Monitoring Microsoft Azure

Microsoft: Azure PowerPack version 108

Table of Contents

Chapter 1

Introduction

Overview

This manual describes how to monitor Microsoft Azure resources that are managed with Azure Resource Manager (ARM) in SL1 using the *Microsoft: Azure* PowerPack.

The following sections provide an overview of Microsoft Azure and the *Microsoft: Azure* PowerPack:

NOTE: ScienceLogic provides this documentation for the convenience of ScienceLogic customers. Some of the configuration information contained herein pertains to third-party vendor software that is subject to change without notice to ScienceLogic. ScienceLogic makes every attempt to maintain accurate technical information and cannot be held responsible for defects or changes in third-party vendor software. There is no written or implied guarantee that information contained herein will work for all third-party variants. See the End User License Agreement (EULA) for more information.

What is Azure?

Azure is a Microsoft service that provides both infrastructure and platform capabilities for cloud computing. Azure enables users to build, deploy, and manage applications and services using Microsoft data centers, and offers users numerous capabilities such as website hosting, virtual machine creation, data management, business analytics, and media services.

What Does the Microsoft: Azure PowerPack Monitor?

To monitor Microsoft Azure resources using SL1, you must install the *Microsoft: Azure* PowerPack. This PowerPack enables you to discover, model, and collect data about Azure resources.

The *Microsoft: Azure* PowerPack includes:

- Example credentials you can use as templates to create SOAP/XML credentials to connect to Azure
- Dynamic Applications to discover, model, and monitor performance metrics and/or collect configuration data for the following Azure resources:
    - Active Directory tenants
    - Application gateways
    - Application services
    - Function apps
    - Load balancers
    - Managed storage disks
    - Network security groups
    - Recovery Services vaults
    - Resource groups
    - Site recovery configurations
    - SQL databases
    - SQL servers
    - Storage accounts
    - Traffic Manager profiles
    - Virtual machine scale sets
    - Virtual machines
    - Virtual network subnets
    - Virtual networks
    - VPN gateways
    - Web apps
- Device Classes for each Azure data center location and all of the Azure resources SL1 monitors
- Event Policies and corresponding alerts that are triggered when Azure resources meet certain status criteria
- Run Book Action and Automation policies that can automate certain Azure monitoring processes

What are Azure Locations?

An Azure location is an individual data center located in a specific geographic locale. The Dynamic Applications in the *Microsoft: Azure* PowerPack create a "location" component device for each discovered data center location.

The PowerPack supports the following Azure data center locations:

- Australia Central (Canberra)
- Australia Central 2 (Canberra)
- Australia East (New South Wales)
- Australia Southeast (Victoria)
- Brazil South (Sao Paulo)
- Canada Central (Toronto)
- Canada East (Quebec)
- Central India (Pune)
- Central US (Iowa)
- China East (Shanghai)
- China East 2 (Shanghai)
- China North (Beijing)
- China North 2 (Beijing)
- East Asia (Hong Kong)
- East US (Virginia)
- East US 2 (Virginia)
- France Central (Paris)
- France South (Marseille)
- Germany Central (Frankfurt)
- Germany North
- Germany Northeast (Magdeburg)
- Germany West Central
- Japan East (Saitama)
- Japan West (Osaka)
- Korea Central (Seoul)
- Korea South (Busan)
- North Central US (Illinois)
- North Europe (Ireland)
- South Central US (Texas)

- South India (Chennai)
- Southeast Asia (Singapore)
- US DoD Central (for Microsoft Azure Government only)
- US DoD East (for Microsoft Azure Government only)
- US Gov Arizona (for Microsoft Azure Government only)
- US Gov Iowa (for Microsoft Azure Government only)
- US Gov Texas (for Microsoft Azure Government only)
- US Gov Virginia (for Microsoft Azure Government only)
- UK South (London)
- UK West (Cardiff)
- West Central US
- West Europe (Netherlands)
- West India (Mumbai)
- West US (California)
- West US 2

Installing the Microsoft: Azure PowerPack

Before completing the steps in this manual, you must import and install the latest version of the *Microsoft: Azure* PowerPack.

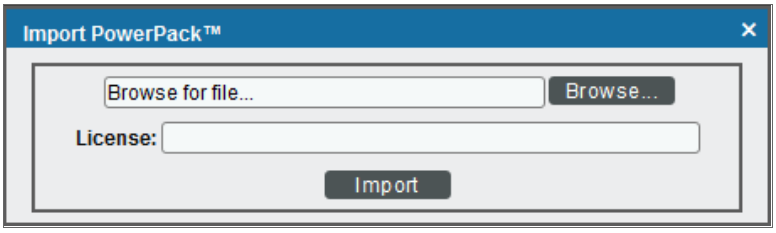
NOTE: The following instructions describe how to install the *Microsoft: Azure* PowerPack for the first time. If you are upgrading to the latest version from a previous version, see the ***Microsoft: Azure* PowerPack** Release Notes for specific upgrade instructions.

TIP: By default, installing a new version of a PowerPack overwrites all content in that PowerPack that has already been installed on the target system. You can use the ***Enable Selective PowerPack Field Protection*** setting in the **Behavior Settings** page (System > Settings > Behavior) to prevent new PowerPacks from overwriting local changes for some commonly customized fields. (For more information, see the ***System Administration*** manual.)

To download and install a PowerPack:

1. Download the PowerPack from the ScienceLogic Customer Portal.
2. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
3. In the **PowerPack Manager** page, click the **[Actions]** button, then select *Import PowerPack*.

4. The **Import PowerPack** dialog box appears:



5. Click the **[Browse]** button and navigate to the PowerPack file.
6. When the **PowerPack Installer** modal page appears, click the **[Install]** button to install the PowerPack.

> **NOTE:** If you exit the **PowerPack Installer** modal page without installing the imported PowerPack, the imported PowerPack will not appear in the **PowerPack Manager** page. However, the imported PowerPack will appear in the **Imported PowerPacks** modal page. This page appears when you click the **[Actions]** menu and select *Install PowerPack*.

Chapter 2

Configuring Azure for Monitoring

Overview

The following sections describe how to configure Microsoft Azure resources for monitoring by SL1 using the *Microsoft: Azure* PowerPack:

> **NOTE**: The *Microsoft: Azure* PowerPack can monitor Microsoft Azure resources, Microsoft Azure Government resources, and Microsoft Azure resources in Germany and China regions.

Configuring an Azure Active Directory Application

To create a SOAP/XML credential that allows SL1 to access Microsoft Azure, you must provide the following information about an Azure application that is already registered with an Azure AD tenant:

- Application ID
- Subscription ID (if monitoring a single subscription)
- OAuth 2.0 token endpoint URL
- Tenant ID
- Secret key

To capture the above information, you must first create (or already have) an application that is registered with Azure Active Directory. The registered application must have Reader access. You can then enter the required information about the application when configuring the SOAP/XML credential in SL1. The registered application and the ScienceLogic credential allow SL1 to retrieve information from Microsoft Azure.

TIP: For details on registering an Azure application with Azure AD, see https://docs.microsoft.com/en-us/azure/active-directory/develop/active-directory-integrating-applications.

Creating an Active Directory Application in the Azure Portal

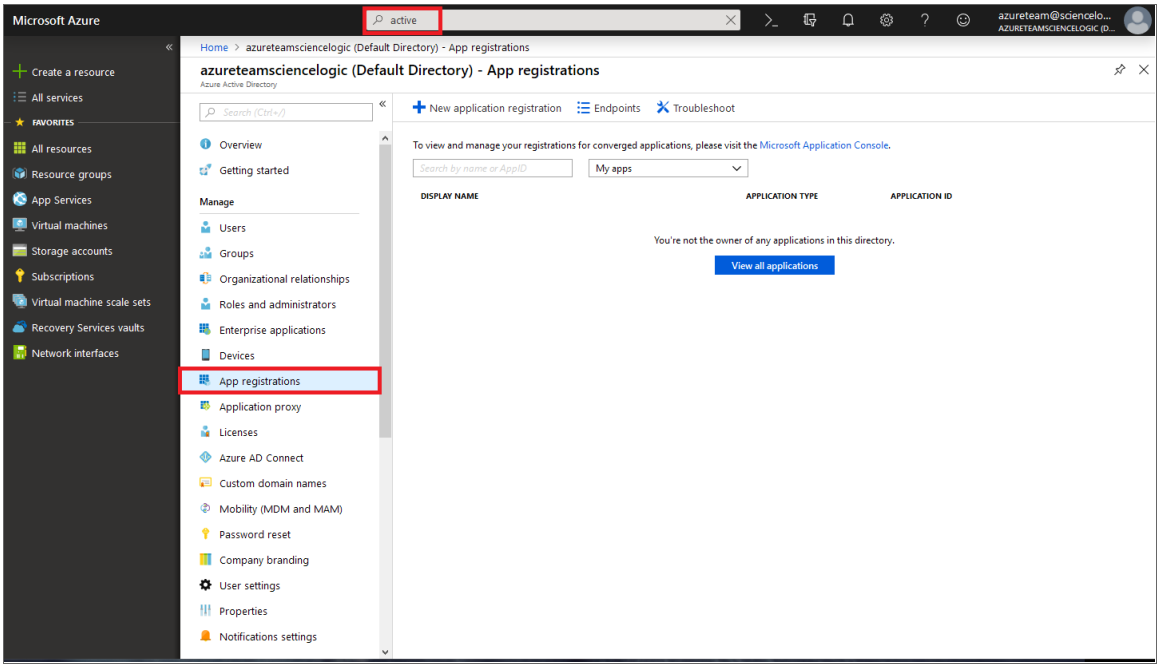
When configuring a SOAP/XML credential in SL1, you must provide the application ID, subscription ID, OAuth 2.0 token endpoint URL, tenant ID, and secret key of an application that is registered with Azure Active Directory. You will use this registered application to authenticate your Azure account.

NOTE: You must have Service Administrator rights to create an Azure Active Directory application.

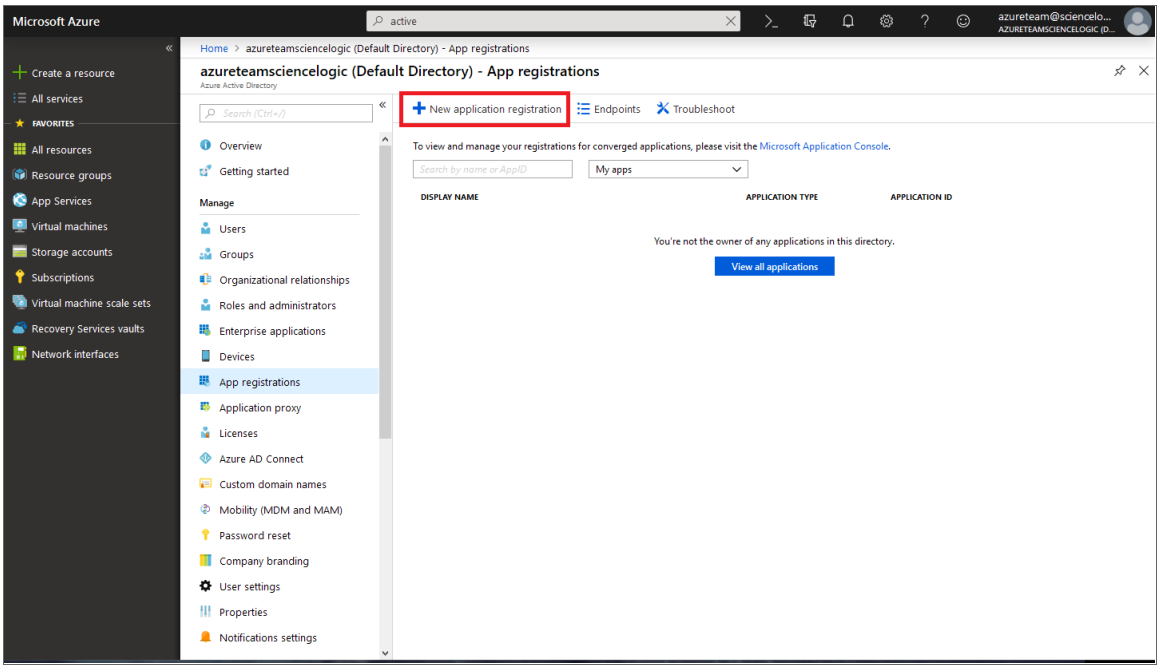
To create an application in Azure and register it with Azure Active Directory:

1. Log in to the Azure portal at https://portal.azure.com, and type "active directory" in the **Search** field at the top of the window.

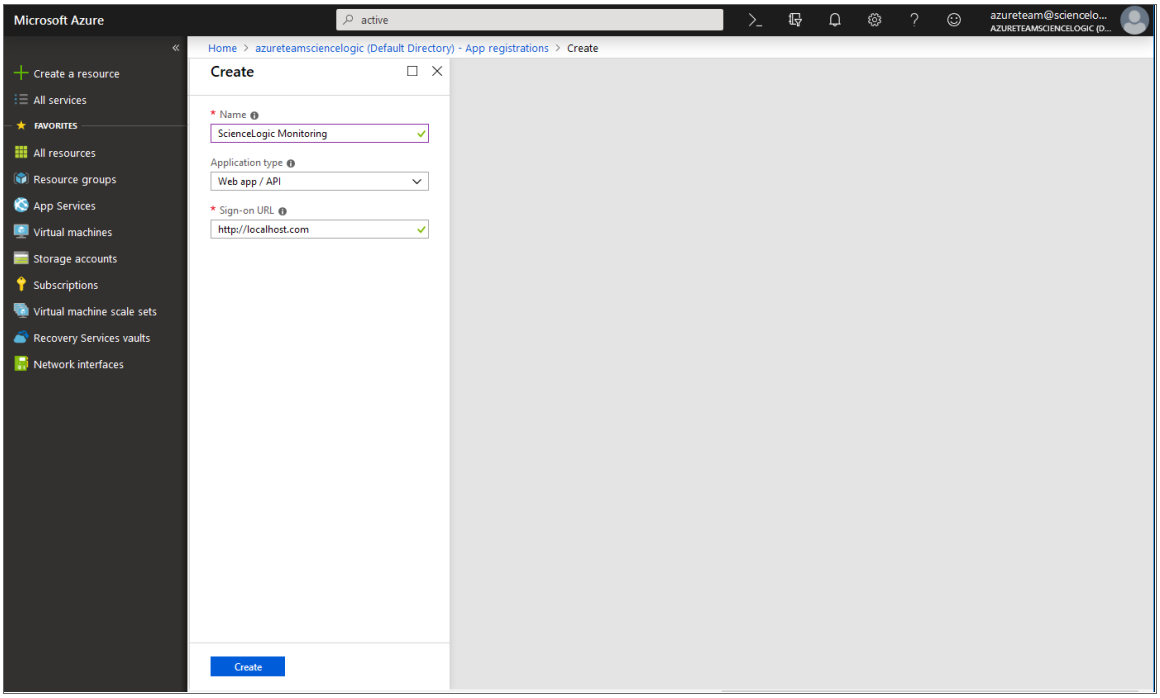
2. From the search results, select *Azure Active Directory*, and then click **App registrations**. The **App registrations** page appears:



3. Click the **[New application registration]** button.



4. Type a ***Name*** for the application and select *Web app / API* in the ***Application Type*** field. In the ***Sign-On URL*** field, type any valid URL, and then click the **[Create]** button.



5. A message appears confirming that your application was added.

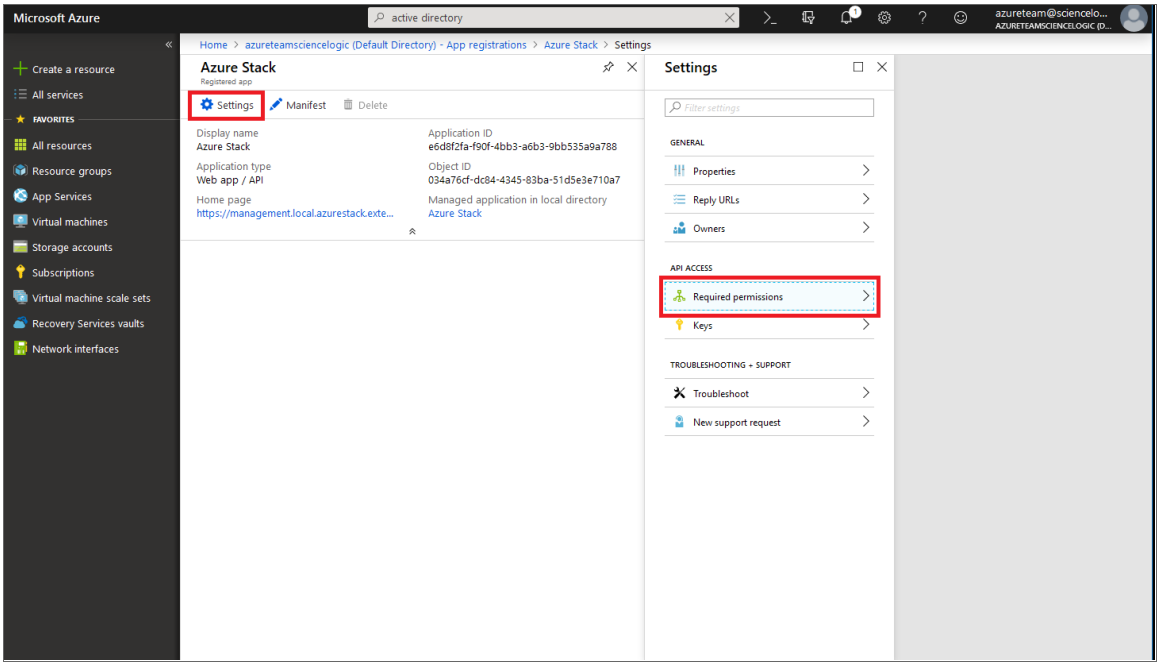
Adding the Microsoft Graph and Windows Azure Active Directory APIs

You must add the Microsoft Graph and Windows Azure Active Directory application programmable interfaces (APIs) to the Azure Active Directory application you will use to authenticate your Azure account. At a minimum, the Microsoft Graph and Windows Azure Active Directory APIs must have permission to read directory data.

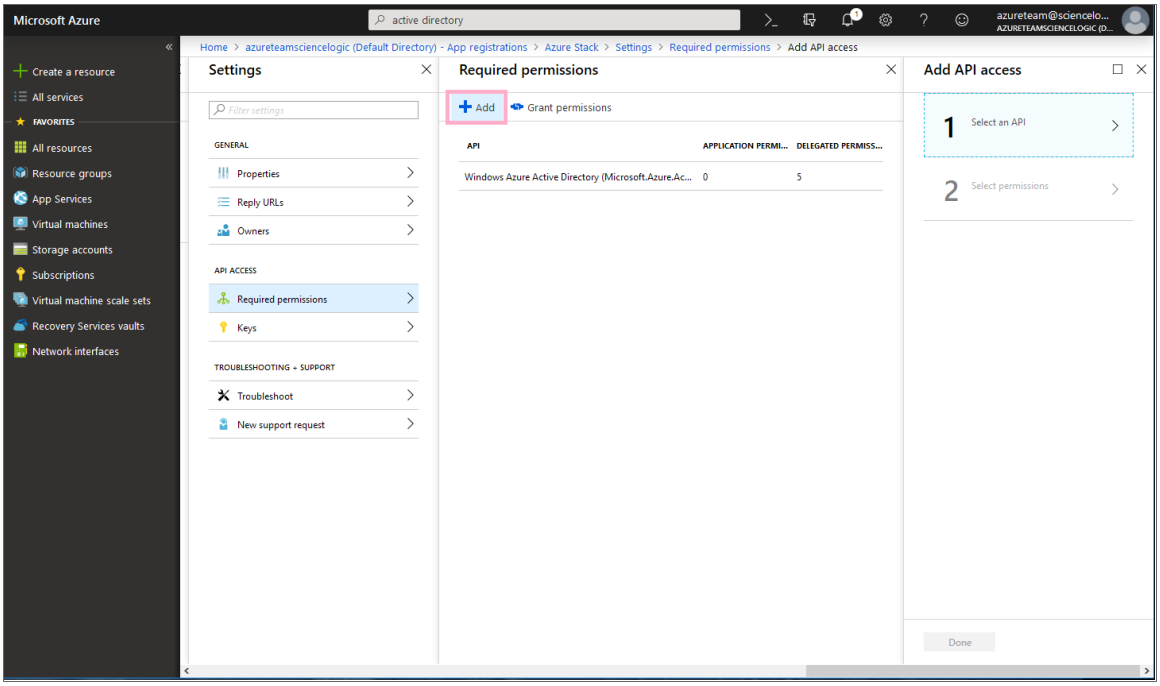
To add the Microsoft Graph and Windows Azure Active Directory APIs:

1. In the **Search** field of the Azure portal (https://portal.azure.com), type "active directory".

2. Click **[App registrations]**, and then click on the name of the Azure Active Directory application you will use to authenticate your Azure account.
3. If needed, click the **[Settings]** button, and then click **Required permissions** in the **Settings** pane.

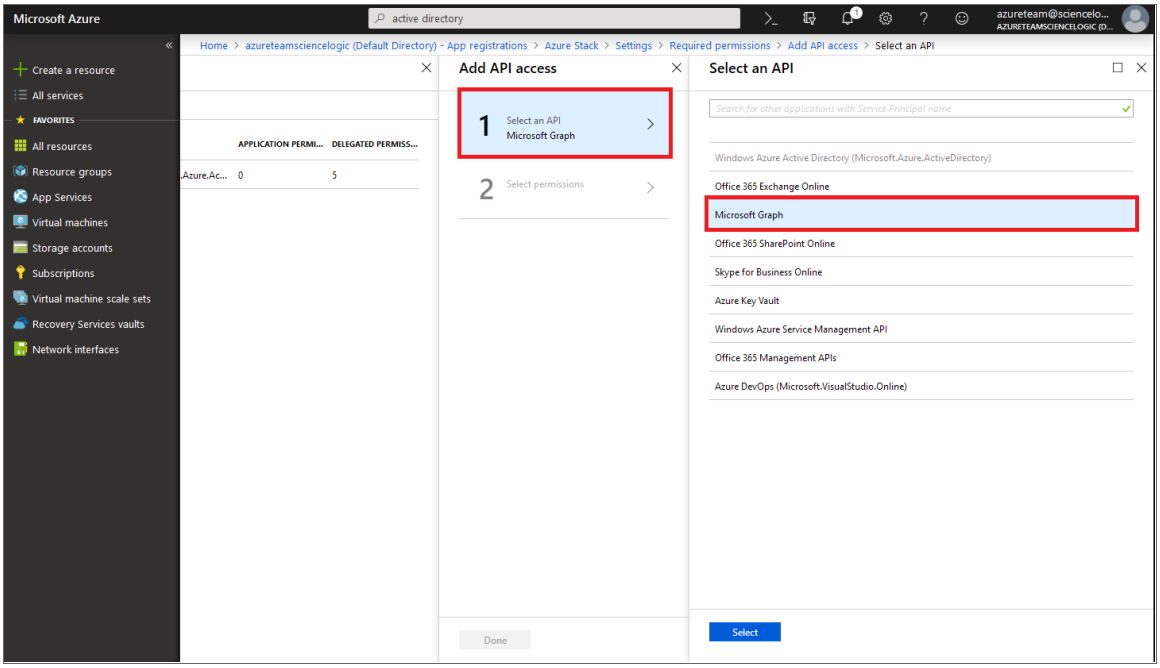


4. In the **Required permissions** pane, click **[Add]**.

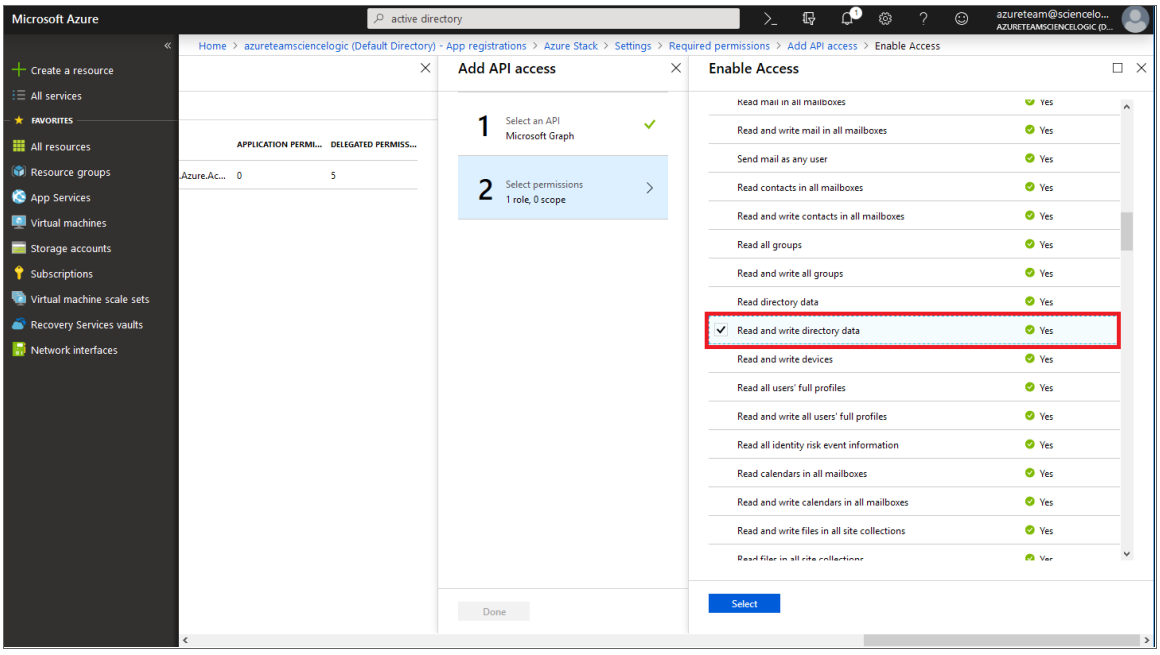


5. In the **Add API access** pane, click **[Select an API]**.

6. In the **Select an API** pane, select *Microsoft Graph* and click **[Select]**.



7. In the **Enable access** pane, under **Application Permissions**, select *Read directory data*. Click **[Select]**.



8. Click **[Done]** to save.
9. Repeat these steps for the Windows Azure Active Directory API. To do so, choose *Windows Azure Active Directory* (rather than *Microsoft Graph*) in step 6.

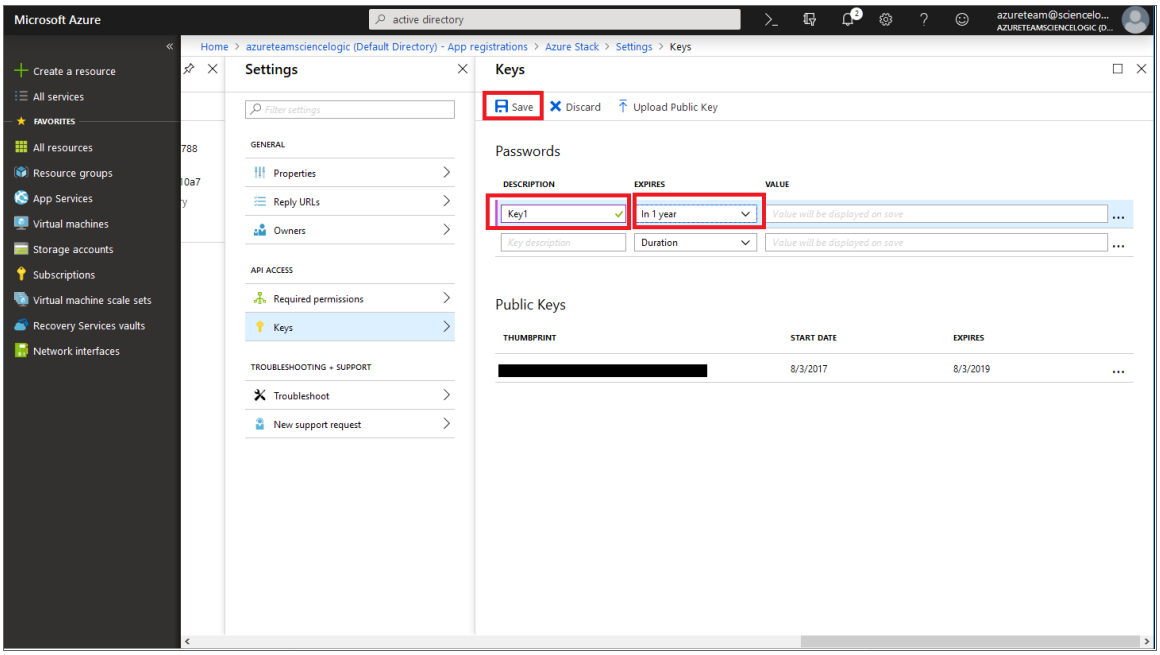
10. After you add the Microsoft Graph and Windows Azure Active Directory APIs, return to the **Required permissions** pane and click **[Grant Permissions]**.

Generating the Secret Key

When configuring a SOAP/XML credential for Azure in SL1, you need to provide a secret key for the Azure Active Directory application that you will use to authenticate your account.

To generate a secret key:

1. Log in to the Azure portal at https://portal.azure.com, and type "active directory" in the **Search** field at the top of the window.
2. From the search results, select *Azure Active Directory*, and then click **App registrations**.
3. Select the app and then click **[Settings]**.
4. In the **Settings** pane, click **[Keys]**.



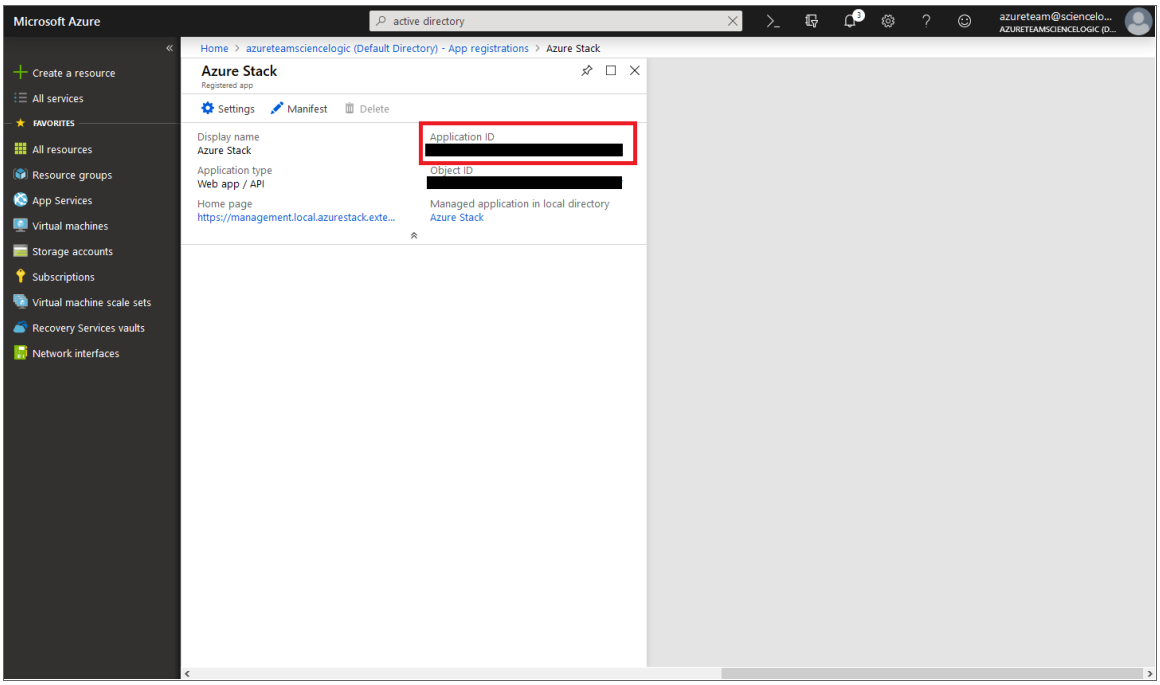
5. In the **Keys** pane, type a name in the ***Description*** field and select a duration in the ***Expires*** field.
6. Click **[Save]** to generate the secret key. A new key value displays in the ***Value*** field.
7. Copy and save the key value.

Locating the Application ID

When configuring a SOAP/XML credential for Azure in SL1, you need to provide the Application ID of the Azure Active Directory application you will use to authenticate your Azure account.

To locate the Application ID:

1. Log in to the Azure portal at https://portal.azure.com, and type "active directory" in the **Search** field at the top of the window.
2. From the search results, select *Azure Active Directory*, and then click **App registrations**.
3. Click the name of the Active Directory application you will use to authenticate your Azure account. The Application ID appears in the **Essentials** section. The detail pane for the app appears:



4. Copy and save the value in the ***Application ID*** field.

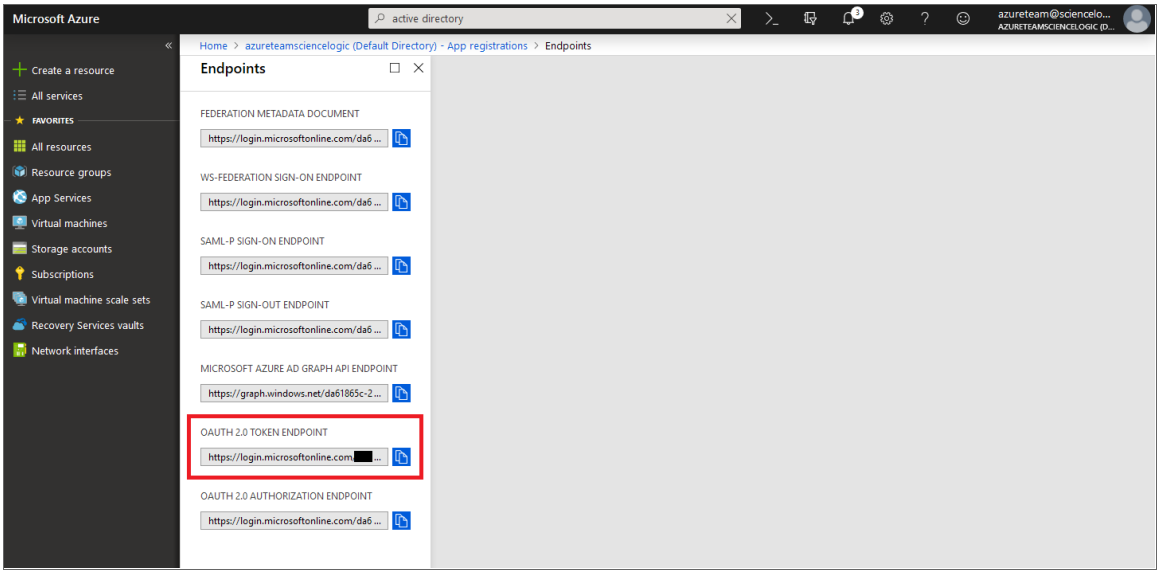
Locating the OAuth 2.0 Token Endpoint URL and the Tenant ID

When configuring a SOAP/XML credential for Azure in SL1, you need to provide the OAuth 2.0 token endpoint URL and the tenant ID of the Azure Active Directory application you will use to authenticate your Azure account.

To locate the OAuth 2.0 token endpoint URL and the tenant ID:

1. Log in to the Azure portal at https://portal.azure.com, and type "active directory" in the **Search** field at the top of the window.
2. From the search results, select *Azure Active Directory*, and then click **App registrations**.

3. Click the **[Endpoints]** button. The ***OAUTH 2.0 TOKEN ENDPOINT*** URL appears in the **Endpoints** pane.



4. Copy and save the ***OAUTH 2.0 TOKEN ENDPOINT*** URL.
5. The OAuth 2.0 token endpoint URL contains a GUID, a string of characters in the middle of the URL.

   For example, in the following OAuth 2.0 token endpoint URL, the GUID is in bold:
   https://login.microsoftonline.com/**eg58975d-1953-5509-4654-b5d4bbga22ceg**/oauth2/token

   Azure uses this GUID as the tenant ID. Copy and save the GUID.

Locating the Subscription ID

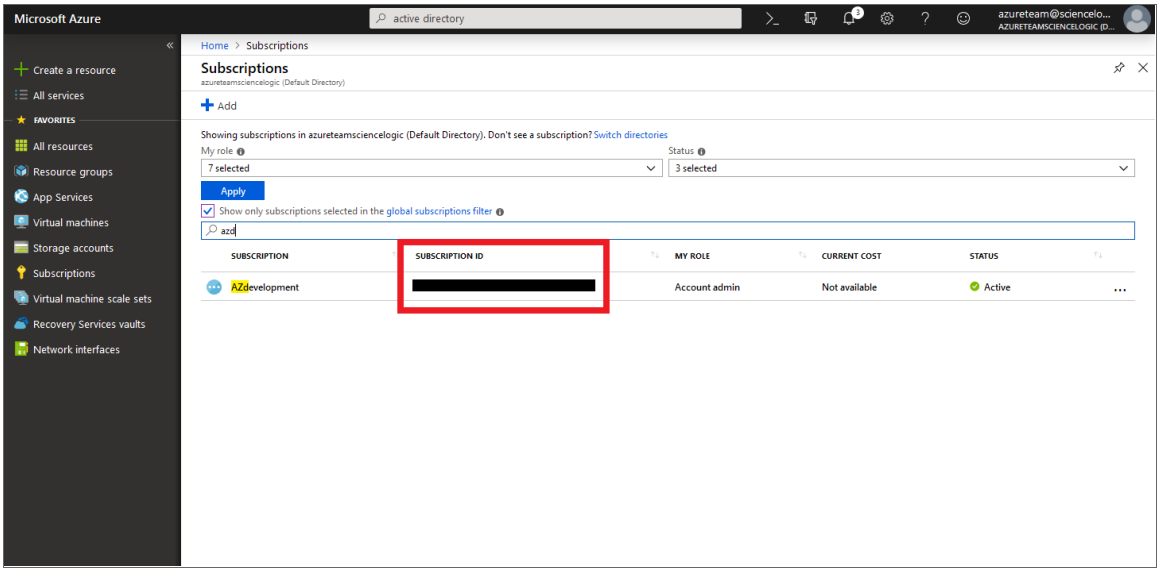
If you are monitoring only a single Azure subscription, you must provide the Subscription ID of the Azure Active Directory application you will use to authenticate your account when you configure your SOAP/XML credential for Azure in SL1.

NOTE: If you are monitoring an account with multiple child subscriptions, you can skip this section.

To locate the Subscription ID:

1. In the left pane of the Azure portal (https://portal.azure.com), click **[Subscriptions]**.

2. Copy and save the ***Subscription ID*** of the subscription where you created the Azure Active Directory application you will use to authenticate your account.



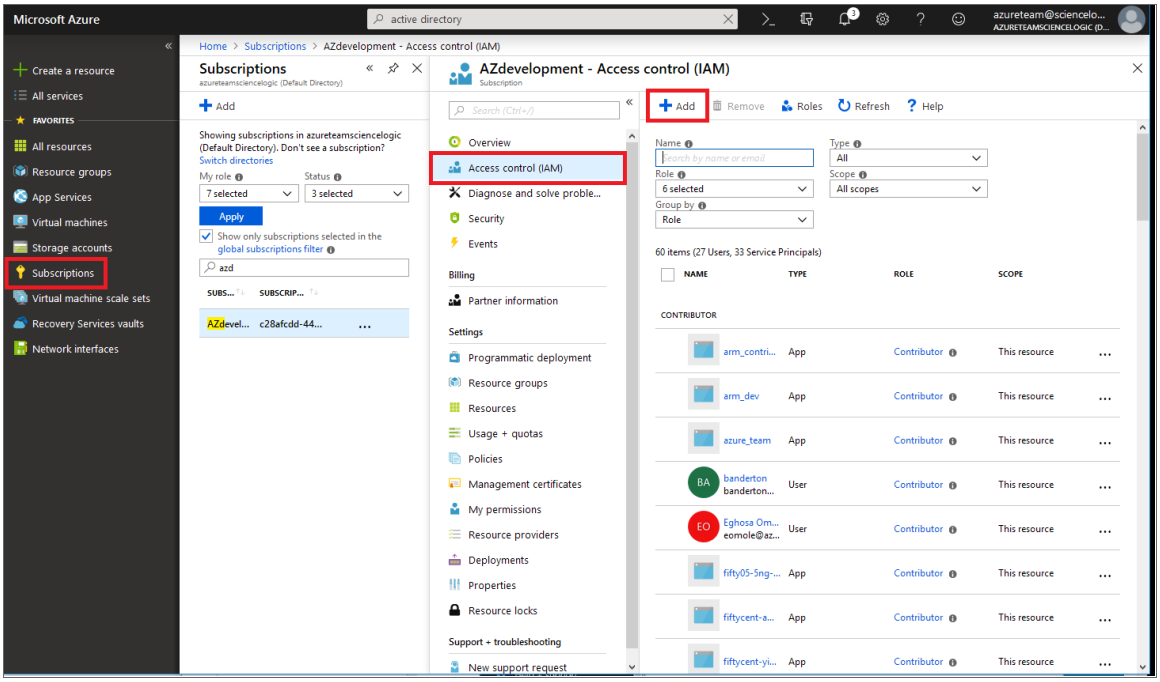
Adding Reader Access to the Active Directory Application

To allow ScienceLogic to access your Azure account, you must specify the type of access the user whose information you will use in your SOAP/XML credential has to the Active Directory application used to authenticate your account. Use the ***Reader*** access role, which is a read-only user that can view everything but cannot make changes.

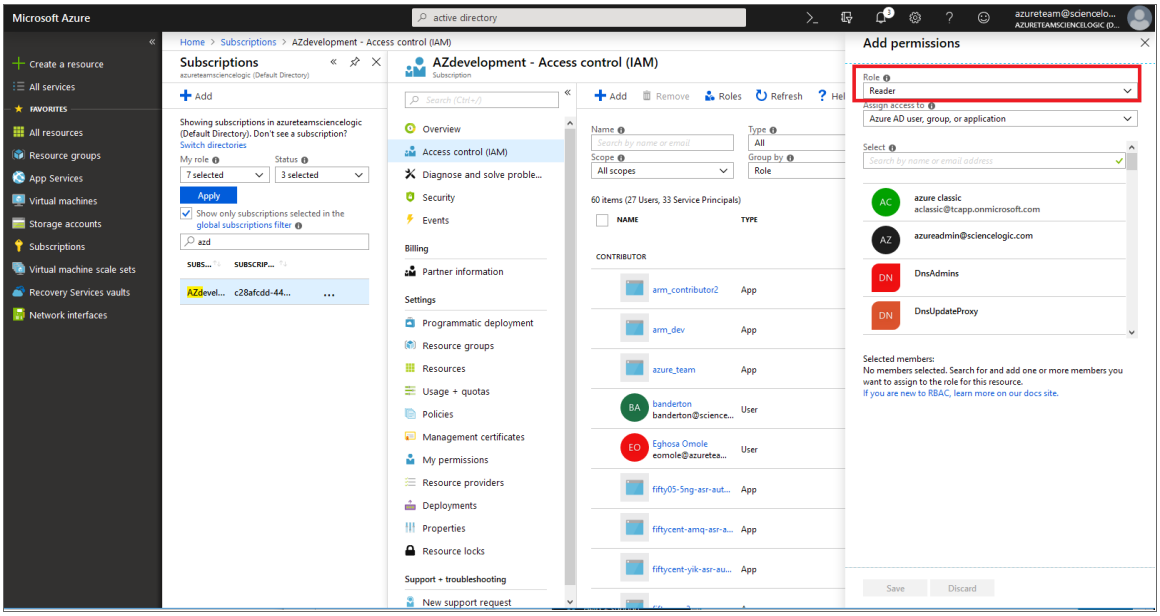
To specify the access role to the Azure Active Directory application:

1. In the left pane of the Azure Portal (https://portal.azure.com), click **[Subscriptions]**.
2. Click the name of your subscription, and then click **[Access control (IAM)]**.

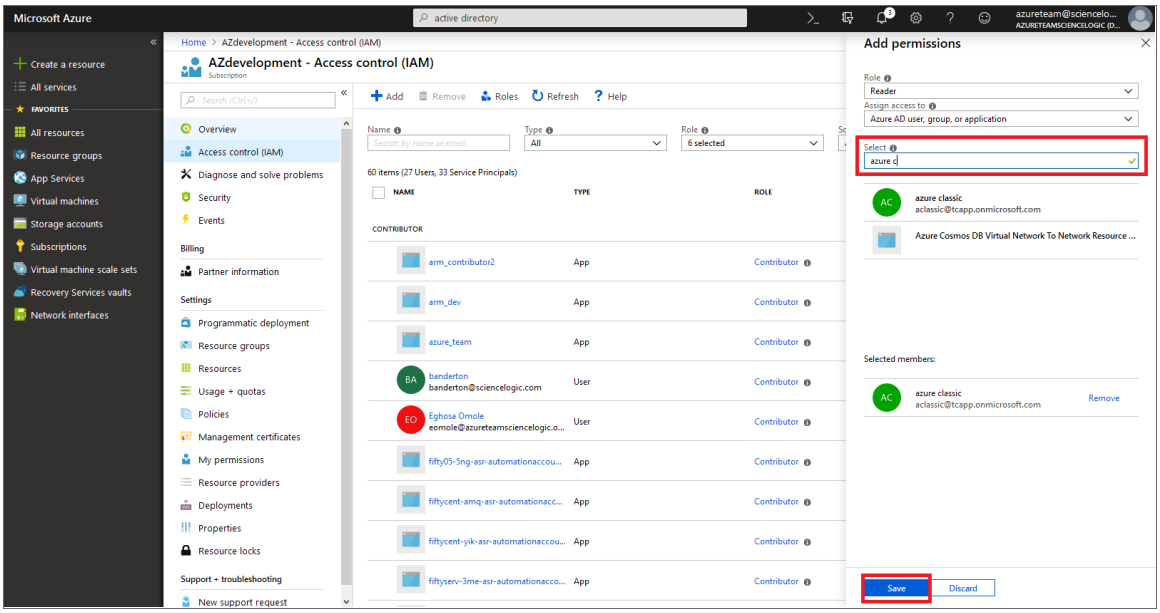
3. In the **Access Control (IAM)** pane, click **[Add]**.



4. In the **Add permissions** pane, select *Reader* from the ***Role*** field.



5. In the ***Select*** field, type the name of the Azure Active Directory application you will use to authenticate your account.



6. Select the application from the search results and click **[Save]**.

Setting Up a Proxy Server

Depending on your needs, you can optionally enable SL1 to connect to Azure through a third-party proxy server such as SQUID. With this configuration, SL1 connects to the proxy server, which then connects to Azure. Azure relays information to the proxy server and SL1 then retrieves that information from the proxy.

NOTE: You can connect to Azure via a proxy server regardless of whether you are monitoring a single subscription or an account with multiple child subscriptions. You can connect to Microsoft Azure, Microsoft Azure Government, and Microsoft Azure Germany and China regions via a proxy server.

NOTE: The *Microsoft: Azure* PowerPack is certified to work with SQUID version 3.5.12 proxy servers.

If you choose to use a proxy server, configure the third-party proxy server based on the third-party documentation. Depending on the type of authentication you require, you might need to specify a user name and password for the proxy server configuration. Also, make a note of the port you opened for the configuration, as this information is needed when creating the SOAP/XML credential.

NOTE: To configure the third-party proxy server, you must have openssh-server.x86_64 and telnet installed.

Creating a SOAP/XML Credential for Azure

After you note the application ID, subscription ID, OAuth 2.0 token endpoint URL, tenant ID, and secret key of the application (that is registered with Azure Active Directory) that you will use to authenticate your Azure account, you can create a SOAP/XML credential for Azure in SL1. This credential allows the Dynamic Applications in the *Microsoft: Azure* PowerPack to communicate with your Azure subscriptions.

If you want to connect to your Azure account through a third-party proxy server, you must also add the proxy information in the credential. This applies to Microsoft Azure, Microsoft Azure Government, and the Microsoft Azure German and Chinese regions.

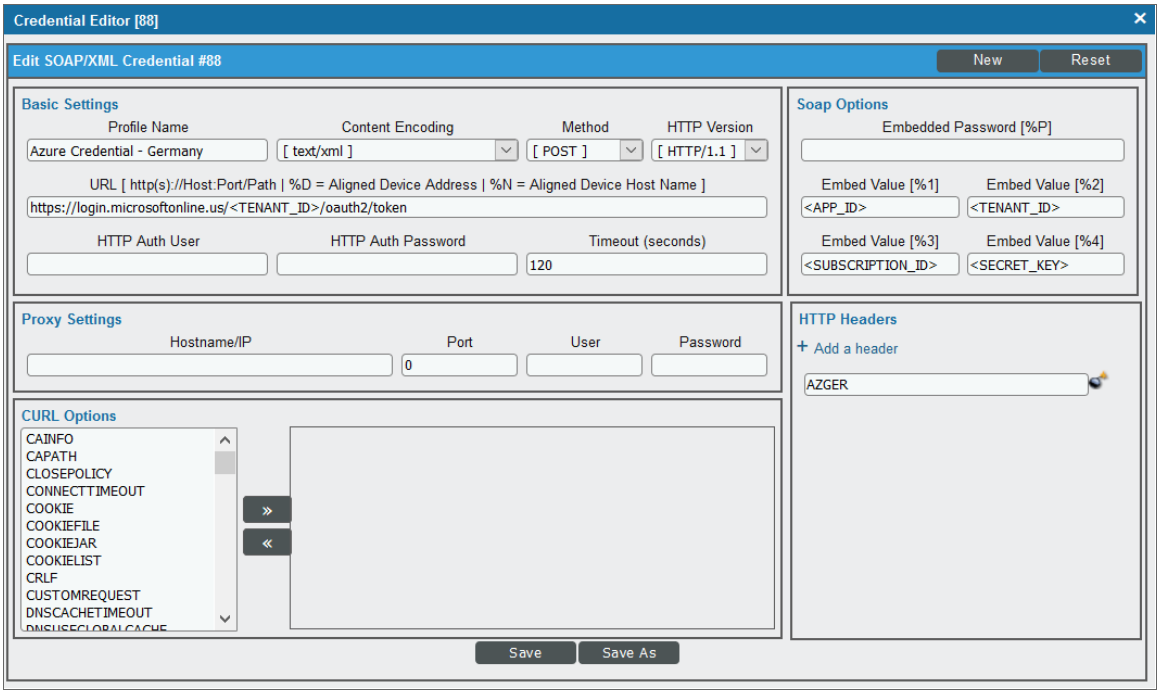
The *Microsoft: Azure* PowerPack includes multiple sample credentials you can use as templates for creating SOAP/XML credentials for Azure. They are:

- **Azure Credential - China**, for users who connect to an Azure data center in a Chinese region
- **Azure Credential - Germany**, for users who connect to an Azure data center in a German region (requires a subscription in Germany or Europe)
- **Azure Credential - Government**, for users who subscribe to Microsoft Azure Government
- **Azure Credential - Proxy**, for users who connect to Azure through a third-party proxy server
- **Azure Credential - SOAP/XML**, for all other users

To create a SOAP/XML credential for Azure:

1. Go to the **Credential Management** page (System > Manage > Credentials).

2. Locate the sample credential you want to use and then click its wrench icon (). The **Edit SOAP/XML Credential** modal page appears:



3. Enter values in the following fields:

Basic Settings

- ***Profile Name***. Type a new name for the Azure credential.
- ***Content Encoding***. Select *text/xml*.
- ***Method***. Select *POST*.
- ***HTTP Version***. Select *HTTP/1.1*.
- ***URL***. Type the OAuth 2.0 token endpoint URL for the Azure Active Directory application.
- ***HTTP Auth User***. Leave this field blank.
- ***HTTP Auth Password***. Leave this field blank.
- ***Timeout (seconds)***. Type "120".

Proxy Settings

- ***Hostname/IP***. If you are connecting to Azure via a proxy server, type the server's hostname or IP address. Otherwise, leave this field blank.
- ***Port***. If you are connecting to Azure via a proxy server, type the port number you opened when *setting up the proxy server*. Otherwise, leave this field blank.

- ***User***. If you are connecting to Azure via a proxy server using basic authentication, type the server's administrator username. Otherwise, leave this field blank.
- ***Password***. If you are connecting to Azure via a proxy server using basic authentication, type the server's administrator password. Otherwise, leave this field blank.

CURL Options

- ***CURL Options***. Do not make any selections in this field.

SOAP Options

- ***Embedded Password [%P]***. Leave this field blank.
- ***Embed Value [%1]***. Type the Application ID for the Azure Active Directory application.
- ***Embed Value [%2]***. Type the Tenant ID for the Azure Active Directory application.
- ***Embed Value [%3]***. If you are monitoring only a single Azure subscription, type the Subscription ID for the Azure Active Directory application. If you are monitoring multiple subscriptions, leave this field blank.
- ***Embed Value [%4]***. Type the secret key for the Azure Active Directory application.

HTTP Headers

- ***HTTP Headers***. Leave this field blank, unless one of the following scenarios applies to you:
  - If you are using Microsoft Azure Government, this field contains the text "AZGOV".
  - If you are monitoring Microsoft Azure resources in Germany, this field contains the text "AZGER".
  - If you are monitoring Microsoft Azure resources in China, this field contains the text "AZCHINA".

4. Click **[Save As]**.
5. In the confirmation message, click **[OK]**.

Load-Balancing an Account with Multiple Subscriptions

When monitoring an account with multiple child subscriptions, instead of discovering all child subscriptions in a single dynamic component map under their parent account, you can load-balance subscriptions and their components across multiple Data Collectors.

To do this:

- The Collector Group that discovers a group of subscriptions can contain only one Data Collector. You cannot use multiple Data Collectors to discover the Azure components in a single dynamic component map or discover the same device in multiple dynamic component maps.
- To group multiple Azure subscriptions into a single dynamic component map, you need to create a shared credential for that group of subscriptions.
- To create the credential:
  - Perform all of the steps in the section on *Configuring an Azure Active Directory Application*.

    - Align each subscription in the group with the same application that you registered with Azure AD.
    - In the credential, enter the application ID in the ***Embed Value [%1]*** field.
    - In the credential, leave the ***Embed Value [%3]*** field blank.
- During discovery, use this credential to discover the group of subscriptions.
- During discovery, specify the Data Collector you want to use for the group of subscriptions.
- The discovered subscriptions will reside in a common dynamic component map.
- Repeat these steps for each group of subscriptions.

Testing the Azure Credential

SL1 includes a Credential Test for Microsoft Azure. Credential Tests define a series of steps that SL1 can execute on demand to validate whether a credential works as expected.

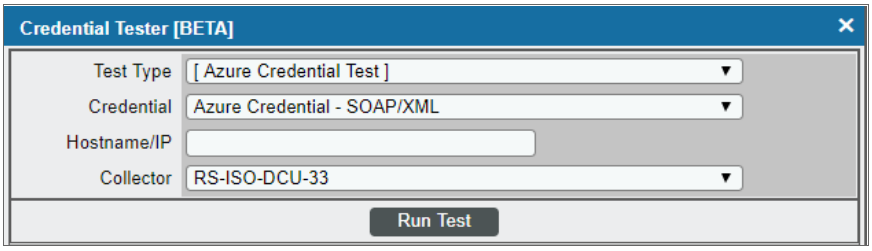
The Azure Credential Test can be used to test a SOAP/XML credential for monitoring Azure using the Dynamic Applications in the *Microsoft: Azure* PowerPack. The Azure Credential Test performs the following steps:

- ***Test Port Availability***. Performs an NMAP request to test the availability of the Azure endpoint HTTPS port.
- ***Test Name Resolution***. Performs an nslookup request on the Azure endpoint.
- ***Make connection to Azure account***. Attempts to connect to the Azure service using the account specified in the credential.
- ***Make Azure Active Directory Request***. Verifies that the Azure Active Directory service can be queried.

NOTE: The Azure Credential Test has not been certified for use with credentials that use a proxy server.

To test the Azure credential:

1. Go to the **Credential Test Management** page (System > Customize > Credential Tests).
2. Locate the **Azure Credential Test** and click its lightning bolt icon ( ). The **Credential Tester** modal page appears:



3. Supply values in the following fields:
    - ***Test Type***. This field is pre-populated with the credential test you selected.

- ***Credential***. Select the credential to test. This drop-down list includes only credentials that you have access to that can be tested using the selected credential test.
- ***Hostname/IP***. Leave this field blank.
- ***Collector***. Select the All-In-One Appliance or Data Collector that will run the test.

4. Click the **[Run Test]** button. The **Test Credential** window appears, displaying a log entry for each step in the credential test. The steps performed are different for each credential test. The log entry for each step includes the following information:

   - ***Step***. The name of the step.
   - ***Description***. A description of the action performed during the step.
   - ***Log Message***. The result of the step for this credential test.
   - ***Status***. Whether the result of this step indicates the credential or the network environment is configured correctly (Passed) or incorrectly (Failed).
   - ***Step Tip***. Mouse over the question mark icon () to display the tip text. The tip text recommends what to do to change the credential or the network environment if the step has a status of "Failed".

Chapter 3

Discovering Azure Resources

Overview

The following sections describe how to discover Microsoft Azure resources for monitoring by SL1 using the *Microsoft: Azure* PowerPack.

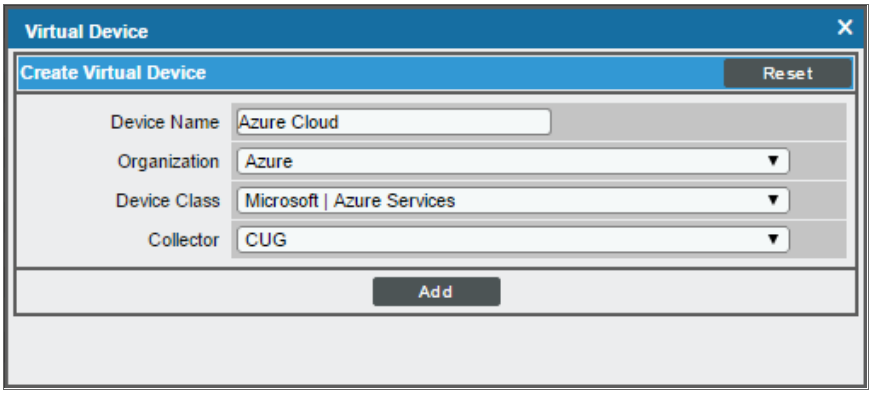
Creating an Azure Virtual Device

Because the Azure service does not have a static IP address, you cannot discover an Azure device using discovery. Instead, you must create a ***virtual device*** that represents the Azure service. A virtual device is a user-defined container that represents a device or service that cannot be discovered by SL1. You can use the virtual device to store information gathered by policies or Dynamic Applications.

To create a virtual device that represents your Azure service:

1. Go to the **Device Manager** page (Registry > Devices > Device Manager).
2. Click the **[Actions]** button and select *Create Virtual Device* from the menu. The **Virtual Device** modal page appears.

3. Enter values in the following fields:



- ***Device Name***. Enter a name for the device. For example, "Azure Cloud".
- ***Organization***. Select the organization for this device. The organization you associate with the device limits the users that will be able to view and edit the device. Typically, only members of the organization will be able to view and edit the device.
- ***Device Class***. Select *Microsoft | Azure Services*.
- ***Collector***. Select the collector group that will monitor the device.

TIP: When monitoring an account with multiple child subscriptions, you can load-balance how SL1 monitors your Azure components by discovering groups of subscriptions and their components across multiple collectors. For details, see the section on *Load-Balancing an Account with Multiple Subscriptions*.

4. Click **[Add]** to create the virtual device.

Aligning the Azure Dynamic Applications

The Dynamic Applications in the *Microsoft: Azure* PowerPack are divided into the following types:

- ***Discovery***. These Dynamic Applications poll Azure for new instances of services or changes to existing instances of services.
- ***Configuration***. These Dynamic Applications retrieve configuration information about each service instance and retrieve any changes to that configuration information.
- ***Performance***. These Dynamic Applications poll Azure for performance metrics.

When configuring SL1 to monitor Azure services, you can manually align Dynamic Applications to discover Azure component devices.

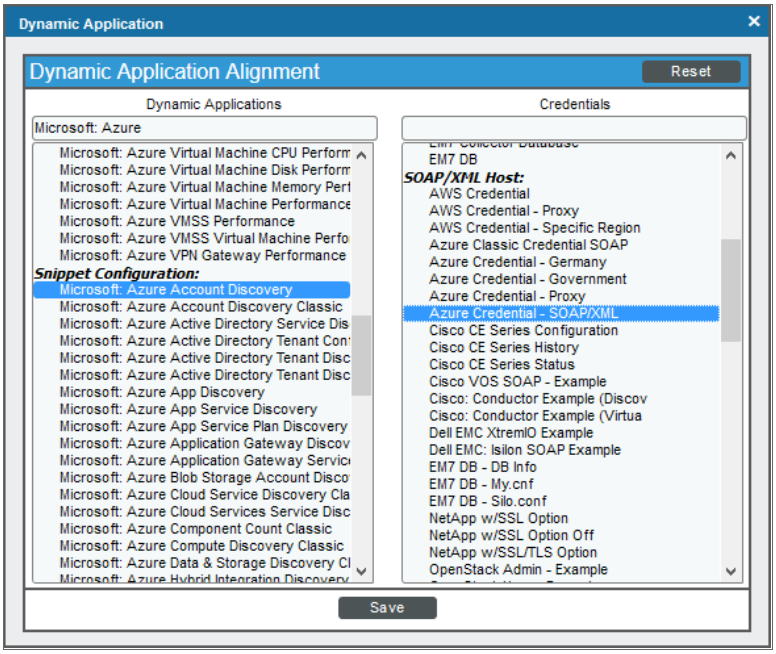
Discovering Azure Component Devices

To discover all the components of your Azure platform, you must manually align the "Microsoft: Azure Account Discovery" Dynamic Application with the Azure virtual device.

TIP: When monitoring an account with multiple child subscriptions, ScienceLogic recommends that you first review your device capacity and load limits to determine the best method for implementation prior to discovery. For details, see the section on *Load-Balancing an Account with Multiple Subscriptions*.

To manually align the "Microsoft: Azure Account Discovery" Dynamic Application:

1. Go to the **Device Manager** page (Registry > Devices > Device Manager).
2. Click the wrench icon () for your Azure virtual device.
3. In the ***Device Administration*** panel, click the **[Collections]** tab. The **Dynamic Application Collections** page appears.
4. Click the **[Actions]** button and select *Add Dynamic Application* from the menu.
5. In the **Dynamic Application Alignment** modal:



- In the ***Dynamic Applications*** field, select *Microsoft: Azure Account Discovery*.
- In the ***Credentials*** field, select the credential you created for your Azure service.

6. Click **[Save]** to align the Dynamic Application with the Azure virtual device.

When you align the "Microsoft: Azure Account Discovery" Dynamic Application with the Azure virtual device, SL1 does one of the following, depending on your subscription model:

- If you are monitoring an account with multiple child subscriptions, SL1 creates a root component device representing the Azure account and one or more child component devices representing all of your Azure subscriptions.
- If you are monitoring a single subscription, SL1 creates a root component device representing your Azure subscription.

TIP: When monitoring an account with multiple child subscriptions, you can load-balance how SL1 monitors your Azure components by discovering groups of subscriptions and their components across multiple collectors. For details, see the section on *Load-Balancing an Account with Multiple Subscriptions*.

SL1 then automatically aligns several other Dynamic Applications to the subscription component devices. These additional Dynamic Applications discover and create component devices for Active Directory tenants, Traffic Manager profiles, and each location used by the Azure account.

Under each location, SL1 then discovers the following component devices:

- Application Gateway Services
    - Application Gateways
- App Services
    - App Service Plan
        - Function App
        - Web App
- Load Balancer Services
    - Load Balancers
- Network Security Group Services
    - Network Security Groups
- Recovery Service Vaults Services
    - Recovery Service Vaults
- Resource Groups Services
    - Resource Groups
- SQL Server Services
    - SQL Servers
    - SQL Databases
- Storage Manage Disks
    - Manage Disk Service
        - Manage Disk

- Storage Services
    - Storage Accounts
- Virtual Machines Services
    - Virtual Machines
- Virtual Network Services
    - VPN Gateways
    - Virtual Networks
        - Virtual Network Subnets
- VM Scale Set Services
    - VM Scale Sets
        - Virtual Machines

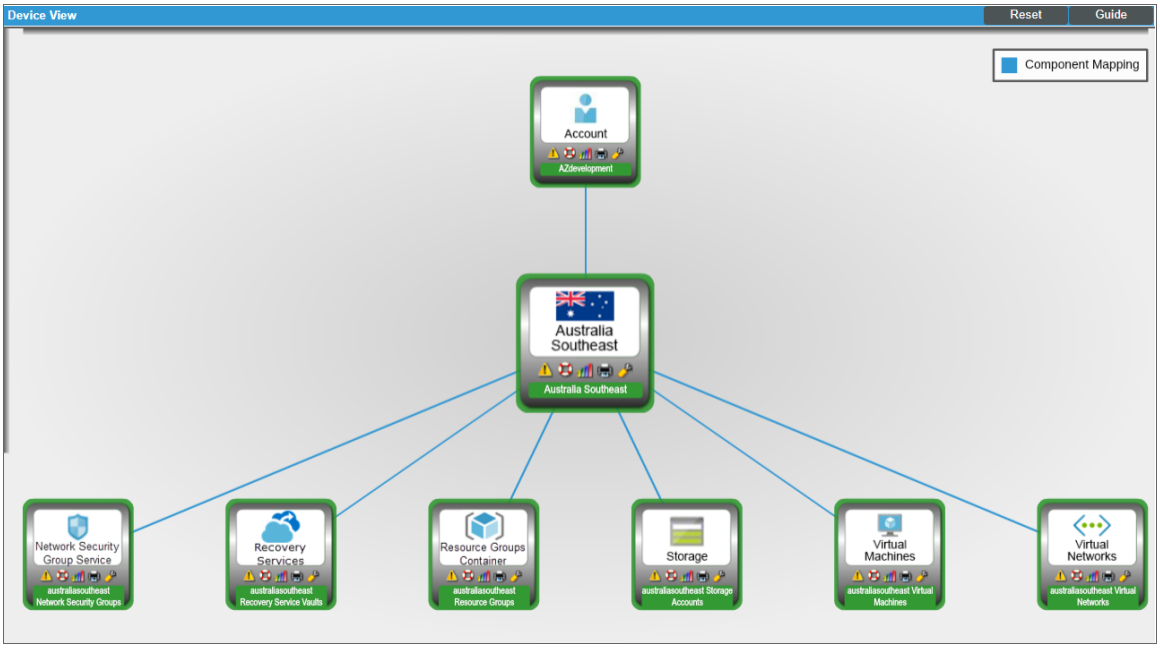
NOTE: SL1 might take several minutes to align these Dynamic Applications and create the component devices in your Azure service.

NOTE: When discovering a large number of component devices, such as when discovering an account with multiple child subscriptions, the discovery process can cause the appearance of numerous critical events with the message, "`Large backlog of asynchronous jobs detected`". This will occur only during the initial discovery session.

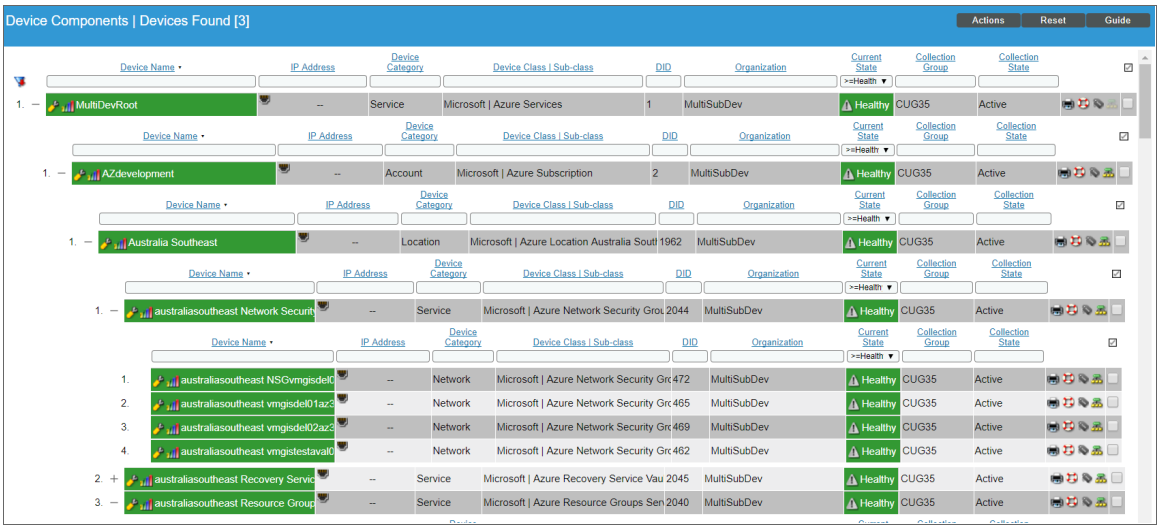
Viewing Azure Component Devices

In addition to the **Device Manager** page (Registry > Devices > Device Manager), you can view the Azure service and all associated component devices in the following places in the user interface:

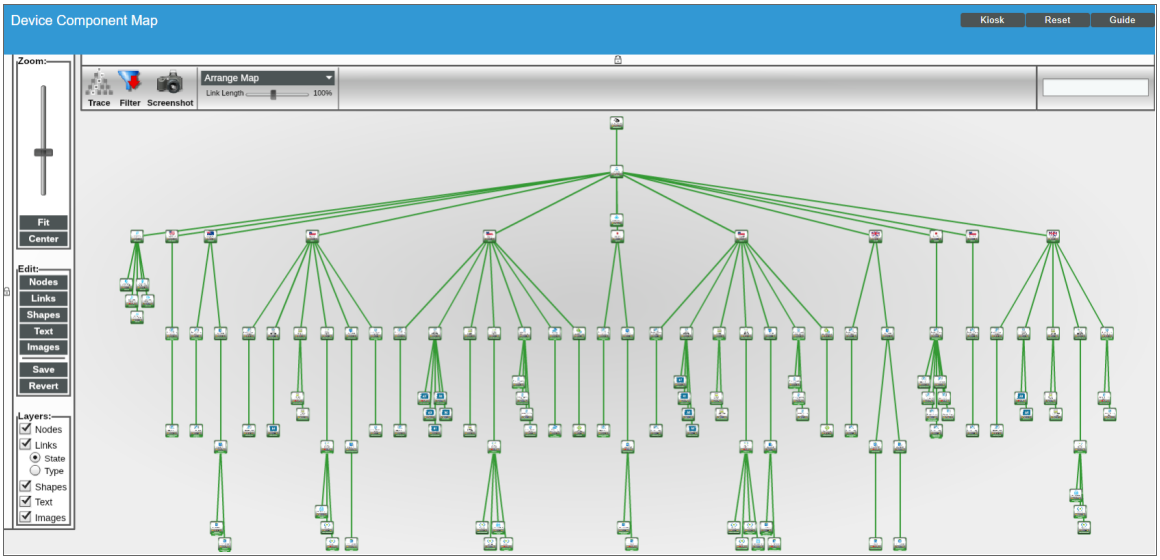
- The **Device View** modal page (click the bar-graph icon [] for a device, then click the **Topology** tab) displays a map of a particular device and all of the devices with which it has parent-child relationships. Double-clicking any of the devices listed reloads the page to make the selected device the primary device:



- The **Device Components** page (Registry > Devices > Device Components) displays a list of all root devices and component devices discovered by SL1 in an indented view, so you can easily view the hierarchy and relationships between child devices, parent devices, and root devices. To view the component devices associated with an Azure service, find the Azure virtual device and click its plus icon (+):



- The **Component Map** page (Views > Device Maps > Components) allows you to view devices by root node and view the relationships between root nodes, parent components, and child components in a map. This makes it easy to visualize and manage root nodes and their components. SL1 automatically updates the **Component Map** as new component devices are discovered. The platform also updates each map with the latest status and event information. To view the map for an Azure service, go to the **Component Map** page and select the map from the list in the left NavBar. To learn more about the **Component Map** page, see the ***Views*** manual.



Relationships Between Component Devices

In addition to parent/child relationships between component devices, SL1 also creates relationships between the following component devices:

- Apps and Resource Groups
- Application Gateways and Resource Groups
- Application Gateways and Virtual Network Subnets
- Azure Traffic Managers and Traffic Managers
- Load Balancers and Resource Groups
- Managed Disks and Resource Groups
- Managed Disks and Virtual Machines
- Network Security Groups and Resource Groups
- Network Security Groups and Virtual Network Subnets
- Recovery Service Vaults and Resource Groups
- SQL Databases and Resource Groups
- SQL Servers and Resource Groups
- Storage Accounts and Resource Groups
- Traffic Manager Profiles and Resource Groups
- Virtual Machines and Network Security Groups
- Virtual Machines and Resource Groups
- Virtual Machines and Storage Accounts
- Virtual Machines and Virtual Networks
- Virtual Machines and Virtual Network Subnets
- Virtual Machine Scale Sets and Load Balancers
- Virtual Machine Scale Sets and Resource Groups
- Virtual Machine Scale Sets and Virtual Network Subnets
- Virtual Machine Scale Set Virtual Machines and Resource Groups
- Virtual Networks and Resource Groups
- VPN Gateways and Resource Groups
- VPN Gateways and Virtual Network Subnets

Additionally, the platform can automatically build relationships between Azure component devices and other associated devices:

- If you discover Cisco Cloud Center devices using the Dynamic Applications in the *Cisco: Cloud Center* PowerPack version 103 or later, SL1 will automatically create relationships between Azure Virtual Machines and Cisco Cloud Center applications.
- If you discover Dynatrace environments using the Dynamic Applications in the *Dynatrace* PowerPack, SL1 will automatically create relationships between the following device types:
    - Azure Virtual Machines and Dynatrace Hosts
    - Azure Virtual Machine Scale Sets and Dynatrace Hosts

Chapter 4

Monitoring Azure Unified Alerts

Overview

The following sections describe the Azure unified alert Event Policies that are included in the *Microsoft: Azure* PowerPack and information about configuring Azure and SL1 to generate events based on Azure unified alerts:

Prerequisites for Configuring Azure Unified Alerts

In addition to SL1 collecting metrics for Azure resources, you can configure Azure to send alert information to SL1 via API. SL1 can then generate an event for each alert.

However, before you can monitor Azure unified alerts in SL1 using the *Microsoft: Azure* PowerPack, you must first configure Azure to proactively send alerts when important conditions are found in your Azure monitoring data. These alerts are based on metrics and activity logs, and are raised when the alert's monitor condition is set to "fired".

You must also create alert rules in Azure that determine the following:

- The resource that the alert is targeting
- The signal from the target resource that could trigger the alert
- The logic that determines whether the signal from the target resource actually triggers the alert

For details about how to create and manage alert rules, see https://docs.microsoft.com/en-us/azure/azure-monitor/platform/alerts-overview.

Azure Unified Alert Event Policies

The *Microsoft: Azure* PowerPack includes several pre-defined event policies for unified alerts, based on their severity:

Event Policy Name	Event Source	Severity
Microsoft: Azure Alert Severity 0	API	Critical
Microsoft: Azure Alert Severity 1	API	Major
Microsoft: Azure Alert Severity 2	API	Minor
Microsoft: Azure Alert Severity 3	API	Notice
Microsoft: Azure Alert Severity 4	API	Notice
Microsoft: Azure Alert Severity 0 Resolved Microsoft: Azure Alert Severity 1 Resolved Microsoft: Azure Alert Severity 2 Resolved Microsoft: Azure Alert Severity 3 Resolved Microsoft: Azure Alert Severity 4 Resolved	API	Healthy

These events are aligned to Azure component devices in the following way:

- If the alert is targeted to a component device that is discovered in SL1, then the event in SL1 will be aligned with that component device.
- If the alert is targeted to a component device that either is not discovered in SL1 or if SL1 cannot determine the appropriate component device, then that alert will be aligned to the Azure subscription component device.

NOTE: The Healthy events are raised when the alert's monitor condition is "resolved" or the alert state is "acknowledged" or "closed".

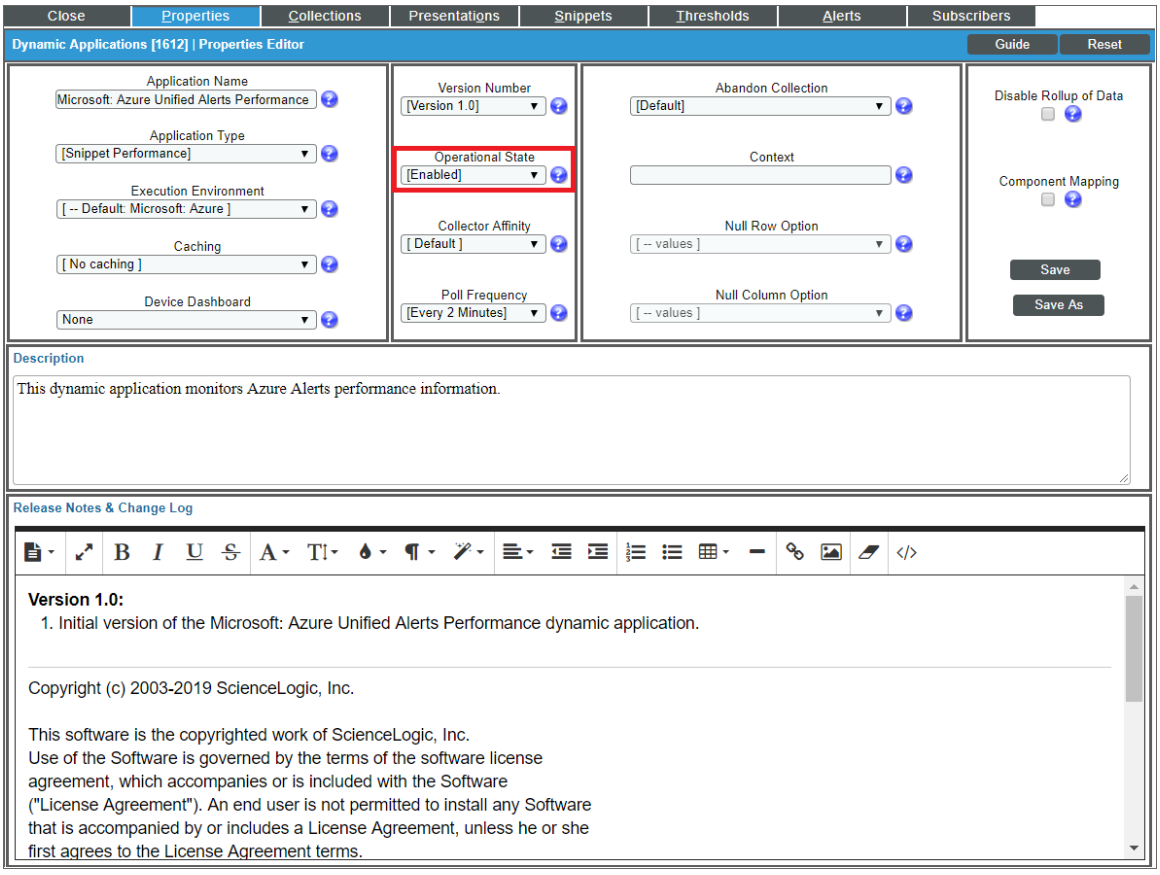
Enabling the "Microsoft: Azure Unified Alerts Performance" Dynamic Application

The *Microsoft: Azure* PowerPack also includes a "Microsoft: Azure Unified Alerts Performance" Dynamic Application. This Dynamic Application collect alerts from the Azure API for all available resources and associates the alerts with the appropriate Azure component devices in SL1, if applicable. If an appropriate component device does not exist in SL1 or cannot be determined, the alert is instead associated with the component device for the Azure subscription.

This Dynamic Application must be enabled if you want SL1 to generate unified alert events.

To enable the "Microsoft: Azure Unified Alerts Performance" Dynamic Application:

1. Go to the **Dynamic Applications Manager** page (System > Manage > Applications).
2. Locate the "Microsoft: Azure Unified Alerts Performance" Dynamic Application and then click its wrench icon ( ). The **Dynamic Applications Properties Editor** page appears.



3. In the ***Operational State*** field, select *Enabled*.
4. Click **[Save]**.

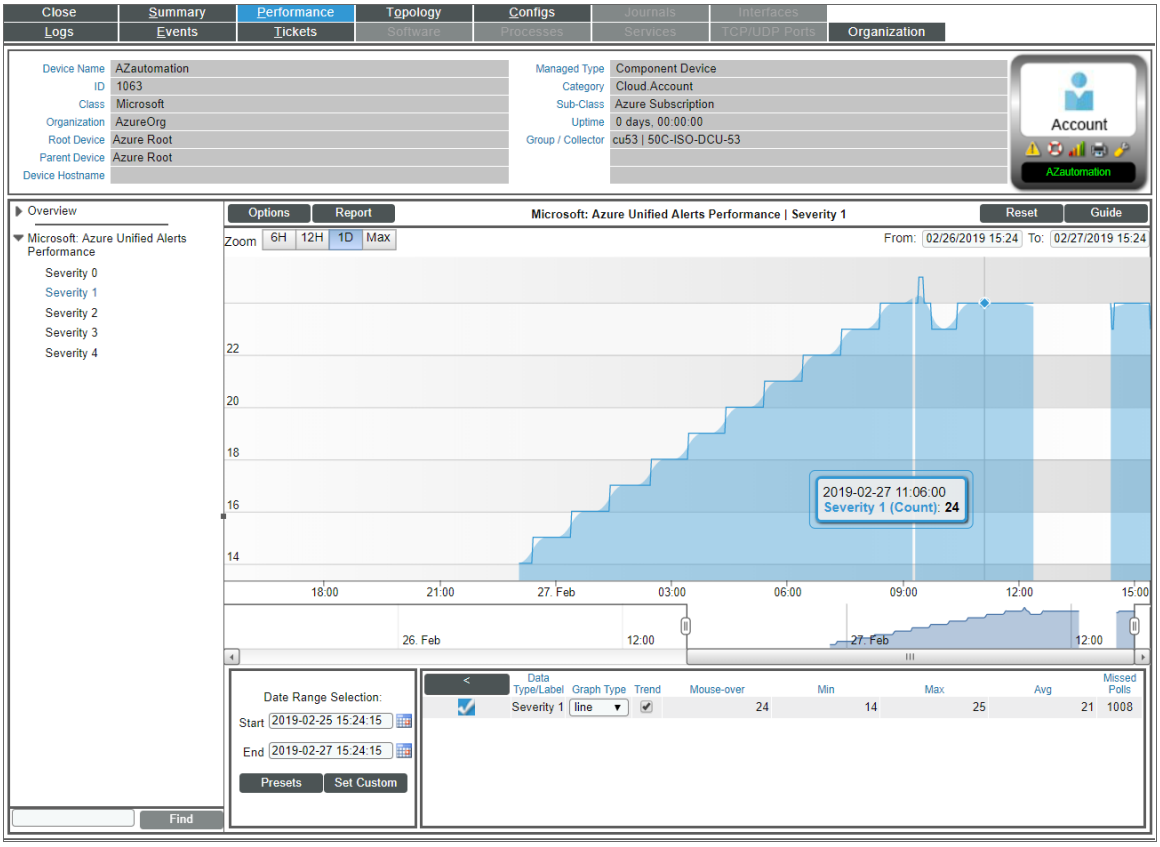
Viewing Azure Unified Alert Counts

After you have enabled the "Microsoft: Azure Unified Alerts Performance" Dynamic Application and it has begun collecting alerts from the Azure API, you can view a count of the total number of alerts generated for each severity level for a given component device.

NOTE: By default, the "Microsoft: Azure Unified Alerts Performance" Dynamic Application collects alerts over a 1-day period.

To view Azure unified alert counts:

1. Go to the **Device Components** page (Registry > Devices > Device Components).
2. Click the plus-sign icon (+) for your Azure service until you locate the Azure component device for which you want to see an alert count. Click its graph icon (). The **Device Summary** page appears.
3. Click the **[Performance]** tab. The **Device Performance** page appears.
4. Click the **Microsoft: Azure Unified Alerts Performance** link to expand the options listed, and then select the alert severity for which you want to see metrics. The performance graph displays a graph detailing the count for your selected alert severity over the selected timespan.



Chapter 5

Azure Run Book Actions and Automations

Overview

The following sections describe how to use the Run Book Action policies and Run Book Automation policies that are included in the *Microsoft: Azure* PowerPack:

About the Azure Run Book Actions and Automations

The *Microsoft: Azure* PowerPack includes Run Book Actions and Run Book Automation policies that can be used to:

- Automatically disable data collection for Virtual Machines, Virtual Machine Scale Sets (VMSS), and Storage Disks based on their VM tag
- Automatically create and start a discovery session using the public or private IP address of a Virtual Machine, and after the device is discovered, merge the physical device with the corresponding component
- Automatically move a Virtual Machine to a vanished state if the component is in a terminated state

The following table describes the Run Book Automation policies and what they do:

Run Book Automation Policy Name	Result
Microsoft Azure: Disable and Discover from IP	If a component device belongs to the Virtual Machines device group and has a relevant Azure tag, SL1 disables the device.
Microsoft Azure: Disable Storage Disks	If a component device belongs to the Storage Disks device group and has a relevant Azure tag, SL1 disables the device.
Microsoft Azure: Discover from IP	SL1 automatically discovers VM instances by public or private IP address.
Microsoft Azure: Merge with VM	If SL1 finds the "Device Record Created" event on the newly discovered physical device, SL1 merges the newly discovered physical device with the corresponding component device.
Microsoft Azure: Vanish Terminated VMs	If a device is in a terminated or terminating state, SL1 un-merges the VM instance and physical device (if applicable), clears the device's associated events, and then moves the device to a vanished state.

NOTE: The Run Book Automation policies in the *Microsoft: Azure* PowerPack are disabled by default. To use these Run Book Automations, you must enable the Run Book Automation policies and modify the parameters in the Run Book Actions as needed. See the following procedures for more information.

As a prerequisite for discovering physical devices, make sure that traffic to the following ports is allowed in the inbound security rules on the Azure Portal for a Virtual Machine:

- **Port 161**. Allows the discovery session to use SNMP credentials.
- **Ports 5985, 5986**. Allows the discovery session to use PowerShell credentials.

If the above ports are not open or cannot be opened, you can include extra credentials for the discovery session by modifying the following parameter in the "Microsoft Azure: Discover from IP" Run Book Action, using a comma-separated list of credential IDs:

```
EXTRA_CREDS = "<ID1>,<ID2>,<ID3>"
```

> **NOTE**: When a discovery session is given a list of credentials, the first credential that successfully authenticates is used to discover a physical device.

For more information about Microsoft Azure inbound security rules, see the following Microsoft article: How to open ports to a virtual machine with the Azure portal.

Disabling VMs or Storage Disks by VM Tag

> **NOTE**: The following Run Book Automation policies do not enable data collection for Azure VMs or Storage Disks. You must manually enable data collection for these VMs or Storage Disks.

Run Book Automation Policy: Disable and Discover from IP

The "Disable and Discover from IP" Run Book Automation policy runs only on newly discovered VMs. The policy takes no action for existing VMs.

The automation for disabling Azure VMs or Azure VMSS includes the following Run Book Actions, which are executed in the following order:

- ***Microsoft Azure: Get Unique ID***. This action retrieves the unique ID of the component. This action runs on the Database Server.
- ***Microsoft Azure: Collect VM Configuration***. This action retrieves VM configuration, including the tags used to disable the VM. This action runs on the Collector.
- ***Microsoft Azure: Disable By VM Tag***. If a newly discovered VM contains the tags specified in the snippet, this action disables collection for this component.
- ***Microsoft Azure: Discover from IP***. If the VM is running and is newly discovered, this action creates the discovery session and runs automatically to discover the physical device. This action will *not* create a discovery session for a discovered VM that was disabled right after being discovered.

The following Run Book Automation policy triggers the above Run Book Actions:

- ***Microsoft Azure: Disable and Discover from IP***. This Run Book Automation policy executes when the "Component Device Record Created" event is active on the matching devices, immediately after the devices are discovered in the system. Enable this Run Book Automation policy if you want to disable VM instances by Azure tag *and* want to enable automated discovery of VM instances by public or private IP address. This policy is configured to run both processes in the correct order for VM instances.

Run Book Automation Policy: Disable Storage Disks

The "Disable Storage Disks" Run Book Automation policy runs only on newly discovered Storage Disks. The policy takes no action for existing Storage Disks.

The automation for disabling Azure Storage Disks includes the following Run Book Actions, which are executed in the following order:

- ***Microsoft Azure: Get Unique ID***. This action retrieves the unique ID of the component. This action runs on the Database Server.
- ***Microsoft Azure: Collect Storage Disk Configuration***. This action retrieves disk and VM configurations, including the tags that belong to the VM used by the Storage Disk. This action runs on the Collector.
- ***Microsoft Azure: Disable By VM Tag***. If a newly discovered Storage Disk belong to a VM that contains the tags specified in the snippet, this action disables collection for the component.

The following Run Book Automation policy triggers the above actions:

- ***Microsoft Azure: Disable Storage Disks***. This Run Book Automation policy executes when the "Component Device Record Created" event is active on the matching devices, immediately after the devices are discovered in the system. Enable this policy if you want to disable Storage Disk instances by Azure tag, but do not want to enable automated discovery of Storage Disk instances by public or private IP address.

Configuration Steps

To use these automations, you must:

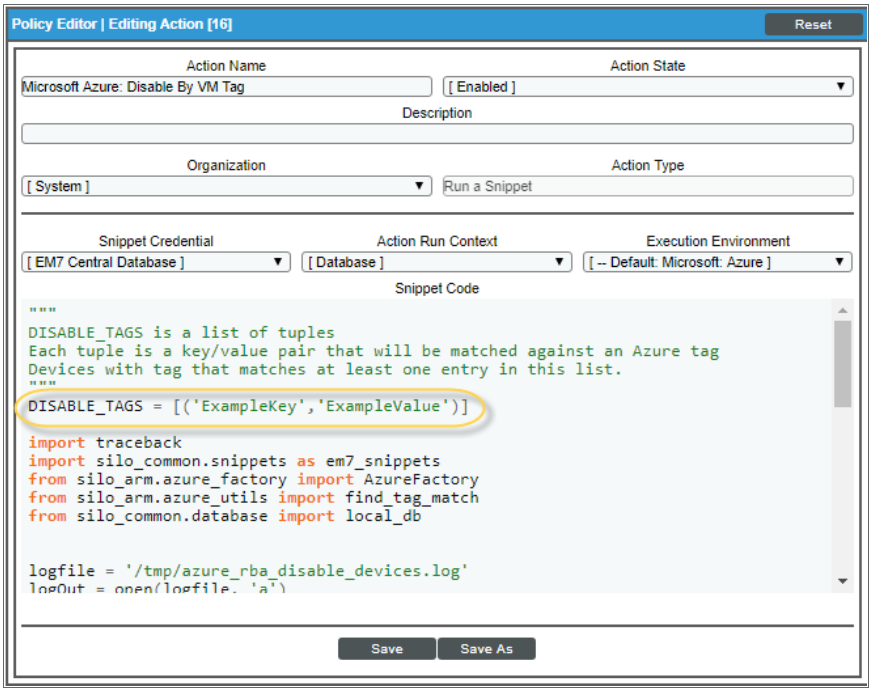
- *Modify the parameters of the "Disable By VM Tag" Run Book Action*
- *Enable the "Component Device Record Created" event policy*
- *Enable the Run Book Automation policies*
- *Configure your system to preserve these changes*

Modifying the Parameters of the "Disable By VM Tag" Run Book Action

The snippet for the "Microsoft Azure: Disable by VM Tag" Run Book Action includes the pre-defined list of key/value pairs that SL1 compares to the tags collected from Azure. You must modify this list to include the key/value pairs that you want to use to disable VM instances.

To modify the parameters for the "Microsoft Azure: Disable by VM Tag" Run Book Action:

1. Go to the **Action Policy Manager** page (Registry > Run Book > Actions).
2. Click the wrench icon ( ) for the "Microsoft Azure: Disable by VM Tag" Run Book Action.



3. In the ***Snippet Code*** field, locate and edit the following line:

   ```
   DISABLE_TAGS = [('ExampleKey','ExampleValue')]
   ```

   The line must be in the following format, with each key and each value inside single-quotes and each key/value pair comma-separated inside parentheses, with commas separating each key/value pair.

   ```
   DISABLE_TAGS [('Key','Value'),('Key','Value'),...,('Key','Value')]
   ```

   For example, suppose you want to disable a VM instance where the "Environment" key is either "dev" or "test" or the "Owner" key is "Sales". You would update the line so it looks like this:

   ```
   DISABLE_TAGS [('Environment','dev'),('Environment','test'),('Owner','Sales')]
   ```

4. As needed, update the following lines:

   - To disable discovery using SNMP credentials:

     ```
     USE_SNMP = False
     Discover_Non_SNMP = '1'
     ```

   - To include additional user-defined credentials in the discovery session, use a comma-separated list of credential IDs:

     ```
     EXTRA_CREDS = "<ID1>,<ID2>,<ID3>"
     ```

- To apply a device template to all newly discovered physical devices, specify the name of the template:

  ```
  TEMPLATE_NAME = "<Name>"
  ```

5. When you are done editing, click the **[Save]** button.

Enabling the "Component Device Record Created" Event Policy

To enable the "Component Device Record Created" event policy:

1. Go to the **Event Policy Manager** page (Registry > Events > Event Manager).
2. Click the wrench icon () for the "Component Device Record Created" event policy.
3. In the ***Operational State*** field, select *Enabled*.
4. Click **[Save]**.

To prevent this change from being overwritten when the PowerPacks installed on the system are updated, you can enable the ***Selective PowerPack Field Protection*** option. To enable this option:

1. Go to the **Behavior Settings** page (System > Settings > Behavior).
2. Check the ***Enable Selective PowerPack Field Protection*** checkbox.
3. Click **[Save]**.

Enabling the Run Book Automation Policies

To enable one or more Run Book Automation policies in the *Microsoft: Azure* PowerPack:

1. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation).
2. Click the wrench icon () for the Run Book Automation policy you want to enable.
3. In the ***Policy State*** field, select *Enabled*.
4. Click **[Save]**.

Preserving Automation Changes

If you have modified Run Book Actions and Run Book Automation policies that are included in the *Microsoft: Azure* PowerPack, those changes will be overwritten when the PowerPack is updated in your system. If you have modified Run Book Actions and Run Book Automation policies that are included in the PowerPack, you can:

- Re-implement those changes after each update of the *Microsoft: Azure*PowerPack.
- Remove the content from the PowerPack on your system. When the *Microsoft: Azure*PowerPack is updated in your system, updated versions of this content will not be installed on your system and your local changes will be preserved.

To remove Run Book Action or Run Book Automation policy content from the *Microsoft: Azure* PowerPack on your system:

1. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
2. Click the wrench icon ( ) for the *Microsoft: Azure* PowerPack. The **Editing PowerPack** page appears.
3. In the left NavBar of the **Editing PowerPack** page, select the type of content you want to remove:
   - To remove a Run Book Action, click **Run Book Actions**. The **Embedded Run Book Actions** and **Available Run Book Actions** panes appear.
   - To remove a Run Book Automation policy, click **Run Book Policies**. The **Embedded Run Book Policies** and **Available Run Book Policies** panes appear.
4. In the upper pane, click the bomb icon ( ) for each Run Book Action or Run Book Automation policy that you want to remove from the *Microsoft: Azure* PowerPack on your system.

Discovering VMs and Merging Physical Devices with Components

Run Book Automation Policy: Discover from IP

The "Discover from IP" Run Book Automation policy runs only on newly discovered VMs. The policy takes no action for existing VMs.

The automation for discovering Azure VMs or VMSSs by public or private IP addresses and then discovering the physical device includes three Run Book Actions that are executed in the following order:

- ***Microsoft Azure: Get Unique ID***. This action retrieves the unique ID of the component. This action runs on the Database Server.
- ***Microsoft Azure: Collect VM Configuration***. This action retrieves VM configuration, including public or private IP address and open ports. This action runs on the Collector.
- ***Microsoft Azure: Discover from IP***. If the VM is running and is newly discovered, this action creates the discovery session and runs automatically to discover the physical device. The discovery session name uses the following format: **Azure_VM-*IP_address***.

The following Run Book Automation policy triggers the above Run Book Actions:

- ***Microsoft Azure: Discover From IP***. This Run Book Automation policy executes when the "Component Device Record Created" event is active on the matching devices, immediately after the devices are discovered in the system. Use this action to enable automated discovery of VM instances by public or private IP address.

> **Note**: If a VM was created as "Stopped" by default, and that VM was discovered by the PowerPack, the Run Book Action will not create a discovery session. The action cannot collect an IP address for a stopped VM.

Run Book Automation Policy: Merge with VM

When the "Merge with VM" Run Book Automation policy finds the "Device Record Created" event on the newly discovered physical device, the policy triggers the following Run Book Action:

- ***Microsoft Azure: Merge Physical with Component***. This action merges the newly discovered physical device with the corresponding component device.

The "Merge with VM" Run Book Automation policy runs only on newly discovered devices. The policy takes no action for existing VMs. The discovery session created with the "Discover from IP" Run Book Action, above, will discover the physical device.

Configuration Steps

To use these automations, you must:

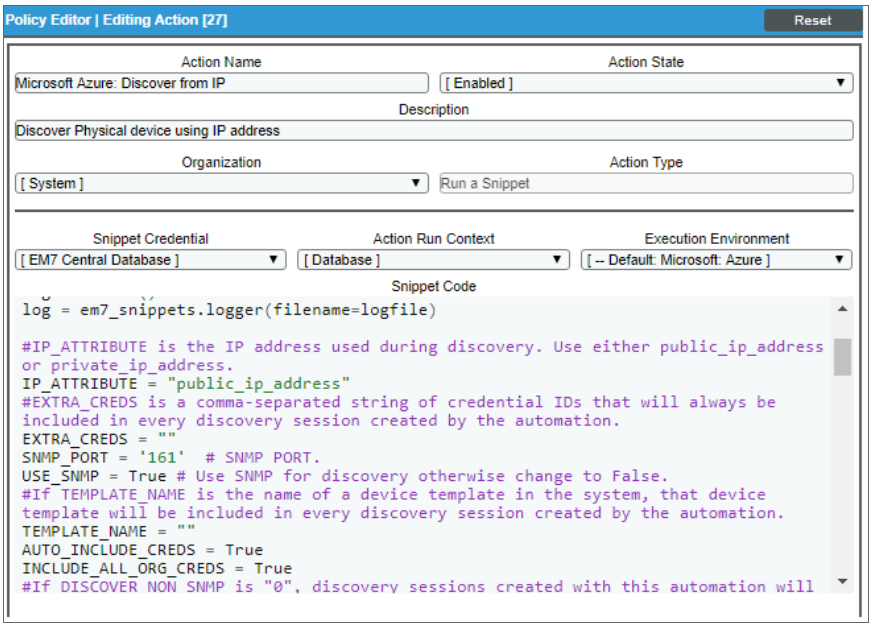
- *Modify the parameters of the Run Book Actions*
- *Enable the "Component Device Record Created" event policy* (Discover from IP policy only)
- *Enable the "Device Record Created" event policy*
- *Enable the Run Book Automation policies*
- *Configure your system to preserve these changes*

Modifying the Parameters of the Run Book Actions

The snippet for the "Microsoft Azure: Discover from IP" Run Book Action includes parameters that define how the Run Book Action creates discovery sessions. By default the snippet uses the public IP address and SNMP port 161 to create the discovery session. You can update these parameters as needed.

To modify the parameters for the "Microsoft Azure: Discover from IP" Run Book Action:

1. Go to the **Action Policy Manager** page (Registry > Run Book > Actions).
2. Click the wrench icon ( ) for the "Microsoft Azure: Discover from IP" Run Book Action.
3. In the ***Snippet Code*** field, locate and edit the lines for the parameters you want to change:



4. As needed, update the following lines:

   - To change from the default public IP address to *private* IP address:

     ```
     IP_ATTRIBUTE = 'private_ip_address'
     ```

     If you change the IP address value to private for this Run Book Action, then you must also update the following line in the "Microsoft Azure: Merge with VM" Run Book Action: `IP_ATTRIBUTE = 'c-VM-public_ipaddress"`.

   - To include additional user-defined credentials in the discovery session, use a comma-separated list of credential IDs:

     ```
     EXTRA_CREDS = "<ID1>,<ID2>,<ID3>"
     ```

   - To *disable* discovery using SNMP credentials, update the following lines:

     ```
     USE_SNMP = False
     DISCOVER_NON_SNMP = '1'
     ```

   - To apply a device template to all newly discovered physical devices, specify the name of the template:

     ```
     TEMPLATE_NAME = "<Name>"
     ```

   - To disable the automatic alignment of credentials to the discovery session, change this line to:

```
AUTO_INCLUDE_CREDS = False
```

- If INCLUDE_ALL_ORG_CREDS is "True" and the AUTO_INCLUDE_CREDS parameter is "True", credentials that are aligned with all organizations (credentials that do not have an explicit organization alignment) are automatically included in the discovery session when that credential meets the other requirements for being automatically included in the discovery session.

  ```
  INCLUDE_ALL_ORG_CREDS = True
  ```

5. When you are done editing, click the **[Save]** button.

Enabling the "Component Device Record Created" Event Policy (Discover from IP Only)

To enable the "Component Device Record Created" event policy:

1. Go to the **Event Policy Manager** page (Registry > Events > Event Manager).
2. Click the wrench icon () for the "Component Device Record Created" event policy.
3. In the ***Operational State*** field, select *Enabled*.
4. Click **[Save]**.

To prevent this change from being overwritten when the PowerPacks installed on the system are updated, you can enable the ***Selective PowerPack Field Protection*** option. To enable this option:

1. Go to the **Behavior Settings** page (System > Settings > Behavior).
2. Check the ***Enable Selective PowerPack Field Protection*** checkbox.
3. Click **[Save]**.

Enabling the "Device Record Created" Event Policy

To enable the "Device Record Created" event policy:

1. Go to the **Event Policy Manager** page (Registry > Events > Event Manager).
2. Click the wrench icon () for the "Device Record Created" event policy.
3. In the ***Operational State*** field, select *Enabled*.
4. Click **[Save]**.

To prevent this change from being overwritten when the PowerPacks installed on the system are updated, you can enable the ***Selective PowerPack Field Protection*** option. To enable this option:

1. Go to the **Behavior Settings** page (System > Settings > Behavior).
2. Check the ***Enable Selective PowerPack Field Protection*** checkbox.
3. Click **[Save]**.

Enabling the Run Book Policies

To enable one or more Run Book Automation policies in the *Microsoft: Azure* PowerPack:

1. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation).
2. Click the wrench icon ( ) for the Run Book Automation policy you want to enable.
3. In the ***Policy State*** field, select *Enabled*.
4. Click **[Save]**.

Preserving Automation Changes

If you have modified Run Book Actions and Run Book Automation policies that are included in the *Microsoft: Azure* PowerPack, those changes will be overwritten when the PowerPack is updated in your system. If you have modified Run Book Actions and Run Book Automation policies that are included in the PowerPack, you can:

- Re-implement those changes after each update of the *Microsoft: Azure*PowerPack.
- Remove the content from the PowerPack on your system. When the *Microsoft: Azure*PowerPack is updated in your system, updated versions of this content will not be installed on your system and your local changes will be preserved.

To remove Run Book Action or Run Book Automation policy content from the *Microsoft: Azure* PowerPack on your system:

1. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
2. Click the wrench icon ( ) for the *Microsoft: Azure* PowerPack. The **Editing PowerPack** page appears.
3. In the left NavBar of the **Editing PowerPack** page, select the type of content you want to remove:
    - To remove a Run Book Action, click **Run Book Actions**. The **Embedded Run Book Actions** and **Available Run Book Actions** panes appear.
    - To remove a Run Book Automation policy, click **Run Book Policies**. The **Embedded Run Book Policies** and **Available Run Book Policies** panes appear.
4. In the upper pane, click the bomb icon ( ) for each Run Book Action or Run Book Automation policy that you want to remove from the *Microsoft: Azure* PowerPack on your system.

Vanishing Terminated or Terminating VM Instances

If a device is in a terminated or terminating state, the "Vanish Terminated VMs" Run Book Action un-merges the VM instance and physical device (if applicable), clears the device's associated events, and then moves the device to a vanished state.

The "Vanish Terminated VMs" Run Book Automation policy runs only on newly discovered VMs. The policy takes no action for existing VMs.

The automation for vanishing terminated VM instances includes the following Run Book Actions:

- ***Microsoft Azure: Get Unique ID***. This action retrieves the unique ID of the component. This action runs on the Database Server.
- ***Microsoft Azure: Check VM Availability***. This action uses the unique ID of the component to get the device availability status. If the device availability status is "Terminated", this action moves to the following Run Book Action, "Vanish Terminated VMs". This action runs on the Collector.
- ***Microsoft Azure: Vanish Terminated VMs***. This action moves the device to the Vanish state when the VM has been terminated in the Azure Portal. This action runs on the Database Server. This action determines if the component was merged with a physical device:
    - If the component was not merged, the action will delete the device's events and move the device to a Vanish state.
    - If the component was merged, the action will un-merge the component with the physical device, and then it will clear the device's events and move the device to a Vanish state.
    - If the component was merged, but the VM was powered off, the action will not do anything until the VM is powered on, at which point the action will update the IP address of the physical device.

      When a merged device is un-merged, the component device vanishes, and the physical device is moved to an automatically created Collector group named "Virtual Group". Any physical device in the Virtual Group can be safely deleted by

The following Run Book Automation policy triggers the above actions:

- ***Microsoft Azure: Vanish Terminated Instances***. This Run Book Automation policy executes when the "Availability Check Failed" event is raised on the virtual machine when it terminated.

To use this automation, you must:

- *Enable the Run Book Automation policies*
- *Configure your system to preserve this change*

Enabling the Run Book Automation Policies

To enable one or more Run Book Automation policies in the *Microsoft: Azure* PowerPack:

1. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation).
2. Click the wrench icon () for the Run Book Automation policy you want to enable.
3. In the ***Policy State*** field, select *Enabled*.
4. Click **[Save]**.

Preserving Automation Changes

If you have modified Run Book Actions and Run Book Automation policies that are included in the *Microsoft: Azure* PowerPack, those changes will be overwritten when the PowerPack is updated in your system. If you have modified Run Book Actions and Run Book Automation policies that are included in the PowerPack, you can:

- Re-implement those changes after each update of the *Microsoft: Azure*PowerPack.
- Remove the content from the PowerPack on your system. When the *Microsoft: Azure*PowerPack is updated in your system, updated versions of this content will not be installed on your system and your local changes will be preserved.

To remove Run Book Action or Run Book Automation policy content from the *Microsoft: Azure* PowerPack on your system:

1. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
2. Click the wrench icon () for the *Microsoft: Azure* PowerPack. The **Editing PowerPack** page appears.
3. In the left NavBar of the **Editing PowerPack** page, select the type of content you want to remove:
    - To remove a Run Book Action, click **Run Book Actions**. The **Embedded Run Book Actions** and **Available Run Book Actions** panes appear.
    - To remove a Run Book Automation policy, click **Run Book Policies**. The **Embedded Run Book Policies** and **Available Run Book Policies** panes appear.
4. In the upper pane, click the bomb icon () for each Run Book Action or Run Book Automation policy that you want to remove from the *Microsoft: Azure* PowerPack on your system.

Copyrights and Trademarks

ScienceLogic, the ScienceLogic logo, and EM7 are trademarks of ScienceLogic, Inc. in the United States, other countries, or both.

Below is a list of trademarks and service marks that should be credited to ScienceLogic, Inc. The ® and ™ symbols reflect the trademark registration status in the U.S. Patent and Trademark Office and may not be appropriate for materials to be distributed outside the United States.

- ScienceLogic™
- EM7™ and em7™
- Simplify IT™
- Dynamic Application™
- Relational Infrastructure Management™

The absence of a product or service name, slogan or logo from this list does not constitute a waiver of ScienceLogic's trademark or other intellectual property rights concerning that name, slogan, or logo.

Please note that laws concerning use of trademarks or product names vary by country. Always consult a local attorney for additional guidance.

Other

If any provision of this agreement shall be unlawful, void, or for any reason unenforceable, then that provision shall be deemed severable from this agreement and shall not affect the validity and enforceability of any remaining provisions. This is the entire agreement between the parties relating to the matters contained herein.

In the U.S. and other jurisdictions, trademark owners have a duty to police the use of their marks. Therefore, if you become aware of any improper use of ScienceLogic Trademarks, including infringement or counterfeiting by third parties, report them to Science Logic's legal department immediately. Report as much detail as possible about the misuse, including the name of the party, contact information, and copies or photographs of the potential misuse to: legal@sciencelogic.com

800-SCI-LOGIC (1-800-724-5644)

International: +1-703-354-1010