



Infoblox IPAM Driver for Terraform

08/23/2020

Contents

- Overview of Infoblox IPAM Plugin for Terraform..... 4**
- Installing Infoblox IPAM Plugin for Terraform 5**
 - System Requirements 5
 - Assumptions 5
 - Downloading vNIOS 5
 - Setting up vNIOS..... 5
- Configuring Infoblox as a Provider 7**
 - Configuring the Provider Authentication 7
 - Static Credentials 7
 - Example: 7
 - Environment Variables 7
 - Example: 8
 - Supported Functions and Limitations of the Provider 8
- Building the Infoblox IPAM Plugin for Terraform 9**
- Infoblox IPAM Plugin for Terraform Resources 10**
 - Limitations 10
 - Creating Network View in NIOS Appliance..... 10
 - Creating a Network View in NIOS Appliance Example..... 10
 - Creating a Resource Network in a NIOS Appliance 11
 - Example 11
 - Allocating an IP Address 12
 - Example 13
 - Allocating DNS Records..... 13
 - infoblox_a_record..... 14
 - infoblox_ptr_record..... 14
 - infoblox_cname_record 15
- Infoblox IPAM Plugin for Terraform Data Sources 17**
 - Fetching Data for a Network Object 17
 - Example 17
- Integrating Infoblox IPAM Plugin for Terraform in Cloud Environment 18**
 - Integrating vSphere 18
 - Prerequisites to Integrate vSphere with Terraform Infoblox Provider 18
 - Integrating vSphere with Terraform Infoblox Provider 18

Integrating Azure	18
Prerequisites to Integrate Azure with Terraform Infoblox Provider	18
Integrating VMware/Azure with Infoblox Provider	19
Viewing the Integration In NIOS	19
Related Documentation.....	23
Technical Support.....	24

Overview of Infoblox IPAM Plugin for Terraform

Terraform is an open-source infrastructure-as-code tool by HashiCorp. It allows you to define and create an execution plan to provision the infrastructure in a high-level configuration language. It is a tool that you can use to manage infrastructure across multiple clouds and platforms.

The **Terraform Provider for Infoblox is an IPAM plugin** (henceforth referred to as Infoblox IPAM Plugin for Terraform) that interfaces with Infoblox VNIOS to provide IP Address Management and DNS Services. You can use the **Infoblox IPAM Plugin for Terraform** to automate the steps to provision the IP addresses and DNS records for network devices and interface in your infrastructure with VNIOS

With this approach, Infoblox IPAM Driver for Terraform provides solutions to the following use cases:

- *Creation of Network View in NIOS appliance*
- *Creation of Network in NIOS appliance*
- *Allocation and Deallocation of IP Address from a Network*
- *Allocation of DNS Records (A, CNAME, Host, and PTR records)*

Currently, you can integrate the **Infoblox IPAM Plugin for Terraform** with VMware and Azure Cloud environments.

Installing Infoblox IPAM Plugin for Terraform

- To use the plugin, you need access to the physical or virtual Infoblox DDI product, NIOS or vNIOS. For evaluation purposes, you can download a virtual version of the product from the [INFOBLOX DOWNLOAD CENTER](#). If you are an existing Infoblox customer, you can download it from the Support site. For information about downloading and setting up vNIOS, see [Downloading vNIOS](#) and [Setting Up vNIOS](#).
- You must install Terraform v0.11.13 or greater on your system, you can download Terraform from [Download Terraform](#). For more information on installing Terraform see, <https://learn.hashicorp.com/terraform/getting-started/install.html>.

System Requirements

System requirements for the Infoblox IPAM Driver for Terraform version 1.1.0 are as follows:

- Terraform 0.11.13 or greater
- Go version 1.12.7 (The client version is used to build the provider plugin)
- Infoblox IPAM Plugin for Terraform is certified from NIOS 8.5.0 onwards

Assumptions

It is assumed that you have the vNIOS appliance already up and running and the necessary licenses are already set. If you haven't, please follow the links below to download and setup vNIOS.

- [Downloading vNIOS](#)
- [Setting up vNIOS](#)

Downloading vNIOS

vNIOS is the Infoblox virtual appliance that you can download from the Infoblox Download Center.

To download vNIOS, complete the following:

1. Point your browser to <https://www.infoblox.com/infoblox-download-center>.
2. Navigate to the Infoblox DDI (DNS, DHCP, IPAM) section.
3. Click **Try it Now** to download the Infoblox DDI product.
4. When the registration is complete, you will receive an email with the link that takes you to the Product Evaluation Portal. In the Product Evaluation Portal, under the **Required Downloads** section, download **Infoblox DDI for VMware**. In the Product Evaluation Portal, you can find download links as well as instructional videos to set up vNIOS.

Note

It is strongly recommended that you download the VMware version of the product, as VMware is the platform on which the videos are based.

5. After the download is complete, install vNIOS

Setting up vNIOS

After you download the vNIOS, complete the following to set up the appliance:

- Add the following extensible attributes in NIOS if the Cloud Network Automation license is not installed on NIOS.
 - **VM Name:** String Type
 - **VM ID:** String Type
 - **Tenant ID:** String Type

- **CMP Type:** String Type
 - **Cloud API Owned:** List Type (Values True, False)
 - **Network Name:** String Type
- Add an Authoritative and Reverse mapping zone manually in NIOS, which is specified in the **Infoblox.tf** file.
- For example: If you are creating a network `cidr="10.10.20.0/24"`, then the corresponding reverse mapping zone should be added in NIOS.

Configuring Infoblox as a Provider

Terraform relies on Infoblox provider to interact with Infoblox Grid objects. You must configure Infoblox as a provider before you can perform the operations described in this document. This topic includes the following sections:

- [Configuring the Provider Authentication](#)
- [Supported Functions and Limitations of the Provider](#)

Configuring the Provider Authentication

You must configure the provider with correct credentials before you can use it to interact with Infoblox grids. The supported methods for configuring the credentials required for authentication are as follows:

- [Static Credentials](#)
- [Environment Variables](#)

Note

The provider is designed considering the cloud network automation aspects of NIOS. If you do not have a cloud license installed in NIOS, add the required extensible attributes manually.

In the Grid Manager, go to **Administration** tab -> **Extensible Attributes** tab, and then add the following extensible attributes:

- **VM Name** as a string
- **VM ID** as a string
- **Cloud API Owned** as a list with values: **True** and **False**
- **CMP Type** as a string
- **Tenant ID** as a string
- **Network Name** as a string

For detailed instructions, refer to the [NIOS online documentation](#).

Static Credentials

You can provide static credentials by adding the specified attributes inline in the `provider` block in the Terraform configuration file.

Example:

```
provider "infoblox"
{
  version="~> 1.0"
  username="infoblox_user"
  password="infoblox"
  server="10.0.0.1"
}
```

Environment Variables

You can configure your credentials using the `INFOBLOX_USERNAME`, `INFOBLOX_PASSWORD`, and `INFOBLOX_SERVER` environmental variables, which correspond to your user name, password, and server respectively.

Example:

```
$ export INFOBLOX_USERNAME="infoblox_user"
$ export INFOBLOX_PASSWORD="infoblox"
$ export INFOBLOX_SERVER="10.0.0.1"
```

Supported Functions and Limitations of the Provider

The supported functions and limitations of the Infoblox provider are as follows:

- The provider supports only create, read, and delete operations for networks/CIDRs. The update operation is not supported.
- The provider supports create, read, and delete operations on A, PTR, and CNAME Records. The update operation is not supported.
- The provider allows only creation of network views. Deletion of network views is not supported.
- If you are using the Infoblox provider with other providers to allocate IP addresses to virtual machines, use the two resource blocks `ip_allocation` and `ip_association` to perform the operation. See the [Examples](#) for using the Infoblox provider.
- If you are not using other providers with the Infoblox provider, then use only the `ip_allocation` block to allocate IP addresses. The `ip_allocation` block supports all CRUD operations.
- Use the `ip_allocation` block to create either a reservation, a fixed address, or a host record. To create a host record, see the [ip_allocation](#) resource documentation for detailed instructions.
- If you are not using other providers with the Infoblox provider to deploy virtual machines and allocate IP addresses from NIOS, then ignore the `ip_association` block. The `ip_association` block is used to update the properties of virtual machines.

Building the Infoblox IPAM Plugin for Terraform

The **Infoblox IPAM Plugin for Terraform** helps to integrate Terraform environment with NIOS appliances. To use the **Infoblox IPAM Plugin for Terraform**, build the provider and install it as a plugin.

To develop the provider and ensure it is set up correctly, complete the following prerequisites:

1. Install Go version 1.12 or later on your machine.
2. Setup the GOPATH and GOROOT.
3. Run the `mkdir -p $GOPATH/src/github.com/infobloxopen` command to create an infobloxopen directory.
4. Run the `git clone` command from the <https://github.com/infobloxopen/terraform-provider-infoblox.git> to clone the repository.
5. Fetch the Infoblox Terraform Provider from GitHub. Once you fetch the Provider, the **infoblox.tf** and **vm.tf** files will be available in the VMware and Azure folder. See the following:
 - **infoblox.tf**: This folder contains the Infoblox resource details, these resource details contains the credentials of the Infoblox Server and resource records like A, PTR, and CNAME. The main purpose of the Infoblox Terraform Provider is to allocate the IP address to the virtual machine.
 - **vm.tf**: This folder contains the Azure or VMware resource details. Resource details for Azure, like Tenant ID, and VMware resources, like Server details, the hardware like Ubuntu where the resource details will be deployed. The **vm.tf** is used to deploy a virtual machine. All the properties of the virtual machine are specified in this directory. In this directory, the field IPV4 specifies the to access the IP address from Infoblox.tf to make the records visible in NIOS once the IP address is allocated to the virtual machine.

To develop the **Infoblox IPAM Plugin for Terraform**, complete the following:

1. Run the `export GOFLAGS=-mod=vendor` command to set the environment variables.
2. Run the `make build` command to compile and build the provider.
3. Run the `cp -r terraform-provider-infoblox ~/.terraform.d/plugins/` path command to copy the Infoblox Terraform binary manually and build the plugin.
4. Run the `terraform init` command in the directory where the **.tf** file is created to initialize the plugin.

Infoblox IPAM Plugin for Terraform Resources

You can use the Infoblox IPAM Plugin for Terraform for DNS purposes. However, a zone must already exist, as the plugin does not support the creation of zones.

Limitations

- After the allocation of DNS records, you cannot update fixed address, Host, A, PTR, and CNAME record, as the provider integrates different resources and once you create the virtual machine, you must not change the virtual machine name
- The terraform destroy command destroys all the resources that are created, however, destroying a particular record using terraform destroy command is currently not supported. This is the current behavior of Terraform.
- When you upgrade the NIOS appliance from an older version to the new version, terraform destroy command is not recommended.
- To update the Azure instance properties in NIOS, you need to run the terraform apply command again. For example, to update the MAC address from an Azure instance to the Host record in NIOS, run the terraform apply command until the MAC address is set to 00:00:00:00:00:00.
- Infoblox IPAM Plugin for Terraform currently supports only IPv4 and IPv6.

Creating Network View in NIOS Appliance

You can create a network view in a NIOS appliance using `infoblox_network_view` resource block. All the parameters used to create a network view are optional. If you do not create a network view, it uses the optional parameters and will reside under the default network view.

The following table describes the mandatory and optional parameters used to create a network view.

Sr. No	Parameter	Mandatory/Optional	Description
1	<code>network_view_name</code>	Optional	Specifies the desired name of the network view, as shown in NIOS appliance.
2	<code>tenant_id</code>	Optional	Specifies the unique identifier of the tenant in the cloud. The tenant ID is the name of the cloud provider you would like to integrate. However, if there is no cloud provider, you can specify any dummy value. The tenant ID differentiates the cloud providers in NIOS. For more information, see Viewing All Tenants .

Creating a Network View in NIOS Appliance Example

To create a network view in NIOS appliance using the `infoblox_network_view` resource block, see the following example:

```
resource "infoblox_network_view" "demo_network_view"
{
network_view_name="demo1"
tenant_id="test"
}
```

Creating a Resource Network in a NIOS Appliance

You can create a resource network in a NIOS appliance using the `infoblox_network` resource block.

The following table describes the mandatory and optional parameters used to create a network.

Sr No	Parameter	Mandatory/Optional	Description
1	<code>network_view_name</code>	Optional	Specifies the network view name available in the NIOS appliance. <div style="border: 1px solid #f0e68c; padding: 5px; margin-top: 10px;"> <p>⚠ Note If you do not specify the <code>network_view_name</code> parameter, the network is created in the default network view.</p> </div>
2	<code>network_name</code>	Mandatory	Specifies the name of your network block.
3	<code>cidr</code>	Mandatory	Specifies the network block in <code>cidr</code> format.
4	<code>tenant_id</code>	Mandatory	Specifies the unique identifier of the tenant in the cloud. The tenant ID is the name of the cloud provider you would like to integrate. However, if there is no cloud provider, you can specify any dummy value. The tenant ID differentiates the cloud providers in NIOS. For more information, see Viewing All Tenants .
5	<code>reserve_ip</code>	Optional	Specifies the number of IP addresses to reserve.
6	<code>gateway</code>	Optional	Specifies the gateway IP address of your network block. By default, the first IPv4 address is set as the gateway address.

⚠ Note

When deploying the Azure instance using the Terraform plugin, a new network will be created. You have to reserve the first four IP's, and since you already reserved one IP by default while creating a new network, you need to set the `reserve_ip` value to three in the `infoblox_network` resource.

Example

To create a resource network in NIOS appliance for a fixed address, host record, and multiple, see the following:

```
resource "infoblox_network" "demo_network"
{
network_view_name="demo1"
network_name="ex1"
cidr="10.10.20.0/24"
tenant_id="test"
```

}

Allocating an IP Address

You can allocate an IP address from a network and for a virtual machine in the NIOS appliance. To create an IP Address for a network, it sends a request to the next available IP address from the network block.

The `infoblox_ip_association` and `infoblox_ip_allocation` resources block together form the IP address allocation. The `infoblox_ip_association` is also used to update the properties of the virtual machine like the `mac_addr` and `vm_id`.

Using the `infoblox_ip_allocation` resources block an IP address is reserved in NIOS appliance. For example, If `ip_addr` parameter is not set then the `infoblox_ip_allocation` resources block reserves the next available IP address from the network. However, to set the fixed IP address you need to specify the `ip_addr` parameter.

If you specify the `dns_view` and `zone` parameters, it creates a host record or it creates a fixed IP address. If you want to use the host record for the DNS or DHCP purposes, you need to set the value of `enable_dns` parameter to `true`.

The following table describes the mandatory and optional parameters used to allocate and associate an IP address.

Sr No	Parameter	Mandatory/Optional	Description
1	<code>network_view_name</code>	Optional	Specifies the network view name available in the NIOS appliance.
2	<code>vm_name</code>	Mandatory	Specifies the name of the virtual machine.
3	<code>cidr</code>	Mandatory	Specifies the address in <code>cidr</code> format. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> Note Ensure the network address is available in NIOS, under the required network view (<code>network_view_name</code>).</p> </div>
4	<code>zone</code>	Optional	Specifies the zone under which the record has to be created.
5	<code>enable_dns</code>	Optional	Specifies the flag that defines if the host record is used for DNS or IPAM purposes.
6	<code>dns_view</code>	Optional	Specifies the DNS view under which the zone is created.
7	<code>ip_addr</code>	Optional	Specifies the IP address of the instance in the cloud. For static allocation, you need to set the field with a valid IP address.
8	<code>mac_addr</code>	Optional	Specifies the MAC address of the instance in the cloud.
9	<code>vm_id</code>	Optional	Specifies the Instance ID.

Sr No	Parameter	Mandatory/Optional	Description
10	tenant_id	Mandatory	<p>Specifies the Unique Identifier of the tenant in the cloud.</p> <p>The tenant ID is the name of the cloud provider you would like to integrate. However, if there is no cloud provider, you can specify any dummy value.</p> <p>The tenant ID differentiates the cloud providers in NIOS. For more information, see Viewing All Tenants.</p>

Note

To create a host record, you need to set the `enable_dns` and `dns_view` parameters for `infoblox_ip_allocation` and `infoblox_ip_association` resources blocks.

Example

The following examples form the IP address allocation:

```
resource "infoblox_ip_allocation" "demo_allocation"
{
network_view_name="demo1"
vm_name="vmname1"
cidr="10.0.0.0./24"
tenant_id="test"
}

resource "infoblox_ip_association" "demo_associate"
{
network_view_name="demo1"
vm_name="vmname1"
cidr="10.0.0.0./24"
mac_addr =11:11:11:11:11:11:11
ip_addr="10.0.0.2"
vm_id =testtenant_id="test"
}
```

References

- [Integrating Azure with Infoblox Provider](#)
- [Integrating vSphere with Infoblox Provider](#)

Allocating DNS Records

For the allocation of DNS records like A Record, PTR Record, and CNAME Record, to create these records, it is mandatory to configure authoritative and reverse mapping zone manually in NIOS. For more information, see [Configuring DNS Zones](#).

infoblox_a_record

For more information, see [A Record](#).

The following table describes the mandatory and optional parameters used for an A Record.

Sr No.	Parameter	Mandatory/Optional	Description
1	network_view_name	Optional	Specifies the network view name available in the NIOS appliance.
2	vm_name	Mandatory	Specifies the name of the virtual machine.
3	cidr	Mandatory	Specifies the address in cidr format.
4	zone	Mandatory	Specifies the zone under which the record has to be created.
5	dns_view	Mandatory	Specifies the DNS view under which the zone that has to be created.
6	ip_addr	Optional	Specifies the IP address of the instance in the cloud. For static allocation, set this field with a valid IP.
7	vm_id	Optional	Specifies the instance ID.
8	tenant_id	Mandatory	Specifies the unique identifier of a tenant in the cloud. The tenant ID is the name of the cloud provider you would like to integrate. However, if there is no cloud provider, you can specify any dummy value. The tenant ID differentiates the cloud providers in NIOS. For more information, see Viewing All Tenants .

Example

```
resource "infoblox_a_record" "demo_record"
{
network_view_name="demo1"
vm_name="vmname1"
cidr="10.0.0.0/24"
ip_addr="10.0.0.2"
dns_view="default"
zone="aa.com"
tenant_id="test"
}
```

infoblox_ptr_record

For more information, see [PTR Record](#).

The following table describes the mandatory and optional parameters used for a PTR Record.

Sr No.	Parameter	Mandatory/Optional	Description
1	network_view_name	Optional	Specifies the network view name available in the NIOS server.
2	vm_name	Mandatory	Specifies the name of the virtual machine.
3	cidr	Mandatory	Specifies the address in cidr format.
4	zone	Mandatory	Specifies the zone under which the record has to be created.
5	dns_view	Mandatory	Specifies the DNS view under which the zone has been created.
6	ip_addr	Optional	Specifies the IP address of the instance in the cloud. For static allocation, set the field with a valid IP.
7	vm_id	Optional	Specifies the instance ID.
8	tenant_id	Mandatory	Specifies the unique identifier of the tenant in the cloud. The tenant ID is the name of the cloud provider you would like to integrate. However, if there is no cloud provider, you can specify any dummy value. The tenant ID differentiates the cloud providers in NIOS. For more information, see Viewing All Tenants .

Example

```
resource "infoblox_ptr_record" "demo_ptr"
{
network_view_name="demo1"
vm_name="vmname1"
cidr="10.10.20.0/24"
ip_addr="10.0.0.2"
dns_view="default"
zone="aa.com"
tenant_id="test"
}
```

infoblox_cname_record

For more information, see [CNAME Record](#).

The following table describes the mandatory and optional parameters used for a CNAME Record.

Sr No.	Parameter	Mandatory/Optional	Description
1	zone	Mandatory	The zone under which the record has to be created.
2	dns_view	Mandatory	Specifies the DNS view under which the zone has been created.
3	canonical	Mandatory	Specifies the Canonical name for the record.

Sr No.	Parameter	Mandatory/Optional	Description
4	alias	Mandatory	Specifies the alias name for the record.
5	vm_name	Optional	Specifies the name of the virtual machine.
6	vm_id	Optional	Specifies the Instance ID.
7	tenant_id	Mandatory	<p>Specifies the unique identifier of the tenant in the cloud.</p> <p>The tenant ID is the name of the cloud provider you would like to integrate. However, if there is no cloud provider, you can specify any dummy value.</p> <p>The tenant ID differentiates the cloud providers in NIOS. For more information, see Viewing All Tenants.</p>

Example

```
resource "infoblox_cname_record" "demo_cname"
{
  canonical="${infoblox_ip_allocation.demo_allocation.vm_name}"
  zone="aa.com"
  dns_view="default"
  alias="ssas"
  tenant_id="test"
}
```

Infoblox IPAM Plugin for Terraform Data Sources

Data sources allow data to be fetched or computed for use elsewhere in the Terraform configuration. Use of data sources allows a Terraform configuration to build on information defined outside of Terraform, or defined by another separate Terraform configuration.

Fetching Data for a Network Object

You can fetch information such as network ID (`_ref`) and name for a network from NIOS to Terraform using the `infoblox_network` data source block.

The following table describes the parameters that are used to fetch the data:

Sr No.	Parameter	Mandatory/Optional	Description
1	<code>network_view_name</code>	Optional	Specifies the network view name. If a network view is not specified, the provider considers the default network view.
2	<code>network_name</code>	Computed	Specifies the network name that is fetched from the data source.
3	<code>cidr</code>	Mandatory	Specifies the network block in <code>cidr</code> format.
4	<code>tenant_id</code>	Mandatory	Specifies the unique identifier of the tenant in the cloud in which the network exists.

Example

As a prerequisite, you must have a network before you can fetch the data. If a network does not exist, use the `infoblox_network` resource block to create a network as explained in [Creating a Resource Network in a NIOS Appliance](#).

Use the data block to fetch the data:

```
data "infoblox_network" "test"
{
  cidr = infoblox_network.test.cidr #add a CIDR for which data needs to be fetched
  tenant_id = "default"
}
```

Integrating Infoblox IPAM Plugin for Terraform in Cloud Environment

Integrating vSphere

Prerequisites to Integrate vSphere with Terraform Infoblox Provider

Before you integrate vSphere with Terraform Infoblox Provider, ensure you complete the following pre-requisites:

- Export the vSphere Server and vSphere credential:
 - export VSPHERE_USER="\${user_name}"
 - export VSPHERE_PASSWORD="\${password}"
 - export VSPHERE_SERVER="\${server}"
 - export VSPHERE_ALLOW_UNVERIFIED_SSL=true
- Export the Infoblox Server and Infoblox Credentials:
 - export INFOBLOX_PASSWORD="\${password}"
 - export INFOBLOX_SERVER="\${server}"
 - export INFOBLOX_USERNAME="\${username}"

Integrating vSphere with Terraform Infoblox Provider

To integrate vSphere with Terraform Infoblox Provider using a fixed address or using the provider without DNS, complete the following:

1. Select terraform-provider-Infoblox -> **Examples** -> **VMware**.
2. Click the **Fixed Address** folder.
3. Use the [infoblox.tf](#) and [vm.tf](#) template.

To integrate vSphere with Terraform Infoblox Provider using host records or using the provider without DNS, complete the following:

1. Select terraform-provider-Infoblox -> **Examples** -> **VMware**.
2. Click the **Host Record** folder.
3. Use the [infoblox.tf](#) and the [vm.tf](#) template.

Note

Enable DNS flag in the [infoblox.tf](#) template, the `infoblox_ip_allocation` resource block decides if it has to be used or not for DNS purposes.

To integrate vSphere with Terraform Infoblox Provider to provision multiple virtual machines and creating of **A Records** for those multiple virtual machines, complete the following:

1. Select terraform-provider-Infoblox -> **Examples** -> **VMware**.
2. Click the **Multiple** folder.
3. Use the [infoblox.tf](#) and the [vm.tf](#) template.

Integrating Azure

Prerequisites to Integrate Azure with Terraform Infoblox Provider

- Export the Azure Client and Azure credential.
 - export subscription_id="\${subscription_id}"

- `export client_id="${client_id}"`
- `export client_secret="${client_secret}"`
- `export tenant_id="${tenant_id}"`
- Export the Infoblox Server and Infoblox Credentials.
 - `export INFOBLOX_PASSWORD="${password}"`
 - `export INFOBLOX_SERVER="${server}"`
 - `export INFOBLOX_USERNAME="${username}"`

Integrating VMware/Azure with Infoblox Provider

To integrate Azure with Terraform Infoblox Provider, complete the following:

1. Run the `terraform init` command to initialize a working directory containing Terraform configuration files.
2. Run the `terraform plan` command to create an execution plan. This command determines the necessary actions to achieve the desired state specified in the configuration files.
3. Run the `terraform apply` command to apply the changes required to reach the desired state of the configuration, or the pre-determined set of actions generated by a `terraform plan` execution plan.

Note

To update the Mac address in NIOS, you need to run `terraform apply` command again as the properties of the virtual machine are received once you start the virtual machine.

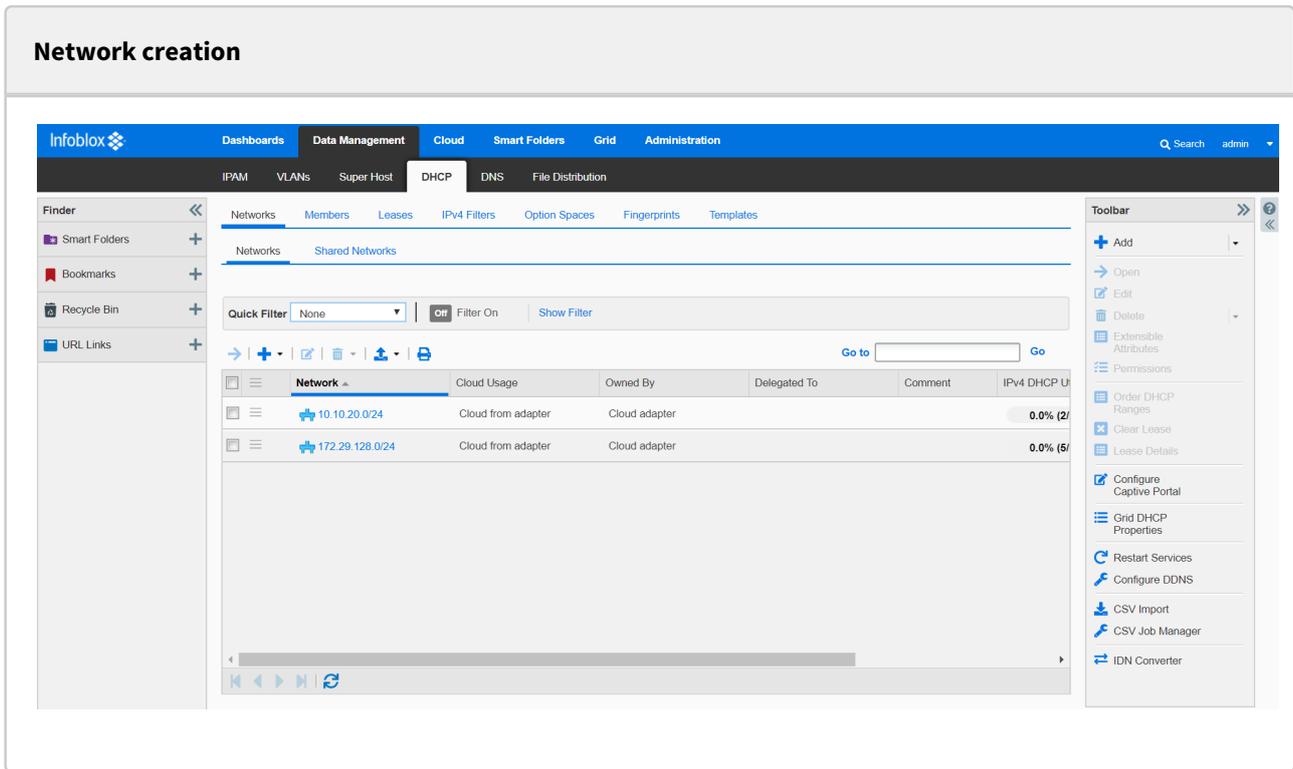
Viewing the Integration In NIOS

Once the `terraform apply` command is executed, you can view the machine instance created in the vSphere client and Azure.com. After the instance is created, you can view the managed records like Host record, A records, and the other records specified in the [infoblox.tf](#) file.

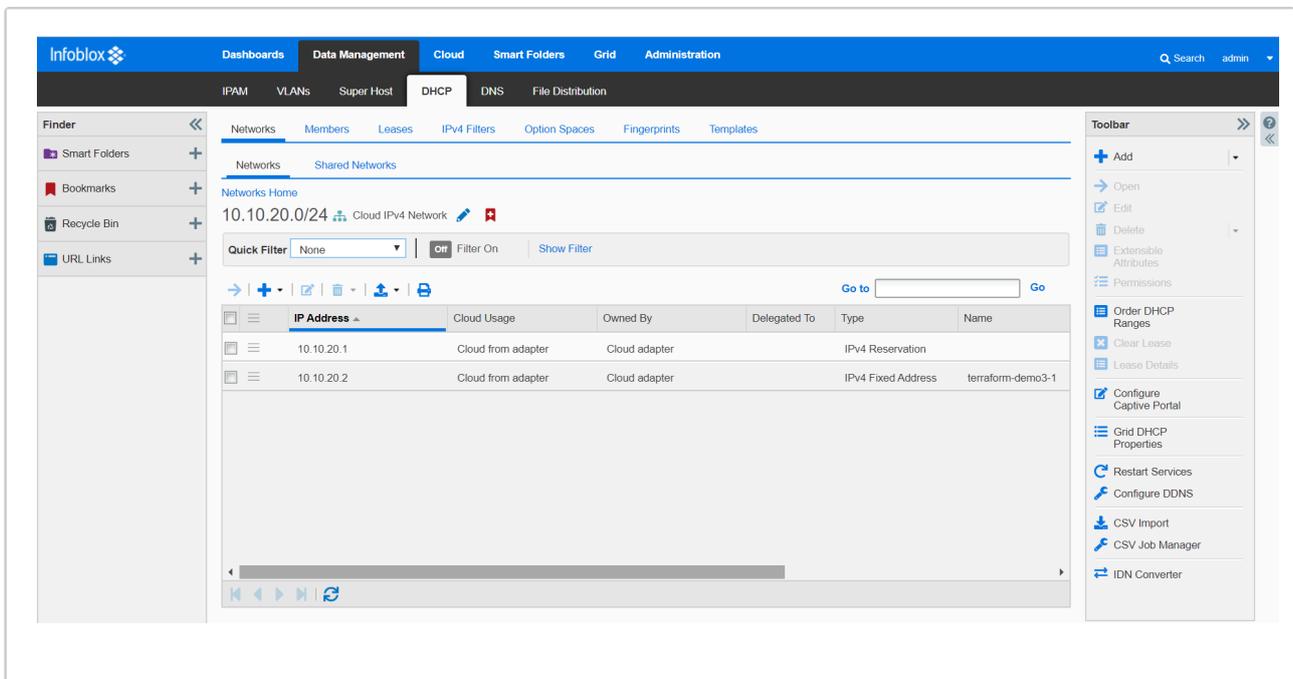
When you run the Infoblox Terraform template for the creation of Azure or VMware, the following actions are performed in NIOS:

- Creation of Network for VMware or Azure.
- Creation of Host records, PTR, and CNAME record for VMware or Azure.
- A record is created after deploying.
- Tenants (Azure and VMware) are displayed.

The following screen displays the network created for VMware and Azure tenants. To view the created network, click **Data Management** -> **DHCP** -> **Network** tab in NIOS.

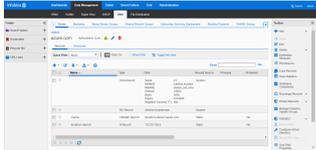


The Fixed address is created when you deploy the Terraform Infoblox Provider. To view the fixed address, you need to click **Data Management** -> **DHCP** -> **Network** tab, and then click the **Network** to view the Fixed Address in NIOS.

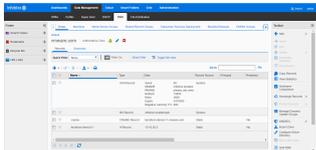


The following screen shows the record that is created in the Authoritative zone for VMware and Azure. To view the created records for Azure, you need to click **Data Management -> DNS -> Zones** tab, and then click **Azure** instance. The **Record** tab displays the records in NIOS.

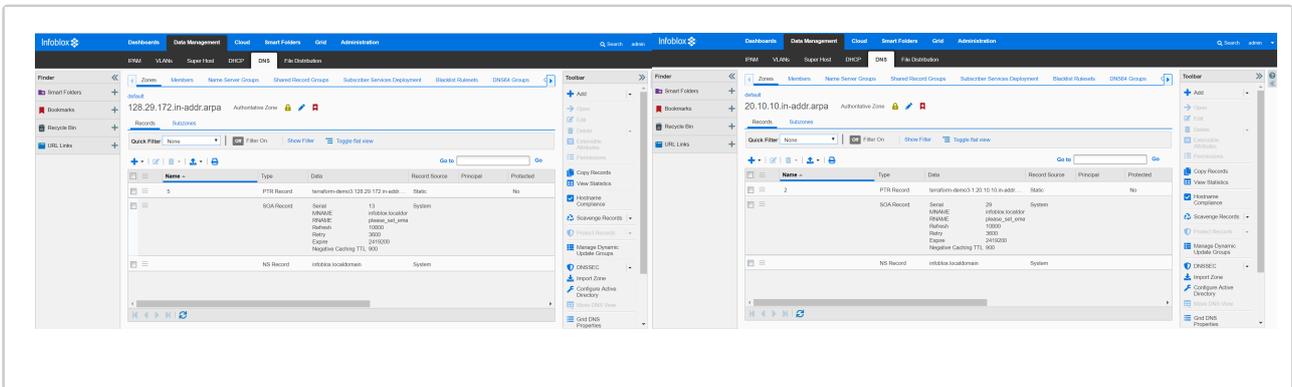
To view the created records for Azure, click **Data Management -> DNS -> Zones** tab and then click the **Azure** instance. The **Record** tab displays the records in NIOS.



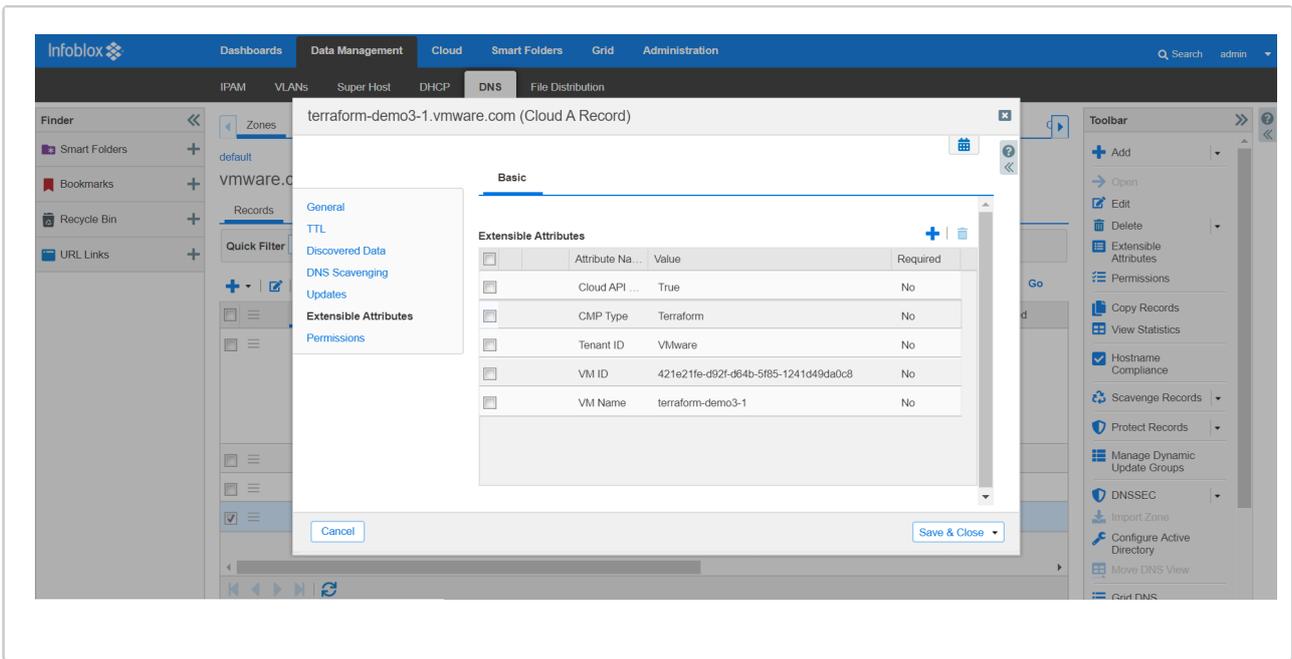
To view the created records for Azure, click **Data Management -> DNS -> Zones** tab, and then click the **VMware** instance. The **Record** tab displays the records in NIOS.



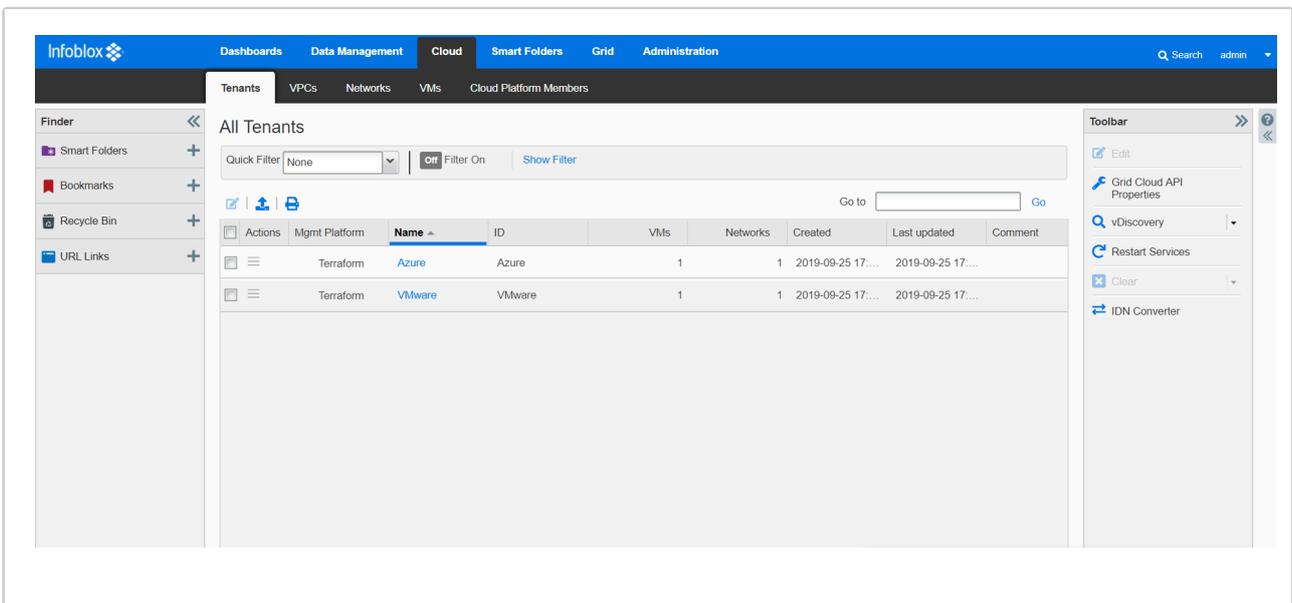
The PTR record is created for Azure and VMware. To view the PTR records, click **Data Management -> DNS -> Zones** tab, and then click the **IP address** assigned for the Azure or VMware instance. The **Records** tab displays the records in NIOS.



You can view the extensible attributes of the records (A, PTR, CNAME, and Host record) which are created for DNS. To view the extensible attributes, click **Data Management -> DNS -> Zones** tab, and then click the **VMware** or **Azure** instance. Select the record and click **Extensible Attribute** in the toolbar in NIOS.



You are able to view the Azure and VMware tenants, which contain the VM and the network. The Azure and VMware tenants are displayed in the **Cloud -> Tenants** tab. The tenant name is created as specified in infoblox.tf. If you click on the Tenant name, you can view the respective networks created.



Related Documentation

Other Terraform documentation:

- [Terraform Documentation](#)
- [Setting up a basic development environment for plugin development.](#)

See also related [Infoblox NIOS documentation](#).

Technical Support

Infoblox Technical Support provides assistance via the Web, e-mail, and telephone. The Infoblox Support web site at <https://support.infoblox.com> provides access to product documentation and release notes, but requires the user ID and password you receive when you register your product online at: <http://www.infoblox.com/support/customer/evaluation-and-registration>.