Analyze System Health charts to assess trigger performance

Published: 2020-02-24

Triggers are a powerful tool that can provide detailed insight about your environment. However, triggers consume resources and affect system performance, which is why you must monitor and assess the impact of triggers on your ExtraHop appliance through system health tools.

In this walkthrough, you will learn how to create a bad trigger, evaluate the negative performance impact with system health tools, and then correct the bad trigger. You will also learn how to create a dashboard to continue monitoring trigger performance.

The tasks in this walkthrough will help you answer the following questions about the impact of triggers on the ExtraHop system:

- Has my new trigger resulted in an exception error?
- How many exceptions errors have occurred?
- What is the performance impact of the my new trigger?

Prerequisites

- You must have access to an ExtraHop Discover appliance with a user account that has limited write or full write privileges.
- Your ExtraHop system must have SMTP traffic.
- Familiarize yourself with the concepts in this walkthrough by reading the System health and Triggers sections in the ExtraHop Web UI Guide.
- Familiarize yourself with the processes of creating triggers and dashboards by completing the Trigger Walkthrough and the Dashboard Walkthrough.

Create a trigger with exceptions

In this procedure, you will create a simple trigger that logs the processing time of SMTP responses. You will introduce a deliberate error into the trigger configuration to ensure that a trigger exception occurs.

1. Click the System Settings icon, and then click **Triggers**.
2. Click **Create**.
3. In the Name field, type `Track Processing Time`.
4. Click **Enable debug log**.
5. Click the **Events** field, and then add the following events to the trigger configuration:
   - SMTP_REQUEST
   - SMTP_RESPONSE
   - SMPP_RESPONSE
6. Copy and paste the following code into the right pane:

```
var proto;
switch(event) {
    case 'SMTP_REQUEST':
    case 'SMTP_RESPONSE':
        proto = SMTP;
        break;
    case 'SMPP_RESPONSE':
        proto = SMPP;
```

```
        break;
}

if (!proto || !proto.processingTime) {
    debug('Processing Time = ' + proto.processingTime + " on " + event);
}
```

7. Click Show Advanced Options, and then select **Assign to all devices**.
8. Click **Save**.
   A confirmation message appears that states that the trigger script contains errors. Ignore the message for the purposes of this walkthrough.
9. Click **Save Trigger**.

Next steps
Let the trigger run for at least ten minutes, and then check the System Health page.

Tip: Always check trigger performance charts on the System Health page after you create a new trigger or modify an existing one. By only checking trigger results, such as metrics on a dashboard or record queries, you might miss the full picture. For example, a trigger might appear to collect metrics as expected, but might also be consuming a large amount of resources, which could block the trigger queue and lead to triggers getting dropped from the queue.

Review trigger charts on the System Health page

The System Health page contains charts that pertain to the health and performance of ExtraHop system components and services. In this procedure, you will consult trigger performance charts on the System Health page to check the impact of the trigger you created in the previous section.

Note: The performance results reported for the example trigger on your system will differ from the results displayed in this section.

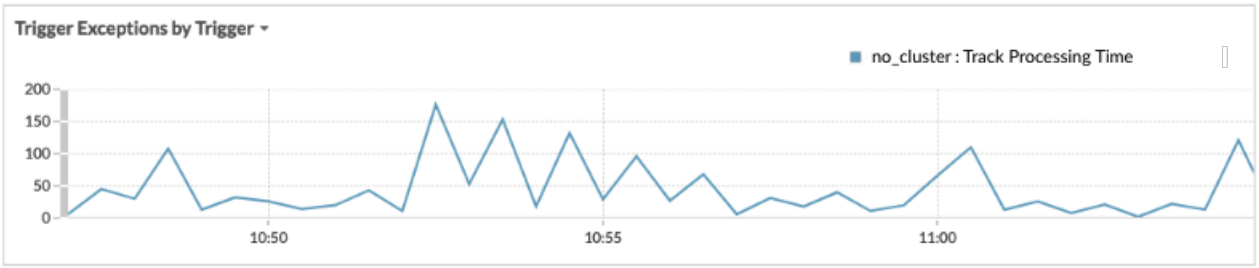
1. From the Triggers page, click **Settings** in the upper-left corner, and then click **System Health**.
2. Scroll down to the Triggers region of the dashboard and locate the Trigger Details chart.
   This chart lists the most active triggers on your system along with the cycles, executes, and exceptions associated with those triggers.

Trigger Details

Trigger	Trigger Cycles ↓	Trigger Executes	Trigger Exceptions
no_cluster: HTTP	13,961,591,464	293,620	0
no_cluster: System	13,486,481,553	587,356	0
no_cluster: Protocols	721,440,475	293,620	0
no_cluster:Track Processing Time	253,942,450	4,872	2,436

3. Locate your Track Processing Time trigger and look at the following information:
   a) Compare the values in the Trigger Executes and Trigger Exceptions columns. This information reveals that half of the time the trigger runs, an exception occurs. Because this trigger does not run very often, the impact is not critical. However, if the trigger were modified to run on a popular event such as HTTP, the impact could be extreme.
   b) Compare the value in the Trigger Cycles column with the same value for other triggers running on your system. This information reveals that the average number of cycles consumed by the trigger is relatively low compared to other running triggers. High cycle consumption can indicate that a trigger script is not efficient and might be prone to stall, causing triggers in the queue to back up and drop from the queue.

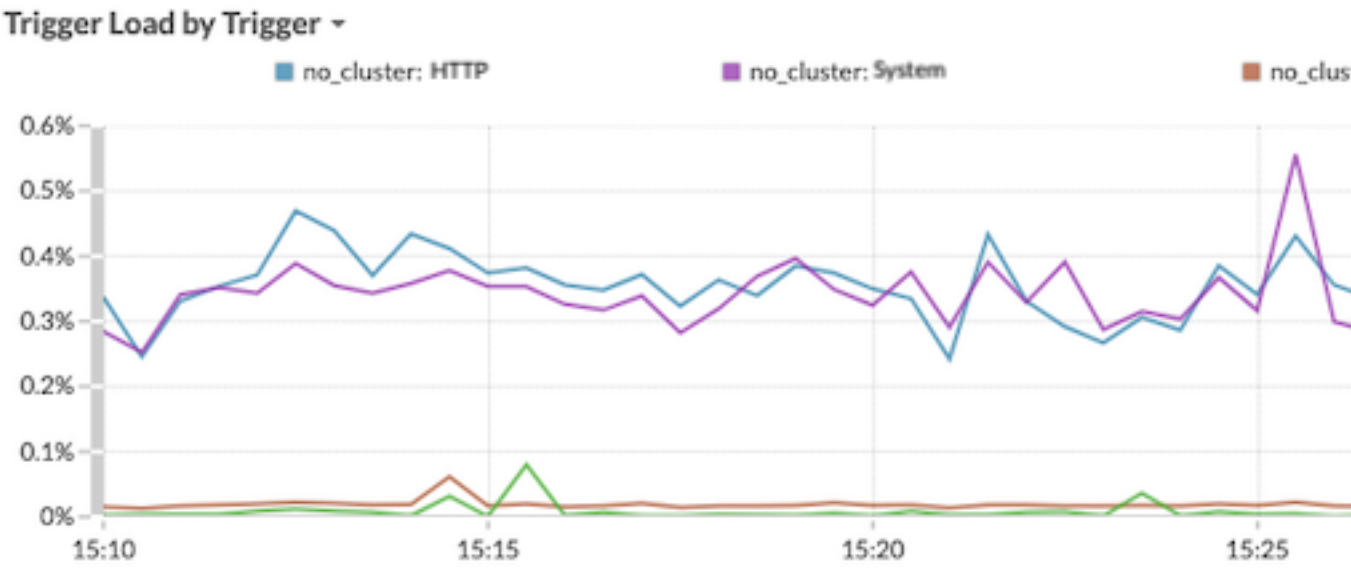
4. Scroll down the System Health dashboard and locate the Trigger Exceptions by Trigger chart. The chart displays the Track Processing Time trigger you created, similar to the following figure:



This chart displays which triggers have exceptions and the number of exceptions generated in the specified time range. Timestamps can help you locate exception error messages in the debug log.

Tip: To highlight a single line in the chart, click the name of the trigger, and then select Hold Focus from the drop-down menu.

5. Scroll up on the System Health dashboard and locate the Trigger Load by Trigger chart. This chart lists the percentage of cycles on the ExtraHop Discover appliance that are being consumed by each trigger.



This data helps you identify if there are increases in resource consumption, if the trigger is running more often than others, and if trigger exceptions are occurring. It is important to check the trigger load for consistent surges in resource consumption, especially if consumption is close to the maximum amount of memory available for running triggers. If the amount of trigger memory is low, you might not be able to run new triggers.

Fix the trigger and view results on the System Health page

In this procedure, you will view exceptions in the trigger debug log that identify where the problem occurs in the trigger script, and then you will resolve the error.

1. In the upper right corner of the window, click the System Settings icon and then select **Triggers**.
2. On the Triggers page, click **Track Processing Time** to open the trigger.
3. Click the **Debug Log** tab.
   In this walkthrough, the debug log displays output similar to the following figure:

```
PROBLEMS 0 0    DEBUG LOG
[Tue Jun 18 13:16:09]  Line 11: Uncaught Error: Action is not valid on event SMTP_REQUEST
[Tue Jun 18 13:16:09]  Line 11: Uncaught Error: Action is not valid on event SMTP_REQUEST
[Tue Jun 18 13:16:29]  Processing Time = NaN on SMTP_RESPONSE
[Tue Jun 18 13:16:49]  Line 11: Uncaught Error: Action is not valid on event SMTP_REQUEST
[Tue Jun 18 13:16:56]  Line 11: Uncaught Error: Action is not valid on event SMTP_REQUEST
```

4. Scroll through the log and look for entries flagged as Uncaught Error. Each error message includes the timestamp when the error occurred, the line number in the script that resulted in the error, and a description of the error.
   You should see the following error message in the log:

   ```
   Line 12: Uncaught Error: Action is not valid on event SMTP_REQUEST.
   ```

   Tip: In addition to exception errors, the debug log also displays uncaught syntax errors, such as an unexpected curly brace, or a type error, such as an invalid value.

5. Click the **Editor** tab, and then locate line 12 in the script to identify the action that is invalid on SMTP requests. In the following figure, line 12 shows that the action is to access the `processingTime` property on events:

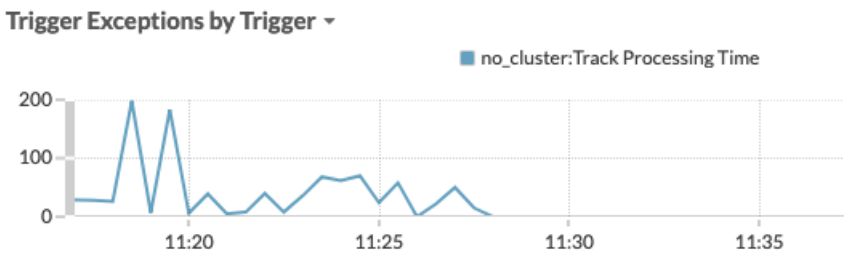
   ```
   12  if (!proto || !proto.processingTime) {
   13      debug('Processing Time = ' + proto.processingTime + " on " + event);
   14  }
   ```

   This information combined with the information from debug log error messages shows that accessing the `processingTime` property is invalid on SMTP request events.

6. Remove the unsupported SMTP event from the script and the trigger configuration by completing the following steps:
   a) Delete the following line from the trigger script:

      ```
      case 'SMTP_REQUEST':
      ```

   b) Click the **Configuration** tab.
   c) Delete SMTP_REQUEST from the Events field.
7. Click **Save and Close**.
   The trigger is saved without displaying a validation error.
8. Click **Settings** in the upper-left corner, and then click **System Health**.
9. Wait 5-10 minutes, and then scroll to the Trigger Exceptions chart that should look similar to the following figure:



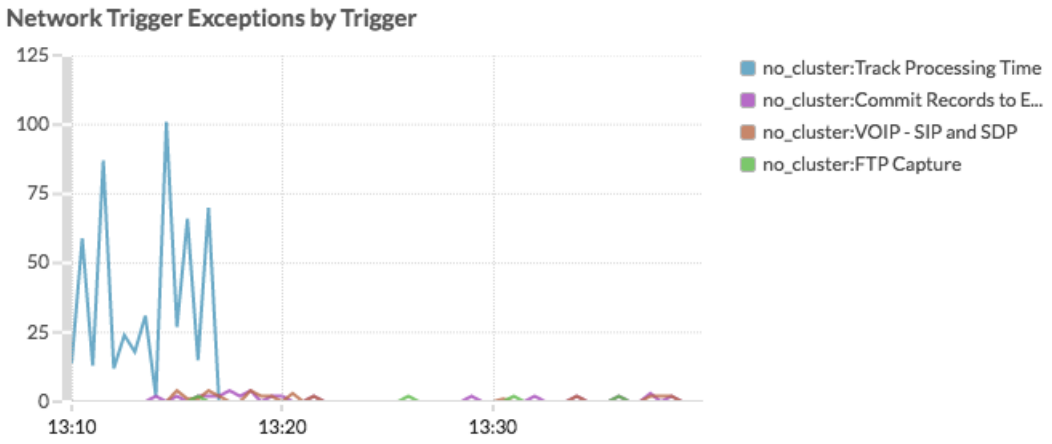
Create a trigger performance dashboard

In this section, you will create a trigger performance dashboard and add several charts discussed in this walkthrough.

Adding system health metrics to a dashboard enables you to customize how you view the data such as choosing the chart type, adding chart notes and tips in text boxes, or adding multiple metrics to a chart.

If you are unfamiliar with creating dashboards, complete the Dashboard Walkthrough. For comprehensive information and procedures for creating and customizing dashboards, see the Dashboards section of the ExtraHop Web UI Guide.

1. Click **Dashboards**.
2. On the Dashboard page, click the command menu ⋮ in the upper-right corner, and select **New Dashboard**.
3. In the Title field, type `Trigger Performance`.
4. Click **Create**.
5. Click the empty chart widget in your newly created dashboard to launch the Metric Explorer.
6. Click **Add Source**.
7. Click the **Any type** drop-down menu and select **Appliances**.
8. From the list, select the name of the appliance you want.
9. In the Metrics field, type `Trigger`, and then select **Capture System Health - Trigger Exceptions by Trigger** from the list.
10. Click **Save** to return to your dashboard.
    The chart should look similar to the following figure:



11. Drag a new chart widget to the region and configure the chart by completing the following steps:
    a) Select the same appliance you specified for the previous chart.
    b) In the Metrics field, type Triggers, and then select **Capture System Health - Trigger Cycles**.
    c) In the Details section, click **None**, and then select **Trigger**.
    d) Click **Save**.
12. Drag a new chart widget to the region and configure the chart by completing the following steps:
    a) Select the same appliance you specified for the previous chart.
    b) In the Metrics field, type triggers, and then select **Capture System Health - Trigger Executes**.
    c) In the Details section, click **None**, and then select **Trigger**.
    d) Click **Save**.
13. Drag a new chart widget to the region and configure the chart by completing the following steps:

a) Select the same appliance you specified for the previous chart.
b) In the Metrics field, type Triggers, and then select **Capture System Health - Trigger Drops**.
c) Click **Save**.

14. Click **Exit Layout Mode** from the upper-right corner.
The dashboard should look similar to the following figure:

Next steps

Tip: As a next step, you can upload the ExtraHealth Bundle to the ExtraHop system, which installs a dashboard that contains a wide variety of system health charts. Customize the ExtraHealth dashboard to suit your needs, or copy the charts you want to a new dashboard. To learn about bundles, see the Bundles section of the ExtraHop Web UI Guide.