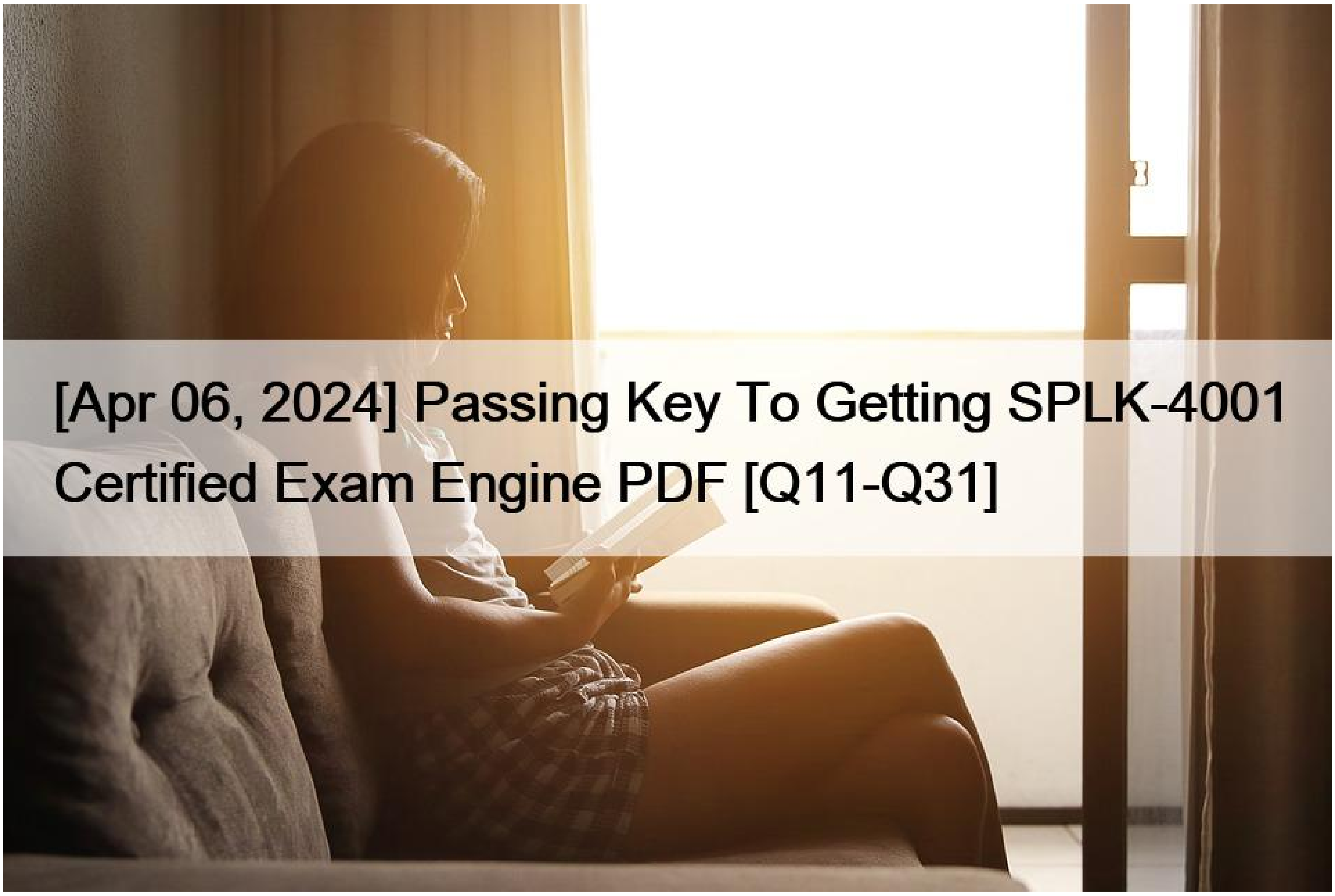
[Apr 06, 2024 Passing Key To Getting SPLK-4001 Certified Exam Engine PDF [Q11-Q31



[Apr 06, 2024] Passing Key To Getting SPLK-4001 Certified Exam Engine PDF
SPLK-4001 Exam Dumps Pass with Updated Apr-2024 Tests Dumps

NO.11 One server in a customer&#8217;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?
* Single-instance dashboard
* Machine dashboard
* Multiple-service dashboard
* Server dashboard
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.

NO.12 A user wants to add a link to an existing dashboard from an alert. When they click the dimension value in the alert message, they are taken to the dashboard keeping the context. How can this be accomplished? (select all that apply)

* Build a global data link.
* Add a link to the Runbook URL.
* Add a link to the field.
* Add the link to the alert message body.

Explanation

The possible ways to add a link to an existing dashboard from an alert are:

Build a global data link. A global data link is a feature that allows you to create a link from any dimension value in any chart or table to a dashboard of your choice. You can specify the source and target dashboards, the dimension name and value, and the query parameters to pass along. When you click on the dimension value in the alert message, you will be taken to the dashboard with the context preserved1 Add a link to the field. A field link is a feature that allows you to create a link from any field value in any search result or alert message to a dashboard of your choice. You can specify the field name and value, the dashboard name and ID, and the query parameters to pass along. When you click on the field value in the alert message, you will be taken to the dashboard with the context preserved2 Therefore, the correct answer is A and C.

To learn more about how to use global data links and field links in Splunk Observability Cloud, you can refer to these documentations12.

1: https://docs.splunk.com/Observability/gdi/metrics/charts.html#Global-data-links 2:

https://docs.splunk.com/Observability/gdi/metrics/search.html#Field-links

NO.13 A DevOps engineer wants to determine if the latency their application experiences is growing fester after a new software release a week ago. They have already created two plot lines, A and B, that represent the current latency and the latency a week ago, respectively. How can the engineer use these two plot lines to determine the rate of change in latency?

* Create a temporary plot by dragging items A and B into the Analytics Explorer window.
* Create a plot C using the formula (A-B) and add a scale:percent function to express the rate of change as a percentage.
* Create a plot C using the formula (A/B-l) and add a scale: 100 function to express the rate of change as a percentage.
* Create a temporary plot by clicking the Change% button in the upper-right corner of the plot showing lines A and B.

Explanation

The correct answer is C. Create a plot C using the formula (A/B-l) and add a scale: 100 function to express the rate of change as a percentage.

To calculate the rate of change in latency, you need to compare the current latency (plot A) with the latency a week ago (plot B). One way to do this is to use the formula (A/B-l), which gives you the ratio of the current latency to the previous latency minus one. This ratio represents how much the current latency has increased or decreased relative to the previous latency. For example, if the current latency is 200 ms and the previous latency is 100 ms, then the ratio is (200/100-l) = 1, which means the current latency is 100% higher than the previous latency1 To express the rate of change as a percentage, you need to multiply the ratio by 100. You can do this by adding a scale: 100 function to the formula. This function scales the values of the plot by a factor of 100. For example, if the ratio is 1, then the scaled value is 100%2 To create a plot C using the formula (A/B-l) and add a scale: 100 function, you need to follow these steps:

Select plot A and plot B from the Metric Finder.

Click on Add Analytics and choose Formula from the list of functions.

In the Formula window, enter (A/B-l) as the formula and click Apply.

Click on Add Analytics again and choose Scale from the list of functions.

In the Scale window, enter 100 as the factor and click Apply.

You should see a new plot C that shows the rate of change in latency as a percentage.

To learn more about how to use formulas and scale functions in Splunk Observability Cloud, you can refer to these documentations34.

1: https://www.mathsisfun.com/numbers/percentage-change.html 2:

https://docs.splunk.com/Observability/gdi/metrics/analytics.html#Scale 3:

https://docs.splunk.com/Observability/gdi/metrics/analytics.html#Formula 4:

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

NO.14 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?
* Rate/Sec
* Median
* Mean (by host)
* Mean (Transformation)
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(&#8220;cpu.utilization&#8221;))
```

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

NO.15 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?
* Info, Warning, Minor, Major, and Emergency.
* Debug, Warning, Minor, Major, and Critical.
* Info, Warning, Minor, Major, and Critical.
* Info, Warning, Minor, Severe, and Critical.
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

NO.16 Where does the Splunk distribution of the OpenTelemetry Collector store the configuration files on Linux machines by default?
* /opt/splunk/
* /etc/otel/collector/
* /etc/opentelemetry/
* /etc/system/default/
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

NO.17 To refine a search for a metric a customer types host: test-*. What does this filter return?
* Only metrics with a dimension of host and a value beginning with test-.
* Error
* Every metric except those with a dimension of host and a value equal to test.
* Only metrics with a value of test- beginning with host.

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

NO.18 The alert recipients tab specifies where notification messages should be sent when alerts are triggered or cleared. Which of the below options can be used? (select all that apply)
* Invoke a webhook URL.
* Export to CSV.
* Send an SMS message.
* Send to email addresses.

Explanation

The alert recipients tab specifies where notification messages should be sent when alerts are triggered or cleared. The options that can be used are:

Invoke a webhook URL. This option allows you to send a HTTP POST request to a custom URL that can perform various actions based on the alert information. For example, you can use a webhook to create a ticket in a service desk system, post a message to a chat channel, or trigger another workflow1 Send an SMS message. This option allows you to send a text message to one or more phone numbers when an alert is triggered or cleared. You can customize the message content and format using variables and templates2 Send to email addresses. This option allows you to send an email notification to one or more recipients when an alert is triggered or cleared. You can customize the email subject, body, and attachments using variables and templates. You can also include information from search results, the search job, and alert triggering in the email3 Therefore, the correct answer is A, C, and D.

1: https://docs.splunk.com/Documentation/Splunk/latest/Alert/Webhooks 2:

https://docs.splunk.com/Documentation/Splunk/latest/Alert/SMSnotification 3:

https://docs.splunk.com/Documentation/Splunk/latest/Alert/Emailnotification

NO.19 How is it possible to create a dashboard group that no one else can edit?

* Ask the admin to lock the dashboard group.
* Restrict the write access on the dashboard group.
* Link the dashboard group to the team.
* Hide the edit menu on the dashboard group.
Explanation

According to the web search results, dashboard groups are a feature of Splunk Observability Cloud that allows you to organize and share dashboards with other users in your organization1. You can set permissions for each dashboard group, such as who can view, edit, or manage the dashboards in the group1. To create a dashboard group that no one else can edit, you need to do the following steps:

Create a dashboard group as usual, by selecting Dashboard Group from the Create menu on the navigation bar, entering a name and description, and adding dashboards to the group1.

Select Alert settings from the Dashboard actions menu () on the top right corner of the dashboard group. This will open a dialog box where you can configure the permissions for the dashboard group1.

Under Write access, select Only me. This will restrict the write access to the dashboard group to yourself only. No one else will be able to edit or delete the dashboards in the group1.

Click Save. This will create a dashboard group that no one else can edit.

NO.20 Which of the following are correct ports for the specified components in the OpenTelemetry Collector?
* gRPC (4000), SignalFx (9943), Fluentd (6060)
* gRPC (6831), SignalFx (4317), Fluentd (9080)
* gRPC (4459), SignalFx (9166), Fluentd (8956)
* gRPC (4317), SignalFx (9080), Fluentd (8006)
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

NO.21 Which of the following is optional, but highly recommended to include in a datapoint?
* Metric name
* Timestamp
* Value
* Metric type
Explanation

The correct answer is D. Metric type.

A metric type is an optional, but highly recommended field that specifies the kind of measurement that a datapoint represents. For example, a metric type can be gauge, counter, cumulative counter, or histogram. A metric type helps Splunk Observability Cloud to interpret and display the data correctly1 To learn more about how to send metrics to Splunk Observability Cloud, you can refer to this documentation2.

1: https://docs.splunk.com/Observability/gdi/metrics/metrics.html#Metric-types 2:

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

NO.22 Which of the following statements about adding properties to MTS are true? (select all that apply)
* Properties can be set via the API.
* Properties are sent in with datapoints.
* Properties are applied to dimension key:value pairs and propagated to all MTS with that dimension
* Properties can be set in the UI under Metric Metadata.

Explanation

According to the web search results, properties are key-value pairs that you can assign to dimensions of existing metric time series (MTS) in Splunk Observability Cloud1. Properties provide additional context and information about the metrics, such as the environment, role, or owner of the dimension. For example, you can add the property use: QA to the host dimension of your metrics to indicate that the host that is sending the data is used for QA.

To add properties to MTS, you can use either the API or the UI. The API allows you to programmatically create, update, delete, and list properties for dimensions using HTTP requests2. The UI allows you to interactively create, edit, and delete properties for dimensions using the Metric Metadata page under Settings3.

Therefore, option A and D are correct.

NO.23 A customer has a very dynamic infrastructure. During every deployment, all existing instances are destroyed, and new ones are created Given this deployment model, how should a detector be created that will not send false notifications of instances being down?
* Create the detector. Select Alert settings, then select Auto-Clear Alerts and enter an appropriate time period.
* Create the detector. Select Alert settings, then select Ephemeral Infrastructure and enter the expected lifetime of an instance.
* Check the Dynamic checkbox when creating the detector.
* Check the Ephemeral checkbox when creating the detector.

Explanation

According to the web search results, ephemeral infrastructure is a term that describes instances that are auto-scaled up or down, or are brought up with new code versions and discarded or recycled when the next code version is deployed1. Splunk Observability Cloud has a feature that allows you to create detectors for ephemeral infrastructure without sending false notifications of instances being down2. To use this feature, you need to do the following steps:

Create the detector as usual, by selecting the metric or dimension that you want to monitor and alert on, and choosing the alert condition and severity level.

Select Alert settings, then select Ephemeral Infrastructure. This will enable a special mode for the detector that will automatically clear alerts for instances that are expected to be terminated.

Enter the expected lifetime of an instance in minutes. This is the maximum amount of time that an instance is expected to live before being replaced by a new one. For example, if your instances are replaced every hour, you can enter 60 minutes as the expected lifetime.

Save the detector and activate it.

With this feature, the detector will only trigger alerts when an instance stops reporting a metric unexpectedly, based on its expected lifetime. If an instance stops reporting a metric within its expected lifetime, the detector will assume that it was terminated on purpose and will not trigger an alert. Therefore, option B is correct.

NO.24 Which of the following can be configured when subscribing to a built-in detector?
* Alerts on team landing page.
* Alerts on a dashboard.
* Outbound notifications.
* Links to a chart.
Explanation

According to the web search results1, subscribing to a built-in detector is a way to receive alerts and notifications from Splunk Observability Cloud when certain criteria are met. A built-in detector is a detector that is automatically created and configured by Splunk Observability Cloud based on the data from your integrations, such as AWS, Kubernetes, or OpenTelemetry1. To subscribe to a built-in detector, you need to do the following steps:

Find the built-in detector that you want to subscribe to. You can use the metric finder or the dashboard groups to locate the built-in detectors that are relevant to your data sources1.

Hover over the built-in detector and click the Subscribe button. This will open a dialog box where you can configure your subscription settings1.

Choose an outbound notification channel from the drop-down menu. This is where you can specify how you want to receive the alert notifications from the built-in detector. You can choose from various channels, such as email, Slack, PagerDuty, webhook, and so on2. You can also create a new notification channel by clicking the + icon2.

Enter the notification details for the selected channel. This may include your email address, Slack channel name, PagerDuty service key, webhook URL, and so on2. You can also customize the notification message with variables and markdown formatting2.

Click Save. This will subscribe you to the built-in detector and send you alert notifications through the chosen channel when the detector triggers or clears an alert.

Therefore, option C is correct.

NO.25 Which of the following are accurate reasons to clone a detector? (select all that apply)
* To modify the rules without affecting the existing detector.
* To reduce the amount of billed TAPM for the detector.
* To add an additional recipient to the detector&#8217;s alerts.
* To explore how a detector was created without risk of changing it.
Explanation

The correct answers are A and D.

According to the Splunk Test Blueprint &#8211; O11y Cloud Metrics User document1, one of the alerting concepts that is covered in the exam is detectors and alerts. Detectors are the objects that define the conditions for generating alerts, and alerts are the notifications that are sent when those conditions are met.

The Splunk O11y Cloud Certified Metrics User Track document2 states that one of the recommended courses for preparing for the exam is Alerting with Detectors, which covers how to create, modify, and manage detectors and alerts.

In the Alerting with Detectors course, there is a section on Cloning Detectors, which explains that cloning a detector creates a copy of the detector with all its settings, rules, and alert recipients. The document also provides some reasons why you might want to clone a detector, such as:

To modify the rules without affecting the existing detector. This can be useful if you want to test different thresholds or conditions before applying them to the original detector.

To explore how a detector was created without risk of changing it. This can be helpful if you want to learn from an existing detector or use it as a template for creating a new one.

Therefore, based on these documents, we can conclude that A and D are accurate reasons to clone a detector.

B and C are not valid reasons because:

Cloning a detector does not reduce the amount of billed TAPM for the detector. TAPM stands for Tracked Active Problem Metric, which is a metric that has been alerted on by a detector. Cloning a detector does not change the number of TAPM that are generated by the original detector or the clone.

Cloning a detector does not add an additional recipient to the detector&#8217;s alerts. Cloning a detector copies the alert recipients from the original detector, but it does not add any new ones. To add an additional recipient to a detector&#8217;s alerts, you need to edit the alert settings of the detector.

NO.26 Which of the following are ways to reduce flapping of a detector? (select all that apply)
* Configure a duration or percent of duration for the alert.
* Establish a reset threshold for the detector.
* Enable the anti-flap setting in the detector options menu.
* Apply a smoothing transformation (like a rolling mean) to the input data for the detector.
Explanation

According to the Splunk Lantern article Resolving flapping detectors in Splunk Infrastructure Monitoring, flapping is a phenomenon where alerts fire and clear repeatedly in a short period of time, due to the signal fluctuating around the threshold value. To reduce flapping, the article suggests the following ways:

Configure a duration or percent of duration for the alert: This means that you require the signal to stay above or below the threshold for a certain amount of time or percentage of time before triggering an alert. This can help filter out noise and focus on more persistent issues.

Apply a smoothing transformation (like a rolling mean) to the input data for the detector: This means that you replace the original signal with the average of its last several values, where you can specify the window length. This can reduce the impact of a single extreme observation and make the signal less fluctuating.

NO.27 What are the best practices for creating detectors? (select all that apply)
* View data at highest resolution.
* Have a consistent value.
* View detector in a chart.
* Have a consistent type of measurement.
Explanation

The best practices for creating detectors are:

View data at highest resolution. This helps to avoid missing important signals or patterns in the data that could indicate anomalies or issues1 Have a consistent value. This means that the metric or dimension used for detection should have a clear and stable meaning across different sources, contexts, and time periods. For example, avoid using metrics that are affected by changes in configuration, sampling, or aggregation2 View detector in a chart. This helps to visualize the data and the detector logic, as well as to identify any false positives or negatives. It also allows to adjust the detector parameters and thresholds based on the data distribution and behavior3 Have a consistent type of measurement. This means that the metric or dimension used for detection should have the same unit and scale across different sources, contexts, and time periods. For example, avoid mixing bytes and bits, or seconds and milliseconds.

1: https://docs.splunk.com/Observability/gdi/metrics/detectors.html#Best-practices-for-detectors 2:

https://docs.splunk.com/Observability/gdi/metrics/detectors.html#Best-practices-for-detectors 3:

https://docs.splunk.com/Observability/gdi/metrics/detectors.html#View-detector-in-a-chart :

https://docs.splunk.com/Observability/gdi/metrics/detectors.html#Best-practices-for-detectors

NO.28 What information is needed to create a detector?
* Alert Status, Alert Criteria, Alert Settings, Alert Message, Alert Recipients
* Alert Signal, Alert Criteria, Alert Settings, Alert Message, Alert Recipients
* Alert Signal, Alert Condition, Alert Settings, Alert Message, Alert Recipients
* Alert Status, Alert Condition, Alert Settings, Alert Meaning, Alert Recipients
Explanation

According to the Splunk Observability Cloud documentation1, to create a detector, you need the following information:

Alert Signal: This is the metric or dimension that you want to monitor and alert on. You can select a signal from a chart or a dashboard, or enter a SignalFlow query to define the signal.

Alert Condition: This is the criteria that determines when an alert is triggered or cleared. You can choose from various built-in alert conditions, such as static threshold, dynamic threshold, outlier, missing data, and so on. You can also specify the severity level and the trigger sensitivity for each alert condition.

Alert Settings: This is the configuration that determines how the detector behaves and interacts with other detectors. You can set the detector name, description, resolution, run lag, max delay, and detector rules. You can also enable or disable the detector, and mute or unmute the alerts.

Alert Message: This is the text that appears in the alert notification and event feed. You can customize the alert message with variables, such as signal name, value, condition, severity, and so on. You can also use markdown formatting to enhance the message appearance.

Alert Recipients: This is the list of destinations where you want to send the alert notifications. You can choose from various channels, such as email, Slack, PagerDuty, webhook, and so on. You can also specify the notification frequency and suppression settings.

NO.29 Which component of the OpenTelemetry Collector allows for the modification of metadata?
* Processors
* Pipelines
* Exporters
* Receivers
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.

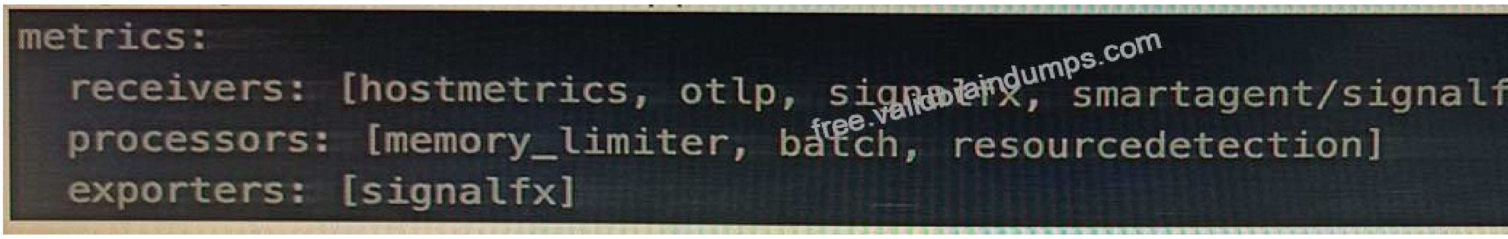
1: https://opentelemetry.io/docs/collector/configuration/#processors 2:

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

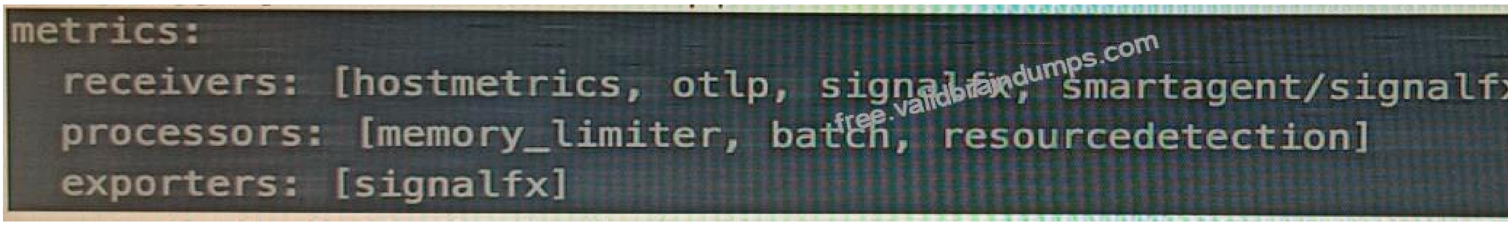
NO.30 A customer is experiencing issues getting metrics from a new receiver they have configured in the OpenTelemetry Collector. How would the customer go about troubleshooting further with the logging exporter?

* Adding debug into the metrics receiver pipeline:



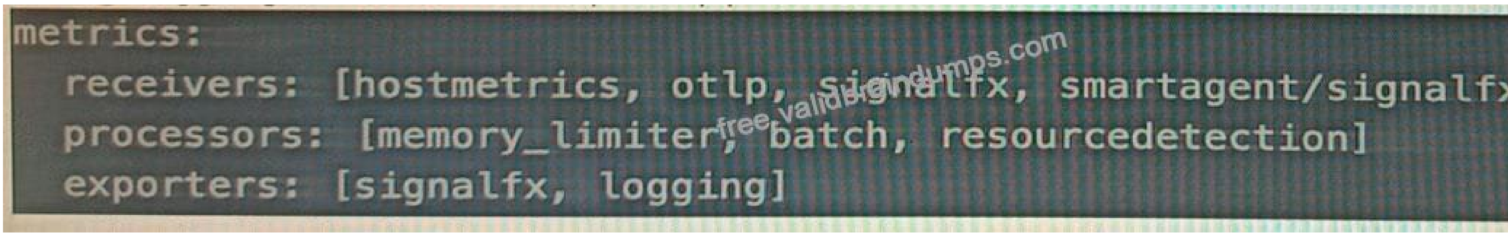
```
metrics:
  receivers: [hostmetrics, otlp, signalfx, smartagent/signalf
  processors: [memory_limiter, batch, resourcedetection]
  exporters: [signalfx]
```

* Adding logging into the metrics receiver pipeline:



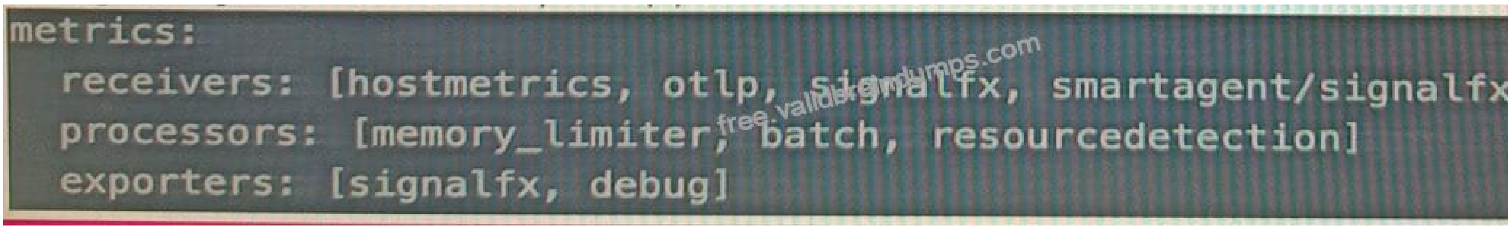
```
metrics:
  receivers: [hostmetrics, otlp, signalfx, smartagent/signalfx
  processors: [memory_limiter, batch, resourcedetection]
  exporters: [signalfx]
```

* Adding logging into the metrics exporter pipeline:



```
metrics:
  receivers: [hostmetrics, otlp, signalfx, smartagent/signalfx
  processors: [memory_limiter, batch, resourcedetection]
  exporters: [signalfx, logging]
```

* Adding debug into the metrics exporter pipeline:



```
metrics:
  receivers: [hostmetrics, otlp, signalfx, smartagent/signalfx
  processors: [memory_limiter, batch, resourcedetection]
  exporters: [signalfx, debug]
```

Explanation

The correct answer is B. Adding logging into the metrics receiver pipeline.

The logging exporter is a component that allows the OpenTelemetry Collector to send traces, metrics, and logs directly to the console. It can be used to diagnose and troubleshoot issues with telemetry received and processed by the Collector, or to obtain samples for other purposes1 To activate the logging exporter, you need to add it to the pipeline that you want to diagnose. In this case, since you are experiencing issues with a new receiver for metrics, you need to add the logging exporter to the metrics receiver pipeline. This will create a new plot that shows the metrics received by the Collector and any errors or warnings that might occur1 The image that you have sent with your question shows how to add the logging exporter to the metrics receiver pipeline. You can see that the exporters section of the metrics pipeline includes logging as one of the options.

This means that the metrics received by any of the receivers listed in the receivers section will be sent to the logging exporter as well as to any other exporters listed2 To learn more about how to use the logging exporter in Splunk Observability Cloud, you can refer to this documentation1.

1: https://docs.splunk.com/Observability/gdi/opentelemetry/components/logging-exporter.html 2:

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

NO.31 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 &#8216;canary&#8217; version dimension. They&#8217;ve already opened the graph of memory utilization for their service.

How does the engineer see if the new release lowered average memory utilization?
* On the chart for plot A, select Add Analytics, then select MeanrTransformation. In the window that appears, select &#8216;version&#8217; from the Group By field.
* On the chart for plot A, scroll to the end and click Enter Function, then enter &#8216;A/B-l&#8217;.
* On the chart for plot A, select Add Analytics, then select Mean:Aggregation. In the window that appears, select &#8216;version&#8217; from the Group By field.
* On the chart for plot A, click the Compare Means button. In the window that appears, type &#8216;version1.
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 &#8216;version&#8217; 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 &#8216;canary&#8217; and &#8216;stable&#8217; 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

The SPLK-4001 exam is intended for experienced IT professionals who have a strong understanding of cloud-based infrastructure and application monitoring. Candidates should have at least six months of experience working with Splunk in a cloud environment, as well as a solid understanding of metrics-based monitoring and analysis. SPLK-4001 exam is designed to validate the skills and knowledge necessary to use Splunk effectively in a cloud-based observability environment.

SPLK-4001 exam questions for practice in 2024 Updated 56 Questions:
https://www.validbraindumps.com/SPLK-4001-exam-prep.html]