Experiment - Creating an alert to monitor an Azure storage account

We can create an alert on multiple available metrics to monitor an Azure storage account. To create an alert, we need to define the trigger condition and the action to be performed when the alert is triggered. In this recipe, we'll create an alert to send an email if the used capacity metrics for an Azure storage account exceed 5 MB. The used capacity threshold of 5 MB is not a standard and is deliberately kept low to explain the alert functionality.

Getting ready

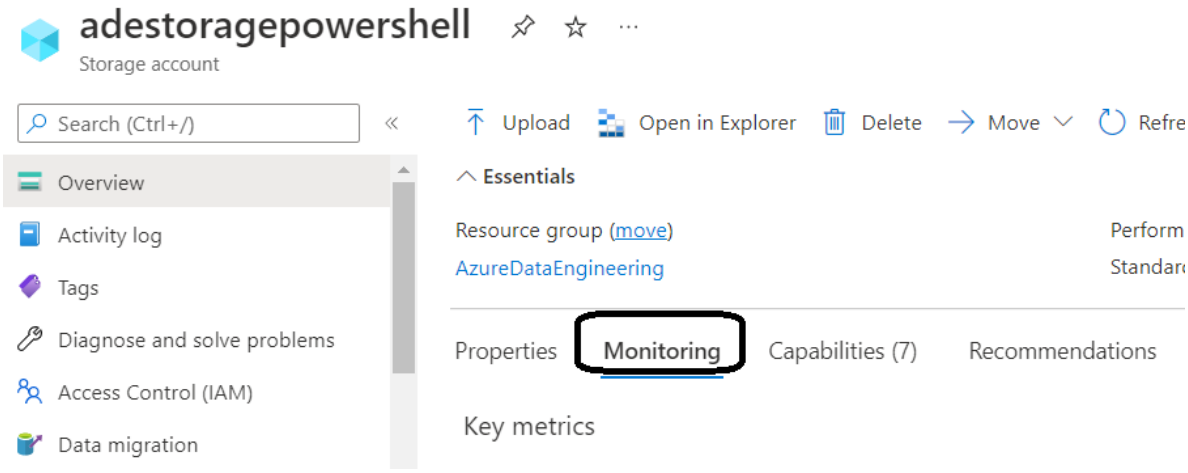
Before you start, perform the following steps: 1. Open a web browser and log in to the Azure portal at **https://portal.azure.com**

2. Make sure you have an existing storage account from our previous experiment (**adestoragepowershell**)

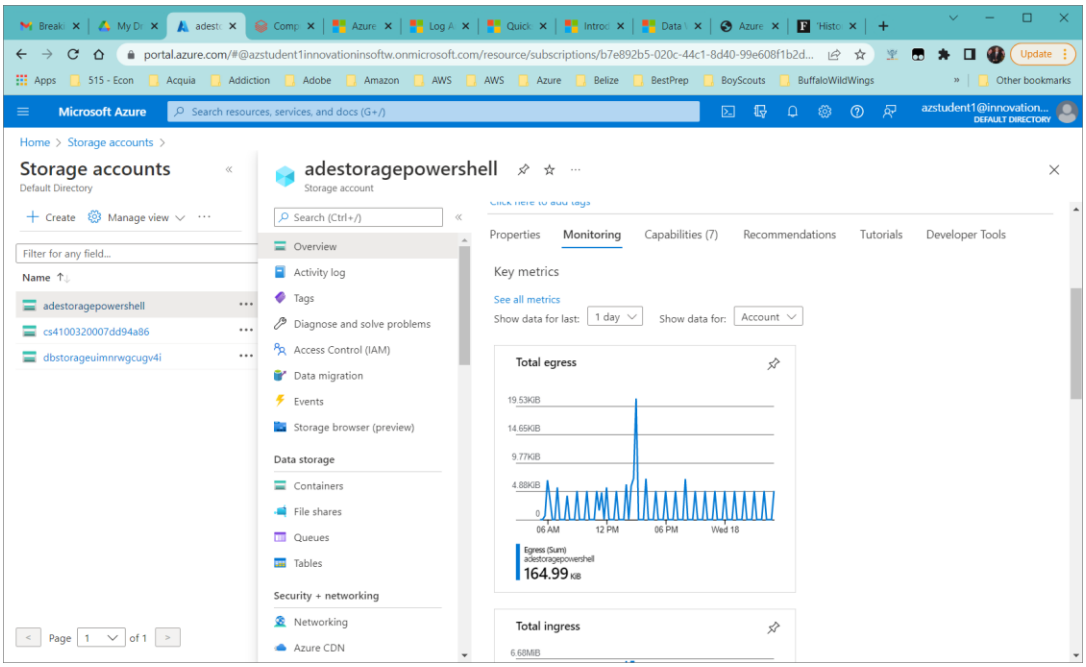
How to do it…

Follow the given steps to create an alert:

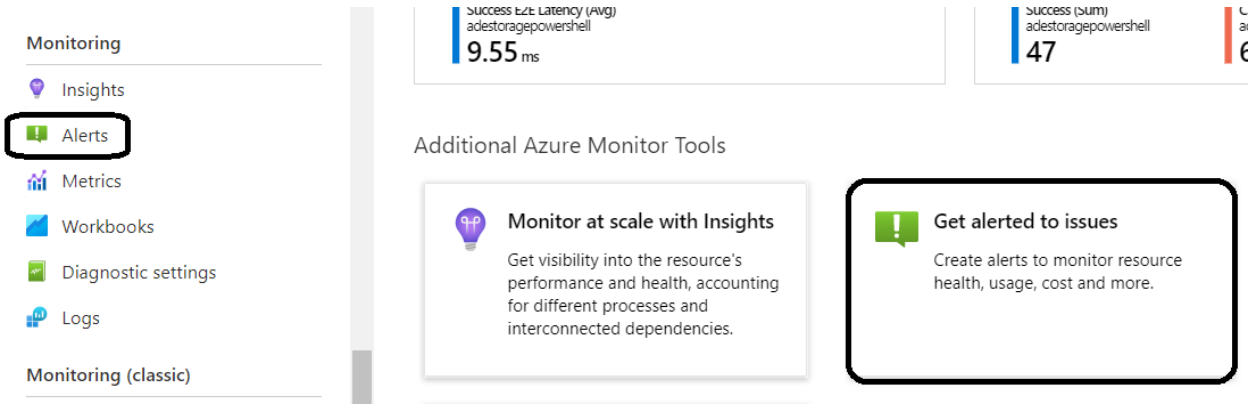
1. In the Azure portal, locate and open the storage account. In our case, the storage account is **adestoragepowershell**. On the storage account page, click the Monitoring tab along the bottom and scroll to **Alerts** under the **Monitoring** section:



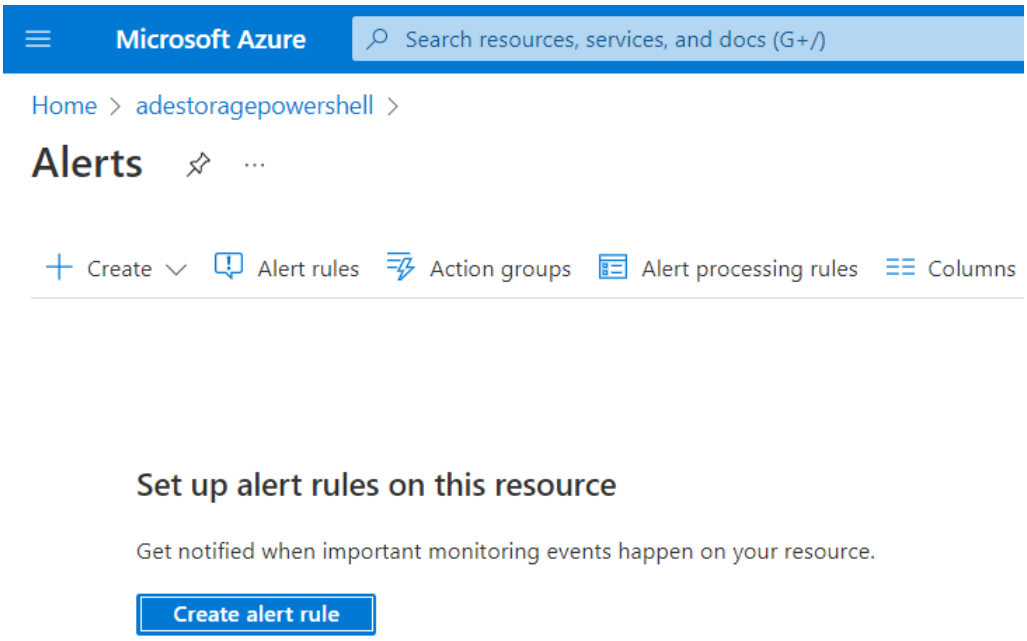
2. Over time we'll see data in our Monitoring, but of course right after a resource is created this will be pretty sparse.



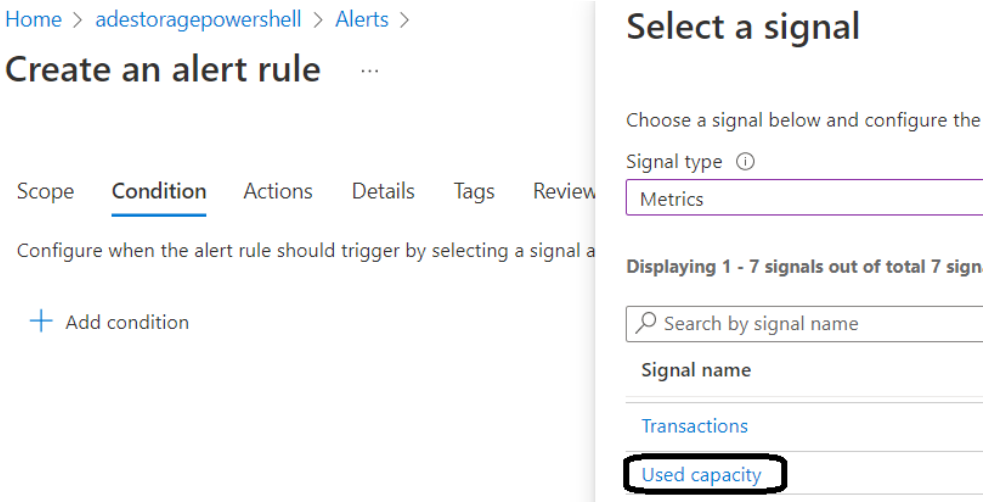
3. Select Alerts



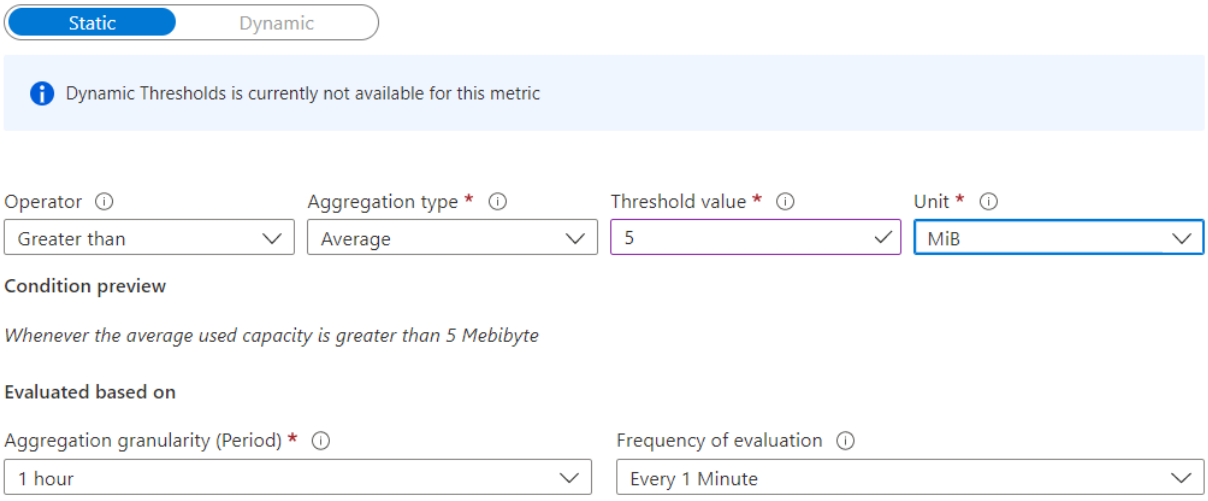
4. On the **Alerts** page, click on **Create alert rule**:



3. On the **Alerts | Create rule** page, observe that the storage account is listed by default under the **RESOURCE** section. You can add multiple storage accounts in the same alert.

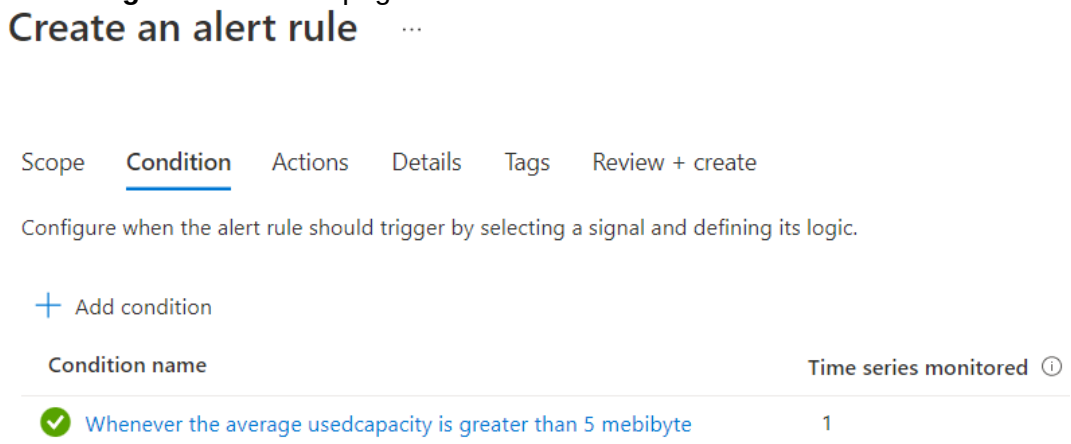


4. On the **Configure signal logic** page, select **Used capacity** under **Signal name**:

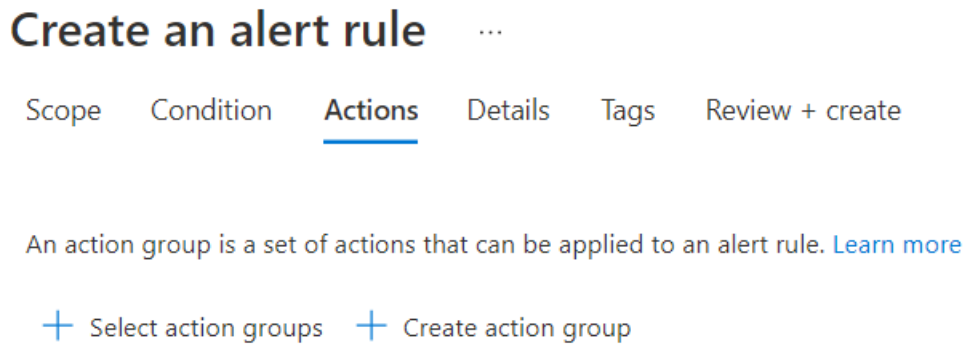


5. On the **Configure signal logic** page, under **Alert logic**, scroll down and set **Operator** as **Greater than**, **Aggregation type** as **Average**, and configure the threshold to 5 MB. We need to provide the value in bytes:

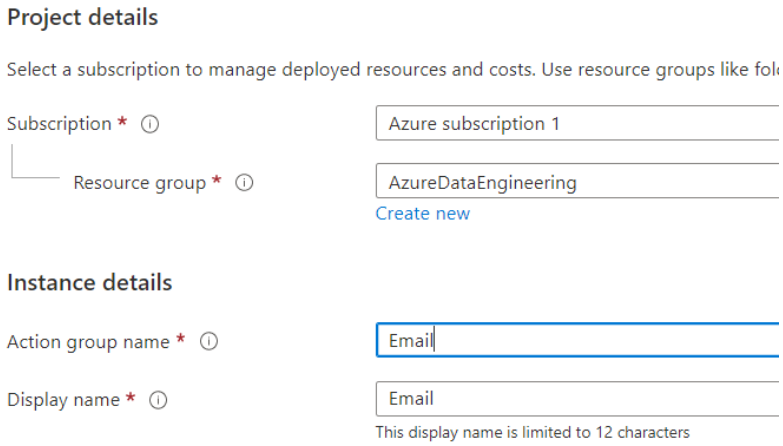
6. Click **Done** to configure the trigger. The condition is added, and we'll be taken back to the **Configure alerts rule** page:



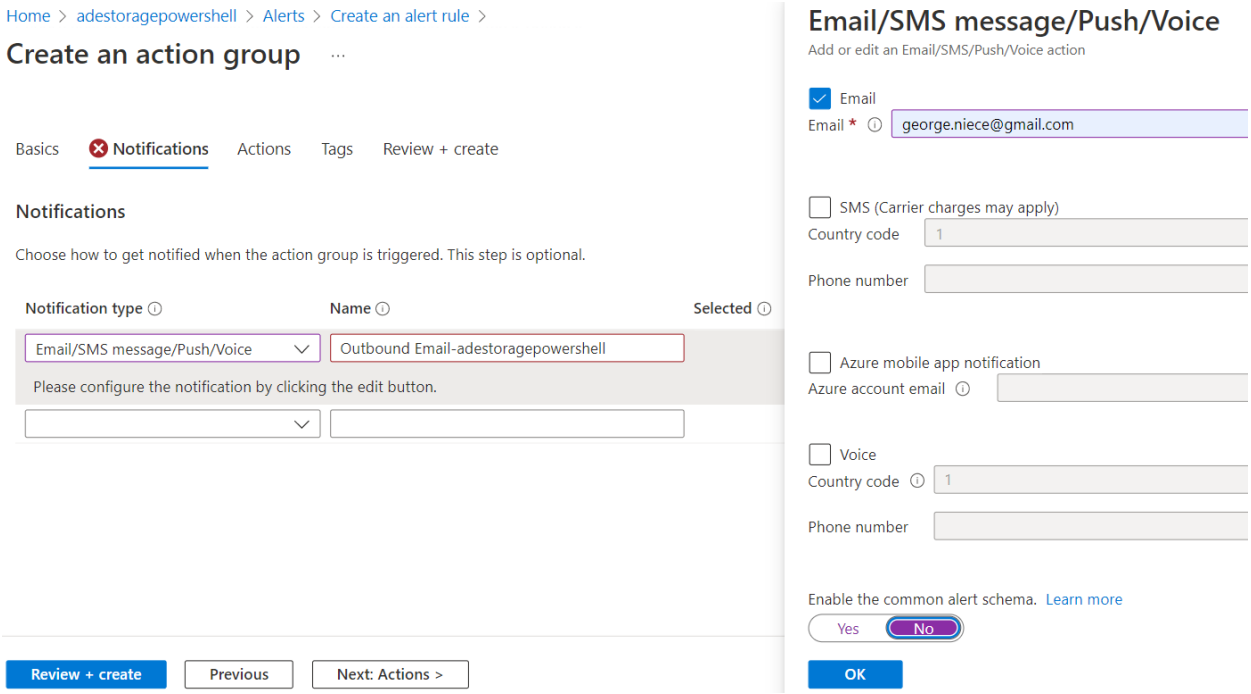
7. The next step is to add an action to perform when the alert condition is reached, select **Next: Actions >**. On the **Configure alerts rule** page, under the **ACTIONS GROUPS** section, click **Create action group**:



8. On the **Add action group** page, provide the action group name, short name, and resource group, select **Next: Notifications >**.



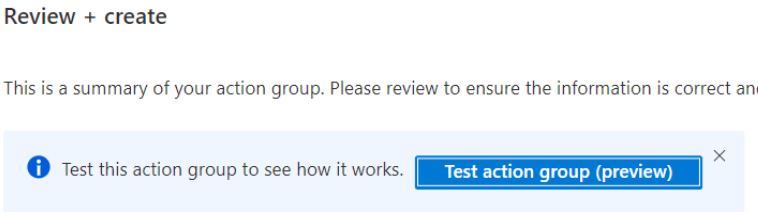
9. As we set **Notification Type** as **Email/SMS/Push/Voice**, a new sub-blade opens. Enter the Name of the Notification as **Outbound Email-adestoragepowershell**. In the **Email/SMS/Push/Voice** sub-blade, check the box for Email, and specify your desired contact email and click **OK**:



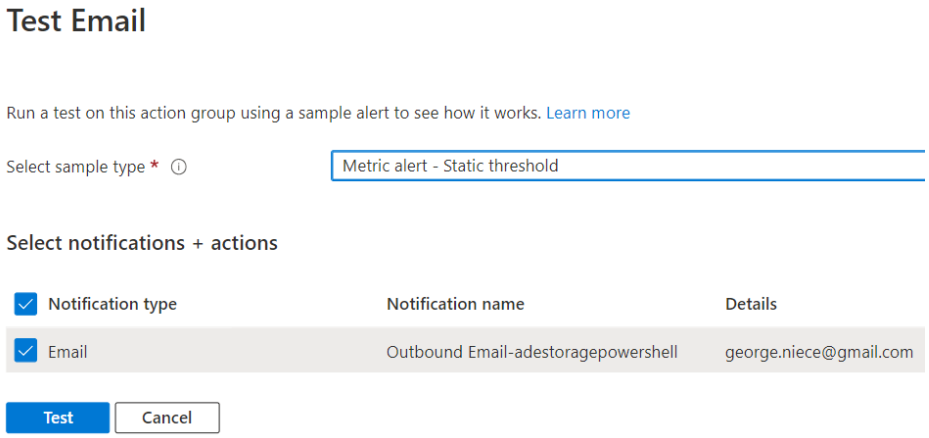
10. Select **Next: Actions >**
11. Click the action dropdown and review the possible additional actions that can be taken. We won't select an additional action beyond the notification we've already configured. Select **Next: Tags >**
12. For tags, enter **project** for Name and **azure-data-engineering** for Value. Select **Next: Review + create >**
13. Select **Review + create**

.

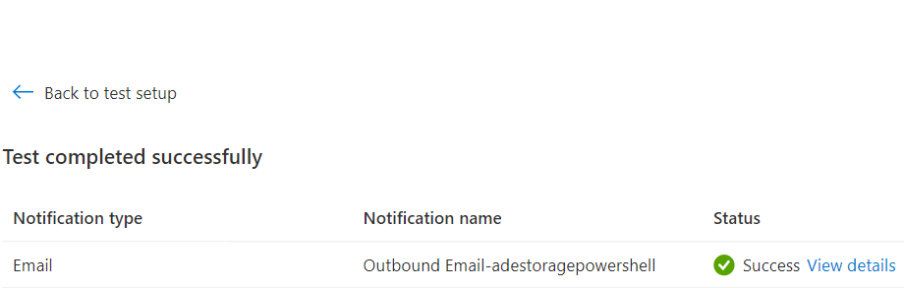
14. We can select to test the alert by configuring in the review screen



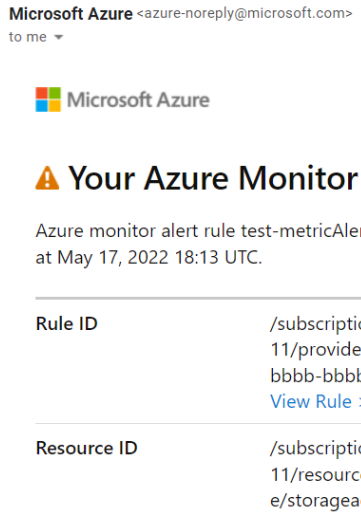
15. Select **Test action group**



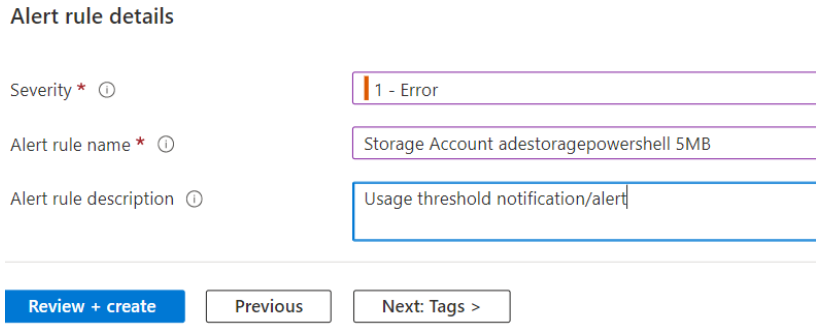
16. Select **Metric alert – Static threshold** and select **Test**



17. Once the test completed as seen in the screenshot we should have an email from Azure for the notification in our inbox as seen below

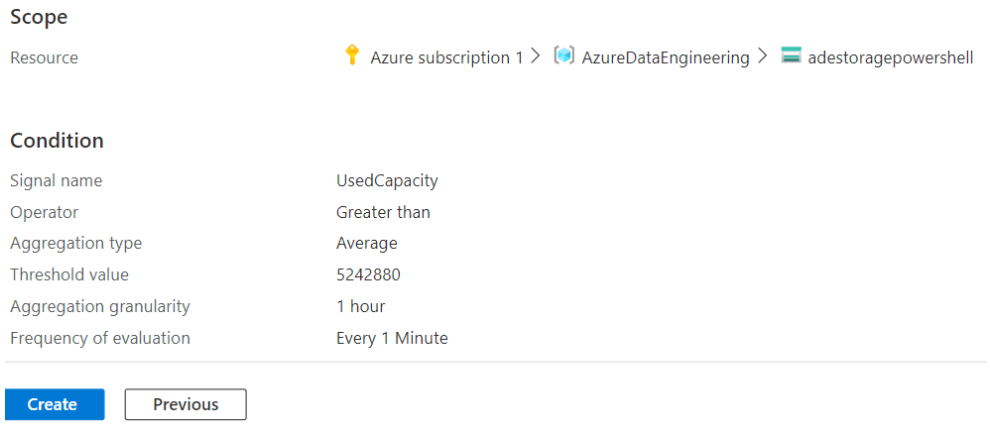


18. Select **Done** on the Test Preview sub-blade and then select **Create** to finish the Alert/Notification creation activity for our Storage Account Usage Threshold. **Select Next: Details>**



19. Enter the detail for the Severity, Name and Description for our Alert rule details, select **Next: Tags>**

20. Enter **project** for Tag Name and **azure-data-engineering** for the Tag Value, select **Next: Review + create>**



21. Click the **Create** button to create the alert.
22. The next step is to trigger the alert. To do that, upload any large 5MB+ file to the Azure storage account following the steps noted in the previous experiment, with Logfiles.

1. Execute the following commands if you've closed the original window that you did the previous experiments in:

```
$storageaccountname="adestoragepowershellxx"
$containername="logfiles"
$resourcegroup="AzureDataEngineeringxx"
#Get the Azure Storage account context
    $storagecontext = (Get-AzStorageAccount -ResourceGroupName
$resourcegroup -Name $storageaccountname).Context;
```

Note: Substitute for the student number for xx, (student1 would be adestoragepowershell01, student12 would be adestoragepowershell12). Similarly as we did earlier for the AzureDataEngineeringxx ResourceGroupName (student1 would be AzureDataEngineering01, student12 would be AzureDataEngineering12).

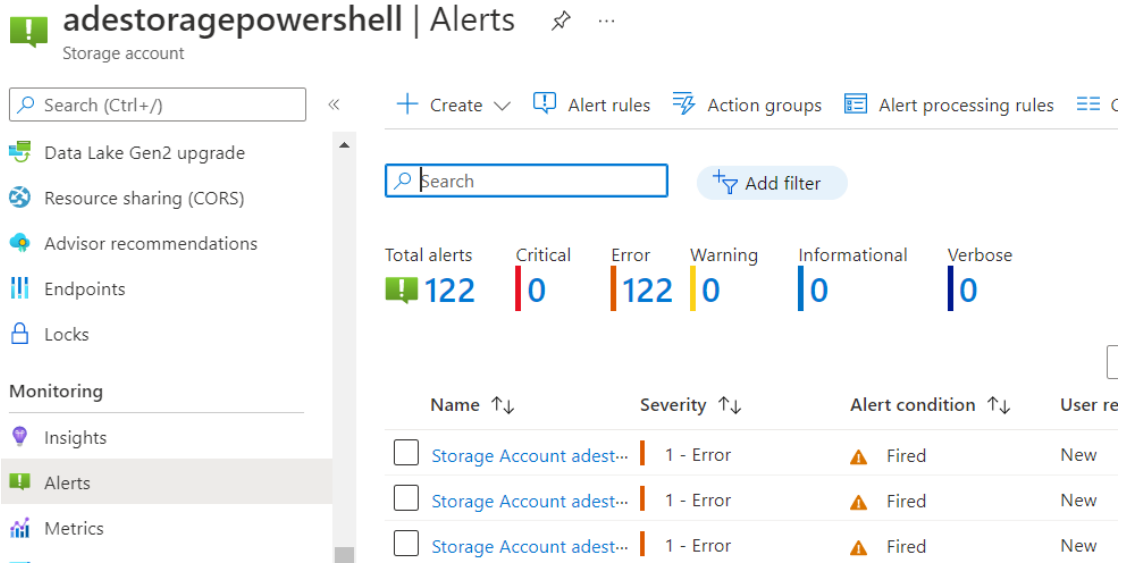
The file noted below, **BoatFileAlertTest.pdf**, is in the labs subfolder for the GitHub course repo or you can use any file that is larger than 5MB to cause the notification to fire.

Set-AzStorageBlobContent -File "BoatFileAlertTest.pdf" -Context $storagecontext -Blob BoatFileAlertTest.pdf -Container $containername

23. We can verify the uploaded files that we've placed in our Storage Container

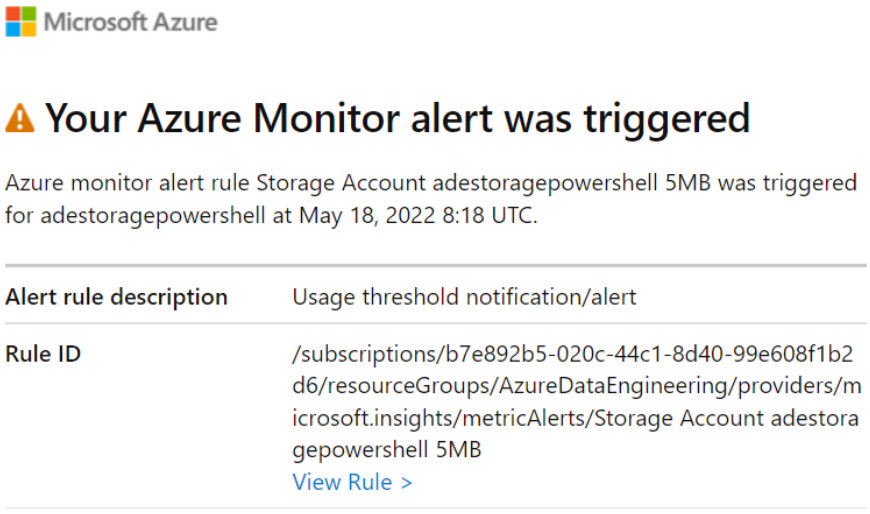
PS C:\Users\kubernetes\Downloads> **Get-AzStorageBlob -Container $containername -Context $storagecontext**

24. The triggered alerts are listed on the **Alerts** page, as shown in the following screenshot:



Note: Alerts are aggregated and for storage container usage capacity don't trigger for 6 hours. So we'll have that alert later today or by tomorrow after the uploads (ensuring from our Get-AzStorageBlob output that we have over 5MB

An email is sent to the email ID specified in the email action group. The email appears as shown in the following snapshot:



How it works…

Setting up an alert is easy. At first, we need to define the alert condition (trigger or signal). An alert condition defines the metrics and the threshold that, when breached, the alert is to be triggered. We can define more than one condition on multiple metrics for one alert.

We then need to define the action to be performed when the alert condition is reached. We can define more than one action for an alert. In our example, in addition to sending an email when the used capacity is more than 5 MB, we can configure Azure Automation to delete the old blobs/files so as to maintain the Azure storage capacity within 5 MB.

There are other signals such as transactions, Ingress, Egress, Availability, Success Server Latency, and Success E2E Latency on which alerts can be defined. Detailed information on monitoring Azure storage is available at **https://docs.microsoft.com/ en-us/azure/storage/common/storage-monitoring-diagnosingtroubleshooting**.

How it works…

Setting up an alert is straight forward. At first, we need to define the alert condition (trigger or signal). An alert condition defines the metrics and the threshold that, when breached, the alert is to be triggered. We can define more than one condition on multiple metrics for one alert.

We then need to define the action to be performed when the alert condition is reached. We can define more than one action for an alert. In our example, in addition to sending an email when the used capacity is more than 5 MB, we can configure Azure Automation to delete the old blobs/files so as to maintain the Azure storage capacity within 5 MB.

There are other signals such as transactions, Ingress, Egress, Availability, Success Server Latency, and Success E2E Latency on which alerts can be defined. Detailed information

on monitoring Azure storage is available at **https://docs.microsoft.com/ en-us/azure/storage/common/storage-monitoring-diagnosingtroubleshooting**.