Free Questions for SPLK-4001 by ebraindumps

Shared by Weaver on 24-05-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

Where does the Splunk distribution of the OpenTelemetry Collector store the configuration files on Linux machines by default?

Options:

A- /opt/splunk/

B- /etc/otel/collector/

C- /etc/opentelemetry/

D- /etc/system/default/

Answer:

B

Explanation:

The correct answer is B. /etc/otel/collector/

According to the web search results, the Splunk distribution of the OpenTelemetry Collector stores the configuration files on Linux machines in the /etc/otel/collector/ directory by default. You can verify this by looking at the first result1, which explains how to install the Collector for Linux manually. It also provides the locations of the default configuration file, the agent configuration file, and the gateway configuration file.

To learn more about how to install and configure the Splunk distribution of the OpenTelemetry Collector, you can refer to this documentation2.

1: https://docs.splunk.com/Observability/gdi/opentelemetry/install-linux-manual.html 2: https://docs.splunk.com/Observability/gdi/opentelemetry.html

Question 2

Question Type: MultipleChoice

An SRE creates a new detector to receive an alert when server latency is higher than 260 milliseconds. Latency below 260 milliseconds is healthy for their service. The SRE creates a New Detector with a Custom Metrics Alert Rule for latency and sets a Static Threshold alert condition at 260ms.

How can the number of alerts be reduced?

Options:

A- Adjust the threshold.

B- Adjust the Trigger sensitivity. Duration set to 1 minute.

C- Adjust the notification sensitivity. Duration set to 1 minute.

D- Choose another signal.

Answer:

B

Explanation:

According to the Splunk O11y Cloud Certified Metrics User Track document1, trigger sensitivity is a setting that determines how long a signal must remain above or below a threshold before an alert is triggered. By default, trigger sensitivity is set to Immediate, which means that an alert is triggered as soon as the signal crosses the threshold. This can result in a lot of alerts, especially if the signal fluctuates frequently around the threshold value. To reduce the number of alerts, you can adjust the trigger sensitivity to a longer duration, such as 1 minute, 5 minutes, or 15 minutes. This means that an alert is only triggered if the signal stays above or below the threshold for the specified duration. This can help filter out noise and focus on more persistent issues.

Question 3

Question Type: MultipleChoice

When writing a detector with a large number of MTS, such as memory. free in a deployment with 30,000 hosts, it is possible to exceed the cap of MTS that can be contained in a single plot. Which of the choices below would most likely reduce the number of MTS below the plot cap?

Options:

A- Select the Sharded option when creating the plot.

B- Add a filter to narrow the scope of the measurement.

C- Add a restricted scope adjustment to the plot.

D- When creating the plot, add a discriminator.

Answer:

B

Explanation:

The correct answer is B. Add a filter to narrow the scope of the measurement.

A filter is a way to reduce the number of metric time series (MTS) that are displayed on a chart or used in a detector. A filter specifies one or more dimensions and values that the MTS must have in order to be included. For example, if you want to monitor the memory.free metric only for hosts that belong to a certain cluster, you can add a filter like cluster:my-cluster to the plot or detector. This will exclude any MTS that do not have the cluster dimension or have a different value for it1

Adding a filter can help you avoid exceeding the plot cap, which is the maximum number of MTS that can be contained in a single plot. The plot cap is 100,000 by default, but it can be changed by contacting Splunk Support2

To learn more about how to use filters in Splunk Observability Cloud, you can refer to this documentation3.

1: https://docs.splunk.com/Observability/gdi/metrics/search.html#Filter-metrics 2: https://docs.splunk.com/Observability/gdi/metrics/detectors.html#Plot-cap 3: https://docs.splunk.com/Observability/gdi/metrics/search.html

Question 4

Question Type: MultipleChoice

Which of the following are correct ports for the specified components in the OpenTelemetry Collector?

Options:

A- gRPC (4000), SignalFx (9943), Fluentd (6060)

B- gRPC (6831), SignalFx (4317), Fluentd (9080)

C- gRPC (4459), SignalFx (9166), Fluentd (8956)

D- gRPC (4317), SignalFx (9080), Fluentd (8006)

Answer:

D

Explanation:

The correct answer is D. gRPC (4317), SignalFx (9080), Fluentd (8006).

According to the web search results, these are the default ports for the corresponding components in the OpenTelemetry Collector. You can verify this by looking at the table of exposed ports and endpoints in the first result1. You can also see the agent and gateway configuration files in the same result for more details.

1: https://docs.splunk.com/observability/gdi/opentelemetry/exposed-endpoints.html

Question 5

Question Type: MultipleChoice

A customer operates a caching web proxy. They want to calculate the cache hit rate for their service. What is the best way to achieve this?

Options:

A- Percentages and ratios

B- Timeshift and Bottom N

C- Timeshift and Top N

D- Chart Options and metadata

Answer:

A

Explanation:

According to the Splunk O11y Cloud Certified Metrics User Track document1, percentages and ratios are useful for calculating the proportion of one metric to another, such as cache hits to cache misses, or successful requests to failed requests. You can use the percentage() or ratio() functions in SignalFlow to compute these values and display them in charts. For example, to calculate the cache hit rate for a service, you can use the following SignalFlow code:

```
percentage(counters("cache.hits"), counters("cache.misses"))
```

This will return the percentage of cache hits out of the total number of cache attempts. You can also use the ratio() function to get the same result, but as a decimal value instead of a percentage.

ratio(counters("cache.hits"), counters("cache.misses"))

Question 6

Question Type: MultipleChoice

To refine a search for a metric a customer types host: test-*. What does this filter return?

Options:

A- Only metrics with a dimension of host and a value beginning with test-.

B- Error

C- Every metric except those with a dimension of host and a value equal to test.

D- Only metrics with a value of test- beginning with host.

Answer:

A

Explanation:

The correct answer is A. Only metrics with a dimension of host and a value beginning with test-.

This filter returns the metrics that have a host dimension that matches the pattern test-. For example, test-01, test-abc, test-xyz, etc. The asterisk () is a wildcard character that can match any string of characters1

To learn more about how to filter metrics in Splunk Observability Cloud, you can refer to this documentation2.

1: https://docs.splunk.com/Observability/gdi/metrics/search.html#Filter-metrics 2: https://docs.splunk.com/Observability/gdi/metrics/search.html

Question 7

Question Type: MultipleChoice

One server in a customer's data center is regularly restarting due to power supply issues. What type of dashboard could be used to view charts and create detectors for this server?

Options:

A- Single-instance dashboard

B- Machine dashboard

C- Multiple-service dashboard

D- Server dashboard

Answer:

A

Explanation:

According to the Splunk O11y Cloud Certified Metrics User Track document1, a single-instance dashboard is a type of dashboard that displays charts and information for a single instance of a service or host. You can use a single-instance dashboard to monitor the performance and health of a specific server, such as the one that is restarting due to power supply issues. You can also create detectors for the metrics that are relevant to the server, such as CPU usage, memory usage, disk usage, and uptime. Therefore, option A is correct.

Question 8

Question Type: MultipleChoice

A Software Engineer is troubleshooting an issue with memory utilization in their application. They released a new canary version to production and now want to determine if the average memory usage is lower for requests with the 'canary' version dimension. They've already opened the graph of memory utilization for their service.

How does the engineer see if the new release lowered average memory utilization?

Options:

A- On the chart for plot A, select Add Analytics, then select MeanrTransformation. In the window that appears, select 'version' from the Group By field.

B- On the chart for plot A, scroll to the end and click Enter Function, then enter 'A/B-l'.

C- On the chart for plot A, select Add Analytics, then select Mean:Aggregation. In the window that appears, select 'version' from the Group By field.

D- On the chart for plot A, click the Compare Means button. In the window that appears, type 'version1.

Answer:

C

Explanation:

The correct answer is C. On the chart for plot A, select Add Analytics, then select Mean:Aggregation. In the window that appears, select 'version' from the Group By field.

This will create a new plot B that shows the average memory utilization for each version of the application. The engineer can then compare the values of plot B for the 'canary' and 'stable' versions to see if there is a significant difference.

To learn more about how to use analytics functions in Splunk Observability Cloud, you can refer to this documentation1.

1: https://docs.splunk.com/Observability/gdi/metrics/analytics.html

Question 9

Question Type: MultipleChoice

Which of the following are supported rollup functions in Splunk Observability Cloud?

Options:

A- average, latest, lag, min, max, sum, rate

B- std_dev, mean, median, mode, min, max

C- sigma, epsilon, pi, omega, beta, tau

D- 1min, 5min, 10min, 15min, 30min

Answer:

A

Explanation:

According to the Splunk O11y Cloud Certified Metrics User Track document1, Observability Cloud has the following rollup functions: Sum: (default for counter metrics): Returns the sum of all data points in the MTS reporting interval. Average (default for gauge metrics): Returns the average value of all data points in the MTS reporting interval. Min: Returns the minimum data point value seen in the MTS reporting interval. Max: Returns the maximum data point value seen in the MTS reporting interval. Latest: Returns the most recent data point value seen in the MTS reporting interval. Lag: Returns the difference between the most recent and the previous data point values seen in the MTS reporting interval. Rate: Returns the rate of change of data points in the MTS reporting interval. Therefore, option A is correct.