Microsoft Entra Verified ID

Hands On Lab

Contents

Pre-requirements

- Welcome to the Microsoft 365 Developer Program | Microsoft Docs
- Set up a Microsoft 365 developer sandbox subscription | Microsoft Docs
- Install GIT https://git-scm.com/downloads
- Install Visual Studio Code https://code.visualstudio.com/Download
- Install .NET 5 https://dotnet.microsoft.com/download/dotnet/5.0
- Download and sign up for a free account https://ngrok.com/

Your account needs to have Global Administrator or Authentication Policy Administrator permission for the setup part of this workshop

Setup the Verified ID service

[Tutorial - Configure your tenant for Microsoft Entra Verified ID - Microsoft Entra | Microsoft Docs]

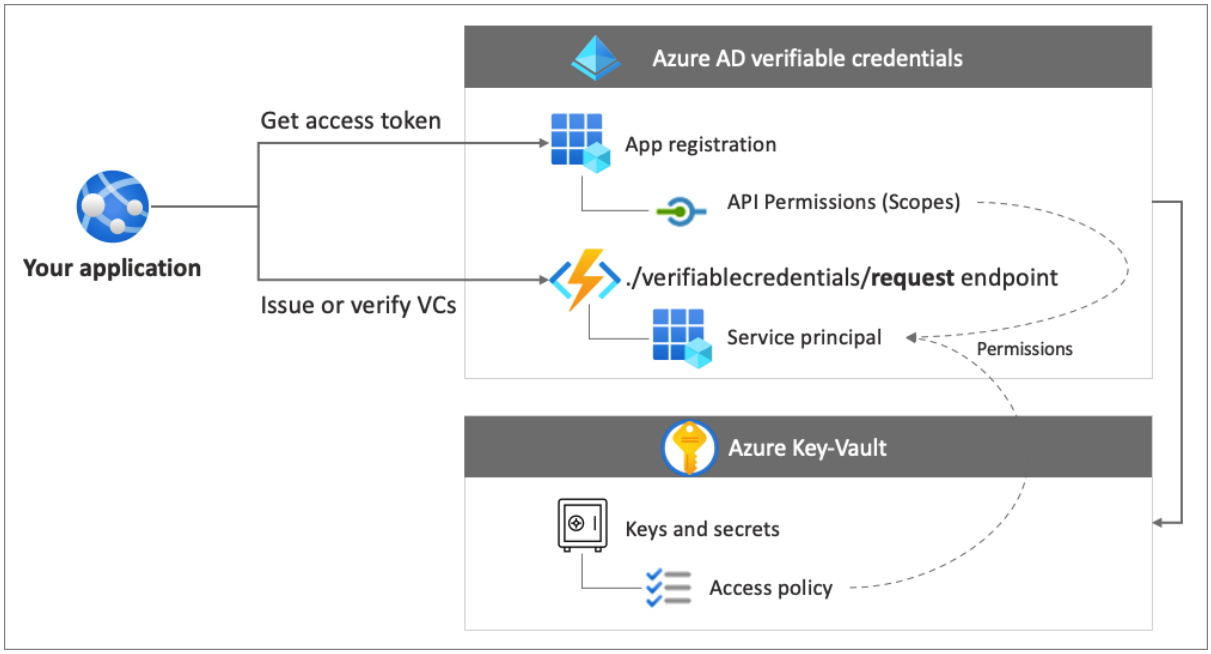
Microsoft Entra Verified ID is a decentralized identity solution that helps you safeguard your organization. The service allows you to issue and verify credentials. Issuers can use the Verified ID service to issue their own customized verifiable credentials. Verifiers can use the service's free REST API to easily request and accept verifiable credentials in their apps and services.

In this part of the workshop, you learn how to configure your Azure AD tenant to use the verifiable credentials service.

Specifically, you learn how to:

- Create an Azure Key Vault instance.
- Set up the Verifiable Credentials service.
- Register an application in Azure AD.

The following diagram illustrates the Verified ID architecture and the component you configure.



Create a key vault

Azure Key Vault is a cloud service that enables the secure storage and access of secrets and keys. The Verified ID service stores public and private keys in Azure Key Vault. These keys are used to sign and verify credentials.

If you don't have an Azure Key Vault instance available, follow these steps to create a key vault using the Azure portal.

Create a vault

1. From the Azure portal menu, or from the Home page, select Create a resource.
2. In the Search box, enter Key Vault.
3. From the results list, choose Key Vault.
4. On the Key Vault section, choose Create.
5. On the Create key vault section provide the following information:

6. Name: A unique name is required. For this hands on, we use identitysummit-<number>.
7. Subscription: Choose a subscription.
8. Under Resource Group, choose Create new and enter a resource group name.
9. In the Location pull-down menu, choose a location.
10. Leave the other options to their defaults.
11. After providing the information above, select Create.

Take note of the two properties listed below:

Vault Name: In the example, this is identitysummit-<number>. You will use this name for other steps.

Vault URI: In the example, this is https:// identitysummit-<number>.vault.azure.net/. Applications that use your vault through its REST API must use this URI.

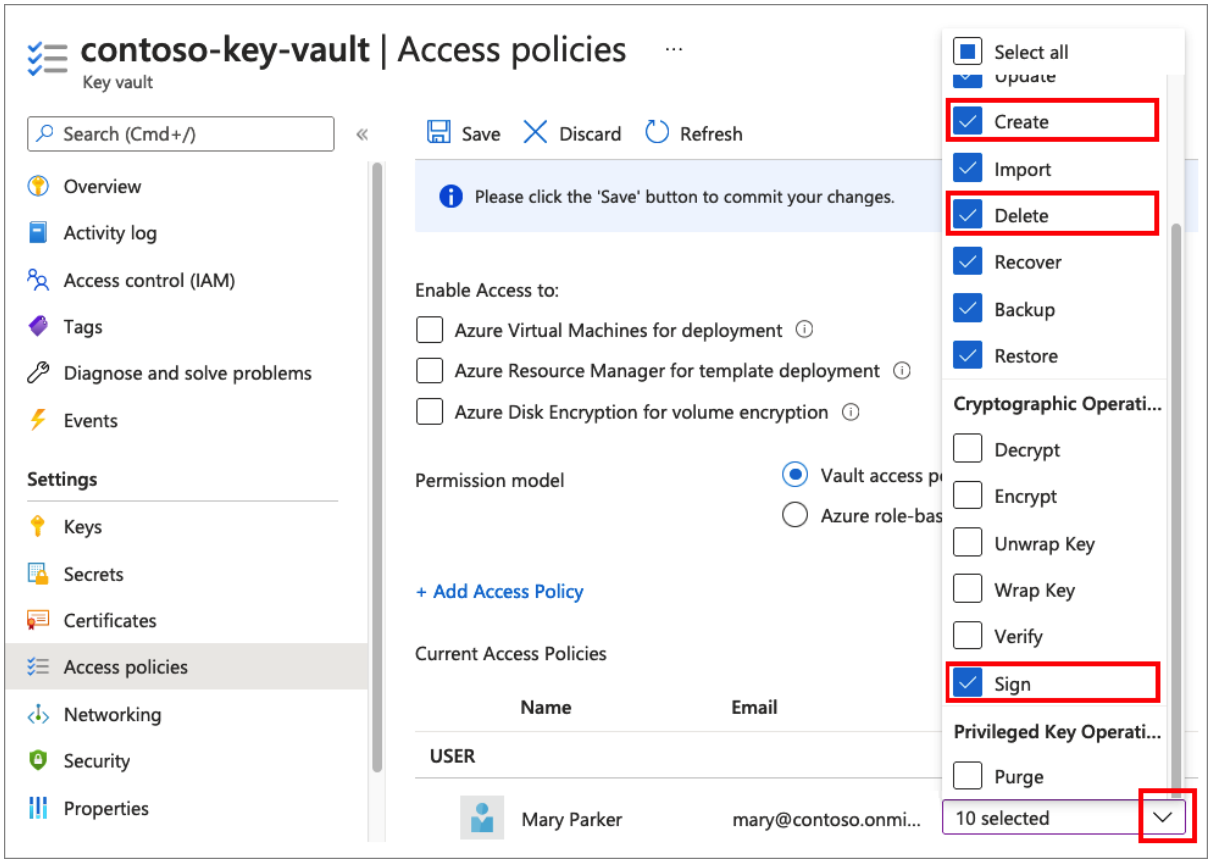
At this point, your Azure account is the only one authorized to perform operations on this new vault.

Set access policies for the key vault

A Key Vault access policy defines whether a specified security principal can perform operations on Key Vault secrets and keys. Set access policies in your key vault for both the Verified ID service administrator account, and for the Request Service API principal that you created. After you create your key vault, Verifiable Credentials generates a set of keys used to provide message security. These keys are stored in Key Vault. You use a key set for signing, updating, and recovering verifiable credentials.

Set access policies for the Verified ID Admin user

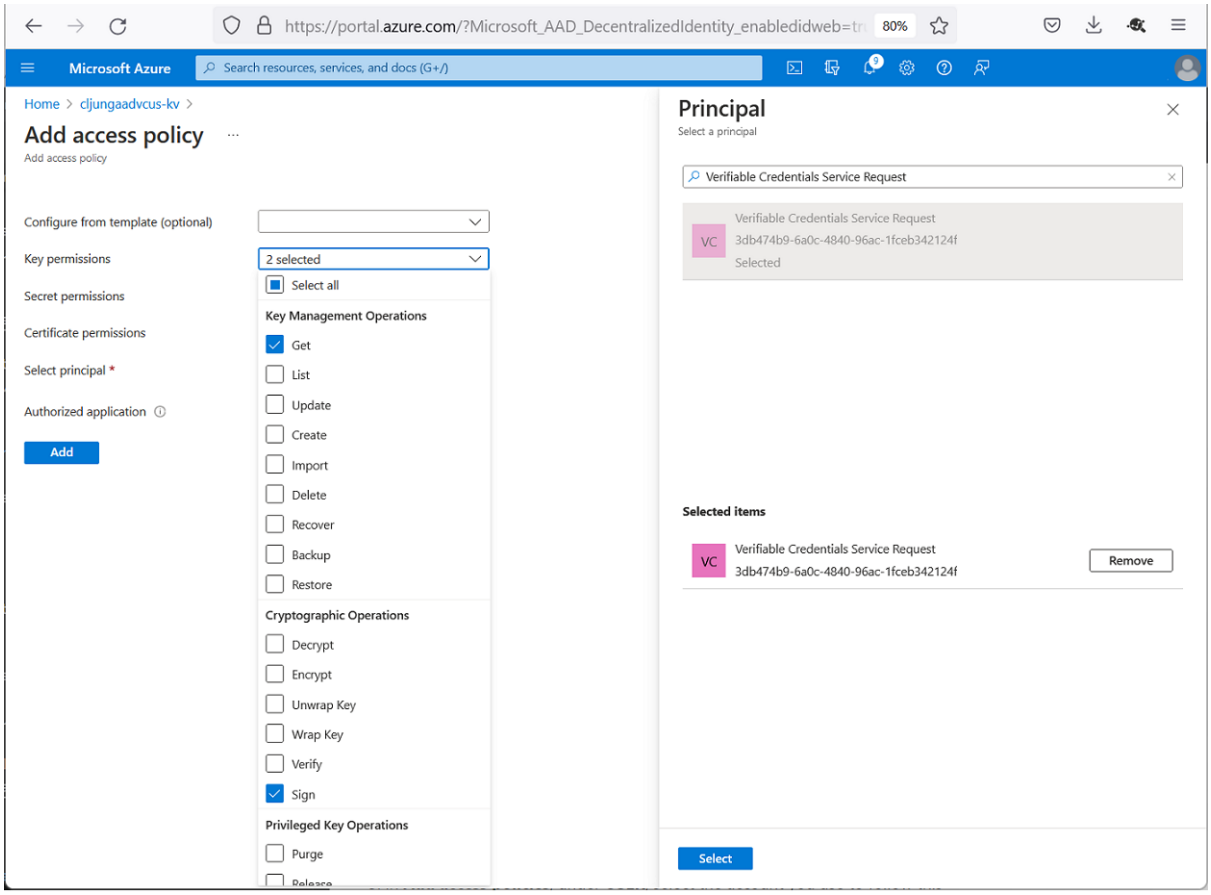
1. In the Azure portal, go to the key vault you use for this tutorial.
2. Under Settings, select Access policies.
3. In Add access policies, under USER, select the account you use to follow this tutorial.
4. For Key permissions, verify that the following permissions are selected: **Create, Delete, and Sign**. By default, Create and Delete are already enabled. Sign should be the only key permission you need to update.



Set access policies for the Verifiable credentials service request service principal

The Verifiable credentials service request is the Request Service API, and it needs access to Key Vault in order to sign issuance and presentation requests.

1. Select + Add Access Policy and select the service principal Verifiable Credentials Service Request with AppId **3db474b9-6a0c-4840-96ac-1fceb342124**.
2. For Key permissions, select permissions **Get and Sign**.



To save the changes, select Save.

Create Azure Storage

The easiest way for a developer to get a domain to use for linked domain is to use Azure Storage's static website feature. You can't control what the domain name will be, other than it will contain your storage account name as part of its hostname.

Follow these steps to quickly set up a domain to use for Linked Domain:

1. Create an Azure Storage account. During storage account creation, choose StorageV2 (general-purpose v2 account) and Locally redundant storage (LRS).
    a. Go to the Azure Portal
    b. Click Create a new resource
    c. Search for Storage account and select storage account
    d. Press create
    e. Choose a subscription and resource group
    f. Pick a name, region
    g. Select standard
    h. Select LRS (Local redundant storage)
    i. Press review and then create
    j. Wait until the deployment is finished and press go to resource
2. Go to that Storage Account and select Static website in the left hand menu and enable static website. If you can't see the Static website menu item, you didn't create a V2 storage account.

3. Copy the primary endpoint name that appears after saving. This value is your domain name. It looks something like https://<your-storageaccountname>.z6.web.core.windows.net/.

Set up Verified ID

To set up Verified ID, follow these steps:

1. In the Azure portal, search for Verified ID. Then, select Verified ID.
2. From the left menu, select Getting started.
3. Set up your organization by providing the following information:
    a. Organization name: Enter a name to reference your business within Verifiable Credentials. Your customers don't see this name.
    b. Domain: *Enter the domain name of the primary endpoint name*
    c. Key vault: Select the key vault that you created earlier.
4. Select Save and get started.

Register an application in Azure AD

Verified ID needs to get access tokens to issue and verify. To get access tokens, register a web application and grant API permission for the API Verified ID Request Service that you set up in the previous step.

1. Sign in to the Azure portal with your administrative account.
2. If you have access to multiple tenants, select the Directory + subscription. Then, search for and select your Azure Active Directory.
3. Under Manage, select App registrations > New registration.
4. Enter a display name for your application. For example: verifiable-credentials-app.
5. For Supported account types, select Accounts in this organizational directory only (Default Directory only - Single tenant).
6. Select Register to create the application.

Grant permissions to get access tokens

In this step, you grant permissions to the Verifiable Credentials Service Request Service principal.

To add the required permissions, follow these steps:

1. Stay in the verifiable-credentials-app application details page. Select API permissions > Add a permission.
2. Select APIs my organization uses.
3. Search for the service principal that you created earlier, Verifiable Credentials Service Request, and select it.
4. Choose Application Permission, and expand **VerifiableCredential.Create.All**.
5. Select Add permissions.
6. Select Grant admin consent for <your tenant name>.

Service endpoint configuration

1. Navigate to the Verified ID in the Azure portal.
2. Select Registration.
3. Notice that there are two sections:
    a. Website ID registration
    b. Domain verification.
4. Select on each section and download the JSON file under each.

1. Go to the Storage Account that you created previously and select Containers in the left hand menu. Then select the container named $web.
2. Before uploaded, open the Advanced section and specify .well-known in the Upload to folder textbox.
3. Select Upload and select on the folder icon to find your file
    a. **You need to upload the did.json and the did-configuration.json**
4. Upload the files.

Become an issuer

Tutorial - Issue Microsoft Entra Verified ID credentials from an application - Microsoft Entra | Microsoft Docs

Become an verifier

Tutorial - Configure Microsoft Entra Verified ID verifier - Microsoft Entra | Microsoft Docs