NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

Azure Service Management Provider

The Azure Service Management provider is used to interact with the many resources supported by Azure. The provider needs to be configured with a publish settings file (https://manage.windowsazure.com/publishsettings) and optionally a subscription ID before it can be used.

Use the navigation to the left to read about the available resources.

Example Usage

```
# Configure the Azure Provider
provider "azure" {
  publish_settings = "${file("credentials.publishsettings")}"
}

# Create a web server
resource "azure_instance" "web" {
  # ...
}
```

Argument Reference

The following arguments are supported:

- `publish_settings` - (Optional) Contents of a valid `publishsettings` file, used to authenticate with the Azure API. You can download the settings file here: https://manage.windowsazure.com/publishsettings (https://manage.windowsazure.com/publishsettings). You must either provide publish settings or both a `subscription_id` and `certificate`. It can also be sourced from the `AZURE_PUBLISH_SETTINGS` environment variable.
- `subscription_id` - (Optional) The subscription ID to use. If a `settings_file` is not provided `subscription_id` is required. It can also be sourced from the `AZURE_SUBSCRIPTION_ID` environment variable.
- `certificate` - (Optional) The certificate used to authenticate with the Azure API. If a `settings_file` is not provided `certificate` is required. It can also be sourced from the `AZURE_CERTIFICATE` environment variable.

These arguments are supported for backwards compatibility, and may be removed in a future version:

- `settings_file` - **Deprecated: please use `publish_settings` instead.** Path to or contents of a valid `publishsettings` file, used to authenticate with the Azure API. You can download the settings file here: https://manage.windowsazure.com/publishsettings (https://manage.windowsazure.com/publishsettings). You must either provide (or source from the `AZURE_SETTINGS_FILE` environment variable) a settings file or both a

`subscription_id` and `certificate`.

Testing:

The following environment variables must be set for the running of the acceptance test suite:

- A valid combination of the above which are required for authentification.
- `AZURE_STORAGE` - The name of a storage account to be used in tests which require a storage backend. The storage account needs to be located in the Western US Azure region.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_affinity_group

Creates a new affinity group on Azure.

Example Usage

```
resource "azure_affinity_group" "terraform-main-group" {
  name        = "terraform-group"
  location    = "North Europe"
  label       = "tf-group-01"
  description = "Affinity group created by Terraform."
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the affinity group. Must be unique on your Azure subscription.
- `location` - (Required) The location where the affinity group should be created. For a list of all Azure locations, please consult this link (https://azure.microsoft.com/en-us/regions/).
- `label` - (Required) A label to be used for tracking purposes.
- `description` - (Optional) A description for the affinity group.

Attributes Reference

The following attributes are exported:

- `id` - The affinity group ID. Coincides with the given `name`.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_data_disk

Adds a data disk to a virtual machine. If the name of an existing disk is given, it will attach that disk. Otherwise it will create and attach a new empty disk.

Example Usage

```
resource "azure_data_disk" "data" {
  lun                 = 0
  size                = 10
  storage_service_name = "yourstorage"
  virtual_machine     = "server1"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Optional) The name of an existing registered disk to attach to the virtual machine. If left empty, a new empty disk will be created and attached instead. Changing this forces a new resource to be created.
- `label` - (Optional) The identifier of the data disk. Changing this forces a new resource to be created (defaults to "virtual_machine-lun")
- `lun` - (Required) The Logical Unit Number (LUN) for the disk. The LUN specifies the slot in which the data drive appears when mounted for usage by the virtual machine. Valid LUN values are 0 through 31.
- `size` - (Optional) The size, in GB, of an empty disk to be attached to the virtual machine. Required when creating a new disk, not used otherwise.
- `caching` - (Optional) The caching behavior of data disk. Valid options are: `None`, `ReadOnly` and `ReadWrite` (defaults `None`)
- `storage_service_name` - (Optional) The name of an existing storage account within the subscription which will be used to store the VHD of this disk. Required if no value is supplied for `media_link`. Changing this forces a new resource to be created.
- `media_link` - (Optional) The location of the blob in storage where the VHD of this disk will be created. The storage account where must be associated with the subscription. Changing this forces a new resource to be created.
- `source_media_link` - (Optional) The location of a blob in storage where a VHD file is located that is imported and registered as a disk. If a value is supplied, `media_link` will not be used.

- `virtual_machine` - (Required) The name of the virtual machine the disk will be attached to.

Attributes Reference

The following attributes are exported:

- `id` - The security group ID.
- `name` - The name of the disk.
- `label` - The identifier for the disk.
- `media_link` - The location of the blob in storage where the VHD of this disk is created.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_dns_server

Creates a new DNS server definition to be used internally in Azure.

Example Usage

```
resource "azure_dns_server" "google-dns" {
  name        = "google"
  dns_address = "8.8.8.8"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the DNS server reference. Changing this forces a new resource to be created.
- `dns_address` - (Required) The IP address of the DNS server.

Attributes Reference

The following attributes are exported:

- `id` - The DNS server definition ID. Coincides with the given `name`.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_hosted_service

Creates a new hosted service on Azure with its own .cloudapp.net domain.

Example Usage

```
resource "azure_hosted_service" "terraform-service" {
  name                = "terraform-service"
  location            = "North Europe"
  ephemeral_contents = false
  description         = "Hosted service created by Terraform."
  label               = "tf-hs-01"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the hosted service. Must be unique on Azure.
- `location` - (Required) The location where the hosted service should be created. For a list of all Azure locations, please consult this link (https://azure.microsoft.com/en-us/regions/).
- `ephemeral_contents` - (Required) A boolean value (true|false), specifying whether all the resources present in the hosted hosted service should be destroyed following the hosted service's destruction.
- `reverse_dns_fqdn` - (Optional) The reverse of the fully qualified domain name for the hosted service.
- `label` - (Optional) A label to be used for tracking purposes. Must be non-void. Defaults to `Made by Terraform.`.
- `description` - (Optional) A description for the hosted service.

Attributes Reference

The following attributes are exported:

- `id` - The hosted service ID. Coincides with the given `name`.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_instance

Creates a hosted service, role and deployment and then creates a virtual machine in the deployment based on the specified configuration.

Example Usage

```
resource "azure_hosted_service" "terraform-service" {
  name               = "terraform-service"
  location           = "North Europe"
  ephemeral_contents = false
  description        = "Hosted service created by Terraform."
  label              = "tf-hs-01"
}

resource "azure_instance" "web" {
  name                 = "terraform-test"
  hosted_service_name  = "${azure_hosted_service.terraform-service.name}"
  image                = "Ubuntu Server 14.04 LTS"
  size                 = "Basic_A1"
  storage_service_name = "yourstorage"
  location             = "West US"
  username             = "terraform"
  password             = "Pass!admin123"
  domain_name          = "contoso.com"
  domain_ou            = "OU=Servers,DC=contoso.com,DC=Contoso,DC=com"
  domain_username      = "Administrator"
  domain_password      = "Pa$$word123"

  endpoint {
    name         = "SSH"
    protocol     = "tcp"
    public_port  = 22
    private_port = 22
  }
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the instance. Changing this forces a new resource to be created.

- `hosted_service_name` - (Optional) The name of the hosted service the instance should be deployed under. If not provided; it will default to the value of `name`. Changes to this parameter forces the creation of a new resource.
- `description` - (Optional) The description for the associated hosted service. Changing this forces a new resource to be created (defaults to the instance name).
- `image` - (Required) The name of an existing VM or OS image to use for this instance. Changing this forces a new resource to be created.
- `size` - (Required) The size of the instance.
- `subnet` - (Optional) The name of the subnet to connect this instance to. If a value is supplied `virtual_network` is required. Changing this forces a new resource to be created.
- `virtual_network` - (Optional) The name of the virtual network the `subnet` belongs to. If a value is supplied `subnet` is required. Changing this forces a new resource to be created.
- `storage_service_name` - (Optional) The name of an existing storage account within the subscription which will be used to store the VHDs of this instance. Changing this forces a new resource to be created. **A Storage Service is required if you are using a Platform Image**
- `reverse_dns` - (Optional) The DNS address to which the IP address of the hosted service resolves when queried using a reverse DNS query. Changing this forces a new resource to be created.
- `location` - (Required) The location/region where the cloud service is created. Changing this forces a new resource to be created.
- `automatic_updates` - (Optional) If true this will enable automatic updates. This attribute is only used when creating a Windows instance. Changing this forces a new resource to be created (defaults false)
- `time_zone` - (Optional) The appropriate time zone for this instance in the format 'America/Los_Angeles'. This attribute is only used when creating a Windows instance. Changing this forces a new resource to be created (defaults false)
- `username` - (Required) The username of a new user that will be created while creating the instance. Changing this forces a new resource to be created.
- `password` - (Optional) The password of the new user that will be created while creating the instance. Required when creating a Windows instance or when not supplying an `ssh_key_thumbprint` while creating a Linux instance. Changing this forces a new resource to be created.
- `ssh_key_thumbprint` - (Optional) The SSH thumbprint of an existing SSH key within the subscription. This attribute is only used when creating a Linux instance. Changing this forces a new resource to be created.
- `security_group` - (Optional) The Network Security Group to associate with this instance.
- `endpoint` - (Optional) Can be specified multiple times to define multiple endpoints. Each `endpoint` block supports fields documented below.
- `domain_name` - (Optional) The name of an Active Directory domain to join.
- `domain_ou` - (Optional) Specifies the LDAP Organizational Unit to place the instance in.
- `domain_username` - (Optional) The username of an account with permission to join the instance to the domain. Required if a domain_name is specified.
- `domain_password` - (Optional) The password for the domain_username account specified above.

- `custom_data` - (Optional) The custom data to provide when launching the instance.

The `endpoint` block supports:

- `name` - (Required) The name of the external endpoint.
- `protocol` - (Optional) The transport protocol for the endpoint. Valid options are: `tcp` and `udp` (defaults `tcp`)
- `public_port` - (Required) The external port to use for the endpoint.
- `private_port` - (Required) The private port on which the instance is listening.

Attributes Reference

The following attributes are exported:

- `id` - The instance ID.
- `description` - The description for the associated hosted service.
- `subnet` - The subnet the instance is connected to.
- `endpoint` - The complete set of configured endpoints.
- `security_group` - The associated Network Security Group.
- `ip_address` - The private IP address assigned to the instance.
- `vip_address` - The public IP address assigned to the instance.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_local_network_connection

Defines a new connection to a remote network through a VPN tunnel.

Example Usage

```
resource "azure_local_network_connection" "localnet" {
  name                  = "terraform-local-network-connection"
  vpn_gateway_address    = "45.12.189.2"
  address_space_prefixes = ["10.10.10.0/24", "10.10.11.0/24"]
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name by which this local network connection will be referenced by. Changing this forces a new resource to be created.
- `vpn_gateway_address` - (Required) The public IPv4 of the VPN endpoint.
- `address_space_prefixes` - (Required) List of address spaces accessible through the VPN connection. The elements are in the CIDR format.

Attributes Reference

The following attributes are exported:

- `id` - The local network connection ID.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_security_group

Creates a new network security group within the context of the specified subscription.

Example Usage

```
resource "azure_security_group" "web" {
  name     = "webservers"
  location = "West US"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the security group. Changing this forces a new resource to be created.
- `label` - (Optional) The identifier for the security group. The label can be up to 1024 characters long. Changing this forces a new resource to be created (defaults to the security group name)
- `location` - (Required) The location/region where the security group is created. Changing this forces a new resource to be created.

Attributes Reference

The following attributes are exported:

- `id` - The security group ID.
- `label` - The identifier for the security group.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_security_group_rule

Creates a new network Security Group Rule to be associated with a number of given Security Groups.

NOTE on Security Group Rules: for usability purposes; Terraform allows the addition of a single Security Group Rule to multiple Security Groups, despite it having to define each rule individually per Security Group on Azure. As a result; in the event that one of the Rules on one of the Groups is modified by external factors, Terraform cannot reason as to whether or not that change should be propagated to the others; let alone choose one changed Rule configuration over another in case of a conflic. As such; `terraform refresh` only checks that the rule is still defined for each of the specified `security_group_names` ; ignoring the actual parameters of the Rule and **not** updating the state with regards to them.

Example Usage

```
resource "azure_security_group" "web" {
  # ...
}

resource "azure_security_group" "apps" {
  # ...
}

resource "azure_security_group_rule" "ssh_access" {
  name                        = "ssh-access-rule"
  security_group_names        = ["${azure_security_group.web.name}", "${azure_security_group.apps.name}"]
  type                        = "Inbound"
  action                      = "Allow"
  priority                    = 200
  source_address_prefix       = "100.0.0.0/32"
  source_port_range           = "*"
  destination_address_prefix = "10.0.0.0/32"
  destination_port_range      = "22"
  protocol                    = "TCP"
}
```

Argument Reference

The following arguments are supported: * `name` - (Required) The name of the security group rule.

- `security_group_names` - (Required) A list of the names of the security groups the rule should be applied to. Changing this list forces the creation of a new resource.

- `type` - (Required) The type of the security rule. Valid options are: `Inbound` and `Outbound`.
- `priority` - (Required) The priority of the network security rule. Rules with lower priority are evaluated first. This value can be between 100 and 4096.
- `action` - (Optional) The action that is performed when the security rule is matched. Valid options are: `Allow` and `Deny`.
- `source_address_prefix` - (Required) The address prefix of packet sources that that should be subjected to the rule. An asterisk (*) can also be used to match all source IPs.
- `source_port_range` - (Required) The source port or range. This value can be between 0 and 65535. An asterisk (*) can also be used to match all ports.
- `destination_address_prefix` - (Required) The address prefix of packet destinations that should be subjected to the rule. An asterisk (*) can also be used to match all destination IPs.
- `destination_port_range` - (Required) The destination port or range. This value can be between 0 and 65535. An asterisk (*) can also be used to match all ports.
- `protocol` - (Optional) The protocol of the security rule. Valid options are: `TCP`, `UDP` and `*`.

The following attributes are exported:

- `id` - The security group rule ID. Coincides with its given `name`.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_sql_database_server

Defines a new Firewall Rule to be applied across the given Database Servers.

Example Usage

```
resource "azure_sql_database_server" "sql-serv1" {
  # ...
}

resource "azure_sql_database_server" "sql-serv2" {
  # ...
}

resource "azure_sql_database_server_firewall_rule" "constraint" {
  name     = "terraform-testing-rule"
  start_ip = "154.0.0.0"
  end_ip   = "154.0.0.255"

  database_server_names = [
    "${azure_sql_database_server.sql-serv1.name}",
    "${azure_sql_database_server.sql-serv2.name}",
  ]
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the rule. Changing forces the creation of a new resource.
- `start_ip` - (Required) The IPv4 which will represent the lower bound of the rule's application IP's. Traffic to/from IP's greater than or equal to this one up to the `end_ip` will be permitted.
- `end_ip` - (Required) The IPv4 which will represent the upper bound of the rule's application IP's. Traffic to/from IP's lesser that or equal to this one all the way down to the `start_ip` will be permitted.
- `database_server_names` - (Required) The set of names of the Azure SQL Database servers the rule should be enforced on.

Attributes Reference

The following attributes are exported:

- `id` - The database server ID. Coincides with the given `name`.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_sql_database_server

Allocates a new SQL Database Server on Azure.

Example Usage

```
resource "azure_sql_database_server" "sql-serv" {
  name     = "<computed>"
  location = "West US"
  username = "SuperUser"
  password = "SuperSEKR3T"
  version  = "2.0"
  url      = "<computed>"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Computed) The name of the database server. It is determined upon creation as it is randomly-generated per server.
- `location` - (Required) The location where the database server should be created. For a list of all Azure locations, please consult this link (https://azure.microsoft.com/en-us/regions/).
- `username` - (Required) The username for the administrator of the database server.
- `password` - (Required) The password for the administrator of the database server.
- `version` - (Optional) The version of the database server to be used. Can be any one of `2.0` or `12.0`.
- `url` - (Computed) The fully qualified domain name of the database server. Will be of the form `<name>.database.windows.net`.

Attributes Reference

The following attributes are exported:

- `id` - The database server ID. Coincides with the randomly-generated `name`.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_sql_database_service

Creates a new SQL database service on an Azure database server.

Example Usage

```
resource "azure_sql_database_service" "sql-server" {
  name                = "terraform-testing-db-renamed"
  database_server_name = "flibberflabber"
  edition             = "Standard"
  collation           = "SQL_Latin1_General_CP1_CI_AS"
  max_size_bytes      = "5368709120"
  service_level_id    = "f1173c43-91bd-4aaa-973c-54e79e15235b"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the database service.
- `database_server_name` - (Required) The name of the database server this service should run on. Changes here force the creation of a new resource.
- `edition` - (Optional) The edition of the database service. For more information on each variant, please view this (https://msdn.microsoft.com/library/azure/dn741340.aspx) link.
- `collation` - (Optional) The collation to be used within the database service. Defaults to the standard Latin charset.
- `max_size_bytes` - (Optional) The maximum size in bytes the database service should be allowed to expand to. Range depends on the database `edition` selected above.
- `service_level_id` - (Optional) The ID corresponding to the service level per edition. Please refer to this (https://msdn.microsoft.com/en-us/library/azure/dn505701.aspx) link for more details.

Attributes Reference

The following attributes are exported:

- `id` - The database service ID. Coincides with the given `name` .

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_storage_blob

Creates a new storage blob within a given storage container on Azure.

Example Usage

```
resource "azure_storage_blob" "foo" {
  name                   = "tftesting-blob"
  storage_service_name   = "tfstorserv"
  storage_container_name = "terraform-storage-container"
  type                   = "PageBlob"
  size                   = 1024
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the storage blob. Must be unique within the storage service the blob is located.
- `storage_service_name` - (Required) The name of the storage service within which the storage container in which the blob will be created resides.
- `storage_container_name` - (Required) The name of the storage container in which this blob should be created. Must be located on the storage service given with `storage_service_name`.
- `type` - (Required) The type of the storage blob to be created. One of either `BlockBlob` or `PageBlob`.
- `size` - (Optional) Used only for `PageBlob`'s to specify the size in bytes of the blob to be created. Must be a multiple of 512. Defaults to 0.

Attributes Reference

The following attributes are exported:

- `id` - The storage blob ID. Coincides with the given `name`.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_storage_container

Creates a new storage container within a given storage service on Azure.

Example Usage

```
resource "azure_storage_container" "stor-cont" {
  name                  = "terraform-storage-container"
  container_access_type = "blob"
  storage_service_name  = "tfstorserv"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the storage container. Must be unique within the storage service the container is located.
- `storage_service_name` - (Required) The name of the storage service within which the storage container should be created.
- `container_access_type` - (Required) The 'interface' for access the container provides. Can be either `blob`, `container` or ``.
- `properties` - (Optional) Key-value definition of additional properties associated to the storage service.

Attributes Reference

The following attributes are exported:

- `id` - The storage container ID. Coincides with the given `name`.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_storage_queue

Creates a new storage queue within a given storage service on Azure.

Example Usage

```
resource "azure_storage_queue" "stor-queue" {
  name                 = "terraform-storage-queue"
  storage_service_name = "tfstorserv"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the storage queue. Must be unique within the storage service the queue is located.
- `storage_service_name` - (Required) The name of the storage service within which the storage queue should be created.

Attributes Reference

The following attributes are exported:

- `id` - The storage queue ID. Coincides with the given `name`.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_storage_service

Creates a new storage service on Azure in which storage containers may be created.

Example Usage

```
resource "azure_storage_service" "tfstor" {
  name         = "tfstor"
  location     = "West US"
  description  = "Made by Terraform."
  account_type = "Standard_LRS"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the storage service. Must be between 4 and 24 lowercase-only characters or digits. Must be unique on Azure.
- `location` - (Required) The location where the storage service should be created. For a list of all Azure locations, please consult this link (https://azure.microsoft.com/en-us/regions/).
- `account_type` - (Required) The type of storage account to be created. Available options include `Standard_LRS`, `Standard_ZRS`, `Standard_GRS`, `Standard_RAGRS` and `Premium_LRS`. To learn more about the differences of each storage account type, please consult this link (http://blogs.msdn.com/b/windowsazurestorage/archive/2013/12/11/introducing-read-access-geo-replicated-storage-ra-grs-for-windows-azure-storage.aspx).
- `affinity_group` - (Optional) The affinity group the storage service should belong to.
- `properties` - (Optional) Key-value definition of additional properties associated to the storage service. For additional information on what these properties do, please consult this link (https://msdn.microsoft.com/en-us/library/azure/hh452235.aspx).
- `label` - (Optional) A label to be used for tracking purposes. Must be non-void. Defaults to `Made by Terraform.`.
- `description` - (Optional) A description for the storage service.

Attributes Reference

The following attributes are exported:

- `id` - The storage service ID. Coincides with the given `name`.

NOTE: The Azure Service Management Provider has been superseded by the Azure Resource Manager Provider (/docs/providers/azurerm) and is no longer being actively developed by HashiCorp employees. It continues to be supported by the community. We recommend using the Azure Resource Manager based Microsoft Azure Provider (/docs/providers/azurerm) if possible.

azure_virtual_network

Creates a new virtual network including any configured subnets. Each subnet can optionally be configured with a security group to be associated with the subnet.

Example Usage

```
resource "azure_virtual_network" "default" {
  name          = "test-network"
  address_space = ["10.1.2.0/24"]
  location      = "West US"

  subnet {
    name           = "subnet1"
    address_prefix = "10.1.2.0/25"
  }
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the virtual network. Changing this forces a new resource to be created.
- `address_space` - (Required) The address space that is used the virtual network. You can supply more than one address space. Changing this forces a new resource to be created.
- `location` - (Required) The location/region where the virtual network is created. Changing this forces a new resource to be created.
- `dns_servers` - (Optional) List of names of DNS servers previously registered on Azure.
- `subnet` - (Required) Can be specified multiple times to define multiple subnets. Each `subnet` block supports fields documented below.

The `subnet` block supports:

- `name` - (Required) The name of the subnet.
- `address_prefix` - (Required) The address prefix to use for the subnet.
- `security_group` - (Optional) The Network Security Group to associate with the subnet.

Attributes Reference

The following attributes are exported:

- `id` - The virtual NetworkConfiguration ID.