Splunk

Exam Questions SPLK-4001

Splunk O11y Cloud Certified Metrics User

NEW QUESTION 1
Where does the Splunk distribution of the OpenTelemetry Collector store the configuration files on Linux machines by default?

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

NEW QUESTION 2
Which of the following chart visualization types are unaffected by changing the time picker on a dashboard? (select all that apply)

A. Single Value
B. Heatmap
C. Line
D. List

Answer: AD

Explanation:
The chart visualization types that are unaffected by changing the time picker on a dashboard are:
? Single Value: A single value chart shows the current value of a metric or an expression. It does not depend on the time range of the dashboard, but only on the data resolution and rollup function of the chart1
? List: A list chart shows the values of a metric or an expression for each dimension value in a table format. It does not depend on the time range of the dashboard, but only on the data resolution and rollup function of the chart2
Therefore, the correct answer is A and D.
To learn more about how to use different chart visualization types in Splunk Observability Cloud, you can refer to this documentation3.
1: https://docs.splunk.com/Observability/gdi/metrics/charts.html#Single-value 2:
https://docs.splunk.com/Observability/gdi/metrics/charts.html#List 3: https://docs.splunk.com/Observability/gdi/metrics/charts.html

NEW QUESTION 3
What Pod conditions does the Analyzer panel in Kubernetes Navigator monitor? (select all that apply)

A. Not Scheduled
B. Unknown
C. Failed
D. Pending

Answer: ABCD

Explanation:
The Pod conditions that the Analyzer panel in Kubernetes Navigator monitors are:
? Not Scheduled: This condition indicates that the Pod has not been assigned to a Node yet. This could be due to insufficient resources, node affinity, or other scheduling constraints1
? Unknown: This condition indicates that the Pod status could not be obtained or is not known by the system. This could be due to communication errors, node failures, or other unexpected situations1
? Failed: This condition indicates that the Pod has terminated in a failure state. This could be due to errors in the application code, container configuration, or external factors1
? Pending: This condition indicates that the Pod has been accepted by the system, but one or more of its containers has not been created or started yet. This could be due to image pulling, volume mounting, or network issues1
Therefore, the correct answer is A, B, C, and D.
To learn more about how to use the Analyzer panel in Kubernetes Navigator, you can refer to this documentation2.
1: https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/#pod-phase 2: https://docs.splunk.com/observability/infrastructure/monitor/k8s-nav.html#Analyzer-panel

NEW QUESTION 4
With exceptions for transformations or timeshifts, at what resolution do detectors operate?

A. 10 seconds
B. The resolution of the chart
C. The resolution of the dashboard
D. Native resolution

Answer: D

Explanation:
According to the Splunk Observability Cloud documentation1, detectors operate at the native resolution of the metric or dimension that they monitor, with some exceptions for transformations or timeshifts. The native resolution is the frequency at which the data points are reported by the source. For example, if a metric is reported every 10 seconds, the detector will evaluate the metric every 10 seconds. The native resolution ensures that the detector uses the most granular and

accurate data available for alerting.

NEW QUESTION 5
What is one reason a user of Splunk Observability Cloud would want to subscribe to an alert?

A. To determine the root cause of the Issue triggering the detector.
B. To perform transformations on the data used by the detector.
C. To receive an email notification when a detector is triggered.
D. To be able to modify the alert parameters.

Answer: C

Explanation:
One reason a user of Splunk Observability Cloud would want to subscribe to an alert is C. To receive an email notification when a detector is triggered.
A detector is a component of Splunk Observability Cloud that monitors metrics or events and triggers alerts when certain conditions are met. A user can create and configure detectors to suit their monitoring needs and goals1
A subscription is a way for a user to receive notifications when a detector triggers an alert. A user can subscribe to a detector by entering their email address in the Subscription tab of
the detector page. A user can also unsubscribe from a detector at any time2
When a user subscribes to an alert, they will receive an email notification that contains information about the alert, such as the detector name, the alert status, the alert severity, the alert time, and the alert message. The email notification also includes links to view the detector, acknowledge the alert, or unsubscribe from the detector2
To learn more about how to use detectors and subscriptions in Splunk Observability Cloud, you can refer to these documentations12.
1: https://docs.splunk.com/Observability/alerts-detectors-notifications/detectors.html 2: https://docs.splunk.com/Observability/alerts-detectors-notifications/subscribe-to- detectors.html

NEW QUESTION 6
For a high-resolution metric, what is the highest possible native resolution of the metric?

A. 2 seconds
B. 15 seconds
C. 1 second
D. 5 seconds

Answer: C

Explanation:
The correct answer is C. 1 second.
According to the Splunk Test Blueprint - O11y Cloud Metrics User document1, one of the metrics concepts that is covered in the exam is data resolution and rollups. Data resolution refers to the granularity of the metric data points, and rollups are the process of aggregating data points over time to reduce the amount of data stored.
The Splunk O11y Cloud Certified Metrics User Track document2 states that one of the recommended courses for preparing for the exam is Introduction to Splunk Infrastructure Monitoring, which covers the basics of metrics monitoring and visualization.
In the Introduction to Splunk Infrastructure Monitoring course, there is a section on Data Resolution and Rollups, which explains that Splunk Observability Cloud collects high- resolution metrics at 1-second intervals by default, and then applies rollups to reduce the data volume over time. The document also provides a table that shows the different rollup intervals and retention periods for different resolutions.
Therefore, based on these documents, we can conclude that for a high-resolution metric, the highest possible native resolution of the metric is 1 second.

NEW QUESTION 7
To smooth a very spiky cpu.utilization metric, what is the correct analytic function to better see if the cpu. utilization for servers is trending up over time?

A. Rate/Sec
B. Median
C. Mean (by host)
D. Mean (Transformation)

Answer: D

Explanation:
The correct answer is D. Mean (Transformation).
According to the web search results, a mean transformation is an analytic function that returns the average value of a metric or a dimension over a specified time interval1. A mean transformation can be used to smooth a very spiky metric, such as cpu.utilization, by reducing the impact of outliers and noise. A mean transformation can also help to see if the metric is trending up or down over time, by showing the general direction of the average value. For example, to smooth the cpu.utilization metric and see if it is trending up over time, you can use the following SignalFlow code:
mean(1h, counters(“cpu.utilization”))
This will return the average value of the cpu.utilization counter metric for each metric time series (MTS) over the last hour. You can then use a chart to visualize the results and compare the mean values across different MTS.
Option A is incorrect because rate/sec is not an analytic function, but rather a rollup function that returns the rate of change of data points in the MTS reporting interval1. Rate/sec can be used to convert cumulative counter metrics into counter metrics, but it does not smooth or trend a metric. Option B is incorrect because median is not an analytic function, but rather an aggregation function that returns the middle value of a metric or a dimension over the entire time range1. Median can be used to find the typical value of a metric, but it does not smooth or trend a metric. Option C is incorrect because mean (by host) is not an analytic function, but rather an aggregation function that returns the average value of a metric or a dimension across all MTS with the same host dimension1. Mean (by host) can be used to compare the performance of different hosts, but it does not smooth or trend a metric.
Mean (Transformation) is an analytic function that allows you to smooth a very spiky metric by applying a moving average over a specified time window. This can help you see the general trend of the metric over time, without being distracted by the short-term fluctuations1
To use Mean (Transformation) on a cpu.utilization metric, you need to select the metric from the Metric Finder, then click on Add Analytics and choose Mean (Transformation) from the list of functions. You can then specify the time window for the moving average, such as 5 minutes, 15 minutes, or 1 hour. You can also group the metric by host or any other dimension to compare the smoothed values across different servers2
To learn more about how to use Mean (Transformation) and other analytic functions in Splunk Observability Cloud, you can refer to this documentation2.
1: https://docs.splunk.com/Observability/gdi/metrics/analytics.html#Mean-Transformation 2: https://docs.splunk.com/Observability/gdi/metrics/analytics.html

NEW QUESTION 8
A customer is sending data from a machine that is over-utilized. Because of a lack of system resources, datapoints from this machine are often delayed by up to 10 minutes. Which setting can be modified in a detector to prevent alerts from firing before the datapoints arrive?

A. Max Delay
B. Duration
C. Latency
D. Extrapolation Policy

Answer: A

Explanation:
The correct answer is A. Max Delay.
Max Delay is a parameter that specifies the maximum amount of time that the analytics engine can wait for data to arrive for a specific detector. For example, if Max Delay is set to 10 minutes, the detector will wait for only a maximum of 10 minutes even if some data points have not arrived. By default, Max Delay is set to Auto, allowing the analytics engine to determine the appropriate amount of time to wait for data points1
In this case, since the customer knows that the data from the over-utilized machine can be delayed by up to 10 minutes, they can modify the Max Delay setting for the detector to 10 minutes. This will prevent the detector from firing alerts before the data points arrive, and avoid false positives or missing data1
To learn more about how to use Max Delay in Splunk Observability Cloud, you can refer to this documentation1.
1: https://docs.splunk.com/observability/alerts-detectors-notifications/detector- options.html#Max-Delay

NEW QUESTION 9
Which component of the OpenTelemetry Collector allows for the modification of metadata?

A. Processors
B. Pipelines
C. Exporters
D. Receivers

Answer: A

Explanation:
The component of the OpenTelemetry Collector that allows for the modification of metadata is A. Processors.
Processors are components that can modify the telemetry data before sending it to exporters or other components. Processors can perform various transformations on metrics, traces, and logs, such as filtering, adding, deleting, or updating attributes, labels, or resources. Processors can also enrich the telemetry data with additional metadata from various sources, such as Kubernetes, environment variables, or system information1
For example, one of the processors that can modify metadata is the attributes processor. This processor can update, insert, delete, or replace existing attributes on metrics or traces. Attributes are key-value pairs that provide additional information about the telemetry data, such as the service name, the host name, or the span kind2
Another example is the resource processor. This processor can modify resource attributes on metrics or traces. Resource attributes are key-value pairs that describe the entity that produced the telemetry data, such as the cloud provider, the region, or the instance type3 To learn more about how to use processors in the OpenTelemetry Collector, you can refer to this documentation1.
1: https://opentelemetry.io/docs/collector/configuration/#processors 2: https://github.com/open-telemetry/opentelemetry-collector-contrib/tree/main/processor/attributesprocessor 3: https://github.com/open- telemetry/opentelemetry-collector-contrib/tree/main/processor/resourceprocessor

NEW QUESTION 10
When creating a standalone detector, individual rules in it are labeled according to severity. Which of the choices below represents the possible severity levels that can be selected?

A. Info, Warning, Minor, Major, and Emergency.
B. Debug, Warning, Minor, Major, and Critical.
C. Info, Warning, Minor, Major, and Critical.
D. Info, Warning, Minor, Severe, and Critical.

Answer: C

Explanation:
The correct answer is C. Info, Warning, Minor, Major, and Critical.
When creating a standalone detector, you can define one or more rules that specify the alert conditions and the severity level for each rule. The severity level indicates how urgent or important the alert is, and it can also affect the notification settings and the escalation policy for the alert1
Splunk Observability Cloud provides five predefined severity levels that you can choose from when creating a rule: Info, Warning, Minor, Major, and Critical. Each severity level has a different color and icon to help you identify the alert status at a glance. You can also customize the severity levels by changing their names, colors, or icons2
To learn more about how to create standalone detectors and use severity levels in Splunk Observability Cloud, you can refer to these documentations12.
1: https://docs.splunk.com/Observability/alerts-detectors-notifications/detectors.html#Create-a-standalone-detector 2: https://docs.splunk.com/Observability/alerts-detectors-notifications/detector- options.html#Severity-levels

NEW QUESTION 10
.....

Thank You for Trying Our Product

We offer two products:

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questons and Answers in PDF Format

SPLK-4001 Practice Exam Features:

* SPLK-4001 Questions and Answers Updated Frequently

* SPLK-4001 Practice Questions Verified by Expert Senior Certified Staff

* SPLK-4001 Most Realistic Questions that Guarantee you a Pass on Your FirstTry

* SPLK-4001 Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year