ivanti

Pulse Secure Virtual Traffic Manager: Terraform Provider Reference Guide

21.3

Copyright Notice

This document is provided strictly as a guide. No guarantees can be provided or expected. This document contains the confidential information and/or proprietary property of Ivanti, Inc. and its affiliates (referred to collectively as "Ivanti") and may not be disclosed or copied without prior written consent of Ivanti.

Ivanti retains the right to make changes to this document or related product specifications and descriptions, at any time, without notice. Ivanti makes no warranty for the use of this document and assumes no responsibility for any errors that can appear in the document nor does it make a commitment to update the information contained herein. For the most current product information, please visit www.Ivanti.com.

Copyright © 2021, Ivanti. All rights reserved.

Protected by patents, see https://www.ivanti.com/patents.

Contents

Preface

Document conventions

The document conventions describe text formatting conventions, command syntax conventions, and important notice formats used in Ivanti technical documentation.

Text formatting conventions

Text formatting conventions such as boldface, italic, or Courier font may be used in the flow of the text to highlight specific words or phrases.

Format	Description
bold text	Identifies command names
	Identifies keywords and operands
	Identifies the names of user-manipulated GUI elements
	Identifies text to enter at the GUI
italic text	Identifies emphasis
	Identifies variables
	Identifies document titles
Courier Font	Identifies command output
	Identifies command syntax examples

Command syntax conventions

Bold and italic text identify command syntax components. Delimiters and operators define groupings of parameters and their logical relationships.

Convention	Description
bold text	Identifies command names, keywords, and command options.

Convention	Description
italic text	Identifies a variable.
[ ]	Syntax components displayed within square brackets are optional. Default responses to system prompts are enclosed in square brackets.
{ x | y | z }	A choice of required parameters is enclosed in curly brackets separated by vertical bars. You must select one of the options.
x | y	A vertical bar separates mutually exclusive elements.
< >	Non-printing characters, for example, passwords, are enclosed in angle brackets.
...	Repeat the previous element, for example, member[member...].
\	Indicates a "soft" line break in command examples. If a backslash separates two lines of a command input, enter the entire command at the prompt without the backslash.

Notes and Warnings

Note, Attention, and Caution statements might be used in this document.

A Note provides a tip, guidance, or advice, emphasizes important information, or provides a reference to related information.

ATTENTION
An Attention statement indicates a stronger note, for example, to alert you when traffic might be interrupted or the device might reboot.

CAUTION
A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

Requesting Technical Support

Technical product support is available through the Ivanti Global Support Center (PSGSC). If you have a support contract, file a ticket with PSGSC.

- Product warranties—For product warranty information, visit https://support.pulsesecure.net/product-service-policies/

Self-Help Online Tools and Resources

For quick and easy problem resolution, Ivanti provides an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: https://support.pulsesecure.net

- Search for known bugs: https://support.pulsesecure.net

- Find product documentation: https://www.ivanti.com/support/product-documentation

- Download the latest versions of software and review release notes: https://support.pulsesecure.net

- Open a case online in the CSC Case Management tool: https://support.pulsesecure.net

- To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: https://support.pulsesecure.net

For important product notices, technical articles, and to ask advice:

- Search the Pulse Secure Knowledge Center for technical bulletins and security advisories: https://kb.pulsesecure.net

- Ask questions and find solutions at the Pulse Community online forum: https://community.pulsesecure.net

Opening a Case with PSGSC

You can open a case with PSGSC on the Web or by telephone.

- Use the Case Management tool in the PSGSC at https://support.pulsesecure.net.

- Call 1-844 751 7629 (Toll Free, US).

For international or direct-dial options in countries without toll-free numbers, see https://support.pulsesecure.net/support/support-contacts/

Overview

This chapter provides an overview of Pulse Secure Virtual Traffic Manager (the Traffic Manager).

Introducing the Traffic Manager

The Traffic Manager product family provides high-availability, application-centric traffic management and load balancing solutions in a range of software, hardware-ready, virtual appliance, and cloud-compute product variants. They provide control, intelligence, security and resilience for all your application traffic.

The Traffic Manager is intended for organizations hosting valuable business-critical services, such as TCP-based and UDP-based services like HTTP (web) and media delivery, and XML-based services such as Web Services.

Introducing the Terraform Provider

Terraform is an open-source infrastructure-as-code tool produced by HashiCorp, Inc.

System administrators use Terraform to create and maintain configuration templates for your application infrastructure. Ivanti provides a set of tools to Terraform, known as a *provider*, that enable you to define Traffic Manager configurations within your application templates. Terraform can then provision Traffic Manager instances as part of a wider deployment.

To learn more about Terraform, see https://www.terraform.io.

Getting Started

This chapter contains information about getting started using the Traffic Manager Terraform provider.

Installing the Traffic Manager Terraform Provider

To obtain the Pulse Secure Virtual Traffic Manager Terraform provider, download the source files from the *pulse-vadc* pages at GitHub: https://github.com/pulse-vadc/terraform-provider-vtm

Make sure you download the version that corresponds to your target Traffic Manager's REST API version.

To install the provider in Terraform, follow the third-party plugin instructions at:https://www.terraform.io/docs/configuration/providers.html#third-party-plugins

Initializing the Terraform Provider

To use the Traffic Manager Terraform provider, initialize it within your Terraform template. The provider requires four arguments:

Field Name	Type	Description
base_url	string	The base URL of the Traffic Manager REST API. Use either a direct Traffic Manager URL, or through a Pulse Secure Services Director (Services Director) proxy.
username	string	The admin username of either the Traffic Manager or the Services Director proxy you are connecting to.
password	string	The admin password of either the Traffic Manager or the Services Director proxy you are connecting to.
verify_ssl_cert	boolean	If set to true, the provider checks that the Traffic Manager/Services Director provides a trusted SSL certificate when connecting to the REST API. If set to false, untrusted (for example, self-signed) certificates are accepted. Default: true.

The following code provides examples of how to initialize the provider:

```
# Direct to the Traffic Manager
```

```
provider "vtm" {

   base_url = "https://192.0.2.10:9070/api"

   username = "admin"

   password = "mYvtm53cret!"

   verify_ssl_cert = true

}


# Through a Services Director proxy

provider "vtm" {

   base_url = "https://192.0.2.100:8100/api/tmcm/<VERSION>/instance/<INSTANCE_ID>"

   username = "admin"

   password = "mYsd53cret!"

   verify_ssl_cert = true

}
```

Resource and Data Source Reference

This chapter introduces the resource model, demonstrates it's usage through the provider, and includes a complete listing of all available Traffic Manager configuration resources and data sources.

About the Resource Model

The resource model reference included in this chapter lists all configuration resources and data sources available through the Traffic Manager REST API. Each sub-section relates to a specific resource type or data source, showing its name, description, unique URI path, and a table of properties.

Each property contains a description and data type. Additional information is provided where applicable, such as default value and permitted values (for enumerated types). For Table-type properties, a list of the Primary and Sub keys is provided.

For configuration resources, each property, and table keys where applicable, include an indicator of whether or not that item is a required value when creating configuration based on that resource.

For further information concerning the Traffic Manager REST API, see the *Pulse Secure Virtual Traffic Manager: REST API Guide*, available from the Ivanti website.

Using the Resource Model

Resources can represent objects of a singular nature (for example, global settings) where only one instance of that resource type is ever present, or collections of objects (for example, virtual servers) where multiple instances might be created. To uniquely identify a specific instance of a collection-type resource, use the "name" property. Resources of a singular type do not require the "name" property.

Furthermore, there are two types of collection:

- Key/value resources. For example, virtual servers.

- Raw text files. For example, TrafficScript rules

Key/value resources have unique data structures, as defined in the resource model reference, whereas raw file resources have only two parameters; "name" and "content".

ATTENTION
Ivanti strongly recommends that singular object resources are only specified in one template for any cluster. Specifying these resources in multiple templates can lead to altering cluster-wide settings that impact other services.

Configuration Resources

Configuration resources map to objects in the Traffic Manager's configuration system. The Terraform provider uses these resource definitions to provision services hosted on your Traffic Manager instances.

Setting Configuration Values Using the Resource Model

The following code samples show how to set configuration properties for different resource types:

```
# Single object configuration
resource "vtm_global_settings" "global_settings" {
   admin_support_tls1 = false
   connection_idle_timeout = 30
}
# Collection-type key/value object configuration
resource "vtm_pool" "my_pool" {
   name = "MyPool"
   monitors = ["Ping", "Simple HTTP"]
   load_balancing_algorithm = "weighted_least_connections"
   nodes_table {
      node = "192.0.2.101:80"
      weight = 1
   }
   nodes_table {
      node = "192.0.2.102:80"
      weight = 3
   }
   nodes_table {
      node = "192.0.2.103:80"
      weight = 3
      state = "disabled"
   }
}



# Collection-type raw text object configuration
resource "vtm_rule" "my_rule" {
   name = "MyRule"
   content = <<EOF
$hostHeader = http.getHostHeader();
```

```
if($hostHeader == "www.example.com") {
   pool.use("ExamplePool");
}
EOF
}
```

Dependancy Tree

Generally speaking, the Traffic Manager REST API is tolerant of objects being created out-of-order. In other words, the REST API allows you to create a virtual server that references a TrafficScript rule that does not yet exist, even though this would trigger an error in the Traffic Manager diagnostics and event log. This error is suppressed after the corresponding rule is subsequently added. However, there are exceptions to this arrangement.

Ivanti recommends that all resource dependencies are explicitly specified by using parameter interpolation when creating relationships, rather than hard-coding names. For example, rather than providing a list of rule names to a virtual server as strings, specify the names as variables to aid Terraform in building the correct dependancy tree. The following code samples demonstrate this process.

The code sample below shows the partial configuration of a virtual server with named request rules:

```
resource "vtm_virtual_server" "example" {
   name = "ExampleVS"
   request_rules = ["rule1", "rule2"]
   ...
}
```

Rather than specifying the hard-coded strings "rule1" and "rule2" in the list of request rules, use instead a set of variables to provide the rule names. The following code sample demonstrates the syntax used to reference properties of other resource objects in the provider:

```
resource "vtm_virtual_server" "example" {
   name = "ExampleVS"
   request_rules = ["${vtm_rule.rule1.name}", "${vtm_rule.rule2.name}"]
   ...
}
```

Resource Reference

Use the configuration resource definitions in this section to provision services on your Traffic Manager instances.

vtm_action_program

URI Endpoint: /api/tm/8.3/config/active/action_programs

This is a program or script that can be referenced and used by actions of type 'Program'

```
# Example usage of vtm_action_program resource...
resource "vtm_action_program" "my_action_program" {
    content = ""
    name = "MyActionProgram"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
content	Object content • Type: string • Required: true

vtm_ssl_admin_ca

URI Endpoint: /api/tm/8.3/config/active/ssl/admin_cas

The conf/ssl/admin_cas directory contains SSL certificate authority certificates (CAs) and certificate revocation lists (CRLs) which can be used when validating connections made by the admin server for user authentication. CAs and CRLs can be managed under the Catalogs > SSL > Admin CAs and CRLs section of the Admin Server UI or by using functions under the Catalog.SSL.AdminCertificateAuthorities section of the SOAP API and CLI.

Property	Description
name	Name of the object • Type: string • Required: true
content	Object content • Type: string • Required: true

vtm_action

URI Endpoint: /api/tm/8.3/config/active/actions

A response to an event occurring in your traffic manager. An example of an action might be sending an email or writing a line to a log file.

```
# Example usage of vtm_action resource...
resource "vtm_action" "my_action" {
    name = "MyAction"
    type =
}
```

Property	Description
name	Name of the object • Type: string • Required: true
email_from	The e-mail address from which messages will appear to originate. • Type: String • Required: false • Default value: "vTM@%hostname%"

Property	Description
email_server	The SMTP server to which messages should be sent. This must be a valid IPv4 address or resolvable hostname (with optional port). • Type: String • Required: false • Default value: <none>
email_to	A set of e-mail addresses to which messages will be sent. • Type: Set(String) • Required: false • Default value: <none>
log_file	The full path of the file to log to. The text %zeushome% will be replaced with the location where the software is installed. • Type: String • Required: false • Default value: <none>
note	A description of the action. • Type: FreeformString • Required: false • Default value: <none>
program_arguments	A table containing arguments and argument values to be passed to the event handling program. • Type: Table • Required: false • Primary key:

Property	Description

	• name (String): The name of the argument to be passed to the event handling program. (Required)
	
	• Sub keys:
	
	    • value (String): The value of the argument to be passed to the event handling program. (Required)
	
	    • description (String): A description for the argument provided to the program.

program_arguments_json	A Traffic Manager REST-compatible JSON representation of the "program_arguments" table property. Use this field with the "vtm_action_arguments_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
program_program	The program to run. • Type: String • Required: false • Default value: <none>
soap_additional_data	Additional information to send with the SOAP call. • Type: String • Required: false • Default value: <none>
soap_password	The password for HTTP basic authentication. • Type: Password • Required: false

Property	Description
	• Default value: <none>
soap_proxy	The address of the server implementing the SOAP interface (For example, https://example.com). • Type: String • Required: false • Default value: <none>
soap_username	Username for HTTP basic authentication. Leave blank if you do not wish to use authentication. • Type: String • Required: false • Default value: <none>
syslog_msg_len_limit	Maximum length in bytes of a message sent to the remote syslog. Messages longer than this will be truncated before they are sent. • Type: UInt • Required: false • Default value: "2048"
syslog_sysloghost	The host and optional port to send syslog messages to (if empty, messages will be sent to localhost). • Type: String • Required: false • Default value: <none>
timeout	How long the action can run for before it is stopped automatically (set to 0 to disable timeouts). • Type: UInt • Required: false

Property	Description
	• Default value: "60"
trap_auth_password	The authentication password for sending a Notify over SNMPv3. Blank to send unauthenticated traps. • Type: Password • Required: false • Default value: \<none\>
trap_community	The community string to use when sending a Trap over SNMPv1 or a Notify over SNMPv2c. • Type: String • Required: false • Default value: \<none\>

trap_hash_algorithm	The hash algorithm for SNMPv3 authentication.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "md5"
	
	• Permitted values:
	
	    "md5": MD5
	
	    "sha1": SHA-1

trap_priv_password	The encryption password to encrypt a Notify message for SNMPv3. Requires that authentication also be configured. Blank to send unencrypted traps. • Type: Password • Required: false • Default value: \<none\>

Property	Description
trap_traphost	The hostname or IPv4 address and optional port number that should receive traps. • Type: String • Required: false • Default value: <none>
trap_username	The SNMP username to use to send the Notify over SNMPv3. • Type: String • Required: false • Default value: <none>

trap_version	The SNMP version to use to send the Trap/Notify.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "snmpv1"
	
	• Permitted values:
	
	    "snmpv1": SNMPv1
	
	    "snmpv2c": SNMPv2c
	
	    "snmpv3": SNMPv3

type	The action type. • Type: Enum(String) • Required: true • Default value: <none> • Permitted values: "email": E-Mail

Property	Description
	"log": Log to File
	"program": Program
	"soap": SOAP Callback
	"syslog": Log to Syslog
	"trap": SNMP Notify or Trap
verbose	Enable or disable verbose logging for this action. • Type: Boolean • Required: false • Default value: false

vtm_aptimizer_scope

URI Endpoint: /api/tm/8.3/config/active/aptimizer/scopes

Application scopes define criteria that match URLs to specific logical web applications hosted by a virtual server.

```
# Example usage of vtm_aptimizer_scope resource...
resource "vtm_aptimizer_scope" "my_aptimizer_scope" {
    name = "MyAptimizerScope"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
canonical_hostname	If the hostnames for this scope are aliases of each other, the canonical hostname will be used for requests to the server. • Type: String

Property	Description
	• Required: false • Default value: <none>
hostnames	The hostnames to limit acceleration to. • Type: Set(String) • Required: false • Default value: <none>
root	The root path of the application defined by this application scope. • Type: String • Required: false • Default value: "/"

vtm_bgpneighbor

URI Endpoint: /api/tm/8.3/config/active/bgpneighbors

The conf/bgpneighbors directory contains configuration files for BGP neighbors. The name of a file is the name of the neighbor configuration that it defines. BGP neighbors can be managed under the System > Fault Tolerance > BGP Neighbors section of the Admin UI, or by using functions under the BGPNeighbors section of the SOAP API and CLI.

```
# Example usage of vtm_bgpneighbor resource...
resource "vtm_bgpneighbor" "my_bgpneighbor" {
    name = "MyBgpneighbor"
}
```

Property	Description
name	Name of the object • Type: string • Required: true

Property	Description
address	The IP address of the BGP neighbor • Type: String • Required: false • Default value: \<none\>
advertisement_interval	The minimum interval between the sending of BGP routing updates to neighbors. Note that as a result of jitter, as defined for BGP, the interval during which no advertisements are sent will be between 75% and 100% of this value. • Type: UInt • Required: false • Default value: "5"
as_number	The AS number for the BGP neighbor • Type: UInt • Required: false • Default value: "65534"
authentication_password	The password to be used for authentication of sessions with neighbors • Type: String • Required: false • Default value: \<none\>
holdtime	The period after which the BGP session with the neighbor is deemed to have become idle - and requires re-establishment - if the neighbor falls silent. • Type: UInt • Required: false

Property	Description
	• Default value: "90"
keepalive	The interval at which messages are sent to the BGP neighbor to keep the mutual BGP session established. • Type: UInt • Required: false • Default value: "30"
machines	The traffic managers that are to use this neighbor • Type: Set(String) • Required: false • Default value: <none>

vtm_bandwidth

URI Endpoint: /api/tm/8.3/config/active/bandwidth

A Bandwidth class, which can be assigned to a virtual server or pool in order to limit the number of bytes per second used by inbound or outbound traffic.

```
# Example usage of vtm_bandwidth resource...
resource "vtm_bandwidth" "my_bandwidth" {
    name = "MyBandwidth"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
maximum	The maximum bandwidth to allocate to connections that are associated with this bandwidth class (in kbits/second).

Property	Description
	• Type: UInt • Required: false • Default value: "10000"
note	A description of this bandwidth class. • Type: FreeformString • Required: false • Default value: <none>

sharing	The scope of the bandwidth class.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "cluster"
	
	• Permitted values:
	
	    "cluster": Bandwidth is shared across all traffic managers
	
	    "connection": Each connection can use the maximum rate
	
	    "machine": Bandwidth is shared per traffic manager

vtm_cloud_api_credential

URI Endpoint: /api/tm/8.3/config/active/cloud_api_credentials

Cloud credentials used in cloud API calls

```
# Example usage of vtm_cloud_api_credential resource...
resource "vtm_cloud_api_credential" "my_cloud_api_credential" {
    name = "MyCloudApiCredential"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
api_server	The vCenter server hostname or IP address. • Type: String • Required: false • Default value: <none>
cloud_api_timeout	The traffic manager creates and destroys nodes via API calls. This setting specifies (in seconds) how long to wait for such calls to complete. • Type: UInt • Required: false • Default value: "200"
cred1	The first part of the credentials for the cloud user. Typically this is some variation on the username concept. • Type: String • Required: false • Default value: <none>
cred2	The second part of the credentials for the cloud user. Typically this is some variation on the password concept. • Type: Password • Required: false • Default value: <none>
cred3	The third part of the credentials for the cloud user. Typically this is some variation on the authentication token concept.

Property	Description
	• Type: Password • Required: false • Default value: <none>
script	The script to call for communication with the cloud API. • Type: String • Required: false • Default value: <none>
update_interval	The traffic manager will periodically check the status of the cloud through an API call. This setting specifies the interval between such updates. • Type: UInt • Required: false • Default value: "30"

vtm_custom

URI Endpoint: /api/tm/8.3/config/active/custom

Custom configuration sets store arbitrary named values. These values can be read by SOAP or REST clients.

```
# Example usage of vtm_custom resource...
resource "vtm_custom" "my_custom" {
    name = "MyCustom"
}
```

Property	Description
name	Name of the object • Type: string

Property	Description
	• Required: true

string_lists	This table contains named lists of strings
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • name (String): Name of list (Required)
	
	• Sub keys:
	
	    • value (List(String)): Named list of user-specified strings. (Required)

string_lists_json	A Traffic Manager REST-compatible JSON representation of the "string_lists" table property. Use this field with the "vtm_custom_string_lists_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>

vtm_dns_server_zone

URI Endpoint: /api/tm/8.3/config/active/dns_server/zones

The conf/dnsserver/zones/ file contains zone metadata

```
# Example usage of vtm_dns_server_zone resource...
resource "vtm_dns_server_zone" "my_dns_server_zone" {
    name = "MyDnsServerZone"
    origin =
    zonefile =
}
```

Property	Description
name	Name of the object • Type: string • Required: true
origin	The domain origin of this Zone. • Type: String • Required: true • Default value: <none>
zonefile	The Zone File encapsulated by this Zone. • Type: String • Required: true • Default value: <none>

vtm_dns_server_zone_file

URI Endpoint: /api/tm/8.3/config/active/dns_server/zone_files

The conf/dnsserver/zonefiles/ directory contains files that define DNS zones.

```
# Example usage of vtm_dns_server_zone_file resource...
resource "vtm_dns_server_zone_file" "my_dns_server_zone_file" {
    content = ""
    name = "MyDnsServerZoneFile"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
content	Object content

Property	Description
	• Type: string • Required: true

vtm_event_type

URI Endpoint: /api/tm/8.3/config/active/event_types

Configuration that ties actions to a set of events that trigger them.

```
# Example usage of vtm_event_type resource...
resource "vtm_event_type" "my_event_type" {
    name = "MyEventType"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
actions	The actions triggered by events matching this event type, as a list of action references. • Type: List(Reference(config-event-action)) • Required: false • Default value: <none>
built_in	If set to Yes this indicates that this configuration is built-in (provided as part of the software) and must not be deleted or edited. • Type: Boolean • Required: false • Default value: false

Property	Description
cloudcredentials_event_tags	Cloud credentials event tags • Type: List(String) • Required: false • Default value: \<none\>
cloudcredentials_objects	Cloud credentials object names • Type: List(String) • Required: false • Default value: \<none\>
config_event_tags	Configuration file event tags • Type: List(String) • Required: false • Default value: \<none\>
faulttolerance_event_tags	Fault tolerance event tags • Type: List(String) • Required: false • Default value: \<none\>
general_event_tags	General event tags • Type: List(String) • Required: false • Default value: \<none\>
glb_event_tags	GLB service event tags • Type: List(String) • Required: false

Property	Description
	• Default value: <none>
glb_objects	GLB service object names • Type: List(String) • Required: false • Default value: <none>
java_event_tags	Java event tags • Type: List(String) • Required: false • Default value: <none>
licensekeys_event_tags	License key event tags • Type: List(String) • Required: false • Default value: <none>
licensekeys_objects	License key object names • Type: List(String) • Required: false • Default value: <none>
locations_event_tags	Location event tags • Type: List(String) • Required: false • Default value: <none>
locations_objects	Location object names • Type: List(String)

Property	Description
	• Required: false • Default value: \<none\>
monitors_event_tags	Monitor event tags • Type: List(String) • Required: false • Default value: \<none\>
monitors_objects	Monitors object names • Type: List(String) • Required: false • Default value: \<none\>
note	A description of this event type. • Type: FreeformString • Required: false • Default value: \<none\>
pools_event_tags	Pool key event tags • Type: List(String) • Required: false • Default value: \<none\>
pools_objects	Pool object names • Type: List(String) • Required: false • Default value: \<none\>
protection_event_tags	Service protection class event tags

Property	Description
	• Type: List(String) • Required: false • Default value: \<none>
protection_objects	Service protection class object names • Type: List(String) • Required: false • Default value: \<none>
rules_event_tags	Rule event tags • Type: List(String) • Required: false • Default value: \<none>
rules_objects	Rule object names • Type: List(String) • Required: false • Default value: \<none>
slm_event_tags	SLM class event tags • Type: List(String) • Required: false • Default value: \<none>
slm_objects	SLM class object names • Type: List(String) • Required: false • Default value: \<none>

Property	Description
ssl_event_tags	SSL event tags • Type: List(String) • Required: false • Default value: <none>
sslhw_event_tags	SSL hardware event tags • Type: List(String) • Required: false • Default value: <none>
trafficscript_event_tags	TrafficScript event tags • Type: List(String) • Required: false • Default value: <none>
vservers_event_tags	Virtual server event tags • Type: List(String) • Required: false • Default value: <none>
vservers_objects	Virtual server object names • Type: List(String) • Required: false • Default value: <none>
zxtms_event_tags	Traffic manager event tags • Type: List(String) • Required: false

Property	Description
	• Default value: <none>
zxtms_objects	Traffic manager object names • Type: List(String) • Required: false • Default value: <none>

vtm_extra_file

URI Endpoint: /api/tm/8.3/config/active/extra_files

A user-uploaded file. Such files can be used in TrafficScript code using the resource.get function.

```
# Example usage of vtm_extra_file resource...
resource "vtm_extra_file" "my_extra_file" {
    content = ""
    name = "MyExtraFile"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
content	Object content • Type: string • Required: true

vtm_glb_service

URI Endpoint: /api/tm/8.3/config/active/glb_services

A global load balancing service is used by a virtual server to modify DNS requests in order load balance data across different GLB locations.

```
# Example usage of vtm_glb_service resource...
resource "vtm_glb_service" "my_glb_service" {
    name = "MyGlbService"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
algorithm	Defines the global load balancing algorithm to be used. • Type: Enum(String) • Required: false • Default value: "hybrid" • Permitted values: "chained": Sends traffic to one location at a time, until that location fails where the next one in the chain is used. "geo": Distributes traffic based solely on the geographic location of each client. "hybrid": Distribute traffic based on both the load and geographic location. "load": Distributes traffic based on the current load to each location. "round_robin": Distributes traffic by assigning each request to a new location in turn. Over a period of time, all locations will receive the same number of requests.

Property	Description
	"weighted_random": Distributes traffic in a random way, but according to a weighted policy defined by individual location weights
all_monitors_needed	Do all monitors assigned to a location need to report success in order for it to be considered healthy? • Type: Boolean • Required: false • Default value: true
autorecovery	The last location to fail will be available as soon as it recovers. • Type: Boolean • Required: false • Default value: true
chained_auto_failback	Enable/Disable automatic failback mode. • Type: Boolean • Required: false • Default value: false
chained_location_order	The locations this service operates for and defines the order in which locations fail. • Type: List(String) • Required: false • Default value: <none>
disable_on_failure	Locations recovering from a failure will become disabled. • Type: Boolean • Required: false

Property	Description
	• Default value: false

dnssec_keys	A table mapping domains to the private keys that authenticate them
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • domain (String): A domain authenticated by the associated private keys. (Required)
	
	• Sub keys:
	
	    • ssl_key (Set(String)): Private keys that authenticate the associated domain. (Required)

dnssec_keys_json	A Traffic Manager REST-compatible JSON representation of the "dnssec_keys" table property. Use this field with the "vtm_glb_ service_dnssec_keys_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
domains	The domains shown here should be a list of Fully Qualified Domain Names that you would like to balance globally. Responses from the back end DNS servers for queries that do not match this list will be forwarded to the client unmodified. Note: "*" may be used as a wild card. • Type: Set(String) • Required: false • Default value: <none>
enabled	Enable/Disable our response manipulation of DNS.

Property	Description
	• Type: Boolean • Required: false • Default value: false
geo_effect	How much should the locality of visitors affect the choice of location used? This value is a percentage, 0% means that no locality information will be used, and 100% means that locality will always control which location is used. Values between the two extremes will act accordingly. • Type: UInt • Required: false • Default value: "50"
last_resort_response	The response to be sent in case there are no locations available. • Type: Set(String) • Required: false • Default value: <none>
location_draining	This is the list of locations for which this service is draining. A location that is draining will never serve any of its service IP addresses for this domain. This can be used to take a location off-line. • Type: Set(String) • Required: false • Default value: <none>
location_settings	Table containing location specific settings. • Type: Table • Required: false

Property	Description

	• Primary key:
	
	    • location (String): Location to which the associated settings apply. (Required)
	
	• Sub keys:
	
	    • weight (UInt): Weight for this location, for use by the weighted random algorithm.
	
	    • ips (Set(String)): The IP addresses that are present in a location. If the Global Load Balancer decides to direct a DNS query to this location, then it will filter out all IPs that are not in this list. (Required)
	
	    • monitors (Set(String)): The monitors that are present in a location.

location_settings_json	A Traffic Manager REST-compatible JSON representation of the "location_settings" table property. Use this field with the "vtm_glb_ service_location_settings_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
log_enabled	Log connections to this GLB service? • Type: Boolean • Required: false • Default value: false
log_filename	The filename the verbose query information should be logged to. Appliances will ignore this. • Type: String

Property	Description
	• Required: false • Default value: "%zeushome%/zxtm/log/services/%g.log"
log_format	The format of the log lines. • Type: String • Required: false • Default value: "%t, %s, %l, %q, %g, %n, %d, %a"
return_ips_on_fail	Return all or none of the IPs under complete failure. • Type: Boolean • Required: false • Default value: true
rules	Response rules to be applied in the context of the service, in order, comma separated. • Type: List(Reference(config-trafficscript)) • Required: false • Default value: <none>
ttl	The TTL for the DNS resource records handled by the GLB service. • Type: Int • Required: false • Default value: "-1"

vtm_global_settings

URI Endpoint: /api/tm/8.3/config/active/global_settings

General settings that apply to every machine in the cluster.

```
# Example usage of vtm_global_settings resource...
resource "vtm_global_settings" "my_global_settings" {


}
```

Property	Description
accepting_delay	How often, in milliseconds, each traffic manager child process (that isn't listening for new connections) checks to see whether it should start listening for new connections. • Type: UInt • Required: false • Default value: "50"
admin_honor_fallback_scsv	Whether or not the admin server, the internal control port and the config daemon honor the Fallback SCSV to protect connections against downgrade attacks. • Type: Boolean • Required: false • Default value: true
admin_ssl3_allow_ rehandshake	Whether or not SSL3/TLS re-handshakes should be supported for admin server and internal connections. • Type: Enum(String) • Required: false • Default value: "rfc5746" • Permitted values: "always": Always allow "never": Never allow "rfc5746": Only if client uses RFC 5746 (Secure Renegotiation Extension)

Property	Description
	"safe": Allow safe re-handshakes
admin_ssl3_ciphers	The SSL ciphers to use for admin server and internal connections. For information on supported ciphers see the online help. • Type: String • Required: false • Default value: <none>
admin_ssl3_diffie_hellman_ key_length	The length in bits of the Diffie-Hellman key for ciphers that use Diffie-Hellman key agreement for admin server and internal connections. • Type: Enum(UInt) • Required: false • Default value: "dh_2048" • Permitted values: "dh_1024": Use 1024 bit keys for Diffie-Hellman ciphers. "dh_2048": Use 2048 bit keys for Diffie-Hellman ciphers. "dh_3072": Use 3072 bit keys for Diffie-Hellman ciphers. "dh_4096": Use 4096 bit keys for Diffie-Hellman ciphers.
admin_ssl3_min_ rehandshake_interval	If SSL3/TLS re-handshakes are supported on the admin server, this defines the minimum time interval (in milliseconds) between handshakes on a single SSL3/TLS connection that is permitted. To disable the minimum interval for handshakes the key should be set to the value 0. • Type: UInt

Property	Description
	• Required: false • Default value: "1000"
admin_ssl_elliptic_curves	The SSL elliptic curve preference list for admin and internal connections. The named curves P256, P384 and P521 may be configured. • Type: List(String) • Required: false • Default value: \<none\>
admin_ssl_insert_extra_ fragment	Whether or not SSL3 and TLS1 use one-byte fragments as a BEAST countermeasure for admin server and internal connections. • Type: Boolean • Required: false • Default value: false
admin_ssl_max_handshake_ message_size	The maximum size (in bytes) of SSL handshake messages that the admin server and internal connections will accept. To accept any size of handshake message the key should be set to the value 0. • Type: UInt • Required: false • Default value: "10240"
admin_ssl_signature_ algorithms	The SSL signature algorithms preference list for admin and internal connections using TLS version 1.2 or higher. For information on supported algorithms see the online help. • Type: String • Required: false • Default value: \<none\>

Property	Description
admin_support_ssl3	Whether or not SSL3 support is enabled for admin server and internal connections. • Type: Boolean • Required: false • Default value: false
admin_support_tls1	Whether or not TLS1.0 support is enabled for admin server and internal connections. • Type: Boolean • Required: false • Default value: true
admin_support_tls1_1	Whether or not TLS1.1 support is enabled for admin server and internal connections. • Type: Boolean • Required: false • Default value: true
admin_support_tls1_2	Whether or not TLS1.2 support is enabled for admin server and internal connections. • Type: Boolean • Required: false • Default value: true
admin_support_tls1_3	Whether or not TLS1.3 support is enabled for admin server and internal connections. • Type: Boolean • Required: false • Default value: true

Property	Description
afm_enabled	Is the application firewall enabled. • Type: Boolean • Required: false • Default value: false
appliance_bootloader_ password	The password used to protect the bootloader. An empty string means there will be no protection. • Type: Password • Required: false • Default value: <none>
appliance_return_path_ routing_enabled	Whether or not the traffic manager will attempt to route response packets back to clients via the same route on which the corresponding request arrived. Note that this applies only to the last hop of the route - the behaviour of upstream routers cannot be altered by the traffic manager. • Type: Boolean • Required: false • Default value: false
aptimizer_max_dependent_ fetch_size	The maximum size of a dependent resource that can undergo Web Accelerator optimization. Any content larger than this size will not be optimized. Units of KB and MB can be used, no postfix denotes bytes. A value of 0 disables the limit. • Type: String • Required: false • Default value: "2MB"

Property	Description
aptimizer_max_original_content_buffer_size	The maximum size of unoptimized content buffered in the traffic manager for a single backend response that is undergoing Web Accelerator optimization. Responses larger than this will not be optimized. Note that if the backend response is compressed then this setting pertains to the compressed size, before Web Accelerator decompresses it. Units of KB and MB can be used, no postfix denotes bytes. Value range is 1 - 128MB. • Type: String • Required: false • Default value: "2MB"
aptimizer_watchdog_interval	The period of time (in seconds) after which a previous failure will no longer count towards the watchdog limit. • Type: UInt • Required: false • Default value: "300"
aptimizer_watchdog_limit	The maximum number of times the Web Accelerator sub-process will be started or restarted within the interval defined by the aptimizer!watchdog_interval setting. If the process fails this many times, it must be restarted manually from the Diagnose page. Zero means no limit. • Type: UInt • Required: false • Default value: "3"
auditlog_via_eventd	Whether to mirror the audit log to EventD. • Type: Boolean • Required: false • Default value: false

Property	Description
auditlog_via_syslog	Whether to output audit log message to the syslog. • Type: Boolean • Required: false • Default value: false
auth_saml_key_lifetime	Lifetime in seconds of cryptographic keys used to decrypt SAML SP sessions stored externally (client-side). • Type: UInt • Required: false • Default value: "86400"
auth_saml_key_rotation_interval	Rotation interval in seconds for cryptographic keys used to encrypt SAML SP sessions stored externally (client-side). • Type: UInt • Required: false • Default value: "14400"
autoscaler_verbose	Whether or not detailed messages about the autoscaler's activity are written to the error log. • Type: Boolean • Required: false • Default value: false
bgp_as_number	The number of the BGP AS in which the traffic manager will operate. Must be entered in decimal. • Type: UInt • Required: false • Default value: "65534"

Property	Description
bgp_enabled	Whether BGP Route Health Injection is enabled • Type: Boolean • Required: false • Default value: false
chunk_size	The default chunk size for reading/writing requests. • Type: UInt • Required: false • Default value: "16384"
client_first_opt	Whether or not your traffic manager should make use of TCP optimisations to defer the processing of new client-first connections until the client has sent some data. • Type: Boolean • Required: false • Default value: false
cluster_comms_allow_update_default	The default value of allow_update for new cluster members. If you have cluster members joining from less trusted locations (such as cloud instances) this can be set to false in order to make them effectively "read-only" cluster members. • Type: Boolean • Required: false • Default value: true
cluster_comms_allowed_update_hosts	The hosts that can contact the internal administration port on each traffic manager. This should be a list containing IP addresses, CIDR IP subnets, and localhost; or it can be set to all to allow any host to connect. • Type: List(String)

Property	Description
	• Required: false • Default value: "all"
cluster_comms_state_sync_interval	How often to propagate the session persistence and bandwidth information to other traffic managers in the same cluster. Set this to 0 (zero) to disable propagation. Note that a cluster using "unicast" heartbeat messages cannot turn off these messages. • Type: UInt • Required: false • Default value: "3"
cluster_comms_state_sync_timeout	The maximum amount of time to wait when propagating session persistence and bandwidth information to other traffic managers in the same cluster. Once this timeout is hit the transfer is aborted and a new connection created. • Type: UInt • Required: false • Default value: "6"
cluster_identifier	Cluster identifier. Generally supplied by Services Director. • Type: String • Required: false • Default value: <none>
connection_idle_connections_max	The maximum number of unused HTTP keepalive connections with back-end nodes that the traffic manager should maintain for re-use. Setting this to 0 (zero) will cause the traffic manager to auto-size this parameter based on the available number of file-descriptors. • Type: UInt • Required: false

Property	Description
	• Default value: \<none>
connection_idle_timeout	How long an unused HTTP keepalive connection should be kept before it is discarded. • Type: UInt • Required: false • Default value: "10"
connection_listen_queue_size	The listen queue size for managing incoming connections. It may be necessary to increase the system's listen queue size if this value is altered. If the value is set to 0 then the default system setting will be used. • Type: UInt • Required: false • Default value: \<none>
connection_max_accepting	Number of processes that should accept new connections. Only this many traffic manager child processes will listen for new connections at any one time. Setting this to 0 (zero) will cause your traffic manager to select an appropriate default value based on the architecture and number of CPUs. • Type: UInt • Required: false • Default value: \<none>
connection_multiple_accept	Whether or not the traffic manager should try to read multiple new connections each time a new client connects. This can improve performance under some very specific conditions. However, in general it is recommended that this be set to 'false'. • Type: Boolean • Required: false

Property	Description
	• Default value: false
connection_udp_read_multiple	Whether or not the traffic manager should try to read multiple UDP packets from clients each time the kernel reports data received from clients. This can improve performance for the situation with high UDP traffic throughput from clients to the traffic manager. Therefore, in general it is recommended that this be set to 'Yes'. • Type: Boolean • Required: false • Default value: true
dns_max_ttl	Maximum Time To Live (expiry time) for entries in the DNS cache. • Type: UInt • Required: false • Default value: "86400"
dns_min_ttl	Minimum Time To Live (expiry time) for entries in the DNS cache. • Type: UInt • Required: false • Default value: "86400"
dns_negative_expiry	Expiry time for failed lookups in the DNS cache. • Type: UInt • Required: false • Default value: "60"
dns_size	Maximum number of entries in the DNS cache. • Type: UInt • Required: false

Property	Description
	• Default value: "10867"
dns_timeout	Timeout for receiving a response from a DNS server. • Type: UInt • Required: false • Default value: "12"
ec2_access_key_id	Deprecated: This key is unused. Amazon authentication credentials are now extracted from IAM Roles assigned to an EC2 instance. • Type: String • Required: false • Default value: <none>
ec2_awstool_timeout	The maximum amount of time requests to the AWS Query API can take before timing out. • Type: UInt • Required: false • Default value: "10"
ec2_metadata_server	URL for the EC2 metadata server, http://169.254.169.254/latest/meta-data for example. • Type: String • Required: false • Default value: <none>
ec2_query_server	URL for the Amazon EC2 endpoint, https://ec2.amazonaws.com/ for example. • Type: String • Required: false

Property	Description
	• Default value: <none>
ec2_secret_access_key	Deprecated: This key is unused. Amazon authentication credentials are now extracted from IAM Roles assigned to an EC2 instance. • Type: Password • Required: false • Default value: <none>
ec2_verify_query_server_cert	Whether to verify Amazon EC2 endpoint's certificate using CA(s) present in SSL Certificate Authorities Catalog. • Type: Boolean • Required: false • Default value: false
eventing_mail_interval	The minimum length of time that must elapse between alert emails being sent. Where multiple alerts occur inside this timeframe, they will be retained and sent within a single email rather than separately. • Type: UInt • Required: false • Default value: "30"
eventing_max_attempts	The number of times to attempt to send an alert email before giving up. • Type: UInt • Required: false • Default value: "10"
fault_tolerance_arp_count	The number of ARP packets a traffic manager should send when an IP address is raised. • Type: UInt

Property	Description
	• Required: false
	• Default value: "10"
fault_tolerance_auto_ failback	Whether or not traffic IPs automatically move back to machines that have recovered from a failure and have dropped their traffic IPs.
	• Type: Boolean
	• Required: false
	• Default value: true
fault_tolerance_ autofailback_delay	Configure the delay of automatic failback after a previous failover event. This setting has no effect if autofailback is disabled.
	• Type: UInt
	• Required: false
	• Default value: "10"
fault_tolerance_child_ timeout	How long the traffic manager should wait for status updates from any of the traffic manager's child processes before assuming one of them is no longer servicing traffic.
	• Type: UInt
	• Required: false
	• Default value: "5"
fault_tolerance_frontend_ check_ips	The IP addresses used to check front-end connectivity. The text %gateway% will be replaced with the default gateway on each system. Set this to an empty string if the traffic manager is on an Intranet with no external connectivity.
	• Type: Set(String)
	• Required: false
	• Default value: "%gateway%"

Property	Description

fault_tolerance_heartbeat_method	The method traffic managers should use to exchange cluster heartbeat messages.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "unicast"
	
	• Permitted values:
	
	    "multicast": multicast
	
	    "unicast": unicast

fault_tolerance_igmp_interval	The interval between unsolicited periodic IGMP Membership Report messages for Multi-Hosted Traffic IP Groups. • Type: UInt • Required: false • Default value: "30"
fault_tolerance_monitor_interval	The frequency, in milliseconds, that each traffic manager machine should check and announce its connectivity. • Type: UInt • Required: false • Default value: "500"
fault_tolerance_monitor_timeout	How long, in seconds, each traffic manager should wait for a response from its connectivity tests or from other traffic manager machines before registering a failure. • Type: UInt • Required: false • Default value: "5"

Property	Description
fault_tolerance_multicast_ address	The multicast address and port to use to exchange cluster heartbeat messages. • Type: String • Required: false • Default value: "239.100.1.1:9090"
fault_tolerance_unicast_port	The unicast UDP port to use to exchange cluster heartbeat messages. • Type: UInt • Required: false • Default value: "9090"
fault_tolerance_use_bind_ip	Whether or not cluster heartbeat messages should only be sent and received over the management network. • Type: Boolean • Required: false • Default value: false
fault_tolerance_verbose	Whether or not a traffic manager should log all connectivity tests. This is very verbose, and should only be used for diagnostic purposes. • Type: Boolean • Required: false • Default value: false
fips_enabled	Enable FIPS Mode (requires software restart). • Type: Boolean • Required: false • Default value: false

Property	Description
ftp_data_bind_low	Whether or not the traffic manager should permit use of FTP data connection source ports lower than 1024. If No the traffic manager can completely drop root privileges, if Yes some or all privileges may be retained in order to bind to low ports. • Type: Boolean • Required: false • Default value: false
glb_verbose	Write a message to the logs for every DNS query that is load balanced, showing the source IP address and the chosen datacenter. • Type: Boolean • Required: false • Default value: false
historical_activity_keep_days	Number of days to store historical traffic information, if set to 0 the data will be kept indefinitely. • Type: UInt • Required: false • Default value: "90"
ip_appliance_returnpath	A table of MAC address/network interface to IP address mappings for each router where return path routing is required. • Type: Table • Required: false • Primary key: • mac (String): The MAC address/network interface of a router the software is connected to. (Required) • Sub keys:

Property	Description
	• ipv4 (String): The MAC address/network interface to IPv4 address mapping of a router the software is connected to. The value is the IPv4 address, the * (asterisk) in the key name is the MAC address and an optional network interface name, for example, 00:50:56:a6:24:3d or 00:50:56:a6:24:3d#eth0. • ipv6 (String): The MAC address/network interface to IPv6 address mapping of a router the software is connected to. The value is the IPv6 address, the * (asterisk) in the key name is the MAC address and an optional network interface name, for example, 00:50:56:a6:24:3d or 00:50:56:a6:24:3d#eth0.
ip_appliance_returnpath_json	A Traffic Manager REST-compatible JSON representation of the "ip_appliance_returnpath" table property. Use this field with the "vtm_global_settings_appliance_returnpath_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
java_classpath	CLASSPATH to use when starting the Java runner. • Type: String • Required: false • Default value: <none>
java_command	Java command to use when starting the Java runner, including any additional options. • Type: String • Required: false • Default value: "java -server"

Property	Description
java_enabled	Whether or not Java support should be enabled. If this is set to No, then your traffic manager will not start any Java processes. Java support is only required if you are using the TrafficScript java.run() function. • Type: Boolean • Required: false • Default value: false
java_lib	Java library directory for additional jar files. The Java runner will load classes from any .jar files stored in this directory, as well as the * jar files and classes stored in traffic manager's catalog. • Type: String • Required: false • Default value: <none>
java_max_connections	Maximum number of simultaneous Java requests. If there are more than this many requests, then further requests will be queued until the earlier requests are completed. This setting is per-CPU, so if your traffic manager is running on a machine with 4 CPU cores, then each core can make this many requests at one time. • Type: UInt • Required: false • Default value: "256"
java_session_age	Default time to keep a Java session. • Type: UInt • Required: false • Default value: "86400"

Property	Description
kerberos_verbose	Whether or not a traffic manager should log all Kerberos related activity. This is very verbose, and should only be used for diagnostic purposes. • Type: Boolean • Required: false • Default value: false
license_servers	A list of license servers for FLA licensing. A license server should be specified as a <ip/host>:<port> pair. • Type: Set(String) • Required: false • Default value: <none>

log_error_level	The minimum severity of events/alerts that should be logged to disk. INFO will log all events; a higher severity setting will log fewer events. More fine-grained control can be achieved using events and actions.
	
	• Type: Enum(UInt)
	
	• Required: false
	
	• Default value: "info"
	
	• Permitted values:
	
	    "fatal": Only fatal errors are logged
	
	    "info": All events are logged to disk
	
	    "serious": Only serious errors or worse
	
	    "warn": Only warnings and errors are logged

log_export_auth_hec_token	The HTTP Event Collector token to use for HTTP authentication with a Splunk server.

Property	Description
	• Type: String • Required: false • Default value: \<none\>

log_export_auth_http	The HTTP authentication method to use when exporting log entries.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "none"
	
	• Permitted values:
	
	      "basic": Basic (Username and Password)
	
	      "none": None
	
	      "splunk": Splunk (HEC token)

log_export_auth_password	The password to use for HTTP basic authentication. • Type: Password • Required: false • Default value: \<none\>
log_export_auth_username	The username to use for HTTP basic authentication. • Type: String • Required: false • Default value: \<none\>
log_export_enabled	Monitor log files and export entries to the configured endpoint. • Type: Boolean • Required: false • Default value: false

Property	Description
log_export_endpoint	The URL to which log entries should be sent. Entries are sent using HTTP(S) POST requests. • Type: String • Required: false • Default value: <none>
log_export_request_timeout	The number of seconds after which HTTP requests sent to the configured endpoint will be considered to have failed if no response is received. A value of 0 means that HTTP requests will not time out. • Type: UInt • Required: false • Default value: "30"
log_export_tls_verify	Whether the server certificate should be verified when connecting to the endpoint. If enabled, server certificates that do not match the server name, are self-signed, have expired, have been revoked, or that are signed by an unknown CA will be rejected. • Type: Boolean • Required: false • Default value: true
log_flush_time	How long to wait before flushing the request log files for each virtual server. • Type: UInt • Required: false • Default value: "5"
log_log_file	The file to log event messages to. • Type: String

Property	Description
	• Required: false • Default value: "%zeushome%/zxtm/log/errors"
log_rate	The maximum number of connection errors logged per second when connection error reporting is enabled. • Type: UInt • Required: false • Default value: "50"
log_reopen	How long to wait before re-opening request log files, this ensures that log files will be recreated in the case of log rotation. • Type: UInt • Required: false • Default value: "30"
log_time	The minimum time between log messages for log intensive features such as SLM. • Type: UInt • Required: false • Default value: "60"
max_fds	The maximum number of file descriptors that your traffic manager will allocate. • Type: UInt • Required: false • Default value: "1048576"

Property	Description
max_tcp_buff_mem	The maximum amount of memory allowed to be used to buffer network data in user space for all TCP connections. The TCP data buffered are either received from clients but before sending to pool nodes, or recevied from pool nodes but before sending to clients. This is specified as either a percentage of system RAM, 5% for example, or an absolute size such as 1024MB and 2GB. A numeric value without suffix MB, GB or % defaults to MB. A value of 800 means 800MB. A value of 0 means unlimited. • Type: String • Required: false • Default value: <none>
monitor_memory_size	The maximum number of each of nodes, pools or locations that can be monitored. The memory used to store information about nodes, pools and locations is allocated at start-up, so the traffic manager must be restarted after changing this setting. • Type: UInt • Required: false • Default value: "4096"
ospfv2_area	The OSPF area in which the traffic manager will operate. May be entered in decimal or IPv4 address format. • Type: String • Required: false • Default value: "0.0.0.1"
ospfv2_area_type	The type of OSPF area in which the traffic manager will operate. This must be the same for all routers in the area, as required by OSPF. • Type: Enum(String) • Required: false

Property	Description

	• Default value: "normal"
	
	• Permitted values:
	
	    "normal": Normal area
	
	    "nssa": Not So Stubby Area (RFC3101)
	
	    "stub": Stub area

ospfv2_authentication_key_id_a	OSPFv2 authentication key ID. If set to 0, which is the default value, the key is disabled. • Type: UInt • Required: false • Default value: <none>
ospfv2_authentication_key_id_b	OSPFv2 authentication key ID. If set to 0, which is the default value, the key is disabled. • Type: UInt • Required: false • Default value: <none>
ospfv2_authentication_shared_secret_a	OSPFv2 authentication shared secret (MD5). If set to blank, which is the default value, the key is disabled. • Type: String • Required: false • Default value: <none>
ospfv2_authentication_shared_secret_b	OSPFv2 authentication shared secret (MD5). If set to blank, which is the default value, the key is disabled. • Type: String • Required: false

Property	Description
	• Default value: \<none\>
ospfv2_dead_interval	The number of seconds before declaring a silent router down. • Type: UInt • Required: false • Default value: "40"
ospfv2_enabled	Whether OSPFv2 Route Health Injection is enabled • Type: Boolean • Required: false • Default value: false
ospfv2_hello_interval	The interval at which OSPF "hello" packets are sent to the network. • Type: UInt • Required: false • Default value: "10"
protection_conncount_size	The amount of shared memory reserved for an inter-process table of combined connection counts, used by all Service Protection classes that have per_process_connection_count set to No. The amount is specified as an absolute size, eg 20MB. • Type: String • Required: false • Default value: "20MB"
rate_class_limit	The maximum number of Rate classes that can be created. Approximately 100 bytes will be pre-allocated per Rate class. • Type: UInt • Required: false

Property	Description
	• Default value: "25000"
recent_connections_max_per_process	How many recently closed connections each traffic manager process should save. These saved connections will be shown alongside currently active connections when viewing the Connections page. You should set this value to 0 in a benchmarking or performance-critical environment. • Type: UInt • Required: false • Default value: "500"
recent_connections_retain_time	The amount of time for which snapshots will be retained on the Connections page. • Type: UInt • Required: false • Default value: "60"
recent_connections_snapshot_size	The maximum number of connections each traffic manager process should show when viewing a snapshot on the Connections page. This value includes both currently active connections and saved connections. If set to 0 all active and saved connection will be displayed on the Connections page. • Type: UInt • Required: false • Default value: "500"
remote_licensing_comm_channel_enabled	Whether to create a Communications Channel agent to send and receive messages from the Services Director Registration Server. This will be disabled when performing self-registration with a Services Director which does not support this feature. • Type: Boolean

Property	Description
	• Required: false • Default value: true
remote_licensing_comm_ channel_port	The port number the Services Director instance is using for access to the traffic manager Communications Channel. • Type: UInt • Required: false • Default value: "8102"
remote_licensing_owner	The Owner of a Services Director instance, used for self-registration. • Type: String • Required: false • Default value: <none>
remote_licensing_owner_ secret	The secret associated with the Owner. • Type: String • Required: false • Default value: <none>
remote_licensing_policy_id	The auto-accept Policy ID that this instance should attempt to use. • Type: String • Required: false • Default value: <none>
remote_licensing_ registration_server	A Services Director address for self-registration. A registration server should be specified as a <ip/host>:<port> pair. • Type: String • Required: false

Property	Description
	• Default value: <none>
remote_licensing_server_certificate	The certificate of a Services Director instance, used for self-registration. • Type: String • Required: false • Default value: <none>
rest_api_auth_timeout	The length of time after a successful request that the authentication of a given username and password will be cached for an IP address. A setting of 0 disables the cache forcing every REST request to be authenticated which will adversely affect performance. • Type: UInt • Required: false • Default value: "120"
rest_api_enabled	Whether or not the REST service is enabled. • Type: Boolean • Required: false • Default value: true
rest_api_http_max_header_length	The maximum allowed length in bytes of a HTTP request's headers. • Type: UInt • Required: false • Default value: "4096"
rest_api_maxfds	Maximum number of file descriptors that the REST API will allocate. The REST API must be restarted for a change to this setting to take effect. • Type: UInt

Property	Description
	• Required: false • Default value: "1048576"
rest_api_replicate_absolute	Configuration changes will be replicated across the cluster after this period of time, regardless of whether additional API requests are being made. • Type: UInt • Required: false • Default value: "20"
rest_api_replicate_lull	Configuration changes made via the REST API will be propagated across the cluster when no further API requests have been made for this period of time. • Type: UInt • Required: false • Default value: "5"
rest_api_replicate_timeout	The period of time after which configuration replication across the cluster will be cancelled if it has not completed. • Type: UInt • Required: false • Default value: "10"
security_login_banner	Banner text displayed on the Admin Server login page and before logging in to appliance SSH servers. • Type: FreeformString • Required: false • Default value: <none>

Property	Description
security_login_banner_accept	Whether or not users must explicitly agree to the displayed login_banner text before logging in to the Admin Server. • Type: Boolean • Required: false • Default value: false
security_login_delay	The number of seconds before another login attempt can be made after a failed attempt. • Type: UInt • Required: false • Default value: "4"
security_max_login_attempts	The number of sequential failed login attempts that will cause a user account to be suspended. Setting this to 0 disables this feature. To apply this to users who have never successfully logged in, track_unknown_users must also be enabled. • Type: UInt • Required: false • Default value: <none>
security_max_login_external	Whether or not usernames blocked due to the max_login_attempts limit should also be blocked from authentication against external services (such as LDAP and RADIUS). • Type: Boolean • Required: false • Default value: false
security_max_login_suspension_time	The number of minutes to suspend users who have exceeded the max_login_attempts limit. • Type: UInt

Property	Description
	• Required: false • Default value: "15"
security_password_allow_ consecutive_chars	Whether or not to allow the same character to appear consecutively in passwords. • Type: Boolean • Required: false • Default value: true
security_password_changes_ per_day	The maximum number of times a password can be changed in a 24-hour period. Set to 0 to disable this restriction. • Type: UInt • Required: false • Default value: <none>
security_password_min_ alpha_chars	Minimum number of alphabetic characters a password must contain. Set to 0 to disable this restriction. • Type: UInt • Required: false • Default value: <none>
security_password_min_ length	Minimum number of characters a password must contain. Set to 0 to disable this restriction. • Type: UInt • Required: false • Default value: <none>
security_password_min_ numeric_chars	Minimum number of numeric characters a password must contain. Set to 0 to disable this restriction. • Type: UInt

Property	Description
	• Required: false • Default value: <none>
security_password_min_ special_chars	Minimum number of special (non-alphanumeric) characters a password must contain. Set to 0 to disable this restriction. • Type: UInt • Required: false • Default value: <none>
security_password_min_ uppercase_chars	Minimum number of uppercase characters a password must contain. Set to 0 to disable this restriction. • Type: UInt • Required: false • Default value: <none>
security_password_reuse_ after	The number of times a password must have been changed before it can be reused. Set to 0 to disable this restriction. • Type: UInt • Required: false • Default value: <none>
security_post_login_banner	Banner text to be displayed on the appliance console after login. • Type: String • Required: false • Default value: <none>
security_track_unknown_ users	Whether to remember past login attempts from usernames that are not known to exist (should be set to false for an Admin Server accessible from the public Internet). This does not affect the audit log.

Property	Description
	• Type: Boolean • Required: false • Default value: false
security_ui_page_banner	Banner text to be displayed on all Admin Server pages. • Type: String • Required: false • Default value: \<none\>
session_asp_cache_size	The maximum number of entries in the ASP session persistence cache. This is used for storing session mappings for ASP session persistence. Approximately 100 bytes will be pre-allocated per entry. • Type: UInt • Required: false • Default value: "32768"
session_ip_cache_expiry	IP session persistence cache expiry time in seconds. A session will not be reused if the time since it was last used exceeds this value. 0 indicates no expiry timeout. • Type: UInt • Required: false • Default value: \<none\>
session_ip_cache_size	The maximum number of entries in the IP session persistence cache. This is used to provide session persistence based on the source IP address. Approximately 100 bytes will be pre-allocated per entry. • Type: UInt • Required: false • Default value: "32768"

Property	Description
session_j2ee_cache_expiry	J2EE session persistence cache expiry time in seconds. A session will not be reused if the time since it was last used exceeds this value. 0 indicates no expiry timeout. • Type: UInt • Required: false • Default value: <none>
session_j2ee_cache_size	The maximum number of entries in the J2EE session persistence cache. This is used for storing session mappings for J2EE session persistence. Approximately 100 bytes will be pre-allocated per entry. • Type: UInt • Required: false • Default value: "32768"
session_ssl_cache_size	The maximum number of entries in the SSL session persistence cache. This is used to provide session persistence based on the SSL session ID. Approximately 200 bytes will be pre-allocated per entry. • Type: UInt • Required: false • Default value: "32768"
session_universal_cache_expiry	Universal session persistence cache expiry time in seconds. A session will not be reused if the time since it was last used exceeds this value. 0 indicates no expiry timeout. • Type: UInt • Required: false • Default value: <none>

Property	Description
session_universal_cache_size	The maximum number of entries in the global universal session persistence cache. This is used for storing session mappings for universal session persistence. Approximately 100 bytes will be pre-allocated per entry. • Type: UInt • Required: false • Default value: "32768"
shared_pool_size	The size of the shared memory pool used for shared storage across worker processes (e.g. bandwidth shared data).This is specified as either a percentage of system RAM, 5% for example, or an absolute size such as 10MB. • Type: String • Required: false • Default value: "10MB"
slm_class_limit	The maximum number of SLM classes that can be created. Approximately 100 bytes will be pre-allocated per SLM class. • Type: UInt • Required: false • Default value: "1024"
snmp_user_counters	The number of user defined SNMP counters. Approximately 100 bytes will be pre-allocated at start-up per user defined SNMP counter. • Type: UInt • Required: false • Default value: "10"

Property	Description
so_rbuff_size	The size of the operating system's read buffer. A value of 0 (zero) means to use the OS default; in normal circumstances this is what should be used. • Type: UInt • Required: false • Default value: <none>
so_wbuff_size	The size of the operating system's write buffer. A value of 0 (zero) means to use the OS default; in normal circumstances this is what should be used. • Type: UInt • Required: false • Default value: <none>
soap_idle_minutes	The number of minutes that the SOAP server should remain idle before exiting. The SOAP server has a short startup delay the first time a SOAP request is made, subsequent SOAP requests don't have this delay. • Type: UInt • Required: false • Default value: "10"
socket_optimizations	Whether or not the traffic manager should use potential network socket optimisations. If set to auto, a decision will be made based on the host platform. • Type: Enum(String) • Required: false • Default value: "auto" • Permitted values:

Property	Description
	"auto": Decide based on local platform "no": Disable socket optimizations "yes": Enable socket optimizations
ssl_allow_rehandshake	Whether or not SSL/TLS re-handshakes should be supported. Enabling support for re-handshakes can expose services to Man-in-the-Middle attacks. It is recommended that only "safe" handshakes be permitted, or none at all. • Type: Enum(String) • Required: false • Default value: "safe" • Permitted values: "always": Always allow "never": Never allow "rfc5746": Only if client uses RFC 5746 (Secure Renegotiation Extension) "safe": Allow safe re-handshakes
ssl_cache_enabled	Whether or not the SSL server session cache is enabled, unless overridden by virtual server settings. • Type: Boolean • Required: false • Default value: true
ssl_cache_expiry	How long the SSL session IDs for SSL decryption should be stored for. • Type: UInt • Required: false

Property	Description
	• Default value: "1800"
ssl_cache_per_virtualserver	Whether an SSL session created by a given virtual server can only be resumed by a connection to the same virtual server. • Type: Boolean • Required: false • Default value: true
ssl_cache_size	How many entries the SSL session ID cache should hold. This cache is used to cache SSL sessions to help speed up SSL handshakes when performing SSL decryption. Each entry will allocate approximately 1.75kB of metadata. • Type: UInt • Required: false • Default value: "6151"
ssl_cipher_suites	The SSL/TLS cipher suites preference list for SSL/TLS connections, unless overridden by virtual server or pool settings. For information on supported cipher suites see the online help. • Type: String • Required: false • Default value: <none>
ssl_client_cache_enabled	Whether or the SSL client cache will be used, unless overridden by pool settings. • Type: Boolean • Required: false • Default value: true

Property	Description
ssl_client_cache_expiry	How long in seconds SSL sessions should be stored in the client cache for, by default. Servers returning session tickets may also provide a lifetime hint, which will be used if it is less than this value. • Type: UInt • Required: false • Default value: "14400"
ssl_client_cache_size	How many entries the SSL client session cache should hold, per child. This cache is used to cache SSL sessions to help speed up SSL handshakes when performing SSL encryption. Each entry will require approx 100 bytes of memory plus space for either an SSL session id or an SSL session ticket, which may be as small as 16 bytes or may be as large as a few kilobytes, depending upon the server behavior. • Type: UInt • Required: false • Default value: "1024"
ssl_client_cache_tickets_enabled	Whether or not session tickets, including TLS >= 1.3 PSKs, may be requested and stored in the SSL client cache. • Type: Boolean • Required: false • Default value: true
ssl_crl_mem_size	How much shared memory to allocate for loading Certificate Revocation Lists. This should be at least 3 times the total size of all CRLs on disk. This is specified as either a percentage of system RAM, 1% for example, or an absolute size such as 10MB. • Type: String • Required: false • Default value: "5MB"

Property	Description

ssl_diffie_hellman_modulus_size	The size in bits of the modulus for the domain parameters used for cipher suites that use finite field Diffie-Hellman key agreement.
	
	• Type: Enum(UInt)
	
	• Required: false
	
	• Default value: "dh_2048"
	
	• Permitted values:
	
	    "dh_1024": 1024 bit modulus
	
	    "dh_2048": 2048 bit modulus
	
	    "dh_3072": 3072 bit modulus
	
	    "dh_4096": 4096 bit modulus

ssl_elliptic_curves	The SSL/TLS elliptic curve preference list for SSL/TLS connections using TLS version 1.0 or higher, unless overridden by virtual server or pool settings. For information on supported curves see the online help. • Type: List(String) • Required: false • Default value: <none>
ssl_hardware_accel	Whether or not the SSL hardware is an "accelerator" (faster than software). By default the traffic manager will only use the SSL hardware if a key requires it (i.e. the key is stored on secure hardware and the traffic manager only has a placeholder/identifier key). With this option enabled, your traffic manager will instead try to use hardware for all SSL decrypts. • Type: Boolean • Required: false • Default value: false

Property	Description
ssl_hardware_azure_client_id	The client identifier used when accessing the Microsoft Azure Key Vault. • Type: String • Required: false • Default value: <none>
ssl_hardware_azure_client_secret	The client secret used when accessing the Microsoft Azure Key Vault. • Type: Password • Required: false • Default value: <none>
ssl_hardware_azure_vault_url	The URL for the REST API of the Microsoft Azure Key Vault. • Type: String • Required: false • Default value: <none>
ssl_hardware_azure_verify_rest_api_cert	Whether or not the Azure Key Vault REST API certificate should be verified. • Type: Boolean • Required: false • Default value: true
ssl_hardware_driver_pkcs11_debug	Print verbose information about the PKCS11 hardware security module to the event log. • Type: Boolean • Required: false • Default value: false

Property	Description
ssl_hardware_driver_pkcs11_lib	The location of the PKCS#11 library for your SSL hardware if it is not in a standard location. The traffic manager will search the standard locations by default. • Type: String • Required: false • Default value: <none>
ssl_hardware_driver_pkcs11_slot_desc	The label of the SSL Hardware slot to use. Only required if you have multiple HW accelerator slots. • Type: String • Required: false • Default value: <none>

ssl_hardware_driver_pkcs11_slot_type	The type of SSL hardware slot to use.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "operator"
	
	• Permitted values:
	
	    "module": Module Protected
	
	    "operator": Operator Card Set
	
	    "softcard": Soft Card

ssl_hardware_driver_pkcs11_user_pin	The User PIN for the PKCS token (PKCS#11 devices only). • Type: Password • Required: false • Default value: <none>

Property	Description
ssl_hardware_failure_count	The number of consecutive failures from the SSL hardware that will be tolerated before the traffic manager assumes its session with the device is invalid and tries to log in again. This is necessary when the device reboots following a power failure. • Type: UInt • Required: false • Default value: "5"

ssl_hardware_library	The type of SSL hardware to use. The drivers for the SSL hardware should be installed and accessible to the traffic manager software.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "none"
	
	• Permitted values:
	
	    "azure": Microsoft Azure Key Vault
	
	    "none": None
	
	    "pkcs11": PKCS#11

ssl_honor_fallback_scsv	Whether or not ssl-decrypting Virtual Servers honor the Fallback SCSV to protect connections against downgrade attacks. • Type: Boolean • Required: false • Default value: true
ssl_insert_extra_fragment	Whether or not SSL3 and TLS1 use one-byte fragments as a BEAST countermeasure. • Type: Boolean • Required: false

Property	Description
	• Default value: false
ssl_log_keys	Whether SSL connection key logging should be available via the ssl.sslkeylogline() TrafficScript function. If this setting is disabled then ssl.sslkeylogline() will always return the empty string. • Type: Boolean • Required: false • Default value: false
ssl_max_handshake_message_size	The maximum size (in bytes) of SSL handshake messages that SSL connections will accept. To accept any size of handshake message the key should be set to the value 0. • Type: UInt • Required: false • Default value: "10240"
ssl_middlebox_compatibility	Whether or not TLS 1.3 middlebox compatibility mode as described in RFC 8446 appendix D.4 will be used in connections to pool nodes, unless overridden by pool settings. • Type: Boolean • Required: false • Default value: true
ssl_min_rehandshake_interval	If SSL3/TLS re-handshakes are supported, this defines the minimum time interval (in milliseconds) between handshakes on a single SSL3/TLS connection that is permitted. To disable the minimum interval for handshakes the key should be set to the value 0. • Type: UInt • Required: false • Default value: "1000"

Property	Description
ssl_ocsp_cache_size	The maximum number of cached client certificate OCSP results stored. This cache is used to speed up OCSP checks against client certificates by caching results. Approximately 1040 bytes are pre-allocated per entry. • Type: UInt • Required: false • Default value: "2048"
ssl_ocsp_stapling_default_refresh_interval	How long to wait before refreshing requests on behalf of the store of certificate status responses used by OCSP stapling, if we don't have an up-to-date OCSP response. • Type: UInt • Required: false • Default value: "60"
ssl_ocsp_stapling_maximum_refresh_interval	Maximum time to wait before refreshing requests on behalf of the store of certificate status responses used by OCSP stapling. (0 means no maximum.) • Type: UInt • Required: false • Default value: "864000"
ssl_ocsp_stapling_mem_size	How much shared memory to allocate for the store of certificate status responses for OCSP stapling. This should be at least 2kB times the number of certificates configured to use OCSP stapling. This is specified as either a percentage of system RAM, 1% for example, or an absolute size such as 10MB. • Type: String • Required: false • Default value: "1MB"

Property	Description
ssl_ocsp_stapling_time_tolerance	How many seconds to allow the current time to be outside the validity time of an OCSP response before considering it invalid. • Type: UInt • Required: false • Default value: "30"
ssl_ocsp_stapling_verify_response	Whether the OCSP response signature should be verified before the OCSP response is cached. • Type: Boolean • Required: false • Default value: false
ssl_signature_algorithms	The SSL/TLS signature algorithms preference list for SSL/TLS connections using TLS version 1.2 or higher, unless overridden by virtual server or pool settings. For information on supported algorithms see the online help. • Type: String • Required: false • Default value: <none>
ssl_support_ssl3	Whether or not SSL3 support is enabled. • Type: Boolean • Required: false • Default value: false
ssl_support_tls1	Whether or not TLS1.0 support is enabled. • Type: Boolean • Required: false • Default value: true

Property	Description
ssl_support_tls1_1	Whether or not TLS1.1 support is enabled. • Type: Boolean • Required: false • Default value: true
ssl_support_tls1_2	Whether or not TLS1.2 support is enabled. • Type: Boolean • Required: false • Default value: true
ssl_support_tls1_3	Whether or not TLS1.3 support is enabled. • Type: Boolean • Required: false • Default value: true
ssl_tickets_enabled	Whether or not session tickets will be issued to and accepted from clients that support them, unless overridden by virtual server settings. • Type: Boolean • Required: false • Default value: true
ssl_tickets_reissue_policy	When an SSL session ticket will be reissued (ie when a new ticket will be generated for the same SSL session). • Type: Enum(String) • Required: false • Default value: "never" • Permitted values:

Property	Description
	"always": always "never": never
ssl_tickets_ticket_expiry	The length of time for which an SSL session ticket will be accepted by a virtual server after the ticket is created. If a ticket is reissued (if ssl!tickets!reissue_policy is set to 'always') this time starts at the time when the ticket was reissued. • Type: UInt • Required: false • Default value: "14400"
ssl_tickets_ticket_key_expiry	The length of time for which an auto-generated SSL ticket key will be used to decrypt old session ticket, before being deleted from memory. This setting is ignored if there are any entries in the (REST-only) SSL ticket keys catalog. • Type: UInt • Required: false • Default value: "86400"
ssl_tickets_ticket_key_ rotation	The length of time for which an auto-generated SSL ticket key will be used to encrypt new session tickets, before a new SSL ticket key is generated. The ticket encryption key will be held in memory for ssl!tickets!ticket_key_expiry, so that tickets encrypted using the key can still be decrypted and used. This setting is ignored if there are any entries in the (REST-only) SSL ticket keys catalog. • Type: UInt • Required: false • Default value: "14400"
ssl_tickets_time_tolerance	How many seconds to allow the current time to be outside the validity time of an SSL ticket before considering it invalid.

Property	Description
	• Type: UInt • Required: false • Default value: "30"
ssl_validate_server_ certificates_catalog	Whether the traffic manager should validate that SSL server certificates form a matching key pair before the certificate gets used on an SSL decrypting virtual server. • Type: Boolean • Required: false • Default value: true
telemetry_enabled	Allow the reporting of anonymized usage data for product improvement and customer support purposes. • Type: Boolean • Required: false • Default value: true
tip_class_limit	The maximum number of Traffic IP Groups that can be created. • Type: UInt • Required: false • Default value: "10000"
trafficscript_data_local_size	The maximum amount of memory available to store TrafficScript data.local.set() information. This can be specified as a percentage of system RAM, 5% for example; or an absolute size such as 200MB. • Type: String • Required: false • Default value: "5%"

Property	Description
trafficscript_data_size	The maximum amount of memory available to store TrafficScript data.set() information. This can be specified as a percentage of system RAM, 5% for example; or an absolute size such as 200MB. • Type: String • Required: false • Default value: "5%"
trafficscript_execution_time_warning	Raise an event if a TrafficScript rule runs for more than this number of milliseconds in a single invocation. If you get such events repeatedly, you may want to consider re-working some of your TrafficScript rules. A value of 0 means no warnings will be issued. • Type: UInt • Required: false • Default value: "500"
trafficscript_max_instr	The maximum number of instructions a TrafficScript rule will run. A rule will be aborted if it runs more than this number of instructions without yielding, preventing infinite loops. • Type: UInt • Required: false • Default value: "100000"
trafficscript_memory_warning	Raise an event if a TrafficScript rule requires more than this amount of buffered network data. If you get such events repeatedly, you may want to consider re-working some of your TrafficScript rules to use less memory or to stream the data that they process rather than storing it all in memory. This setting also limits the amount of data that can be returned by request.GetLine(). • Type: UInt • Required: false

Property	Description
	• Default value: "1048576"
trafficscript_regex_cache_size	The maximum number of regular expressions to cache in TrafficScript. Regular expressions will be compiled in order to speed up their use in the future. • Type: UInt • Required: false • Default value: "57"
trafficscript_regex_match_limit	The maximum number of ways TrafficScript will attempt to match a regular expression at each position in the subject string, before it aborts the rule and reports a TrafficScript error. • Type: UInt • Required: false • Default value: "10000000"
trafficscript_regex_match_warn_percentage	The percentage of regex_match_limit at which TrafficScript reports a performance warning. • Type: UInt • Required: false • Default value: "5"
trafficscript_variable_pool_use	Allow the pool.use and pool.select TrafficScript functions to accept variables instead of requiring literal strings. Enabling this feature has the following effects1. Your traffic manager may no longer be able to know whether a pool is in use.2. Errors for pools that aren't in use will not be hidden.3. Some settings displayed for a Pool may not be appropriate for the type of traffic being managed.4. Pool usage information on the pool edit pages and config summary may not be accurate.5. Monitors will run for all pools (with this option disabled monitors will only run for Pools that are used). • Type: Boolean

Property	Description
	• Required: false • Default value: false
transaction_export_enabled	Export metadata about transactions processed by the traffic manager to an external location. • Type: Boolean • Required: false • Default value: false
transaction_export_ endpoint	The endpoint to which transaction metadata should be exported. The endpoint is specified as a hostname or IP address with a port. • Type: String • Required: false • Default value: <none>
transaction_export_tls	Whether the connection to the specified endpoint should be encrypted. • Type: Boolean • Required: false • Default value: true
transaction_export_tls_verify	Whether the server certificate presented by the endpoint should be verified, preventing a connection from being established if the certificate does not match the server name, is self-signed, is expired, is revoked, or has an unknown CA. • Type: Boolean • Required: false • Default value: true

Property	Description
watchdog_timeout	The maximum time in seconds a process can fail to update its heartbeat, before the watchdog considers it to have stalled. • Type: UInt • Required: false • Default value: "5"
web_cache_avg_path_length	The estimated average length of the path (including query string) for resources being cached. An amount of memory equal to this figure multiplied by max_file_num will be allocated for storing the paths for cache entries. This setting can be increased if your web site makes extensive use of long URLs. • Type: UInt • Required: false • Default value: "512"
web_cache_disk	Whether or not to use a disk-backed (typically SSD) cache. If set to Yes cached web pages will be stored in a file on disk. This enables the traffic manager to use a cache that is larger than available RAM. The size setting should also be adjusted to select a suitable maximum size based on your disk space. Note that the disk caching is optimized for use with SSD storage. • Type: Boolean • Required: false • Default value: false
web_cache_disk_dir	If disk caching is enabled, this sets the directory where the disk cache file will be stored. The traffic manager will create a file called webcache.data in this location. Note that the disk caching is optimized for use with SSD storage. • Type: String

Property	Description
	• Required: false • Default value: "%zeushome%/zxtm/internal"
web_cache_max_file_num	Maximum number of entries in the cache. Approximately 0.9 KB will be pre-allocated per entry for metadata, this is in addition to the memory reserved for the content cache and for storing the paths of the cached resources. • Type: UInt • Required: false • Default value: "10000"
web_cache_max_file_size	Largest size of a cacheable object in the cache. This is specified as either a percentage of the total cache size, 2% for example, or an absolute size such as 20MB. • Type: String • Required: false • Default value: "2%"
web_cache_max_path_length	The maximum length of the path (including query string) for the resource being cached. If the path exceeds this length then it will not be added to the cache. • Type: UInt • Required: false • Default value: "2048"
web_cache_normalize_query	Enable normalization (lexical ordering of the parameter-assignments) of the query string. • Type: Boolean • Required: false • Default value: true

Property	Description
web_cache_size	The maximum size of the HTTP web page cache. This is specified as either a percentage of system RAM, 20% for example, or an absolute size such as 200MB. • Type: String • Required: false • Default value: "20%"
web_cache_verbose	Add an X-Cache-Info header to every HTTP response, showing whether the request and/or the response was cacheable. • Type: Boolean • Required: false • Default value: false

vtm_kerberos_krb5conf

URI Endpoint: /api/tm/8.3/config/active/kerberos/krb5confs

A Kerberos krb5.conf file that provides the raw configuration for a Kerberos principal.

```
# Example usage of vtm_kerberos_krb5conf resource...
resource "vtm_kerberos_krb5conf" "my_kerberos_krb5conf" {
    content = ""
    name = "MyKerberosKrb5Conf"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
content	Object content • Type: string

Property	Description
	• Required: true

vtm_kerberos_keytab

URI Endpoint: /api/tm/8.3/config/active/kerberos/keytabs

A Kerberos keytab file contains credentials to authenticate as (a number of) Kerberos principals.

```
# Example usage of vtm_kerberos_keytab resource...
resource "vtm_kerberos_keytab" "my_kerberos_keytab" {
    content = ""
    name = "MyKerberosKeytab"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
content	Object content • Type: string • Required: true

vtm_kerberos_principal

URI Endpoint: /api/tm/8.3/config/active/kerberos/principals

A Kerberos principal can be used by the traffic manager to participate in a Kerberos realm.

```
# Example usage of vtm_kerberos_principal resource...
resource "vtm_kerberos_principal" "my_kerberos_principal" {
    keytab =
    name = "MyKerberosPrincipal"
    service =
}
```

Property	Description
name	Name of the object • Type: string • Required: true
kdcs	A list of <hostname/ip>:<port> pairs for Kerberos key distribution center (KDC) services to be explicitly used for the realm of the principal. If no KDCs are explicitly configured, DNS will be used to discover the KDC(s) to use. • Type: List(String) • Required: false • Default value: <none>
keytab	The name of the Kerberos keytab file containing suitable credentials to authenticate as the specified Kerberos principal. • Type: String • Required: true • Default value: <none>
krb5conf	The name of an optional Kerberos configuration file (krb5.conf). • Type: String • Required: false • Default value: <none>
realm	The Kerberos realm where the principal belongs. • Type: String • Required: false • Default value: <none>
service	The service name part of the Kerberos principal name the traffic manager should use to authenticate itself.

Property	Description
	• Type: String • Required: true • Default value: <none>

vtm_license_key

URI Endpoint: /api/tm/8.3/config/active/license_keys

A license key is an encoded text file that controls what functionality is available from each traffic manager in the cluster. Every production traffic manager must have a valid licence key in order to function; a traffic manager without a license will operate as Community Edition, which provides most of the functionality, but places restrictions on bandwidth and cluster size.

```
# Example usage of vtm_license_key resource...
resource "vtm_license_key" "my_license_key" {
    content = ""
    name = "MyLicenseKey"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
content	Object content • Type: string • Required: true

vtm_location

URI Endpoint: /api/tm/8.3/config/active/locations

These are geographic locations as used by Global Load Balancing services. Such a location may not necessarily contain a traffic manager; instead it could refer to the location of a remote datacenter.

```
# Example usage of vtm_location resource...
resource "vtm_location" "my_location" {
    id =
    name = "MyLocation"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
identifier	The identifier of this location. • Type: UInt • Required: true • Default value: <none>
latitude	The latitude of this location. • Type: Float • Required: false • Default value: "0.0"
longitude	The longitude of this location. • Type: Float • Required: false • Default value: "0.0"
note	A note, used to describe this location. • Type: FreeformString • Required: false • Default value: <none>

Property	Description

type	Does this location contain traffic managers and configuration or is it a recipient of GLB requests?
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "config"
	
	• Permitted values:
	
	    "config": Configuration
	
	    "glb": GLB

vtm_log_export

URI Endpoint: /api/tm/8.3/config/active/log_export

Definitions of log files which should be exported to the analytics engine

```
# Example usage of vtm_log_export resource...
resource "vtm_log_export" "my_log_export" {
    name = "MyLogExport"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
appliance_only	Whether entries from the specified log files should be exported only from appliances. • Type: Boolean • Required: false • Default value: false

Property	Description
enabled	Export entries from the log files included in this category. • Type: Boolean • Required: false • Default value: false
files	The set of files to export as part of this category, specified as a list of glob patterns. • Type: Set(String) • Required: false • Default value: <none>
history	How much historic log activity should be exported. • Type: Enum(String) • Required: false • Default value: "none" • Permitted values: "all": Export all historic entries "none": Do not export any historic entries "recent": Export recent historic entries, according to the 'history_period' setting
history_period	The number of days of historic log entries that should be exported. • Type: UInt • Required: false • Default value: "10"
metadata	This is table 'metadata'

Property	Description

	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • name (String): The name of a metadata item which should be sent to the analytics engine along with entries from these log files. (Required)
	
	• Sub keys:
	
	    • value (String): Additional metadata to include with the log entries when exporting them to the configured endpoint. Metadata can be used by the system that is receiving the exported data to categorise and parse the log entries. (Required)

metadata_json	A Traffic Manager REST-compatible JSON representation of the "metadata" table property. Use this field with the "vtm_log_export_metadata_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
note	A description of this category of log files. • Type: String • Required: false • Default value: <none>

vtm_monitor

URI Endpoint: /api/tm/8.3/config/active/monitors

Monitors check important remote services are running, by periodically sending them traffic and checking the response is correct. They are used by virtual servers to detect the failure of backend nodes.

```
# Example usage of vtm_monitor resource...
resource "vtm_monitor" "my_monitor" {
    name = "MyMonitor"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
back_off	Should the monitor slowly increase the delay after it has failed? • Type: Boolean • Required: false • Default value: true
delay	The minimum time between calls to a monitor. • Type: UInt • Required: false • Default value: "3"
failures	The number of times in a row that a node must fail execution of the monitor before it is classed as unavailable. • Type: UInt • Required: false • Default value: "3"
health_only	Should this monitor only report health (ignore load)? • Type: Boolean • Required: false

Property	Description
	• Default value: false
http_authentication	The HTTP basic-auth <user>:<password> to use for the test HTTP request. • Type: String • Required: false • Default value: <none>
http_body_regex	A regular expression that the HTTP response body must match. If the response body content doesn't matter then set this to .* (match anything). • Type: String • Required: false • Default value: <none>
http_host_header	The host header to use in the test HTTP request. • Type: String • Required: false • Default value: <none>
http_path	The path to use in the test HTTP request. This must be a string beginning with a / (forward slash). • Type: String • Required: false • Default value: "/"
http_status_regex	A regular expression that the HTTP status code must match. If the status code doesn't matter then set this to .* (match anything). • Type: String • Required: false

Property	Description
	• Default value: "^[234][0-9][0-9]$"
machine	The machine to monitor, where relevant this should be in the form <hostname>:<port>, for "ping" monitors the :<port> part must not be specified. • Type: String • Required: false • Default value: <none>
note	A description of the monitor. • Type: FreeformString • Required: false • Default value: <none>
rtsp_body_regex	The regular expression that the RTSP response body must match. • Type: String • Required: false • Default value: <none>
rtsp_path	The path to use in the RTSP request (some servers will return 500 Internal Server Error unless this is a valid media file). • Type: String • Required: false • Default value: "/"
rtsp_status_regex	The regular expression that the RTSP response status code must match. • Type: String • Required: false

Property	Description
	• Default value: "^[234][0-9][0-9]$"

scope	A monitor can either monitor each node in the pool separately and disable an individual node if it fails, or it can monitor a specific machine and disable the entire pool if that machine fails. GLB location monitors must monitor a specific machine.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "pernode"
	
	• Permitted values:
	
	    "pernode": Node: Monitor each node in the pool separately
	
	    "poolwide": Pool/GLB: Monitor a specified machine

script_arguments	A table containing arguments and argument values to be passed to the monitor program.
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • name (String): The name of the argument to be passed to the monitor program. (Required)
	
	• Sub keys:
	
	    • value (String): The value of the argument to be passed to the monitor program. (Required)
	
	    • description (String): A description for the argument provided to the program.

Property	Description
script_arguments_json	A Traffic Manager REST-compatible JSON representation of the "script_arguments" table property. Use this field with the "vtm_monitor_arguments_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
script_program	The program to run. This must be an executable file, either within the monitor scripts directory or specified as an absolute path to some other location on the filesystem. • Type: String • Required: false • Default value: <none>
sip_body_regex	The regular expression that the SIP response body must match. • Type: String • Required: false • Default value: <none>
sip_status_regex	The regular expression that the SIP response status code must match. • Type: String • Required: false • Default value: "^[234][0-9][0-9]$"
sip_transport	Which transport protocol the SIP monitor will use to query the server. • Type: Enum(String)

Property	Description

	• Required: false
	
	 • Default value: "udp"
	
	 • Permitted values:
	
	    "tcp": TCP
	
	    "udp": UDP

tcp_close_string	An optional string to write to the server before closing the connection. • Type: String • Required: false • Default value: \<none>
tcp_max_response_len	The maximum amount of data to read back from a server, use 0 for unlimited. Applies to TCP and HTTP monitors. • Type: UInt • Required: false • Default value: "2048"
tcp_response_regex	A regular expression to match against the response from the server. Applies to TCP monitors only. • Type: String • Required: false • Default value: ".+"
tcp_write_string	The string to write down the TCP connection. • Type: String • Required: false • Default value: \<none>

Property	Description
timeout	The maximum runtime for an individual instance of the monitor. • Type: UInt • Required: false • Default value: "3"
type	The internal monitor implementation of this monitor. • Type: Enum(String) • Required: false • Default value: "ping" • Permitted values: "connect": TCP Connect monitor "http": HTTP monitor "ping": Ping monitor "program": External program monitor "rtsp": RTSP monitor "sip": SIP monitor "tcp_transaction": TCP transaction monitor
udp_accept_all	If this monitor uses UDP, should it accept responses from any IP and port? • Type: Boolean • Required: false • Default value: false
use_ssl	Whether or not the monitor should connect using SSL. • Type: Boolean

Property	Description
	• Required: false • Default value: false
verbose	Whether or not the monitor should emit verbose logging. This is useful for diagnosing problems. • Type: Boolean • Required: false • Default value: false

vtm_monitor_script

URI Endpoint: /api/tm/8.3/config/active/monitor_scripts

An executable program that can be used to by external program monitors to report the health of backend services.

```
# Example usage of vtm_monitor_script resource...
resource "vtm_monitor_script" "my_monitor_script" {
    content = ""
    name = "MyMonitorScript"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
content	Object content • Type: string • Required: true

vtm_appliance_nat

URI Endpoint: /api/tm/8.3/config/active/appliance/nat

The NAT configuration file stores rules controlling NAT on an appliance.

```
# Example usage of vtm_appliance_nat resource...
resource "vtm_appliance_nat" "my_appliance_nat" {

}
```

Property	Description

many_to_one_all_ports	This is table 'many_to_one_all_ports'
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	   • rule_number (String): A unique rule identifier (Required)
	
	• Sub keys:
	
	   • pool (String): Pool of a "many to one overload" type NAT rule. (Required)
	
	   • tip (String): TIP Group of a "many to one overload" type NAT rule. (Required)

many_to_one_all_ports_json	A Traffic Manager REST-compatible JSON representation of the "many_to_one_all_ports" table property. Use this field with the "vtm_appliance_nat_many_to_one_all_ports_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
many_to_one_port_locked	This is table 'many_to_one_port_locked' • Type: Table

Property	Description

	• Required: false
	
	• Primary key:
	
	    • rule_number (String): A unique rule identifier (Required)
	
	• Sub keys:
	
	    • pool (String): Pool of a "many to one port locked" type NAT rule. (Required)
	
	    • port (UInt): Port number of a "many to one port locked" type NAT rule. (Required)
	
	    • protocol (Enum(String)): Protocol of a "many to one port locked" type NAT rule. (Required)
	
	    Permitted values:
	
	"icmp": ICMP
	
	"sctp": SCTP
	
	"tcp": TCP
	
	"udp": UDP
	
	"udplite": UDPLITE
	
	    • tip (String): TIP Group of a "many to one port locked" type NAT rule. (Required)

many_to_one_port_locked_json	A Traffic Manager REST-compatible JSON representation of the "many_to_one_port_locked" table property. Use this field with the "vtm_appliance_nat_many_to_one_port_locked_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: \<none\>

Property	Description

one_to_one	This is table 'one_to_one'
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • rule_number (String): A unique rule identifier (Required)
	
	• Sub keys:
	
	    • enable_inbound (Boolean): Enabling the inbound part of a "one to one" type NAT rule. (Required)
	
	    • ip (String): IP Address of a "one to one" type NAT rule. (Required)
	
	    • tip (String): TIP group of a "one to one" type NAT rule. (Required)

one_to_one_json	A Traffic Manager REST-compatible JSON representation of the "one_to_one" table property. Use this field with the "vtm_appliance_ nat_one_to_one_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
port_mapping	This is table 'port_mapping' • Type: Table • Required: false • Primary key: • rule_number (String): A unique rule identifier (Required) • Sub keys:

Property	Description
	• dport_first (UInt): First port of the dest. port range of a "port mapping" rule. (Required) • dport_last (UInt): Last port of the dest. port range of a "port mapping" rule. (Required) • virtual_server (String): Target Virtual Server of a "port mapping" rule. (Required)
port_mapping_json	A Traffic Manager REST-compatible JSON representation of the "port_mapping" table property. Use this field with the "vtm_appliance_nat_port_mapping_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>

vtm_pool

URI Endpoint: /api/tm/8.3/config/active/pools

The conf/pools directory contains configuration files for backend node pools. The name of a file is the name of the pool it defines. Pools can be configured under the Services > Pools section of the Admin Server UI or by using functions under the Pool section of the SOAP API and CLI.

```
# Example usage of vtm_pool resource...
resource "vtm_pool" "my_pool" {
    name = "MyPool"
}
```

Property	Description
name	Name of the object • Type: string • Required: true

Property	Description
auto_scaling_addnode_delaytime	The time in seconds from the creation of the node which the traffic manager should wait before adding the node to the autoscaled pool. Set this to allow applications on the newly created node time to intialize before being sent traffic. • Type: UInt • Required: false • Default value: \<none\>
auto_scaling_cloud_credentials	The Cloud Credentials object containing authentication credentials to use in cloud API calls. • Type: Reference(cloud-api) • Required: false • Default value: \<none\>
auto_scaling_cluster	The ESX host or ESX cluster name to put the new virtual machine instances on. • Type: String • Required: false • Default value: \<none\>
auto_scaling_data_center	The name of the logical datacenter on the vCenter server. Virtual machines will be scaled up and down under the datacenter root folder. • Type: String • Required: false • Default value: \<none\>
auto_scaling_data_store	The name of the datastore to be used by the newly created virtual machine. • Type: String • Required: false • Default value: \<none\>

Property	Description
auto_ scaling_ enabled	Are the nodes of this pool subject to autoscaling? If yes, nodes will be automatically added and removed from the pool by the chosen autoscaling mechanism. • Type: Boolean • Required: false • Default value: false
auto_ scaling_ external	Whether or not autoscaling is being handled by an external system. Set this value to Yes if all aspects of autoscaling are handled by an external system, such as RightScale. If set to No, the traffic manager will determine when to scale the pool and will communicate with the cloud provider to create and destroy nodes as necessary. • Type: Boolean • Required: false • Default value: true
auto_ scaling_ extraarg s	Any extra arguments to the autoscaling API. Each argument can be separated by comma. E.g in case of EC2, it can take extra parameters to the Amazon's RunInstance API say DisableApiTermination=false,Placement.Tenancy=default. • Type: String • Required: false • Default value: <none>
auto_ scaling_ hysteres is	The time period in seconds for which a change condition must persist before the change is actually instigated. • Type: UInt • Required: false • Default value: "20"
auto_ scaling_ imageid	The identifier for the image of the instances to create. • Type: String

Property	Description
	• Required: false • Default value: \<none\>

auto_scaling_ips_to_use	Which type of IP addresses on the node to use. Choose private IPs if the traffic manager is in the same cloud as the nodes, otherwise choose public IPs.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "publicips"
	
	• Permitted values:
	
	    "private_ips": Private IP addresses
	
	    "publicips": Public IP addresses

auto_scaling_last_node_idle_time	The time in seconds for which the last node in an autoscaled pool must have been idle before it is destroyed. This is only relevant if min_nodes is 0. • Type: UInt • Required: false • Default value: "3600"
auto_scaling_max_nodes	The maximum number of nodes in this autoscaled pool. • Type: UInt • Required: false • Default value: "4"
auto_scaling_min_nodes	The minimum number of nodes in this autoscaled pool. • Type: UInt • Required: false • Default value: "1"

Property	Description
auto_ scaling_ name	The beginning of the name of nodes in the cloud that are part of this autoscaled pool. • Type: String • Required: false • Default value: <none>
auto_ scaling_ port	The port number to use for each node in this autoscaled pool. • Type: UInt • Required: false • Default value: "80"
auto_ scaling_ refracto ry	The time period in seconds after the instigation of a re-size during which no further changes will be made to the pool size. • Type: UInt • Required: false • Default value: "180"
auto_ scaling_ respons e_time	The expected response time of the nodes in ms. This time is used as a reference when deciding whether a node's response time is conforming. All responses from all the nodes will be compared to this reference and the percentage of conforming responses is the base for decisions about scaling the pool up or down. • Type: UInt • Required: false • Default value: "1000"
auto_ scaling_ scale_ down_ level	The fraction, in percent, of conforming requests above which the pool size is decreased. If the percentage of conforming requests exceeds this value, the pool is scaled down. • Type: UInt • Required: false • Default value: "95"

Property	Description
auto_scaling_scale_up_level	The fraction, in percent, of conforming requests below which the pool size is increased. If the percentage of conforming requests drops below this value, the pool is scaled up. • Type: UInt • Required: false • Default value: "40"
auto_scaling_security groupids	List of security group IDs to associate to the new EC2 instance. • Type: Set(String) • Required: false • Default value: <none>
auto_scaling_size_id	The identifier for the size of the instances to create. • Type: String • Required: false • Default value: <none>
auto_scaling_subnetids	List of subnet IDs where the new EC2-VPC instance(s) will be launched. Instances will be evenly distributed among the subnets. If the list is empty, instances will be launched inside EC2-Classic. • Type: Set(String) • Required: false • Default value: <none>
bandwidth_class	The Bandwidth Management Class this pool uses, if any. • Type: Reference(config-bandwidth) • Required: false • Default value: <none>

Property	Description
connection_max_connect_time	How long the pool should wait for a connection to a node to be established before giving up and trying another node. • Type: UInt • Required: false • Default value: "4"
connection_max_connections_per_node	The maximum number of concurrent connections allowed to each back-end node in this pool per machine. A value of 0 means unlimited connections. • Type: UInt • Required: false • Default value: <none>
connection_max_queue_size	The maximum number of connections that can be queued due to connections limits. A value of 0 means unlimited queue size. • Type: UInt • Required: false • Default value: <none>
connection_max_reply_time	How long the pool should wait for a response from the node before either discarding the request or trying another node (retryable requests only). • Type: UInt • Required: false • Default value: "30"
connection_max_transactions_per_node	The maximum number of concurrent transactions allowed to each back-end node in this pool per machine. A value of 0 means unlimited transactions. Idle connections kept alive for reuse do not count against this limit. A transaction begins by allocating a connection for sending the request, and ends (for the purposes of queuing) after a complete response has been received from the node. • Type: UInt

Property	Description
	• Required: false • Default value: \<none\>
connecti on_ queue_ timeout	The maximum time to keep a connection queued in seconds. • Type: UInt • Required: false • Default value: "10"
dns_ autoscal e_ enabled	When enabled, the Traffic Manager will periodically resolve the hostnames in the "hostnames" list using a DNS query, and use the results to automatically add, remove or update the IP addresses of the nodes in the pool. • Type: Boolean • Required: false • Default value: false
dns_ autoscal e_ hostna mes	A list of hostnames which will be used for DNS-derived autoscaling • Type: Set(String) • Required: false • Default value: \<none\>
dns_ autoscal e_port	The port number to use for each node when using DNS-derived autoscaling • Type: UInt • Required: false • Default value: "80"
failure_ pool	If all of the nodes in this pool have failed, then requests can be diverted to another pool. • Type: Reference(config-pool) • Required: false

Property	Description
	• Default value: <none>
ftp_suppor t_rfc_2428	Whether or not the backend IPv4 nodes understand the EPRT and EPSV command from RFC 2428. It is always assumed that IPv6 nodes support these commands. • Type: Boolean • Required: false • Default value: false
http_keepaliv e	Whether or not the pool should maintain HTTP keepalive connections to the nodes. • Type: Boolean • Required: false • Default value: true
http_keepaliv e_non_idempot ent	Whether or not the pool should maintain HTTP keepalive connections to the nodes for non-idempotent requests. • Type: Boolean • Required: false • Default value: false
kerbero s_protoco l_transitio n_principal	The Kerberos principal the traffic manager should use when performing Kerberos Protocol Transition. • Type: String • Required: false • Default value: <none>
kerbero s_protoco l_transitio n_target	The Kerberos principal name of the service this pool targets. • Type: String • Required: false • Default value: <none>

Property	Description
load_ balancin g_ algorith m	The load balancing algorithm that this pool uses to distribute load across its nodes. • Type: Enum(String) • Required: false • Default value: "round_robin" • Permitted values: "fastest_response_time": The Response Time algorithm monitors the response times for recent requests to each node. It sends each new request to the node that has recently been responding the most quickly. "least_connections": This algorithm sends each new request to the node with the fewest currently active connections. "perceptive": The Perceptive algorithm uses a combination of response time data and connection counts to predict which node is likely to have the fastest response time for each request. "random": This algorithm chooses a random node for each request. "round_robin": This algorithm distributes traffic by assigning each request to a new node in turn. "weighted_least_connections": This algorithm works in a similar way to the Least Connections algorithm, but assigns more requests to nodes with a greater 'weight'. "weighted_round_robin": Weighted Round Robin works in a similar way to Round Robin, but assigns more requests to nodes with a greater 'weight'.

Property	Description
load_ balancin g_ priority_ enabled	Enable priority lists. • Type: Boolean • Required: false • Default value: false
load_ balancin g_ priority_ nodes	Minimum number of highest-priority active nodes. • Type: UInt • Required: false • Default value: "1"
max_ connecti on_ attempt s	The maximum number of nodes to which the traffic manager will attempt to send a request before returning an error to the client. Requests that are non-retryable will be attempted against only one node. Zero signifies no limit. • Type: UInt • Required: false • Default value: <none>
max_ idle_ connecti ons_ pernode	The maximum number of unused HTTP keepalive connections that should be maintained to an individual node. Zero signifies no limit. • Type: UInt • Required: false • Default value: "50"
max_ timed_ out_ connecti on_ attempt s	The maximum number of connection attempts the traffic manager will make where the server fails to respond within the time limit defined by the max_reply_time setting. Zero signifies no limit. • Type: UInt • Required: false • Default value: "2"

Property	Description
monitor s	The monitors assigned to this pool, used to detect failures in the back end nodes. • Type: Set(Reference(config-monitor)) • Required: false • Default value: <none>
node_ close_ on_ death	Close all connections to a node once we detect that it has failed. • Type: Boolean • Required: false • Default value: false
node_ close_ with_rst	Whether or not connections to the back-end nodes should be closed with a RST packet, rather than a FIN packet. This avoids the TIME_WAIT state, which on rare occasions allows wandering duplicate packets to be safely ignored. • Type: Boolean • Required: false • Default value: false
node_ connecti on_ attempt s	The number of times the software will attempt to connect to the same back-end node before marking it as failed. This is only used when passive_monitoring is enabled. • Type: UInt • Required: false • Default value: "3"
node_ delete_ behavio r	Specify the deletion behavior for nodes in this pool. • Type: Enum(String) • Required: false • Default value: "immediate" • Permitted values:

Property	Description
	"drain": Allow existing connections to the node to finish before deletion. "immediate": All connections to the node are closed immediately.
node_ drain_ to_ delete_ timeout	The maximum time that a node will be allowed to remain in a draining state after it has been deleted. A value of 0 means no maximum time. • Type: UInt • Required: false • Default value: <none>
node_ retry_ fail_time	The amount of time, in seconds, that a traffic manager will wait before re-trying a node that has been marked as failed by passive monitoring. • Type: UInt • Required: false • Default value: "60"

nodes_ table	A table of all nodes in this pool. A node should be specified as a <ip>:<port> pair, and has a state, weight and priority.
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • node (String): A node is a combination of an ip address and port (Required)
	
	• Sub keys:
	
	    • priority (UInt): The priority of the node, higher values signify higher priority. If a priority is not specified for a node it is assumed to be 1.
	
	    • state (Enum(String)): The state of the pool, which can either be Active, Draining or Disabled

Property	Description
	Permitted values: "active": The node is is active. "disabled": The node is disabled. "draining": The node is draining. • weight (Int): Weight for the node. The actual value in isolation does not matter: As long as it is a valid integer 1-100, the per-node weightings are calculated on the relative values between the nodes. • source_ip (String): The source address the Traffic Manager uses to connect to this node.
nodes_ table_ json	A Traffic Manager REST-compatible JSON representation of the "nodes_table" table property. Use this field with the "vtm_pool_nodes_table_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
note	A description of the pool. • Type: String • Required: false • Default value: <none>
passive_ monitor ing	Whether or not the software should check that 'real' requests (i.e. not those from monitors) to this pool appear to be working. This should normally be enabled, so that when a node is refusing connections, responding too slowly, or sending back invalid data, it can mark that node as failed, and stop sending requests to it. If this is disabled, you should ensure that suitable health monitors are configured to check your servers instead, otherwise failed requests will not be detected and subsequently retried. • Type: Boolean

Property	Description
	• Required: false • Default value: true
persistence_class	The default Session Persistence class this pool uses, if any. • Type: Reference(config-persistence) • Required: false • Default value: <none>
service_discovery_enabled	Are the nodes of this pool determined by a Service Discovery plugin? If yes, nodes will be automatically added and removed from the pool by the traffic manager. • Type: Boolean • Required: false • Default value: false
service_discovery_interval	The minimum time before rerunning the Service Discovery plugin • Type: UInt • Required: false • Default value: "10"
service_discovery_plugin	The plugin script a Service Discovery autoscaled pool should use to retrieve the list of nodes. • Type: String • Required: false • Default value: <none>
service_discovery_plugin_args	The arguments for the script specified in "service_discovery!plugin", e.g. a common instance tag, or name of a managed group of cloud instances. • Type: String • Required: false

Property	Description
	• Default value: <none>
service_discovery_timeout	The maximum time a plugin should be allowed to run before timing out. Set to 0 for no timeout. • Type: UInt • Required: false • Default value: <none>
smtp_send_starttls	If we are encrypting traffic for an SMTP connection, should we upgrade to SSL using STARTTLS. • Type: Boolean • Required: false • Default value: true

ssl_ca_sites	This is table 'ca_sites'
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	  • host (String): The host'keygivesthehostnameorIPdestinationaddressusedtomatchincomingTLSconnectionstokeysoftable'ca_sites'.ThehostcanbeaspecificDNSnameforusewiththeSNIextension,aspecific destinationIPaddresswhennoSNImatches,oreitherofthosewithwildcard*/?characters.(Required)'
	
	• Sub keys:
	
	  • cert_headers (Enum(String)): Which parts of the client certificate, if any, should be inserted into requests to a back-end node, as header fields. The same fields as for ssl_client_cert_headers are made available, and optionally the base64 encoded certificate itself. (Required)

Property	Description
	Permitted values: "all": Fields and PEM "none": None "simple": Fields • client_cas (Set(String)): The certificate authorities used to verify client certificates for a particular destination site IP or SNI hostname. The specific site replaces the * (asterisk) in the key name, the value must be a valid file name in the conf/ssl/cas directory. The key can be specified multiple times to cover multiple IP addresses or SNI hostnames. (Required) • request_cert (Enum(UInt)): Whether or not the virtual server should request an identifying certificate from each client connecting to particular destination IP address or SNI hostname. If a client certificate is requested this setting also determines whether the TLS handshake can continue successfully if the client does not present a certificate. (Required) Permitted values: "dont_request": No "request": Yes, allow if absent "require": Yes, deny if absent
ssl_ca_ sites_ json	A Traffic Manager REST-compatible JSON representation of the "ssl_ca_sites" table property. Use this field with the "vtm_virtual_server_ca_sites_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>

Property	Description
ssl_ cipher_ suites	The SSL/TLS cipher suites to allow for connections to a back-end node. Leaving this empty will make the pool use the globally configured cipher suites, see configuration key ssl!cipher_suites in the Global Settings section of the System tab. See there for how to specify SSL/TLS cipher suites. • Type: String • Required: false • Default value: \<none\>
ssl_ client_ auth	Whether or not a suitable certificate and private key from the SSL Client Certificates catalog be used if the back-end server requests client authentication. • Type: Boolean • Required: false • Default value: false
ssl_ commo n_ name_ match	A list of names against which the 'common name' of the certificate is matched; these names are used in addition to the node's hostname or IP address as specified in the config file or added by the autoscaler process. • Type: Set(String) • Required: false • Default value: \<none\>
ssl_ elliptic_ curves	The SSL elliptic curve preference list for SSL connections from this pool using TLS version 1.0 or higher. Leaving this empty will make the pool use the globally configured preference list. The named curves P256, P384 and P521 may be configured. • Type: List(String) • Required: false • Default value: \<none\>
ssl_ enable	Whether or not the pool should encrypt data before sending it to a back-end node. • Type: Boolean

Property	Description
	• Required: false • Default value: false
ssl_ enhance	SSL protocol enhancements allow your traffic manager to prefix each new SSL connection with information about the client. This enables Pulse Secure Virtual Traffic Manager virtual servers referenced by this pool to discover the original client's IP address. Only enable this if you are using nodes for this pool which are Pulse Secure vTMs, whose virtual servers have the ssl_trust_magic setting enabled. • Type: Boolean • Required: false • Default value: false

ssl_ middleb ox_ compati bility	Whether or not TLS 1.3 middlebox compatibility mode as described in RFC 8446 appendix D.4 will be used in connections to pool nodes. Choosing the global setting means the value of configuration key ssl!middlebox_compatibility from the Global Settings section of the System tab will be enforced.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "use_default"
	
	• Permitted values:
	
	    "disabled": Disable use of middlebox compatibility
	
	    "enabled": Enable use of middlebox compatibility
	
	    "use_default": Use the global setting for use of middlebox compatibility

ssl_ send_ close_ alerts	Whether or not to send an SSL/TLS "close alert" when initiating a socket disconnection. • Type: Boolean • Required: false • Default value: true

Property	Description
ssl_ server_ name	Whether or not the software should use the TLS 1.0 server_name extension, which may help the back-end node provide the correct certificate. Enabling this setting will force the use of at least TLS 1.0. • Type: Boolean • Required: false • Default value: false

ssl_ session_ cache_ enabled	Whether or not the SSL client cache will be used for this pool. Choosing the global setting means the value of the configuration key ssl!client_cache!enabled from the Global Settings section of the System tab will be enforced.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "use_default"
	
	• Permitted values:
	
	      "disabled": Disable use of the session cache
	
	      "enabled": Enable use of the session cache
	
	      "use_default": Use the global setting for use of the session cache

ssl_ session_ tickets_ enabled	Whether or not SSL session tickets, including TLS >= 1.3 PSKs, will be used for this pool if the session cache is also enabled. Choosing the global setting means the value of the configuration key ssl!client_cache!tickets_enabled from the Global Settings section of the System tab will be enforced. • Type: Enum(String) • Required: false • Default value: "use_default" • Permitted values:

Property	Description
	"disabled": Disable use of session tickets "enabled": Enable use of session tickets "use_default": Use the global setting for use of session tickets
ssl_ signatur e_ algorith ms	The SSL signature algorithms preference list for SSL connections from this pool using TLS version 1.2 or higher. Leaving this empty will make the pool use the globally configured preference list, signature_algorithms in the ssl section of the global_settings resource. See there and in the online help for how to specify SSL signature algorithms. • Type: String • Required: false • Default value: <none>
ssl_ssl_ fixed_ client_ certifica te	An entry in the SSL client certificates catalog, containing a certificate and private key to be used whenever client authentication is requested. If set, this overrides server request parameters. • Type: String • Required: false • Default value: <none>
ssl_ strict_ verify	Whether or not strict certificate verification should be performed. This will turn on checks to disallow server certificates that don't match the server name or a name in the ssl_common_name_match list, are self-signed, expired, revoked, or have an unknown CA. • Type: Boolean • Required: false • Default value: false
ssl_ suppor t_ssl3	Whether or not SSLv3 is enabled for this pool. Choosing the global setting means the value of the configuration key ssl!support_ssl3 from the Global Settings section of the System tab will be enforced.

Property	Description
	• Type: Enum(String)
	• Required: false
	• Default value: "use_default"
	• Permitted values:
	"disabled": Disable SSLv3
	"enabled": Enable SSLv3
	"use_default": Use the global setting for SSLv3
ssl_suppor t_tls1	Whether or not TLSv1.0 is enabled for this pool. Choosing the global setting means the value of the configuration key ssl!support_tls1 from the Global Settings section of the System tab will be enforced. • Type: Enum(String) • Required: false • Default value: "use_default" • Permitted values: "disabled": Disable TLSv1.0 "enabled": Enable TLSv1.0 "use_default": Use the global setting for TLSv1.0
ssl_suppor t_tls1_1	Whether or not TLSv1.1 is enabled for this pool. Choosing the global setting means the value of the configuration key ssl!support_tls1_1 from the Global Settings section of the System tab will be enforced. • Type: Enum(String) • Required: false • Default value: "use_default" • Permitted values:

Property	Description
	"disabled": Disable TLSv1.1 "enabled": Enable TLSv1.1 "use_default": Use the global setting for TLSv1.1

ssl_ suppor t_tls1_2	Whether or not TLSv1.2 is enabled for this pool. Choosing the global setting means the value of the configuration key ssl!support_tls1_2 from the Global Settings section of the System tab will be enforced.
	
	 • Type: Enum(String)
	
	 • Required: false
	
	 • Default value: "use_default"
	
	 • Permitted values:
	
	     "disabled": Disable TLSv1.2
	
	     "enabled": Enable TLSv1.2
	
	     "use_default": Use the global setting for TLSv1.2

ssl_ suppor t_tls1_3	Whether or not TLSv1.3 is enabled for this pool. Choosing the global setting means the value of the configuration key ssl!support_tls1_3 from the Global Settings section of the System tab will be enforced.
	
	 • Type: Enum(String)
	
	 • Required: false
	
	 • Default value: "use_default"
	
	 • Permitted values:
	
	     "disabled": Disable TLSv1.3
	
	     "enabled": Enable TLSv1.3
	
	     "use_default": Use the global setting for TLSv1.3

Property	Description
tcp_ nagle	Whether or not Nagle's algorithm should be used for TCP connections to the back-end nodes. • Type: Boolean • Required: false • Default value: true
transpar ent	Whether or not connections to the back-ends appear to originate from the source client IP address. • Type: Boolean • Required: false • Default value: false

udp_ accept_ from	The IP addresses and ports from which responses to UDP requests should be accepted. If set to accept responses from a specific set of IP addresses, you will need to enter a CIDR Mask (such as 10.100.0.0/16).
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "dest_only"
	
	• Permitted values:
	
	    "all": Any IP address and any port.
	
	    "dest_ip_only": Only the IP address to which the request was sent, but from any port.
	
	    "dest_only": Only the IP address and port to which the request was sent.
	
	    "ip_mask": Only a specific set of IP addresses, but from any port.

Property	Description
udp_ accept_ from_ mask	The CIDR mask that matches IPs we want to receive responses from. • Type: String • Required: false • Default value: \<none\>
udp_ respons e_ timeout	The maximum length of time that a node is permitted to take after receiving a UDP request packet before sending a reply packet. Zero indicates that there is no maximum, preventing a node that does not send replies from being presumed to have failed. • Type: UInt • Required: false • Default value: \<none\>

vtm_protection

URI Endpoint: /api/tm/8.3/config/active/protection

A protection class specifies the level of protection against network attacks for a virtual server.

```
# Example usage of vtm_protection resource...
resource "vtm_protection" "my_protection" {
    name = "MyProtection"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
access_restriction_allowed	Always allow access to these IP addresses. This overrides the connection limits for these machines, but does not stop other restrictions such as HTTP validity checks. • Type: Set(String)

Property	Description
	• Required: false • Default value: <none>
access_restriction_banned	Disallow access to these IP addresses. • Type: Set(String) • Required: false • Default value: <none>
concurrent_connections_max_10_connections	Additional limit on maximum concurrent connections from the top 10 busiest connecting IP addresses combined. The value should be between 1 and 10 times the max_1_connections limit. (This limit is disabled if per_process_connection_count is No, or max_1_connections is 0, or min_connections is 0.) • Type: UInt • Required: false • Default value: "200"
concurrent_connections_max_1_connections	Maximum concurrent connections each connecting IP address is allowed. Set to 0 to disable this limit. • Type: UInt • Required: false • Default value: "30"
concurrent_connections_min_connections	Entry threshold for the max_10_connections limit: the max_10_connections limit is not applied to connecting IP addresses with this many or fewer concurrent connections. Setting to 0 disables both the max_1_connections and max_10_connections limits, if per_process_connection_count is Yes. (If per_process_connection_count is No, this setting is ignored.) • Type: UInt

Property	Description
	• Required: false • Default value: "4"
concurrent_connections_ per_process_connection_ count	Whether concurrent connection counting and limits are per-process. (Each Traffic Manager typically has several processes: one process per available CPU core.) If Yes, a connecting IP address may make that many connections to each process within a Traffic Manager. If No, a connecting IP address may make that many connections to each Traffic Manager as a whole. • Type: Boolean • Required: false • Default value: true
connection_rate_max_ connection_rate	Maximum number of new connections each connecting IP address is allowed to make in the rate_timer interval. Set to 0 to disable this limit. If applied to an HTTP Virtual Server each request sent on a connection that is kept alive counts as a new connection. The rate limit is per process: each process within a Traffic Manager accepts new connections from the connecting IP address at this rate. (Each Traffic Manager typically has several processes: one process per available CPU core). • Type: UInt • Required: false • Default value: <none>
connection_rate_rate_timer	How frequently the max_connection_rate is assessed. For example, a value of 1 (second) will impose a limit of max_connection_rate connections per second; a value of 60 will impose a limit of max_connection_rate connections per minute. The valid range is 1-99999 seconds. • Type: UInt • Required: false

Property	Description
	• Default value: "60"
debug	Whether or not to output verbose logging. • Type: Boolean • Required: false • Default value: false
enabled	Enable or disable this service protection class. • Type: Boolean • Required: false • Default value: true
http_check_rfc2396	Whether or not requests with poorly-formed URLs be should be rejected. This tests URL compliance as defined in RFC2396. Note that enabling this may block some older, non-conforming web browsers. • Type: Boolean • Required: false • Default value: false
http_max_body_length	Maximum permitted length of HTTP request body data, set to 0 to disable the limit. • Type: UInt • Required: false • Default value: <none>
http_max_header_length	Maximum permitted length of a single HTTP request header (key and value), set to 0 to disable the limit. • Type: UInt • Required: false

Property	Description
	• Default value: <none>
http_max_request_length	Maximum permitted size of all the HTTP request headers, set to 0 to disable the limit. • Type: UInt • Required: false • Default value: <none>
http_max_url_length	Maximum permitted URL length, set to 0 to disable the limit. • Type: UInt • Required: false • Default value: <none>
http_reject_binary	Whether or not URLs and HTTP request headers that contain binary data (after decoding) should be rejected. • Type: Boolean • Required: false • Default value: false
http_send_error_page	This setting tells the traffic manager to send an HTTP error message if a connection fails the service protection tests, instead of just dropping it. Details of which HTTP response will be sent when particular tests fail can be found in the Help section for this page. • Type: Boolean • Required: false • Default value: true
log_time	Log service protection messages at these intervals. If set to 0 no messages will be logged and no alerts will be sent. • Type: UInt

Property	Description
	• Required: false • Default value: "60"
note	A description of the service protection class. • Type: String • Required: false • Default value: <none>
rule	A TrafficScript rule that will be run on the connection after the service protection criteria have been evaluated. This rule will be executed prior to normal rules configured for the virtual server. • Type: Reference(config-trafficscript) • Required: false • Default value: <none>
testing	Place the service protection class into testing mode. (Log when this class would have dropped a connection, but allow all connections through). • Type: Boolean • Required: false • Default value: false

vtm_application_firewall

URI Endpoint: /api/tm/8.3/config/active/application_firewall

The conf/zeusafm.conf file contains configuration files for the application firewall. Some keys present in the zeusafm.conf are not documented here. Refer to the Pulse Secure Web Application Firewall documentation for further details. The configuration can be edited under the System > Application Firewall section of the Administration Server or by using functions under the AFM section of the SOAP API and CLI.

Property	Description
There are no properties to display for this resource.	

vtm_rate

URI Endpoint: /api/tm/8.3/config/active/rate

A rate shaping class restricts the number of connections being processed by a virtual server at once.

```
# Example usage of vtm_rate resource...
resource "vtm_rate" "my_rate" {
    name = "MyRate"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
max_rate_per_minute	Requests that are associated with this rate class will be rate-shaped to this many requests per minute, set to 0 to disable the limit. • Type: UInt • Required: false • Default value: \<none\>
max_rate_per_second	Although requests will be rate-shaped to the max_rate_per_minute, the traffic manager will also rate limit per-second. This smooths traffic so that a full minute's traffic will not be serviced in the first second of the minute, set this to 0 to disable the per-second limit. • Type: UInt • Required: false • Default value: \<none\>
note	A description of the rate class.

Property	Description
	• Type: FreeformString • Required: false • Default value: \<none\>

vtm_rule

URI Endpoint: /api/tm/8.3/config/active/rules

TrafficScript rules allow traffic inspection and modification.

```
# Example usage of vtm_rule resource...
resource "vtm_rule" "my_rule" {
    content = ""
    name = "MyRule"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
content	Object content • Type: string • Required: true

vtm_service_level_monitor

URI Endpoint: /api/tm/8.3/config/active/service_level_monitors

Service level monitoring is used to produce alerts when an application's performance is degraded. This is done by monitoring the response time of connections to a virtual server.

```
# Example usage of vtm_service_level_monitor resource...
resource "vtm_service_level_monitor" "my_service_level_monitor" {
    name = "MyServiceLevelMonitor"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
note	A description for the SLM class. • Type: FreeformString • Required: false • Default value: <none>
response_time	Responses that start being sent to the client within this time limit, expressed in milliseconds, are treated as conforming. • Type: UInt • Required: false • Default value: "1000"
serious_threshold	When the percentage of conforming responses drops below this level, a serious error level message will be emitted. • Type: UInt • Required: false • Default value: <none>
warning_threshold	When the percentage of conforming responses drops below this level, a warning message will be emitted. • Type: UInt • Required: false

Property	Description
	• Default value: "50"

vtm_ssl_client_key

URI Endpoint: /api/tm/8.3/config/active/ssl/client_keys

SSL Client Certificates are used when connecting to backend nodes that require client certificate authentication.

```
# Example usage of vtm_ssl_client_key resource...
resource "vtm_ssl_client_key" "my_ssl_client_key" {
    name = "MySslClientKey"
    note =
    private =
    public =
    request =
}
```

Property	Description
name	Name of the object • Type: string • Required: true
note	Notes for this certificate • Type: FreeformString • Required: true • Default value: \<none\>
private	Private key for certificate • Type: FreeformString • Required: true • Default value: \<none\>

Property	Description
public	Public certificate • Type: FreeformString • Required: true • Default value: <none>
request	Certificate Signing Request for certificate • Type: FreeformString • Required: true • Default value: <none>

vtm_ssl_server_key

URI Endpoint: /api/tm/8.3/config/active/ssl/server_keys

SSL Server Certificates are presented to clients by virtual servers when SSL decryption is enabled.

```
# Example usage of vtm_ssl_server_key resource...
resource "vtm_ssl_server_key" "my_ssl_server_key" {
    name = "MySslServerKey"
    note =
    private =
    public =
    request =
}
```

Property	Description
name	Name of the object • Type: string • Required: true
note	Notes for this certificate • Type: FreeformString

Property	Description
	• Required: true • Default value: \<none\>
private	Private key for certificate • Type: FreeformString • Required: true • Default value: \<none\>
public	Public certificate • Type: FreeformString • Required: true • Default value: \<none\>
request	Certificate Signing Request for certificate • Type: FreeformString • Required: true • Default value: \<none\>

vtm_ssl_ticket_key

URI Endpoint: /api/tm/8.3/config/active/ssl/ticket_keys

Configuration for SSL ticket encryption keys when managed externally via the ssl/ticket_keys REST API endpoints.

```
# Example usage of vtm_ssl_ticket_key resource...
resource "vtm_ssl_ticket_key" "my_ssl_ticket_key" {
    id =
    key =
    name = "MySslTicketKey"
    validity_end =
    validity_start =
}
```

Property	Description
name	Name of the object • Type: string • Required: true
algorithm	The algorithm used to encrypt session tickets. The algorithm determines the length of the key that must be provided. • Type: Enum(String) • Required: false • Default value: "aes_256_cbc_hmac_sha256" • Permitted values: "aes_256_cbc_hmac_sha256": AES-256 CBC with HMAC-SHA256. Requires a total of 64 bytes of key material.
identifier	A 16-byte key identifier, with each byte encoded as two hexadecimal digits. Key identifiers are transmitted in plaintext at the beginning of a TLS session ticket, and are used to identify the ticket encryption key that was used to encrypt a ticket. (They correspond to the 'key_name' field in RFC 5077.) They are required to be unique across the set of SSL ticket encryption keys. • Type: String • Required: true • Default value: <none>
key	The session ticket encryption key, with each byte encoded as two hexadecimal digits. The required key length is determined by the chosen key algorithm. See the documentation for the 'algorithm' field for more details. • Type: Password • Required: true

Property	Description
	• Default value: <none>
validity_end	The latest time at which this key may be used to encrypt new session tickets. Given as number of seconds since the epoch (1970-01-01T00:00:00Z). • Type: UInt • Required: true • Default value: <none>
validity_start	The earliest time at which this key may be used to encrypt new session tickets. Given as number of seconds since the epoch (1970-01-01T00:00:00Z). • Type: UInt • Required: true • Default value: <none>

vtm_ssl_ca

URI Endpoint: /api/tm/8.3/config/active/ssl/cas

SSL certificate authority certificates (CAs) and certificate revocation lists (CRLs) can be used when validating server and client certificates.

```
# Example usage of vtm_ssl_ca resource...
resource "vtm_ssl_ca" "my_ssl_ca" {
    content = ""
    name = "MySslCa"
}
```

Property	Description
name	Name of the object • Type: string • Required: true

Property	Description
content	Object content • Type: string • Required: true

vtm_security

URI Endpoint: /api/tm/8.3/config/active/security

Security settings that restrict remote administration for the cluster. Additional security options can be found in Global Settings.

```
# Example usage of vtm_security resource...
resource "vtm_security" "my_security" {


}
```

Property	Description
access	Access to the admin server and REST API is restricted by usernames and passwords. You can further restrict access to just trusted IP addresses, CIDR IP subnets or DNS wildcards. These access restrictions are also used when another traffic manager initially joins the cluster, after joining the cluster these restrictions are no longer used. Care must be taken when changing this setting, as it can cause the administration server to become inaccessible.Access to the admin UI will not be affected until it is restarted. • Type: Set(String) • Required: false • Default value: <none>
ssh_intrusion_bantime	The amount of time in seconds to ban an offending host for. • Type: UInt • Required: false

Property	Description
	• Default value: "600"
ssh_intrusion_blacklist	The list of hosts to permanently ban, identified by IP address or DNS hostname in a space-separated list. • Type: Set(String) • Required: false • Default value: <none>
ssh_intrusion_enabled	Whether or not the SSH Intrusion Prevention tool is enabled. • Type: Boolean • Required: false • Default value: true
ssh_intrusion_findtime	The window of time in seconds the maximum number of connection attempts applies to. More than (maxretry) failed attempts in this time span will trigger a ban. • Type: UInt • Required: false • Default value: "600"
ssh_intrusion_maxretry	The number of failed connection attempts a host can make before being banned. • Type: UInt • Required: false • Default value: "6"
ssh_intrusion_whitelist	The list of hosts to never ban, identified by IP address, DNS hostname or subnet mask, in a space-separated list. • Type: Set(String) • Required: false

Property	Description
	• Default value: \<none\>

vtm_servicediscovery

URI Endpoint: /api/tm/8.3/config/active/servicediscovery

The conf/servicediscovery directory contains plugins for use with Service Discovery for pool nodes.

Property	Description
name	Name of the object • Type: string • Required: true
content	Object content • Type: string • Required: true

vtm_persistence

URI Endpoint: /api/tm/8.3/config/active/persistence

A session persistence class is used to identify the session a new connection belongs too and deliver it to the same backend node.

```
# Example usage of vtm_persistence resource...
resource "vtm_persistence" "my_persistence" {
    name = "MyPersistence"
}
```

Property	Description
name	Name of the object • Type: string • Required: true

Property	Description
cookie	The cookie name to use for tracking session persistence. • Type: String • Required: false • Default value: <none>
delete	Whether or not the session should be deleted when a session failure occurs. (Note, setting a failure mode of 'choose a new node' implicitly deletes the session.) • Type: Boolean • Required: false • Default value: true

failure_mode	The action the pool should take if the session data is invalid or it cannot contact the node specified by the session.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "new_node"
	
	• Permitted values:
	
	    "close": Close the connection (using error_file on Virtual Servers > Edit > Protocol Settings)
	
	    "new_node": Choose a new node to use
	
	    "url": Redirect the user to a given URL

note	A description of the session persistence class. • Type: FreeformString • Required: false

Property	Description
	• Default value: <none>
subnet_prefix_length_v4	When using IP-based session persistence, ensure all requests from this IPv4 subnet, specified as a prefix length, are sent to the same node. If set to 0, requests from different IPv4 addresses will be load-balanced individually. • Type: Int • Required: false • Default value: <none>
subnet_prefix_length_v6	When using IP-based session persistence, ensure all requests from this IPv6 subnet, specified as a prefix length, are sent to the same node. If set to 0, requests from different IPv6 addresses will be load-balanced individually. • Type: Int • Required: false • Default value: <none>
transparent_always_set_cookie	Whether or not the cookie should be inserted in every response sent to the client when using transparent session affinity. If set to No then the cookie is inserted only if the corresponding request did not already contain a matching cookie. • Type: Boolean • Required: false • Default value: false
transparent_directives	The cookie directives to include in the cookie sent when using transparent session affinity. If more than one directive is included, the semi-colon separator between them must be included in this string. The semi-colon separator between the cookie value and the first directive should not be included in this string. • Type: String

Property	Description
	• Required: false • Default value: \<none\>
type	The type of session persistence to use. • Type: Enum(String) • Required: false • Default value: "ip" • Permitted values: "asp": ASP and ASP.NET session persistence "cookie": Monitor application cookies "ip": IP-based persistence "j2ee": J2EE session persistence "named": Named Node session persistence "ssl": SSL Session ID persistence "transparent": Transparent session affinity "universal": Universal session persistence "x_zeus": X-Zeus-Backend cookies
url	The redirect URL to send clients to if the session persistence is configured to redirect users when a node dies. • Type: String • Required: false • Default value: \<none\>

vtm_traffic_ip_group

URI Endpoint: /api/tm/8.3/config/active/traffic_ip_groups

Traffic IP groups are sets of IP addresses that are distributed across a cluster for fault tolerance.

```
# Example usage of vtm_traffic_ip_group resource...
resource "vtm_traffic_ip_group" "my_traffic_ip_group" {
    name = "MyTrafficIpGroup"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
enabled	If set to No, the traffic IP group will be disabled and none of the traffic IP addresses will be raised. • Type: Boolean • Required: false • Default value: true
hash_source_port	Whether or not the source port should be taken into account when deciding which traffic manager should handle a request. • Type: Boolean • Required: false • Default value: false
ip_assignment_mode	Configure how traffic IPs are assigned to traffic managers in Single-Hosted mode • Type: Enum(String) • Required: false • Default value: "balanced" • Permitted values: "alphabetic": Alphabetical order of traffic manager hostnames

Property	Description
	"balanced": Approximately balanced between traffic managers

ip_mapping	A table assigning traffic IP addresses to machines that should host them. Traffic IP addresses not specified in this table will automatically be assigned to a machine.
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • ip (String): A traffic IP address (from the ipaddresses property). (Required)
	
	• Sub keys:
	
	    • traffic_manager (String): The name of the traffic manager that should host the IP address. (Required)

ip_mapping_json	A Traffic Manager REST-compatible JSON representation of the "ip_mapping" table property. Use this field with the "vtm_traffic_ip_group_ip_mapping_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
ipaddresses	The IP addresses that belong to the Traffic IP group. • Type: Set(String) • Required: false • Default value: <none>

Property	Description
keeptogether	If set to Yes then all the traffic IPs will be raised on a single traffic manager. By default they're distributed across all active traffic managers in the traffic IP group. • Type: Boolean • Required: false • Default value: false
location	The location in which the Traffic IP group is based. • Type: Int • Required: false • Default value: <none>
machines	The traffic managers that can host the traffic IP group's IP addresses. • Type: Set(Reference(config-tm)) • Required: false • Default value: <none>

mode	The method used to distribute traffic IPs across machines in the cluster. If "multihosted" is used then multicast must be set to an appropriate multicast IP address.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "singlehosted"
	
	• Permitted values:
	
	    "ec2elastic": Use an EC2-Classic Elastic IP address.
	
	    "ec2vpcelastic": Use an EC2-VPC Elastic IP address.

Property	Description
	"ec2vpcprivate": Use an EC2-VPC Private IP address. "gceexternal": Use GCE External IP addresses. "multihosted": Raise each address on every machine in the group (Multi-Hosted mode) - IPv4 only "rhi": Use route health injection to route traffic to the active machine - IPv4 only "singlehosted": Raise each address on a single machine (Single-Hosted mode)
multicast	The multicast IP address used to duplicate traffic to all traffic managers in the group. • Type: String • Required: false • Default value: <none>
note	A note, used to describe this Traffic IP Group • Type: String • Required: false • Default value: <none>
rhi_bgp_metric_base	The base BGP routing metric for this Traffic IP group. This is the advertised routing cost for the active traffic manager in the cluster. It can be used to set up inter-cluster failover. • Type: UInt • Required: false • Default value: "10"

Property	Description
rhi_bgp_passive_metric_offset	The BGP routing metric offset for this Traffic IP group. This is the difference between the advertised routing cost for the active and passive traffic manager in the cluster. • Type: UInt • Required: false • Default value: "10"
rhi_ospfv2_metric_base	The base OSPFv2 routing metric for this Traffic IP group. This is the advertised routing cost for the active traffic manager in the cluster. It can be used to set up inter-cluster failover. • Type: UInt • Required: false • Default value: "10"
rhi_ospfv2_passive_metric_offset	The OSPFv2 routing metric offset for this Traffic IP group. This is the difference between the advertised routing cost for the active and passive traffic manager in the cluster. • Type: UInt • Required: false • Default value: "10"
rhi_protocols	A list of protocols to be used for RHI. Currently must be 'ospf' or 'bgp' or both. The default, if empty, is 'ospf', which means that it is not possible to specify no protocol. • Type: String • Required: false • Default value: "ospf"

Property	Description
slaves	A list of traffic managers that are in 'passive' mode. This means that in a fully working environment, they will not have any traffic IP addresses assigned to them. • Type: Set(Reference(config-tm)) • Required: false • Default value: \<none\>

vtm_traffic_manager

URI Endpoint: /api/tm/8.3/config/active/traffic_managers

The conf/zxtms directory contains a configuration file for each traffic manager in your cluster. The name of each file is the hostname of the traffic manager it represents. These files contain host-specific configuration data and on each installation of the software, the conf/../global.cfg file is sym-linked to the host's own configuration in the conf/zxtms directory. The files may contain a variety of configuration options that are configured in various locations under the System section of the Admin Server UI and the System section of the SOAP API and CLI.

```
# Example usage of vtm_traffic_manager resource...
resource "vtm_traffic_manager" "my_traffic_manager" {
    name = "MyTrafficManager"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
admin_hsts_enable	Whether or not HSTS (RFC 6797) is enabled for admin server connections. • Type: Boolean • Required: false

Property	Description
	• Default value: false
admin_hsts_max_age	The number of seconds that the HSTS header field max-age will be set to • Type: UInt • Required: false • Default value: "31536000"
adminmasterxmlip	The Application Firewall master XML IP. • Type: String • Required: false • Default value: "0.0.0.0"
adminslavexmlip	The Application Firewall slave XML IP. • Type: String • Required: false • Default value: "0.0.0.0"

appliance_card	The table of network cards of a hardware appliance
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	  • name (String): Network card PCI ID (Required)
	
	• Sub keys:
	
	  • interfaces (List(String)): The order of the interfaces of a network card (Required)
	
	  • label (String): The labels of the installed network cards

Property	Description
appliance_card_json	A Traffic Manager REST-compatible JSON representation of the "appliance_card" table property. Use this field with the "vtm_traffic_manager_appliance_card_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
appliance_disable_kpti	Whether the traffic manager appliance should run without kernel page table isolation (KPTI). KPTI provides protection to prevent unprivileged software from being potentially able to read arbitrary memory from the kernel (i.e. the Meltdown attack, CVE-2017-5754); however this protection incurs a general system performance penalty. If you are running trusted software on the appliance, and the trade-off between performance at the cost of 'defense in depth' favors the former in your deployment, you may wish to enable this configuration key. If you are unsure, it is recommended that you leave this key disabled, which is also the default. • Type: Boolean • Required: false • Default value: false
appliance_dnscache	The DNS cache setting the appliance should use and place in /etc/systemd/resolved.conf. • Type: Boolean • Required: false • Default value: true
appliance_dnssec	The DNSSEC setting the appliance should use and place in /etc/systemd/resolved.conf. • Type: Enum(String)

Property	Description

	• Required: false
	
	• Default value: "no"
	
	• Permitted values:
	
	    "allow_downgrade": Use DNSSEC when available
	
	    "no": DNSSEC disabled
	
	    "yes": DNSSEC enabled

appliance_gateway_ipv4	The default gateway. • Type: String • Required: false • Default value: <none>
appliance_gateway_ipv6	The default IPv6 gateway. • Type: String • Required: false • Default value: <none>
appliance_hostname	Name (hostname.domainname) of the appliance. • Type: String • Required: false • Default value: <none>
appliance_hosts	A table of hostname to static ip address mappings, to be placed in the /etc/hosts file. • Type: Table • Required: false

Property	Description
	• Primary key:
	• name (String): The name of a host. (Required)
	• Sub keys:
	• ip_address (String): The static IP address of the host. (Required)
appliance_hosts_json	A Traffic Manager REST-compatible JSON representation of the "appliance_hosts" table property. Use this field with the "vtm_traffic_manager_hosts_table" data source for dynamic table generation from input variables.
	• Type: String
	• Required: false
	• Default value: \<none\>
appliance_if	A table of network interface specific settings.
	• Type: Table
	• Required: false
	• Primary key:
	• name (String): A network interface name. (Required)
	• Sub keys:
	• autoneg (Boolean): Whether auto-negotiation should be enabled for the interface.
	• bmode (Enum(String)): The trunking mode used for the interface (only 802.3ad is currently supported).
	Permitted values:
	"802_3ad": IEEE 802.3ad
	"balance_alb": Adaptive Load Balancing

Property	Description
	• bond (String): The trunk of which the interface should be a member. • duplex (Boolean): Whether full-duplex should be enabled for the interface. • mode (Enum(String)): Set the configuriation mode of an interface, the interface name is used in place of the * (asterisk). Permitted values: "dhcp": DHCP "static": Static • mtu (UInt): The maximum transmission unit (MTU) of the interface. • speed (Enum(String)): The speed of the interface. Permitted values: "10": 10Mbs "100": 100Mbs "1000": 1Gbs "10000": 10Gbs "100000": 100Gbs "40000": 40Gbs
appliance_if_json	A Traffic Manager REST-compatible JSON representation of the "appliance_if" table property. Use this field with the "vtm_traffic_manager_if_table" data source for dynamic table generation from input variables. • Type: String

Property	Description
	• Required: false • Default value: <none>

appliance_ip	A table of network interfaces and their network settings.
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	  • name (String): A network interface name. (Required)
	
	• Sub keys:
	
	  • addr (String): The IP address for the interface. (Required)
	
	  • isexternal (Boolean): Whether the interface is externally facing.
	
	  • mask (String): The IP mask (netmask) for the interface. (Required)

appliance_ip_json	A Traffic Manager REST-compatible JSON representation of the "appliance_ip" table property. Use this field with the "vtm_traffic_manager_ip_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
appliance_ipmi_lan_access	Whether IPMI LAN access should be enabled or not. • Type: Boolean • Required: false • Default value: false

Property	Description
appliance_ipmi_lan_addr	The IP address of the appliance IPMI LAN channel. • Type: String • Required: false • Default value: \<none\>
appliance_ipmi_lan_gateway	The default gateway of the IPMI LAN channel. • Type: String • Required: false • Default value: "0.0.0.0"

appliance_ipmi_lan_ipsrc	The addressing mode the IPMI LAN channel operates.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "static"
	
	• Permitted values:
	
	    "dhcp": Address obtained by DHCP
	
	    "static": Static IP Address

appliance_ipmi_lan_mask	Set the IP netmask for the IPMI LAN channel. • Type: String • Required: false • Default value: \<none\>
appliance_ipv4_forwarding	Whether or not IPv4 forwarding is enabled. • Type: Boolean • Required: false • Default value: false

Property	Description
appliance_ipv6_forwarding	Whether or not IPv6 forwarding is enabled. • Type: Boolean • Required: false • Default value: false
appliance_licence_agreed	Whether or not the license agreement has been accepted. This determines whether or not the Initial Configuration wizard is displayed. • Type: Boolean • Required: false • Default value: false
appliance_ manageazureroutes	Whether or not the software manages the Azure policy routing. • Type: Boolean • Required: false • Default value: true
appliance_manageec2conf	Whether or not the software manages the EC2 config. • Type: Boolean • Required: false • Default value: true
appliance_managegceroutes	Whether or not the software manages the GCE routing. • Type: Boolean • Required: false • Default value: true
appliance_manageiptrans	Whether or not the software manages the IP transparency

Property	Description
	• Type: Boolean • Required: false • Default value: true
appliance_managereservedports	Whether or not the software manages the system configuration for reserved ports • Type: Boolean • Required: false • Default value: true
appliance_managereturnpath	Whether or not the software manages return path routing. If disabled, the appliance won't modify iptables / rules / routes for this feature. • Type: Boolean • Required: false • Default value: true
appliance_manageservices	Whether or not the software manages the system services • Type: Boolean • Required: false • Default value: true
appliance_managevpcconf	Whether or not the software manages the EC2-VPC secondary IPs. • Type: Boolean • Required: false • Default value: true
appliance_name_servers	The IP addresses of the nameservers the appliance should use and place in /etc/systemd/resolved.conf.

Property	Description
	• Type: Set(String) • Required: false • Default value: \<none\>
appliance_ntpservers	The NTP servers the appliance should use to synchronize its clock. • Type: List(String) • Required: false • Default value: "0.zeus.pool.ntp.org 1.zeus.pool.ntp.org 2.zeus.pool.ntp.org 3.zeus.pool.ntp.org"

appliance_routes	A table of destination IP addresses and routing details to reach them.
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	  • name (String): A destination IP address. (Required)
	
	• Sub keys:
	
	  • gw (String): The gateway IP to configure for the route. (Required)
	
	  • if (String): The network interface to configure for the route. (Required)
	
	  • mask (String): The netmask to apply to the IP address. (Required)

appliance_routes_json	A Traffic Manager REST-compatible JSON representation of the "appliance_routes" table property. Use this field with the "vtm_traffic_manager_routes_table" data source for dynamic table generation from input variables. • Type: String

Property	Description
	• Required: false • Default value: <none>
appliance_search_domains	The search domains the appliance should use and place in /etc/systemd/resolved.conf. • Type: Set(String) • Required: false • Default value: <none>
appliance_ssh_enabled	Whether or not the SSH server is enabled on the appliance. • Type: Boolean • Required: false • Default value: true
appliance_ssh_password_allowed	Whether or not the SSH server allows password based login. • Type: Boolean • Required: false • Default value: true
appliance_ssh_port	The port that the SSH server should listen on. • Type: UInt • Required: false • Default value: "22"
appliance_sysctl	Custom kernel parameters applied by the user with sysctl interface • Type: Table • Required: false • Primary key:

Property	Description

	• sysctl (String): The name of the kernel parameter, e.g. net.ipv4.forward (Required)
	
	• Sub keys:
	
	    • description (String): Associated optional description for the sysctl
	
	    • value (String): The value of the kernel parameter (Required)

appliance_sysctl_json	A Traffic Manager REST-compatible JSON representation of the "appliance_sysctl" table property. Use this field with the "vtm_traffic_manager_appliance_sysctl_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
appliance_timezone	The timezone the appliance should use. This must be a path to a timezone file that exists under /usr/share/zoneinfo/. • Type: String • Required: false • Default value: "US/Pacific"
appliance_vlans	The VLANs the software should raise. A VLAN should be configured using the format <dev>.<vlanid>, where <dev> is the name of a network device that exists in the host system, eth0.100 for example. • Type: Set(String) • Required: false • Default value: <none>
authenticationserverip	The Application Firewall Authentication Server IP.

Property	Description
	• Type: String • Required: false • Default value: "0.0.0.0"
cloud_platform	Cloud platform where the traffic manager is running. • Type: String • Required: false • Default value: \<none\>
cluster_comms_allow_update	Whether or not this instance of the software can send configuration updates to other members of the cluster. When not clustered this key is ignored. When clustered the value can only be changed by another machine in the cluster that has allow_update set to true. If set to false then it will not be possible to log into the admin server for this instance. • Type: Boolean • Required: false • Default value: true
cluster_comms_bind_ip	The IP address that the software should bind to for internal administration communications. See also port. If the software is not part of a cluster the default is to use 127.0.0.1 and there should be no reason to touch this setting. If the software is part of a cluster then the default is to listen on all raised IPs, in this case an alternative configuration is to listen on a single IP address. This may be useful if you have a separate management network and wish to restrict control messages to it. It is important to ensure that the allowed_update_hosts (in the Global Settings resource) is compatible with the IP configured here. • Type: String • Required: false

Property	Description
	• Default value: "*"
cluster_comms_external_ip	This is the optional external ip of the traffic manager, which is used to circumvent natting when traffic managers in a cluster span different networks. • Type: String • Required: false • Default value: \<none\>
cluster_comms_port	The port that the software should listen on for internal administration communications. See also bind_ip. • Type: UInt • Required: false • Default value: "9080"
ec2_trafficips_public_enis	List of MAC addresses of interfaces which the traffic manager can use to associate the EC2 elastic IPs (Traffic IPs) to the instance. • Type: Set(String) • Required: false • Default value: \<none\>
fault_tolerance_bgp_router_id	The BGP router id If set to empty, then the IPv4 address used to communicate with the default IPv4 gateway is used instead. Specifying 0.0.0.0 will stop the traffic manager routing software from running the BGP protocol. • Type: String • Required: false • Default value: \<none\>

Property	Description
fault_tolerance_ospfv2_ip	The traffic manager's permanent IPv4 address which the routing software will use for peering and transit traffic, and as its OSPF router ID. If set to empty, then the address used to communicate with the default IPv4 gateway is used instead. Specifying 0.0.0.0 will stop the traffic manager routing software from running the OSPF protocol. • Type: String • Required: false • Default value: <none>
fault_tolerance_ospfv2_neighbor_addrs	The IP addresses of routers which are expected to be found as OSPFv2 neighbors of the traffic manager. A warning will be reported if some of the expected routers are not peered, and an error will be reported if none of the expected routers are peered. An empty list disables monitoring. The special value %gateway% is a placeholder for the default gateway. • Type: Set(String) • Required: false • Default value: "%gateway%"
iptables_config_enabled	Whether the Traffic Manager should configure the iptables built-in chains to call Traffic Manager defined rules (e.g. the IP transparency chain). This should only be disabled in case of conflict with other software that manages iptables, e.g. firewalls. When disabled, you will need to add rules manually to use these features - see the user manual for details. • Type: Boolean • Required: false • Default value: true
iptrans_fwmark	The netfilter forwarding mark to use for IP transparency rules

Property	Description
	• Type: UInt • Required: false • Default value: "320"
iptrans_iptables_enabled	Whether IP transparency may be used via netfilter/iptables. This requires the iptables socket extension. • Type: Boolean • Required: false • Default value: true
iptrans_routing_table	The special routing table ID to use for IP transparency rules • Type: UInt • Required: false • Default value: "320"
java_port	The port the Java Extension handler process should listen on. This port will be bound for localhost communications only. • Type: UInt • Required: false • Default value: "9060"
location	This is the location of the local traffic manager is in. • Type: String • Required: false • Default value: <none>
nameip	Replace Traffic Manager name with an IP address. • Type: String

Property	Description
	• Required: false • Default value: \<none>
num_aptimizer_threads	How many worker threads the Web Accelerator process should create to optimise content. By default, one thread will be created for each CPU on the system. • Type: UInt • Required: false • Default value: \<none>
num_children	The number of worker processes the software will run. By default, one child process will be created for each CPU on the system. You may wish to reduce this to effectively "reserve" CPU(s) for other processes running on the host system. • Type: UInt • Required: false • Default value: \<none>
numberofcpus	The number of Application Firewall decider process to run. • Type: UInt • Required: false • Default value: \<none>
remote_licensing_email_ address	The e-mail address sent as part of a remote licensing request. • Type: String • Required: false • Default value: \<none>
remote_licensing_message	A free-text field sent as part of a remote licensing request. • Type: String

Property	Description
	• Required: false • Default value: \<none\>
rest_api_bind_ips	A list of IP Addresses which the REST API will listen on for connections. The list should contain IP addresses (IPv4 or IPv6) or a single entry containing an asterisk (*). This indicates that the REST API should listen on all IP Addresses. • Type: Set(String) • Required: false • Default value: "*"
rest_api_port	The port on which the REST API should listen for requests. • Type: UInt • Required: false • Default value: "9070"
restserverport	The Application Firewall REST Internal API port, this port should not be accessed directly • Type: UInt • Required: false • Default value: \<none\>
snmp_allow	Restrict which IP addresses can access the SNMP command responder service. The value can be all, localhost, or a list of IP CIDR subnet masks. For example 10.100.0.0/16 would allow connections from any IP address beginning with 10.100. • Type: Set(String) • Required: false • Default value: "all"

Property	Description
snmp_auth_password	The authentication password. Required (minimum length 8 characters) if security_level includes authentication. • Type: Password • Required: false • Default value: \<none>
snmp_bind_ip	The IP address the SNMP service should bind its listen port to. The value * (asterisk) means SNMP will listen on all IP addresses. • Type: String • Required: false • Default value: "*"
snmp_community	The community string required for SNMPv1 and SNMPv2c commands. (If empty, all SNMPv1 and SNMPv2c commands will be rejected). • Type: String • Required: false • Default value: "public"
snmp_enabled	Whether or not the SNMP command responder service should be enabled on this traffic manager. • Type: Boolean • Required: false • Default value: false
snmp_hash_algorithm	The hash algorithm for authenticated SNMPv3 communications. • Type: Enum(String) • Required: false • Default value: "md5"

Property	Description
	• Permitted values: "md5": MD5 "sha1": SHA-1
snmp_port	The port the SNMP command responder service should listen on. The value default denotes port 161 if the software is running with root privileges, and 1161 otherwise. • Type: String • Required: false • Default value: "default"
snmp_priv_password	The privacy password. Required (minimum length 8 characters) if security_level includes privacy (message encryption). • Type: Password • Required: false • Default value: <none>
snmp_security_level	The security level for SNMPv3 communications. • Type: Enum(String) • Required: false • Default value: "noauthnopriv" • Permitted values: "authnopriv": Authentication only "authpriv": Authentication and Privacy "noauthnopriv": No Authentication, No Privacy
snmp_username	The username required for SNMPv3 commands. (If empty, all SNMPv3 commands will be rejected).

Property	Description
	• Type: String • Required: false • Default value: <none>

trafficip	A table mapping interfaces to networks, used by the traffic manager to select which interface to raise a Traffic IP on.
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • name (String): A network interface. (Required)
	
	• Sub keys:
	
	    • networks (Set(String)): A set of IP/masks to which the network interface maps. (Required)

trafficip_json	A Traffic Manager REST-compatible JSON representation of the "trafficip" table property. Use this field with the "vtm_traffic_manager_trafficip_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
updaterip	The Application Firewall Updater IP. • Type: String • Required: false • Default value: "0.0.0.0"

vtm_rule_authenticator

URI Endpoint: /api/tm/8.3/config/active/rule_authenticators

TrafficScript authenticators define remote authentication services that can be queried via a TrafficScript rule.

```
# Example usage of vtm_rule_authenticator resource...
resource "vtm_rule_authenticator" "my_rule_authenticator" {
    name = "MyRuleAuthenticator"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
host	The hostname or IP address of the remote authenticator. • Type: String • Required: false • Default value: \<none\>
ldap_attributes	A list of attributes to return from the search. If blank, no attributes will be returned. If set to '*' then all user attributes will be returned. • Type: Set(String) • Required: false • Default value: \<none\>
ldap_bind_dn	The distinguished name (DN) of the 'bind' user. The traffic manager will connect to the LDAP server as this user when searching for user records. • Type: String • Required: false • Default value: \<none\>
ldap_bind_password	The password for the bind user. • Type: Password

Property	Description
	• Required: false • Default value: <none>
ldap_filter	The filter used to locate the LDAP record for the user being authenticated. Any occurrences of '%u' in the filter will be replaced by the name of the user being authenticated. • Type: String • Required: false • Default value: <none>
ldap_filter_base_dn	The base distinguished name (DN) under which user records are located on the server. • Type: String • Required: false • Default value: <none>
ldap_ssl_cert	The SSL certificate that the traffic manager should use to validate the remote server. If no certificate is specified then no signature validation will be performed. • Type: Reference(config-ssl-cacrl) • Required: false • Default value: <none>
ldap_ssl_enabled	Whether or not to enable SSL encryption to the LDAP server. • Type: Boolean • Required: false • Default value: false
ldap_ssl_type	The type of LDAP SSL encryption to use. • Type: Enum(String)

Property	Description

	• Required: false
	
	• Default value: "ldaps"
	
	• Permitted values:
	
	       "ldaps": LDAPS
	
	       "starttls": Start TLS

note	A description of the authenticator. • Type: FreeformString • Required: false • Default value: \<none>
port	The port on which the remote authenticator should be contacted. • Type: UInt • Required: false • Default value: "389"

vtm_saml_trustedidp

URI Endpoint: /api/tm/8.3/config/active/saml/trustedidps

Configuration for SAML IDP trust relationships.

```
# Example usage of vtm_saml_trustedidp resource...
resource "vtm_saml_trustedidp" "my_saml_trustedidp" {
    certificate =
    entity_id =
    name = "MySamlTrustedidp"
    url =
}
```

Property	Description
name	Name of the object • Type: string • Required: true
add_zlib_header	Whether or not to add the zlib header when compressing the AuthnRequest • Type: Boolean • Required: false • Default value: false
certificate	The certificate used to verify Assertions signed by the identity provider • Type: String • Required: true • Default value: <none>
entity_id	The entity id of the IDP • Type: String • Required: true • Default value: <none>
strict_verify	Whether or not SAML responses will be verified strictly • Type: Boolean • Required: false • Default value: true
url	The IDP URL to which Authentication Requests should be sent • Type: String

Property	Description
	• Required: true • Default value: <none>

vtm_user_authenticator

URI Endpoint: /api/tm/8.3/config/active/user_authenticators

A user authenticator is used to allow access to the UI and REST API by querying a remote authentication service.

```
# Example usage of vtm_user_authenticator resource...
resource "vtm_user_authenticator" "my_user_authenticator" {
    name = "MyUserAuthenticator"
    type =
}
```

Property	Description
name	Name of the object • Type: string • Required: true
description	A description of the authenticator. • Type: String • Required: false • Default value: <none>
enabled	Whether or not this authenticator is enabled. • Type: Boolean • Required: false • Default value: false

Property	Description
ldap_base_dn	The base DN (Distinguished Name) under which directory searches will be applied. The entries for your users should all appear under this DN. An example of a typical base DN is: OU=users, DC=mycompany, DC=local • Type: String • Required: false • Default value: <none>
ldap_bind_dn	Template to construct the bind DN (Distinguished Name) from the username. The string %u will be replaced by the username. Examples: %u@mycompany.local for Active Directory or cn=%u, dc=mycompany, dc=local for both LDAP and Active Directory. • Type: String • Required: false • Default value: <none>

ldap_dn_method	The bind DN (Distinguished Name) for a user can either be searched for in the directory using the base distinguished name and filter values, or it can be constructed from the username.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "none"
	
	• Permitted values:
	
	    "construct": Construct
	
	    "none": No setting configured
	
	    "search": Search

Property	Description
ldap_fallback_group	If the group attribute is not defined, or returns no results for the user logging in, the group named here will be used. If not specified, users will be denied access to the traffic manager if no groups matching a Permission Group can be found for them in the directory. • Type: String • Required: false • Default value: \<none\>
ldap_filter	A filter that can be used to extract a unique user record located under the base DN (Distinguished Name). The string %u will be replaced by the username. This filter is used to find a user's bind DN when dn_method is set to "Search", and to extract group information if the group filter is not specified. Examples: sAMAccountName=%u for Active Directory, or uid=%u for some Unix LDAP schemas. • Type: String • Required: false • Default value: \<none\>
ldap_group_attribute	The LDAP attribute that gives a user's group. If there are multiple entries for the attribute all will be extracted and they'll be lexicographically sorted, then the first one to match a Permission Group name will be used. • Type: String • Required: false • Default value: \<none\>
ldap_group_field	The sub-field of the group attribute that gives a user's group. For example, if group_attribute is memberOf and this retrieves values of the form CN=mygroup, OU=groups, OU=users, DC=mycompany, DC=local you would set group_field to CN. If there are multiple matching fields only the first matching field will be used.

Property	Description
	• Type: String • Required: false • Default value: <none>
ldap_group_filter	If the user record returned by filter does not contain the required group information you may specify an alternative group search filter here. This will usually be required if you have Unix/POSIX-style user records. If multiple records are returned the list of group names will be extracted from all of them. The string %u will be replaced by the username. Example: (&(memberUid=%u)(objectClass=posixGroup)) • Type: String • Required: false • Default value: <none>
ldap_port	The port to connect to the LDAP server on. • Type: UInt • Required: false • Default value: "389"
ldap_search_dn	The bind DN (Distinguished Name) to use when searching the directory for a user's bind DN. You can leave this blank if it is possible to perform the bind DN search using an anonymous bind. • Type: String • Required: false • Default value: <none>
ldap_search_password	If binding to the LDAP server using search_dn requires a password, enter it here. • Type: Password • Required: false

Property	Description
	• Default value: <none>
ldap_server	The IP or hostname of the LDAP server. • Type: String • Required: false • Default value: <none>

ldap_ssl	The type of TLS encryption, if any, to use. Usually STARTTLS will be used with port 389, and LDAPS with port 636. A Certificate Authority that the LDAP server's certificate chains back to must be present in the "Admin Certificate Authorities and Certificate Revocation Lists Catalog" under "SSL catalogs", otherwise the connection will fail.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "none"
	
	• Permitted values:
	
	    "ldaps": LDAPS
	
	    "none": None
	
	    "starttls": STARTTLS

ldap_timeout	Connection timeout in seconds. • Type: UInt • Required: false • Default value: "30"
radius_fallback_group	If no group is found using the vendor and group identifiers, or the group found is not valid, the group specified here will be used. • Type: String • Required: false

Property	Description
	• Default value: <none>
radius_group_attribute	The RADIUS identifier for the attribute that specifies an account's group. May be left blank if fallback group is specified. • Type: UInt • Required: false • Default value: "1"
radius_group_vendor	The RADIUS identifier for the vendor of the RADIUS attribute that specifies an account's group. Leave blank if using a standard attribute (i.e. for Filter-Id set group_attribute to 11). • Type: UInt • Required: false • Default value: "7146"
radius_nas_identifier	This value is sent to the RADIUS server. • Type: String • Required: false • Default value: <none>
radius_nas_ip_address	This value is sent to the RADIUS server, if left blank the address of the interfaced used to connect to the server will be used. • Type: String • Required: false • Default value: <none>
radius_port	The port to connect to the RADIUS server on. • Type: UInt • Required: false

Property	Description
	• Default value: "1812"
radius_secret	Secret key shared with the RADIUS server. • Type: Password • Required: false • Default value: <none>
radius_server	The IP or hostname of the RADIUS server. • Type: String • Required: false • Default value: <none>
radius_timeout	Connection timeout in seconds. • Type: UInt • Required: false • Default value: "30"

tacacs_plus_auth_type	Authentication type to use.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "pap"
	
	• Permitted values:
	
	    "ascii": ASCII
	
	    "pap": PAP

tacacs_plus_fallback_group	If group_service is not used, or no group value is provided for the user by the TACACS+ server, the group specified here will be used. If this is not specified, users with no TACACS+ defined group will be denied access.

Property	Description
	• Type: String • Required: false • Default value: <none>
tacacs_plus_group_field	The TACACS+ "service" field that provides each user's group. • Type: String • Required: false • Default value: "permission-group"
tacacs_plus_group_service	The TACACS+ "service" that provides each user's group field. • Type: String • Required: false • Default value: "zeus"
tacacs_plus_port	The port to connect to the TACACS+ server on. • Type: UInt • Required: false • Default value: "49"
tacacs_plus_secret	Secret key shared with the TACACS+ server. • Type: Password • Required: false • Default value: <none>
tacacs_plus_server	The IP or hostname of the TACACS+ server. • Type: String • Required: false • Default value: <none>

Property	Description
tacacs_plus_timeout	Connection timeout in seconds. • Type: UInt • Required: false • Default value: "30"

type	The type and protocol used by this authentication service.
	
	• Type: Enum(String)
	
	• Required: true
	
	• Default value: \<none>
	
	• Permitted values:
	
	    "ldap": LDAP
	
	    "radius": RADIUS
	
	    "tacacs_plus": TACACS+

vtm_user_group

URI Endpoint: /api/tm/8.3/config/active/user_groups

Permission groups specify permissions for groups of users. These groups can be given read-write or read-only access to different parts of the configuration hierarchy. Each group will contain a table of permissions. Each table entry has a name that corresponds to a part of the configuration hierarchy, and a corresponding access level. The access level may have values of either none, ro (read only, this is the default), or full. Some permissions have sub-permissions, these are denoted by following the parent permission name with a colon (:) followed by the sub-permission name. The built-in admin group has a special permission key of all with the value full, this must not be altered for the admin group but can be used in other group configuration files to change the default permission level for the group.

```
# Example usage of vtm_user_group resource...
resource "vtm_user_group" "my_user_group" {
    name = "MyUserGroup"
}
```

Property	Description
name	Name of the object • Type: string • Required: true
description	A description for the group. • Type: String • Required: false • Default value: <none>
password_expire_time	Members of this group must renew their passwords after this number of days. To disable password expiry for the group set this to 0 (zero). Note that this setting applies only to local users. • Type: UInt • Required: false • Default value: <none>

permissions	A table defining which level of permission this group has for specific configuration elements.
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • name (String): Configuration element to which this group has a level of permission. (Required)
	
	• Sub keys:
	
	    • access_level (String): Permission level for the configuration element (none, ro or full) (Required)

Property	Description
permissions_json	A Traffic Manager REST-compatible JSON representation of the "permissions" table property. Use this field with the "vtm_user_group_permissions_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: \<none\>
timeout	Inactive UI sessions will timeout after this number of seconds. To disable inactivity timeouts for the group set this to 0 (zero). • Type: UInt • Required: false • Default value: "30"

vtm_virtual_server

URI Endpoint: /api/tm/8.3/config/active/virtual_servers

The conf/vservers directory contains configuration files that define virtual servers. The name of a file is the name of the virtual server it defines. Virtual servers can be configured under the Services > Virtual Servers section of the Admin Server UI or by using functions under the VirtualServer section of the SOAP API and CLI.

```
# Example usage of vtm_virtual_server resource...
resource "vtm_virtual_server" "my_virtual_server" {
    name = "MyVirtualServer"
    pool =
    port =
}
```

Property	Description
name	Name of the object • Type: string

Property	Description
	• Required: true
aptimizer_enabled	Whether the virtual server should optimize web content. • Type: Boolean • Required: false • Default value: false

aptimizer_profile	A table of Aptimizer profiles and the application scopes that apply to them.
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	  • name (String): The name of an Aptimizer acceleration profile. (Required)
	
	• Sub keys:
	
	  • urls (Set(String)): The application scopes which apply to the acceleration profile. (Required)

aptimizer_profile_json	A Traffic Manager REST-compatible JSON representation of the "aptimizer_profile" table property. Use this field with the "vtm_virtual_server_profile_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
auth_saml_idp	Name of the Trusted Identity Provider configuration to use. To create Identity Providers, please visit section Trusted Identity Providers • Type: String

Property	Description
	• Required: false • Default value: \<none>

auth_saml_nameid_format	The NameID format to request and expect from the identity provider.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "none"
	
	• Permitted values:
	
	    "emailaddress": emailAddress
	
	    "none": none
	
	    "unspecified": unspecified

auth_saml_sp_acs_url	The 'Assertion Consumer Service' endpoint for the SAML service provider on this virtual server, ie the endpoint to which the identity provider will cause the user agent to send SAML assertions. This should be an HTTPS URL, must be in the same cookie domain as all hostnames used by the end user to access the virtual server (see cookie configuration) and the port must be the port on which this virtual server is listening. It must match the URI placed by the identity provider in the 'Recipient' attribute in the SAML assertion, if present. • Type: String • Required: false • Default value: \<none>
auth_saml_sp_entity_id	The entity ID to be used by the SAML service provider function on this virtual server. This should usually be a URL, or a URN, however it may be any string. It must match the entity ID placed by the identity provider in the 'Audience' field in the SAML assertion.

Property	Description
	• Type: String • Required: false • Default value: <none>
auth_saml_time_tolerance	Time tolerance on authentication checks. When checking time-stamps and expiry dates against the current time on the system, allow a tolerance of this many seconds. For example, if a SAML response contains a 'NotOnOrAfter' that is 4 seconds in the past according to the local time, and the tolerance is set to 5 seconds, it will still be accepted. This is to prevent a lack of clock synchronization from resulting in rejection of SAML responses. • Type: UInt • Required: false • Default value: "5"
auth_session_cookie_attributes	Attributes of cookie used for authentication session. • Type: String • Required: false • Default value: "HttpOnly; SameSite=Strict"
auth_session_cookie_name	Name of cookie used for authentication session. • Type: String • Required: false • Default value: "VS_SamlSP_Auth"
auth_session_log_external_state	Whether or not to include state of authentication sessions stored encrypted on the client as plaintext in the logs. • Type: Boolean • Required: false

Property	Description
	• Default value: false
auth_session_timeout	Timeout on authentication session. • Type: UInt • Required: false • Default value: "7200"

auth_type	Type of authentication to apply to requests to the virtual server.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "none"
	
	• Permitted values:
	
	    "none": None
	
	    "saml_sp": SAML Service Provider

auth_verbose	Whether or not detailed messages about virtual server authentication should be written to the error log. • Type: Boolean • Required: false • Default value: false
bandwidth_class	The bandwidth management class that this server should use, if any. • Type: Reference(config-bandwidth) • Required: false • Default value: <none>
completion_rules	Rules that are run at the end of a transaction, in order, comma separated.

Property	Description
	• Type: List(String) • Required: false • Default value: <none>
connect_timeout	The time, in seconds, for which an established connection can remain idle waiting for some initial data to be received from the client. The initial data is defined as a complete set of request headers for HTTP, SIP and RTSP services, or the first byte of data for all other services. A value of 0 will disable the timeout. • Type: UInt • Required: false • Default value: "10"
connection_errors_error_file	The error message to be sent to the client when the traffic manager detects an internal or backend error for the virtual server. • Type: Reference(config-extra-file) • Required: false • Default value: "Default"
connection_keepalive	Whether or not the virtual server should use keepalive connections with the remote clients. • Type: Boolean • Required: false • Default value: true
connection_keepalive_ timeout	The length of time that the virtual server should keep an idle keepalive connection before discarding it. A value of 0 (zero) will mean that the keepalives are never closed by the traffic manager. • Type: UInt • Required: false

Property	Description
	• Default value: "10"
connection_max_client_buffer	The amount of memory, in bytes, that the virtual server should use to store data sent by the client through one TCP connection or HTTP/2 stream. Larger values will use more memory, but will minimise the number of read() and write() system calls that the traffic manager must perform. • Type: UInt • Required: false • Default value: "65536"
connection_max_server_buffer	The amount of memory, in bytes, that the virtual server should use to store data returned by the server through one TCP connection. Larger values will use more memory, but will minimise the number of read() and write() system calls that the traffic manager must perform. • Type: UInt • Required: false • Default value: "65536"
connection_max_transaction_duration	The total amount of time a transaction can take, counted from the first byte being received until the transaction is complete. For HTTP, this can mean all data has been written in both directions, or the connection has been closed; in most other cases it is the same as the connection being closed. The default value of 0 means there is no maximum duration, i.e., transactions can take arbitrarily long if none of the other timeouts occur. • Type: UInt • Required: false • Default value: <none>

Property	Description
connection_server_first_banner	If specified, the traffic manager will use the value as the banner to send for server-first protocols such as FTP, POP, SMTP and IMAP. This allows rules to use the first part of the client data (such as the username) to select a pool. The banner should be in the correct format for the protocol, e.g. for FTP it should start with "220 " • Type: String • Required: false • Default value: <none>
connection_timeout	A connection should be closed if no additional data has been received for this period of time. A value of 0 (zero) will disable this timeout. • Type: UInt • Required: false • Default value: "300"

cookie_domain	The way in which the traffic manager should rewrite the domain portion of any cookies set by a back-end web server.
	
	• Type: Enum(UInt)
	
	• Required: false
	
	• Default value: "no_rewrite"
	
	• Permitted values:
	
	    "no_rewrite": Do not rewrite the domain
	
	    "set_to_named": Rewrite the domain to the named domain value
	
	    "set_to_request": Rewrite the domain to the host header of the request

Property	Description
cookie_new_domain	The domain to use when rewriting a cookie's domain to a named value. • Type: String • Required: false • Default value: \<none>
cookie_path_regex	If you wish to rewrite the path portion of any cookies set by a back-end web server, provide a regular expression to match the path: • Type: String • Required: false • Default value: \<none>
cookie_path_replace	If cookie path regular expression matches, it will be replaced by this substitution. Parameters $1-$9 can be used to represent bracketed parts of the regular expression. • Type: String • Required: false • Default value: \<none>

cookie_secure	Whether or not the traffic manager should modify the "secure" tag of any cookies set by a back-end web server.
	
	• Type: Enum(UInt)
	
	• Required: false
	
	• Default value: "no_modify"
	
	• Permitted values:
	
	   "no_modify": Do not modify the 'secure' tag
	
	   "set_secure": Set the 'secure' tag

Property	Description
	"unset_secure": Unset the 'secure' tag
dns_edns_client_subnet	Enable/Disable use of EDNS client subnet option • Type: Boolean • Required: false • Default value: true
dns_edns_udpsize	EDNS UDP size advertised in responses. • Type: UInt • Required: false • Default value: "4096"
dns_max_udpsize	Maximum UDP answer size. • Type: UInt • Required: false • Default value: "4096"

dns_rrset_order	Response record ordering.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "fixed"
	
	• Permitted values:
	
	    "cyclic": Cyclic
	
	    "fixed": Fixed

dns_verbose	Whether or not the DNS Server should emit verbose logging. This is useful for diagnosing problems. • Type: Boolean

Property	Description
	• Required: false • Default value: false
dns_zones	The DNS zones • Type: Set(String) • Required: false • Default value: \<none\>
enabled	Whether the virtual server is enabled. • Type: Boolean • Required: false • Default value: false
ftp_data_source_port	The source port to be used for active-mode FTP data connections. If 0, a random high port will be used, otherwise the specified port will be used. If a port below 1024 is required you must first explicitly permit use of low ports with the data_bind_low global setting. • Type: UInt • Required: false • Default value: \<none\>
ftp_force_client_secure	Whether or not the virtual server should require that incoming FTP data connections from the client originate from the same IP address as the corresponding client control connection. • Type: Boolean • Required: false • Default value: true

Property	Description
ftp_force_server_secure	Whether or not the virtual server should require that incoming FTP data connections from the nodes originate from the same IP address as the node. • Type: Boolean • Required: false • Default value: true
ftp_port_range_high	If non-zero, then this controls the upper bound of the port range to use for FTP data connections. • Type: UInt • Required: false • Default value: <none>
ftp_port_range_low	If non-zero, then this controls the lower bound of the port range to use for FTP data connections. • Type: UInt • Required: false • Default value: <none>
ftp_ssl_data	Use SSL on the data connection as well as the control connection (if not enabled it is left to the client and server to negotiate this). • Type: Boolean • Required: false • Default value: true
glb_services	The associated GLB services for this DNS virtual server. • Type: Set(String) • Required: false • Default value: <none>

Property	Description
gzip_compress_level	Compression level (1-9, 1=low, 9=high). • Type: UInt • Required: false • Default value: "1"
gzip_enabled	Compress web pages sent back by the server. • Type: Boolean • Required: false • Default value: false

gzip_etag_rewrite	How the ETag header should be manipulated when compressing content.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "wrap"
	
	• Permitted values:
	
	    "delete": Delete the ETag header
	
	    "ignore": Leave the ETag unchanged
	
	    "weaken": Change the ETag header to specify a weak match
	
	    "wrap": Wrap the ETag, and attempt to unwrap safe conditional requests

gzip_include_mime	MIME types to compress. Complete MIME types can be used, or a type can end in a '*' to match multiple types. • Type: Set(String) • Required: false

Property	Description
	• Default value: "text/html text/plain"
gzip_max_size	Maximum document size to compress (0 means unlimited). • Type: UInt • Required: false • Default value: "10000000"
gzip_min_size	Minimum document size to compress. • Type: UInt • Required: false • Default value: "1000"
gzip_no_size	Compress documents with no given size. • Type: Boolean • Required: false • Default value: true
http2_connect_timeout	The time, in seconds, to wait for a request on a new HTTP/2 connection. If no request is received within this time, the connection will be closed. This setting overrides the connect_timeout setting. If set to 0 (zero), the value of connect_timeout will be used instead. • Type: UInt • Required: false • Default value: <none>
http2_data_frame_size	This setting controls the preferred frame size used when sending body data to the client. If the client specifies a smaller maximum size than this setting, the client's maximum size will be used. Every data frame sent has at least a 9-byte header, in addition to this frame size, prepended to it.

Property	Description
	• Type: UInt • Required: false • Default value: "4096"
http2_enabled	This setting allows the HTTP/2 protocol to be used by a HTTP virtual server. Unless use of HTTP/2 is negotiated by the client, the virtual server will fall back to HTTP 1.x automatically. • Type: Boolean • Required: false • Default value: true
http2_header_table_size	This setting controls the amount of memory allowed for header compression on each HTTP/2 connection. • Type: UInt • Required: false • Default value: "4096"
http2_headers_index_ blacklist	A list of header names that should never be compressed using indexing. • Type: Set(String) • Required: false • Default value: <none>
http2_headers_index_ default	The HTTP/2 HPACK compression scheme allows for HTTP headers to be compressed using indexing. Sensitive headers can be marked as "never index", which prevents them from being compressed using indexing. When this setting is Yes, only headers included in http2!headers_index_blacklist are marked as "never index". When this setting is No, all headers will be marked as "never index" unless they are included in http2!headers_index_whitelist.

Property	Description
	• Type: Boolean
	• Required: false
	• Default value: true
http2_headers_index_ whitelist	A list of header names that can be compressed using indexing when the value of http2!headers_index_default is set to No.
	• Type: Set(String)
	• Required: false
	• Default value: <none>
http2_headers_size_limit	The maximum size, in bytes, of decompressed headers for an HTTP/2 request. If the limit is exceeded, the connection on which the request was sent will be dropped. A value of 0 disables the limit check. If a service protection class with http!max_header_length configured is associated with this service then that setting will take precedence.
	• Type: UInt
	• Required: false
	• Default value: "262144"
http2_http2_client_buffer_ multiplier	The amount of memory, in multiples of the value specified by max_ client_buffer, that the virtual server should use to store data sent by a client through a HTTP/2 connection. The value specified can be between 0 and 200. The value of 0 means unlimited. This setting limits buffer size for a HTTP/2 connection and does not affect buffer size for HTTP/1 connections or TCP stream connections. The number of HTTP/2 streams that can be opened in a single HTTP/2 connection is given by the http2!max_concurrent_streams. An overall cap to the amount of memory allocated for buffers for all TCP connections is given by the global max_tcp_buff_mem setting.
	• Type: UInt

Property	Description
	• Required: false • Default value: \<none>
http2_http2_server_buffer_multiplier	The amount of memory, in multiples of the value specified by max_server_buffer, that the virtual server should use to store data sent to a client through HTTP/2 connection. The value specified can be between 0 and 200. The value of 0 means unlimited. This setting limits buffer size for a HTTP/2 connection and does not affect buffer size for HTTP/1 connections or TCP stream connections. The number of HTTP/2 streams that can be opened in a single HTTP/2 connection is given by the http2!max_concurrent_streams. An overall cap to the amount of memory allocated for buffers for all TCP connections is given by the global max_tcp_buff_mem setting. • Type: UInt • Required: false • Default value: \<none>
http2_idle_timeout_no_streams	The time, in seconds, to wait for a new HTTP/2 request on a previously used HTTP/2 connection that has no open HTTP/2 streams. If an HTTP/2 request is not received within this time, the connection will be closed. A value of 0 (zero) will disable the timeout. • Type: UInt • Required: false • Default value: "120"
http2_idle_timeout_open_streams	The time, in seconds, to wait for data on an idle HTTP/2 connection, which has open streams, when no data has been sent recently (e.g. for long-polled requests). If data is not sent within this time, all open streams and the HTTP/2 connection will be closed. A value of 0 (zero) will disable the timeout. • Type: UInt • Required: false

Property	Description
	• Default value: "600"
http2_max_concurrent_ streams	This setting controls the number of streams a client is permitted to open concurrently on a single connection. • Type: UInt • Required: false • Default value: "200"
http2_max_frame_size	This setting controls the maximum HTTP/2 frame size clients are permitted to send to the traffic manager. • Type: UInt • Required: false • Default value: "16384"
http2_max_header_padding	The maximum size, in bytes, of the random-length padding to add to HTTP/2 header frames. The padding, a random number of zero bytes up to the maximum specified. • Type: UInt • Required: false • Default value: <none>
http2_merge_cookie_ headers	Whether Cookie headers received from an HTTP/2 client should be merged into a single Cookie header using RFC6265 rules before forwarding to an HTTP/1.1 server. Some web applications do not handle multiple Cookie headers correctly. • Type: Boolean • Required: false • Default value: true

Property	Description
http2_stream_window_size	This setting controls the flow control window for each HTTP/2 stream. This will limit the memory used for buffering when the client is sending body data faster than the pool node is reading it. • Type: UInt • Required: false • Default value: "65535"
http_add_cluster_ip	Whether or not the virtual server should add an "X-Cluster-Client-Ip" header to the request that contains the remote client's IP address. • Type: Boolean • Required: false • Default value: true
http_add_x_forwarded_for	Whether or not the virtual server should append the remote client's IP address to the X-Forwarded-For header. If the header does not exist, it will be added. • Type: Boolean • Required: false • Default value: false
http_add_x_forwarded_proto	Whether or not the virtual server should add an "X-Forwarded-Proto" header to the request that contains the original protocol used by the client to connect to the traffic manager. • Type: Boolean • Required: false • Default value: false

Property	Description
http_autodetect_upgrade_headers	Whether the traffic manager should check for HTTP responses that confirm an HTTP connection is transitioning to the WebSockets protocol. If that such a response is detected, the traffic manager will cease any protocol-specific processing on the connection and just pass incoming data to the client/server as appropriate. • Type: Boolean • Required: false • Default value: true

http_chunk_overhead_forwarding	Handling of HTTP chunk overhead. When vTM receives data from a server or client that consists purely of protocol overhead (contains no payload), forwarding of such segments is delayed until useful payload data arrives (setting "lazy"). Changing this key to "eager" will make vTM incur the overhead of immediately passing such data on; it should only be used with HTTP peers whose chunk handling requires it.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "lazy"
	
	• Permitted values:
	
	    "eager": Forward all data, even when no new payload information is available.
	
	    "lazy": Only forward segments when useful payload data is available.

http_location_regex	If the 'Location' header matches this regular expression, rewrite the header using the 'location_replace' pattern. • Type: String • Required: false

Property	Description
	• Default value: <none>
http_location_replace	If the 'Location' header matches the 'location_regex' regular expression, rewrite the header with this pattern (parameters such as $1-$9 can be used to match parts of the regular expression): • Type: String • Required: false • Default value: <none>

http_location_rewrite	The action the virtual server should take if the "Location" header does not match the location_regex regular expression.
	
	• Type: Enum(UInt)
	
	• Required: false
	
	• Default value: "if_host_matches"
	
	• Permitted values:
	
	    "always": Rewrite the hostname to the request's "Host" header, and rewrite the protocol and port if necessary;
	
	    "if_host_matches": Do not rewrite the hostname. Rewrite the protocol and port if the hostname matches the request's "Host" header.
	
	    "never": Nothing;

http_mime_default	Auto-correct MIME types if the server sends the "default" MIME type for files. • Type: String • Required: false • Default value: "text/plain"

Property	Description
http_mime_detect	Auto-detect MIME types if the server does not provide them. • Type: Boolean • Required: false • Default value: false
http_strip_x_forwarded_ proto	Whether or not the virtual server should strip the 'X-Forwarded-Proto' header from incoming requests. • Type: Boolean • Required: false • Default value: true
kerberos_protocol_ transition_enabled	Whether or not the virtual server should use Kerberos Protocol Transition. • Type: Boolean • Required: false • Default value: false
kerberos_protocol_ transition_principal	The Kerberos principal this virtual server should use to perform Kerberos Protocol Transition. • Type: String • Required: false • Default value: <none>
kerberos_protocol_ transition_target	The Kerberos principal name of the service this virtual server targets. • Type: String • Required: false • Default value: <none>
listen_on_any	Whether to listen on all IP addresses

Property	Description
	• Type: Boolean • Required: false • Default value: true
listen_on_hosts	Hostnames and IP addresses to listen on • Type: Set(String) • Required: false • Default value: \<none\>
listen_on_traffic_ips	Traffic IP Groups to listen on • Type: Set(String) • Required: false • Default value: \<none\>
log_client_connection_ failures	Should the virtual server log failures occurring on connections to clients. • Type: Boolean • Required: false • Default value: false
log_enabled	Whether or not to log connections to the virtual server to a disk on the file system. • Type: Boolean • Required: false • Default value: false
log_filename	The name of the file in which to store the request logs. The filename can contain macros which will be expanded by the traffic manager to generate the full filename.

Property	Description
	• Type: String • Required: false • Default value: "%zeushome%/zxtm/log/%v.log"
log_format	The log file format. This specifies the line of text that will be written to the log file when a connection to the traffic manager is completed. Many parameters from the connection can be recorded using macros. • Type: String • Required: false • Default value: "%h %l %u %t "%r" %s %b "%{Referer}i" "%{User-agent}i""
log_save_all	Whether to log all connections by default, or log no connections by default. Specific connections can be selected for addition to or exclusion from the log using the TrafficScript function requestlog.include(). • Type: Boolean • Required: false • Default value: true
log_server_connection_ failures	Should the virtual server log failures occurring on connections to nodes. • Type: Boolean • Required: false • Default value: false
log_session_persistence_ verbose	Should the virtual server log session persistence events. • Type: Boolean • Required: false

Property	Description
	• Default value: false
log_ssl_failures	Should the virtual server log failures occurring on SSL secure negotiation. • Type: Boolean • Required: false • Default value: false
log_ssl_resumption_failures	Should the virtual server log messages when attempts to resume SSL sessions (either from the session cache or a session ticket) fail. Note that failure to resume an SSL session does not result in the SSL connection being closed, but it does cause a full SSL handshake to take place. • Type: Boolean • Required: false • Default value: false
max_concurrent_ connections	The maximum number of concurrent TCP connections that will be handled by this virtual server. If set to a non-zero value, the traffic manager will limit the number of concurrent TCP connections that this virtual server will accept to the value specified. When the limit is reached, new connections to this virtual server will not be accepted. If set to 0 the number of concurrent TCP connections will not be limited. • Type: UInt • Required: false • Default value: <none>
note	A description for the virtual server. • Type: FreeformString • Required: false

Property	Description
	• Default value: \<none\>
pool	The default pool to use for traffic. • Type: Reference(config-pool) • Required: true • Default value: \<none\>
port	The port on which to listen for incoming connections. • Type: UInt • Required: true • Default value: \<none\>
protection_class	The service protection class that should be used to protect this server, if any. • Type: String • Required: false • Default value: \<none\>

protocol	The protocol that the virtual server is using.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "http"
	
	• Permitted values:
	
	      "client_first": Generic client first
	
	      "dns": DNS (UDP)
	
	      "dns_tcp": DNS (TCP)
	
	      "ftp": FTP

Property	Description
	"http": HTTP
	"https": SSL (HTTPS)
	"imaps": SSL (IMAPS)
	"imapv2": IMAPv2
	"imapv3": IMAPv3
	"imapv4": IMAPv4
	"ldap": LDAP
	"ldaps": SSL (LDAPS)
	"pop3": POP3
	"pop3s": SSL (POP3S)
	"rtsp": RTSP
	"server_first": Generic server first
	"siptcp": SIP (TCP)
	"sipudp": SIP (UDP)
	"smtp": SMTP
	"ssl": SSL
	"stream": Generic streaming
	"telnet": Telnet
	"udp": UDP
	"udpstreaming": UDP - Streaming

Property	Description
proxy_protocol	Expect connections to the traffic manager to be prefixed with a PROXY protocol header. If enabled, the information contained in the PROXY header will be available in TrafficScript. Connections that are not prefixed with a valid PROXY protocol header will be discarded. • Type: Boolean • Required: false • Default value: false
recent_connections_enabled	Whether or not connections handled by this virtual server should be shown on the Activity > Connections page. • Type: Boolean • Required: false • Default value: true
recent_connections_save_all	Whether or not all connections handled by this virtual server should be shown on the Connections page. Individual connections can be selectively shown on the Connections page using the recentconns.include() TrafficScript function. • Type: Boolean • Required: false • Default value: false
request_rules	Rules to be applied to incoming requests, in order, comma separated. • Type: List(String) • Required: false • Default value: <none>
request_tracing_enabled	Record a trace of major connection processing events for each request and response.

Property	Description
	• Type: Boolean • Required: false • Default value: false
request_tracing_trace_io	Include details of individual I/O events in request and response traces. Requires request tracing to be enabled. • Type: Boolean • Required: false • Default value: false
response_rules	Rules to be applied to responses, in order, comma separated. • Type: List(Reference(config-trafficscript)) • Required: false • Default value: <none>
rtsp_streaming_port_range_ high	If non-zero this controls the upper bound of the port range to use for streaming data connections. • Type: UInt • Required: false • Default value: <none>
rtsp_streaming_port_range_ low	If non-zero this controls the lower bound of the port range to use for streaming data connections. • Type: UInt • Required: false • Default value: <none>
rtsp_streaming_timeout	If non-zero data-streams associated with RTSP connections will timeout if no data is transmitted for this many seconds.

Property	Description
	• Type: UInt • Required: false • Default value: "30"
sip_dangerous_requests	The action to take when a SIP request with body data arrives that should be routed to an external IP. • Type: Enum(String) • Required: false • Default value: "node" • Permitted values: "forbid": Send a 403 Forbidden response to the client "forward": Forward the request to its target URI (dangerous) "node": Send the request to a back-end node
sip_follow_route	Should the virtual server follow routing information contained in SIP requests. If set to No requests will be routed to the chosen back-end node regardless of their URI or Route header. • Type: Boolean • Required: false • Default value: true
sip_max_connection_mem	SIP clients can have several pending requests at one time. To protect the traffic manager against DoS attacks, this setting limits the amount of memory each client can use. When the limit is reached new requests will be sent a 413 response. If the value is set to 0 (zero) the memory limit is disabled.

Property	Description
	• Type: UInt • Required: false • Default value: "65536"
sip_mode	The mode that this SIP virtual server should operate in. • Type: Enum(String) • Required: false • Default value: "sip_gateway" • Permitted values: "full_gateway": All SIP requests and responses and all session data will pass through vTM. A port range to use for the session data and a timeout value for inactive data connections can be specified in the additional settings that are displayed when the Full Gateway mode is selected. "route": The first SIP request in a session will pass through vTM, along with its responses, but all future requests that are part of the same session will go directly to the back-end node that was chosen by the traffic manager. "sip_gateway": All SIP requests and responses will pass through the traffic manager.
sip_rewrite_uri	Replace the Request-URI of SIP requests with the address of the selected back-end node. • Type: Boolean • Required: false

Property	Description
	• Default value: false
sip_streaming_port_range_high	If non-zero this controls the upper bound of the port range to use for streaming data connections. • Type: UInt • Required: false • Default value: \<none\>
sip_streaming_port_range_low	If non-zero, then this controls the lower bound of the port range to use for streaming data connections. • Type: UInt • Required: false • Default value: \<none\>
sip_streaming_timeout	If non-zero a UDP stream will timeout when no data has been seen within this time. • Type: UInt • Required: false • Default value: "60"
sip_timeout_messages	When timing out a SIP transaction, send a 'timed out' response to the client and, in the case of an INVITE transaction, a CANCEL request to the server. • Type: Boolean • Required: false • Default value: true
sip_transaction_timeout	The virtual server should discard a SIP transaction when no further messages have been seen within this time. • Type: UInt

Property	Description
	• Required: false • Default value: "30"
sip_udp_associate_by_ source	Require that SIP datagrams which are part of the same transaction are received from the same address and port. • Type: Boolean • Required: false • Default value: true
slm_class	The service level monitoring class that this server should use, if any. • Type: Reference(config-slm) • Required: false • Default value: <none>
smtp_expect_starttls	Whether or not the traffic manager should expect the connection to start off in plain text and then upgrade to SSL using STARTTLS when handling SMTP traffic. • Type: Boolean • Required: false • Default value: true
ssl_add_http_headers	Whether or not the virtual server should add HTTP headers to each request to show the SSL connection parameters. • Type: Boolean • Required: false • Default value: false

Property	Description
ssl_cipher_suites	The SSL/TLS cipher suites to allow for connections to this virtual server. Leaving this empty will make the virtual server use the globally configured cipher suites, see configuration key ssl!cipher_suites in the Global Settings section of the System tab. See there for how to specify SSL/TLS cipher suites. • Type: String • Required: false • Default value: <none>
ssl_client_cert_cas	The certificate authorities that this virtual server should trust to validate client certificates. If no certificate authorities are selected, and client certificates are requested, then all client certificates will be accepted. • Type: Set(String) • Required: false • Default value: <none>

ssl_client_cert_headers	What HTTP headers the virtual server should add to each request to show the data in the client certificate.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "none"
	
	• Permitted values:
	
	    "all": Certificate fields and certificate text
	
	    "none": No data
	
	    "simple": Certificate fields

ssl_decrypt	Whether or not the virtual server should decrypt incoming SSL traffic.

Property	Description
	• Type: Boolean • Required: false • Default value: false
ssl_elliptic_curves	The SSL elliptic curve preference list for SSL connections to this virtual server using TLS version 1.0 or higher. Leaving this empty will make the virtual server use the globally configured curve preference list. The named curves P256, P384 and P521 may be configured. • Type: List(String) • Required: false • Default value: <none>

ssl_honor_fallback_scsv	Whether or not the Fallback SCSV sent by TLS clients is honored by this virtual server. Choosing the global setting means the value of configuration key ssl!honor_fallback_scsv from the Global Settings section of the System tab will be enforced.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "use_default"
	
	• Permitted values:
	
	    "disabled": Disable Fallback SCSV
	
	    "enabled": Enable Fallback SCSV
	
	    "use_default": Use the global setting for Fallback SCSV

ssl_issued_certs_never_expire	When the virtual server verifies certificates signed by these certificate authorities, it doesn't check the 'not after' date, i.e., they are considered valid even after their expiration date has passed (but not if they have been revoked).

Property	Description
	• Type: Set(String) • Required: false • Default value: <none>
ssl_issued_certs_never_ expire_depth	This setting gives the number of certificates in a certificate chain beyond those listed as issued_certs_never_expire whose certificate expiry will not be checked. For example "0" will result in the expiry checks being made for certificates issued by issued_certs_never_expire certificates, "1" will result in no expiry checks being performed for the certificates directly issued by issued_certs_never_expire certificates, "2" will avoid checking expiry for certificates issued by certificates issued by the issued_certs_never_expire certificates as well, and so on. • Type: UInt • Required: false • Default value: "1"
ssl_ocsp_enable	Whether or not the traffic manager should use OCSP to check the revocation status of client certificates. • Type: Boolean • Required: false • Default value: false
ssl_ocsp_issuers	A table of certificate issuer specific OCSP settings. • Type: Table • Required: false • Primary key: • issuer (String): The name of an issuer (or DEFAULT for default OCSP settings). (Required)

Property	Description

	• Sub keys:
	
	    • aia (Boolean): Whether the traffic manager should use AIA information contained in a client certificate to determine which OCSP responder to contact.
	
	    • nonce (Enum(String)): How to use the OCSP nonce extension, which protects against OCSP replay attacks. Some OCSP servers do not support nonces.
	
	    Permitted values:
	
	"off": No nonce check
	
	"on": Use nonce, server does not have to reply with nonce
	
	"strict": Use nonce, server must reply with nonce
	
	    • required (Enum(String)): Whether we should do an OCSP check for this issuer, and whether it is required or optional.
	
	    Permitted values:
	
	"none": None
	
	"optional": OCSP check optional
	
	"strict": OCSP check required
	
	    • responder_cert (String): The expected responder certificate.
	
	    • signer (String): The certificate with which to sign the request, if any.
	
	    • url (String): Which OCSP responders this virtual server should use to verify client certificates.

Property	Description
ssl_ocsp_issuers_json	A Traffic Manager REST-compatible JSON representation of the "ssl_ocsp_issuers" table property. Use this field with the "vtm_virtual_server_ocsp_issuers_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: \<none>
ssl_ocsp_max_response_age	The number of seconds for which an OCSP response is considered valid if it has not yet exceeded the time specified in the 'nextUpdate' field. If set to 0 (zero) then OCSP responses are considered valid until the time specified in their 'nextUpdate' field. • Type: UInt • Required: false • Default value: \<none>
ssl_ocsp_stapling	If OCSP URIs are present in certificates used by this virtual server, then enabling this option will allow the traffic manager to provide OCSP responses for these certificates as part of the handshake, if the client sends a TLS status_request extension in the ClientHello. • Type: Boolean • Required: false • Default value: false
ssl_ocsp_time_tolerance	The number of seconds outside the permitted range for which the 'thisUpdate' and 'nextUpdate' fields of an OCSP response are still considered valid. • Type: UInt • Required: false • Default value: "30"

Property	Description
ssl_ocsp_timeout	The number of seconds after which OCSP requests will be timed out. • Type: UInt • Required: false • Default value: "10"

ssl_request_client_cert	Whether or not the virtual server should request an identifying SSL certificate from each client.
	
	• Type: Enum(UInt)
	
	• Required: false
	
	• Default value: "dont_request"
	
	• Permitted values:
	
	    "dont_request": Do not request a client certificate
	
	    "request": Request, but do not require a client certificate
	
	    "require": Require a client certificate

ssl_send_close_alerts	Whether or not to send an SSL/TLS "close alert" when the traffic manager is initiating an SSL socket disconnection. • Type: Boolean • Required: false • Default value: true
ssl_server_cert_alt_certificates	The SSL certificates and corresponding private keys. • Type: List(String) • Required: false • Default value: <none>

Property	Description
ssl_server_cert_default	The default SSL certificate to use for this virtual server. • Type: String • Required: false • Default value: <none>

ssl_server_cert_host_mapping	Host specific SSL server certificate mappings.
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	  • host (String): Host which this entry refers to. (Required)
	
	• Sub keys:
	
	  • certificate (String): The SSL server certificate for a particular destination site IP. (Required)
	
	  • alt_certificates (List(String)): The SSL server certificates for a particular destination site IP.

ssl_server_cert_host_mapping_json	A Traffic Manager REST-compatible JSON representation of the "ssl_server_cert_host_mapping" table property. Use this field with the "vtm_virtual_server_server_cert_host_mapping_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>
ssl_session_cache_enabled	Whether or not use of the session cache is enabled for this virtual server. Choosing the global setting means the value of configuration key ssl!session_cache_enabled from the Global Settings section of the System tab will be enforced. • Type: Enum(String)

Property	Description
	• Required: false • Default value: "use_default" • Permitted values: "disabled": Disable use of the session cache "enabled": Enable use of the session cache "use_default": Use the global setting for use of the session cache
ssl_session_tickets_enabled	Whether or not use of session tickets is enabled for this virtual server. Choosing the global setting means the value of configuration key ssl!tickets!enabled from the Global Settings section of the System tab will be enforced. • Type: Enum(String) • Required: false • Default value: "use_default" • Permitted values: "disabled": Disable use of the session tickets "enabled": Enable use of the session tickets "use_default": Use the global setting for use of session tickets
ssl_signature_algorithms	The SSL signature algorithms preference list for SSL connections to this virtual server using TLS version 1.2 or higher. Leaving this empty will make the virtual server use the globally configured preference list, signature_algorithms in the ssl section of the global_settings resource. See there and in the online help for how to specify SSL signature algorithms. • Type: String

Property	Description
	• Required: false • Default value: <none>

ssl_support_ssl3	Whether or not SSLv3 is enabled for this virtual server. Choosing the global setting means the value of configuration key ssl!support_ssl3 from the Global Settings section of the System tab will be enforced.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "use_default"
	
	• Permitted values:
	
	    "disabled": Disable SSLv3
	
	    "enabled": Enable SSLv3
	
	    "use_default": Use the global setting for SSLv3

ssl_support_tls1	Whether or not TLSv1.0 is enabled for this virtual server. Choosing the global setting means the value of configuration key ssl!support_tls1 from the Global Settings section of the System tab will be enforced.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "use_default"
	
	• Permitted values:
	
	    "disabled": Disable TLSv1.0
	
	    "enabled": Enable TLSv1.0
	
	    "use_default": Use the global setting for TLSv1.0

Property	Description
ssl_support_tls1_1	Whether or not TLSv1.1 is enabled for this virtual server. Choosing the global setting means the value of configuration key ssl!support_tls1_1 from the Global Settings section of the System tab will be enforced. • Type: Enum(String) • Required: false • Default value: "use_default" • Permitted values: "disabled": Disable TLSv1.1 "enabled": Enable TLSv1.1 "use_default": Use the global setting for TLSv1.1
ssl_support_tls1_2	Whether or not TLSv1.2 is enabled for this virtual server. Choosing the global setting means the value of configuration key ssl!support_tls1_2 from the Global Settings section of the System tab will be enforced. • Type: Enum(String) • Required: false • Default value: "use_default" • Permitted values: "disabled": Disable TLSv1.2 "enabled": Enable TLSv1.2 "use_default": Use the global setting for TLSv1.2

Property	Description

ssl_support_tls1_3	Whether or not TLSv1.3 is enabled for this virtual server. Choosing the global setting means the value of configuration key ssl!support_tls1_3 from the Global Settings section of the System tab will be enforced.
	
	• Type: Enum(String)
	
	• Required: false
	
	• Default value: "use_default"
	
	• Permitted values:
	
	    "disabled": Disable TLSv1.3
	
	    "enabled": Enable TLSv1.3
	
	    "use_default": Use the global setting for TLSv1.3

ssl_trust_magic	If the traffic manager is receiving traffic sent from another traffic manager, then enabling this option will allow it to decode extra information on the true origin of the SSL connection. This information is supplied by the first traffic manager. • Type: Boolean • Required: false • Default value: false
syslog_enabled	Whether or not to log connections to the virtual server to a remote syslog host. • Type: Boolean • Required: false • Default value: false

Property	Description
syslog_format	The log format for the remote syslog. This specifies the line of text that will be sent to the remote syslog when a connection to the traffic manager is completed. Many parameters from the connection can be recorded using macros. • Type: String • Required: false • Default value: "%h %l %u %t "%r" %s %b "%{Referer}i" "%{User-agent}i""
syslog_ip_end_point	The remote host and port (default is 514) to send request log lines to. • Type: String • Required: false • Default value: <none>
syslog_msg_len_limit	Maximum length in bytes of a message sent to the remote syslog. Messages longer than this will be truncated before they are sent. • Type: UInt • Required: false • Default value: "2048"
tcp_close_with_rst	Whether or not connections from clients should be closed with a RST packet, rather than a FIN packet. This avoids the TIME_WAIT state, which on rare occasions allows wandering duplicate packets to be safely ignored. • Type: Boolean • Required: false • Default value: false

Property	Description
tcp_nagle	Whether or not Nagle's algorithm should be used for TCP connections. • Type: Boolean • Required: false • Default value: false
tcp_proxy_close	If set to Yes the traffic manager will send the client FIN to the back-end server and wait for a server response instead of closing the connection immediately. This is only necessary for protocols that require half-close support to function correctly, such as "rsh". If the traffic manager is responding to the request itself, setting this key to Yes will cause the traffic manager to continue writing the response even after it has received a FIN from the client. • Type: Boolean • Required: false • Default value: false
transaction_export_brief	Whether to export a restricted set of metadata about transactions processed by this virtual server. If enabled, more verbose information such as client and server headers and request tracing events will be omitted from the exported data. • Type: Boolean • Required: false • Default value: false
transaction_export_enabled	Export metadata about transactions handled by this service to the globally configured endpoint. Data will be exported only if the global transaction_export!enabled setting is enabled. • Type: Boolean • Required: false

Property	Description
	• Default value: true
transaction_export_hi_res	Whether the transaction processing timeline included in the metadata export is recorded with a high, microsecond, resolution. If set to No, timestamps will be recorded with a resolution of milliseconds. • Type: Boolean • Required: false • Default value: false
transaction_export_http_header_blacklist	The set of HTTP header names for which corresponding values should be redacted from the metadata exported by this virtual server. • Type: Set(String) • Required: false • Default value: "Authorization"
transparent	Whether or not bound sockets should be configured for transparent proxying. • Type: Boolean • Required: false • Default value: false
udp_end_point_persistence	Whether UDP datagrams received from the same IP address and port are sent to the same pool node if they match an existing UDP session. Sessions are defined by the protocol being handled, for example SIP datagrams are grouped based on the value of the Call-ID header. • Type: Boolean • Required: false

Property	Description
	• Default value: true
udp_port_smp	Whether or not UDP datagrams should be distributed across all traffic manager processes, if this behaviour is not normally selected automatically due to other settings. • Type: Boolean • Required: false • Default value: false
udp_rbuff_size	If this setting is non-zero, the virtual server will set the socket receive buffer size to this number of bytes. If set, this will override the so_rbuff_size setting. An OS-specified limit on socket buffer sizes such as given by sysctl net.core.rmem_max can be exceeded using this setting. • Type: UInt • Required: false • Default value: <none>
udp_response_datagrams_ expected	The virtual server should discard any UDP connection and reclaim resources when the node has responded with this number of datagrams. For simple request/response protocols this can be often set to 1. If set to -1, the connection will not be discarded until the timeout is reached. • Type: Int • Required: false • Default value: "1"
udp_smp_mode	Whether the traffic manager should try to use SO_REUSEPORT for distributing incoming UDP datagrams across multiple processes (if kernel support is detected) or whether the legacy (pre-20.2) multi-processing mode should be used. • Type: Enum(String)

Property	Description

	• Required: false
	
	• Default value: "auto"
	
	• Permitted values:
	
	    "auto": auto
	
	    "legacy": legacy

udp_timeout	The virtual server should discard any UDP connection and reclaim resources when no further UDP traffic has been seen within this time. • Type: UInt • Required: false • Default value: "7"
udp_wbuff_size	If this setting is non-zero, the virtual server will set the socket send buffer size to this number of bytes. If set, this will override the so_wbuff_size setting. An OS-specified limit on socket buffer sizes such as given by sysctl net.core.wmem_max can be exceeded using this setting. • Type: UInt • Required: false • Default value: <none>
web_cache_control_out	The "Cache-Control" header to add to every cached HTTP response, no-cache or max-age=600 for example. • Type: String • Required: false • Default value: <none>
web_cache_enabled	If set to Yes the traffic manager will attempt to cache web server responses.

Property	Description
	• Type: Boolean • Required: false • Default value: false
web_cache_error_page_time	Time period to cache error pages for. • Type: UInt • Required: false • Default value: "30"
web_cache_max_time	Maximum time period to cache web pages for. • Type: UInt • Required: false • Default value: "600"
web_cache_refresh_time	If a cached page is about to expire within this time, the traffic manager will start to forward some new requests on to the web servers. A maximum of one request per second will be forwarded; the remainder will continue to be served from the cache. This prevents "bursts" of traffic to your web servers when an item expires from the cache. Setting this value to 0 will stop the traffic manager updating the cache before it expires. • Type: UInt • Required: false • Default value: "2"

vtm_aptimizer_profile

URI Endpoint: /api/tm/8.3/config/active/aptimizer/profiles

A Web Accelerator profile can be applied to an HTTP virtual server to enable automatic web content optimization.

```
# Example usage of vtm_aptimizer_profile resource...

resource "vtm_aptimizer_profile" "my_aptimizer_profile" {

    name = "MyAptimizerProfile"

}
```

Property	Description
name	Name of the object • Type: string • Required: true
background_after	If Web Accelerator can finish optimizing the resource within this time limit then serve the optimized content to the client, otherwise complete the optimization in the background and return the original content to the client. If set to 0, Web Accelerator will always wait for the optimization to complete before sending a response to the client. • Type: UInt • Required: false • Default value: <none>
background_on_additional_resources	If a web page contains resources that have not yet been optimized, fetch and optimize those resources in the background and send a partially optimized web page to clients until all resources on that page are ready. • Type: Boolean • Required: false • Default value: false
mode	Set the Web Accelerator mode to turn acceleration on or off. • Type: Enum(String) • Required: false • Default value: "active"

Property	Description

	• Permitted values:
	
	    "active": On - Web Accelerator acceleration is enabled
	
	    "idle": Off - Acceleration is disabled, but requests for Web Accelerator resources are served
	
	    "stealth": Stealth - Acceleration is controlled by a cookie

show_info_bar	Show the Web Accelerator information bar on optimized web pages. This requires HTML optimization to be enabled in the acceleration settings. • Type: Boolean • Required: false • Default value: false

Data Sources

Data sources are used to obtain information about an object or environment.

Resources

Each configuration resource is also available as an identical Terraform "data source" object. All properties in a data source remain the same as the configuration resource on which it is based, but read-only. Configuration data sources allow a Terraform template to query the contents of a Traffic Manager configuration object that was created externally from the template. For example:

```
# Create a data source object to represent an existing virtual server
data "vtm_virtual_server" "my_vs_data" {
   name = "MyVS"
}

# Set up a new virtual server referencing the data source
resource "vtm_virtual_server" "my_new_vs" {
```

```
    name = "MyNewVS
    pool = "discard"
    port = 1234
    ssl_decrypt = true
    ssl_server_cert_default = "${data.vtm_virtual_server.my_vs_data.ssl_server_cert_
default}"
}
```

For details of the available properties for a particular resource, refer to the corresponding configuration resource reference earlier in this chapter.

Resource Lists

Resource list data sources represent a list of stored configuration objects of the named type. For example, to obtain the current list of virtual servers, use the "vtm_virtual_server_list" data source. By default, a resource list data source returns all objects of the given type. To filter the results, use the "starts_with", "ends_with", "contains", or "regex_match" filters in the data source definition. To obtain the resulting filtered object list, use the "object_list" property.

```
# Create a list of all pools
data "vtm_pool_list" "my_pool_list" {}

# Create a list of virtual servers with names starting "Exchange_"
data "vtm_virtual_server_list" "my_vs_list" {
    starts_with = "Exchange_"
}

# Create a TrafficScript rule list based on a regex match
data "vtm_rule_list" "my_rule_list" {
    regex_match = "^Rule-.*?-[0-9]$"
}
# Use the rule list in a virtual server definition
resource "vtm_virtual_server" "my_vs" {
    ...
    request_rules = ["${data.vtm_rule_list.my_rule_list.object_list}"]
    ...
}
```

The available resource list data sources are:

- vtm_action_list

- vtm_action_program_list

- vtm_aptimizer_profile_list

- vtm_aptimizer_scope_list

- vtm_bandwidth_list

- vtm_bgpneighbor_list

- vtm_cloud_api_credential_list

- vtm_custom_list

- vtm_dns_server_zone_file_list

- vtm_dns_server_zone_list

- vtm_event_type_list

- vtm_extra_file_list

- vtm_glb_service_list

- vtm_kerberos_keytab_list

- vtm_kerberos_krb5conf_list

- vtm_kerberos_principal_list

- vtm_license_key_list

- vtm_location_list

- vtm_log_export_list

- vtm_monitor_list

- vtm_monitor_script_list

- vtm_persistence_list

- vtm_pool_list

- vtm_protection_list

- vtm_rate_list

- vtm_rule_authenticator_list

- vtm_rule_list

- vtm_saml_trustedidp_list

- vtm_service_level_monitor_list

- vtm_servicediscovery_list

- vtm_ssl_admin_ca_list

- vtm_ssl_ca_list

- vtm_ssl_client_key_list

- vtm_ssl_server_key_list

- vtm_ssl_ticket_key_list

- vtm_traffic_ip_group_list

- vtm_traffic_manager_list

- vtm_user_authenticator_list

- vtm_user_group_list

- vtm_virtual_server_list

Statistics

Statistics data sources allow Terraform to access a Traffic Manager's SNMP counter values. Some statistics data sources access single objects and take no parameters (for example, vtm_globals_stats), while others represent a collection of objects and require a "name" parameter (for example, vtm_virtual_server_stats):

```
# Single object
data "vtm_globals_stats" "global_stats" {}



# Collection of objects
data "vtm_virtual_server_stats" "vs_stats" {
```

```
        name = "MyVirtualServer"
    }
```

vtm_action_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/actions/*

Actions statistics values.

```
# Example usage of vtm_action_stats data source...
data "vtm_action_stats" "my_action_stats" {
    name = "MyAction"
}


# Access with...
"${data.vtm_action_stats.my_action_stats.<FIELD_NAME>}"
```

Field Name	Description
processed	Number of times this action has been processed. • Type: UInt • SNMP name: "actionsProcessed"

vtm_cache_asp_session_cache_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/cache/asp_session_cache

Asp session cache statistics values.

```
# Example usage of vtm_cache_asp_session_cache_stats data source...
data "vtm_cache_asp_session_cache_stats" "my_cache_asp_session_cache_stats" {}


# Access with...
"${data.vtm_cache_asp_session_cache_stats.my_cache_asp_session_cache_stats.<FIELD_
NAME>}"
```

Field Name	Description
entries	The total number of ASP sessions stored in the cache.

Field Name	Description
	• Type: UInt • SNMP name: "aspSessionCacheEntries"
entries_max	The maximum number of ASP sessions in the cache. • Type: UInt • SNMP name: "aspSessionCacheEntriesMax"
hit_rate	The percentage of ASP session lookups that succeeded. • Type: UInt • SNMP name: "aspSessionCacheHitRate"
hits	Number of times a ASP session entry has been successfully found in the cache. • Type: UInt • SNMP name: "aspSessionCacheHits"
lookups	Number of times a ASP session entry has been looked up in the cache. • Type: UInt • SNMP name: "aspSessionCacheLookups"
misses	Number of times a ASP session entry has not been available in the cache. • Type: UInt • SNMP name: "aspSessionCacheMisses"
oldest	The age of the oldest ASP session in the cache (in seconds). • Type: UInt • SNMP name: "aspSessionCacheOldest"

vtm_bandwidth_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/bandwidth/*

Bandwidth statistics values.

```
# Example usage of vtm_bandwidth_stats data source...
data "vtm_bandwidth_stats" "my_bandwidth_stats" {
    name = "MyBandwidth"
}

# Access with...
"${data.vtm_bandwidth_stats.my_bandwidth_stats.<FIELD_NAME>}"
```

Field Name	Description
bytes_out	Bytes output by connections assigned to this bandwidth class. • Type: UInt64 • SNMP name: "bandwidthClassBytesOut"
guarantee	Guaranteed bandwidth class limit (kbits/s). Currently unused. • Type: UInt • SNMP name: "bandwidthClassGuarantee"
maximum	Maximum bandwidth class limit (kbits/s). • Type: UInt • SNMP name: "bandwidthClassMaximum"

vtm_cloud_api_credential_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/cloud_api_credentials/*

Cloud api credentials statistics values.

```
# Example usage of vtm_cloud_api_credential_stats data source...
data "vtm_cloud_api_credential_stats" "my_cloud_api_credential_stats" {
    name = "MyCloudApiCredential"
}


# Access with...
"${data.vtm_cloud_api_credential_stats.my_cloud_api_credential_stats.<FIELD_NAME>}"
```

Field Name	Description
node_creations	The number of instance creation API requests made with this set of cloud credentials. • Type: UInt • SNMP name: "cloudcredentialsNodeCreations"
node_deletions	The number of instance destruction API requests made with this set of cloud credentials. • Type: UInt • SNMP name: "cloudcredentialsNodeDeletions"
status_requests	The number of status API requests made with this set of cloud credentials. • Type: UInt • SNMP name: "cloudcredentialsStatusRequests"

vtm_connection_rate_limit_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/connection_rate_limit/*

Connection rate limit statistics values.

```
# Example usage of vtm_connection_rate_limit_stats data source...
data "vtm_connection_rate_limit_stats" "my_connection_rate_limit_stats" {
    name = "MyConnectionRateLimit"
}


# Access with...
"${data.vtm_connection_rate_limit_stats.my_connection_rate_limit_stats.<FIELD_NAME>}"
```

Field Name	Description
conns_entered	Connections that have entered the rate class and have been queued. • Type: UInt • SNMP name: "rateClassConnsEntered"
conns_left	Connections that have left the rate class. • Type: UInt • SNMP name: "rateClassConnsLeft"
current_rate	The average rate that requests are passing through this rate class. • Type: UInt • SNMP name: "rateClassCurrentRate"
dropped	Requests dropped from this rate class without being processed (e.g. timeouts). • Type: UInt • SNMP name: "rateClassDropped"
max_rate_per_min	The maximum rate that requests may pass through this rate class (requests/min). • Type: UInt • SNMP name: "rateClassMaxRatePerMin"
max_rate_per_sec	The maximum rate that requests may pass through this rate class (requests/sec). • Type: UInt • SNMP name: "rateClassMaxRatePerSec"
queue_length	The current number of requests queued by this rate class. • Type: UInt • SNMP name: "rateClassQueueLength"

vtm_event_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/events/*

Events statistics values.

```
# Example usage of vtm_event_stats data source...
data "vtm_event_stats" "my_event_stats" {
    name = "MyEvent"
}


# Access with...
"${data.vtm_event_stats.my_event_stats.<FIELD_NAME>}"
```

Field Name	Description

matched	Number of times this event configuration has matched.
	
	    • Type: UInt
	
	    • SNMP name: "eventsMatched"

vtm_glb_service_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/glb_services/*

Glb services statistics values.

```
# Example usage of vtm_glb_service_stats data source...
data "vtm_glb_service_stats" "my_glb_service_stats" {
    name = "MyGlbService"
}


# Access with...
"${data.vtm_glb_service_stats.my_glb_service_stats.<FIELD_NAME>}"
```

Field Name	Description
discarded	Number of A records this GLB Service has discarded. • Type: UInt

Field Name	Description
	• SNMP name: "glbServiceDiscarded"
responses	Number of A records this GLB Service has altered. • Type: UInt • SNMP name: "glbServiceResponses"
unmodified	Number of A records this GLB Service has passed through unmodified. • Type: UInt • SNMP name: "glbServiceUnmodified"

vtm_globals_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/globals

Globals statistics values.

```
# Example usage of vtm_globals_stats data source...
data "vtm_globals_stats" "my_globals_stats" {}

# Access with...
"${data.vtm_globals_stats.my_globals_stats.<FIELD_NAME>}"
```

Field Name	Description
analytics_transactions_dropped	Count of transaction metadata records that have been dropped • Type: UInt • SNMP name: "analyticsTransactionsDropped"
analytics_transactions_exported	Count of transaction metadata records that have been exported • Type: UInt • SNMP name: "analyticsTransactionsExported"
analytics_transactions_memory_usage	Number of bytes queued in the transaction export transmit buffers.

Field Name	Description
	• Type: UInt • SNMP name: "analyticsTransactionsMemoryUsage"
data_entries	Number of entries in the TrafficScript data.get()/set() storage. • Type: UInt • SNMP name: "dataEntries"
data_memory_usage	Number of bytes used in the TrafficScript data.get()/set() storage. • Type: UInt • SNMP name: "dataMemoryUsage"
events_seen	Events seen by the traffic Manager's event handling process. • Type: UInt • SNMP name: "eventsSeen"
hourly_peak_bytes_in_per_second	The peak bytes received from clients per second in the last hour. • Type: UInt • SNMP name: "hourlyPeakBytesInPerSecond"
hourly_peak_bytes_out_per_second	The peak bytes sent to clients per second in the last hour. • Type: UInt • SNMP name: "hourlyPeakBytesOutPerSecond"
hourly_peak_requests_per_second	The peak requests per second in the last hour. • Type: UInt • SNMP name: "hourlyPeakRequestsPerSecond"
hourly_peak_ssl_connections_per_second	The peak ssl connections per second in the last hour. • Type: UInt • SNMP name: "hourlyPeakSSLConnectionsPerSecond"

Field Name	Description
num_idle_connections	Total number of idle HTTP connections to all nodes (used for future HTTP requests). • Type: UInt • SNMP name: "numIdleConnections"
number_child_processes	The number of traffic manager child processes. • Type: UInt • SNMP name: "numberChildProcesses"
number_dnsa_cache_hits	Requests for DNS A records resolved from the traffic manager's local cache. • Type: UInt • SNMP name: "numberDNSACacheHits"
number_dnsa_requests	Requests for DNS A records (hostname->IP address) made by the traffic manager. • Type: UInt • SNMP name: "numberDNSARequests"
number_dnsptr_cache_hits	Requests for DNS PTR records resolved from the traffic manager's local cache. • Type: UInt • SNMP name: "numberDNSPTRCacheHits"
number_dnsptr_requests	Requests for DNS PTR records (IP address->hostname) made by the traffic manager. • Type: UInt • SNMP name: "numberDNSPTRRequests"
number_snmp_bad_ requests	Malformed SNMP requests received. • Type: UInt

Field Name	Description
	• SNMP name: "numberSNMPBadRequests"
number_snmp_get_bulk_requests	SNMP GetBulkRequests received. • Type: UInt • SNMP name: "numberSNMPGetBulkRequests"
number_snmp_get_next_requests	SNMP GetNextRequests received. • Type: UInt • SNMP name: "numberSNMPGetNextRequests"
number_snmp_get_requests	SNMP GetRequests received. • Type: UInt • SNMP name: "numberSNMPGetRequests"
number_snmp_unauthorised_requests	SNMP requests dropped due to access restrictions. • Type: UInt • SNMP name: "numberSNMPUnauthorisedRequests"
ssl_cipher_3des_decrypts	Bytes decrypted with 3DES. • Type: UInt • SNMP name: "sslCipher3DESDecrypts"
ssl_cipher_3des_encrypts	Bytes encrypted with 3DES. • Type: UInt • SNMP name: "sslCipher3DESEncrypts"
ssl_cipher_aes_decrypts	Bytes decrypted with AES. • Type: UInt • SNMP name: "sslCipherAESDecrypts"
ssl_cipher_aes_encrypts	Bytes encrypted with AES.

Field Name	Description
	• Type: UInt • SNMP name: "sslCipherAESEncrypts"
ssl_cipher_aes_gcm_decrypts	Bytes decrypted with AES-GCM. • Type: UInt • SNMP name: "sslCipherAESGCMDecrypts"
ssl_cipher_aes_gcm_encrypts	Bytes encrypted with AES-GCM. • Type: UInt • SNMP name: "sslCipherAESGCMEncrypts"
ssl_cipher_decrypts	Bytes decrypted with a symmetric cipher. • Type: UInt • SNMP name: "sslCipherDecrypts"
ssl_cipher_des_decrypts	Bytes decrypted with DES. • Type: UInt • SNMP name: "sslCipherDESDecrypts"
ssl_cipher_des_encrypts	Bytes encrypted with DES. • Type: UInt • SNMP name: "sslCipherDESEncrypts"
ssl_cipher_dh_agreements	Number of Diffie Hellman key agreements. • Type: UInt • SNMP name: "sslCipherDHAgreements"
ssl_cipher_dh_generates	Number of Diffie Hellman keys generated. • Type: UInt • SNMP name: "sslCipherDHGenerates"

Field Name	Description
ssl_cipher_dsa_signs	Number of DSA signing operations. • Type: UInt • SNMP name: "sslCipherDSASigns"
ssl_cipher_dsa_verifies	Number of DSA verifications. • Type: UInt • SNMP name: "sslCipherDSAVerifies"
ssl_cipher_ecdh_agreements	Number of Elliptic Curve Diffie Hellman key agreements. • Type: UInt • SNMP name: "sslCipherECDHAgreements"
ssl_cipher_ecdh_generates	Number of Elliptic Curve Diffie Hellman keys generated. • Type: UInt • SNMP name: "sslCipherECDHGenerates"
ssl_cipher_ecdsa_signs	Number of ECDSA signing operations. • Type: UInt • SNMP name: "sslCipherECDSASigns"
ssl_cipher_ecdsa_verifies	Number of ECDSA verifications. • Type: UInt • SNMP name: "sslCipherECDSAVerifies"
ssl_cipher_encrypts	Bytes encrypted with a symmetric cipher. • Type: UInt • SNMP name: "sslCipherEncrypts"
ssl_cipher_rc4_decrypts	Bytes decrypted with RC4. • Type: UInt

Field Name	Description
	• SNMP name: "sslCipherRC4Decrypts"
ssl_cipher_rc4_encrypts	Bytes encrypted with RC4. • Type: UInt • SNMP name: "sslCipherRC4Encrypts"
ssl_cipher_rsa_decrypts	Number of RSA decrypts. • Type: UInt • SNMP name: "sslCipherRSADecrypts"
ssl_cipher_rsa_decrypts_external	Number of external RSA decrypts. • Type: UInt • SNMP name: "sslCipherRSADecryptsExternal"
ssl_cipher_rsa_encrypts	Number of RSA encrypts. • Type: UInt • SNMP name: "sslCipherRSAEncrypts"
ssl_cipher_rsa_encrypts_external	Number of external RSA encrypts. • Type: UInt • SNMP name: "sslCipherRSAEncryptsExternal"
ssl_client_cert_expired	Number of times a client certificate has expired. • Type: UInt • SNMP name: "sslClientCertExpired"
ssl_client_cert_invalid	Number of times a client certificate was invalid. • Type: UInt • SNMP name: "sslClientCertInvalid"
ssl_client_cert_not_sent	Number of times a client certificate was required but not supplied.

Field Name	Description
	• Type: UInt • SNMP name: "sslClientCertNotSent"
ssl_client_cert_revoked	Number of times a client certificate was revoked. • Type: UInt • SNMP name: "sslClientCertRevoked"
ssl_connections	Number of SSL connections negotiated. • Type: UInt • SNMP name: "sslConnections"
ssl_handshake_sslv3	Number of SSLv3 handshakes. • Type: UInt • SNMP name: "sslHandshakeSSLv3"
ssl_handshake_tlsv1	Number of TLSv1.0 handshakes. • Type: UInt • SNMP name: "sslHandshakeTLSv1"
ssl_handshake_tlsv11	Number of TLSv1.1 handshakes. • Type: UInt • SNMP name: "sslHandshakeTLSv11"
ssl_handshake_tlsv12	Number of TLSv1.2 handshakes. • Type: UInt • SNMP name: "sslHandshakeTLSv12"
ssl_handshake_tlsv13	Number of TLSv1.3 handshakes. • Type: UInt • SNMP name: "sslHandshakeTLSv13"

Field Name	Description
ssl_session_id_mem_cache_hit	Number of times the SSL session id was found in the cache and reused. • Type: UInt • SNMP name: "sslSessionIDMemCacheHit"
ssl_session_id_mem_cache_miss	Number of times the SSL session id was not found in the cache. • Type: UInt • SNMP name: "sslSessionIDMemCacheMiss"
sys_cpu_busy_percent	Percentage of time that the CPUs are busy. • Type: UInt • SNMP name: "sysCPUBusyPercent"
sys_cpu_idle_percent	Percentage of time that the CPUs are idle. • Type: UInt • SNMP name: "sysCPUIdlePercent"
sys_cpu_system_busy_percent	Percentage of time that the CPUs are busy running system code. • Type: UInt • SNMP name: "sysCPUSystemBusyPercent"
sys_cpu_user_busy_percent	Percentage of time that the CPUs are busy running user-space code. • Type: UInt • SNMP name: "sysCPUUserBusyPercent"
sys_fds_free	Number of free file descriptors. • Type: UInt • SNMP name: "sysFDsFree"
sys_mem_buffered	Buffer memory (MBytes).

Field Name	Description
	• Type: UInt • SNMP name: "sysMemBuffered"
sys_mem_free	Free memory (MBytes). • Type: UInt • SNMP name: "sysMemFree"
sys_mem_in_use	Memory used (MBytes). • Type: UInt • SNMP name: "sysMemInUse"
sys_mem_swap_total	Total swap space (MBytes). • Type: UInt • SNMP name: "sysMemSwapTotal"
sys_mem_swapped	Amount of swap space in use (MBytes). • Type: UInt • SNMP name: "sysMemSwapped"
sys_mem_total	Total memory (MBytes). • Type: UInt • SNMP name: "sysMemTotal"
time_last_config_update	The time (in hundredths of a second) since the configuration of traffic manager was updated (this value will wrap if no configuration changes are made for 497 days). • Type: UInt • SNMP name: "timeLastConfigUpdate"
total_backend_server_errors	Total errors returned from the backend servers. • Type: UInt

Field Name	Description
	• SNMP name: "totalBackendServerErrors"
total_bad_dns_packets	Total number of malformed DNS response packets encountered from the backend servers. • Type: UInt • SNMP name: "totalBadDNSPackets"
total_bytes_in	Bytes received by the traffic manager from clients. • Type: UInt64 • SNMP name: "totalBytesIn"
total_bytes_out	Bytes sent by the traffic manager to clients. • Type: UInt64 • SNMP name: "totalBytesOut"
total_conn	Total number of TCP connections received. • Type: UInt • SNMP name: "totalConn"
total_current_conn	Number of TCP connections currently established. • Type: UInt • SNMP name: "totalCurrentConn"
total_dns_responses	Total number of DNS response packets handled. • Type: UInt • SNMP name: "totalDNSResponses"
total_requests	Total number of TCP requests received. • Type: UInt • SNMP name: "totalRequests"

Field Name	Description
total_transactions	Total number of TCP requests being processed, after applying TPS limits. • Type: UInt • SNMP name: "totalTransactions"
total_udp_bytes_in_dropped	UDP bytes received by the traffic manager from clients but failed to be sent to backend servers. • Type: UInt64 • SNMP name: "totalUDPBytesInDropped"
total_udp_bytes_out_dropped	UDP bytes received by the traffic manager from backend servers but failed to be sent to clients. • Type: UInt64 • SNMP name: "totalUDPBytesOutDropped"
up_time	The time (in hundredths of a second) that vTM software has been operational for (this value will wrap if it has been running for more than 497 days). • Type: UInt • SNMP name: "upTime"

vtm_traffic_ips_ip_gateway_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/traffic_ips/ip_gateway

Ip gateway statistics values.

```
# Example usage of vtm_traffic_ips_ip_gateway_stats data source...
data "vtm_traffic_ips_ip_gateway_stats" "my_traffic_ips_ip_gateway_stats" {}

# Access with...
"${data.vtm_traffic_ips_ip_gateway_stats.my_traffic_ips_ip_gateway_stats.<FIELD_
NAME>}"
```

Field Name	Description
arp_message	Number of ARP messages sent for raised Traffic IP Addresses. • Type: UInt • SNMP name: "trafficIPARPMessage"
gateway_ping_requests	Number of ping requests sent to the gateway machine. • Type: UInt • SNMP name: "trafficIPGatewayPingRequests"
gateway_ping_responses	Number of ping responses received from the gateway machine. • Type: UInt • SNMP name: "trafficIPGatewayPingResponses"
node_ping_requests	Number of ping requests sent to the backend nodes. • Type: UInt • SNMP name: "trafficIPNodePingRequests"
node_ping_responses	Number of ping responses received from the backend nodes. • Type: UInt • SNMP name: "trafficIPNodePingResponses"
number_inet46	The number of traffic IP addresses on this system (includes IPv4 and IPv6 addresses). • Type: UInt • SNMP name: "trafficIPNumberInet46"
number_raised_inet46	The number of traffic IP addresses currently raised on this system (includes IPv4 and IPv6 addresses). • Type: UInt • SNMP name: "trafficIPNumberRaisedInet46"
ping_response_errors	Number of ping response errors.

Field Name	Description
	• Type: UInt • SNMP name: "trafficIPPingResponseErrors"

vtm_cache_ip_session_cache_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/cache/ip_session_cache

Ip session cache statistics values.

```
# Example usage of vtm_cache_ip_session_cache_stats data source...
data "vtm_cache_ip_session_cache_stats" "my_cache_ip_session_cache_stats" {}

# Access with...
"${data.vtm_cache_ip_session_cache_stats.my_cache_ip_session_cache_stats.<FIELD_
NAME>}"
```

Field Name	Description
entries	The total number of IP sessions stored in the cache. • Type: UInt • SNMP name: "ipSessionCacheEntries"
entries_max	The maximum number of IP sessions in the cache. • Type: UInt • SNMP name: "ipSessionCacheEntriesMax"
expiries	Number of times an expired IP session entry has been dropped on lookup. • Type: UInt • SNMP name: "ipSessionCacheExpiries"
hit_rate	The percentage of IP session lookups that succeeded. • Type: UInt

Field Name	Description
	• SNMP name: "ipSessionCacheHitRate"
hits	Number of times a IP session entry has been successfully found in the cache. • Type: UInt • SNMP name: "ipSessionCacheHits"
lookups	Number of times a IP session entry has been looked up in the cache. • Type: UInt • SNMP name: "ipSessionCacheLookups"
misses	Number of times a IP session entry has not been available in the cache. • Type: UInt • SNMP name: "ipSessionCacheMisses"
oldest	The age of the oldest IP session in the cache (in seconds). • Type: UInt • SNMP name: "ipSessionCacheOldest"

vtm_cache_j2ee_session_cache_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/cache/j2ee_session_cache

J2ee session cache statistics values.

```
# Example usage of vtm_cache_j2ee_session_cache_stats data source...
data "vtm_cache_j2ee_session_cache_stats" "my_cache_j2ee_session_cache_stats" {}

# Access with...
"${data.vtm_cache_j2ee_session_cache_stats.my_cache_j2ee_session_cache_stats.<FIELD_
NAME>}"
```

Field Name	Description
entries	The total number of J2EE sessions stored in the cache. • Type: UInt • SNMP name: "j2eeSessionCacheEntries"
entries_max	The maximum number of J2EE sessions in the cache. • Type: UInt • SNMP name: "j2eeSessionCacheEntriesMax"
expiries	Number of times an expired J2EE session entry has been dropped on lookup. • Type: UInt • SNMP name: "j2eeSessionCacheExpiries"
hit_rate	The percentage of J2EE session lookups that succeeded. • Type: UInt • SNMP name: "j2eeSessionCacheHitRate"
hits	Number of times a J2EE session entry has been successfully found in the cache. • Type: UInt • SNMP name: "j2eeSessionCacheHits"
lookups	Number of times a J2EE session entry has been looked up in the cache. • Type: UInt • SNMP name: "j2eeSessionCacheLookups"
misses	Number of times a J2EE session entry has not been available in the cache. • Type: UInt

Field Name	Description
	• SNMP name: "j2eeSessionCacheMisses"
oldest	The age of the oldest J2EE session in the cache (in seconds). • Type: UInt • SNMP name: "j2eeSessionCacheOldest"

vtm_listen_ip_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/listen_ips/*

Listen ips statistics values.

```
# Example usage of vtm_listen_ip_stats data source...
data "vtm_listen_ip_stats" "my_listen_ip_stats" {
    name = "MyListenIp"
}


# Access with...
"${data.vtm_listen_ip_stats.my_listen_ip_stats.<FIELD_NAME>}"
```

Field Name	Description
bytes_in	Bytes sent to this listening IP. • Type: UInt64 • SNMP name: "listenIPBytesIn"
bytes_out	Bytes sent from this listening IP. • Type: UInt64 • SNMP name: "listenIPBytesOut"
current_conn	TCP connections currently established to this listening IP. • Type: UInt • SNMP name: "listenIPCurrentConn"

Field Name	Description
max_conn	Maximum number of simultaneous TCP connections this listening IP has processed at any one time. • Type: UInt • SNMP name: "listenIPMaxConn"
total_requests	Requests sent to this listening IP. • Type: UInt64 • SNMP name: "listenIPTotalRequests"

vtm_location_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/locations/*

Locations statistics values.

```
# Example usage of vtm_location_stats data source...
data "vtm_location_stats" "my_location_stats" {
    name = "MyLocation"
}

# Access with...
"${data.vtm_location_stats.my_location_stats.<FIELD_NAME>}"
```

Field Name	Description
load	The mean load metric for this location. • Type: UInt • SNMP name: "locationLoad"
responses	Number of A records that have been altered to point to this location. • Type: UInt • SNMP name: "locationResponses"

vtm_network_interface_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/network_interface/*

Network interface statistics values.

```
# Example usage of vtm_network_interface_stats data source...
data "vtm_network_interface_stats" "my_network_interface_stats" {
    name = "MyNetworkInterface"
}


# Access with...
"${data.vtm_network_interface_stats.my_network_interface_stats.<FIELD_NAME>}"
```

Field Name	Description
collisions	The number of collisions reported by this interface. • Type: UInt • SNMP name: "interfaceCollisions"
rx_bytes	Bytes received by this interface. • Type: UInt64 • SNMP name: "interfaceRxBytes"
rx_errors	The number of receive errors reported by this interface. • Type: UInt • SNMP name: "interfaceRxErrors"
rx_packets	The number of packets received by this interface. • Type: UInt • SNMP name: "interfaceRxPackets"
tx_bytes	Bytes transmitted by this interface. • Type: UInt64 • SNMP name: "interfaceTxBytes"

Field Name	Description
tx_errors	The number of transmit errors reported by this interface. • Type: UInt • SNMP name: "interfaceTxErrors"
tx_packets	The number of packets transmitted by this interface. • Type: UInt • SNMP name: "interfaceTxPackets"

vtm_nodes_node_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/nodes/node/*

Node statistics values.

```
# Example usage of vtm_nodes_node_stats data source...
data "vtm_nodes_node_stats" "my_nodes_node_stats" {
    name = "MyNodesNode"
}

# Access with...
"${data.vtm_nodes_node_stats.my_nodes_node_stats.<FIELD_NAME>}"
```

Field Name	Description
bytes_from_node_hi	Bytes received from this node (high 32bits). • Type: UInt • SNMP name: "nodeBytesFromNodeHi"
bytes_from_node_lo	Bytes received from this node (low 32bits). • Type: UInt • SNMP name: "nodeBytesFromNodeLo"
bytes_to_node_hi	Bytes sent to this node (high 32bits).

Field Name	Description
	• Type: UInt • SNMP name: "nodeBytesToNodeHi"
bytes_to_node_lo	Bytes sent to this node (low 32bits). • Type: UInt • SNMP name: "nodeBytesToNodeLo"
current_conn	Requests currently established to this node. (does not include idle keepalives). • Type: UInt • SNMP name: "nodeCurrentConn"
current_requests	Connections currently established to this node. • Type: UInt • SNMP name: "nodeCurrentRequests"
errors	Number of timeouts, connection problems and other errors for this node. • Type: UInt • SNMP name: "nodeErrors"
failures	Failures of this node. • Type: UInt • SNMP name: "nodeFailures"
new_conn	Requests that created a new connection to this node. • Type: UInt • SNMP name: "nodeNewConn"
pooled_conn	Requests that reused an existing pooled/keepalive connection rather than creating a new TCP connection.

Field Name	Description
	• Type: UInt • SNMP name: "nodePooledConn"
port	The port this node listens on. • Type: UInt • SNMP name: "nodePort"
response_max	Maximum response time (ms) in the last second for this node. • Type: UInt • SNMP name: "nodeResponseMax"
response_mean	Mean response time (ms) in the last second for this node. • Type: UInt • SNMP name: "nodeResponseMean"
response_min	Minimum response time (ms) in the last second for this node. • Type: UInt • SNMP name: "nodeResponseMin"

state	The state of this node.
	
	• Type: Enum(String)
	
	• SNMP name: "nodeState"
	
	• Permitted values:
	
	    "alive": alive(1)
	
	    "dead": dead(2)
	
	    "unknown": unknown(3)

total_conn	Requests sent to this node. • Type: UInt

Field Name	Description
	• SNMP name: "nodeTotalConn"

vtm_nodes_node_inet46_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/nodes/node_inet46/*

Node inet46 statistics values.

```
# Example usage of vtm_nodes_node_inet46_stats data source...
data "vtm_nodes_node_inet46_stats" "my_nodes_node_inet46_stats" {
    name = "MyNodesNodeInet46"
}


# Access with...
"${data.vtm_nodes_node_inet46_stats.my_nodes_node_inet46_stats.<FIELD_NAME>}"
```

Field Name	Description
bytes_from_node	Bytes received from this node. • Type: UInt64 • SNMP name: "nodeInet46BytesFromNode"
bytes_to_node	Bytes sent to this node. • Type: UInt64 • SNMP name: "nodeInet46BytesToNode"
current_conn	Current connections established to this node, includes idle connections and connections being established. • Type: UInt • SNMP name: "nodeInet46CurrentConn"
current_requests	Active connections established to this node, does not include idle connections. • Type: UInt

Field Name	Description
	• SNMP name: "nodeInet46CurrentRequests"
errors	Number of timeouts, connection problems and other errors for this node. • Type: UInt • SNMP name: "nodeInet46Errors"
failures	Failures of this node. • Type: UInt • SNMP name: "nodeInet46Failures"
idle_conns	Number of idle HTTP connections to this node. • Type: UInt • SNMP name: "nodeInet46IdleConns"
new_conn	Requests that created a new connection to this node. • Type: UInt • SNMP name: "nodeInet46NewConn"
pooled_conn	Requests that reused an existing pooled/keepalive connection rather than creating a new TCP connection. • Type: UInt • SNMP name: "nodeInet46PooledConn"
port	The port this node listens on. • Type: UInt • SNMP name: "nodeInet46Port"
response_max	Maximum response time (ms) in the last second for this node. • Type: UInt • SNMP name: "nodeInet46ResponseMax"

Field Name	Description
response_mean	Mean response time (ms) in the last second for this node. • Type: UInt • SNMP name: "nodeInet46ResponseMean"
response_min	Minimum response time (ms) in the last second for this node. • Type: UInt • SNMP name: "nodeInet46ResponseMin"

state	The state of this node.
	
	• Type: Enum(String)
	
	• SNMP name: "nodeInet46State"
	
	• Permitted values:
	
	    "alive": alive(1)
	
	    "dead": dead(2)
	
	    "unknown": unknown(3)

total_conn	Requests sent to this node. • Type: UInt • SNMP name: "nodeInet46TotalConn"

vtm_per_location_service_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/per_location_service/*

Per location service statistics values.

```
# Example usage of vtm_per_location_service_stats data source...
data "vtm_per_location_service_stats" "my_per_location_service_stats" {
    name = "MyPerLocationService"
}

# Access with...
"${data.vtm_per_location_service_stats.my_per_location_service_stats.<FIELD_NAME>}"
```

Field Name	Description

draining	The draining state of this location for this GLB Service.
	
	• Type: Enum(String)
	
	• SNMP name: "perLocationServiceDraining"
	
	• Permitted values:
	
	    "draining": draining(1)
	
	    "active": active(2)

frontend_state	The frontend state of this location for this GLB Service.
	
	• Type: Enum(String)
	
	• SNMP name: "perLocationServiceFrontendState"
	
	• Permitted values:
	
	    "alive": alive(1)
	
	    "dead": dead(2)

load	The load metric for this location for this GLB Service. • Type: UInt • SNMP name: "perLocationServiceLoad"

monitor_state	The monitor state of this location for this GLB Service.
	
	• Type: Enum(String)
	
	• SNMP name: "perLocationServiceMonitorState"
	
	• Permitted values:
	
	    "alive": alive(1)
	
	    "dead": dead(2)

responses	Number of A records that have been altered to point to this location for this GLB Service.

Field Name	Description
	• Type: UInt • SNMP name: "perLocationServiceResponses"

state	The state of this location for this GLB Service.
	
	• Type: Enum(String)
	
	• SNMP name: "perLocationServiceState"
	
	• Permitted values:
	
	    "alive": alive(1)
	
	    "dead": dead(2)

vtm_per_node_slm_per_node_service_level_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/per_node_slm/per_node_service_level/*

Per node service level statistics values.

```
# Example usage of vtm_per_node_slm_per_node_service_level_stats data source...
data "vtm_per_node_slm_per_node_service_level_stats" "my_per_node_slm_per_node_
service_level_stats" {
    name = "MyPerNodeSlmPerNodeServiceLevel"
}

# Access with...
"${data.vtm_per_node_slm_per_node_service_level_stats.my_per_node_slm_per_node_
service_level_stats.<FIELD_NAME>}"
```

Field Name	Description
node_port	The port number of this node. • Type: UInt • SNMP name: "perNodeServiceLevelNodePort"
response_max	Maximum response time (ms) in the last second for this SLM class to this node.

Field Name	Description
	• Type: UInt • SNMP name: "perNodeServiceLevelResponseMax"
response_mean	Mean response time (ms) in the last second for this SLM class to this node. • Type: UInt • SNMP name: "perNodeServiceLevelResponseMean"
response_min	Minimum response time (ms) in the last second for this SLM class to this node. • Type: UInt • SNMP name: "perNodeServiceLevelResponseMin"
total_conn	Requests handled by this SLM class to this node. • Type: UInt • SNMP name: "perNodeServiceLevelTotalConn"
total_non_conf	Non-conforming requests handled by this SLM class to this node. • Type: UInt • SNMP name: "perNodeServiceLevelTotalNonConf"

vtm_per_node_slm_per_node_service_level_inet46_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/per_node_slm/per_node_service_level_inet46/*

Per node service level inet46 statistics values.

```
# Example usage of vtm_per_node_slm_per_node_service_level_inet46_stats data source...
data "vtm_per_node_slm_per_node_service_level_inet46_stats" "my_per_node_slm_per_node_
service_level_inet46_stats" {
    name = "MyPerNodeSlmPerNodeServiceLevelInet46"
}

# Access with...
"${data.vtm_per_node_slm_per_node_service_level_inet46_stats.my_per_node_slm_per_node_
service_level_inet46_stats.<FIELD_NAME>}"
```

Field Name	Description
node_port	The port number of this node. • Type: UInt • SNMP name: "perNodeServiceLevelInet46NodePort"
response_max	Maximum response time (ms) in the last second for this SLM class to this node. • Type: UInt • SNMP name: "perNodeServiceLevelInet46ResponseMax"
response_mean	Mean response time (ms) in the last second for this SLM class to this node. • Type: UInt • SNMP name: "perNodeServiceLevelInet46ResponseMean"
response_min	Minimum response time (ms) in the last second for this SLM class to this node. • Type: UInt • SNMP name: "perNodeServiceLevelInet46ResponseMin"
total_conn	Requests handled by this SLM class to this node. • Type: UInt • SNMP name: "perNodeServiceLevelInet46TotalConn"

Field Name	Description
total_non_conf	Non-conforming requests handled by this SLM class to this node. • Type: UInt • SNMP name: "perNodeServiceLevelInet46TotalNonConf"

vtm_nodes_per_pool_node_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/nodes/per_pool_node/*

Per pool node statistics values.

```
# Example usage of vtm_nodes_per_pool_node_stats data source...
data "vtm_nodes_per_pool_node_stats" "my_nodes_per_pool_node_stats" {
    name = "MyNodesPerPoolNode"
}

# Access with...
"${data.vtm_nodes_per_pool_node_stats.my_nodes_per_pool_node_stats.<FIELD_NAME>}"
```

Field Name	Description
bytes_from_node	Bytes received from this node. • Type: UInt64 • SNMP name: "perPoolNodeBytesFromNode"
bytes_to_node	Bytes sent to this node. • Type: UInt64 • SNMP name: "perPoolNodeBytesToNode"
current_conn	Current connections established to a node, includes idle connections and connections being established. • Type: UInt • SNMP name: "perPoolNodeCurrentConn"

Field Name	Description
current_requests	Active connections established to this node, does not include idle connections. • Type: UInt • SNMP name: "perPoolNodeCurrentRequests"
errors	Number of timeouts, connection problems and other errors for this node. • Type: UInt • SNMP name: "perPoolNodeErrors"
failures	Failures of this node. • Type: UInt • SNMP name: "perPoolNodeFailures"
idle_conns	Number of idle HTTP connections to this node. • Type: UInt • SNMP name: "perPoolNodeIdleConns"
new_conn	Requests that created a new connection to this node. • Type: UInt • SNMP name: "perPoolNodeNewConn"
node_port	The port that this node listens on. • Type: UInt • SNMP name: "perPoolNodeNodePort"
pooled_conn	Requests that reused an existing pooled/keepalive connection rather than creating a new TCP connection. • Type: UInt • SNMP name: "perPoolNodePooledConn"

Field Name	Description
response_max	Maximum response time (ms) in the last second for this node. • Type: UInt • SNMP name: "perPoolNodeResponseMax"
response_mean	Mean response time (ms) in the last second for this node. • Type: UInt • SNMP name: "perPoolNodeResponseMean"
response_min	Minimum response time (ms) in the last second for this node. • Type: UInt • SNMP name: "perPoolNodeResponseMin"
state	The state of this node. • Type: Enum(String) • SNMP name: "perPoolNodeState" • Permitted values: "alive": alive(1) "dead": dead(2) "unknown": unknown(3) "draining": draining(4) "drainingtodelete": drainingtodelete(5)
total_conn	Requests sent to this node. • Type: UInt • SNMP name: "perPoolNodeTotalConn"

vtm_pool_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/pools/*

Pools statistics values.

```
# Example usage of vtm_pool_stats data source...

data "vtm_pool_stats" "my_pool_stats" {

    name = "MyPool"

}


# Access with...

"${data.vtm_pool_stats.my_pool_stats.<FIELD_NAME>}"
```

Field Name	Description

algorithm	The load-balancing algorithm the pool uses.
	
	• Type: Enum(String)
	
	• SNMP name: "poolAlgorithm"
	
	• Permitted values:
	
	    "roundrobin": roundrobin(1)
	
	    "weightedRoundRobin": weightedRoundRobin(2)
	
	    "perceptive": perceptive(3)
	
	    "leastConnections": leastConnections(4)
	
	    "fastestResponseTime": fastestResponseTime(5)
	
	    "random": random(6)
	
	    "weightedLeastConnections": weightedLeastConnections(7)

bytes_in	Bytes received by this pool from nodes. • Type: UInt64 • SNMP name: "poolBytesIn"
bytes_out	Bytes sent by this pool to nodes.

Field Name	Description
	• Type: UInt64 • SNMP name: "poolBytesOut"
conns_queued	Total connections currently queued to this pool. • Type: UInt • SNMP name: "poolConnsQueued"
disabled	The number of nodes in this pool that are disabled. • Type: UInt • SNMP name: "poolDisabled"
draining	The number of nodes in this pool which are draining. • Type: UInt • SNMP name: "poolDraining"
http1xx_responses	Number of HTTP 1xx responses returned by this pool. • Type: UInt64 • SNMP name: "poolHTTP1xxResponses"
http2xx_responses	Number of HTTP 2xx responses returned by this pool. • Type: UInt64 • SNMP name: "poolHTTP2xxResponses"
http3xx_responses	Number of HTTP 3xx responses returned by this pool. • Type: UInt64 • SNMP name: "poolHTTP3xxResponses"
http4xx_responses	Number of HTTP 4xx responses returned by this pool. • Type: UInt64 • SNMP name: "poolHTTP4xxResponses"

Field Name	Description
http503_retries	Number of times the pool received an HTTP 503 response from a node and retried it against a different node. • Type: UInt64 • SNMP name: "poolHTTP503Retries"
http5xx_responses	Number of HTTP 5xx responses returned by this pool. • Type: UInt64 • SNMP name: "poolHTTP5xxResponses"
max_queue_time	Maximum time, measured in milliseconds, for which a connection was queued over the last second. • Type: UInt • SNMP name: "poolMaxQueueTime"
mean_queue_time	Mean time, measured in milliseconds, for which a connection was queued over the last second. • Type: UInt • SNMP name: "poolMeanQueueTime"
min_queue_time	Minimum time, measured in milliseconds, for which a connection was queued over the last second. • Type: UInt • SNMP name: "poolMinQueueTime"
nodes	The number of nodes registered with this pool. • Type: UInt • SNMP name: "poolNodes"
persistence	The session persistence method this pool uses • Type: Enum(String)

Field Name	Description
	• SNMP name: "poolPersistence"
	• Permitted values:
	"none": none(1)
	"ip": ip(2)
	"rule": rule(3)
	"transparent": transparent(4)
	"applicationCookie": applicationCookie(5)
	"xZeusBackend": xZeusBackend(6)
	"ssl": ssl(7)
queue_timeouts	Total connections that timed-out while queued. • Type: UInt • SNMP name: "poolQueueTimeouts"
session_migrated	Sessions migrated to a new node because the desired node was unavailable. • Type: UInt • SNMP name: "poolSessionMigrated"
state	The state of this pool. • Type: Enum(String) • SNMP name: "poolState" • Permitted values: "active": active(1) "disabled": disabled(2) "draining": draining(3)

Field Name	Description
	"unused": unused(4) "unknown": unknown(5)
total_conn	Requests sent to this pool. • Type: UInt • SNMP name: "poolTotalConn"

vtm_rule_authenticator_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/rule_authenticators/*

Rule authenticators statistics values.

```
# Example usage of vtm_rule_authenticator_stats data source...
data "vtm_rule_authenticator_stats" "my_rule_authenticator_stats" {
    name = "MyRuleAuthenticator"
}

# Access with...
"${data.vtm_rule_authenticator_stats.my_rule_authenticator_stats.<FIELD_NAME>}"
```

Field Name	Description
errors	Number of connection errors that have occurred when trying to connect to an authentication server. • Type: UInt • SNMP name: "authenticatorErrors"
fails	Number of times this Authenticator has failed to authenticate. • Type: UInt • SNMP name: "authenticatorFails"
passes	Number of times this Authenticator has successfully authenticated. • Type: UInt

Field Name	Description
	• SNMP name: "authenticatorPasses"
requests	Number of times this Authenticator has been asked to authenticate. • Type: UInt • SNMP name: "authenticatorRequests"

vtm_rule_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/rules/*

Rules statistics values.

```
# Example usage of vtm_rule_stats data source...
data "vtm_rule_stats" "my_rule_stats" {
    name = "MyRule"
}

# Access with...
"${data.vtm_rule_stats.my_rule_stats.<FIELD_NAME>}"
```

Field Name	Description
aborts	Number of times this TrafficScript rule has aborted. • Type: UInt • SNMP name: "ruleAborts"
discards	Number of times this TrafficScript rule has discarded the connection. • Type: UInt • SNMP name: "ruleDiscards"
execution_time_warnings	Number of times this TrafficScript rule has exceeded the execution time warning threshold. • Type: UInt • SNMP name: "ruleExecutionTimeWarnings"

Field Name	Description
executions	Number of times this TrafficScript rule has been executed. • Type: UInt • SNMP name: "ruleExecutions"
pool_select	Number of times this TrafficScript rule has selected a pool to use. • Type: UInt • SNMP name: "rulePoolSelect"
responds	Number of times this TrafficScript rule has responded directly to the client. • Type: UInt • SNMP name: "ruleResponds"
retries	Number of times this TrafficScript rule has forced the request to be retried. • Type: UInt • SNMP name: "ruleRetries"

vtm_service_level_monitor_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/service_level_monitors/*

Service level monitors statistics values.

```
# Example usage of vtm_service_level_monitor_stats data source...
data "vtm_service_level_monitor_stats" "my_service_level_monitor_stats" {
    name = "MyServiceLevelMonitor"
}

# Access with...
"${data.vtm_service_level_monitor_stats.my_service_level_monitor_stats.<FIELD_NAME>}"
```

Field Name	Description
conforming	Percentage of requests associated with this SLM class that are conforming • Type: UInt • SNMP name: "serviceLevelConforming"
current_conns	The number of connections currently associated with this SLM class. • Type: UInt • SNMP name: "serviceLevelCurrentConns"

is_o_k	Indicates if this SLM class is currently conforming.
	
	• Type: Enum(String)
	
	• SNMP name: "serviceLevelIsOK"
	
	• Permitted values:
	
	    "notok": notok(1)
	
	    "ok": ok(2)

response_max	Maximum response time (ms) in the last second for this SLM class. • Type: UInt • SNMP name: "serviceLevelResponseMax"
response_mean	Mean response time (ms) in the last second for this SLM class. • Type: UInt • SNMP name: "serviceLevelResponseMean"
response_min	Minimum response time (ms) in the last second for this SLM class. • Type: UInt • SNMP name: "serviceLevelResponseMin"
total_conn	Requests handled by this SLM class.

Field Name	Description
	• Type: UInt • SNMP name: "serviceLevelTotalConn"
total_non_conf	Non-conforming requests handled by this SLM class. • Type: UInt • SNMP name: "serviceLevelTotalNonConf"

vtm_service_protection_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/service_protection/*

Service protection statistics values.

```
# Example usage of vtm_service_protection_stats data source...
data "vtm_service_protection_stats" "my_service_protection_stats" {
    name = "MyServiceProtection"
}

# Access with...
"${data.vtm_service_protection_stats.my_service_protection_stats.<FIELD_NAME>}"
```

Field Name	Description
last_refusal_time	The time (in hundredths of a second) since this service protection class last refused a connection (this value will wrap if no connections are refused in more than 497 days). • Type: UInt • SNMP name: "serviceProtLastRefusalTime"
refusal_binary	Connections refused by this service protection class because the request contained disallowed binary content. • Type: UInt • SNMP name: "serviceProtRefusalBinary"

Field Name	Description
refusal_conc10_ip	Connections refused by this service protection class because the top 10 source IP addresses issued too many concurrent connections. • Type: UInt • SNMP name: "serviceProtRefusalConc10IP"
refusal_conc1_ip	Connections refused by this service protection class because the source IP address issued too many concurrent connections. • Type: UInt • SNMP name: "serviceProtRefusalConc1IP"
refusal_conn_rate	Connections refused by this service protection class because the source IP address issued too many connections within 60 seconds. • Type: UInt • SNMP name: "serviceProtRefusalConnRate"
refusal_ip	Connections refused by this service protection class because the source IP address was banned. • Type: UInt • SNMP name: "serviceProtRefusalIP"
refusal_rfc2396	Connections refused by this service protection class because the HTTP request was not RFC 2396 compliant. • Type: UInt • SNMP name: "serviceProtRefusalRFC2396"
refusal_size	Connections refused by this service protection class because the request was larger than the defined limits allowed. • Type: UInt • SNMP name: "serviceProtRefusalSize"
total_refusal	Connections refused by this service protection class.

Field Name	Description
	• Type: UInt • SNMP name: "serviceProtTotalRefusal"

vtm_cache_ssl_cache_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/cache/ssl_cache

Ssl cache statistics values.

```
# Example usage of vtm_cache_ssl_cache_stats data source...
data "vtm_cache_ssl_cache_stats" "my_cache_ssl_cache_stats" {}

# Access with...
"${data.vtm_cache_ssl_cache_stats.my_cache_ssl_cache_stats.<FIELD_NAME>}"
```

Field Name	Description
entries	The total number of SSL sessions stored in the server cache. • Type: UInt • SNMP name: "sslCacheEntries"
entries_max	The maximum number of SSL entries in the server cache. • Type: UInt • SNMP name: "sslCacheEntriesMax"
hit_rate	The percentage of SSL server cache lookups that succeeded. • Type: UInt • SNMP name: "sslCacheHitRate"
hits	Number of times a SSL entry has been successfully found in the server cache. • Type: UInt • SNMP name: "sslCacheHits"

Field Name	Description
lookups	Number of times a SSL entry has been looked up in the server cache. • Type: UInt • SNMP name: "sslCacheLookups"
misses	Number of times a SSL entry has not been available in the server cache. • Type: UInt • SNMP name: "sslCacheMisses"
oldest	The age of the oldest SSL session in the server cache (in seconds). • Type: UInt • SNMP name: "sslCacheOldest"

vtm_ssl_ocsp_stapling_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/ssl_ocsp_stapling

Ssl ocsp stapling statistics values.

```
# Example usage of vtm_ssl_ocsp_stapling_stats data source...
data "vtm_ssl_ocsp_stapling_stats" "my_ssl_ocsp_stapling_stats" {}

# Access with...
"${data.vtm_ssl_ocsp_stapling_stats.my_ssl_ocsp_stapling_stats.<FIELD_NAME>}"
```

Field Name	Description
cache_count	The number of entries in the OCSP stapling cache. • Type: UInt • SNMP name: "sslOcspStaplingCacheCount"
counter	The number of outgoing OCSP requests for OCSP stapling. • Type: UInt

Field Name	Description
	• SNMP name: "sslOcspStaplingCount"
failure_count	The number of failed outgoing OCSP requests for OCSP stapling. • Type: UInt • SNMP name: "sslOcspStaplingFailureCount"
good_count	The number of 'good' OCSP responses for OCSP stapling. • Type: UInt • SNMP name: "sslOcspStaplingGoodCount"
revoked_count	The number of 'revoked' OCSP responses for OCSP stapling. • Type: UInt • SNMP name: "sslOcspStaplingRevokedCount"
success_count	The number of successful outgoing OCSP requests for OCSP stapling. • Type: UInt • SNMP name: "sslOcspStaplingSuccessCount"
unknown_count	The number of 'unknown' OCSP requests for OCSP stapling. • Type: UInt • SNMP name: "sslOcspStaplingUnknownCount"

vtm_cache_ssl_session_cache_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/cache/ssl_session_cache

Ssl session cache statistics values.

```
# Example usage of vtm_cache_ssl_session_cache_stats data source...

data "vtm_cache_ssl_session_cache_stats" "my_cache_ssl_session_cache_stats" {}


# Access with...

"${data.vtm_cache_ssl_session_cache_stats.my_cache_ssl_session_cache_stats.<FIELD_
NAME>}"
```

Field Name	Description
entries	The total number of SSL session persistence entries stored in the cache. • Type: UInt • SNMP name: "sslSessionCacheEntries"
entries_max	The maximum number of SSL session persistence entries in the cache. • Type: UInt • SNMP name: "sslSessionCacheEntriesMax"
hit_rate	The percentage of SSL session persistence lookups that succeeded. • Type: UInt • SNMP name: "sslSessionCacheHitRate"
hits	Number of times a SSL session persistence entry has been successfully found in the cache. • Type: UInt • SNMP name: "sslSessionCacheHits"
lookups	Number of times a SSL session persistence entry has been looked up in the cache. • Type: UInt • SNMP name: "sslSessionCacheLookups"
misses	Number of times a SSL session persistence entry has not been available in the cache.

Field Name	Description
	• Type: UInt • SNMP name: "sslSessionCacheMisses"
oldest	The age of the oldest SSL session in the cache (in seconds). • Type: UInt • SNMP name: "sslSessionCacheOldest"

vtm_traffic_ips_traffic_ip_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/traffic_ips/traffic_ip/*

Traffic ip statistics values.

```
# Example usage of vtm_traffic_ips_traffic_ip_stats data source...
data "vtm_traffic_ips_traffic_ip_stats" "my_traffic_ips_traffic_ip_stats" {
    name = "MyTrafficIpsTrafficIp"
}

# Access with...
"${data.vtm_traffic_ips_traffic_ip_stats.my_traffic_ips_traffic_ip_stats.<FIELD_
NAME>}"
```

Field Name	Description

state	Whether this traffic IP address is currently being hosted by this traffic manager.
	
	• Type: Enum(String)
	
	• SNMP name: "trafficIPState"
	
	• Permitted values:
	
	    "raised": raised(1)
	
	    "lowered": lowered(2)

Field Name	Description
time	The time (in hundredths of a second) since trafficIPState last changed (this value will wrap if the state hasn't changed for 497 days). • Type: UInt • SNMP name: "trafficIPTime"

vtm_traffic_ips_traffic_ip_inet46_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/traffic_ips/traffic_ip_inet46/*

Traffic ip inet46 statistics values.

```
# Example usage of vtm_traffic_ips_traffic_ip_inet46_stats data source...
data "vtm_traffic_ips_traffic_ip_inet46_stats" "my_traffic_ips_traffic_ip_inet46_
stats" {
    name = "MyTrafficIpsTrafficIpInet46"
}

# Access with...
"${data.vtm_traffic_ips_traffic_ip_inet46_stats.my_traffic_ips_traffic_ip_inet46_
stats.<FIELD_NAME>}"
```

Field Name	Description

state	Whether this traffic IP address is currently being hosted by this traffic manager.
	
	• Type: Enum(String)
	
	• SNMP name: "trafficIPInet46State"
	
	• Permitted values:
	
	    "raised": raised(1)
	
	    "lowered": lowered(2)

Field Name	Description
time	The time (in hundredths of a second) since trafficIPState last changed (this value will wrap if the state hasn't changed for 497 days). • Type: UInt • SNMP name: "trafficIPInet46Time"

vtm_cache_uni_session_cache_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/cache/uni_session_cache

Uni session cache statistics values.

```
# Example usage of vtm_cache_uni_session_cache_stats data source...
data "vtm_cache_uni_session_cache_stats" "my_cache_uni_session_cache_stats" {}

# Access with...
"${data.vtm_cache_uni_session_cache_stats.my_cache_uni_session_cache_stats.<FIELD_
NAME>}"
```

Field Name	Description
entries	The total number of universal sessions stored in the cache. • Type: UInt • SNMP name: "uniSessionCacheEntries"
entries_max	The maximum number of universal sessions in the cache. • Type: UInt • SNMP name: "uniSessionCacheEntriesMax"
expiries	Number of times an expired universal session entry has been dropped on lookup. • Type: UInt • SNMP name: "uniSessionCacheExpiries"

Field Name	Description
hit_rate	The percentage of universal session lookups that succeeded. • Type: UInt • SNMP name: "uniSessionCacheHitRate"
hits	Number of times a universal session entry has been successfully found in the cache. • Type: UInt • SNMP name: "uniSessionCacheHits"
lookups	Number of times a universal session entry has been looked up in the cache. • Type: UInt • SNMP name: "uniSessionCacheLookups"
misses	Number of times a universal session entry has not been available in the cache. • Type: UInt • SNMP name: "uniSessionCacheMisses"
oldest	The age of the oldest universal session in the cache (in seconds). • Type: UInt • SNMP name: "uniSessionCacheOldest"

vtm_extras_user_counters_32_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/extras/user_counters_32

User counters 32 statistics values.

```
# Example usage of vtm_extras_user_counters_32_stats data source...
data "vtm_extras_user_counters_32_stats" "my_extras_user_counters_32_stats" {}

# Access with...
"${data.vtm_extras_user_counters_32_stats.my_extras_user_counters_32_stats.<FIELD_
NAME>}"
```

Field Name	Description
counter	The value of the user counter. • Type: UInt • SNMP name: "userCounterValue"

vtm_extras_user_counters_64_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/extras/user_counters_64

User counters 64 statistics values.

```
# Example usage of vtm_extras_user_counters_64_stats data source...
data "vtm_extras_user_counters_64_stats" "my_extras_user_counters_64_stats" {}

# Access with...
"${data.vtm_extras_user_counters_64_stats.my_extras_user_counters_64_stats.<FIELD_
NAME>}"
```

Field Name	Description
counter	The value of the 64-bit user counter. • Type: UInt64 • SNMP name: "userCounter64Value"

vtm_virtual_server_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/virtual_servers/*

Virtual servers statistics values.

```
# Example usage of vtm_virtual_server_stats data source...
data "vtm_virtual_server_stats" "my_virtual_server_stats" {
    name = "MyVirtualServer"
}

# Access with...
"${data.vtm_virtual_server_stats.my_virtual_server_stats.<FIELD_NAME>}"
```

Field Name	Description
auth_saml_redirects	Number of times a user agent was redirected to SAML Identity Provider. • Type: UInt64 • SNMP name: "virtualserverAuthSamlRedirects"
auth_saml_responses	Number of times a SAML Response was processed. • Type: UInt64 • SNMP name: "virtualserverAuthSamlResponses"
auth_saml_responses_accepted	Number of times a SAML Response was accepted. • Type: UInt64 • SNMP name: "virtualserverAuthSamlResponsesAccepted"
auth_saml_responses_rejected	Number of times a SAML Response was rejected. • Type: UInt64 • SNMP name: "virtualserverAuthSamlResponsesRejected"
auth_sessions_created	Number of times an authentication session was created. • Type: UInt64 • SNMP name: "virtualserverAuthSessionsCreated"
auth_sessions_rejected	Number of times an authentication session was rejected. • Type: UInt64 • SNMP name: "virtualserverAuthSessionsRejected"

Field Name	Description
auth_sessions_used	Number of times an authentication session was used. • Type: UInt64 • SNMP name: "virtualserverAuthSessionsUsed"
bytes_in	Bytes received by this virtual server from clients. • Type: UInt64 • SNMP name: "virtualserverBytesIn"
bytes_out	Bytes sent by this virtual server to clients. • Type: UInt64 • SNMP name: "virtualserverBytesOut"
cert_status_requests	Number of incoming TLS handshakes for this virtual server with certificate status requests. • Type: UInt • SNMP name: "virtualserverCertStatusRequests"
cert_status_responses	Number of incoming TLS handshakes for this virtual server to which certificate status responses were attached. • Type: UInt • SNMP name: "virtualserverCertStatusResponses"
connect_timed_out	Connections closed by this virtual server because the 'connect_timeout' interval was exceeded. • Type: UInt • SNMP name: "virtualserverConnectTimedOut"
connection_errors	Number of transaction or protocol errors in this virtual server. • Type: UInt • SNMP name: "virtualserverConnectionErrors"

Field Name	Description
connection_failures	Number of connection failures in this virtual server. • Type: UInt • SNMP name: "virtualserverConnectionFailures"
current_conn	TCP connections currently established to this virtual server. • Type: UInt • SNMP name: "virtualserverCurrentConn"
data_timed_out	Connections closed by this virtual server because the 'timeout' interval was exceeded. • Type: UInt • SNMP name: "virtualserverDataTimedOut"
direct_replies	Direct replies from this virtual server, without forwarding to a node. • Type: UInt • SNMP name: "virtualserverDirectReplies"
discard	Connections discarded by this virtual server. • Type: UInt • SNMP name: "virtualserverDiscard"
gzip	Responses which have been compressed by content compression. • Type: UInt • SNMP name: "virtualserverGzip"
gzip_bytes_saved	Bytes of network traffic saved by content compression. • Type: UInt64 • SNMP name: "virtualserverGzipBytesSaved"
http1xx_responses	Number of HTTP 1xx responses returned by this virtual server.

Field Name	Description
	• Type: UInt64 • SNMP name: "virtualserverHTTP1xxResponses"
http2xx_responses	Number of HTTP 2xx responses returned by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTP2xxResponses"
http3xx_responses	Number of HTTP 3xx responses returned by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTP3xxResponses"
http4xx_responses	Number of HTTP 4xx responses returned by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTP4xxResponses"
http5xx_responses	Number of HTTP 5xx responses returned by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTP5xxResponses"
http_cache2xx_responses	Number of HTTP 2xx responses returned from webcache by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTPCache2xxResponses"
http_cache3xx_responses	Number of HTTP 3xx responses returned from webcache by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTPCache3xxResponses"
http_cache4xx_responses	Number of HTTP 4xx responses returned from webcache by this virtual server.

Field Name	Description
	• Type: UInt64 • SNMP name: "virtualserverHTTPCache4xxResponses"
http_cache5xx_responses	Number of HTTP 5xx responses returned from webcache by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTPCache5xxResponses"
http_cache_hit_rate	Percentage hit rate of the web cache for this virtual server. • Type: UInt • SNMP name: "virtualserverHttpCacheHitRate"
http_cache_hits	HTTP responses sent directly from the web cache by this virtual server. • Type: UInt • SNMP name: "virtualserverHttpCacheHits"
http_cache_lookups	HTTP requests that are looked up in the web cache by this virtual server. • Type: UInt • SNMP name: "virtualserverHttpCacheLookups"
http_generated2xx_responses	Number of HTTP 2xx responses generated by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTPGenerated2xxResponses"
http_generated3xx_responses	Number of HTTP 3xx responses generated by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTPGenerated3xxResponses"

Field Name	Description
http_generated4xx_responses	Number of HTTP 4xx responses generated by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTPGenerated4xxResponses"
http_generated5xx_responses	Number of HTTP 5xx responses generated by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTPGenerated5xxResponses"
http_rewrite_cookie	HTTP Set-Cookie headers, supplied by a node, that have been rewritten. • Type: UInt • SNMP name: "virtualserverHttpRewriteCookie"
http_rewrite_location	HTTP Location headers, supplied by a node, that have been rewritten. • Type: UInt • SNMP name: "virtualserverHttpRewriteLocation"
http_server1xx_responses	Number of HTTP 1xx responses returned from a backend server or TrafficScript rule by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTPServer1xxResponses"
http_server2xx_responses	Number of HTTP 2xx responses returned from a backend server or TrafficScript rule by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTPServer2xxResponses"
http_server3xx_responses	Number of HTTP 3xx responses returned from a backend server or TrafficScript rule by this virtual server. • Type: UInt64

Field Name	Description
	• SNMP name: "virtualserverHTTPServer3xxResponses"
http_server4xx_responses	Number of HTTP 4xx responses returned from a backend server or TrafficScript rule by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTPServer4xxResponses"
http_server5xx_responses	Number of HTTP 5xx responses returned from a backend server or TrafficScript rule by this virtual server. • Type: UInt64 • SNMP name: "virtualserverHTTPServer5xxResponses"
keepalive_timed_out	Connections closed by this virtual server because the 'keepalive_timeout' interval was exceeded. • Type: UInt • SNMP name: "virtualserverKeepaliveTimedOut"
max_conn	Maximum number of simultaneous TCP connections this virtual server has processed at any one time. • Type: UInt • SNMP name: "virtualserverMaxConn"
max_duration_timed_out	Connections closed by this virtual server because the 'max_transaction_duration' interval was exceeded. • Type: UInt • SNMP name: "virtualserverMaxDurationTimedOut"
port	The port the virtual server listens on. • Type: UInt • SNMP name: "virtualserverPort"

Field Name	Description
processing_timed_out	Connections closed by this virtual server because the 'timeout' interval was exceeded while waiting for rules or external processing. • Type: UInt • SNMP name: "virtualserverProcessingTimedOut"

protocol	The protocol the virtual server is operating.
	
	• Type: Enum(String)
	
	• SNMP name: "virtualserverProtocol"
	
	• Permitted values:
	
	    "http": http(1)
	
	    "https": https(2)
	
	    "ftp": ftp(3)
	
	    "imaps": imaps(4)
	
	    "imapv2": imapv2(5)
	
	    "imapv3": imapv3(6)
	
	    "imapv4": imapv4(7)
	
	    "pop3": pop3(8)
	
	    "pop3s": pop3s(9)
	
	    "smtp": smtp(10)
	
	    "ldap": ldap(11)
	
	    "ldaps": ldaps(12)
	
	    "telnet": telnet(13)
	
	    "sslforwarding": sslforwarding(14)
	
	    "udpstreaming": udpstreaming(15)

Field Name	Description
	"udp": udp(16)
	"dns": dns(17)
	"genericserverfirst": genericserverfirst(18)
	"genericclientfirst": genericclientfirst(19)
	"dnstcp": dnstcp(20)
	"sipudp": sipudp(21)
	"siptcp": siptcp(22)
	"rtsp": rtsp(23)
	"stream": stream(24)
	"l4acceltcp": l4acceltcp(25)
	"l4acceludp": l4acceludp(26)
	"l4accelgeneric": l4accelgeneric(27)
	"l4accelstateless": l4accelstateless(28)
sip_rejected_requests	Number of SIP requests rejected due to them exceeding the maximum amount of memory allocated to the connection. • Type: UInt • SNMP name: "virtualserverSIPRejectedRequests"
sip_total_calls	Total number of SIP INVITE requests seen by this virtual server. • Type: UInt • SNMP name: "virtualserverSIPTotalCalls"
ssl_cache_lookup	Number of times a lookup for an existing SSL session was performed. • Type: UInt64 • SNMP name: "virtualserverSslCacheLookup"

Field Name	Description
ssl_cache_miss	Number of times a lookup failed to find an existing SSL session. • Type: UInt64 • SNMP name: "virtualserverSslCacheMiss"
ssl_cache_rejected	Number of times an SSL session was found in the cache but rejected and not resumed. • Type: UInt64 • SNMP name: "virtualserverSslCacheRejected"
ssl_cache_resumed	Number of times an SSL session was resumed from the cache. • Type: UInt64 • SNMP name: "virtualserverSslCacheResumed"
ssl_cache_saved	Number of times an SSL session was saved to the cache. • Type: UInt64 • SNMP name: "virtualserverSslCacheSaved"
ssl_hello_retry_requested	Number of times a HelloRetryRequest message was sent to TLS clients. • Type: UInt64 • SNMP name: "virtualserverSslHelloRetryRequested"
ssl_new_session	Number of times a new SSL session was created. • Type: UInt64 • SNMP name: "virtualserverSslNewSession"
ssl_ticket_expired	Number of SSL session tickets that were rejected because they had expired. • Type: UInt64 • SNMP name: "virtualserverSslTicketExpired"

Field Name	Description
ssl_ticket_issued	Number of SSL session tickets that were issued to clients. • Type: UInt64 • SNMP name: "virtualserverSslTicketIssued"
ssl_ticket_key_not_found	Number of SSL session tickets that could not be decrypted because the ticket key they referenced could not be found. • Type: UInt64 • SNMP name: "virtualserverSslTicketKeyNotFound"
ssl_ticket_received	Number of SSL session tickets received. • Type: UInt64 • SNMP name: "virtualserverSslTicketReceived"
ssl_ticket_rejected	Number of SSL session tickets that were rejected for a reason other than because they had expired. • Type: UInt64 • SNMP name: "virtualserverSslTicketRejected"
ssl_ticket_resumed	Number of SSL session tickets that were successfully used to resume a session. • Type: UInt64 • SNMP name: "virtualserverSslTicketResumed"
total_dgram	UDP datagrams processed by this virtual server. • Type: UInt • SNMP name: "virtualserverTotalDgram"
total_http1_requests	HTTP/1.x Requests received by this virtual server. • Type: UInt64 • SNMP name: "virtualserverTotalHTTP1Requests"

Field Name	Description
total_http2_requests	HTTP/2 Requests received by this virtual server. • Type: UInt64 • SNMP name: "virtualserverTotalHTTP2Requests"
total_http_requests	HTTP Requests received by this virtual server. • Type: UInt64 • SNMP name: "virtualserverTotalHTTPRequests"
total_requests	Requests received by this virtual server. • Type: UInt64 • SNMP name: "virtualserverTotalRequests"
udp_bytes_in_dropped	UDP bytes received by this virtual server from clients but failed to be sent to backend servers. • Type: UInt64 • SNMP name: "virtualserverUDPBytesInDropped"
udp_bytes_out_dropped	UDP bytes received by this virtual server from backend servers but failed to be sent to clients. • Type: UInt64 • SNMP name: "virtualserverUDPBytesOutDropped"
udp_timed_out	Connections closed by this virtual server because the 'udp_timeout' interval was exceeded. • Type: UInt • SNMP name: "virtualserverUdpTimedOut"

vtm_cache_web_cache_stats

URI Endpoint: /api/tm/8.3/status/local_tm/statistics/cache/web_cache

Web cache statistics values.

```
# Example usage of vtm_cache_web_cache_stats data source...
data "vtm_cache_web_cache_stats" "my_cache_web_cache_stats" {}


# Access with...
"${data.vtm_cache_web_cache_stats.my_cache_web_cache_stats.<FIELD_NAME>}"
```

Field Name	Description
entries	The number of items in the web cache. • Type: UInt • SNMP name: "webCacheEntries"
hit_rate	The percentage of web cache lookups that succeeded. • Type: UInt • SNMP name: "webCacheHitRate"
hits	Number of times a page has been successfully found in the web cache. • Type: UInt64 • SNMP name: "webCacheHits"
lookups	Number of times a page has been looked up in the web cache. • Type: UInt64 • SNMP name: "webCacheLookups"
max_entries	The maximum number of items in the web cache. • Type: UInt • SNMP name: "webCacheMaxEntries"
mem_maximum	The maximum amount of memory the web cache can use in kilobytes. • Type: UInt • SNMP name: "webCacheMemMaximum"

Field Name	Description
mem_used	Total memory used by the web cache in kilobytes. • Type: UInt • SNMP name: "webCacheMemUsed"
misses	Number of times a page has not been found in the web cache. • Type: UInt64 • SNMP name: "webCacheMisses"
oldest	The age of the oldest item in the web cache (in seconds). • Type: UInt • SNMP name: "webCacheOldest"
url_store_allocated	Amount of allocated space in the web cache URL store. • Type: UInt64 • SNMP name: "webCacheURLStoreAllocated"
url_store_free	Amount of free space in the web cache URL store. • Type: UInt64 • SNMP name: "webCacheURLStoreFree"
url_store_size	Total amount of space in the web cache URL store. • Type: UInt64 • SNMP name: "webCacheURLStoreSize"
url_store_total_allocations	Total number of allocations for the web cache URL store. • Type: UInt64 • SNMP name: "webCacheURLStoreTotalAllocations"
url_store_total_failures	Total number of allocation failures for the web cache URL store. • Type: UInt64

Field Name	Description
	• SNMP name: "webCacheURLStoreTotalFailures"
url_store_total_frees	Total number of blocks freed in the web cache URL store. • Type: UInt64 • SNMP name: "webCacheURLStoreTotalFrees"

System Information

System information data sources allow Terraform to access information about the current state of a Traffic Manager, including errors and service statuses.

```
data "vtm_state" "state_information" {}
```

vtm_backups_full

URI Endpoint: /api/tm/8.3/status/local_tm/backups/full/*

Full backups.

```
# Example usage of vtm_backups_full data source...
data "vtm_backups_full" "my_backups_full" {
    name = "MyBackupsFull"
}

# Access with...
"${data.vtm_backups_full.my_backups_full.<FIELD_NAME>}"
```

Property	Description
name	Name of the object • Type: string • Required: true
backup_description	Description of the backup • Type: String

Property	Description
backup_time_stamp	Time the backup was created. Expressed as a UTC value. • Type: Int
backup_version	Version of the traffic manager used to create the backup • Type: String

vtm_information

URI Endpoint: /api/tm/8.3/status/local_tm/information

Static information for the system.

```
# Example usage of vtm_information data source...
data "vtm_information" "my_information" {}

# Access with...
"${data.vtm_information.my_information.<FIELD_NAME>}"
```

Property	Description
information_platform	The type of platform on which the Traffic Manager instance is running on. • Type: String
information_tm_version	Version number of the Traffic Manager instance. • Type: String
information_uuid	The universally unique identifier for the Traffic Manager instance. • Type: String

vtm_state

URI Endpoint: /api/tm/8.3/status/local_tm/state

State information for the traffic manager.

```
# Example usage of vtm_state data source...
data "vtm_state" "my_state" {}

# Access with...
"${data.vtm_state.my_state.<FIELD_NAME>}"
```

Property	Description

state_error_level	The error_level of the traffic manager.
	
	• Type: Enum(String)
	
	• Permitted values:
	
	     "ok": System has no problems
	
	     "warn": System has minor issues
	
	     "error": System has major issues
	
	     "fatal": System has issues which causes it to die/crash/fail to startup

state_errors	List of configuration errors for the traffic manager • Type: Set(String)

state_failed_nodes	A table of nodes which have failed on the traffic manager
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • node (String): A node which has failed (Required)
	
	• Sub keys:
	
	    • pools (Set(String)): List of pools which use this node.

state_failed_nodes_json	A Traffic Manager REST-compatible JSON representation of the "state_failed_nodes" table property. Use this field with the "vtm_state_failed_nodes_table" data source for dynamic table generation from input variables.

Property	Description
	• Type: String • Required: false • Default value: \<none\>
state_license	Current active license or Community_Edition • Type: String

state_pools	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • name (String): Name of the pool (Required)
	
	• Sub keys:
	
	    • failure_pool (String): Failure pool associated with this pool.
	
	    • active_nodes (Set(String)): List of nodes which are in the active state.
	
	    • draining_nodes (Set(String)): List of nodes which are in the draining state.
	
	    • disabled_nodes (Set(String)): List of nodes which are in the disabled state.

state_pools_json	A Traffic Manager REST-compatible JSON representation of the "state_pools" table property. Use this field with the "vtm_state_pools_table" data source for dynamic table generation from input variables. • Type: String • Required: false

Property	Description
	• Default value: <none>
state_tip_errors	List of traffic IP errors for the traffic manager • Type: Set(String)

state_virtual_servers	A table of virtual server status
	
	• Type: Table
	
	• Required: false
	
	• Primary key:
	
	    • name (String): Name of the virtual server (Required)
	
	• Sub keys:
	
	    • pool (String): Pool associated with this virtual server.
	
	    • port (UInt): Port the virtual server listens on.
	
	    • throughput (UInt64): Through put for the virtual server.
	
	    • ts_redirect_pools (Set(String)): List of pools which may be selected by Traffic Script Request Rules.

state_virtual_servers_json	A Traffic Manager REST-compatible JSON representation of the "state_virtual_servers" table property. Use this field with the "vtm_state_virtual_servers_table" data source for dynamic table generation from input variables. • Type: String • Required: false • Default value: <none>

Table Field Data Sources

When hard-coding a Terraform template, you can include nested data structures that represent the table structures used by some Traffic Manager configuration fields. For example, the "nodes_table" field of the "vtm_pool" resource can have multiple rows specified as shown in the code sample that follows:

```
resource "vtm_pool" "my_pool" {
    name = "MyPool"
    nodes_table {
        node = "192.0.2.1:80"
        state = "active"
    }
    nodes_table {
        node = "192.0.2.2:80"
        state = "active"
    }
    nodes_table {
        node = "192.0.2.3:80"
        state = "disabled"
    }
}
```

To add a new row to the "nodes_table" field through this method, add another "nodes_table {}" block with the relevant field values. Often, however, it is desirable to assign configuration data to a template using input variables, although currently there is no supported method for passing in nested structures.

To facilitate the use-case of passing table configuration into a template through input variables, the Traffic Manager provides table field data sources that can accept standard list variables as inputs to each column of the table field. To identify the corresponding table property, each table field data source uses the naming convention "<RESOURCE_NAME>_<TABLE_FIELD_NAME>_table". Each data source supports the same fields, with the same default values, as the corresponding table and returns a JSON string that can be supplied to a configuration resource. Each data source also contains a "json" field that automatically computes the properly-formatted JSON representation of the table row. These json values can then be assembled into a JSON list and assigned to a special field on the resource object called "<TABLE_FIELD_NAME>_json".

For example, using the "vtm_pool" -> "nodes_table" data from the preceding code sample, use the table field data source "vtm_pool_nodes_table_table" to apply the same configuration through input variables:

```
# In the terraform.tfvars file (or equivalent through the command line)
node_list = ["192.0.2.1:80", "192.0.2.2:80", "192.0.2.3:80"]
```

```
        node_state_list = ["active", "active", "disabled"]


        # In the Terraform template...
        variable "node_list" {
           type = list
        }


        variable "node_state_list" {
           type = list
        }


        data "vtm_pool_nodes_table_table" "nodes_table" {
           count = "${length(var.node_list)}"
           node = "${var.node_list[count.index]}"
           state = "${var.node_state_list[count.index]}"
        }


        resource "vtm_pool" "my_pool" {
           name = "MyPool"
           nodes_table_json = "[${join(",", data.vtm_pool_nodes_table_table.nodes_
table.*.json)}]"
        }
```

The following list contains all available table field data sources:

- vtm_action_arguments_table

- vtm_appliance_nat_many_to_one_all_ports_table

- vtm_appliance_nat_many_to_one_port_locked_table

- vtm_appliance_nat_one_to_one_table

- vtm_appliance_nat_port_mapping_table

- vtm_custom_string_lists_table

- vtm_glb_service_dnssec_keys_table

- vtm_glb_service_location_settings_table

- vtm_global_settings_appliance_returnpath_table

- vtm_log_export_metadata_table

- vtm_monitor_arguments_table

- vtm_pool_nodes_table_table

- vtm_state_failed_nodes_table

- vtm_state_pools_table

- vtm_state_virtual_servers_table

- vtm_traffic_ip_group_ip_mapping_table

- vtm_traffic_manager_appliance_card_table

- vtm_traffic_manager_appliance_sysctl_table

- vtm_traffic_manager_hosts_table

- vtm_traffic_manager_if_table

- vtm_traffic_manager_ip_table

- vtm_traffic_manager_routes_table

- vtm_traffic_manager_trafficip_table

- vtm_user_group_permissions_table

- vtm_virtual_server_ca_sites_table

- vtm_virtual_server_ocsp_issuers_table

- vtm_virtual_server_profile_table

- vtm_virtual_server_server_cert_host_mapping_table

Example Deployment Template

This chapter contains an example template to demonstrate deployment of your Traffic Manager infrastructure through the Terraform provider.

```
provider "vtm" {
    base_url = "https://192.0.2.100:9070/api"
    username = "admin"
    password = "admin"
    verify_ssl_cert = false
}


####################
# TIP GROUPS
####################

resource "vtm_traffic_ip_group" "web_tip" {
    name = "web_tip"
    enabled = true
    ipaddresses = ["192.0.2.200"]
    machines = ["192.0.2.250"]
}



####################
# POOLS
####################

resource "vtm_pool" "html_pool" {
    name = "html_pool"
    load_balancing_algorithm = "weighted_least_connections"
    nodes_table {
        node = "192.0.2.10:80"
        weight = 1
    }
    nodes_table {
        node = "192.0.2.11:80"
        weight = 3
    }
    nodes_table {
        node = "192.0.2.12:80"
```

```
        weight = 3
        state = "disabled"
    }
}


resource "vtm_pool" "images_pool" {
    name = "images_pool"
    nodes_table {
        node = "192.0.2.20:80"
    }
    nodes_table {
        node = "192.0.2.21:80"
    }
}



####################
# RULES
####################

resource "vtm_rule" "use_images_pool" {
    name = "UseImagesPool"
    content = <<EOF
if(http.getPath() == "/images") {
    pool.use("${vtm_pool.images_pool.name}");
}
EOF
}

resource "vtm_rule" "redirect_to_https" {
    name = "RedirectToHTTPS"
    content = <<EOF
$hostHeader = http.getHostHeader();
http.redirect("https://" . $hostHeader);
EOF
}

####################
# SSL CERTS
####################
```

```
resource "vtm_server_key" "ssl_cert" {
    name    = "SSL-Cert"
    note    = ""
    private = <<EOF
-----BEGIN RSA PRIVATE KEY-----
MIIEogIBAAKCAQEApEPRlK+xJbQfUenl9H4nLEkQaH5L8/9F+pcjJW14EdlSkI3s
6bPe+eGtWf0XSzDXzOqAufERWrKyhw21c+UYoTA64i43T9nwvlRtxXxcok+VqmqC
HMCT5V7d82DKXnEEE6J5LwmHo48MsaQsBrjeyGVA8n40JDoM3qC1IlYxEoqouRf8
5eYWYAugL1PSnMT71fZy6VUAaeFRVRwREc3RFxkKa9GEraCaDGp7jfmdrNH1A8Pn
ns3um6kWuRHIliewFhUmc1gTxoQurVTtQh/FQViA6UbQtVCcScYarjqo9dgcuz7R
ABCEBnTx7gcmDeB6VR7luN3MGTc8gupJOjNEawIDAQABAoIBAHl+s8/ulu7VJ+k1
P+EzQZQOwnUXHORuZfrvuI4hRpKlE91ZK+a7JGvcEJSjjowNpz+oHufotrZHv6YG
bLQ4uZvXCWZrWnvULa0I01wjHHzssj0mo+/SPFGFdlJhv6xUmPhQzqMMwGcoEfJ+
BBZAvH6p7Xyt/bDhws0TPoYUDB2yg0VN0lgdjg3S/xR2od8ggZBWow5DV3S0j5ta
QlLojcDgHh6MQ1VOHIUHhillHUKuhlHLhukUHUKhkHKLldMPUrqhmJCaGVSDGr9+
2pjPHz04fp7CZ264rmUyHCGhwhjz4FH1KqkVM5LxoKhbOobwivhrhI/g4skgPxfb
Y7lwxsECgYEA2JgEzk6tnCZO8UfiR+FSvCLWmFJAlHuFasUy8B0jpZIlRNxt88iZ
BhsHIJadC09fmMF6WdE1KTZgNnMnm675FDSSKjk12DhGzU/zRqaBh0NE/Lemqe8/
wwn3oLd3Z++zcRJj7P6NCoQN8BEIJ0voiqje+ppEkumfTQ8VEHxGEuECgYEAwiaN
F4W5Z0Y2EEnkwbz1w+Fzaur0g3balipfjt/Up4xWDnsMdP7xPzny2rGzfqfOpUOF
dRR2YwXGhAcTMpo+Nblg9/0YpJ500RDxMMYEj8bYjTaEFJQ4a7SSvki35pFM0/pT
smhyWA04U/ttbs3+XosN335JsbcopZfT3zF3zMsCgYBWDniCXAJgs1vURAJlGuKb
e6AV30BnfnhxBq8Jdhpus5V5Obe6D661HVIEobL+BmhuMhlhzFy55i/jeq9ua398
p7pn/x998Yb0FlsLbCa0zoZ/fpyKklOcM76eraaUtklumKb5R95UGknLY4kAzAk1
5ojJuzeZw5cWr/JnnWjeIQKBgE09explmBpPI4kdbMXrADeaxQk/SmHW8iWV3AiC
Yh76RO5j49PT7XSDAGwjEE8OQbccAsdOib7heFXkXq3eEWvcQYjHh3tOkxjtzZbi
4MO2j0a27pslUMEAyPStB4TSP6eByrSKuxruv38h4yqXB2Djn3RP0M/EF4axvZfp
HUk7AoGAL2OaOtwy5lk/Oc6bwuPjTAR0wmBX9zelgsLIPiON2jHY287syVzh7lPm
vQGRYZlbGMseXj++s9nQJ24gLTokX2FwGioKvXwFX1ujah7ccJR9iVEwKpQMtDY5
cTiUZkme5oO3Idw6IO115A2EB1/BPpoWBP0M+y2BQYuTGk8F2LU=
-----END RSA PRIVATE KEY-----
EOF
    public  = <<EOF
-----BEGIN CERTIFICATE-----
MIIDIDCCAgigAwIBAgIJAMD7f7Ux92lVMA0GCSqGSIb3DQEBCwUAMD4xCzAJBgNV
BAYTAkdCMRAwDgYDVQQHEwdkYXNkYXNkMQ0wCwYDVQQKEwRCbGFoMQ4wDAYDVQQD
EwVhLmIuYzAeFw0xODAxMjUyMzI3MDNaFw0yODAxMjUyMzI3MDNaMD4xCzAJBgNV
BAYTAkdCMRAwDgYDVQQHEwdkYXNkYXNkMQ0wCwYDVQQKEwRCbGFoMQ4wDAYDVQQD
EwVhLmIuYzCCASIwDQYJKoZIhvcNAQEhkhdakdhJLUihiQoCggEBAKRD0ZSvsSW0
H1Hp5fR+JyxJEWERWEewfrRtIyVteBHZUpCN7Omz3vnhrVn9F0sw18zqgLnxEVqy
socNtXPlGKEwOuIuN0/Z8L5UbcV8XKJPlapqghzAk+Ve3fNgyl5xBBOieS8Jh6OP
```

```
DLGkLAa43shlQPJ+NCQ6DN6gtSJWMRKKqLkX/OXmFmALoC9T0pzE+9X2culVAGnh
UVUcERHN0RcZCmvRhK2gmgxqe435nazR9QPD557N7pupFrkRyJYnsBYVJnNYE8aE
Lq1U7UIfxUFYgOlG0LVQnEnGGq46qPXYHLs+0QAQhAZ08e4HJg3gelUe5bjdzBk3
PILqSTozRGsCAwEAAaMhMB8wHQYDVR0OBBYEFOsQUOxzga482TRQfgcvWsOXHu3k
MA0GCSqGSIb3DQEBCwUAA4IBAQAOufUIugke4ZHRAXYmgM5cUX1MbBUs5S71u+Ao
79RfGkDL1kfvPdAoQx1/EoWc7LRIzvbuIZu6BiarU+/Te6mirmjF+dFdCfEka7cY
ZR5/BvU/+xJNEFDz2bEL0f4LTKnEiloEcUsHAt3vaqRdBGNt3vvpJ5FjyaDXjmpA
idvAjkqXEbUUBgt0kWuaQU8CDCv5FiGr9XhmK8YnoABCsyALbF+NP41EyUfZzt0Z
bj25+V9mexgCGRR6HJI9whhz33v51SXjxlAX5vsDiXRhfhLST7MBGamE6nqew2k9
cMbhPHfTIYYM1ijaqFU/LEXOQ6jTieldVIvC0KVSue7+eQtn
-----END CERTIFICATE-----
EOF
    request = ""
}


#####################
# VIRTUAL SERVERS
#####################

resource "vtm_virtual_server" "http_vs" {
    name = "HTTP"
    enabled = true
    pool = "discard"
    port = 80
    listen_on_any = false
    listen_on_traffic_ips = ["${vtm_traffic_ip_group.web_tip.name}"]
    request_rules = [
        "${vtm_rule.use_images_pool.name}",
        "${vtm_rule.redirect_to_https.name}",
    ]
    web_cache_enabled = true
    web_cache_refresh_time = 30
}

resource "vtm_virtual_server" "https_vs" {
    name = "HTTPS"
    enabled = true
    pool = "${vtm_pool.html_pool.name}"
    port = 443
    listen_on_any = false
    listen_on_traffic_ips = ["${vtm_traffic_ip_group.web_tip.name}"]
```

```
    ssl_decrypt = true
    ssl_server_cert_default = "${vtm_server_key.ssl_cert.name}"
}
```