# » vcd_dnat

Provides a vCloud Director DNAT resource. This can be used to create, modify, and delete destination NATs to map an external IP/port to an internal IP/port.

## » Example Usage

```
resource "vcd_dnat" "web" {
  edge_gateway = "Edge Gateway Name"
  external_ip  = "78.101.10.20"
  port         = 80
  internal_ip  = "10.10.0.5"
  translated_port = 8080
}
```

## » Argument Reference

The following arguments are supported:

- `edge_gateway` - (Required) The name of the edge gateway on which to apply the DNAT
- `external_ip` - (Required) One of the external IPs available on your Edge Gateway
- `port` - (Required) The port number to map
- `internal_ip` - (Required) The IP of the VM to map to

# » vcd_firewall_rules

Provides a vCloud Director Firewall resource. This can be used to create, modify, and delete firewall settings and rules.

## » Example Usage

```
resource "vcd_firewall_rules" "fw" {
  edge_gateway   = "Edge Gateway Name"
  default_action = "deny"

  rule {
    description      = "deny-ftp-out"
    policy           = "deny"
    protocol         = "tcp"
    destination_port = "21"
```

```
    destination_ip   = "any"
    source_port      = "any"
    source_ip        = "10.10.0.0/24"
  }

  rule {
    description      = "allow-outbound"
    policy           = "allow"
    protocol         = "any"
    destination_port = "any"
    destination_ip   = "any"
    source_port      = "any"
    source_ip        = "10.10.0.0/24"
  }
}

resource "vcd_vapp" "web" {
  # ...
}

resource "vcd_firewall_rules" "fw-web" {
  edge_gateway   = "Edge Gateway Name"
  default_action = "drop"

  rule {
    description      = "allow-web"
    policy           = "allow"
    protocol         = "tcp"
    destination_port = "80"
    destination_ip   = "${vcd_vapp.web.ip}"
    source_port      = "any"
    source_ip        = "any"
  }
}
```

## » Argument Reference

The following arguments are supported:

- `edge_gateway` - (Required) The name of the edge gateway on which to apply the Firewall Rules
- `default_action` - (Required) Either ”allow” or ”deny”. Specifies what to do should none of the rules match
- `rule` - (Optional) Configures a firewall rule; see Rules below for details.

### » Rules

Each firewall rule supports the following attributes:

- `description` - (Required) Description of the fireall rule
- `policy` - (Required) Specifies what to do when this rule is matched. Either "allow" or "deny"
- `protocol` - (Required) The protocol to match. One of "tcp", "udp", "icmp" or "any"
- `destination_port` - (Required) The destination port to match. Either a port number or "any"
- `destination_ip` - (Required) The destination IP to match. Either an IP address, IP range or "any"
- `source_port` - (Required) The source port to match. Either a port number or "any"
- `source_ip` - (Required) The source IP to match. Either an IP address, IP range or "any"

## » vcd_network

Provides a vCloud Director VDC Network. This can be used to create, modify, and delete internal networks for vApps to connect.

### » Example Usage

```
resource "vcd_network" "net" {
  name         = "my-net"
  edge_gateway = "Edge Gateway Name"
  gateway      = "10.10.0.1"

  dhcp_pool {
    start_address = "10.10.0.2"
    end_address   = "10.10.0.100"
  }

  static_ip_pool {
    start_address = "10.10.0.152"
    end_address   = "10.10.0.254"
  }
}
```

### » Argument Reference

The following arguments are supported:

- `name` - (Required) A unique name for the network
- `edge_gateway` - (Required) The name of the edge gateway
- `netmask` - (Optional) The netmask for the new network. Defaults to `255.255.255.0`
- `gateway` (Required) The gateway for this network
- `dns1` - (Optional) First DNS server to use. Defaults to `8.8.8.8`
- `dns2` - (Optional) Second DNS server to use. Defaults to `8.8.4.4`
- `dns_suffix` - (Optional) A FQDN for the virtual machines on this network
- `dhcp_pool` - (Optional) A range of IPs to issue to virtual machines that don't have a static IP; see IP Pools below for details.
- `static_ip_pool` - (Optional) A range of IPs permitted to be used as static IPs for virtual machines; see IP Pools below for details.

### » IP Pools

Network interfaces support the following attributes:

- `start_address` - (Required) The first address in the IP Range
- `end_address` - (Required) The final address in the IP Range

## » vcd_snat

Provides a vCloud Director SNAT resource. This can be used to create, modify, and delete source NATs to allow vApps to send external traffic.

### » Example Usage

```
resource "vcd_snat" "outbound" {
  edge_gateway = "Edge Gateway Name"
  external_ip  = "78.101.10.20"
  internal_ip  = "10.10.0.0/24"
}
```

### » Argument Reference

The following arguments are supported:

- `edge_gateway` - (Required) The name of the edge gateway on which to apply the SNAT

- `external_ip` - (Required) One of the external IPs available on your Edge Gateway
- `internal_ip` - (Required) The IP or IP Range of the VM(s) to map from

## » vcd_edgegateway_vpn

Provides a vCloud Director IPsec VPN. This can be used to create, modify, and delete VPN settings and rules.

### » Example Usage

```
resource "vcd_edgegateway_vpn" "vpn" {
    edge_gateway        = "Internet_01(nti0000bi2_123-456-2)"
    name                = "west-to-east"
  description         = "Description"
  encryption_protocol = "AES256"
    mtu                 = 1400
    peer_id             = "64.121.123.11"
    peer_ip_address     = "64.121.123.11"
    local_id            = "64.121.123.10"
    local_ip_address    = "64.121.123.10"
    shared_secret       = "**********************"

    peer_subnets {
        peer_subnet_name = "DMZ_WEST"
        peer_subnet_gateway = "10.0.10.1"
        peer_subnet_mask = "255.255.255.0"
    }

    peer_subnets {
        peer_subnet_name = "WEB_WEST"
        peer_subnet_gateway = "10.0.20.1"
        peer_subnet_mask = "255.255.255.0"
    }

    local_subnets {
        local_subnet_name = "DMZ_EAST"
        local_subnet_gateway = "10.0.1.1"
        local_subnet_mask = "255.255.255.0"
    }

    local_subnets {
        local_subnet_name = "WEB_EAST"
```

```
        local_subnet_gateway = "10.0.22.1"
        local_subnet_mask = "255.255.255.0"
    }
}
```

## » Argument Reference

The following arguments are supported:

- `edge_gateway` - (Required) The name of the edge gateway on which to apply the Firewall Rules
- `name` - (Required) The name of the VPN
- `description` - (Required) A description for the VPN
- `encryption_protocol` - (Required) - E.g. `AES256`
- `local_ip_address` - (Required) - Local IP Address
- `local_id` - (Required) - Local ID
- `mtu` - (Required) - The MTU setting
- `peer_ip_address` - (Required) - Peer IP Address
- `peer_id` - (Required) - Peer ID
- `shared_secret` - (Required) - Shared Secret
- `local_subnets` - (Required) - List of Local Subnets see Local Subnets below for details.
- `peer_subnets` - (Required) - List of Peer Subnets see Peer Subnets below for details.

## » Local Subnets

Each Local Subnet supports the following attributes:

- `local_subnet_name` - (Required) Name of the local subnet
- `local_subnet_gateway` - (Required) Gateway of the local subnet
- `local_subnet_mask` - (Required) Subnet mask of the local subnet

## » Peer Subnets

Each Peer Subnet supports the following attributes:

- `peer_subnet_name` - (Required) Name of the peer subnet
- `peer_subnet_gateway` - (Required) Gateway of the peer subnet
- `peer_subnet_mask` - (Required) Subnet mask of the peer subnet

## » vcd_vapp

Provides a vCloud Director vApp resource. This can be used to create, modify, and delete vApps.

### » Example Usage

```
resource "vcd_network" "net" {
  # ...
}

resource "vcd_vapp" "web" {
  name          = "web"
  catalog_name  = "Boxes"
  template_name = "lampstack-1.10.1-ubuntu-10.04"
  memory        = 2048
  cpus          = 1

  network_name = "${vcd_network.net.name}"
  network_href = "${vcd_network.net.href}"
  ip           = "10.10.104.160"

  metadata {
    role    = "web"
    env     = "staging"
    version = "v1"
  }

  ovf {
    hostname = "web"
  }
}
```

### » Example RAW vApp with No VMS

```
resource "vcd_network" "net" {
  # ...
}

resource "vcd_vapp" "web" {
  name          = "web"
}
```

### » Argument Reference

The following arguments are supported:

- `name` - (Required) A unique name for the vApp
- `catalog_name` - (Optional) The catalog name in which to find the given vApp Template
- `template_name` - (Optional) The name of the vApp Template to use
- `memory` - (Optional) The amount of RAM (in MB) to allocate to the vApp
- `cpus` - (Optional) The number of virtual CPUs to allocate to the vApp
- `initscript` (Optional) A script to be run only on initial boot
- `network_name` - (Optional) Name of the network this vApp should join
- `network_href` - (Deprecated) The vCloud Director generated href of the network this vApp should join. If empty it will use the network name and query vCloud Director to discover this
- `ip` - (Optional) The IP to assign to this vApp. Must be an IP address or one of dhcp, allocated or none. If given the address must be within the `static_ip_pool` set for the network. If left blank, and the network has `dhcp_pool` set with at least one available IP then this will be set with DHCP.
- `metadata` - (Optional) Key value map of metadata to assign to this vApp
- `ovf` - (Optional) Key value map of ovf parameters to assign to VM product section
- `power_on` - (Optional) A boolean value stating if this vApp should be powered on. Default to `true`

## » vcd_vapp_vm

Provides a vCloud Director VM resource. This can be used to create, modify, and delete VMs within a vApp.

**Note:** There is known bug with this implementation, that to use the vcd_vapp_vm resource, you must set the paralellism parameter to 1. We are working on this.

### » Example Usage

```
resource "vcd_network" "net" {
  # ...
}

resource "vcd_vapp" "web" {
  name          = "web"
}
```

```
resource "vcd_vapp_vm" "web2" {
  vapp_name     = "${vcd_vapp.web.name}"
  name          = "web2"
  catalog_name  = "Boxes"
  template_name = "lampstack-1.10.1-ubuntu-10.04"
  memory        = 2048
  cpus          = 1

  ip           = "10.10.104.161"
}

resource "vcd_vapp_vm" "web3" {
  vapp_name     = "${vcd_vapp.web.name}"
  name          = "web3"
  catalog_name  = "Boxes"
  template_name = "lampstack-1.10.1-ubuntu-10.04"
  memory        = 2048
  cpus          = 1

  ip           = "10.10.104.162"
}
```

## » Argument Reference

The following arguments are supported:

- `vapp_name` - (Required) The vApp this VM should belong to.
- `name` - (Required) A unique name for the vApp
- `catalog_name` - (Required) The catalog name in which to find the given vApp Template
- `template_name` - (Required) The name of the vApp Template to use
- `memory` - (Optional) The amount of RAM (in MB) to allocate to the vApp
- `cpus` - (Optional) The number of virtual CPUs to allocate to the vApp
- `initscript` (Optional) A script to be run only on initial boot
- `ip` - (Optional) The IP to assign to this vApp. Must be an IP address or one of dhcp, allocated or none. If given the address must be within the `static_ip_pool` set for the network. If left blank, and the network has `dhcp_pool` set with at least one available IP then this will be set with DHCP.
- `power_on` - (Optional) A boolean value stating if this vApp should be powered on. Default to `true`