» Data Source: azuread_application

Use this data source to access information about an existing Application within Azure Active Directory.

NOTE: If you're authenticating using a Service Principal then it must have permissions to both `Read and write all applications` and `Sign in and read user profile` within the `Windows Azure Active Directory` API.

» Example Usage

```
data "azuread_application" "test" {
  name = "My First AzureAD Application"
}

output "azure_ad_object_id" {
  value = "${data.azuread_application.test.id}"
}
```

» Argument Reference

- `object_id` - (Optional) Specifies the Object ID of the Application within Azure Active Directory.
- `name` - (Optional) Specifies the name of the Application within Azure Active Directory.

NOTE: Either an `object_id` or `name` must be specified.

» Attributes Reference

- `id` - the Object ID of the Azure Active Directory Application.
- `application_id` - the Application ID of the Azure Active Directory Application.
- `available_to_other_tenants` - Is this Azure AD Application available to other tenants?
- `identifier_uris` - A list of user-defined URI(s) that uniquely identify a Web application within it's Azure AD tenant, or within a verified custom domain if the application is multi-tenant.
- `oauth2_allow_implicit_flow` - Does this Azure AD Application allow OAuth2.0 implicit flow tokens?
- `object_id` - the Object ID of the Azure Active Directory Application.

- `reply_urls` - A list of URLs that user tokens are sent to for sign in, or the redirect URIs that OAuth 2.0 authorization codes and access tokens are sent to.
- `required_resource_access` - A collection of `required_resource_access` blocks as documented below.

`required_resource_access` block exports the following:

- `resource_app_id` - The unique identifier for the resource that the application requires access to.
- `resource_access` - A collection of `resource_access` blocks as documented below

`resource_access` block exports the following:

- `id` - The unique identifier for one of the `OAuth2Permission` or `AppRole` instances that the resource application exposes.
- `type` - Specifies whether the id property references an `OAuth2Permission` or an `AppRole`.

» Data Source: azuread_domains

Use this data source to access information about an existing Domains within Azure Active Directory.

NOTE: If you're authenticating using a Service Principal then it must have permissions to `Directory.Read.All` within the `Windows Azure Active Directory` API.

» Example Usage

```
data "azuread_domains" "aad_domains" {}

output "domains" {
  value = "${data.azuread_domains.aad_domains.domains}"
}
```

» Argument Reference

- `include_unverified` - (Optional) Set to `true` if unverified Azure AD Domains should be included. Defaults to `false`.
- `only_default` - (Optional) Set to `true` to only return the default domain.
- `only_initial` - (Optional) Set to `true` to only return the initial domain, which is your primary Azure Active Directory tenant domain. Defaults to `false`.

NOTE: If `include_unverified` is set to `true` you cannot specify `only_default` or `only_initial`. Additionally you cannot combine `only_default` with `only_initial`.

» Attributes Reference

- `domains` - One or more `domain` blocks as defined below.

The `domain` block contains:

- `domain_name` - The name of the domain.
- `authentication_type` - The authentication type of the domain (Managed or Federated).
- `is_default` - `True` if this is the default domain that is used for user creation.
- `is_initial` - `True` if this is the initial domain created by Azure Activie Directory.
- `is_verified` - `True` if the domain has completed domain ownership verification.

» Data Source: azuread_group

Gets information about an Azure Active Directory group.

NOTE: If you're authenticating using a Service Principal then it must have permissions to `Read directory data` within the `Windows Azure Active Directory` API.

» Example Usage (by Group Display Name)

```
data "azuread_group" "test_group" {
  name = "MyTestGroup"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The Name of the Azure AD Group we want to lookup.

WARNING: `name` is not unique within Azure Active Directory. The data source will only return the first Group found.

» Attributes Reference

The following attributes are exported:

- `id` - The Object ID of the Azure AD Group.

» Data Source: azuread_service_principal

Gets information about an existing Service Principal associated with an Application within Azure Active Directory.

NOTE: If you're authenticating using a Service Principal then it must have permissions to both `Read and write all applications` and `Sign in and read user profile` within the `Windows Azure Active Directory` API.

» Example Usage (by Application Display Name)

```
data "azuread_service_principal" "test" {
  display_name = "my-awesome-application"
}
```

» Example Usage (by Application ID)

```
data "azuread_service_principal" "test" {
  application_id = "00000000-0000-0000-0000-000000000000"
}
```

» Example Usage (by Object ID)

```
data "azuread_service_principal" "test" {
  object_id = "00000000-0000-0000-0000-000000000000"
}
```

» Argument Reference

The following arguments are supported:

- `application_id` - (Optional) The ID of the Azure AD Application for which to create a Service Principal.
- `object_id` - (Optional) The ID of the Azure AD Service Principal.
- `display_name` - (Optional) The Display Name of the Azure AD Application associated with this Service Principal.

NOTE: At least one of `application_id`, `display_name` or `object_id` must be specified.

» Attributes Reference

The following attributes are exported:

- `id` - The Object ID for the Service Principal.

» Data Source: azuread_user

Gets information about an Azure Active Directory user.

NOTE: If you're authenticating using a Service Principal then it must have permissions to `Read directory data` within the `Windows Azure Active Directory` API.

» Example Usage

```
data "azuread_user" "test_user" {
  user_principal_name = "john@hashicorp.com"
}
```

» Argument Reference

The following arguments are supported:

- `user_principal_name` - (Required) The User Principal Name of the Azure AD User.

» Attributes Reference

The following attributes are exported:

- `id` - The Object ID of the Azure AD User.
- `user_principal_name` - The User Principal Name of the Azure AD User.
- `account_enabled` - `True` if the account is enabled; otherwise `False`.
- `display_name` - The Display Name of the Azure AD User.
- `mail` - The primary email address of the Azure AD User.
- `mail_nickname` - The email alias of the Azure AD User.

» azuread_application

Manages an Application within Azure Active Directory.

NOTE: If you're authenticating using a Service Principal then it must have permissions to both `Read and write all applications` and `Sign in and read user profile` within the `Windows Azure Active Directory` API.

» Example Usage

```
resource "azuread_application" "test" {
  name                       = "example"
  homepage                   = "https://homepage"
  identifier_uris            = ["https://uri"]
  reply_urls                 = ["https://replyurl"]
  available_to_other_tenants = false
  oauth2_allow_implicit_flow = true

  required_resource_access {
    resource_app_id = "00000003-0000-0000-c000-000000000000"

    resource_access {
      id = "..."
      type = "Role"
    }
    resource_access {
      id = "..."
      type = "Scope"
    }

    resource_access {
      id = "..."
      type = "Scope"
```

```
    }
  }

  required_resource_access {
    resource_app_id = "00000002-0000-0000-c000-000000000000"

    resource_access {
      id = "..."
      type = "Scope"
    }
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The display name for the application.
- `homepage` - (optional) The URL to the application's home page. If no homepage is specified this defaults to `https://{name}`.
- `identifier_uris` - (Optional) A list of user-defined URI(s) that uniquely identify a Web application within it's Azure AD tenant, or within a verified custom domain if the application is multi-tenant.
- `reply_urls` - (Optional) A list of URLs that user tokens are sent to for sign in, or the redirect URIs that OAuth 2.0 authorization codes and access tokens are sent to.
- `available_to_other_tenants` - (Optional) Is this Azure AD Application available to other tenants? Defaults to `false`.
- `oauth2_allow_implicit_flow` - (Optional) Does this Azure AD Application allow OAuth2.0 implicit flow tokens? Defaults to `false`.
- `required_resource_access` - (Optional) A collection of `required_resource_access` blocks as documented below.

`required_resource_access` supports the following:

- `resource_app_id` - (Required) The unique identifier for the resource that the application requires access to. This should be equal to the appId declared on the target resource application.
- `resource_access` - (Required) A collection of `resource_access` blocks as documented below

---

`resource_access` supports the following:

- `id` - (Required) The unique identifier for one of the `OAuth2Permission` or `AppRole` instances that the resource application exposes.
- `type` - (Required) Specifies whether the id property references an `OAuth2Permission` or an `AppRole`. Possible values are `Scope` or `Role`.

» Attributes Reference

The following attributes are exported:

- `application_id` - The Application ID.

» Import

Azure Active Directory Applications can be imported using the `object id`, e.g.

```
terraform import azuread_application.test 00000000-0000-0000-0000-000000000000
```

» azuread_group

Manages a Group within Azure Active Directory.

NOTE: If you're authenticating using a Service Principal then it must have permissions to `Read and write all groups` within the `Windows Azure Active Directory` API. In addition it must also have either the `Company Administrator` or `User Account Administrator` Azure Active Directory roles assigned in order to be able to delete groups. You can assign one of the required Azure Active Directory Roles with the **AzureAD PowerShell Module**, which is available for Windows PowerShell or in the Azure Cloud Shell. Please refer to this documentation for more details.

» Example Usage

```
resource "azuread_group" "my_group" {
  name = "MyGroup"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The display name for the Group.

NOTE: Group names are not unique within Azure Active Directory.

» Attributes Reference

The following attributes are exported:

- `id` - The Object ID of the Group.
- `name` - The Display Name of the Group.

» Import

Azure Active Directory Groups can be imported using the `object id`, e.g.

```
terraform import azuread_group.my_group 00000000-0000-0000-0000-000000000000
```

» azuread_service_principal

Manages a Service Principal associated with an Application within Azure Active Directory.

NOTE: If you're authenticating using a Service Principal then it must have permissions to both `Read and write all applications` and `Sign in and read user profile` within the `Windows Azure Active Directory` API.

» Example Usage

```
resource "azuread_application" "test" {
  name                       = "example"
  homepage                   = "http://homepage"
  identifier_uris            = ["http://uri"]
  reply_urls                 = ["http://replyurl"]
  available_to_other_tenants = false
  oauth2_allow_implicit_flow = true
}

resource "azuread_service_principal" "test" {
  application_id = "${azuread_application.test.application_id}"

  tags = ["example", "tags", "here"]
}
```

» Argument Reference

The following arguments are supported:

- `application_id` - (Required) The ID of the Azure AD Application for which to create a Service Principal.
- `tags` - (Optional) A list of tags to apply to the Service Principal.

» Attributes Reference

The following attributes are exported:

- `id` - The Object ID for the Service Principal.
- `display_name` - The Display Name of the Azure Active Directory Application associated with this Service Principal.

» Import

Azure Active Directory Service Principals can be imported using the `object id`, e.g.

```
terraform import azuread_service_principal.test 00000000-0000-0000-0000-000000000000
```

» azuread_service_principal_password

Manages a Password associated with a Service Principal within Azure Active Directory.

NOTE: If you're authenticating using a Service Principal then it must have permissions to both `Read and write all applications` and `Sign in and read user profile` within the `Windows Azure Active Directory` API.

» Example Usage

```
resource "azuread_application" "test" {
  name                       = "example"
  homepage                   = "http://homepage"
  identifier_uris            = ["http://uri"]
  reply_urls                 = ["http://replyurl"]
  available_to_other_tenants = false
  oauth2_allow_implicit_flow = true
}
```

```
resource "azuread_service_principal" "test" {
  application_id = "${azuread_application.test.application_id}"
}

resource "azuread_service_principal_password" "test" {
  service_principal_id = "${azuread_service_principal.test.id}"
  value                = "VT=uSgbTanZhyz@%nL9Hpd+Tfay_MRV#"
  end_date             = "2020-01-01T01:02:03Z"
}
```

» Argument Reference

The following arguments are supported:

- `service_principal_id` - (Required) The ID of the Service Principal for which this password should be created. Changing this field forces a new resource to be created.
- `value` - (Required) The Password for this Service Principal.
- `end_date` - (Optional) The End Date which the Password is valid until, formatted as a RFC3339 date string (e.g. `2018-01-01T01:02:03Z`). Changing this field forces a new resource to be created.
- `end_date_relative` - (Optional) A relative duration for which the Password is valid until, for example `240h` (10 days) or `2400h30m`. Changing this field forces a new resource to be created.

NOTE: One of `end_date` or `end_date_relative` must be set.

- `key_id` - (Optional) A GUID used to uniquely identify this Key. If not specified a GUID will be created. Changing this field forces a new resource to be created.
- `start_date` - (Optional) The Start Date which the Password is valid from, formatted as a RFC3339 date string (e.g. `2018-01-01T01:02:03Z`). If this isn't specified, the current date is used. Changing this field forces a new resource to be created.

» Attributes Reference

The following attributes are exported:

- `id` - The Key ID for the Service Principal Password.

» Import

Service Principal Passwords can be imported using the `object id`, e.g.

```
terraform import azuread_service_principal_password.test 00000000-0000-0000-0000-0000000000
```

NOTE: This ID format is unique to Terraform and is composed of the Service Principal's Object ID and the Service Principal Password's Key ID in the format `{ServicePrincipalObjectId}/{ServicePrincipalPasswordKeyId}`.

» azuread_user

Manages a User within Azure Active Directory.

NOTE: If you're authenticating using a Service Principal then it must have permissions to `Directory.ReadWrite.All` within the `Windows Azure Active Directory` API.

» Example Usage

```
resource "azuread_user" "test_user" {
  user_principal_name = "john@hashicorp.com"
  display_name        = "John Doe"
  mail_nickname       = "johnd"
  password            = "SecretP@sswd99!"
}
```

» Argument Reference

The following arguments are supported:

- `user_principal_name` - (Required) The User Principal Name of the Azure AD User.
- `display_name` - (Required) The name to display in the address book for the user.
- `account_enabled` - (Optional) `true` if the account should be enabled, otherwise `false`. Defaults to `true`.
- `mail_nickname`- (Optional) The mail alias for the user. Defaults to the user name part of the User Principal Name.
- `password` - (Required) The password for the User. The password must satisfy minimum requirements as specified by the password policy. The maximum length is 16 characters.
- `force_password_change` - (Optional) `true` if the User is forced to change the password during the next sign-in. Defaults to `false`.

» Attributes Reference

The following attributes are exported:

- `id` - The Object ID of the Azure AD User.
- `mail` - The primary email address of the Azure AD User.