Exercise: Assign an RBAC Role to a User or Group in Azure Subscription

Role-Based Access Control (RBAC) in Azure is used to grant users or groups access to Azure resources within a particular scope. Here's an exercise that walks you through the steps to assign an RBAC role to a user or group for an Azure subscription:

1. **Sign in to Azure Portal**
    a. Navigate to https://portal.azure.com/
    b. Sign in with an account that has sufficient privileges to assign roles at the subscription level (e.g., Owner or User Access Administrator).
2. **Navigate to Subscriptions**
    a. In the search bar at the top, type “Subscriptions” and select “Subscriptions” from the dropdown results.
3. **Select Your Subscription**
    a. From the list of available subscriptions, click on the subscription where you want to assign the role.
4. **Access Control (IAM)**
    a. In the subscription blade, click on “Access control (IAM)” from the left-hand menu.
5. **Add a Role Assignment**
    a. Click on the “+ Add” button and then select “Add role assignment”.
6. **Configure the Role Assignment**
    a. Role: Click on the "Role" dropdown. A list of roles like 'Contributor', 'Owner', 'Reader', etc., will be displayed. Select the role that you wish to assign to the user or group.
    b. Assign access to: Ensure "User, group, or service principal" is selected.
    c. Select: In the “Members” field, click on the “Select members” button.
        i. In the search box, type the name of the user or group you wish to assign the role to.
        ii. From the search results, select the user or group.
        iii. Click on the "Select" button at the bottom.
    d. Confirm your selections.
7. **Save the Role Assignment**
    a. Click on the “Save” button. Azure will now assign the selected role to the chosen user or group for the specified subscription.
8. **Verify the Assignment**
    a. After assigning the role, you should be redirected back to the “Role assignments” tab under “Access control (IAM)”. You can search for the user or group to verify that they now have the intended role for that subscription.

Conclusion

You've successfully assigned an RBAC role to a user or group for an Azure subscription! This user or group now has permissions based on the role you've assigned within the scope of that subscription.

Note: As always, adhere to the principle of least privilege. Grant users or groups only the permissions they require. Regularly review and audit permissions to ensure security and compliance.

Steps to Create a Custom Role from an Existing Role in an Azure Subscription using Azure Portal

Creating a custom role from an existing role in an Azure subscription involves leveraging the Azure Portal, Azure CLI, or Azure PowerShell. The Azure Portal provides a graphical interface to facilitate this, so we'll outline those steps here:

1. **Sign in to the Azure Portal**
    a. Go to https://portal.azure.com/
    b. Sign in with your Azure account.
2. **Navigate to Subscriptions**
    a. In the search bar at the top, type “Subscriptions” and select it from the dropdown results.
3. **Select Your Subscription**
    a. From the list of subscriptions, click on the one in which you wish to create the custom role.
4. **Access Control (IAM)**
    a. In the selected subscription blade, choose "Access control (IAM)" from the left-hand menu.
5. **Add a custom Role**
    a. Click on the “+ Add” button and then select “Add custom role”.
    b. Select clone a role from Baseline permissions
6. **Find and Select the Role to Copy**
    a. Scroll or search to find the existing role you wish to use as a basis for your custom role.
7. **Define the Custom Role**
    a. Basics:
        i. Name: Provide a unique name for your custom role.
        ii. Description: Offer a short description detailing the purpose of this custom role or how it differs from the original.

    b. Permissions:
        i. You'll see a list of permissions inherited from the original role. You can add or remove permissions based on your requirements.
    c. Assignable Scopes:
        i. By default, the scope might be set to the current subscription. If you need the custom role to be available for other subscriptions or specific resource groups, adjust the scopes accordingly.
8. **Save the Custom Role**
    a. Once you've configured the role to your liking, click the “Save” or “Create” button to finalize your custom role.
9. **Verification**
    a. Back in the “Role definitions” tab, you should see your newly created custom role. Click on it to review the permissions and ensure they match your intentions.

Remember to practice the principle of least privilege when creating and assigning roles. This ensures that users, groups, or services have only the permissions necessary to perform their tasks, enhancing security and reducing the potential for unintended changes or access.

Steps to Assign a Policy to an Azure Subscription

Assigning a policy to an Azure subscription involves defining rules that enforce specific conditions or effects for resources within that subscription. Here are the steps to assign a policy to a subscription using the Azure Portal:

1. Sign in to Azure Portal
    a. Navigate to https://portal.azure.com/
    b. Log in with your Azure credentials.
2. Navigate to Policy Service
    a. In the search bar at the top, type “Policy” and select "Policy" from the dropdown results.
3. Go to Assignments
    a. On the Policy blade, click on “Assignments” in the left-hand menu.
4. Add a Policy Assignment
    a. Click on the "+ Assign Policy" button at the top of the blade.
5. Select the Subscription
    a. In the "Scope" section:
        i. Click on the ellipsis (...) next to the "Subscription" field.
        ii. Choose the desired subscription to which you want to assign the policy.

        iii. Click on the "Select" button at the bottom.(Note: If you wish to narrow down the policy application to a specific resource group within the subscription, you can select that as well.)
6. Select the Policy Definition
    a. Under the "Policy definition" section, click on the ellipsis (...). App Service apps should only be accessible over HTTPS
    b. Search and select the policy definition you want to assign.
    c. Click on the "Select" button.
7. Configure Parameters (If Applicable)
    a. Some policy definitions require additional parameters. If the chosen policy has parameters, fill them in accordingly.
8. Policy Assignment Settings
    a. Assignment name: Provide a name for the policy assignment.
    b. Description (Optional): Add a description for clarity, especially if other team members might manage or review this policy.
    c. Policy enforcement: Ensure it's set to "Enabled" if you want the policy to be enforced. If you're testing or auditing without enforcement, set it to "Disabled".
9. Review and Save
    a. After configuring the policy assignment settings, click on the “Review + create” button at the bottom.
    b. Ensure all details are correct, then click on the “Create” button to assign the policy to the chosen subscription.
10. Verification
    a. After assigning the policy, you'll return to the "Assignments" blade. Here, you can verify your policy assignment. It should be listed with the specified subscription scope.

Remember, Azure Policy evaluates every resource in the scope for compliance with the conditions defined in the policy. Non-compliant resources are flagged, and depending on the policy effects, may be automatically corrected, audited, or even denied creation. Always test new policies in non-production environments first to understand their impact.

Test the policy

1. Navigate to the Bicep Folder
    a. Access the /bicep/ directory.
2. Edit the Parameter File
    a. Open the file /configDev/Lab3.parameters.json.
    b. Modify the value of the “projectName” parameter.
3. Deploy Using Bicep Template

First, ensure you're logged in and have set the correct subscription:

```
az login
az account set --subscription [Your Subscription ID]
```

Then, deploy the app service using the following command:

```
az deployment group create --resource-group [Your Resource Group Name]
--template-file generic.main.bicep --parameters
./configDev/DevTemplateCoraxRG.parameters.json
```

4. Notice the deployment error

5. Discuss and remediate the error
6. Redeploy the template
    a. In the Azure Portal, navigate to your resource group.
    b. Under "Settings", click on the "Deployments" tab.
    c. Monitor the deployment progress and wait for it to complete.
7. Access the App Service
    a. From the resource group's main page, locate and open the newly created app service.
8. Modify Configuration Settings
    a. Within the app service, select "Configuration" from the left-hand menu.
    b. Click on the "General settings" tab.
    c. Locate the "Https Only" setting and toggle it to "false".
9. Click on "Save".
10. Review the Outcome
11. Observe the error message (if any) after attempting to save.