Erstellen Sie eine Regel für Erkennungsbenachrichtigungen

Veröffentlicht: 2024-08-08

Erstellen Sie eine Benachrichtigungsregel, wenn Sie eine Benachrichtigung über Entdeckungen erhalten möchten , die bestimmten Kriterien entsprechen.

Video Sehen Sie sich die entsprechende Schulung an: Erkennungsbenachrichtigungen konfigurieren

Wenn eine Erkennung generiert wird, die Ihren Kriterien entspricht, wird eine Benachrichtigung mit Informationen von Erkennungskarte.

Sie können das System so konfigurieren, dass es eine E-Mail an eine Empfängerliste sendet oder einen bestimmten Webhook aufruft.

Bevor Sie beginnen

- Benutzern muss der Zugriff auf das NDR- oder NPM-Modul gewährt werden und sie müssen über vollständige Schreibberechtigungen verfügen Privilegien oder höher, um die Aufgaben in diesem Handbuch abzuschließen.
- RevealX 360 benötigt eine Verbindung zu ExtraHop Cloud Services um Benachrichtigungen per E-Mail und Webhooks zu senden. RevealX Enterprise benötigt eine Verbindung zu ExtraHop Cloud Services, um Benachrichtigungen per E-Mail zu senden, kann aber auch ohne Verbindung eine Benachrichtigung über einen Webhook senden.
- E-Mail-Benachrichtigungen werden über ExtraHop Cloud Services gesendet und können identifizierbare Informationen wie IP-Adressen, Benutzernamen, Hostnamen, Domainnamen, Gerätenamen oder Dateinamen enthalten. RevealX Enterprise-Benutzer, deren behördliche Anforderungen externe Verbindungen verbieten, können Benachrichtigungen mit Webhook-Aufrufen so konfigurieren, dass Benachrichtigungen ohne externe Verbindung gesendet werden.
- RevealX 360 kann keine Webhook-Aufrufe an Endpunkte in Ihrem internen Netzwerk senden. Webhook-Ziele müssen für externen Verkehr geöffnet sein.
- Webhook-Ziele müssen über ein Zertifikat verfügen, das von einer Zertifizierungsstelle (CA) des Mozilla CA Certificate Program signiert wurde. siehe https://wiki.mozilla.org/CA/Included_Certificates für Zertifikate von vertrauenswürdigen öffentlichen CAs.
- RevealX Enterprise muss eine direkte Verbindung zu Webhook-Endpunkten herstellen, um Benachrichtigungen zu senden.
- E-Mail-Benachrichtigungen werden von no-reply@notify.extrahop.com gesendet. Stellen Sie sicher, dass Sie diese Adresse zu Ihrer Liste der zulässigen Absender hinzufügen.

1. Loggen Sie sich in das ExtraHop-System ein über `https://<extrahop-hostname-or-IP-address>`.
2. Klicken Sie auf das Symbol Systemeinstellungen und klicken Sie dann **Regeln für Benachrichtigungen**.
3. Klicken Sie **Erstellen**.
4. In der Name Feld, geben Sie einen eindeutigen Namen für die Benachrichtigungsregel ein.
5. In der Beschreibung Feld, fügen Sie Informationen zur Benachrichtigungsregel hinzu.
6. In der Art der Veranstaltung Abschnitt, auswählen **Sicherheitserkennung** oder **Leistungserkennung**.
7. In der Kriterien Abschnitt, klicken **Kriterien hinzufügen** um Kriterien anzugeben, nach denen eine Benachrichtigung generiert wird.
   - **Mindestrisikobewertung**
   - **Typ**
   - **Kategorie**

- **Technik**
- **Täter**
- **Opfer**
- **Rolle des Geräts**
- **Quelle**
- **Standort**

Die Kriterienoptionen entsprechen den Filteroptionen auf der Seite „Erkennungen“.

8. Klicken Sie im Abschnitt Aktionen auf **Aktion hinzufügen** um festzulegen, wie die Benachrichtigung gesendet wird.
   - Klicken Sie **E-Mail senden** und geben Sie einzelne E-Mail-Adressen an, getrennt durch ein Komma.
   - Klicken Sie **Webhook aufrufen** und geben Sie die folgenden Einstellungen an:
     1. In der Nutzlast-URL Feld, geben Sie die URL des Webhooks ein.
     2. In der Nutzlast (JSON) Feld, geben Sie die JSON-Nutzlast ein, die an die Nutzlast-URL gesendet wird.

        Sehen Sie die Referenz zur Webhook-Benachrichtigung zum Beispiel Nutzlasten.
     3. (Optional) Klicken Sie im Abschnitt Benutzerdefinierte Header auf **Header hinzufügen** um benutzerdefinierte Schlüssel:Wert-Paare anzugeben.

        Benutzerdefinierte Header werden dem Header der Webhook-HTTP-POST-Anforderung hinzugefügt.
     4. Klicken Sie **Speichern**.
     5. Klicken Sie **Verbindung testen**.

        Eine Nachricht mit dem Titel Testbenachrichtigung wird an die Payload-URL gesendet, um die Verbindung zu bestätigen.

        Hinweis Bestätigen Sie nach dem Testen der Verbindung, dass Sie die Benachrichtigung in der Zielanwendung erhalten haben. RevealX Enterprise zeigt eine Fehlermeldung an, wenn die Testbenachrichtigung nicht erfolgreich war.
     6. Wählen Sie einen Authentifizierungstyp aus.
        - **Keine Authentifizierung**
        - **Standardauthentifizierung**

          Geben Sie den Benutzernamen und das Passwort für die Zielanwendung ein.
        - **Inhaber-Token**

          Geben Sie das Zugriffstoken für die Zielanwendung ein.
9. In der Optionen Abschnitt, wählen Sie den **Benachrichtigungsregel aktivieren** Checkbox, um die Benachrichtigung zu aktivieren.

Wenn eine Erkennung den Kriterien entspricht, wird eine Benachrichtigung gesendet. Eine einzelne Erkennung generiert niemals mehr als eine Benachrichtigung pro Benachrichtigungsregel.

Referenz zur Webhook-Benachrichtigung

Dieses Handbuch enthält Referenzinformationen, die Ihnen beim Schreiben der JSON-Payload für Webhook-basierte Benachrichtigungen helfen sollen. Das Handbuch enthält einen Überblick über die Payload-Schnittstelle (JSON), eine Liste von Erkennungsvariablen, die für Webhooks verfügbar sind, und Beispiele für die JSON-Struktur für gängige Webhook-Ziele wie Slack, Microsoft Teams und Google Chat.

Weitere Informationen zu Benachrichtigungsregeln finden Sie unter Erstellen Sie eine Regel für Erkennungsbenachrichtigungen.

Nutzlast JSON

ExtraHop-Webhooks sind in JSON formatiert und werden unterstützt von Jinja2-Vorlagen-Engine. Wenn Sie eine Benachrichtigungsregel erstellen und die Webhook-Option auswählen, wird der Webhook-Editor auf der rechten Seite geöffnet, und Sie können die Payload bearbeiten.

Du kannst die Standard-Payload mit benutzerdefinierten Eigenschaften ändern oder eine JSON-Vorlage für Slack, Microsoft Teams oder Google Chat kopieren, und zwar aus dem Beispiele Abschnitt.

Standardmäßig enthält die Payload ein Beispiel `text` Eigentum. Das JSON-Beispiel in der Abbildung unten sendet eine Benachrichtigung mit dem Text „ExtraHop Erkennung“, gefolgt vom Erkennungstitel, der die Variable ersetzt.

Wir empfehlen, dass Sie Ihre Verbindung zur Webhook-URL testen, bevor Sie die Nutzlast ändern. Auf diese Weise können Sie sicher sein, dass Probleme nicht auf einen Verbindungsfehler zurückzuführen sind.

Syntaxvalidierung

Der Webhook-Editor bietet JSON- und Jinja2-Syntaxvalidierung. Wenn Sie eine Zeile eingeben , die eine falsche JSON- oder Jinja2-Syntax enthält, wird unter dem Feld Payload ein Fehler mit dem Fehler angezeigt.

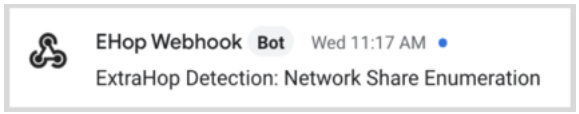
Variablen

Erkennungsvariablen werden der Nutzlast hinzugefügt, indem der Variablenname zwischen doppelten Gruppen geschweifter Klammern ({{und}}) eingefügt wird.

Das Beispiel in der Payload enthält beispielsweise eine Variable für den Erkennungstitel:

```
"text": "ExtraHop Detection: {{title}}"
```

Wenn eine Erkennung einer Benachrichtigungsregel mit der Variablen entspricht, wird die Variable durch den Erkennungstitel ersetzt. Wenn die Benachrichtigungsregel beispielsweise mit der Erkennung für Network Share Enumeration übereinstimmt, wird die Variable durch den Titel in der Benachrichtigung ersetzt, ähnlich der folgenden Abbildung:



Sehen Sie eine Liste von Erkennungsvariablen.

Filter

Filter ermöglichen es Ihnen, eine Variable zu ändern.

JSON übergeben

Wenn die Variable einen Wert zurückgibt, der in JSON formatiert ist, wird der Wert automatisch maskiert und in eine Zeichenfolge übersetzt. Wenn Sie gültiges JSON an Ihr Webhook-Ziel übergeben möchten, müssen Sie Folgendes angeben: `safe` filtern:

```
{{<variable> | safe }}
```

Im folgenden Beispiel gibt die Variable Erkennungsdaten über Teilnehmer im JSON-Format direkt an das Webhook-Ziel zurück:

```
{{api.participants | safe }}
```

IF-Kontoauszüge
Eine IF-Anweisung kann überprüfen, ob ein Wert für die Variable verfügbar ist. Wenn die Variable leer ist, können Sie eine alternative Variable angeben.

```
 {% if {{<variable>}} %}
```

Im folgenden Beispiel prüft die IF-Anweisung, ob ein Wert für die Opfervariable verfügbar ist:

```
{% if victims %}
```

Im folgenden Beispiel prüft die IF-Anweisung, ob ein Tätername verfügbar ist. Wenn es keinen Wert für den Namen des Täters gibt, wird stattdessen der Wert für die Variable IP-Adresse des Täters zurückgegeben.

```
{% if offender.name %}{{offender.name}}{%else%}{{offender.ipaddr}}
{% endif %}
```

FÜR Schleifen
Eine FOR-Schleife kann es der Benachrichtigung ermöglichen, ein Array von Objekten anzuzeigen.

```
{% for <array-object-variable> in <array-variable> %}
```

Im folgenden Beispiel wird eine Liste mit Täternamen aus dem Täter-Array in der Benachrichtigung angezeigt. Eine IF-Anweisung sucht nach weiteren Elementen im Array (`{% if not loop.last %}`) und fügt einen Zeilenumbruch hinzu, bevor der nächste Wert gedruckt wird (`\n`). Wenn ein Tätername leer ist, gibt der Standardfilter „Unbekannter Name“ für den Wert zurück.

```
{% for offender in offenders %}
  {{offender.name | default ("Unknown Name")}}
  {% if not loop.last %}\n
  {% endif %}
{% endfor %}
```

Verfügbare Erkennungsvariablen
Die folgenden Variablen sind für Webhook-Benachrichtigungen über Erkennungen verfügbar.

titel: ***Schnur***

Der Titel der Erkennung.

Erkennung: ***Schnur***

Eine Beschreibung der Erkennung.

typ: ***Schnur***

Die Art der Erkennung.

ID: ***Zahl***

Die eindeutige Kennung für die Erkennung.

URL: ***Schnur***

Die URL für die Erkennung im ExtraHop-System.

risk_score: ***Zahl***

Die Risikoscore der Erkennung.

Standort: ***Schnur***

Die Standort, an der die Erkennung stattgefunden hat.

Startzeit_Text: ***Schnur***

Der Zeitpunkt, zu dem die Erkennung begann.

Endzeit_Text: ***Schnur***

Der Zeitpunkt, zu dem die Erkennung endete.

kategorien_array: ***Reihe von Zeichenketten***

Eine Reihe von Kategorien, zu denen die Erkennung gehört.

Kategorien_Zeichenfolge: ***Schnur***

Eine Zeichenfolge, die die Kategorien auflistet, zu denen die Erkennung gehört.

Mitre_Tactics: ***Reihe von Zeichenketten***

Eine Reihe von MITRE-Taktik-IDs, die mit der Erkennung verknüpft sind.

mitre_tactics_string: ***Schnur***

Eine Zeichenfolge, die die MITRE-Taktik-IDs auflistet, die mit der Erkennung verknüpft sind.

Mitre_Techniken: ***Reihe von Zeichenketten***

Eine Reihe von MITRE-Technik-IDs, die mit der Erkennung verknüpft sind.

mitre_techniques_string: ***Schnur***

Eine Zeichenfolge, die die MITRE-Technik-IDs auflistet, die der Erkennung zugeordnet sind.

primärer Täter: ***Objekt***

Ein Objekt, das den Haupttäter identifiziert und die folgenden Eigenschaften enthält:

extern: ***Boolescher Wert***

Der Wert ist `true` wenn sich die IP-Adresse des primären Täters außerhalb Ihres Netzwerk befindet.

iPaddr: ***Schnur***

Die IP-Adresse des Haupttäters.

name: ***Schnur***

Der Name des Haupttäters.

Täter: ***Reihe von Objekten***

Eine Reihe von Täterobjekten, die mit der Erkennung in Verbindung stehen. Jedes Objekt enthält die folgenden Eigenschaften:

extern: ***Boolescher Wert***

Der Wert ist `true` wenn sich die IP-Adresse des Täters außerhalb Ihres Netzwerk befindet.

iPaddr: ***Schnur***

Die IP-Adresse des Täters. Gilt für Feststellungen mit mehreren Tätern.

name: ***Schnur***

Der Name des Täters. Gilt für Feststellungen mit mehreren Tätern.

primäres Opfer: ***Objekt***

Ein Objekt, das das primäre Opfer identifiziert und die folgenden Eigenschaften enthält:

extern: ***Boolescher Wert***

Der Wert ist `true` wenn die IP-Adresse des primären Opfers außerhalb Ihres Netzwerk liegt.

iPaddr: ***Schnur***

Die IP-Adresse des primären Opfers.

name: ***Schnur***

Der Name des Hauptopfers.

Opfer: ***Reihe von Objekten***

Eine Reihe von Opferobjekten, die mit der Erkennung in Verbindung stehen. Jedes Objekt enthält die folgenden Eigenschaften:

extern: ***Boolescher Wert***

Der Wert ist `true` wenn sich die IP-Adresse des Opfers außerhalb Ihres Netzwerk befindet.

iPaddr: ***Schnur***

Die IP-Adresse des Opfers. Gilt für Erkennungen mit mehreren Opfern.

name: ***Schnur***

Der Name des Opfers. Gilt für Erkennungen mit mehreren Opfern.

api: ***Objekt***

Ein Objekt, das alle Felder enthält, die von `GET /detections/{id}operation`. Weitere Informationen finden Sie in der Einführung in die ExtraHop REST API.

Webhook-Beispiele

Die folgenden Abschnitte enthalten JSON-Vorlagen für gängige Webhook-Ziele.

Slack

Nachdem du eine Slack-App erstellt und eingehende Webhooks für die App aktiviert hast, kannst du einen eingehenden Webhook erstellen. Wenn du einen eingehenden Webhook erstellst, generiert Slack die URL, die du in das Feld Payload-URL in deiner Benachrichtigungsregel eingibst.

Das folgende Beispiel zeigt die JSON-Nutzlast für einen Slack-Webhook:

```
{
    "blocks": [
        {
            "type": "header",
            "text": {
                "type": "plain_text",
                "text": "Detection: {{ title }}"
             }
        },
        {
            "type": "section",
            "text": {
                "type": "mrkdwn",
                "text": "• *Risk Score:* {{ risk_score }}\n • *Category:*
 {{ categories_string }}\n • *Site:* {{ site }}\n • *Primary Offender:*
 {{ offender_primary.name}} ({{ offender_primary.ipaddr}})\n • *Primary
 Victim:* {{ victim_primary.name }} ({{ victim_primary.ipaddr }})\n"
            }
        },
        {
            "type": "section",
            "text": {
                "type": "plain_text",
                "text": "Detection ID: {{ id }}"
            },
            "text": {
                "type": "mrkdwn",
                "text": "<{{ url }}|View Detection Details>"
            }
        }
    ]
}
```

Microsoft-Teams

Du kannst einem Teams-Kanal einen eingehenden Webhook als Connector hinzufügen. Nachdem Sie einen eingehenden Webhook konfiguriert haben, generiert Teams die URL, die Sie in das Feld Payload-URL in Ihrer Benachrichtigungsregel eingeben müssen.

Das folgende Beispiel zeigt die JSON-Nutzlast für einen Microsoft Teams-Webhook:

```
{
   "type":"message",
   "attachments":[
      {
         "contentType":"application/vnd.microsoft.card.adaptive",
         "contentUrl":null,
         "content":{
            "$schema":"https://adaptivecards.io/schemas/adaptive-card.json",
            "type":"AdaptiveCard",
            "body":[
               {
                  "type":"ColumnSet",
                  "columns":[
                     {
                        "type": "Column",
                        "width":"16px",
                        "items":[
                           {
                              "type":"Image",
                              "horizontalAlignment":"center",
                              "url":"https://assets.extrahop.com/
favicon.ico",
                              "altText":"ExtraHop Logo"
                           }
                        ]
                     },
                     {
                        "type": "Column",
                        "width":"stretch",
                        "items":[
                           {
                              "type":"TextBlock",
                              "text":"ExtraHop RevealX",
                              "weight": "bolder"
                           }
                        ]
                     }
                  ]
               },
               {
                  "type":"TextBlock",
                  "text":"**{{ title }}**"
               },
               {
                  "type":"TextBlock",
                  "spacing":"small",
                  "isSubtle":true,
                  "wrap":true,
                  "text":"{{ description }}"
               },
               {
                  "type":"FactSet",
                  "facts":[
                     {
                        "title":"Risk Score:",
                        "value":"{{ risk_score }}"
```

```
                    },
                    {
                       "title":"Category:",
                       "value":"{{ categories_string }}"
                    },
                    {
                       "title":"Site:",
                       "value":"{{ site }}"
                    },
                    {
                       "title":"Primary Offender:",
                       "value":"{{ offender_primary.name }}
 ({{ offender_primary.ipaddr }})"
                    },
                    {
                       "title":"Primary Victim:",
                       "value":"{{ victim_primary.name }}
 ({{ victim_primary.ipaddr }})"
                    }
                 ]
              },
              {
                 "type":"ActionSet",
                 "actions":[
                    {
                       "type":"Action.OpenUrl",
                       "title":"View Detection Details",
                       "url":"{{ url }}"
                    }
                 ]
              }
           ]
        }
     }
  ]
}
```

Google Chat

In einem Google-Chatroom können Sie auf das Drop-down-Menü neben dem Raumnamen klicken und Webhooks verwalten auswählen. Nachdem Sie einen Webhook hinzugefügt und ihm einen Namen gegeben haben, generiert Google Chat die URL, die Sie in das Feld Payload-URL in Ihrer Benachrichtigungsregel eingeben müssen.

Das folgende Beispiel zeigt die JSON-Nutzlast für einen Google Chat-Webhook:

```
{
    "cards": [
        {
            "header": {
                "title": "{{title}}"
            },
            "sections": [
                {
                    "widgets": [
                        {
                            "keyValue": {
                                "topLabel": "Risk score",
                                "content": "{{risk_score}}"
                            }
                        },
                        {
```

```
                            "keyValue": {
                                "topLabel": "Categories",
                                "content": "{{categories_string}}"
                            }
                        }
                        {% if offenders %}
                        ,{
                            "keyValue": {
                                "topLabel": "Offenders",
                                "contentMultiline": "true",
                                "content": "{% for offender in offenders %}
{% if offender.name %}{{offender.name}}{% else %}{{offender.ipaddr}}{% endif
 %}{% if not loop.last %}\n{% endif %}{% endfor %}"
                            }
                        }
                        {% endif %}
                        {% if victims %}
                        ,{
                            "keyValue": {
                                "topLabel": "Victims",
                                "contentMultiline": "true",
                                "content": "{% for victim in victims %}{%
 if victim.name %}{{victim.name}}{% else %}{{victim.ipaddr}}{% endif %}{% if
 not loop.last %}\n{% endif %}{% endfor %}"
                            }
                        }
                        {% endif %}
                    ]
                },
                {
                    "widgets": [
                        {
                            "buttons": [
                                {
                                    "textButton": {
                                        "text": "VIEW DETECTION DETAILS",
                                        "onClick": {
                                            "openLink": {
                                                "url": "{{url}}"
                                            }
                                        }
                                    }
                                }
                            ]
                        }
                    ]
                }
            ]
        }
    ]
}
```