**Enable Azure ARC SQL**

This document provides a step-by-step guide on how to enable Azure Arc for hybrid and multi-cloud scenarios. Azure Arc allows you to manage your servers, Kubernetes clusters, and applications across different environments from a single control plane. By following this document, you will be able to deploy Azure Arc agents, register your resources, and apply policies and governance to your hybrid and multi-cloud infrastructure. Ensure you have assigned all the necessary permissions you need before starting, which have been outlined in the prerequisite-enable azure arc documentation.

1. **Enable Azure Arc on Machines: We will need to first enable azure arc at the server level.** 
   * 1. Login to Azure using <https://portal.azure.com/>
     2. Using search bar at the top search and select Azure Arc
     3. From the Infrastructure section in the left-hand menu, select Machines.
     4. Click +add/create, add a machine.
     5. You can choose from single or multiple, we will select multiple (this option uses Service Principal which you created from the previous document stated above)
     6. Select Generate Script
     7. Select pre-existing Resource Group or create new.
     8. Select Region closest where servers reside.
     9. Select Operating System
     10. For this dev/test environment we will go with public, keep in mind for prod we prob want to go with a private endpoint.
     11. Under authentication you want to select the service principal you created earlier, if this is just a single server it’s not needed, next.

A screenshot of a computer

Description automatically generated

* + 1. Feel free to add tags that suit your needs, next.
    2. View the different deployment options available and choose the one that will work best in your environment. You can also download the basic script to run on the servers you choose using your organization's preferred automation tool.

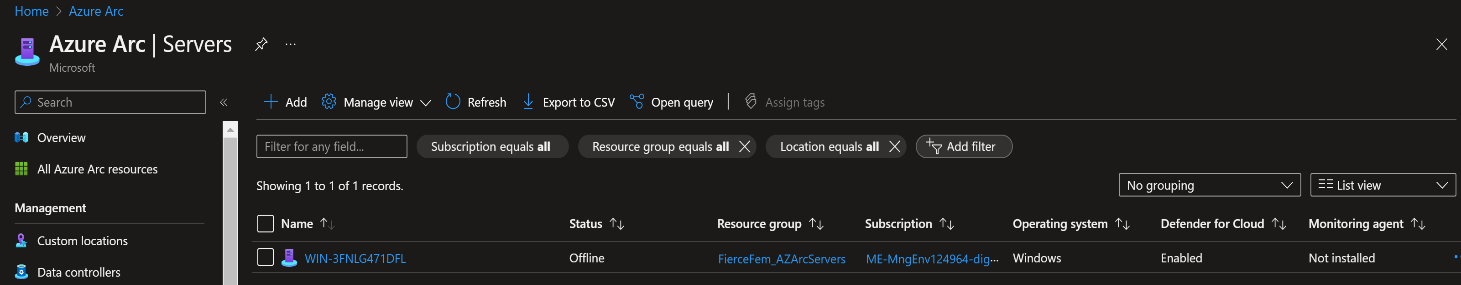
1. **Edit Machines Deployment Script**
   * 1. Before running the script, you will need to add the $ServicePrincipalSecret to the script which is the secret value you copied earlier when creating the service principal.

A screenshot of a computer

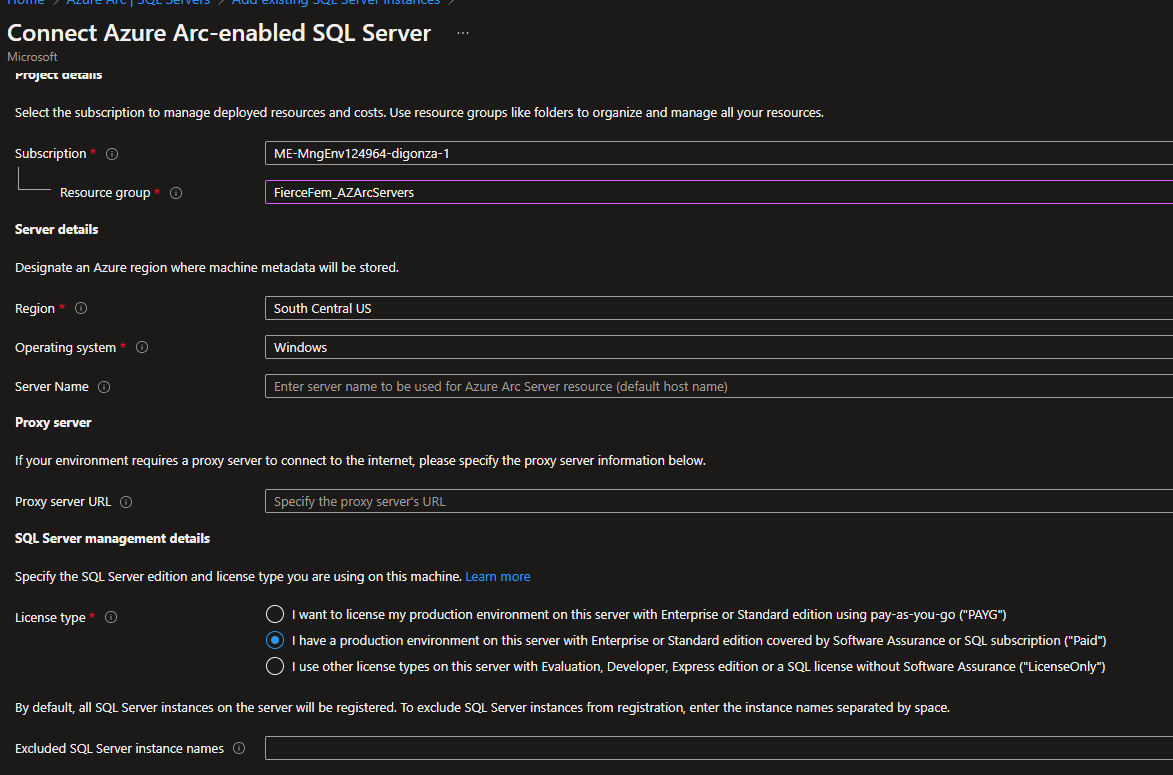
Description automatically generated

Note: The user running script on the machine must be a member of the local admins group on the windows machine, or root account for Linux.

* + 1. After executing the script on the desired servers, the following actions will be performed:
       - Downloads the agent from the Microsoft Download Center.
       - Installs the agent on the server.
       - Creates the Azure Arc-enabled server resource and associates it with the agent.
    2. Once the script is complete you should see the servers in the azure portal under azure arc, servers list.



1. **Onboard SQL Servers to Azure Arc** 
   * 1. In the Azure portal, select SQL Servers Instances from the Data Services section in the left-hand menu.
     2. Select +add.
     3. Select Connect SQL Server to Azure Arc, ensure you meet the pre-requisites. Hit Next.
     4. Choose the resource group you created for Azure Arc Resources.
     5. The Azure Region where machine metadata will be stored.
     6. Select operating system.
     7. You can choose to apply a different server name or leave it blank to use the default host name for SQL server.
     8. Set proxy server URL if applicable, if not leave blank.
     9. Select the license type.
     10. By default, all SQL instances are registered on the servers, so you may exclude any SQL instance name you don’t want to onboard. Next



* + 1. Apply any tags you’d like to add if applicable, this is optional.
    2. Download or copy the script provided. You will need to run the script on each machine you set up beforehand (You can choose to use your organization's preferred automation tool to run at scale).

**Note:** Make sure the machine has network connectivity to Azure and to your target machine with SQL Server. Note (windows only): The PowerShell script you download will be unsigned. You can [sign the PowerShell script](https://go.microsoft.com/fwlink/?linkid=2222919) or you can run the script by [setting the PowerShell execution policy to allow running unsigned scripts](https://go.microsoft.com/fwlink/?linkid=2223106).

1. **Edit SQL Server Script** 
   * 1. To run this on multiple machines you will need to change the script. Uncomment the following parameters and add the values generated earlier:

$servicePrincipalAppId="{serviceprincipalAppID}"

(To find the "service principal AppID", go to the Azure portal and navigate to "Microsoft Entra". From there, select "App registration" and find your app name. Under "Essentials", you will find the "Application (client) ID", which is the service principal AppID.)

$servicePrincipalSecret="{serviceprincipalPassword}"

(secret value you copied and saved earlier from pre-requisites)

$servicePrincipalTenantId="{serviceprincipalTenantId}"

(Go to the Azure portal and navigate to "Microsoft Entra". From there, select "App registration" and find your app name. Under "Essentials” under essentials grab the Directory (tenant) id.)

* + 1. The script Execution Process:
* Checks connectivity from your environment to Azure and specified machine
* Onboard the host machine by deploying the Azure Connected Machine agent if not already onboarded.
* Initiates SQL Server instance discovery.
* d. Adds SQL Server instances on your target machine to Azure.
  + 1. Once the script is complete you should see your instances of SQL. You successfully onboarded SQL Instances Azure Arc.

