Azure Terraform

1. Create the App Registration (Service Principal Identity) 🤵

This step creates the identity in Microsoft Entra ID (formerly Azure Active Directory) that Terraform will use to log in.

1. **Sign in** to the Azure Portal.
2. In the search bar, type `Microsoft Entra ID` and select the service.
3. In the left-hand menu, under **Manage**, select **App registrations**.
4. Click **+ New registration**.
5. On the **Register an application** page, provide the following:
   - **Name:** Enter a meaningful name, e.g., `Terraform-SP-for-Prod`.
   - **Supported account types:** Select **Accounts in this organizational directory only (Default Directory only - Single tenant)** (This is typically the correct choice).
   - **Redirect URI (optional):** Leave this blank or set to `Web` and leave the URL blank.
6. Click the **Register** button.

Capture Required IDs

Once the registration is complete, you will be taken to the App's **Overview** page. Immediately copy and save the following two values, as they are crucial for Terraform:

- **Application (client) ID:** (This is your `ARM_CLIENT_ID`)
- **Directory (tenant) ID:** (This is your `ARM_TENANT_ID`)

2. Create the Client Secret (The "Password") 🔒

This is the credential (secret key) that Terraform will use along with the IDs to authenticate.

1. From the App Registration's menu (left side), select **Certificates & secrets**.
2. Click the **+ New client secret** button.
3. On the **Add a client secret** pane:

- **Description:** Enter a description, e.g., `Terraform Secret`.
- **Expires:** Select an appropriate expiration period (e.g., 24 months). **Best practice** is to choose the shortest duration possible and set a calendar reminder to rotate it.

4. Click **Add**.
5. **IMMEDIATELY COPY THE VALUE.** The **Value** of the client secret will only be displayed **once**. If you navigate away, you must create a new secret.
   - **Value:** (This is your `ARM_CLIENT_SECRET`). **Save this value securely.**

3. Assign a Role (Grant Permissions) ✅

The Service Principal exists, but it doesn't have permission to manage Azure resources yet. You must assign a role to it. The **scope** is the level where you assign the role (e.g., Subscription, Resource Group).

1. Navigate to the scope you want Terraform to manage (e.g., search for **Subscriptions** and select your target subscription).
2. In the left-hand menu for the Subscription, select **Access control (IAM)**.
3. Click **+ Add** and select **Add role assignment**.
4. On the **Role** tab:
   - Select the role: Search for and select the **Contributor** role (for full management access) or a more restricted role based on the Principle of Least Privilege.
5. Click **Next**.
6. On the **Members** tab:
   - **Assign access to:** Ensure this is set to **User, group, or service principal**.
   - Click **+ Select members**.
   - Search for the name of the App Registration you created in Step 1 (e.g., `Terraform-SP-for-Prod`).
   - Select the Service Principal from the results.
7. Click **Review + assign** to complete the assignment.

Capture Final ID

- **Subscription ID:** (This is your `ARM_SUBSCRIPTION_ID`). You can get this from the Subscription's **Overview** page.

4. Configure Terraform

With the four required values obtained from the GUI steps, you can now configure Terraform to use them via environment variables:

Credential Name	Azure Portal Location	Terraform Variable
Client ID	App Registration **Overview**	`ARM_CLIENT_ID`
Client Secret	App Registration **Certificates & secrets**	`ARM_CLIENT_SECRET`
Tenant ID	App Registration **Overview**	`ARM_TENANT_ID`
Subscription ID	**Subscription** Overview page	`ARM_SUBSCRIPTION_ID`

You then use these in your shell (e.g., Linux/macOS):

```
export ARM_SUBSCRIPTION_ID="<your-subscription-id>"
export ARM_TENANT_ID="<tenant-id>"
export ARM_CLIENT_ID="<application-id/client-id>"
export ARM_CLIENT_SECRET="<client-secret>"
```

And your Terraform configuration will look like this:

```
terraform {
  required_providers {
    azurerm = {
      source  = "hashicorp/azurerm"
      version = "~> 3.0"
    }
  }
}

 "azurerm" ">provider "azurerm" {
  features {}
  # Terraform will automatically read the ARM_* environment variables
}
```

===