

Get Alert Rules and Incidents

Last updated by | Balaji Barmavat | Nov 30, 2020 at 11:07 PM PST

Contents

- [Using Jarvis to get alert rules configurations:](#)
- [Using Jarvis to get alert incidents:](#)
- [Using Kusto to get the alert jobs and emails sent:](#)
- [Using PowerShell from customer side to get alerts and inci...](#)

- How to get all alerts rules by subscription or resource group.
- How to get incidents by subscription or alert rule.
- How to query alert jobs telemetry on Kusto.

Using Jarvis to get alert rules configurations:

1. Browse to <https://jarvis-west.dc.ad.msft.net> > Public
2. Go to the path **MonitoringService** > **Get alerts rules and incidents** > **00 - Get alert rules by subscription id**:

Or **MonitoringService** > **Get alerts rules and incidents** > **01 - Get alert incidents by subscription id**

MonitoringService > **Get alerts rules and incidents** > **00 - Get alert rules by subscription id**

MonitoringService > **Get alerts rules and incidents** > **01 - Get alert incidents by subscription id**

3. Select Endpoint region and insert subscription id(, resource group):

Endpoint*	Australia East
Subscription Id ?	
Resource group name ?	

4. Search for **Microsoft.Sql** or the database full resource id on result:

00 - Get alert rules by subscription id

```

2=  "value": {
3=    {
4=      "id": "/subscriptions/4b5029c5-b077-4428-bddf-aa27fac98761/resourceGroups/v-victmo/providers/microsoft.insights/alertrules/cpu",
5=      "name": "cpu",
6=      "type": "Microsoft.Insights/alertRules",
7=      "location": "northeurope",
8=      "tags": {
9=        "$type": "Microsoft.WindowsAzure.Management.Common.Storage.CasePreservedDictionary, Microsoft.WindowsAzure.Management.Common.Storage",
10=        "hidden-link:/subscriptions/*SubscriptionId*/resourceGroups/v-victmo/providers/Microsoft.Sql/servers/vsqueunorth/databases/AdventureWorksLT": "Resource"
11=      },
12=      "properties": {
13=        "name": "cpu",
14=        "description": "",
15=        "isEnabled": false,
16=        "condition": {
17=          "odata.type": "Microsoft.Azure.Management.Monitoring.Alerts.Models.ThresholdRuleCondition, Microsoft.WindowsAzure.Management.Mon.Client",
18=          "dataSource": {
19=            "odata.type": "Microsoft.WindowsAzure.Management.Monitoring.Alerts.Models.RuleMetricDataSource, Microsoft.WindowsAzure.Management.Mon.Client",
20=            "odata.type": "Microsoft.Azure.Management.Insights.Models.RuleMetricDataSource",
21=            "resourceId": "/subscriptions/*SubscriptionId*/resourceGroups/v-victmo/providers/Microsoft.Sql/servers/vsqueunorth/databases/AdventureWorksLT",
22=            "resourceLocation": null,
23=            "metricNamespace": null,
24=            "metricName": "cpu_percent",
25=            "legacyResourceId": null
26=          },
27=          "operator": "GreaterThan",
28=          "threshold": 1,
29=          "windowSize": "PT5M",
30=          "timeAggregation": "Average"
31=        },
32=        "action": {
33=          "odata.type": "Microsoft.WindowsAzure.Management.Monitoring.Alerts.Models.RuleEmailAction, Microsoft.WindowsAzure.Management.Mon.Client",
34=          "odata.type": "Microsoft.Azure.Management.Insights.Models.RuleEmailAction",
35=          "sendToServiceOwners": false,
36=          "customEmails": [
37=            "moreno.victor@gmail.com"
38=          ]
39=        },
40=        "lastUpdatedTime": "2018-01-09T17:28:25.7659861Z",
41=        "provisioningState": "Succeeded",
42=        "actions": []
43=      },
44=    },
45=  },

```



5. Sample JSON output:

```

{
  "id": "/subscriptions/*SubscriptionId*/resourceGroups/v-victmo/providers/microsoft.insights/alertrules/test",
  "name": "test",
  "type": "Microsoft.Insights/alertRules",
  "location": "northeurope",
  "tags": {
    "$type": "Microsoft.WindowsAzure.Management.Common.Storage.CasePreservedDictionary, Microsoft.WindowsAzure.Management.Common.Storage",
    "hidden-link:/subscriptions/*SubscriptionId*/resourceGroups/v-victmo/providers/Microsoft.Sql/servers/vsqueunorth/databases/AdventureWorksLT": "Resource"
  },
  "properties": {
    "name": "test",
    "description": "123",
    "isEnabled": false,
    "condition": {
      "$type": "Microsoft.WindowsAzure.Management.Monitoring.Alerts.Models.ThresholdRuleCondition, Microsoft.WindowsAzure.Management.Mon.Client",
      "odata.type": "Microsoft.Azure.Management.Insights.Models.ThresholdRuleCondition",
      "dataSource": {

```

```
"$type":  
"Microsoft.WindowsAzure.Management.Monitoring.Alerts.Models.RuleMetricDataSource,  
Microsoft.WindowsAzure.Management.Mon.Client",  
  
"odata.type": "Microsoft.Azure.Management.Insights.Models.RuleMetricDataSource",  
  
"resourceUri": "/subscriptions/*SubscriptionId*/resourceGroups/v-  
victmo/providers/Microsoft.Sql/servers/vsqueunorth/databases/AdventureWorksLT",  
  
"resourceLocation": null,  
  
"metricNamespace": null,  
  
"metricName": "connection_successful",  
  
"legacyResourceId": null  
  
},  
  
"operator": "GreaterThan",  
  
"threshold": 1,  
  
"windowSize": "PT5M",  
  
"timeAggregation": "Total"  
  
},  
  
"action": {  
  
    "$type": "Microsoft.WindowsAzure.Management.Monitoring.Alerts.Models.RuleEmailAction,  
Microsoft.WindowsAzure.Management.Mon.Client",  
  
    "odata.type": "Microsoft.Azure.Management.Insights.Models.RuleEmailAction",  
  
    "sendToServiceOwners": true,  
  
    "customEmails": [  
  
        "mmoreno.victor@gmail.com"  
  
    ]  
  
},  
  
"lastUpdatedTime": "2018-01-09T17:28:30.9320812Z",  
  
"provisioningState": "Succeeded",  
  
"actions": []  
  
}
```

```
}
```

Using Jarvis to get alert incidents:

1. Go to the path **MonitoringService** > **Get alerts rules and incidents** > **03 - Get alert incidents by alert rule name**

MonitoringService > Get alerts rules and incidents > 03 - Get alert incidents by alert rule name

2. Select Endpoint region and insert subscription id, resource group name and alert rule name:

Endpoint*	North Europe
Subscription Id ?	15002008-0777-4403-8446-0361-00764
Resource group name ?	v-victmo
Alert rule name ?	cpu

3. Sample JSON output:

```
{
```

```
"value": [
```

```
{
```

```
"id":
```

```
"L3N1YnNjcmlwdGlvbnMvNGI1MDI5YzUtMGY3Ny00NDI4LWJkZGYtYWUyN2ZhYzk4NzYxL3Jlc291cmNIR3JvdXBzL3YtdmljdG1vL3Byb3ZpZGVycy9taWNyb3NvZnQuaW5zaWdodHMvYWxlcuRydWxlcY9jcHUwNjM2NTExMTUyOTg5Nzc0OTU0",
```

```
"ruleName": "/subscriptions/*SubscriptionId*/resourceGroups/v-victmo/providers/microsoft.insights/alertrules/cpu",
```

```
"isActive": true,
```

```
"activatedTime": "2018-01-09T17:21:38.9774954+00:00",
```

```
"resolvedTime": null,
```

```
"targetResourceId": null,
```

```
"targetResourceLocation": null,
```

```
"legacyResourceId": null
```

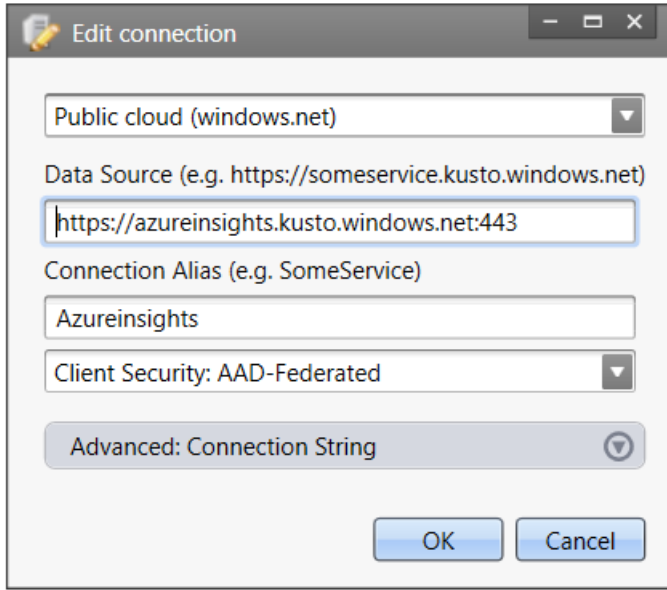
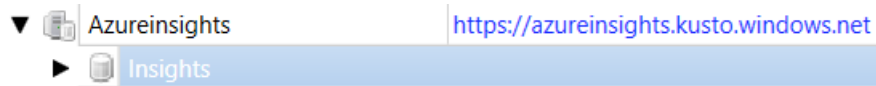
```
}
```

```
]
```

}

Using Kusto to get the alert jobs and emails sent:

1. Go to <http://idweb> and join security group: **Insight Kusto Users**.
2. On Kusto, create a new tab for Azureinsights.Insights.



3. Queries:
 - a. Get email action.

```
JobTraces
| where jobPartition contains "{SubscriptionId}"
| where jobId startswith "ALERT:"
| where message contains "Email"

| project TIMESTAMP , jobPartition , ActivityId , operationName , message , exception , jobId
```

- b. Get operations history from activity id retrieved from previous query.

```
JobTraces
| where ActivityId contains "{ActivityId}"
| project TIMESTAMP , jobPartition , ActivityId , operationName , message , exception , jobId
```

Using PowerShell from customer side to get alerts and incidents:

1. Alerts:

```
Get-AzureRmResource -ResourceGroupName "ResourceGroupName" -ResourceType
microsoft.insights/alertrules
```

```
Get-AzureRmResource -ResourceGroupName "ResourceGroupName" -ResourceType  
microsoft.insights/alertrules -ResourceName "ResourceName" -ApiVersion 2016-03-01
```

2. Incidents:

```
Get-AzureRmResource -ResourceGroupName "ResourceGroupName" -ResourceType  
microsoft.insights/alertrules/incidents -ResourceName "ResourceName" -ApiVersion 2016-03-01
```

```
Get-AzureRmResource -ResourceGroupName "ResourceGroupName" -ResourceType  
microsoft.insights/alertrules/incidents -ResourceName "ResourceName/IncidentId" -ApiVersion 2016-03-01
```

How good have you found this content?



-