# GC Azure PBMM Guardrail Toolset v0.1 Implementation Guide

## Overview

The GC PBMM Guardrail toolbox includes three automation tools that can aid departments in meeting the requirements to onboard the GC PBMM Cloud Contract:

1. An interim Azure PBMM Guardrails policy blueprint vending machine, which can create and publish a policy blueprint definition for one or more of the GC Cloud Usage Profiles as identified by TBS OCIO.

Once created, the blueprint definition can then be assigned to a subscription within the department’s Azure tenant.

1. An Azure PBMM Guardrails Compliance Verification Reporting tool, which will report on the department tenant’s compliance to the Azure policies created via the policy blueprint vending machine, as well as a number of additional compliance verification checks which cannot be expressed through Azure Policy.
2. A script which can be used to update the policy scope exclusion list for the Guardrail Azure Policies based on resource group and resource level tagging.

Use case 1 example: network interfaces for UTM firewall VMs need to be excluded from the “Network interfaces should not have public IPs” policy.

Use case 2 example: a department with a subscription that has mixed cloud usage profiles (eg. Sandbox and PBMM) need a way to exclude the PBMM policies from the Sandbox resources, otherwise there will be a huge non-compliance report with a lot of noise, as well as resource manage create and update failures due to policies which only operate in enforcement mode (“Deny” action).

Note that the Guardrails policy blueprint vending machine is used in lieu of the current Azure Canada Federal PBMM sample blueprint because the latter can only support one cloud usage profile, but more importantly has some policies which departments will likely not want to implement using Azure services (eg. costs are too high, or we felt that they would be applicable beyond the 30-day window), has some policies that are currently not working correctly (Microsoft is working on fixing those), and is missing a few policies.

The longer term plan is for the Azure Canada Federal PBMM sample blueprint to eventually be replaced with multiple blueprint samples (matching the cloud usage profiles).

## Current Limitations

1. Currently only PBMM cloud usage profile 6 is supported by the toolset, however it is anticipated that many of the policies will be same for the PBMM cloud usage profiles, so it is recommended to use the tool in its current state for all PBMM cloud usage profiles.
2. Currently the Azure PBMM Guardrails Compliance Verification Reporting tool only includes a few additional guardrail compliance verification checks to cover security configuration that cannot be verified through Azure Policy.

TBS IMTD is currently working on automating the compliance verification of around 30 additional security settings recommended by CIS and Microsoft that cannot be implemented through Azure Policy.

1. An Azure Automation Account runbook is used to create the Guardrail Policy Blueprint definition.

Since an Azure Automation Account’s Run As Account Service Principal’s privileges cannot be elevated to the management group level, the runbook can only create Guardrail Policy Blueprint definitions at the subscription level currently.

This will be addressed in the future by changes to the script to support the use of a user credential that can be elevated to the management group level.

## Pre-steps

In the steps below, note that the code for the Azure Automation Account runbooks is not pulled using the “Source Control” feature of Azure Automation because the “Source Control” feature does not support anonymous access to public GitHub repos (ie. you have to authenticate to the repo to sync the code).

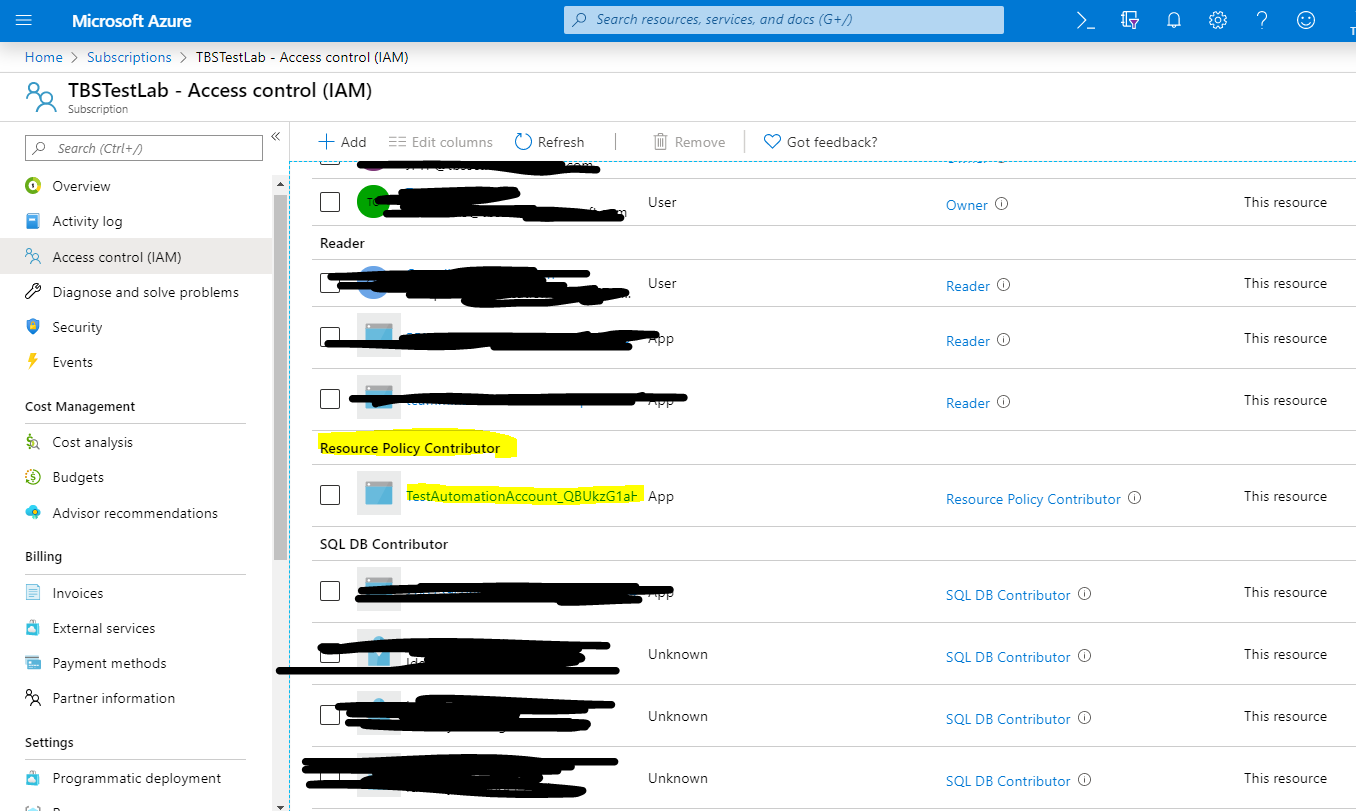
Instead, we use an Automation Account bootstrap script runbook that pulls the code for the other runbooks from the public GitHub repo and creates each of powershell scripts as runbooks in the Automation Account.

1. You need an Azure Automation Account (Azure serverless compute environment for running automated scripts); make sure the Automation Account you use has been created with a Run As account. Follow the steps outlined here if you do not already have an Automation Account: <https://docs.microsoft.com/en-us/azure/automation/automation-quickstart-create-account>

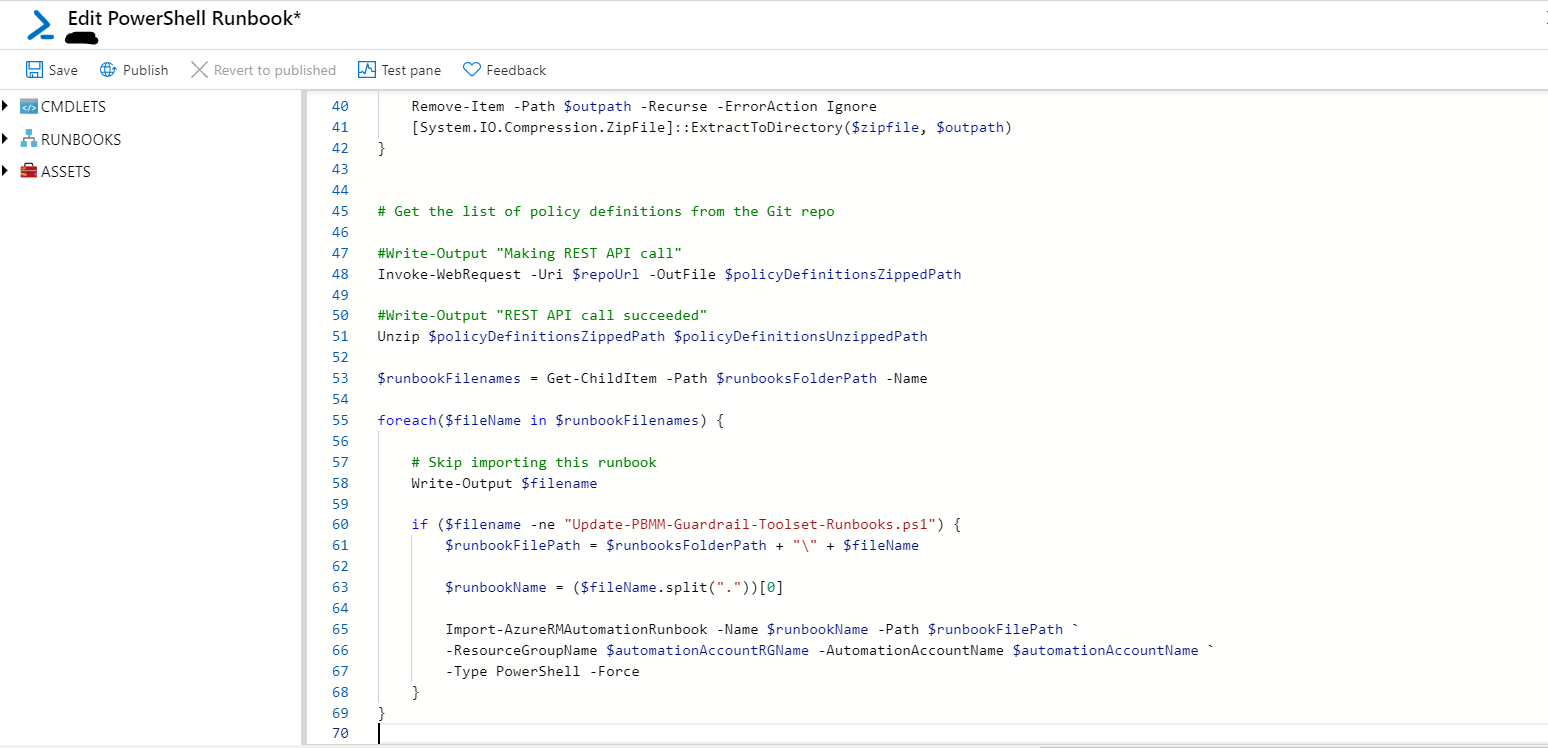
Also ensure that your Automation Account has the latest versions of the powershell modules installed; by default a newly created Automation Account will not have the latest powershell module versions. To update the versions of the installed powershell modules, follow the instructions provided here: <https://docs.microsoft.com/en-us/azure/automation/automation-update-azure-modules>

1. The Azure Automation Account you use is automatically granted the Contributor role at the subscription level, but must also be granted the “Resource Policy Contributor” role at the subscription level, which grants that account permission to create an Azure Blueprint definition at the subscription level.

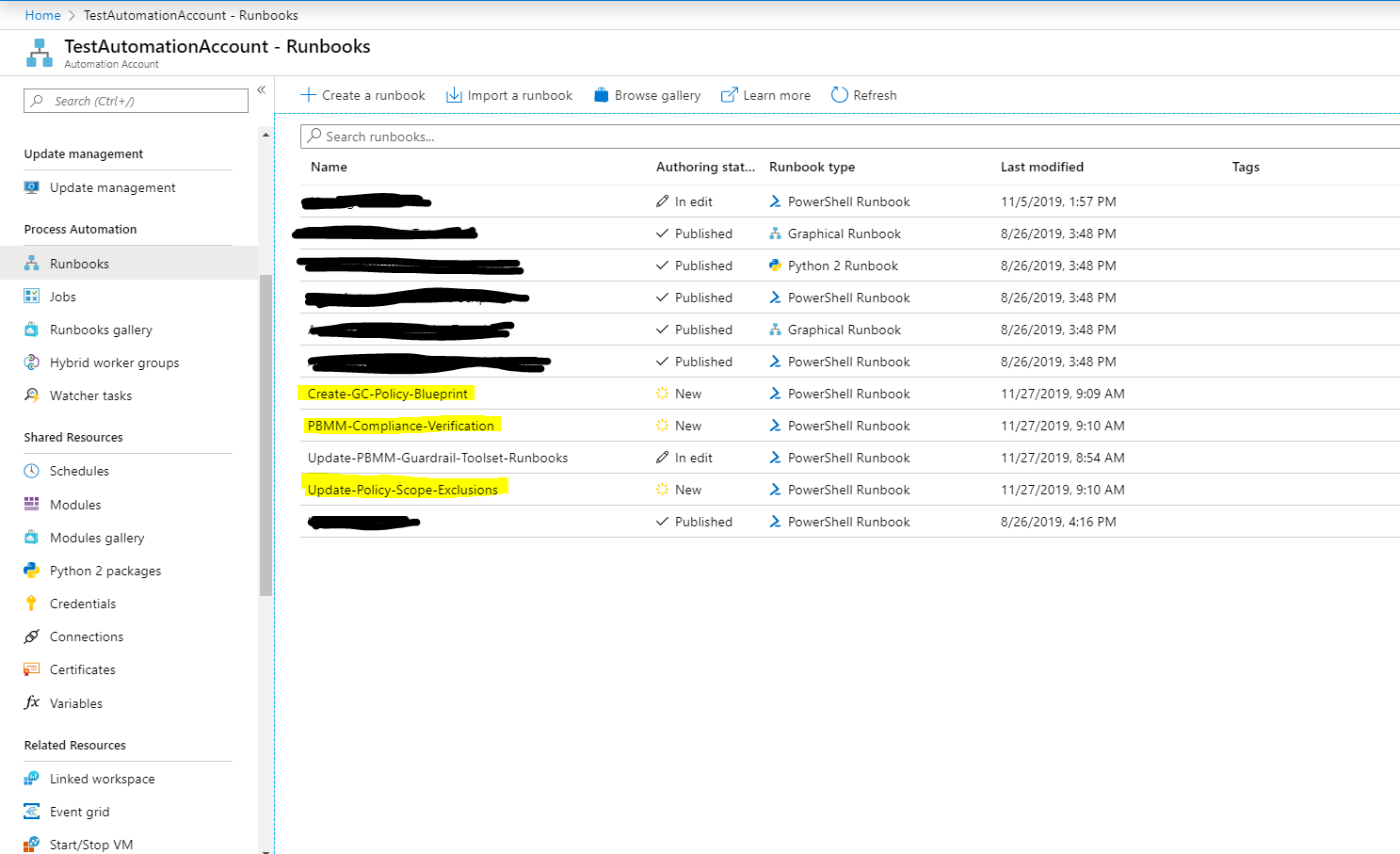
In the Azure Portal, navigate to Subscriptions -> Access Control (IAM) and add a role assignment.



1. Create a new powershell runbook in your Azure Automation Account, as follows:
   1. In the Azure Portal, select the Runbooks blade from your Automation Account.
   2. Select “+ Create a runbook”
   3. Enter “Update-PBMM-Guardrail-Toolset-Runbooks” for the runbook name and select “PowerShell” for the runbook type, then select “Create”
   4. Copy the source code from <https://github.com/canada-ca/cloud-guardrails-azure/blob/master/toolbox/runbooks/Update-PBMM-Guardrail-Toolset-Runbooks.ps1>, and paste it as the source code for the Update-PBMM-Guardrail-Toolset-Runbooks runbook. Click Save and then Publish.



1. Click “Start” to start the runbook you created in step 3.
2. When prompted, enter the name of your Automation Account for the “AutomationAccountName” parameter, and the name of the resource group containing your Automation Account for the “AutomationAccountRGName” parameter, and then click “Ok”
3. Verify that the runbook ran successfully by navigating to your Automation Account’s Runbooks blade and confirming that the following runbooks were created as shown in the screenshot below (click Refresh on the blade to make sure the list has been updated):



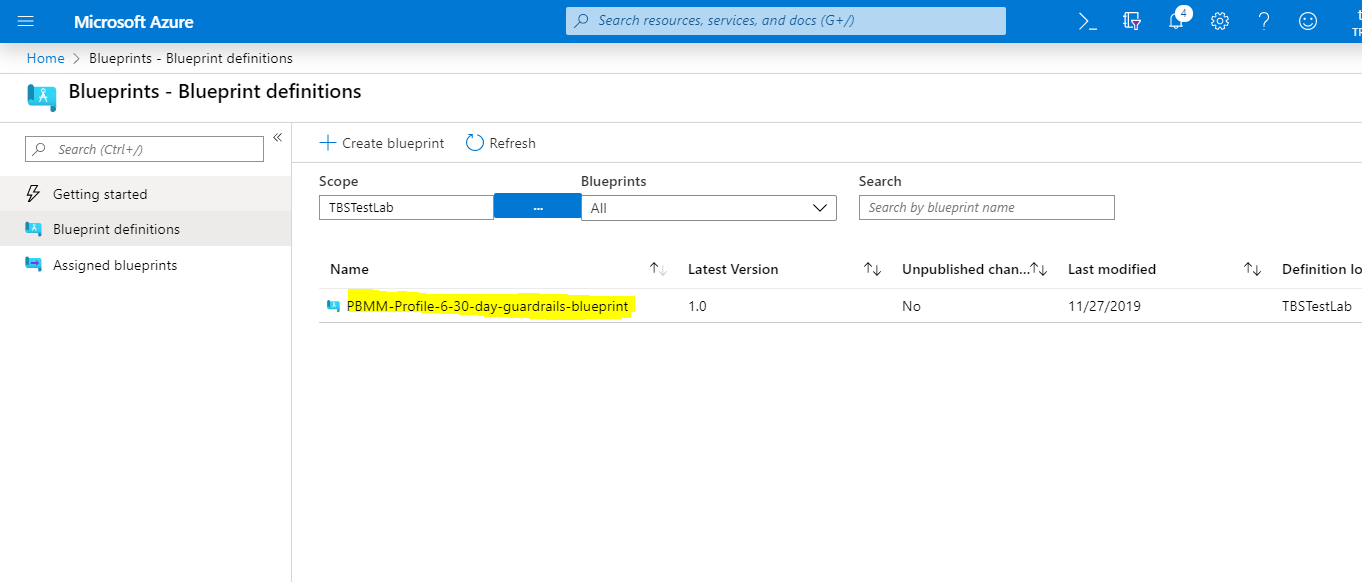
## Azure PBMM Guardrails policy blueprint vending machine implementation steps

### Prerequisites

Make sure you have already run the Pre-Steps identified above.

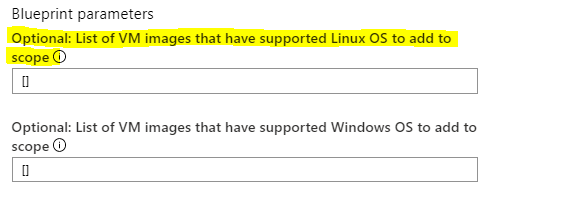
### Steps

1. Navigate to the “Create-GC-Policy-Blueprint” runbook in your Automation Account
2. Select “Edit”
3. Select “Publish”
4. Select “Start” to start the runbook
5. Enter the subscription id of your subscription for the “SUBSCRIPTIONID” parameter, and the cloud usage profile you are intending to comply to for the “CLOUDUSAGEPROFILE” parameter. Note that currently only a cloud usage profile value of “6” or “profile6” is supported, which is the full set of PBMM policies; you can still use this as a starting point for all PBMM Cloud Usage Profiles.
6. Wait until the runbook shows a status of “Completed”
7. Verify that the runbook ran successfully by navigating to Azure Blueprints, selecting the “Blueprint definitions” blade and verifying that the following blueprint definition has been created:

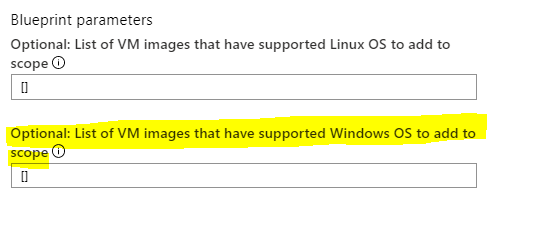


1. You can now assign the newly created blueprint by selecting it and selecting “Assign Blueprint” and entering the following parameter values:

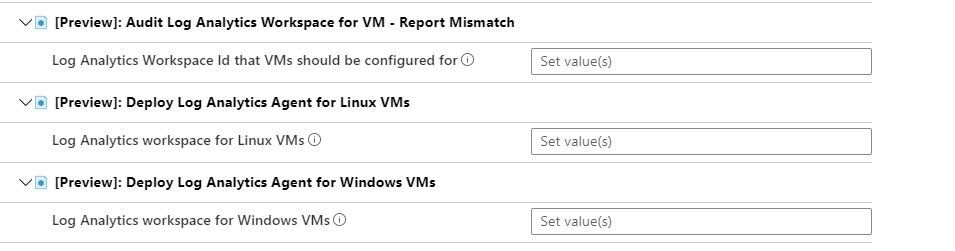
* Select one of the Canadian regions for the Location field.
* Leave the Managed Identity as System Assigned
* If you are using custom Linux OS images for your VMs, enter the image names in JSON format.



* If you are using custom Windows OS images for your VMs, enter the image names in JSON format.



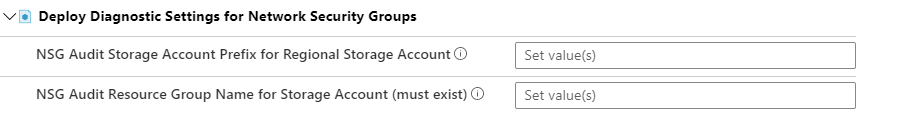
* Enter the Log Analytics Workspace Id to be used in the following fields:



* Specify the Retention days for “Deploy Auditing on SQL servers” policy (minimum 365 days), and an existing resource group name for the storage accounts where the audit records are to be stored:



* Specify the existing resource group name and a storage account prefix to be used for the NSG diagnostic storage account:



1. Select Assign.
2. Navigate to the Assigned Blueprints blade and verify that the assignment completes successfully; this may take some time as it creates a number of policy assignments, etc

## Generate an Azure PBMM Guardrails Compliance Verification Report

If you just assigned the policy blueprint to your subscription, make sure to wait a few hours until the policies have been evaluated for compliance.

This tool runs in one of 2 possible modes:

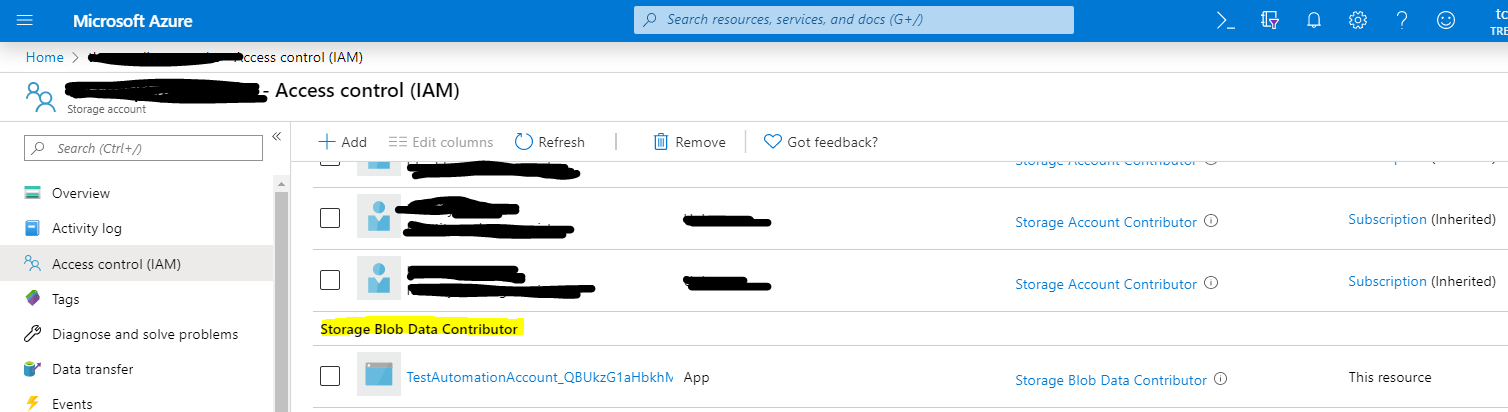
1. Reporting compliance against the same tenant the runbook’s Automation Account is associated with.
2. Reporting compliance against a different tenant than the runbook’s Automation Account is associated with.

Departments and agencies should run the tool using mode 1.

### Prerequisites

1. Make sure you have already run the Pre-Steps identified above.
2. Create a new storage account and container, or use an existing storage account and container to store the compliance report results.

Grant the “Storage Blob Data Contributor” role to your automation account service principal so that the runbook will be able to write to the storage account.



### Steps

1. Navigate to the “Create-GC-Policy-Blueprint” runbook in your Automation Account
2. Select “Edit”
3. Select “Publish”
4. Select “Start” to start the runbook
5. Enter your tenant org name for the TENANTORGANIZATIONNAME either the \*gc.onmicrosoft.com tenant name, or your federated on-prem domain name (if you enabled federation through Azure AD Connect).
6. Enter **yes** for USERUNBOOKRUNASACCOUNT if the Automation Account from which you ran this runbook **is in the same tenant you are running the compliance check against (mode 1 as described above)**, and **no** if you are running a compliance verification against a different tenant (mode 2 as described above).
7. Enter the name of the resource group containing the storage account used to store the results that you created in Prerequisite 2 above for the RESULTSSTORAGEACCOUNTRESOURCEGROUPNAME parameter
8. Enter the name of the storage account used to store the results that you created in Prerequisite 2 above for the RESULTSSTORAGEACCOUNTNAME parameter
9. Enter the name of the container used to store the results that you created in Prerequisite 2 above for the RESULTSCONTAINERNAME parameter
10. If you entered **no** for step 6, you need to:
    1. Ensure you have the credentials for an account with Reader role in the other tenant. These could be user account credentials or the client id and key corresponding to an App Registration in the other tenant.
    2. Create a new Credential within the Automation Account by navigating to the Credentials blade and using the user or registered app credentials.
    3. Enter the name of the new credential that you created in the previous step for the READERACCOUNTCREDENTIALNAME parameter.
11. Select “Ok”

The runbook will take a bit of time to complete. Once it’s completed you can find the results of the compliance verification check in two blobs created within the storage account container you specified:

1. <tenant org name>.json – JSON format version of the compliance report, which can be consumed by a web app and display as HTML
2. <tenant org name>.txt – straight text version of the compliance report

## Exclude resource groups and resources from the Policies assigned by the GC blueprint tool

You can use the “Update-Policy-Scope-Exclusions” runbook to exclude resource group and resource from certain policies or all policies assigned by the GC blueprint tool.

There are two tags that can be used for this purpose:

|  |  |  |
| --- | --- | --- |
| **Tag Name** | **Tag Description** | **Tag value format** |
| PolicyExceptions | List of policyExceptionTagValue names the resource group or resource should be excluded from | Comma separated list of policyExceptionTagValue names. The names are defined in the policyExceptionTagValue setting in the following file:  <https://github.com/canada-ca/cloud-guardrails-azure/blob/master/toolbox/gc-guardrail-blueprint-definitions/all-policies.json>  Example: a **publicIPAllowed** tag value will exclude the resource or resource group from the Network interfaces should not have public IPs Azure policy. |
| CloudUsageProfile | The cloud usage profile associated with the resource. This will control which policies the tagged resource group or resource will be excluded.  **Currently only a value of profile1 (Sandbox) is supported by the tool, and the tool will exclude the resource group or resource from all PBMM policies.** | Example: a tag value of **profile1** will exclude the tagged resource group or resource from all PBMM policies (currently). |

There is a special tag value for the PolicyExceptions tag that can be assigned to VNets that can be used to indicate which subnets within a VNet are allowed to have a direct route to the internet, since this is reported on by the compliance verification check.

The format for the tag value is a JSON string of the format {“<subnet name 1>”: “directRouteToInternetAllowed”, “<subnet name 2>”: “directRouteToInternetAllowed”, …}. In case where all subnets in the VNet should be allowed to have a direct route to the internet, a \* (wildcard) can be used in place of the subnet name.

### Steps to update policy scope exclusions based on the above tag values

1. Navigate to the “Create-GC-Policy-Blueprint” runbook in your Automation Account
2. Select “Edit”
3. Select “Publish”
4. Select “Start” to start the runbook