



Version Control

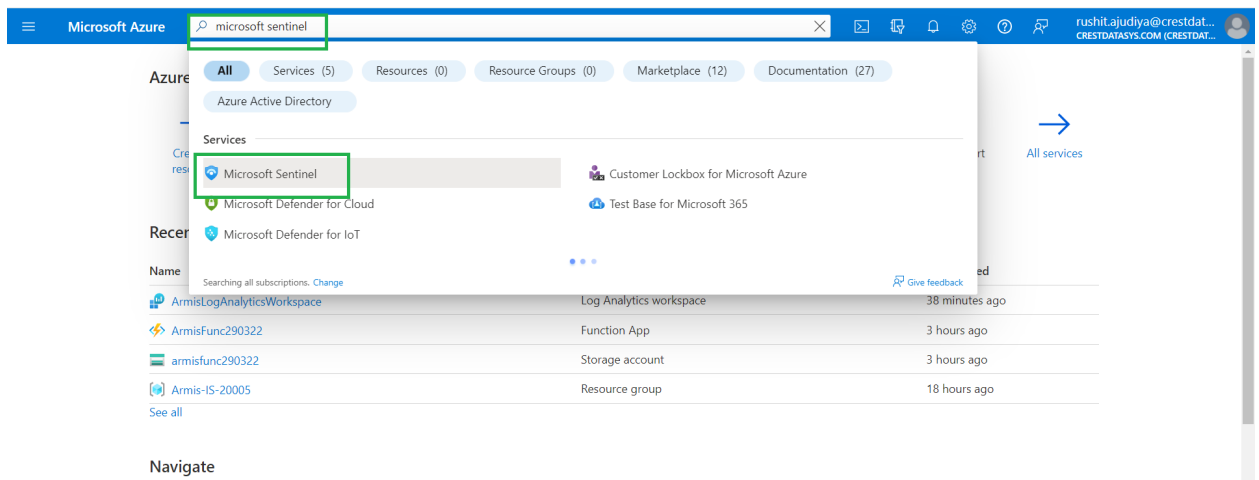
#	Document Version	Date	Owner	Document Status	Comments
1	1.0.0	22th Jul 2022	Crest Data Systems	Initial Draft	

Contents

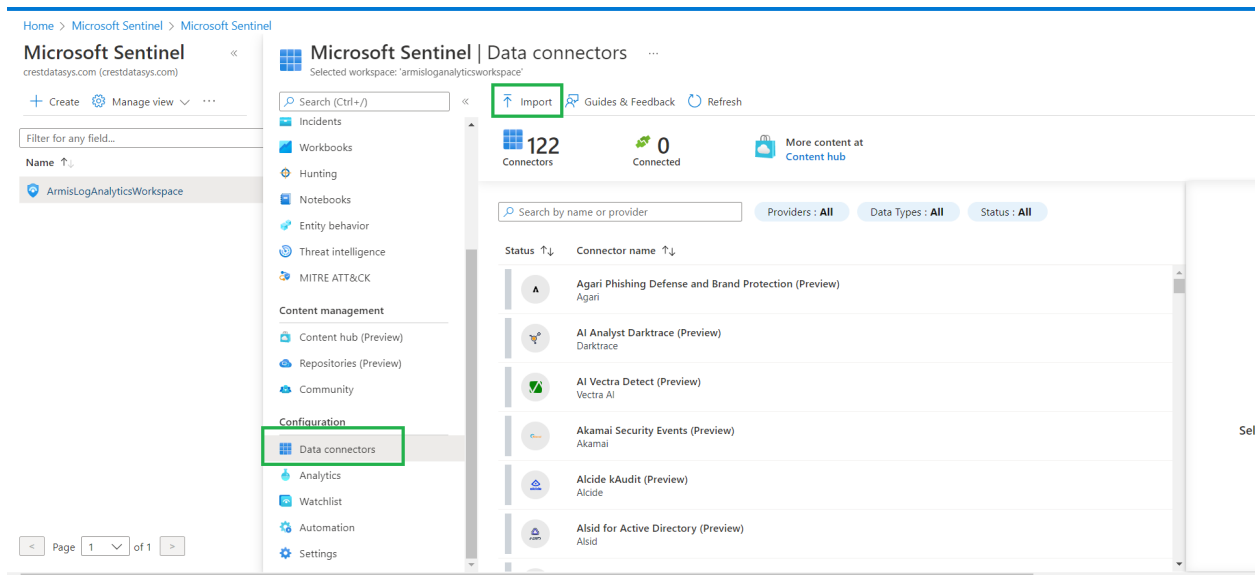
Version Control	2
Contents	3
Armis Data Connector Installation Guide	4
Armis Logic App Installation Guide	11

Armis Data Connector Installation Guide

1. Login to Azure portal with below given link
(<https://portal.azure.com/?feature.BringYourOwnConnector=true>) using the MS Azure credentials
2. Now we have to search for the Microsoft Sentinel service in the azure portal search bar and select **Microsoft Sentinel** service.



3. Then select the Microsoft Sentinel Workspace in which you want to deploy the data connector.
4. After that in the left panel scroll down and select the **"Data Connectors"**.
5. Now click on the import button.




6. Now it will show a window in which you have to select the JSON file (**Armish_Activity_API_FunctionApp.json**) which will be provided by us.
7. After that it will show a window like below.

Home > Microsoft Sentinel > Microsoft Sentinel | Data connectors >

ArmishActivities

Download

**ArmishActivities**

Not connected
Status

Armish
Provider




⌚ --
Last Log Received

Description

The **Armish** connector provides the capability to ingest ArmishXXX logs and events into Azure Sentinel. The connector provides visibility into Message and Click events in Azure Sentinel to view dashboards, create custom alerts, and to improve monitoring and investigation capabilities.

Last data received
--

Related content

 0 Workbooks  1 Queries  0 Analytics rules templates

Instructions

Next steps

Prerequisites

To integrate with ArmishActivities make sure you have:


- ✓ **Workspace:** read and write permissions on the workspace are required.
- ✓ **Keys:** read permissions to shared keys for the workspace are required. [See the documentation to learn more about workspace keys.](#)
- ❗ **Microsoft.Web/sites permissions:** Read and write permissions to Azure Functions to create a Function App is required. [See the documentation to learn more about Azure Functions.](#)
- ❗ **REST API Credentials/permissions:** Armish Secret Key is required. See the documentation to learn more about API on the <https://developer.armish.com/docs/api-basics>

8. In the right panel scroll down and you will be able to see the “**Deploy to Azure**” button.

Home > Microsoft Sentinel > Microsoft Sentinel | Data connectors >

ArmishActivities

Download

**ArmishActivities**

Not connected
Status

Armish
Provider




⌚ --
Last Log Received

Description

The **Armish** connector provides the capability to ingest ArmishXXX logs and events into Azure Sentinel. The connector provides visibility into Message and Click events in Azure Sentinel to view dashboards, create custom alerts, and to improve monitoring and investigation capabilities.

Last data received
--

Related content

 0 Workbooks  1 Queries  0 Analytics rules templates


Instructions

Next steps

Option 1 - Azure Resource Manager (ARM) Template

Use this method for automated deployment of the Armish connector.

1. Click the **Deploy to Azure** button below.



2. Select the preferred **Subscription**, **Resource Group** and **Location**.
3. Enter the **Workspace ID**, **Workspace Key**, **ArmishSecretKey**, **ArmishURL** (<https://<armish-instance>.armish.com/api/v1/>) , **ArmishActivitiesTableName**.
4. Mark the checkbox labeled **I agree to the terms and conditions stated above**.
5. Click **Purchase** to deploy.

9. It will redirect you to the custom deployment screen where you can see the input fields and window like the below.

[Home](#) >

Custom deployment ...

Deploy from a custom template

Basics Review + create

Template



[Customized template](#) 
8 resources


[Edit template](#)


[Edit parameters](#)


[Visualize](#)

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

CrestAzure1



Resource group * ⓘ



[Create new](#)

Instance details

Review + create

[< Previous](#)

[Next : Review + create >](#)

10. Add the information as mentioned below in input fields and click on “**Review + Create**”.

[Home](#) >

Custom deployment

Deploy from a custom template

Subscription * ⓘ

CrestAzure1

Resource group * ⓘ

[Create new](#)

Instance details

Region * ⓘ

East US

Function Name

Activity

Workspace ID

<workspaceID>

Workspace Key

<workspaceKey>

ArmIs Secret Key

ArmIs URL

<ArmIsURL>

ArmIs Activities Table Name

<ArmIsActivitiesTableName>

ArmIs Schedule

<Cron Expression>

Avoid Duplicates

false

Review + create

< Previous

Next : Review + create >

- **Subscription** : Select the subscription of your account.
- **Resource group** : Resource group in which you want to deploy data connector(Function App)
- **Region** : Select the region (By default it will be selected according to your Resource group selection.
- **Function Name** : Name of the function which you would like.
Note : You are able to input Maximum 11 characters in this field as a validation from sentinel.
- **Workspace ID** : Workspace ID of your Log Analytics Workspace in which you want to load ingested armis data.
- **Workspace Key** : Workspace Key of your Log Analytics Workspace in which you want to load ingested armis data.
- **ArmIs Secret Key** : API Secret Key required for Armis Authentication.
- **ArmIs URL** : Armis portal URL.
(Example : <https://<armis-instance>.armis.com/api/v1/>)
- **ArmIs Alerts Table Name** : Alerts table name in which ingested Armis Alerts data will be loaded.
- **Avoid Duplicates** : If true, data duplication is avoided, but there might be a change of data loss
If false, data may get duplicated

11. It will validate the Armis Data Connector deployment configuration content and if the validation is successful, it will show a **“Create”** button at the bottom of the screen.

Home >


Custom deployment

Deploy from a custom template

✓ Validation Passed

Basics **Review + create**

Summary

 Customized template
8 resources

Terms

[Azure Marketplace Terms](#) | [Azure Marketplace](#)

By clicking “Create,” I (a) agree to the applicable legal terms associated with the offering; (b) authorize Microsoft to charge or bill my current payment method for the fees associated the offering(s), including applicable taxes, with the same billing frequency as my Azure subscription, until I discontinue use of the offering(s); and (c) agree that, if the deployment involves 3rd party offerings, Microsoft may share my contact information and other details of such deployment with the publisher of that offering.

Microsoft assumes no responsibility for any actions performed by third-party templates and does not provide rights for third-party products or services. See the [Azure Marketplace Terms](#) for additional terms.

Deploying this template will create one or more Azure resources or Marketplace offerings. You acknowledge that you are responsible for reviewing the applicable pricing and legal terms associated with all resources and offerings deployed as part of this template. Prices and associated legal terms for any Marketplace offerings can be found in the [Azure Marketplace](#); both are subject to change at any time prior to deployment.

Neither subscription credits nor monetary commitment funds may be used to purchase non-Microsoft offerings. These

Create < Previous Next

12. It will take some time to create the function app and storage. You can see the deployment status on the screen below.

Home >

Microsoft.Template-20220330125252 | Overview

Deployment

Search (Ctrl+/) < Delete Cancel Redeploy Refresh

Overview

Inputs

Outputs

Template

Deployment is in progress

Deployment name: Microsoft.Template-20220330125252 Start time: 3/30/2022, 12:53:02 PM
Subscription: CrestAzure1 Correlation ID: 8b8278bb-0a69-4315-a935-f6701149d0a9
Resource group: Armis-IS-20005

Deployment details (Download)

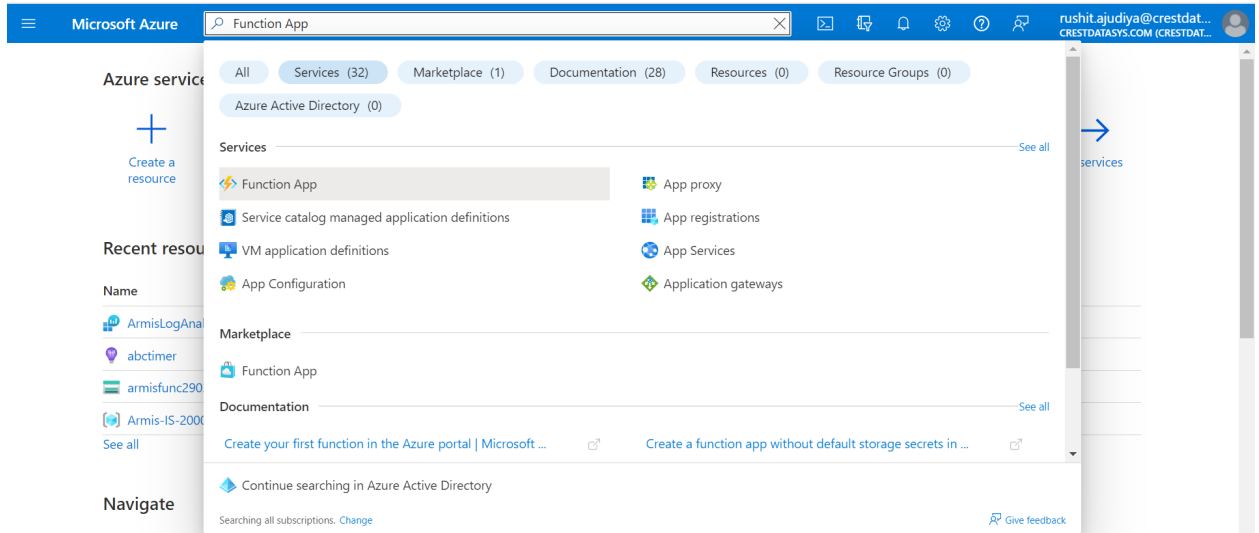
Resource	Type	Status	Operation details
armis12493m7buarpsqxu/default/a	Microsoft.Storage/storageAccount...	Created	Operation details
armis12493m7buarpsqxu/default	Microsoft.Storage/storageAccount...	OK	Operation details
armis12493m7buarpsqxu/default/a	Microsoft.Storage/storageAccount...	Created	Operation details
armis12493m7buarpsqxu/default/a	Microsoft.Storage/storageAccount...	Created	Operation details
armis12493m7buarpsqxu/default	Microsoft.Storage/storageAccount...	OK	Operation details
armis12493m7buarpsqxu	Microsoft.Storage/storageAccounts	OK	Operation details
armis12493m7buarpsqxu	Microsoft.insights/components	OK	Operation details
armis12493m7buarpsqxu	Microsoft.Storage/storageAccounts	OK	Operation details
armis12493m7buarpsqxu	Microsoft.insights/components	OK	Operation details

Microsoft Defender for Cloud
Secure your apps and infrastructure
[Go to Microsoft Defender for Cloud >](#)

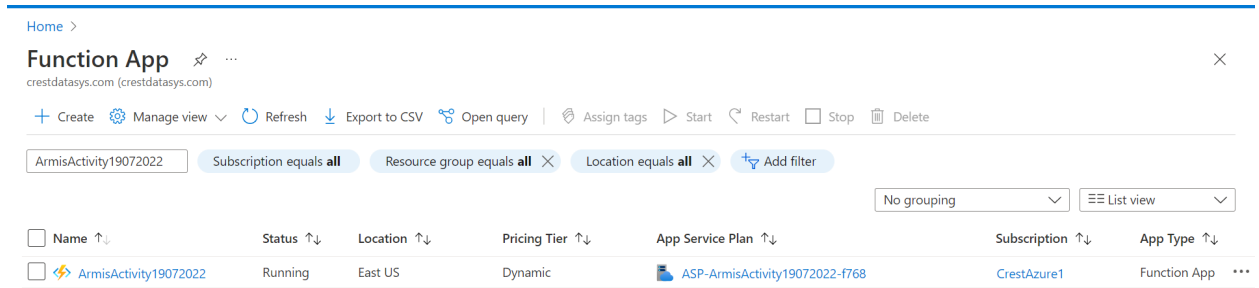
Free Microsoft tutorials
[Start learning today >](#)

Work with an expert
Azure experts are service provider partners who can help manage your assets on Azure and be your first line of support.
[Find an Azure expert >](#)

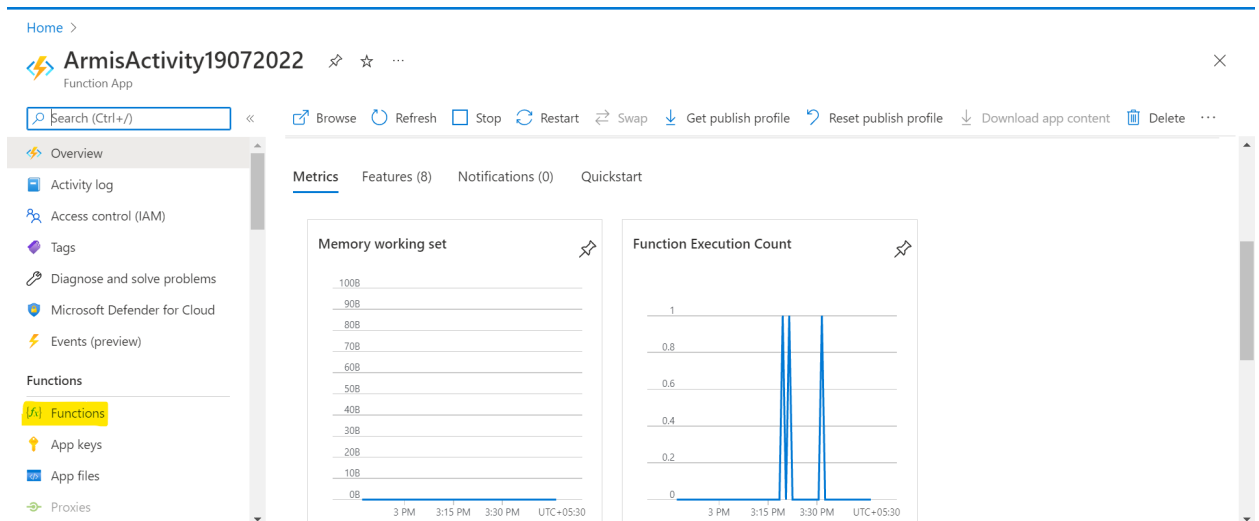
13. Now we need to see if the function app is properly deployed or not. And for that go to the search bar and search for the function app service in the azure portal.



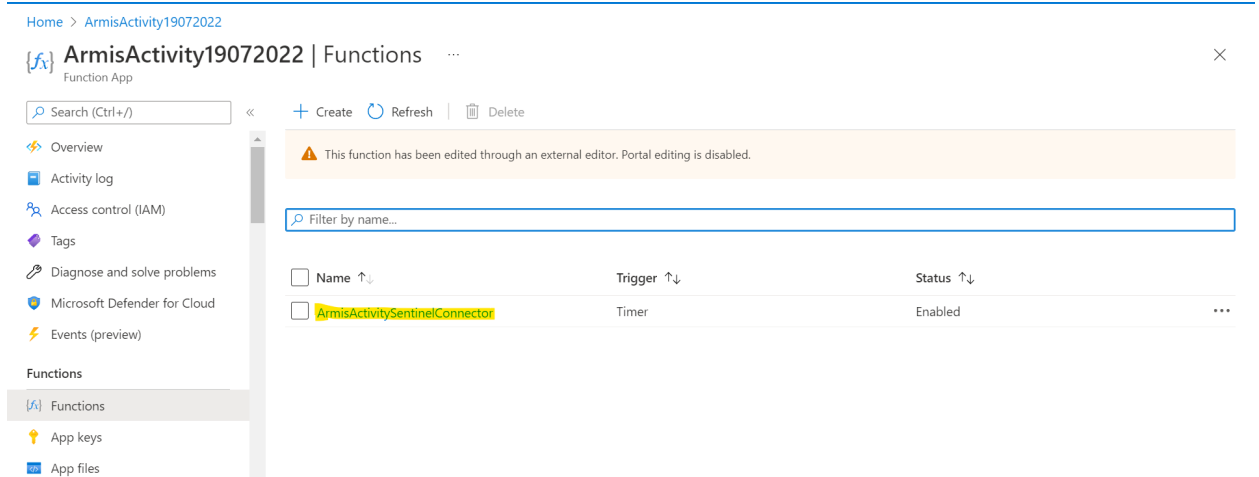
14. Now select the Function App(Data Connector) that you have deployed.
 Example : We have deployed a Function App(Data Connector) with the name **ArmisActivity19072022**.



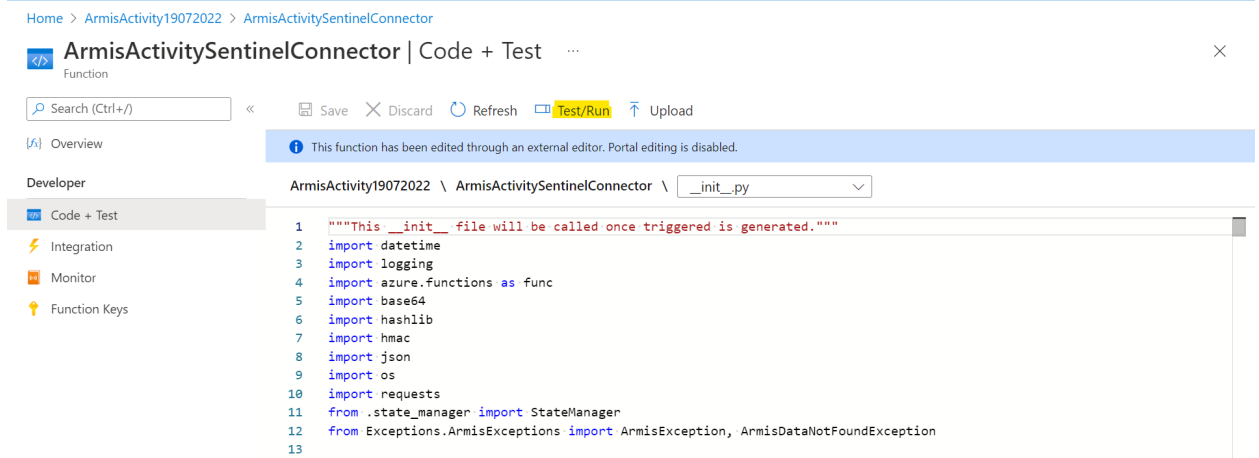
15. Click on the Functions from the left panel.



16. Now select a Function App in the right panel, It will redirect you to the Function App.



17. Now to test the Function App manually, click on the **code + test** and then click on **Test/Run**. It will pop up the window like in the image below.



18. Click on the **Run** button. If you get message **202 Accepted** then it runs successfully.

Note : To deploy Alert and Device function app you can follow the same step no 1 to 18

ArmIS Logic App Installation Guide

1. Login to Azure portal (<https://portal.azure.com/#home>) using the MS Azure credentials
2. **Logic App (Create a Logic App using an exported template)**
 - a. Go to this [link](#) to import the Logic App using the template which has been provided to you
 - b. Click on “Build your own template in the editor” to go to the Edit template screen

[Home](#) >

Custom deployment ...






Deploy from a custom template

Select a template Basics Review + create

Automate deploying resources with Azure Resource Manager templates in a single, coordinated operation. Create or select a template below to get started. [Learn more about template deployment](#)

 Build your own template in the editor

Common templates

-  Create a Linux virtual machine
-  Create a Windows virtual machine
-  Create a web app
-  Create a SQL database
-  Azure landing zone




- c. Click on “**Load file**” and select “**armis_update_alert_status_playbook_template.json**” file which has been provided to you

[Home](#) > [Custom deployment](#) >

Edit template ...

Edit your Azure Resource Manager template

+ Add resource ↑ Quickstart template ↑ Load file ↓ Download

 Parameters (0)
 Variables (0)
 Resources (0)

```
<<
1  {
2    "$schema": "https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#",
3    "contentVersion": "1.0.0.0",
4    "parameters": {},
5    "resources": []
6  }
```

Save

Discard

- d. Click on “Save” button and it will auto populate the data from the uploaded template

Microsoft Azure Search resources, services, and docs (G+/)

Home > Custom deployment >

Edit template

Edit your Azure Resource Manager template

+ Add resource ↑ Quickstart template ↑ Load file ↓ Download

<< 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37

```
21      "name": ""
22    },
23  },
24  "parameters": {
25    "PlaybookName": {
26      "defaultValue": "ArmisUpdateAlertStatus",
27      "type": "string"
28    }
29  },
30  "variables": {
31    "MicrosoftSentinelConnectionName": "[concat('MicrosoftSentinel-', parameters('Playbo
32    "KeyvaultConnectionName": "[concat('Keyvault-', parameters('PlaybookName'))]"
33  },
34  "resources": [
35    {
36      "properties": {
37        "provisioningState": "Succeeded"
```

Save Discard

- e. Select the appropriate “Resource Group”
- f. Enter Playbook Name
- g. Click on “Review + create” button

Home >

Custom deployment

Deploy from a custom template

Customized template 3 resources

Edit template Edit parameters Visualize

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ Pay-As-You-Go

Resource group * ⓘ

Create new

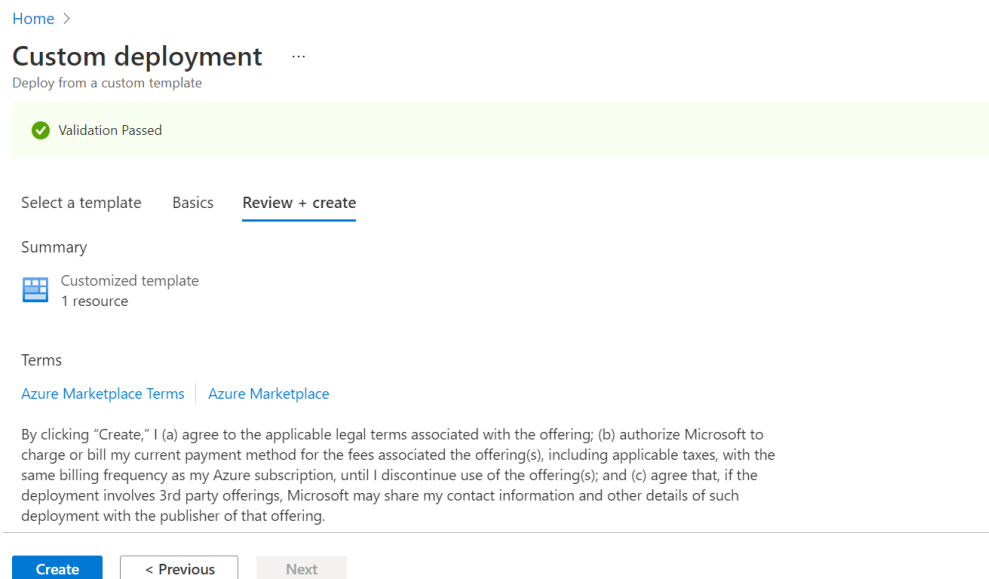
Instance details

Region * ⓘ East US

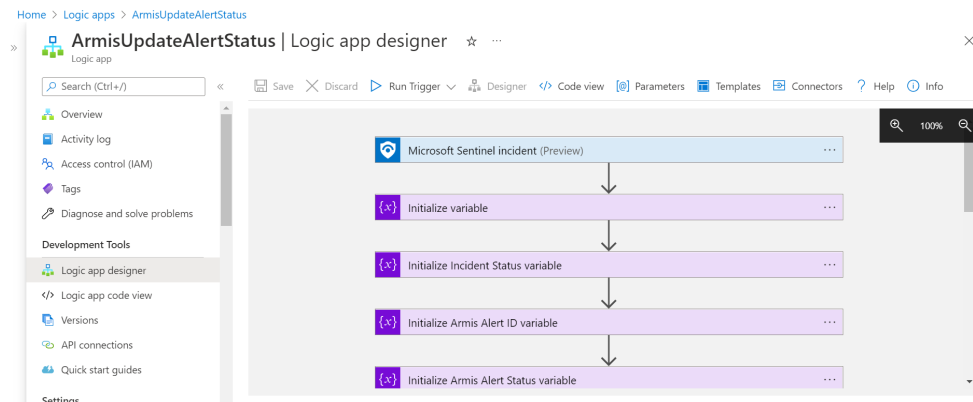
Playbook Name ArmisUpdateAlertStatus

Review + create < Previous Next : Review + create >

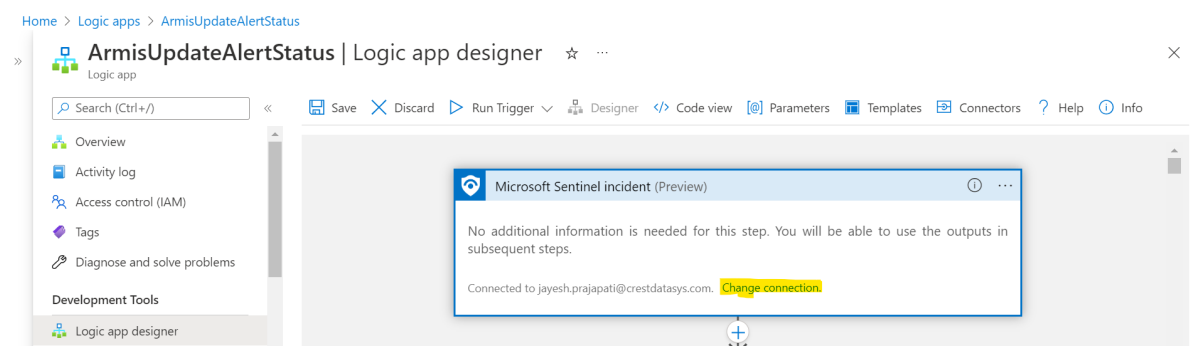
- h. It will validate the Logic app content and if the validation is successful, it will show a “Create” button at the bottom of the screen.
- i. Click on “Create” button



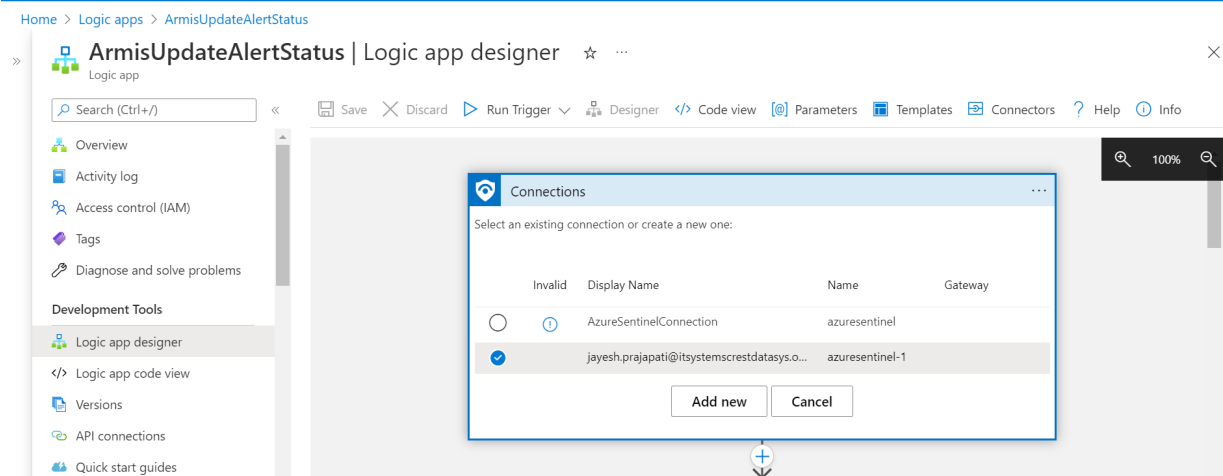
- j. The above step will deploy a Armis Update Alert Status Logic App(Playbook) on Azure Portal
- k. Finally logic app template is deployed successfully and it will be visible in “Logic App” service in azure portal



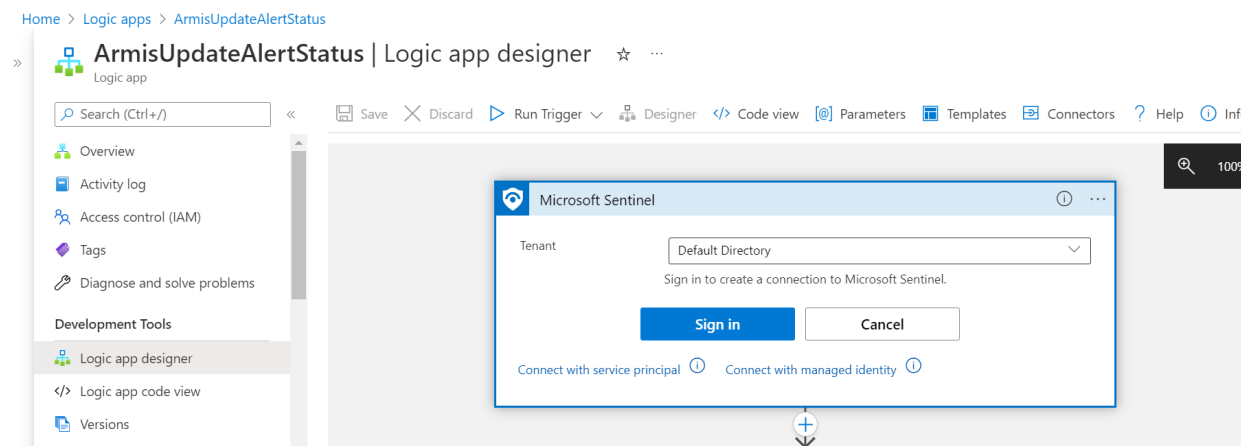
- l. Expand “Microsoft Sentinel incident (Preview)” trigger and click on “Change connection”



m. Click on “Add new” button from connection window



n. Click on “Sign in” button and it will pop-up once dialogue

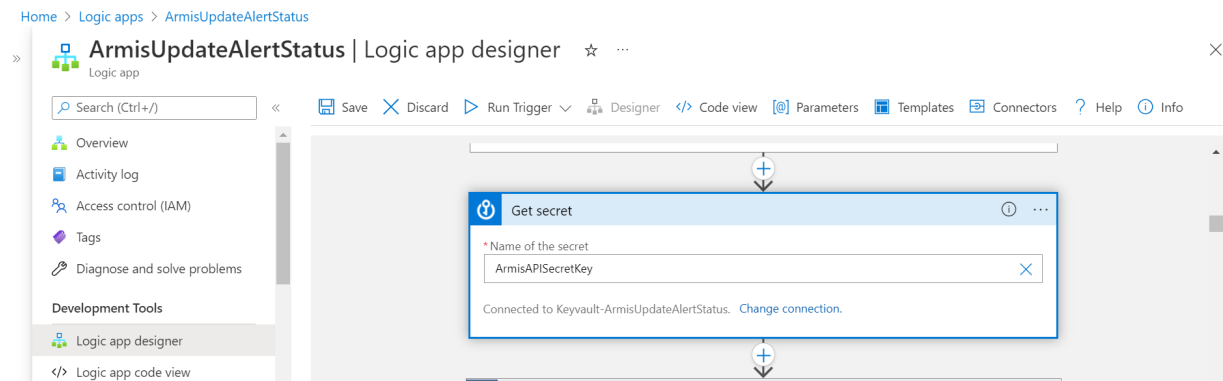


o. Sign in with your azure portal credentials

p. After successful sign in click on “Save” button to save changes

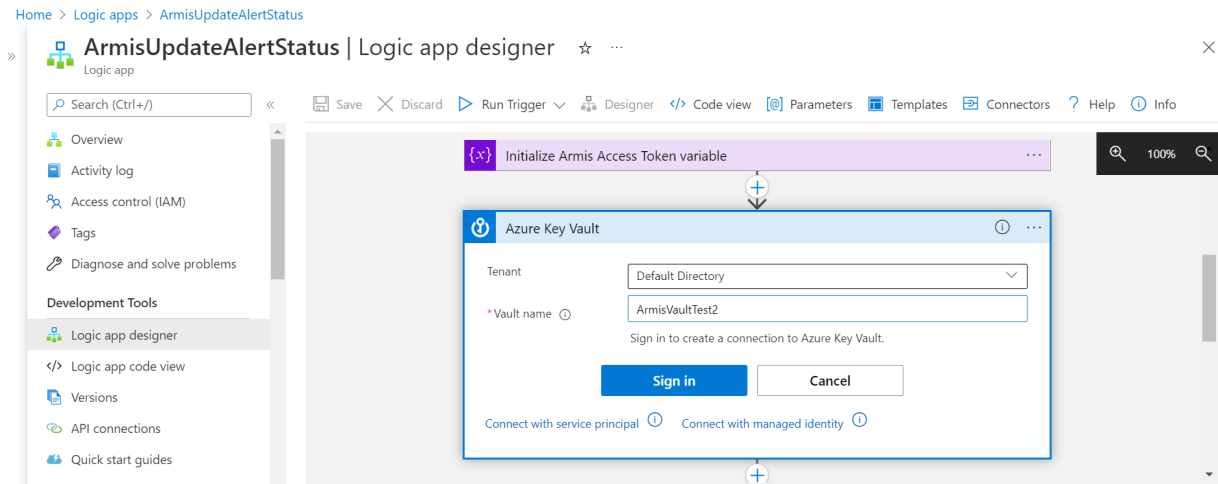
q. Now perform **step 3** to create key vault

r. After successfully creating Key Vaults and Secret go to logic app again and expand “Get Secret” action

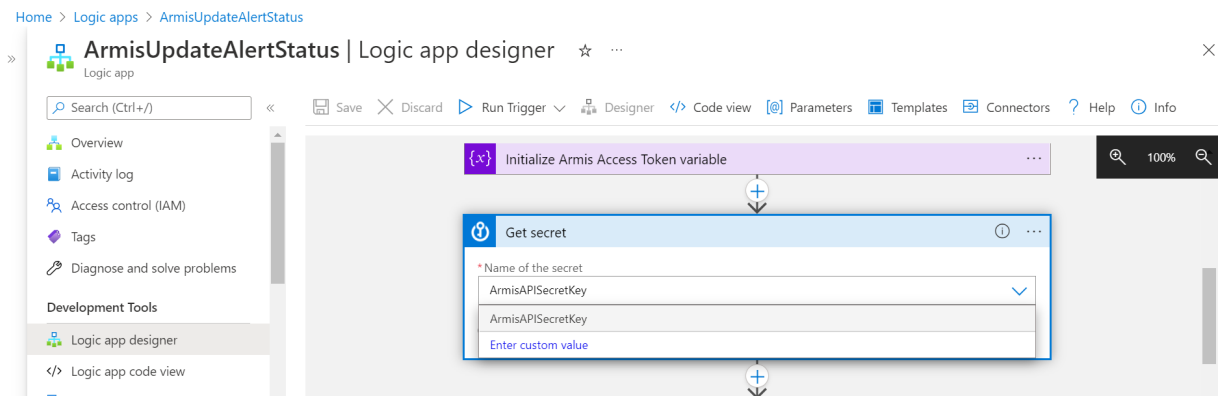


s. Click on “Change connection” and click on “Add new” button from connections screen

t. Now select “Tenant” and provide “Vault name” which you have created in step 3



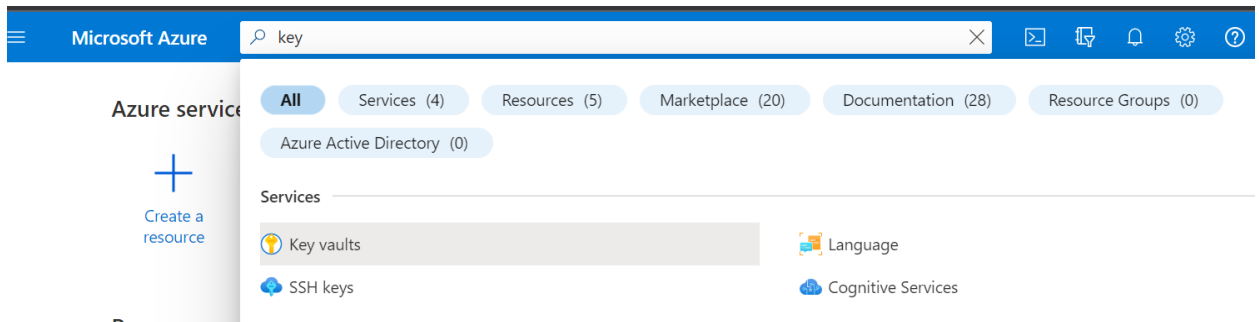
- u. Click on “Sign in” and provide your azure portal credentials in open pop-up window
- v. After that it will show all the secret available in the provided key vault



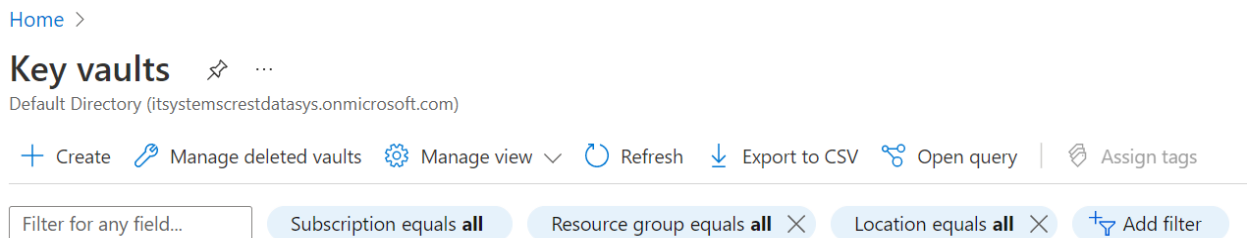
- w. Select the secret in which you have provided Armis Secret keys which will be used for Authentication
- x. Click on “Save” button to save the changes in the playbook

3. Create Key Vault

a. Go to “Key vaults”



b. Click on “Create” button



c. Select appropriate Resource group from dropdown

d. Enter Key vault name which globally unique

e. Select region of your resource group

f. Click on “Review + create” button

Create a key vault

Subscription *

Resource group *

Create new

Instance details

Key vault name * ⓘ

Region *

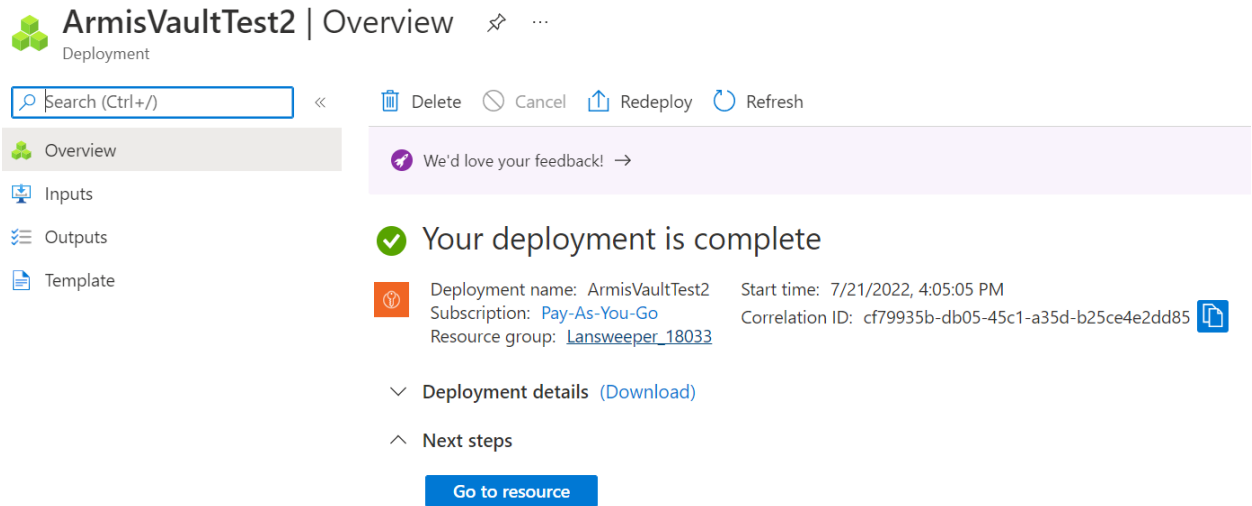
Pricing tier * ⓘ

Recovery options

Soft delete protection will automatically be enabled on this key vault. This feature allows you to recover or permanently delete a key vault and secrets for the duration of the retention period. This protection applies to the key vault and the secrets stored within the key vault.

Previous Next Review + create

g. The above step will create Key Vault



ArmisVaultTest2 | Overview Deployment

Search (Ctrl+/) << Delete Cancel Redeploy Refresh

Overview Inputs Outputs Template

We'd love your feedback! →

✓ Your deployment is complete

Deployment name: ArmisVaultTest2 Start time: 7/21/2022, 4:05:05 PM
Subscription: Pay-As-You-Go Correlation ID: cf79935b-db05-45c1-a35d-b25ce4e2dd85
Resource group: [Lansweeper_18033](#)

Deployment details (Download)

Next steps

[Go to resource](#)

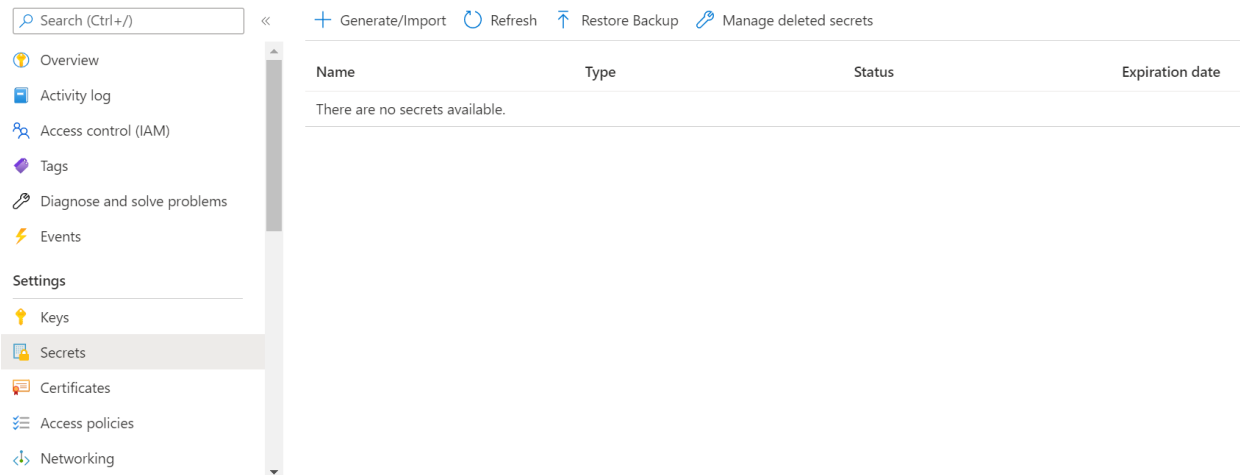
h. Click on “Go to resource” button

i. Click on “Secrets” from left menu

[Home](#) > [ArmisVaultTest2](#) > [ArmisVaultTest2](#)

ArmisVaultTest2 | Secrets

Key vault



Search (Ctrl+/) << + Generate/Import Refresh Restore Backup Manage deleted secrets

Name	Type	Status	Expiration date
There are no secrets available.			

Overview Activity log Access control (IAM) Tags Diagnose and solve problems Events

Settings

Keys Secrets Certificates Access policies Networking

j. Click on “Generate/Import” button

[Home](#) > [ArmisVaultTest2](#) > [ArmisVaultTest2](#) >

Create a secret ...

Upload options

Manual



Name * ⓘ

Value * ⓘ

Enter the secret.

Content type (optional)

Set activation date ⓘ

Set expiration date ⓘ

Enabled

Yes

No

Tags

0 tags

Create

- k. Provide Name of secret
- l. Provide Value for secret
- m. Click on create button

[Home](#) > [ArmisVaultTest2](#) > [ArmisVaultTest2](#) >

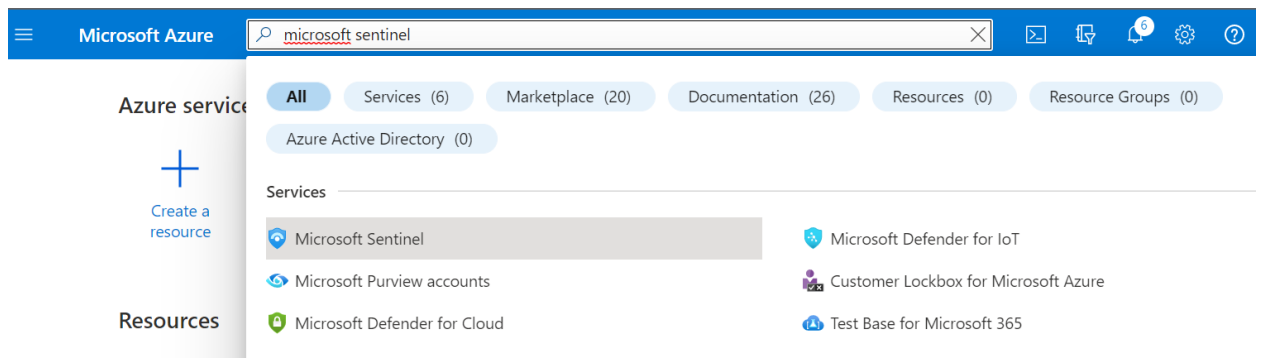
Create a secret ...

Upload options	<div>Manual</div>
Name * ⓘ	<div>ArmisAPISecretKey</div>
Value * ⓘ	<div>..... ✓</div>
Content type (optional)	<div></div>
Set activation date ⓘ	<div><input type="checkbox"/></div>
Set expiration date ⓘ	<div><input type="checkbox"/></div>
Enabled	<div>Yes No</div>
Tags	<div>0 tags</div>

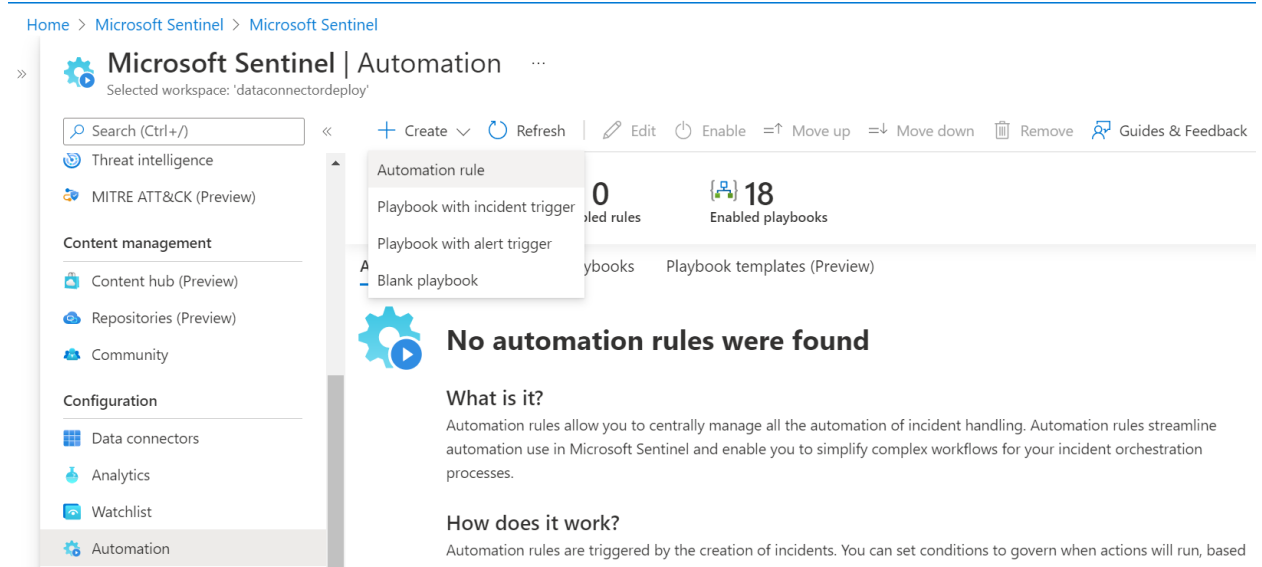
Create

4. Create Automation Rule which Runs Logic App

- a. Go to “Microsoft Sentinel” service



- b. Go to the Sentinel Workspace in which you want to create Automation Rule
- c. Go to “Automation” from the left menu by clicking on it
- d. From the **Automation** blade in the Microsoft Sentinel navigation menu, select **Create** from the top menu and choose the **Automation rule**.



e. The Create new automation rule panel opens. Enter a name for your rule.

f. Enter below information

- **Automation Rule Name** : Enter automation rule name

Automation rule name

ArmisUpdateAlertStatusAutomationRule ✓

- **Trigger** : From the Trigger drop-down, select “When incident is updated (Preview)”

Trigger

When incident is updated (Preview) ▼

- **Add conditions** :
 - If Analytic rule name :
 - Select “**Contains**” and “**All**”
 - Click on + Add Condition :
 - Select from dropdown value as “**Status**”, “**Changed To**” and “**New, Active, Closed**”

Conditions

If

Analytic rule name

Contains ▼

All ▼

And

Status ▼

Changed To ▼


3 selected ▼




+ Add condition

- **Add actions :**
 - Choose “Run playbook” from dropdown
 - Select Playbook which we want to execute when the automation rule condition gets true.

Actions ⓘ

Run playbook 

 ArmisUpdateAlertStatus
Pay-As-You-Go / Lansweeper_18033 

+ Add action


g. Click on “Apply” button to create automation rule

Create new automation rule

Status  Changed To  3 selected 

+ Add condition

Actions ⓘ

Run playbook 

 ArmisUpdateAlertStatus
Pay-As-You-Go / Lansweeper_18033 

+ Add action

Rule expiration ⓘ

Indefinite  Time

Order ⓘ

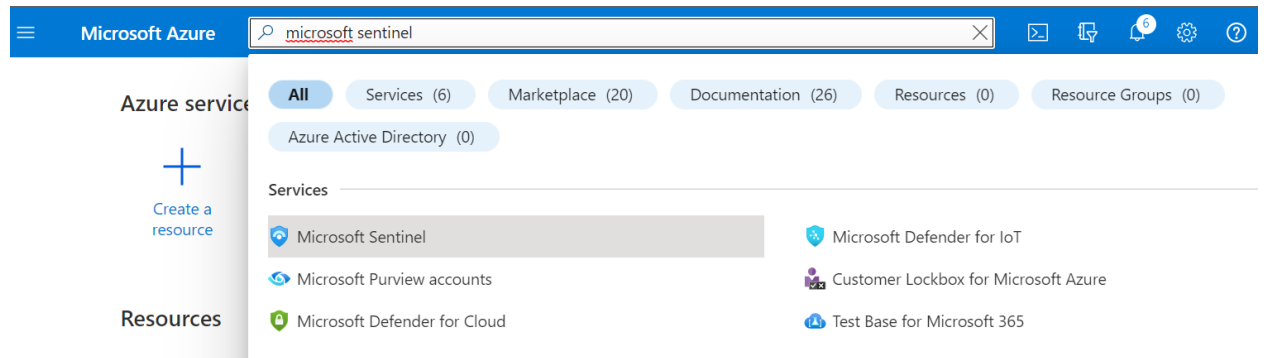
1

Apply

Cancel

5. Create Analytics Rule

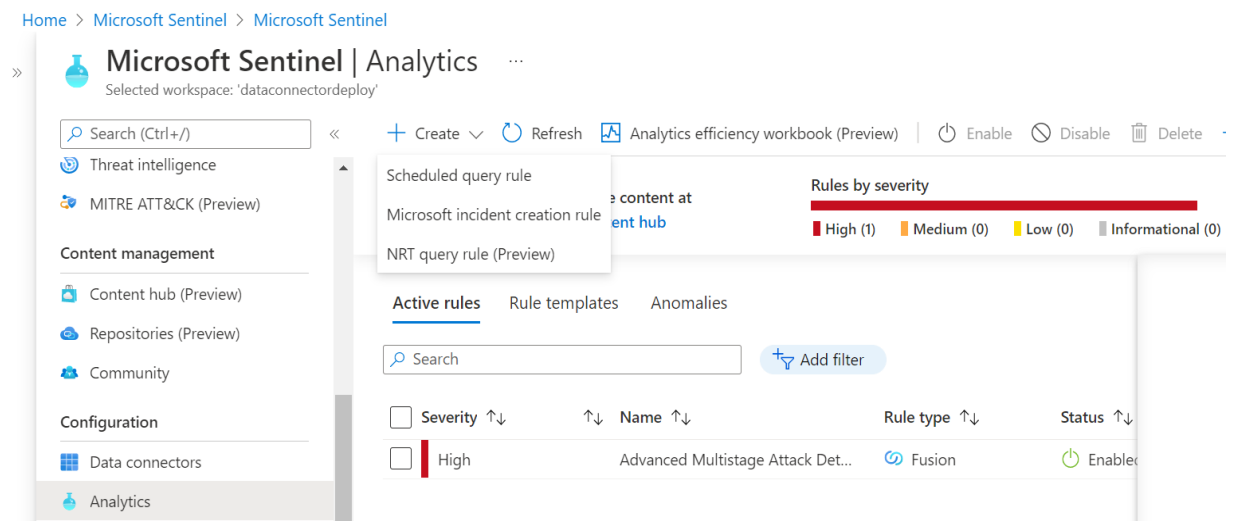
a. Go to “Microsoft Sentinel” service



b. Go to the Sentinel Workspace in which you want to create Analytics Rule

c. Go to “**Analytics**” from the left menu by clicking on it

d. From the **Analytics** blade in the Microsoft Sentinel navigation menu, select **Create** from the top menu and choose the “**Scheduled query rule**”.



e. Analytics rule wizard - **General tab**

- **Name** : Provide unique Analytics Rule Name
- **Description** : Provide Analytics Rule description
- **Other fields are not relevant to our use case so keep it as it is**

Analytics rule wizard - Create a new scheduled rule ...

General Set rule logic Incident settings Automated response Review and create

Create an analytics rule that will run on your data to detect threats.

Analytics rule details

Name *

ArmisUpdateAlertStatusAnalyticsRule



Description

This is the Analytics rule which used to generate incident.



Tactics and techniques

0 selected



Severity

Medium



Status

Enabled

Disabled

Next : Set rule logic >

- f. Define the rule query logic and configure settings
- **Rule Query** : Queries are written in Kusto Query Language (KQL)

General Set rule logic Incident settings Automated response Review and create

Define the logic for your new analytics rule.

Rule query

Any time details set here will be within the scope defined below in the Query scheduling fields.

```
ArmisAlertTest1907_CL  
| where Type == "ArmisAlertTest1907_CL" and status_s == "Unhandled"
```

[View query results >](#)

- **Custom details** : section to extract event data items from your query and surface them in the alerts produced by this rule, giving you immediate event content visibility in your alerts and incidents.

Alert enrichment

∨ Entity mapping

∧ Custom details

Here you can surface particular event parameters and their values in alerts that comprise those events, by adding key-value pairs below. In the Key field, enter a name of your choosing that will appear as the field name in alerts. In the Value field, choose the event parameter you wish to surface in the alerts from the drop-down list. [Learn more >](#)

alertID	alertId_d	∨	🗑
alertStatus	status_s	∨	🗑

+ Add new

∨ Alert details

- **Query scheduling** : Set as mentioned in below image

Query scheduling

Run query every *

5	✓	Minutes	∨
---	---	---------	---

Lookup data from the last * ⓘ

5	✓	Minutes	∨
---	---	---------	---

- Event grouping :
Select **“Trigger an alert for each event”** radio button

Alert threshold

Generate alert when number of query results

Is greater than



0

*

Event grouping

Configure how rule query results are grouped into alerts

☐

Group all events into a single alert

☒

Trigger an alert for each event



A single analytics rule can generate up to 150 alerts. If you choose "Trigger an alert for each event" and the rule query returns more than 150 events, the rule will generate 150 alerts, and the last alert will include a summary of all the events. [Learn more >](#)

Previous

Next : Incident settings >

- g. Configure the incident creation settings
 - No need to set/update any setting in this tab.
- h. Set automated responses and create the rule
 - **Automation rules(Preview)** : The grid displayed under automation rules(Preview) shows the automation rules that already apply to this analytics rule

General Set rule logic Incident settings Automated response Review and create

Automation rules (Preview)

View all automation rules that will be triggered by this analytics rule and create new automation rules.

+ Add new

Order	Automation rule name	Trigger	Action	Status
1	ArmisUpdateAlertStatusAutomationRule	Incident updated (Previe...	Run playbook 'ArmisUpdateAle...	Enabled

- i. Click on "Next : Review >" button
- j. It will validate the Analytics rule content and if the validation is successful, it will show a "Create" button at the bottom of the screen.
- k. Click on "Create" button

Analytics rule wizard - Create a new scheduled rule ...

✔ Validation passed.

General Set rule logic Incident settings Automated response Review and create

Analytics rule details

Name	ArmisUpdateAlertStatusAnalyticsRule
Description	This is the Analytics rule which used to generate incident.
Tactics and techniques	
Severity	Medium
Status	Enabled

Analytics rule settings

Previous

Create

I. The above step will create an Analytics rule.