": "string",
    "enabled": true,
    "from": "string",
    "name": "string",
    "output": {
        "type": "string",
        "width": "string",
        "pagination": "string",
        "theme": "string"
    },
    "owner": "string",
    "schedule": {
        "type": "string",
        "at": {
            "by_day": [],
            "on_day": 0,
            "tz": "string",
            "hour": 0,
            "minute": 0
        },
        "interval": "string"
    },
    "until": "string"
}
```

`GET /reports/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für den Bericht.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "cc": [],
    "description": "string",
    "email_message": "string",
    "email_subject": "string",
    "enabled": true,
    "from": "string",
    "id": 0,
    "include_links": "string",
    "name": "string",
    "output": {},
    "owner": "string",
    "schedule": {},
    "until": "string"
}
```

`GET /reports/{id}/download`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für den Bericht.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "cc": [],
    "description": "string",
    "email_message": "string",
    "email_subject": "string",
    "enabled": true,
    "from": "string",
    "id": 0,
    "include_links": "string",
    "name": "string",
    "output": {},
    "owner": "string",
    "schedule": {},
    "until": "string"
}
```

`DELETE /reports/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für den Bericht.

`GET /reports/{id}/contents`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für den Bericht.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "dashboard_id": 0,
```

```
    "params": {},
    "type": "string"
}
```

`PUT /reports/{id}/contents`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für den Bericht.

`body:` ***Objekt***
Der Inhalt des Berichts.

Software

Sie können sich eine Liste der Software ansehen, die das ExtraHop-System in Ihrem Netzwerk beobachtet hat.

Betrieb	Beschreibung
Holen Sie sich /software	Rufen Sie die vom ExtraHop-System beobachtete Software ab.
GET /software/ {id}	Rufen Sie die vom ExtraHop-System beobachtete Software anhand der ID ab.

Einzelheiten der Operation

`GET /software`

Geben Sie die folgenden Parameter an.

`software_type:` ***Schnur***
(Optional) Die Art der Software.

`name:` ***Schnur***
(Optional) Der Name der Software.

`version:` ***Schnur***
(Optional) Die Version der Software.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "description": "string",
    "id": "string",
    "name": "string",
    "software_type": "string",
    "version": "string"
}
```

`GET /software/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***
Die eindeutige Kennung für die Software.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "description": "string",
    "id": "string",
    "name": "string",
    "software_type": "string",
    "version": "string"
}
```

Tag

Mithilfe von Geräte-Tags können Sie ein Gerät oder eine Gruppe von Geräten anhand eines Merkmals zuordnen.

Sie könnten zum Beispiel alle Ihre taggen HTTP Server oder kennzeichnet alle Geräte, die sich in einem gemeinsamen Subnetz befinden. Weitere Informationen finden Sie unter Taggen Sie ein Gerät über die REST-API.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
GET /tags	Ruft alle Tags ab.
POST /Schlagworte	Erstellen Sie ein neues Tag.
/tags/ {id} LÖSCHEN	Löscht ein bestimmtes Tag.
GET /tags/ {id}	Ruft ein bestimmtes Tag ab.
PATCH /tags/ {id}	Wenden Sie Aktualisierungen auf ein bestimmtes Tag an.
GET /tags/ {id} /devices	Ruft alle Geräte ab, die einem bestimmten Tag zugewiesen sind.
POST /tags/ {id} /Geräte	Weisen Sie Geräten ein bestimmtes Tag zu und heben Sie die Zuweisung auf.
LÖSCHEN /tags/ {id} /devices/ {child-id}	Heben Sie die Zuweisung eines Gerät zu einem bestimmten Tag auf.
POST /tags/ {id} /devices/ {child-id}	Weisen Sie ein Gerät einem bestimmten Tag zu.

Einzelheiten der Operation

`GET /tags`

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "id": 0,
    "mod_time": 0,
    "name": "string"
}
```

`POST /tags`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Wendet die angegebenen Eigenschaftswerte auf das neue Tag an.

`name:` ***Schnur***

Der Zeichenkettenwert für das Tag.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "name": "string"
}
```

`GET /tags/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Tag.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "id": 0,
    "mod_time": 0,
    "name": "string"
}
```

`DELETE /tags/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Tag.

`PATCH /tags/{id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Wendet die angegebenen Eigenschaftswertaktualisierungen auf das Tag an.

`id:` ***Zahl***

Die eindeutige Kennung für das Tag.

`GET /tags/{id}/devices`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für das Tag.

`POST /tags/{id}/devices`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Listen mit eindeutigen Kennungen für Gerät zum Zuweisen und Aufheben der Zuweisung.

`assign:` ***Reihe von Zahlen***

IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***

IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***

Die eindeutige Kennung für das Tag.

`POST /tags/{id}/devices/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für das Gerät.

`id:` ***Zahl***

die eindeutige Kennung für das Tag.

`DELETE /tags/{id}/devices/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für das Gerät.

`id:` ***Zahl***

Die eindeutige Kennung für das Tag.

Erfassung von Bedrohungen

Mit der Threat Collection-Ressource können Sie kostenlose und kommerzielle Inhalte hochladen Sammlungen von Bedrohungen wird von der Security Community für Ihr RevealX-System angeboten.

- Sie müssen Bedrohungssammlungen einzeln auf Ihre Command-Appliance oder RevealX 360 hochladen und auf alle verbundenen Sensoren.
- Benutzerdefinierte Bedrohungssammlungen müssen in Structured Threat Information eXpression (STIX) als TAR.GZ -Dateien formatiert werden. RevealX unterstützt derzeit die STIX-Versionen 1.0 - 1.2.
- Sie können Bedrohungssammlungen direkt auf die RevealX 360-Systeme hochladen, um sie selbst zu verwalten Sensoren. Wenden Sie sich an den ExtraHop-Support, um eine Bedrohungssammlung auf ExtraHop-Managed hochzuladen Sensoren.
- Die maximale Anzahl an Observables, die eine Bedrohungssammlung enthalten kann, hängt von Ihrer Plattform und Lizenz ab. Kontaktieren Sie Ihren ExtraHop-Vertreter für weitere Informationen.

Hinweis: Dieses Thema gilt nur für ExtraHop RevealX Premium und Ultra.

Informationen zum Hochladen von STIX-Dateien über das ExtraHop-System finden Sie unter Laden Sie STIX-Dateien über die REST-API hoch.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Bedienung	Beschreibung
GET /threatcollections	Rufen Sie alle Bedrohungssammlungen ab.
POST /threat collections	Erstellen Sie eine neue Bedrohungssammlung.
/threatcollections/ {id} LÖSCHEN	Löschen Sie eine Bedrohungssammlung.
PUT /threatcollections/ {id}	Laden Sie eine neue Bedrohungssammlung hoch. ExtraHop unterstützt derzeit die STIX-Versionen 1.0 - 1.2. **Hinweis:** Wenn auf dem ExtraHop-System bereits eine Bedrohungssammlung mit demselben Namen existiert, wird die bestehende Bedrohungssammlung überschrieben.
GET /threatcollections/ {id} /observables	Ruft die Anzahl der STIX-Observables ab, die aus einer Bedrohungssammlung geladen wurden, z. B. IP-Adresse, Hostname oder URI.

Einzelheiten der Operation

`GET /threatcollections`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "id": 0,
    "last_updated": 0,
    "name": "string",
    "observables": 0,
    "user_key": "string"
}
```

`POST /threatcollections`

Geben Sie die folgenden Parameter an.

`user_key:` ***Schnur***
(Optional) Die vom Benutzer angegebene Kennung für die Bedrohungssammlung. Wenn dieser Parameter nicht angegeben ist, wird der Name der Bedrohungssammlung für diesen Wert ohne Leerzeichen oder Satzzeichen festgelegt.

`name:` ***Schnur***
Der Name für die Bedrohungssammlung.

`file:` ***Dateiname***
Der Dateiname für die Bedrohungssammlung.

`PUT /threatcollections/~{userKey}`

Geben Sie die folgenden Parameter an.

`userKey:` ***Schnur***
Die vom Benutzer angegebene Kennung für die Bedrohungssammlung.

`name:` ***Schnur***
(Optional) Der Name für die Bedrohungssammlung.

`file:` ***Dateiname***
(Optional) Der Dateiname für die Bedrohungssammlung.

`DELETE /threatcollections/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***
Die eindeutige Kennung für die Bedrohungssammlung.

`GET /threatcollections/{id}/observables`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***
Die eindeutige Kennung für die Bedrohungssammlung.

Auslösen

Trigger sind benutzerdefinierte Skripts, die bei einem vordefinierten Ereignis eine Aktion ausführen.

Sie können beispielsweise einen Auslöser schreiben, um jedes Mal eine benutzerdefinierte Metrik Datensatz, wenn HTTP Eine Anfrage erfolgt, oder klassifizieren Sie den Datenverkehr für einen bestimmten Server als Anwendungsserver. Weitere Informationen finden Sie in der Trigger-API-Referenz. Zusätzliche Implementierungshinweise zu erweiterten Optionen finden Sie unter Erweiterte Trigger-Optionen.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Bedienung	Beschreibung
GET /triggert	Ruft alle Trigger ab.
POST /Trigger	Erstellen Sie einen neuen Auslöser.
POST /triggers/externe Daten	Sendet Daten an die Trigger-API, indem das Ereignis EXTERNAL_DATA ausgeführt wird. Sie können auf die Daten zugreifen über ExternalData Trigger-Klasse. **Hinweis:** Dieser Vorgang ist für Command-Appliances oder RevealX 360 nicht verfügbar.
LÖSCHE /triggers/ {id}	Löscht einen bestimmten Bezeichner.
GET /triggers/ {id}	Ruft einen bestimmten Auslöser anhand einer eindeutigen Kennung ab.
PATCH /trigger/ {id}	Aktualisieren Sie einen vorhandenen Auslöser.
GET /triggers/ {id} /devicegroups	Alle abrufen Gerätegruppen die einem bestimmten Auslöser zugewiesen sind.
POST /triggers/ {id} /devicegroups	Weisen Sie Gerätegruppen einen bestimmten Auslöser zu und heben Sie die Zuweisung auf.
LÖSCHE /triggers/ {id} /devicegroups/ {child-id}	Heben Sie die Zuweisung einer Gerätegruppe zu einem bestimmten Auslöser auf.
POST /triggers/ {id} /devicegroups/ {child-id}	Ordnen Sie eine Gerätegruppe einem bestimmten Auslöser zu.

Bedienung	Beschreibung
GET /triggers/ {id} /devices	Ruft alle Geräte ab, die einem bestimmten Auslöser zugewiesen sind.
POST /trigger/ {id} /geräte	Weisen Sie Geräten einen bestimmten Auslöser zu und heben Sie die Zuweisung auf.
LÖSCHE /triggers/ {id} /devices/ {child-id}	Hebt die Zuweisung eines Gerät zu einem bestimmten Auslöser auf.
POST /trigger/ {id} /devices/ {child-id}	Ordnen Sie ein Gerät einem bestimmten Auslöser zu.

Einzelheiten der Operation

`GET /triggers`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "apply_all": true,
    "author": "string",
    "debug": true,
    "description": "string",
    "disabled": true,
    "event": "string",
    "events": [
        "string"
    ],
    "hints": {},
    "id": 0,
    "mod_time": 0,
    "name": "string",
    "script": "string"
}
```

`DELETE /triggers/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für den Auslöser.

`POST /triggers/externaldata`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Das Objekt, das die zu sendenden Daten enthält, wird durch das Ereignis EXTERNAL_DATA ausgelöst.

`type:` ***Schnur***

Ein Zeichenkettenbezeichner, der die im Body-Parameter enthaltenen Daten beschreibt. Sie könnten beispielsweise „Phantomdaten“ für Daten angeben, die von der Phantom SOAR-Plattform gesendet werden.

`body:` ***Objekt***

Die Daten, an die gesendet werden sollen, werden durch das Ereignis EXTERNAL_DATA ausgelöst. Auf diese Daten kann im Auslöser mit der Eigenschaft 'ExternalData.Body' zugegriffen werden.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "body": {},
    "type": "string"
}
```

`POST /triggers`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Eigenschaftswerte für den neuen Auslöser.

`name:` ***Schnur***

Der freundliche Name für den Auslöser.

`description:` ***Schnur***

(Optional) Eine optionale Beschreibung des Auslöser.

`author:` ***Schnur***

Der Name des Erstellers des Auslöser.

`script:` ***Schnur***

Der JavaScript-Inhalt des Auslöser.

`event:` ***Schnur***

(Optional) Veraltet. Ersetzt durch das Feld Ereignisse.

`events:` ***Reihe von Zeichenketten***

Die Liste der Ereignisse, bei denen der Auslöser ausgeführt wird, ausgedrückt als JSON-Array.

`disabled:` ***Boolesch***

Gibt an, ob der Auslöser ausgeführt werden kann.

`debug:` ***Boolesch***

Gibt an, ob Debug-Anweisungen für den Auslöser gedruckt werden.

`apply_all:` ***Boolesch***

Gibt an, ob der Auslöser für alle relevanten Ressourcen gilt.

`hints:` ***Objekt***

Optionen, die auf ausgewählten Triggerereignissen basieren. Weitere Informationen zum Hints-Objekt finden Sie in der REST-API-Leitfaden.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "apply_all": true,
    "author": "string",
    "debug": true,
    "description": "string",
    "disabled": true,
    "event": "string",
    "events": [
        "string"
    ],
    "hints": {},
```

```
    "name": "string",
    "script": "string"
}
```

`PATCH /triggers/{id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Der Eigenschaftswert wird für den Auslöser aktualisiert.

`id:` ***Zahl***
Die eindeutige Kennung für den Auslöser.

`GET /triggers/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für den Auslöser.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "apply_all": true,
    "author": "string",
    "debug": true,
    "description": "string",
    "disabled": true,
    "event": "string",
    "events": [
        "string"
    ],
    "hints": {},
    "id": 0,
    "mod_time": 0,
    "name": "string",
    "script": "string"
}
```

`GET /triggers/{id}/devicegroups`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für den Auslöser.

`POST /triggers/{id}/devicegroups`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Eine Liste eindeutiger Identifikatoren für Gerätegruppen, die einem Auslöser zugewiesen sind oder nicht.

`assign:` ***Reihe von Zahlen***
IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***
IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***

Die eindeutige Kennung für den Auslöser.

`POST /triggers/{id}/devicegroups/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für die Gerätegruppe.

`id:` ***Zahl***

Die eindeutige Kennung für den Auslöser.

`DELETE /triggers/{id}/devicegroups/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***

Die eindeutige Kennung für die Gerätegruppe.

`id:` ***Zahl***

Die eindeutige Kennung für den Auslöser.

`GET /triggers/{id}/devices`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***

Die eindeutige Kennung für den Auslöser.

`POST /triggers/{id}/devices`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Eine Liste eindeutiger Identifikatoren für Geräte, die einem Auslöser zugewiesen sind oder nicht zugewiesen sind.

`assign:` ***Reihe von Zahlen***

IDs der zuzuweisenden Ressourcen

`unassign:` ***Reihe von Zahlen***

IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```

`id:` ***Zahl***

Die eindeutige Kennung für den Auslöser.

`POST /triggers/{id}/devices/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für das Gerät.

`id:` ***Zahl***
Die eindeutige Kennung für den Auslöser.

`DELETE /triggers/{id}/devices/{child-id}`

Geben Sie die folgenden Parameter an.

`child-id:` ***Zahl***
Die eindeutige Kennung für das Gerät.

`id:` ***Zahl***
Die eindeutige Kennung für den Auslöser.

Benutzergruppe

Mit der Benutzergruppenressource können Sie Benutzergruppen und ihre Dashboard-Freigabezuordnungen verwalten und aktualisieren.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
GET /usergroups	Ruft alle Benutzergruppen ab.
POST /Benutzergruppen	Erstellen Sie eine neue Benutzergruppe.
/usergroups/ {id} LÖSCHEN	Löscht eine bestimmte Benutzergruppe.
GET /usergroups/ {id}	Rufen Sie eine bestimmte Benutzergruppe ab.
PATCH /Benutzergruppen/ {id}	Aktualisieren Sie eine bestimmte Benutzergruppe.
/usergroups/ {id} /associations LÖSCHEN	Löschen Sie alle Verknüpfungen zum Teilen von Dashboard mit einer bestimmten Benutzergruppe.
GET /usergroups/ {id} /members	Ruft alle Mitglieder einer bestimmten Benutzergruppe ab.
PATCH /usergroups/ {id} /members	Weisen Sie einer Benutzergruppe Benutzer zu oder heben Sie deren Zuweisung auf.
PUT /usergroups/ {id} /members	Ersetzen Sie Benutzergruppenzuweisungen.

Einzelheiten der Operation

`GET /usergroups`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "display_name": "string",
    "enabled": true,
    "id": "string",
```

```
    "is_remote": true,
    "last_sync_time": 0,
    "name": "string",
    "rights": []
}
```

`POST /usergroups`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Die Eigenschaften der Benutzergruppe.

`name:` ***Schnur***

Der Name der Benutzergruppe.

`enabled:` ***Boolescher Wert***

Zeigt an, ob die Benutzergruppe aktiviert ist.

Geben Sie den Body-Parameter im folgenden JSON-Format an.

```
{
    "enabled": true,
    "name": "string"
}
```

`PATCH /usergroups/{id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***

Der Eigenschaftswert wird für die spezifische Benutzergruppe aktualisiert.

`id:` ***Schnur***

Die eindeutige Kennung für die Benutzergruppe.

`GET /usergroups/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***

Die eindeutige Kennung für die Benutzergruppe.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "display_name": "string",
    "enabled": true,
    "id": "string",
    "is_remote": true,
    "last_sync_time": 0,
    "name": "string",
    "rights": []
}
```

`DELETE /usergroups/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***
Die eindeutige Kennung für die Benutzergruppe.

`DELETE /usergroups/{id}/associations`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***
Die eindeutige Kennung für die Benutzergruppe.

`GET /usergroups/{id}/members`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***
Die eindeutige Kennung für die Benutzergruppe.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "users": {}
}
```

`PATCH /usergroups/{id}/members`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***
Die eindeutige Kennung für die Benutzergruppe.

`body:` ***Schnur***
Ein Objekt, das angibt, welche Benutzer zugewiesen oder welche Zuweisung aufgehoben werden sollen. Jeder Schlüssel muss ein Benutzername sein und jeder Wert muss entweder „Mitglied“ oder Null sein. Zum Beispiel weist {"Alice“: „member“, „Bob“: null} Alice der Gruppe zu und trennt Bob von der Gruppe.

`PUT /usergroups/{id}/members`

Geben Sie die folgenden Parameter an.

`id:` ***Schnur***
Die eindeutige Kennung für die Benutzergruppe.

`body:` ***Schnur***
Ein Objekt, das angibt, welche Benutzer der Gruppe zugewiesen sind. Jeder Schlüssel muss ein Benutzername sein und jeder Wert muss „Mitglied“ sein. Zum Beispiel weist {"Alice“: „member“, „Bob“: „member"} Alice und Bob als einzige Mitglieder der Gruppe zu.

VLAN

Virtuelle LANs sind logische Gruppierungen von Datenverkehr oder Geräten im Netzwerk.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
Holen Sie sich /vlans	Alle VLANs abrufen

Betrieb	Beschreibung
GET /vlans/ {id}	Rufen Sie ein bestimmtes VLAN ab.
PATCH /vlans/ {id}	Aktualisieren Sie ein bestimmtes VLAN.

Einzelheiten der Operation

`GET /vlans`

Für diesen Vorgang gibt es keine Parameter.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "description": "string",
    "id": 0,
    "mod_time": 0,
    "name": "string",
    "network_id": 0,
    "node_id": 0,
    "vlanid": 0
}
```

`GET /vlans/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für das VLAN.

Wenn die Anfrage erfolgreich ist, gibt das ExtraHop-System ein Objekt im folgenden Format zurück.

```
{
    "description": "string",
    "id": 0,
    "mod_time": 0,
    "name": "string",
    "network_id": 0,
    "node_id": 0,
    "vlanid": 0
}
```

`PATCH /vlans/{id}`

Geben Sie die folgenden Parameter an.

`body:` ***Objekt***
Wenden Sie die angegebenen Eigenschaftswertaktualisierungen auf das VLAN an.

`id:` ***Zahl***
Die eindeutige Kennung für das VLAN.

Beobachtungsliste

Um sicherzustellen, dass für ein Asset, z. B. ein wichtiger Server, eine Datenbank oder ein Laptop, die erweiterte Analyse garantiert ist, können Sie dieses Gerät zur Beobachtungsliste hinzufügen.

Hinweis: Wenn Sie der Beobachtungsliste mehrere Geräte hinzufügen möchten, sollten Sie in Erwägung ziehen, eine Gerätegruppe zu erstellen und diese Gruppe dann für Erweiterte Analyse zu priorisieren.

Hier sind wichtige Überlegungen zur Beobachtungsliste:

- Die Beobachtungsliste gilt nur für Erweiterte Analyse.
- Die Beobachtungsliste kann so viele Geräte enthalten, wie es die Erweiterte Analyse Analysis-Kapazität zulässt, die durch Ihre Lizenz bestimmt wird.
- Ein Gerät bleibt auf der Beobachtungsliste, unabhängig davon, ob es inaktiv oder aktiv ist. Damit das ExtraHop-System Erweiterte Analyse Analysis-Metriken erfassen kann, muss ein Gerät aktiv sein.

Weitere Informationen zu Erweiterte Analyse finden Sie unter Analysestufen.

In der folgenden Tabelle sind alle Operationen aufgeführt, die Sie mit dieser Ressource ausführen können:

Betrieb	Beschreibung
/watchlist/device/ {id} LÖSCHEN	Entferne ein Gerät von der Beobachtungsliste.
POST /watchlist/device/ {id}	Fügen Sie ein Gerät zur Beobachtungsliste.
GET /watchlist/devices	Rufen Sie alle Geräte ab, die sich in der Beobachtungsliste befinden.
POST /watchliste/devices	Geräte zur Beobachtungsliste hinzufügen oder daraus entfernen.

Einzelheiten der Operation

`GET /watchlist/devices`

Für diesen Vorgang gibt es keine Parameter.

`POST /watchlist/device/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für das Gerät.

`DELETE /watchlist/device/{id}`

Geben Sie die folgenden Parameter an.

`id:` ***Zahl***
Die eindeutige Kennung für das Gerät.

`POST /watchlist/devices`

Geben Sie die folgenden Parameter an.

`assignments:` ***Objekt***
Eine Liste von Geräten, die zur Beobachtungsliste hinzugefügt oder daraus entfernt werden sollen.
`assign:` ***Reihe von Zahlen***
IDs der zuzuweisenden Ressourcen
`unassign:` ***Reihe von Zahlen***
IDs der Ressourcen, deren Zuweisung aufgehoben werden soll

Geben Sie den Zuweisungsparameter im folgenden JSON-Format an.

```
{
    "assign": [],
    "unassign": []
}
```