



IaC and Network Insights API

ACE Team





Infrastructure as Code



What it is

- Use Infrastructure as Code to provision and manage any cloud, infrastructure, or service
- Write declarative configuration files – define desired state
- Plan and predict changes
- Create reproducible infrastructure – if resource already exists, it won't recreate it
- Maintains knowledge of resources in a database called **State**
 - State maps config to real world



Aviatrix Terraform Provider

- Multi-lingual entity responsible for API interactions with CSPs
- Exposes resources in those CSPs for any account/subscription that has been onboarded
- Feature parity with Controller code

The screenshot shows a browser window displaying the Terraform Registry at registry.terraform.io/providers/AviatrixSystems/aviatrix/latest/docs. The page title is "Docs overview | AviatrixSystem". The URL bar shows the full path. The top navigation bar includes "Terraform Registry", "Search Providers and Modules", "Browse", "Publish", and "Sign-in". The main content area is titled "Aviatrix Provider". It contains a sidebar with "AVIATRIX DOCUMENTATION" and a list of resources: "aviatrix provider", "Guides", "Accounts", "CloudWAN", "Firewall Network", "Gateway", "Multi-Cloud Transit", "OpenVPN", "Peering", "Security", "Settings", "Site2Cloud", "TGW Orchestrator", "Useful Tools", and "Deprecated". The main content area also features a "NOTE" section about provider versioning and a "Example Usage" section with sample Terraform code.

Provider: AviatrixSystems / aviatrix / Version 2.17.2 / Latest Version

Overview Documentation USE PROVIDER +

Aviatrix Provider

The Aviatrix provider is used to interact with the many resources supported by Aviatrix, which can be built upon various cloud infrastructure providers such as AWS, Azure, Google Cloud, and Oracle Cloud. It needs to be configured with the proper credentials before it can be used.

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

NOTE:

Although version is an optional argument, we highly recommend users to specify the proper Aviatrix provider release version corresponding to their Controller version in order to avoid potential compatibility issues. Please see the [compatibility chart](#) for full details. For more information on versioning, a native Terraform provider argument, see [here](#).

Example Usage

```
# Configure Aviatrix provider
provider "aviatrix" {
  controller_ip      = "1.2.3.4"
  username           = "admin"
  password           = "password"
  skip_version_validation = false
  version            = "2.5.0"
```



Aviatrix Terraform Resources – Examples

- # Create an Aviatrix AWS Gateway

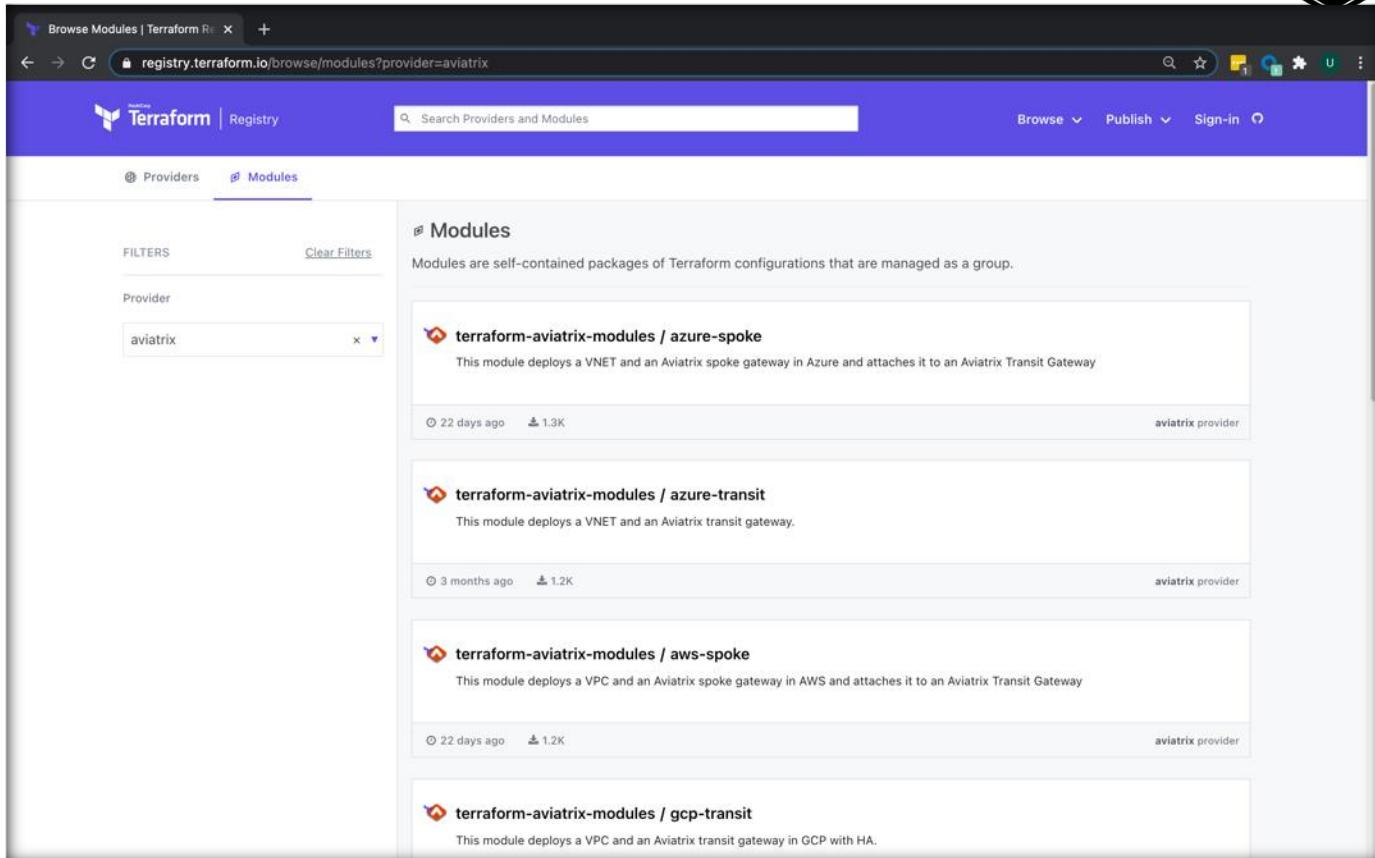
```
resource "aviatrix_gateway" "test_gateway_aws" {  
  
    cloud_type      = 1  
  
    account_name   = "devops-aws"  
  
    gw_name        = "avtx-gw-1"  
    vpc_id         = "vpc-abcdef"  
    vpc_reg        = "us-west-1"  
    gw_size        = "t2.micro"  
  
    subnet         = "10.0.0.0/24"  
  
}
```

- # Create an Aviatrix Azure Gateway

```
resource "aviatrix_gateway" "test_gateway_azure" {  
  
    cloud_type      = 8  
  
    account_name   = "devops-azure"  
  
    gw_name        = "avtx-gw-azure"  
    vpc_id         = "gateway:test-gw-123"  
    vpc_reg        = "West US"  
    gw_size        = "Standard_D2"  
    subnet         = "10.13.0.0/24"  
  
}
```

Aviatrix Terraform Modules

- “*Repeatable++*”
- Similar to the concepts of libraries, packages, or modules found in most programming languages
- Provide many of the same benefits
- ~10X reduction in lines of code
- Can be found on Terraform Registry



The screenshot shows the Terraform Registry interface with the search bar set to "registry.terraform.io/browse/modules?provider=aviatrix". The "Modules" tab is selected. A filter for "Provider" is applied, showing "aviatrix". The results list four modules:

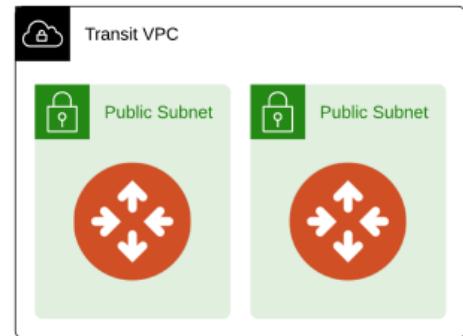
- terraform-aviatrix-modules / azure-spoke**
This module deploys a VNET and an Aviatrix spoke gateway in Azure and attaches it to an Aviatrix Transit Gateway.
22 days ago · 1.3K · aviatrix provider
- terraform-aviatrix-modules / azure-transit**
This module deploys a VNET and an Aviatrix transit gateway.
3 months ago · 1.2K · aviatrix provider
- terraform-aviatrix-modules / aws-spoke**
This module deploys a VPC and an Aviatrix spoke gateway in AWS and attaches it to an Aviatrix Transit Gateway.
22 days ago · 1.2K · aviatrix provider
- terraform-aviatrix-modules / gcp-transit**
This module deploys a VPC and an Aviatrix transit gateway in GCP with HA.



Aviatrix Terraform Module – Example

- # Create a VPC and a set of Aviatrix transit gateways.

```
module "transit_aws_1" {  
  
    source  = "terraform-aviatrix-modules/mc-transit/aviatrix"  
  
    version = "1.1.2"  
  
    cloud    = "aws"  
  
    cidr     = "10.1.0.0/20"  
  
    region   = "eu-west-1"  
  
    account  = "AWS-account"  
  
}  
  
ha_gw set to true by default
```

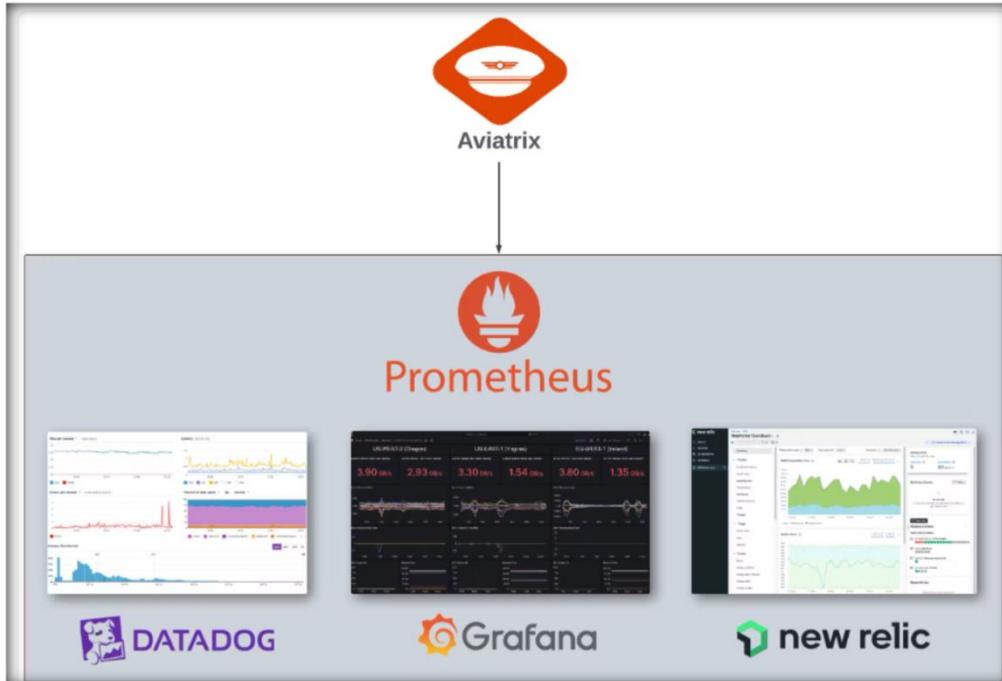




Network Insights API

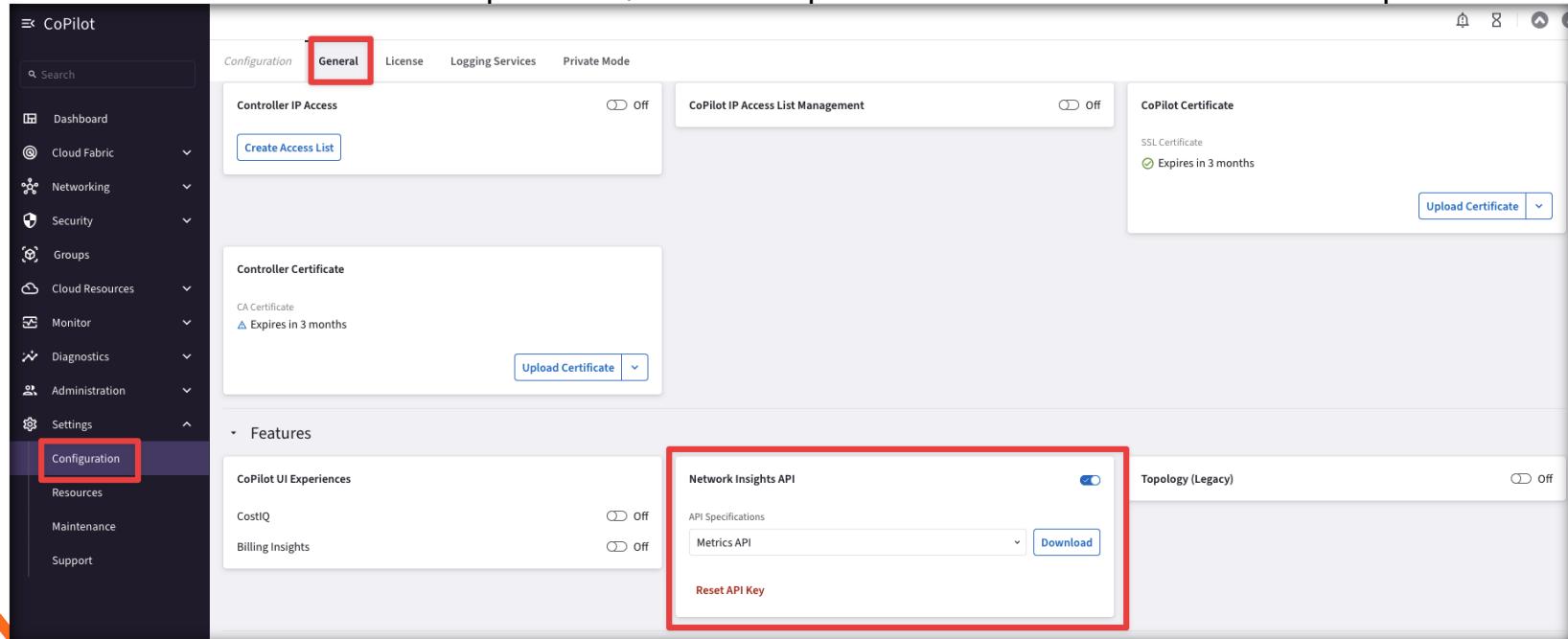
Network Insights API (part.1)

- The Aviatrix Network Insights API allows you to retrieve network metric and status data across your Aviatrix data plane. Using the metric and status APIs, you can integrate with **third-party tools** for data analysis and visualization of the performance and health of your Aviatrix-managed resources. The APIs also support data retention for compliance.



Network Insights API (part.2)

- The Network Insights API supports **Prometheus** and JSON formats. All data transmissions are encrypted using industry-standard protocols.
- An **API key** is used to authenticate requests for your Aviatrix services.
 - The Aviatrix API uses port 443, the same port as the CoPilot UI. Ensure that port 443 is



The screenshot shows the Aviatrix CoPilot UI configuration interface. The left sidebar has sections for Dashboard, Cloud Fabric, Networking, Security, Groups, Cloud Resources, Monitor, Diagnostics, Administration, and Settings. The Configuration section is selected and highlighted with a red box. The main content area has tabs for Configuration, General (which is also highlighted with a red box), License, Logging Services, and Private Mode. Under Configuration, there are sections for Controller IP Access (with a 'Create Access List' button) and Controller Certificate (with a 'CA Certificate' that expires in 3 months and an 'Upload Certificate' button). In the Features section, the Network Insights API is shown with its toggle switch turned on. Below it, the API Specifications dropdown is set to 'Metrics API' and there is a 'Download' button. A 'Reset API Key' link is also present. To the right, there is a section for Topology (Legacy) with its own toggle switch.



Next: Lab 12 - Terraform and Network Insights API