» avi_useraccountprofile

This data source is used to to get avi_useraccountprofile objects.

» Example Usage

```
data "avi_useraccountprofile" "foo_useraccountprofile" {
    uuid = "useraccountprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search UserAccountProfile by name.
- `uuid` - (Optional) Search UserAccountProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `account_lock_timeout` - Lock timeout period (in minutes).
- `credentials_timeout_threshold` - The time period after which credentials expire.
- `max_concurrent_sessions` - Maximum number of concurrent sessions allowed.
- `max_login_failure_count` - Number of login attempts before lockout.
- `max_password_history_count` - Maximum number of passwords to be maintained in the password history.
- `name` - Name of the object.
- `uuid` - Unique object identifier of the object.

» avi_role

This data source is used to to get avi_role objects.

» Example Usage

```
data "avi_role" "foo_role" {
    uuid = "role-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search Role by name.
- `uuid` - (Optional) Search Role by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `name` - Name of the object.
- `privileges` - List of list.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.

» avi_natpolicy

This data source is used to to get avi_natpolicy objects.

» Example Usage

```
data "avi_natpolicy" "foo_natpolicy" {
    uuid = "natpolicy-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search NatPolicy by name.
- `uuid` - (Optional) Search NatPolicy by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `created_by` - Creator name.
- `description` - Field introduced in 18.2.3.
- `name` - Name of the nat policy.
- `rules` - Nat policy rules.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the nat policy.

» avi_ipaddrgroup

This data source is used to to get avi_ipaddrgroup objects.

» Example Usage

```
data "avi_ipaddrgroup" "foo_ipaddrgroup" {
    uuid = "ipaddrgroup-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search IpAddrGroup by name.
- `uuid` - (Optional) Search IpAddrGroup by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `addrs` - Configure ip address(es).
- `apic_epg_name` - Populate ip addresses from members of this cisco apic epg.
- `country_codes` - Populate the ip address ranges from the geo database for this country.
- `description` - User defined description for the object.
- `ip_ports` - Configure (ip address, port) tuple(s).
- `marathon_app_name` - Populate ip addresses from tasks of this marathon app.
- `marathon_service_port` - Task port associated with marathon service port.
- `name` - Name of the ip address group.
- `prefixes` - Configure ip address prefix(es).
- `ranges` - Configure ip address range(s).
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the ip address group.

» avi_microservicegroup

This data source is used to to get avi_microservicegroup objects.

» Example Usage

```
data "avi_microservicegroup" "foo_microservicegroup" {
    uuid = "microservicegroup-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search MicroServiceGroup by name.
- `uuid` - (Optional) Search MicroServiceGroup by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `created_by` - Creator name.
- `description` - User defined description for the object.
- `name` - Name of the microservice group.
- `service_refs` - Configure microservice(es).
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the microservice group.

» avi_stringgroup

This data source is used to to get avi_stringgroup objects.

» Example Usage

```
data "avi_stringgroup" "foo_stringgroup" {
    uuid = "stringgroup-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search StringGroup by name.
- `uuid` - (Optional) Search StringGroup by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - User defined description for the object.
- `kv` - Configure key value in the string group.
- `name` - Name of the string group.
- `tenant_ref` - It is a reference to an object of type tenant.
- `type` - Type of stringgroup.
- `uuid` - Uuid of the string group.

» avi_trafficcloneprofile

This data source is used to to get avi_trafficcloneprofile objects.

» Example Usage

```
data "avi_trafficcloneprofile" "foo_trafficcloneprofile" {
    uuid = "trafficcloneprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
    cloud_ref = "/api/cloud/?tenant=admin&name=Default-Cloud"
  }
```

» Argument Reference

- `name` - (Optional) Search TrafficCloneProfile by name.
- `uuid` - (Optional) Search TrafficCloneProfile by uuid.
- `cloud_ref` - (Optional) Search TrafficCloneProfile by cloud_ref.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `clone_servers` - Field introduced in 17.1.1.
- `cloud_ref` - It is a reference to an object of type cloud.
- `name` - Name for the traffic clone profile.
- `preserve_client_ip` - Specifies if client ip needs to be preserved to clone destination.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the traffic clone profile.

» avi_webhook

This data source is used to to get avi_webhook objects.

» Example Usage

```
data "avi_webhook" "foo_webhook" {
    uuid = "webhook-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search Webhook by name.
- `uuid` - (Optional) Search Webhook by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `callback_url` - Callback url for the webhook.
- `description` - Field introduced in 17.1.1.
- `name` - The name of the webhook profile.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the webhook profile.
- `verification_token` - Verification token sent back with the callback as-query parameters.

» avi_authprofile

This data source is used to to get avi_authprofile objects.

» Example Usage

```
data "avi_authprofile" "foo_authprofile" {
    uuid = "authprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search AuthProfile by name.
- `uuid` - (Optional) Search AuthProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - User defined description for the object.
- `http` - Http user authentication params.
- `ldap` - Ldap server and directory settings.
- `name` - Name of the auth profile.
- `pa_agent_ref` - Pingaccessagent uuid.
- `saml` - Saml settings.
- `tacacs_plus` - Tacacs+ settings.
- `tenant_ref` - It is a reference to an object of type tenant.
- `type` - Type of the auth profile.
- `uuid` - Uuid of the auth profile.

» avi_sslkeyandcertificate

This data source is used to to get avi_sslkeyandcertificate objects.

» Example Usage

```
data "avi_sslkeyandcertificate" "foo_sslkeyandcertificate" {
    uuid = "sslkeyandcertificate-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search SSLKeyAndCertificate by name.
- `uuid` - (Optional) Search SSLKeyAndCertificate by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `ca_certs` - Ca certificates in certificate chain.

- `certificate` - Dict settings for sslkeyandcertificate.
- `certificate_base64` - States if the certificate is base64 encoded.
- `certificate_management_profile_ref` - It is a reference to an object of type certificatemanagementprofile.
- `created_by` - Creator name.
- `dynamic_params` - Dynamic parameters needed for certificate management profile.
- `enckey_base64` - Encrypted private key corresponding to the private key (e.g.
- `enckey_name` - Name of the encrypted private key (e.g.
- `format` - Format of the key/certificate file.
- `hardwaresecuritymodulegroup_ref` - It is a reference to an object of type hardwaresecuritymodulegroup.
- `key` - Private key.
- `key_base64` - States if the private key is base64 encoded.
- `key_params` - Dict settings for sslkeyandcertificate.
- `key_passphrase` - Passphrase used to encrypt the private key.
- `name` - Name of the object.
- `status` - Enum options - ssl_certificate_finished, ssl_certificate_pending.
- `tenant_ref` - It is a reference to an object of type tenant.
- `type` - Enum options - ssl_certificate_type_virtualservice, ssl_certificate_type_system, ssl_certificate_type_ca.
- `uuid` - Unique object identifier of the object.

» avi_sslprofile

This data source is used to to get avi_sslprofile objects.

» Example Usage

```
data "avi_sslprofile" "foo_sslprofile" {
    uuid = "sslprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search SSLProfile by name.
- `uuid` - (Optional) Search SSLProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `accepted_ciphers` - Ciphers suites represented as defined by u(http //www.openssl.org/docs/apps/ciphers.html).
- `accepted_versions` - Set of versions accepted by the server.
- `cipher_enums` - Enum options - tls_ecdhe_ecdsa_with_aes_128_gcm_sha256, tls_ecdhe_ecdsa_with_aes_256_gcm_sha384, tls_ecdhe_rsa_with_aes_128_gcm_sha256, tls_ecdhe_rsa_with_aes_256_gcm_sha384, tls_ecdhe_ecdsa_with_aes_128_cbc_sha256, tls_ecdhe_ecdsa_with_aes_256_cbc_sha384, tls_ecdhe_rsa_with_aes_128_cbc_sha256, tls_ecdhe_rsa_with_aes_256_cbc_sha384, tls_rsa_with_aes_128_gcm_sha256, tls_rsa_with_aes_256_gcm_sha384, tls_rsa_with_aes_128_cbc_sha256, tls_rsa_with_aes_256_cbc_sha256, tls_ecdhe_ecdsa_with_aes_128_cbc_sha, tls_ecdhe_ecdsa_with_aes_256_cbc_sha, tls_ecdhe_rsa_with_aes_128_cbc_sha, tls_ecdhe_rsa_with_aes_256_cbc_sha, tls_rsa_with_aes_128_cbc_sha, tls_rsa_with_aes_256_cbc_sha, tls_rsa_with_3des_ede_cbc_sha, tls_rsa_with_rc4_128_sha.
- `description` - User defined description for the object.
- `dhparam` - Dh parameters used in ssl.
- `enable_ssl_session_reuse` - Enable ssl session re-use.
- `name` - Name of the object.
- `prefer_client_cipher_ordering` - Prefer the ssl cipher ordering presented by the client during the ssl handshake over the one specified in the ssl profile.
- `send_close_notify` - Send 'close notify' alert message for a clean shutdown of the ssl connection.
- `ssl_rating` - Dict settings for sslprofile.
- `ssl_session_timeout` - The amount of time in seconds before an ssl session expires.
- `tags` - List of list.
- `tenant_ref` - It is a reference to an object of type tenant.
- `type` - Ssl profile type.
- `uuid` - Unique object identifier of the object.

» avi_pkiprofile

This data source is used to to get avi_pkiprofile objects.

» Example Usage

```
data "avi_pkiprofile" "foo_pkiprofile" {
    uuid = "pkiprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
```

```
}
```

» Argument Reference

- `name` - (Optional) Search PKIProfile by name.
- `uuid` - (Optional) Search PKIProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `ca_certs` - List of certificate authorities (root and intermediate) trusted that is used for certificate validation.
- `created_by` - Creator name.
- `crl_check` - When enabled, avi will verify via crl checks that certificates in the trust chain have not been revoked.
- `crls` - Certificate revocation lists.
- `ignore_peer_chain` - When enabled, avi will not trust intermediate and root certs presented by a client.
- `is_federated` - This field describes the object's replication scope.
- `name` - Name of the pki profile.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.
- `validate_only_leaf_crl` - When enabled, avi will only validate the revocation status of the leaf certificate using crl.

» avi_certificatemanagementprofile

This data source is used to to get avi_certificatemanagementprofile objects.

» Example Usage

```
data "avi_certificatemanagementprofile" "foo_certificatemanagementprofile" {
    uuid = "certificatemanagementprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search CertificateManagementProfile by name.
- `uuid` - (Optional) Search CertificateManagementProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `name` - Name of the pki profile.
- `script_params` - List of list.
- `script_path` - Placeholder for description of property script_path of obj type certificatemanagementprofile field type string type str.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.

» avi_ssopolicy

This data source is used to to get avi_ssopolicy objects.

» Example Usage

```
data "avi_ssopolicy" "foo_ssopolicy" {
    uuid = "ssopolicy-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search SSOPolicy by name.
- `uuid` - (Optional) Search SSOPolicy by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `authentication_policy` - Authentication policy settings.
- `authorization_policy` - Authorization policy settings.
- `name` - Name of the sso policy.
- `tenant_ref` - Uuid of the tenant.
- `type` - Sso policy type.
- `uuid` - Uuid of the sso policy.

» avi_l4policyset

This data source is used to to get avi_l4policyset objects.

» Example Usage

```
data "avi_l4policyset" "foo_l4policyset" {
    uuid = "l4policyset-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search L4PolicySet by name.
- `uuid` - (Optional) Search L4PolicySet by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `created_by` - Creator name.
- `description` - Field introduced in 17.2.7.
- `is_internal_policy` - Field introduced in 17.2.7.
- `l4_connection_policy` - Policy to apply when a new transport connection is setup.
- `name` - Name of the l4 policy set.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Id of the l4 policy set.

» avi_scheduler

This data source is used to to get avi_scheduler objects.

» Example Usage

```
data "avi_scheduler" "foo_scheduler" {
    uuid = "scheduler-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search Scheduler by name.
- `uuid` - (Optional) Search Scheduler by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `backup_config_ref` - Backup configuration to be executed by this scheduler.
- `enabled` - Boolean flag to set enabled.
- `end_date_time` - Scheduler end date and time.
- `frequency` - Frequency at which custom scheduler will run.
- `frequency_unit` - Unit at which custom scheduler will run.
- `name` - Name of scheduler.
- `run_mode` - Scheduler run mode.
- `run_script_ref` - Control script to be executed by this scheduler.
- `scheduler_action` - Define scheduler action.
- `start_date_time` - Scheduler start date and time.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.

» avi_backupconfiguration

This data source is used to to get avi_backupconfiguration objects.

» Example Usage

```
data "avi_backupconfiguration" "foo_backupconfiguration" {
    uuid = "backupconfiguration-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search BackupConfiguration by name.
- `uuid` - (Optional) Search BackupConfiguration by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `aws_access_key` - Aws access key id.
- `aws_bucket_id` - Aws bucket.
- `aws_secret_access` - Aws secret access key.
- `backup_file_prefix` - Prefix of the exported configuration file.

- `backup_passphrase` - Passphrase of backup configuration.
- `maximum_backups_stored` - Rotate the backup files based on this count.
- `name` - Name of backup configuration.
- `remote_directory` - Directory at remote destination with write permission for ssh user.
- `remote_hostname` - Remote destination.
- `save_local` - Local backup.
- `ssh_user_ref` - Access credentials for remote destination.
- `tenant_ref` - It is a reference to an object of type tenant.
- `upload_to_remote_host` - Remote backup.
- `upload_to_s3` - Cloud backup.
- `uuid` - Unique object identifier of the object.

» avi_tenant

This data source is used to to get avi_tenant objects.

» Example Usage

```
data "avi_tenant" "foo_tenant" {
    uuid = "tenant-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search Tenant by name.
- `uuid` - (Optional) Search Tenant by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `config_settings` - Dict settings for tenant.
- `created_by` - Creator of this tenant.
- `description` - User defined description for the object.
- `local` - Boolean flag to set local.
- `name` - Name of the object.
- `uuid` - Unique object identifier of the object.

» avi_serviceenginegroup

This data source is used to to get avi_serviceenginegroup objects.

» Example Usage

```
data "avi_serviceenginegroup" "foo_serviceenginegroup" {
    uuid = "serviceenginegroup-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
    cloud_ref = "/api/cloud/?tenant=admin&name=Default-Cloud"
  }
```

» Argument Reference

- `name` - (Optional) Search ServiceEngineGroup by name.
- `uuid` - (Optional) Search ServiceEngineGroup by uuid.
- `cloud_ref` - (Optional) Search ServiceEngineGroup by cloud_ref.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `accelerated_networking` - Enable accelerated networking option for azure se.
- `active_standby` - Service engines in active/standby mode for ha failover.
- `advertise_backend_networks` - Advertise reach-ability of backend server networks via adc through bgp for default gateway feature.
- `aggressive_failure_detection` - Enable aggressive failover configuration for ha.
- `algo` - In compact placement, virtual services are placed on existing ses until max_vs_per_se limit is reached.
- `allow_burst` - Allow ses to be created using burst license.
- `app_cache_percent` - A percent value of total se memory reserved for applicationcaching.
- `app_learning_memory_percent` - A percent value of total se memory reserved for application learning.
- `archive_shm_limit` - Amount of se memory in gb until which shared memory is collected in core archive.
- `async_ssl` - Ssl handshakes will be handled by dedicated ssl threads.requires se reboot.
- `async_ssl_threads` - Number of async ssl threads per se_dp.requires se reboot.

- `auto_rebalance` - If set, virtual services will be automatically migrated when load on an se is less than minimum or more than maximum thresholds.
- `auto_rebalance_capacity_per_se` - Capacities of se for auto rebalance for each criteria.
- `auto_rebalance_criteria` - Set of criteria for se auto rebalance.
- `auto_rebalance_interval` - Frequency of rebalance, if 'auto rebalance' is enabled.
- `auto_redistribute_active_standby_load` - Redistribution of virtual services from the takeover se to the replacement se can cause momentary traffic loss.
- `bgp_state_update_interval` - Bgp peer state update interval.
- `buffer_se` - Excess service engine capacity provisioned for ha failover.
- `cloud_ref` - It is a reference to an object of type cloud.
- `config_debugs_on_all_cores` - Enable config debugs on all cores of se.
- `connection_memory_percentage` - Percentage of memory for connection state.
- `cpu_reserve` - Boolean flag to set cpu_reserve.
- `cpu_socket_affinity` - Allocate all the cpu cores for the service engine virtual machines on the same cpu socket.
- `custom_securitygroups_data` - Custom security groups to be associated with data vnics for se instances in openstack and aws clouds.
- `custom_securitygroups_mgmt` - Custom security groups to be associated with management vnic for se instances in openstack and aws clouds.
- `custom_tag` - Custom tag will be used to create the tags for se instance in aws.
- `data_network_id` - Subnet used to spin up the data nic for service engines, used only for azure cloud.
- `datascript_timeout` - Number of instructions before datascript times out.
- `dedicated_dispatcher_core` - Dedicate the core that handles packet receive/transmit from the network to just the dispatching function.
- `description` - User defined description for the object.
- `disable_avi_securitygroups` - By default, avi creates and manages security groups along with custom sg provided by user.
- `disable_csum_offloads` - Stop using tcp/udp and ip checksum offload features of nics.
- `disable_gro` - Disable generic receive offload (gro) in dpdk poll-mode driver packet receive path.
- `disable_se_memory_check` - If set, disable the config memory check done in service engine.
- `disable_tso` - Disable tcp segmentation offload (tso) in dpdk poll-mode driver packet transmit path.
- `disk_per_se` - Amount of disk space for each of the service engine virtual machines.
- `distribute_load_active_standby` - Use both the active and standby

service engines for virtual service placement in the legacy active standby ha mode.

- `distribute_queues` - Distributes queue ownership among cores so multiple cores handle dispatcher duties.requires se reboot.
- `distribute_vnics` - Distributes vnic ownership among cores so multiple cores handle dispatcher duties.requires se reboot.
- `enable_gratarp_permanent` - Enable gratarp for vip_ip.
- `enable_hsm_priming` - (this is a beta feature).
- `enable_multi_lb` - Applicable only for azure cloud with basic sku lb.
- `enable_routing` - Enable routing for this serviceenginegroup .
- `enable_vip_on_all_interfaces` - Enable vip on all interfaces of se.
- `enable_vmac` - Use virtual mac address for interfaces on which floating interface ips are placed.
- `ephemeral_portrange_end` - End local ephemeral port number for outbound connections.
- `ephemeral_portrange_start` - Start local ephemeral port number for outbound connections.
- `extra_config_multiplier` - Multiplier for extra config to support large vs/pool config.
- `extra_shared_config_memory` - Extra config memory to support large geo db configuration.
- `floating_intf_ip` - If serviceenginegroup is configured for legacy 1+1 active standby ha mode, floating ip's will be advertised only by the active se in the pair.
- `floating_intf_ip_se_2` - If serviceenginegroup is configured for legacy 1+1 active standby ha mode, floating ip's will be advertised only by the active se in the pair.
- `flow_table_new_syn_max_entries` - Maximum number of flow table entries that have not completed tcp three-way handshake yet.
- `free_list_size` - Number of entries in the free list.
- `gratarp_permanent_periodicity` - Gratarp periodicity for vip-ip.
- `ha_mode` - High availability mode for all the virtual services using this service engine group.
- `hardwaresecuritymodulegroup_ref` - It is a reference to an object of type hardwaresecuritymodulegroup.
- `heap_minimum_config_memory` - Minimum required heap memory to apply any configuration.
- `hm_on_standby` - Enable active health monitoring from the standby se for all placed virtual services.
- `host_attribute_key` - Key of a (key, value) pair identifying a label for a set of nodes usually in container clouds.
- `host_attribute_value` - Value of a (key, value) pair identifying a label for a set of nodes usually in container clouds.
- `host_gateway_monitor` - Enable the host gateway monitor when service engine is deployed as docker container.
- `hypervisor` - Override default hypervisor.

- `ignore_rtt_threshold` - Ignore rtt samples if it is above threshold.
- `ingress_access_data` - Program se security group ingress rules to allow vip data access from remote cidr type.
- `ingress_access_mgmt` - Program se security group ingress rules to allow ssh/icmp management access from remote cidr type.
- `instance_flavor` - Instance/flavor name for se instance.
- `iptables` - Iptable rules.
- `least_load_core_selection` - Select core with least load for new flow.
- `license_tier` - Specifies the license tier which would be used.
- `license_type` - If no license type is specified then default license enforcement for the cloud type is chosen.
- `log_disksz` - Maximum disk capacity (in mb) to be allocated to an se.
- `max_cpu_usage` - When cpu usage on an se exceeds this threshold, virtual services hosted on this se may be rebalanced to other ses to reduce load.
- `max_memory_per_mempool` - Max bytes that can be allocated in a single mempool.
- `max_public_ips_per_lb` - Applicable to azure platform only.
- `max_rules_per_lb` - Applicable to azure platform only.
- `max_scaleout_per_vs` - Maximum number of active service engines for the virtual service.
- `max_se` - Maximum number of services engines in this group.
- `max_vs_per_se` - Maximum number of virtual services that can be placed on a single service engine.
- `mem_reserve` - Boolean flag to set mem_reserve.
- `memory_for_config_update` - Indicates the percent of memory reserved for config updates.
- `memory_per_se` - Amount of memory for each of the service engine virtual machines.
- `mgmt_network_ref` - Management network to use for avi service engines.
- `mgmt_subnet` - Management subnet to use for avi service engines.
- `min_cpu_usage` - When cpu usage on an se falls below the minimum threshold, virtual services hosted on the se may be consolidated onto other underutilized ses.
- `min_scaleout_per_vs` - Minimum number of active service engines for the virtual service.
- `min_se` - Minimum number of services engines in this group (relevant for se autorebalance only).
- `minimum_connection_memory` - Indicates the percent of memory reserved for connections.
- `n_log_streaming_threads` - Number of threads to use for log streaming.
- `name` - Name of the object.
- `non_significant_log_throttle` - This setting limits the number of non-significant logs generated per second per core on this se.
- `num_dispatcher_cores` - Number of dispatcher cores (0,1,2,4,8 or 16).
- `num_flow_cores_sum_changes_to_ignore` - Number of changes in num flow cores sum to ignore.

- `openstack_availability_zones` - Field introduced in 17.1.1.
- `openstack_mgmt_network_name` - Avi management network name.
- `openstack_mgmt_network_uuid` - Management network uuid.
- `os_reserved_memory` - Amount of extra memory to be reserved for use by the operating system on a service engine.
- `per_app` - Per-app se mode is designed for deploying dedicated load balancers per app (vs).
- `placement_mode` - If placement mode is 'auto', virtual services are automatically placed on service engines.
- `realtime_se_metrics` - Enable or disable real time se metrics.
- `reboot_on_panic` - Reboot the vm or host on kernel panic.
- `se_bandwidth_type` - Select the se bandwidth for the bandwidth license.
- `se_deprovision_delay` - Duration to preserve unused service engine virtual machines before deleting them.
- `se_dos_profile` - Dict settings for serviceenginegroup.
- `se_dp_vnic_queue_stall_event_sleep` - Time (in seconds) service engine waits for after generating a vnic transmit queue stall event before resetting thenic.
- `se_dp_vnic_queue_stall_threshold` - Number of consecutive transmit failures to look for before generating a vnic transmit queue stall event.
- `se_dp_vnic_queue_stall_timeout` - Time (in milliseconds) to wait for network/nic recovery on detecting a transmit queue stall after which service engine resets the nic.
- `se_dp_vnic_restart_on_queue_stall_count` - Number of consecutive transmit queue stall events in se_dp_vnic_stall_se_restart_window to look for before restarting se.
- `se_dp_vnic_stall_se_restart_window` - Window of time (in seconds) during which se_dp_vnic_restart_on_queue_stall_count number of consecutive stalls results in a se restart.
- `se_dpdk_pmd` - Determines if dpdk pool mode driver should be used or not 0 automatically determine based on hypervisor/nic type 1 unconditionally use dpdk poll mode driver 2 don't use dpdk poll mode driver.requires se reboot.
- `se_flow_probe_retries` - Flow probe retry count if no replies are received.requires se reboot.
- `se_flow_probe_retry_timer` - Timeout in milliseconds for flow probe retries.requires se reboot.
- `se_ipc_udp_port` - Udp port for se_dp ipc in docker bridge mode.
- `se_lro` - Enable or disable large receive optimization for vnics.
- `se_name_prefix` - Prefix to use for virtual machine name of service engines.
- `se_pcap_lookahead` - Enables lookahead mode of packet receive in pcap mode.
- `se_pcap_pkt_count` - Max number of packets the pcap interface can hold and if the value is 0 the optimum value will be chosen.
- `se_pcap_pkt_sz` - Max size of each packet in the pcap interface.

- `se_pcap_reinit_frequency` - Frequency in seconds at which periodically a pcap reinit check is triggered.
- `se_pcap_reinit_threshold` - Threshold for input packet receive errors in pcap mode exceeding which a pcap reinit is triggered.
- `se_probe_port` - Tcp port on se where echo service will be run.
- `se_remote_punt_udp_port` - Udp port for punted packets in docker bridge mode.
- `se_routing` - Enable routing via service engine datapath.
- `se_sb_dedicated_core` - Sideband traffic will be handled by a dedicated core.requires se reboot.
- `se_sb_threads` - Number of sideband threads per se.requires se reboot.
- `se_thread_multiplier` - Multiplier for se threads based on vcpu.
- `se_tracert_port_range` - Traceroute port range.
- `se_tunnel_mode` - Determines if dsr from secondary se is active or not 0 automatically determine based on hypervisor type.
- `se_tunnel_udp_port` - Udp port for tunneled packets from secondary to primary se in docker bridge mode.requires se reboot.
- `se_tx_batch_size` - Number of packets to batch for transmit to the nic.
- `se_udp_encap_ipc` - Determines if se-se ipc messages are encapsulated in a udp header 0 automatically determine based on hypervisor type.
- `se_use_dpdk` - Determines if dpdk library should be used or not 0 automatically determine based on hypervisor type 1 use dpdk if pcap is not enabled 2 don't use dpdk.
- `se_vs_hb_max_pkts_in_batch` - Maximum number of aggregated vs heartbeat packets to send in a batch.
- `se_vs_hb_max_vs_in_pkt` - Maximum number of virtualservices for which heartbeat messages are aggregated in one packet.
- `self_se_election` - Enable ses to elect a primary amongst themselves in the absence of a connectivity to controller.
- `service_ip6_subnets` - Ipv6 subnets assigned to the se group.
- `service_ip_subnets` - Subnets assigned to the se group.
- `shm_minimum_config_memory` - Minimum required shared memory to apply any configuration.
- `significant_log_throttle` - This setting limits the number of significant logs generated per second per core on this se.
- `ssl_preprocess_sni_hostname` - (beta) preprocess ssl client hello for sni hostname extension.if set to true, this will apply sni child's ssl protocol(s), if they are different from sni parent's allowed ssl protocol(s).
- `tenant_ref` - It is a reference to an object of type tenant.
- `udf_log_throttle` - This setting limits the number of udf logs generated per second per core on this se.
- `use_standard_alb` - Use standard sku azure load balancer.
- `uuid` - Unique object identifier of the object.
- `vcenter_clusters` - Dict settings for serviceenginegroup.
- `vcenter_datastore_mode` - Enum options - vcenter_datastore_any, vcenter_datastore_local, vcenter_datastore_shared.

- `vcenter_datastores` - List of list.
- `vcenter_datastores_include` - Boolean flag to set vcenter_datastores_include.
- `vcenter_folder` - Folder to place all the service engine virtual machines in vcenter.
- `vcenter_hosts` - Dict settings for serviceenginegroup.
- `vcpus_per_se` - Number of vcpus for each of the service engine virtual machines.
- `vip_asg` - When vip_asg is set, vip configuration will be managed by avi.user will be able to configure vip_asg or vips individually at the time of create.
- `vs_host_redundancy` - Ensure primary and secondary service engines are deployed on different physical hosts.
- `vs_scalein_timeout` - Time to wait for the scaled in se to drain existing flows before marking the scalein done.
- `vs_scalein_timeout_for_upgrade` - During se upgrade, time to wait for the scaled-in se to drain existing flows before marking the scalein done.
- `vs_scaleout_timeout` - Time to wait for the scaled out se to become ready before marking the scaleout done.
- `vs_se_scaleout_additional_wait_time` - Wait time for sending scale-out ready notification after virtual service is marked up.
- `vs_se_scaleout_ready_timeout` - Timeout in seconds for service engine to sendscaleout ready notification of a virtual service.
- `vs_switchover_timeout` - During se upgrade in a legacy active/standby segroup, time to wait for the new primary se to accept flows before marking the switchover done.
- `vss_placement` - Parameters to place virtual services on only a subset of the cores of an se.
- `vss_placement_enabled` - If set, virtual services will be placed on only a subset of the cores of an se.
- `waf_mempool` - Enable memory pool for waf.requires se reboot.
- `waf_mempool_size` - Memory pool size used for waf.requires se reboot.

» avi_networkservice

This data source is used to to get avi_networkservice objects.

» Example Usage

```
data "avi_networkservice" "foo_networkservice" {
    uuid = "networkservice-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
    cloud_ref = "/api/cloud/?tenant=admin&name=Default-Cloud"
  }
```

» Argument Reference

- `name` - (Optional) Search NetworkService by name.
- `uuid` - (Optional) Search NetworkService by uuid.
- `cloud_ref` - (Optional) Search NetworkService by cloud_ref.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `cloud_ref` - It is a reference to an object of type cloud.
- `name` - Name of the networkservice.
- `routing_service` - Routing information of the networkservice.
- `se_group_ref` - Service engine group to which the service is applied.
- `service_type` - Indicates the type of networkservice.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the networkservice.
- `vrf_ref` - Vrf context to which the service is scoped.

» avi_dnspolicy

This data source is used to to get avi_dnspolicy objects.

» Example Usage

```
data "avi_dnspolicy" "foo_dnspolicy" {
    uuid = "dnspolicy-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search DnsPolicy by name.
- `uuid` - (Optional) Search DnsPolicy by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `created_by` - Creator name.
- `description` - Field introduced in 17.1.1.

- `name` - Name of the dns policy.
- `rule` - Dns rules.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the dns policy.

» avi_hardwaresecuritymodulegroup

This data source is used to to get avi_hardwaresecuritymodulegroup objects.

» Example Usage

```
data "avi_hardwaresecuritymodulegroup" "foo_hardwaresecuritymodulegroup" {
    uuid = "hardwaresecuritymodulegroup-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search HardwareSecurityModuleGroup by name.
- `uuid` - (Optional) Search HardwareSecurityModuleGroup by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `hsm` - Hardware security module configuration.
- `name` - Name of the hsm group configuration object.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the hsm group configuration object.

» avi_vrfcontext

This data source is used to to get avi_vrfcontext objects.

» Example Usage

```
data "avi_vrfcontext" "foo_vrfcontext" {
    uuid = "vrfcontext-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
```

```
    cloud_ref = "/api/cloud/?tenant=admin&name=Default-Cloud"
  }
```

» Argument Reference

- `name` - (Optional) Search VrfContext by name.
- `uuid` - (Optional) Search VrfContext by uuid.
- `cloud_ref` - (Optional) Search VrfContext by cloud_ref.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `bgp_profile` - Bgp local and peer info.
- `cloud_ref` - It is a reference to an object of type cloud.
- `debugvrfcontext` - Configure debug flags for vrf.
- `description` - User defined description for the object.
- `gateway_mon` - Configure ping based heartbeat check for gateway in service engines of vrf.
- `internal_gateway_monitor` - Configure ping based heartbeat check for all default gateways in service engines of vrf.
- `name` - Name of the object.
- `static_routes` - List of list.
- `system_default` - Boolean flag to set system_default.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.

» avi_securitypolicy

This data source is used to to get avi_securitypolicy objects.

» Example Usage

```
data "avi_securitypolicy" "foo_securitypolicy" {
    uuid = "securitypolicy-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search SecurityPolicy by name.

- `uuid` - (Optional) Search SecurityPolicy by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - Security policy is used to specify various configuration information used to perform distributed denial of service (ddos) attacks detection and mitigation.
- `dns_attacks` - Attacks utilizing the dns protocol operations.
- `dns_policy_index` - Index of the dns policy to use for the mitigation rules applied to the dns attacks.
- `name` - The name of the security policy.
- `network_security_policy_index` - Index of the network security policy to use for the mitigation rules applied to the attacks.
- `oper_mode` - Mode of dealing with the attacks - perform detection only, or detect and mitigate the attacks.
- `tcp_attacks` - Attacks utilizing the tcp protocol operations.
- `tenant_ref` - Tenancy of the security policy.
- `udp_attacks` - Attacks utilizing the udp protocol operations.
- `uuid` - The uuid of the security policy.

» avi_protocolparser

This data source is used to to get avi_protocolparser objects.

» Example Usage

```
data "avi_protocolparser" "foo_protocolparser" {
    uuid = "protocolparser-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search ProtocolParser by name.
- `uuid` - (Optional) Search ProtocolParser by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - Description of the protocol parser.
- `name` - Name of the protocol parser.
- `parser_code` - Command script provided inline.
- `tenant_ref` - Tenant uuid of the protocol parser.
- `uuid` - Uuid of the protocol parser.

» avi_cloudproperties

This data source is used to to get avi_cloudproperties objects.

» Example Usage

```
data "avi_cloudproperties" "foo_cloudproperties" {
    uuid = "cloudproperties-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search CloudProperties by name.
- `uuid` - (Optional) Search CloudProperties by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `cc_props` - Cloudconnector properties.
- `cc_vtypes` - Cloud types supported by cloudconnector.
- `hyp_props` - Hypervisor properties.
- `info` - Properties specific to a cloud type.
- `uuid` - Unique object identifier of the object.

» avi_applicationpersistenceprofile

This data source is used to to get avi_applicationpersistenceprofile objects.

» Example Usage

```
data "avi_applicationpersistenceprofile" "foo_applicationpersistenceprofile" {
    uuid = "applicationpersistenceprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search ApplicationPersistenceProfile by name.
- `uuid` - (Optional) Search ApplicationPersistenceProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `app_cookie_persistence_profile` - Specifies the application cookie persistence profile parameters.
- `description` - User defined description for the object.
- `hdr_persistence_profile` - Specifies the custom http header persistence profile parameters.
- `http_cookie_persistence_profile` - Specifies the http cookie persistence profile parameters.
- `ip_persistence_profile` - Specifies the client ip persistence profile parameters.
- `is_federated` - This field describes the object's replication scope.
- `name` - A user-friendly name for the persistence profile.
- `persistence_type` - Method used to persist clients to the same server for a duration of time or a session.
- `server_hm_down_recovery` - Specifies behavior when a persistent server has been marked down by a health monitor.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the persistence profile.

» avi_backup

This data source is used to to get avi_backup objects.

» Example Usage

```
data "avi_backup" "foo_backup" {
    uuid = "backup-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
```

```
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search Backup by name.
- `uuid` - (Optional) Search Backup by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `backup_config_ref` - Backupconfiguration information.
- `file_name` - The file name of backup.
- `local_file_url` - Url to download the backup file.
- `remote_file_url` - Url to download the backup file.
- `scheduler_ref` - Scheduler information.
- `tenant_ref` - It is a reference to an object of type tenant.
- `timestamp` - Unix timestamp of when the backup file is created.
- `uuid` - Unique object identifier of the object.

» avi_networksecuritypolicy

This data source is used to to get avi_networksecuritypolicy objects.

» Example Usage

```
data "avi_networksecuritypolicy" "foo_networksecuritypolicy" {
    uuid = "networksecuritypolicy-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search NetworkSecurityPolicy by name.
- `uuid` - (Optional) Search NetworkSecurityPolicy by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `cloud_config_cksum` - Checksum of cloud configuration for network sec policy.
- `created_by` - Creator name.
- `description` - User defined description for the object.
- `name` - Name of the object.
- `rules` - List of list.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.

» avi_seproperties

This data source is used to to get avi_seproperties objects.

» Example Usage

```
data "avi_seproperties" "foo_seproperties" {
    uuid = "seproperties-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search SeProperties by name.
- `uuid` - (Optional) Search SeProperties by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `se_agent_properties` - Dict settings for seproperties.
- `se_bootup_properties` - Dict settings for seproperties.
- `se_runtime_properties` - Dict settings for seproperties.
- `uuid` - Unique object identifier of the object.

» avi_pingaccessagent

This data source is used to to get avi_pingaccessagent objects.

» Example Usage

```
data "avi_pingaccessagent" "foo_pingaccessagent" {
    uuid = "pingaccessagent-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search PingAccessAgent by name.
- `uuid` - (Optional) Search PingAccessAgent by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - Field introduced in 18.2.3.
- `name` - Name of the pingaccess agent.
- `pingaccess_pool_ref` - Pool containing a primary pingaccess server, as well as any failover servers included in the agent.properties file.
- `primary_server` - The ip and port of the primary pingaccess server.
- `properties_file_data` - Pingaccessagent's agent.properties file generated by pingaccess server.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the pingaccess agent.

» avi_gslbgeodbprofile

This data source is used to to get avi_gslbgeodbprofile objects.

» Example Usage

```
data "avi_gslbgeodbprofile" "foo_gslbgeodbprofile" {
    uuid = "gslbgeodbprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search GslbGeoDbProfile by name.
- `uuid` - (Optional) Search GslbGeoDbProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - Field introduced in 17.1.1.
- `entries` - List of geodb entries.
- `is_federated` - This field indicates that this object is replicated across gslb federation.
- `name` - A user-friendly name for the geodb profile.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the geodb profile.

» avi_gslbservice

This data source is used to to get avi_gslbservice objects.

» Example Usage

```
data "avi_gslbservice" "foo_gslbservice" {
    uuid = "gslbservice-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search GslbService by name.
- `uuid` - (Optional) Search GslbService by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `application_persistence_profile_ref` - The federated application persistence associated with gslbservice site persistence functionality.
- `controller_health_status_enabled` - Gs member's overall health status is derived based on a combination of controller and datapath health-status inputs.
- `created_by` - Creator name.
- `description` - User defined description for the object.
- `domain_names` - Fully qualified domain name of the gslb service.
- `down_response` - Response to the client query when the gslb service is down.

- `enabled` - Enable or disable the gslb service.
- `groups` - Select list of pools belonging to this gslb service.
- `health_monitor_refs` - Verify vs health by applying one or more health monitors.
- `health_monitor_scope` - Health monitor probe can be executed for all the members or it can be executed only for third-party members.
- `hm_off` - This field is an internal field and is used in se.
- `is_federated` - This field indicates that this object is replicated across gslb federation.
- `min_members` - The minimum number of members to distribute traffic to.
- `name` - Name for the gslb service.
- `num_dns_ip` - Number of ip addresses of this gslb service to be returned by the dns service.
- `pool_algorithm` - The load balancing algorithm will pick a gslb pool within the gslb service list of available pools.
- `site_persistence_enabled` - Enable site-persistence for the gslbservice.
- `tenant_ref` - It is a reference to an object of type tenant.
- `ttl` - Ttl value (in seconds) for records served for this gslb service by the dns service.
- `use_edns_client_subnet` - Use the client ip subnet from the edns option as source ipaddress for client geo-location and consistent hash algorithm.
- `uuid` - Uuid of the gslb service.
- `wildcard_match` - Enable wild-card match of fqdn if an exact match is not found in the dns table, the longest match is chosen by wild-carding the fqdn in the dns request.

» avi_gslb

This data source is used to to get avi_gslb objects.

» Example Usage

```
data "avi_gslb" "foo_gslb" {
    uuid = "gslb-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search Gslb by name.
- `uuid` - (Optional) Search Gslb by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `async_interval` - Frequency with which messages are propagated to vs mgr.
- `clear_on_max_retries` - Max retries after which the remote site is treated as a fresh start.
- `client_ip_addr_group` - Group to specify if the client ip addresses are public or private.
- `description` - User defined description for the object.
- `dns_configs` - Sub domain configuration for the gslb.
- `error_resync_interval` - Frequency with which errored messages are resynced to follower sites.
- `is_federated` - This field indicates that this object is replicated across gslb federation.
- `leader_cluster_uuid` - Mark this site as leader of gslb configuration.
- `maintenance_mode` - This field disables the configuration operations on the leader for all federated objects.
- `name` - Name for the gslb object.
- `send_interval` - Frequency with which group members communicate.
- `send_interval_prior_to_maintenance_mode` - The user can specify a send-interval while entering maintenance mode.
- `sites` - Select avi site member belonging to this gslb.
- `tenant_ref` - It is a reference to an object of type tenant.
- `third_party_sites` - Third party site member belonging to this gslb.
- `uuid` - Uuid of the gslb object.
- `view_id` - The view-id is used in change-leader mode to differentiate partitioned groups while they have the same gslb namespace.

» avi_cluster

This data source is used to to get avi_cluster objects.

» Example Usage

```
data "avi_cluster" "foo_cluster" {
    uuid = "cluster-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search Cluster by name.
- `uuid` - (Optional) Search Cluster by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `name` - Name of the object.
- `nodes` - List of list.
- `rejoin_nodes_automatically` - Re-join cluster nodes automatically in the event one of the node is reset to factory.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.
- `virtual_ip` - A virtual ip address.

» avi_clusterclouddetails

This data source is used to to get avi_clusterclouddetails objects.

» Example Usage

```
data "avi_clusterclouddetails" "foo_clusterclouddetails" {
    uuid = "clusterclouddetails-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search ClusterCloudDetails by name.
- `uuid` - (Optional) Search ClusterCloudDetails by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `azure_info` - Azure info to configure cluster_vip on the controller.
- `name` - Field introduced in 17.2.5.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Field introduced in 17.2.5.

» avi_wafpolicy

This data source is used to to get avi_wafpolicy objects.

» Example Usage

```
data "avi_wafpolicy" "foo_wafpolicy" {
    uuid = "wafpolicy-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search WafPolicy by name.
- `uuid` - (Optional) Search WafPolicy by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `allow_mode_delegation` - Allow rules to overwrite the policy mode.
- `created_by` - Creator name.
- `crs_groups` - Waf rules are categorized in to groups based on their characterization.
- `description` - Field introduced in 17.2.1.
- `enable_app_learning` - Enable application learning for this waf policy.
- `failure_mode` - Waf policy failure mode.
- `mode` - Waf policy mode.
- `name` - Field introduced in 17.2.1.
- `paranoia_level` - Waf ruleset paranoia mode.
- `positive_security_model` - The positive security model.
- `post_crs_groups` - Waf rules are categorized in to groups based on their characterization.
- `pre_crs_groups` - Waf rules are categorized in to groups based on their characterization.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Field introduced in 17.2.1.
- `waf_crs_ref` - Waf core ruleset used for the crs part of this policy.
- `waf_profile_ref` - Waf profile for waf policy.
- `whitelist` - A set of rules which describe conditions under which the request will bypass the waf.

» avi_wafcrs

This data source is used to to get avi_wafcrs objects.

» Example Usage

```
data "avi_wafcrs" "foo_wafcrs" {
    uuid = "wafcrs-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search WafCRS by name.
- `uuid` - (Optional) Search WafCRS by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - A short description of this ruleset.
- `groups` - Waf rules are sorted in groups based on their characterization.
- `integrity` - Integrity protection value.
- `name` - The name of this ruleset object.
- `release_date` - The release date of this version in rfc 3339 / iso 8601 format.
- `tenant_ref` - Tenant that this object belongs to.
- `uuid` - Field introduced in 18.1.1.
- `version` - The version of this ruleset object.

» avi_wafprofile

This data source is used to to get avi_wafprofile objects.

» Example Usage

```
data "avi_wafprofile" "foo_wafprofile" {
    uuid = "wafprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search WafProfile by name.
- `uuid` - (Optional) Search WafProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `config` - Config params for waf.
- `description` - Field introduced in 17.2.1.
- `files` - List of data files used for waf rules.
- `name` - Field introduced in 17.2.1.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Field introduced in 17.2.1.

» avi_wafpolicypsmgroup

This data source is used to to get avi_wafpolicypsmgroup objects.

» Example Usage

```
data "avi_wafpolicypsmgroup" "foo_wafpolicypsmgroup" {
    uuid = "wafpolicypsmgroup-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search WafPolicyPSMGroup by name.
- `uuid` - (Optional) Search WafPolicyPSMGroup by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - Freetext comment about this group.
- `enable` - Enable or disable this waf rule group.
- `hit_action` - If a rule in this group matches the match_value pattern, this action will be executed.
- `is_learning_group` - This field indicates that this group is used for learning.

- `locations` - Positive security model locations.
- `miss_action` - If a rule in this group does not match the match_value pattern, this action will be executed.
- `name` - User defined name of the group.
- `tenant_ref` - Tenant that this object belongs to.
- `uuid` - Uuid of this object.

» avi_snmptrapprofile

This data source is used to to get avi_snmptrapprofile objects.

» Example Usage

```
data "avi_snmptrapprofile" "foo_snmptrapprofile" {
    uuid = "snmptrapprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search SnmpTrapProfile by name.
- `uuid` - (Optional) Search SnmpTrapProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `name` - A user-friendly name of the snmp trap configuration.
- `tenant_ref` - It is a reference to an object of type tenant.
- `trap_servers` - The ip address or hostname of the snmp trap destination server.
- `uuid` - Uuid of the snmp trap profile object.

» avi_systemconfiguration

This data source is used to to get avi_systemconfiguration objects.

» Example Usage

```
data "avi_systemconfiguration" "foo_systemconfiguration" {
    uuid = "systemconfiguration-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search SystemConfiguration by name.
- `uuid` - (Optional) Search SystemConfiguration by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `admin_auth_configuration` - Dict settings for systemconfiguration.
- `default_license_tier` - Specifies the default license tier which would be used by new clouds.
- `dns_configuration` - Dict settings for systemconfiguration.
- `dns_virtualservice_refs` - Dns virtualservices hosting fqdn records for applications across avi vantage.
- `docker_mode` - Boolean flag to set docker_mode.
- `email_configuration` - Dict settings for systemconfiguration.
- `global_tenant_config` - Dict settings for systemconfiguration.
- `linux_configuration` - Dict settings for systemconfiguration.
- `mgmt_ip_access_control` - Configure ip access control for controller to restrict open access.
- `ntp_configuration` - Dict settings for systemconfiguration.
- `portal_configuration` - Dict settings for systemconfiguration.
- `proxy_configuration` - Dict settings for systemconfiguration.
- `secure_channel_configuration` - Configure secure channel properties.
- `snmp_configuration` - Dict settings for systemconfiguration.
- `ssh_ciphers` - Allowed ciphers list for ssh to the management interface on the controller and service engines.
- `ssh_hmacs` - Allowed hmac list for ssh to the management interface on the controller and service engines.
- `uuid` - Unique object identifier of the object.
- `welcome_workflow_complete` - This flag is set once the initial controller setup workflow is complete.

» avi_controllersite

This data source is used to to get avi_controllersite objects.

» Example Usage

```
data "avi_controllersite" "foo_controllersite" {
    uuid = "controllersite-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search ControllerSite by name.
- `uuid` - (Optional) Search ControllerSite by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `address` - Ip address or a dns resolvable, fully qualified domain name of the site controller cluster.
- `name` - Name for the site controller cluster.
- `port` - The controller site cluster's rest api port number.
- `tenant_ref` - Reference for the tenant.
- `uuid` - Reference for the site controller cluster.

» avi_networkprofile

This data source is used to to get avi_networkprofile objects.

» Example Usage

```
data "avi_networkprofile" "foo_networkprofile" {
    uuid = "networkprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search NetworkProfile by name.
- `uuid` - (Optional) Search NetworkProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `connection_mirror` - When enabled, avi mirrors all tcp fastpath connections to standby.
- `description` - User defined description for the object.
- `name` - The name of the network profile.
- `profile` - Dict settings for networkprofile.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the network profile.

» avi_errorpagebody

This data source is used to to get avi_errorpagebody objects.

» Example Usage

```
data "avi_errorpagebody" "foo_errorpagebody" {
    uuid = "errorpagebody-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search ErrorPageBody by name.
- `uuid` - (Optional) Search ErrorPageBody by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `error_page_body` - Error page body sent to client when match.
- `format` - Format of an error page body html or json.
- `name` - Field introduced in 17.2.4.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Field introduced in 17.2.4.

» avi_errorpageprofile

This data source is used to to get avi_errorpageprofile objects.

» Example Usage

```
data "avi_errorpageprofile" "foo_errorpageprofile" {
    uuid = "errorpageprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search ErrorPageProfile by name.
- `uuid` - (Optional) Search ErrorPageProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `error_pages` - Defined error pages for http status codes.
- `name` - Field introduced in 17.2.4.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Field introduced in 17.2.4.

» avi_controllerproperties

This data source is used to to get avi_controllerproperties objects.

» Example Usage

```
data "avi_controllerproperties" "foo_controllerproperties" {
    uuid = "controllerproperties-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search ControllerProperties by name.
- `uuid` - (Optional) Search ControllerProperties by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `allow_ip_forwarding` - Field introduced in 17.1.1.
- `allow_unauthenticated_apis` - Allow unauthenticated access for special apis.
- `allow_unauthenticated_nodes` - Boolean flag to set allow_unauthenticated_nodes.
- `api_idle_timeout` - Allowed values are 0-1440.
- `api_perf_logging_threshold` - Threshold to log request timing in portal_performance.log and server-timing response header.
- `appviewx_compat_mode` - Export configuration in appviewx compatibility mode.
- `attach_ip_retry_interval` - Placeholder for description of property attach_ip_retry_interval of obj type controllerproperties field type integer type int.
- `attach_ip_retry_limit` - Placeholder for description of property attach_ip_retry_limit of obj type controllerproperties field type integer type int.
- `bm_use_ansible` - Use ansible for se creation in baremetal.
- `cleanup_expired_authtoken_timeout_period` - Period for auth token cleanup job.
- `cleanup_sessions_timeout_period` - Period for sessions cleanup job.
- `cloud_reconcile` - Enable/disable periodic reconcile for all the clouds.
- `cluster_ip_gratuitous_arp_period` - Period for cluster ip gratuitous arp job.
- `consistency_check_timeout_period` - Period for consistency check job.
- `crashed_se_reboot` - Placeholder for description of property crashed_se_reboot of obj type controllerproperties field type integer type int.
- `dead_se_detection_timer` - Placeholder for description of property dead_se_detection_timer of obj type controllerproperties field type integer type int.
- `dns_refresh_period` - Period for refresh pool and gslb dns job.
- `dummy` - Placeholder for description of property dummy of obj type controllerproperties field type integer type int.
- `enable_api_sharding` - This setting enables the controller leader to shard api requests to the followers (if any).
- `enable_memory_balancer` - Enable/disable memory balancer.
- `fatal_error_lease_time` - Placeholder for description of property fatal_error_lease_time of obj type controllerproperties field type integer type int.
- `max_dead_se_in_grp` - Placeholder for description of property max_dead_se_in_grp of obj type controllerproperties field type integer type int.
- `max_pcap_per_tenant` - Maximum number of pcap files stored per tenant.
- `max_seq_attach_ip_failures` - Maximum number of consecutive attach

ip failures that halts vs placement.

- `max_seq_vnic_failures` - Placeholder for description of property max_seq_vnic_failures of obj type controllerproperties field type integer type int.
- `persistence_key_rotate_period` - Period for rotate app persistence keys job.
- `portal_token` - Token used for uploading tech-support to portal.
- `process_locked_useraccounts_timeout_period` - Period for process locked user accounts job.
- `process_pki_profile_timeout_period` - Period for process pki profile job.
- `query_host_fail` - Placeholder for description of property query_host_fail of obj type controllerproperties field type integer type int.
- `safenet_hsm_version` - Version of the safenet package installed on the controller.
- `se_create_timeout` - Placeholder for description of property se_create_timeout of obj type controllerproperties field type integer type int.
- `se_failover_attempt_interval` - Interval between attempting failovers to an se.
- `se_from_marketplace` - This setting decides whether se is to be deployed from the cloud marketplace or to be created by the controller.
- `se_offline_del` - Placeholder for description of property se_offline_del of obj type controllerproperties field type integer type int.
- `se_vnic_cooldown` - Placeholder for description of property se_vnic_cooldown of obj type controllerproperties field type integer type int.
- `secure_channel_cleanup_timeout` - Period for secure channel cleanup job.
- `secure_channel_controller_token_timeout` - Placeholder for description of property secure_channel_controller_token_timeout of obj type controllerproperties field type integer type int.
- `secure_channel_se_token_timeout` - Placeholder for description of property secure_channel_se_token_timeout of obj type controllerproperties field type integer type int.
- `seupgrade_fabric_pool_size` - Pool size used for all fabric commands during se upgrade.
- `seupgrade_segroup_min_dead_timeout` - Time to wait before marking segroup upgrade as stuck.
- `ssl_certificate_expiry_warning_days` - Number of days for ssl certificate expiry warning.
- `unresponsive_se_reboot` - Placeholder for description of property unresponsive_se_reboot of obj type controllerproperties field type integer type int.
- `upgrade_dns_ttl` - Time to account for dns ttl during upgrade.
- `upgrade_lease_time` - Placeholder for description of property upgrade_lease_time of obj type controllerproperties field type integer type int.

- `uuid` - Unique object identifier of the object.
- `vnic_op_fail_time` - Placeholder for description of property vnic_op_fail_time of obj type controllerproperties field type integer type int.
- `vs_apic_scaleout_timeout` - Time to wait for the scaled out se to become ready before marking the scaleout done, applies to apic configuration only.
- `vs_awaiting_se_timeout` - Placeholder for description of property vs_awaiting_se_timeout of obj type controllerproperties field type integer type int.
- `vs_key_rotate_period` - Period for rotate vs keys job.
- `vs_scaleout_ready_check_interval` - Interval for checking scaleout_ready status while controller is waiting for scaleoutready rpc from the service engine.
- `vs_se_attach_ip_fail` - Time to wait before marking attach ip operation on an se as failed.
- `vs_se_bootup_fail` - Placeholder for description of property vs_se_bootup_fail of obj type controllerproperties field type integer type int.
- `vs_se_create_fail` - Placeholder for description of property vs_se_create_fail of obj type controllerproperties field type integer type int.
- `vs_se_ping_fail` - Placeholder for description of property vs_se_ping_fail of obj type controllerproperties field type integer type int.
- `vs_se_vnic_fail` - Placeholder for description of property vs_se_vnic_fail of obj type controllerproperties field type integer type int.
- `vs_se_vnic_ip_fail` - Placeholder for description of property vs_se_vnic_ip_fail of obj type controllerproperties field type integer type int.
- `warmstart_se_reconnect_wait_time` - Placeholder for description of property warmstart_se_reconnect_wait_time of obj type controllerproperties field type integer type int.
- `warmstart_vs_resync_wait_time` - Timeout for warmstart vs resync.

» avi_healthmonitor

This data source is used to to get avi_healthmonitor objects.

» Example Usage

```
data "avi_healthmonitor" "foo_healthmonitor" {
    uuid = "healthmonitor-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search HealthMonitor by name.
- `uuid` - (Optional) Search HealthMonitor by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - User defined description for the object.
- `dns_monitor` - Dict settings for healthmonitor.
- `external_monitor` - Dict settings for healthmonitor.
- `failed_checks` - Number of continuous failed health checks before the server is marked down.
- `http_monitor` - Dict settings for healthmonitor.
- `https_monitor` - Dict settings for healthmonitor.
- `is_federated` - This field describes the object's replication scope.
- `monitor_port` - Use this port instead of the port defined for the server in the pool.
- `name` - A user friendly name for this health monitor.
- `radius_monitor` - Health monitor for radius.
- `receive_timeout` - A valid response from the server is expected within the receive timeout window.
- `send_interval` - Frequency, in seconds, that monitors are sent to a server.
- `sip_monitor` - Health monitor for sip.
- `successful_checks` - Number of continuous successful health checks before server is marked up.
- `tcp_monitor` - Dict settings for healthmonitor.
- `tenant_ref` - It is a reference to an object of type tenant.
- `type` - Type of the health monitor.
- `udp_monitor` - Dict settings for healthmonitor.
- `uuid` - Uuid of the health monitor.

» avi_analyticsprofile

This data source is used to to get avi_analyticsprofile objects.

» Example Usage

```
data "avi_analyticsprofile" "foo_analyticsprofile" {
    uuid = "analyticsprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search AnalyticsProfile by name.
- `uuid` - (Optional) Search AnalyticsProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `apdex_response_threshold` - If a client receives an http response in less than the satisfactory latency threshold, the request is considered satisfied.
- `apdex_response_tolerated_factor` - Client tolerated response latency factor.
- `apdex_rtt_threshold` - Satisfactory client to avi round trip time(rtt).
- `apdex_rtt_tolerated_factor` - Tolerated client to avi round trip time(rtt) factor.
- `apdex_rum_threshold` - If a client is able to load a page in less than the satisfactory latency threshold, the pageload is considered satisfied.
- `apdex_rum_tolerated_factor` - Virtual service threshold factor for tolerated page load time (plt) as multiple of apdex_rum_threshold.
- `apdex_server_response_threshold` - A server http response is considered satisfied if latency is less than the satisfactory latency threshold.
- `apdex_server_response_tolerated_factor` - Server tolerated response latency factor.
- `apdex_server_rtt_threshold` - Satisfactory client to avi round trip time(rtt).
- `apdex_server_rtt_tolerated_factor` - Tolerated client to avi round trip time(rtt) factor.
- `client_log_config` - Configure which logs are sent to the avi controller from ses and how they are processed.
- `client_log_streaming_config` - Configure to stream logs to an external server.
- `conn_lossy_ooo_threshold` - A connection between client and avi is considered lossy when more than this percentage of out of order packets are received.
- `conn_lossy_timeo_rexmt_threshold` - A connection between client and avi is considered lossy when more than this percentage of packets are retransmitted due to timeout.
- `conn_lossy_total_rexmt_threshold` - A connection between client and avi is considered lossy when more than this percentage of packets are retransmitted.
- `conn_lossy_zero_win_size_event_threshold` - A client connection is considered lossy when percentage of times a packet could not be trasmitted due to tcp zero window is above this threshold.
- `conn_server_lossy_ooo_threshold` - A connection between avi and

server is considered lossy when more than this percentage of out of order packets are received.

- `conn_server_lossy_timeo_rexmt_threshold` - A connection between avi and server is considered lossy when more than this percentage of packets are retransmitted due to timeout.
- `conn_server_lossy_total_rexmt_threshold` - A connection between avi and server is considered lossy when more than this percentage of packets are retransmitted.
- `conn_server_lossy_zero_win_size_event_threshold` - A server connection is considered lossy when percentage of times a packet could not be trasmitted due to tcp zero window is above this threshold.
- `description` - User defined description for the object.
- `disable_ondemand_metrics` - Virtual service (vs) metrics are processed only when there is live data traffic on the vs.
- `disable_se_analytics` - Disable node (service engine) level analytics forvs metrics.
- `disable_server_analytics` - Disable analytics on backend servers.
- `disable_vs_analytics` - Disable virtualservice (frontend) analytics.
- `enable_advanced_analytics` - Enables advanced analytics features like anomaly detection.
- `exclude_client_close_before_request_as_error` - Exclude client closed connection before an http request could be completed from being classified as an error.
- `exclude_dns_policy_drop_as_significant` - Exclude dns policy drops from the list of errors.
- `exclude_gs_down_as_error` - Exclude queries to gslb services that are operationally down from the list of errors.
- `exclude_http_error_codes` - List of http status codes to be excluded from being classified as an error.
- `exclude_invalid_dns_domain_as_error` - Exclude dns queries to domains outside the domains configured in the dns application profile from the list of errors.
- `exclude_invalid_dns_query_as_error` - Exclude invalid dns queries from the list of errors.
- `exclude_no_dns_record_as_error` - Exclude queries to domains that did not have configured services/records from the list of errors.
- `exclude_no_valid_gs_member_as_error` - Exclude queries to gslb services that have no available members from the list of errors.
- `exclude_persistence_change_as_error` - Exclude persistence server changed while load balancing' from the list of errors.
- `exclude_server_dns_error_as_error` - Exclude server dns error response from the list of errors.
- `exclude_server_tcp_reset_as_error` - Exclude server tcp reset from errors.
- `exclude_sip_error_codes` - List of sip status codes to be excluded from being classified as an error.

- `exclude_syn_retransmit_as_error` - Exclude 'server unanswered syns' from the list of errors.
- `exclude_tcp_reset_as_error` - Exclude tcp resets by client from the list of potential errors.
- `exclude_unsupported_dns_query_as_error` - Exclude unsupported dns queries from the list of errors.
- `healthscore_max_server_limit` - Skips health score computation of pool servers when number of servers in a pool is more than this setting.
- `hs_event_throttle_window` - Time window (in secs) within which only unique health change events should occur.
- `hs_max_anomaly_penalty` - Maximum penalty that may be deducted from health score for anomalies.
- `hs_max_resources_penalty` - Maximum penalty that may be deducted from health score for high resource utilization.
- `hs_max_security_penalty` - Maximum penalty that may be deducted from health score based on security assessment.
- `hs_min_dos_rate` - Dos connection rate below which the dos security assessment will not kick in.
- `hs_performance_boost` - Adds free performance score credits to health score.
- `hs_pscore_traffic_threshold_l4_client` - Threshold number of connections in 5min, below which apdexr, apdexc, rum_apdex, and other network quality metrics are not computed.
- `hs_pscore_traffic_threshold_l4_server` - Threshold number of connections in 5min, below which apdexr, apdexc, rum_apdex, and other network quality metrics are not computed.
- `hs_security_certscore_expired` - Score assigned when the certificate has expired.
- `hs_security_certscore_gt30d` - Score assigned when the certificate expires in more than 30 days.
- `hs_security_certscore_le07d` - Score assigned when the certificate expires in less than or equal to 7 days.
- `hs_security_certscore_le30d` - Score assigned when the certificate expires in less than or equal to 30 days.
- `hs_security_chain_invalidity_penalty` - Penalty for allowing certificates with invalid chain.
- `hs_security_cipherscore_eq000b` - Score assigned when the minimum cipher strength is 0 bits.
- `hs_security_cipherscore_ge128b` - Score assigned when the minimum cipher strength is greater than equal to 128 bits.
- `hs_security_cipherscore_lt128b` - Score assigned when the minimum cipher strength is less than 128 bits.
- `hs_security_encalgo_score_none` - Score assigned when no algorithm is used for encryption.
- `hs_security_encalgo_score_rc4` - Score assigned when rc4 algorithm is used for encryption.

- `hs_security_hsts_penalty` - Penalty for not enabling hsts.
- `hs_security_nonpfs_penalty` - Penalty for allowing non-pfs handshakes.
- `hs_security_selfsignedcert_penalty` - Deprecated.
- `hs_security_ssl30_score` - Score assigned when supporting ssl3.0 encryption protocol.
- `hs_security_tls10_score` - Score assigned when supporting tls1.0 encryption protocol.
- `hs_security_tls11_score` - Score assigned when supporting tls1.1 encryption protocol.
- `hs_security_tls12_score` - Score assigned when supporting tls1.2 encryption protocol.
- `hs_security_weak_signature_algo_penalty` - Penalty for allowing weak signature algorithm(s).
- `name` - The name of the analytics profile.
- `ondemand_metrics_idle_timeout` - This flag sets the time duration of no live data traffic after which virtual service metrics processing is suspended.
- `ranges` - List of http status code ranges to be excluded from being classified as an error.
- `resp_code_block` - Block of http response codes to be excluded from being classified as an error.
- `sensitive_log_profile` - Rules applied to the http application log for filtering sensitive information.
- `sip_log_depth` - Maximum number of sip messages added in logs for a sip transaction.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the analytics profile.

» avi_cloud

This data source is used to to get avi_cloud objects.

» Example Usage

```
data "avi_cloud" "foo_cloud" {
    uuid = "cloud-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search Cloud by name.
- `uuid` - (Optional) Search Cloud by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `apic_configuration` - Dict settings for cloud.
- `apic_mode` - Boolean flag to set apic_mode.
- `autoscale_polling_interval` - Cloudconnector polling interval in seconds for external autoscale groups, minimum 60 seconds.
- `aws_configuration` - Dict settings for cloud.
- `azure_configuration` - Field introduced in 17.2.1.
- `cloudstack_configuration` - Dict settings for cloud.
- `custom_tags` - Custom tags for all avi created resources in the cloud infrastructure.
- `dhcp_enabled` - Select the ip address management scheme.
- `dns_provider_ref` - Dns profile for the cloud.
- `docker_configuration` - Dict settings for cloud.
- `east_west_dns_provider_ref` - Dns profile for east-west services.
- `east_west_ipam_provider_ref` - Ipam profile for east-west services.
- `enable_vip_static_routes` - Use static routes for vip side network resolution during virtualservice placement.
- `gcp_configuration` - Google cloud platform configuration.
- `ip6_autocfg_enabled` - Enable ipv6 auto configuration.
- `ipam_provider_ref` - Ipam profile for the cloud.
- `license_tier` - Specifies the default license tier which would be used by new se groups.
- `license_type` - If no license type is specified then default license enforcement for the cloud type is chosen.
- `linuxserver_configuration` - Dict settings for cloud.
- `mtu` - Mtu setting for the cloud.
- `name` - Name of the object.
- `nsx_configuration` - Configuration parameters for nsx manager.
- `obj_name_prefix` - Default prefix for all automatically created objects in this cloud.
- `openstack_configuration` - Dict settings for cloud.
- `oshiftk8s_configuration` - Dict settings for cloud.
- `prefer_static_routes` - Prefer static routes over interface routes during virtualservice placement.
- `proxy_configuration` - Dict settings for cloud.
- `rancher_configuration` - Dict settings for cloud.
- `se_group_template_ref` - The service engine group to use as template.
- `state_based_dns_registration` - Dns records for vips are added/deleted based on the operational state of the vips.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.
- `vca_configuration` - Dict settings for cloud.
- `vcenter_configuration` - Dict settings for cloud.

- `vtype` - Cloud type.

» avi_cloudconnectoruser

This data source is used to to get avi_cloudconnectoruser objects.

» Example Usage

```
data "avi_cloudconnectoruser" "foo_cloudconnectoruser" {
    uuid = "cloudconnectoruser-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search CloudConnectorUser by name.
- `uuid` - (Optional) Search CloudConnectorUser by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `azure_serviceprincipal` - Field introduced in 17.2.1.
- `azure_userpass` - Field introduced in 17.2.1.
- `gcp_credentials` - Credentials for google cloud platform.
- `name` - Name of the object.
- `oci_credentials` - Credentials for oracle cloud infrastructure.
- `password` - Placeholder for description of property password of obj type cloudconnectoruser field type string type str.
- `private_key` - Placeholder for description of property private_key of obj type cloudconnectoruser field type string type str.
- `public_key` - Placeholder for description of property public_key of obj type cloudconnectoruser field type string type str.
- `tenant_ref` - It is a reference to an object of type tenant.
- `tencent_credentials` - Credentials for tencent cloud.
- `uuid` - Unique object identifier of the object.

» avi_virtualservice

This data source is used to to get avi_virtualservice objects.

» Example Usage

```
data "avi_virtualservice" "foo_virtualservice" {
    uuid = "virtualservice-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
    cloud_ref = "/api/cloud/?tenant=admin&name=Default-Cloud"
  }
```

» Argument Reference

- `name` - (Optional) Search VirtualService by name.
- `uuid` - (Optional) Search VirtualService by uuid.
- `cloud_ref` - (Optional) Search VirtualService by cloud_ref.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `active_standby_se_tag` - This configuration only applies if the virtualservice is in legacy active standby ha mode and load distribution among active standby is enabled.
- `allow_invalid_client_cert` - Process request even if invalid client certificate is presented.
- `analytics_policy` - Determines analytics settings for the application.
- `analytics_profile_ref` - Specifies settings related to analytics.
- `apic_contract_graph` - The name of the contract/graph associated with the virtual service.
- `application_profile_ref` - Enable application layer specific features for the virtual service.
- `bulk_sync_kvcache` - (this is a beta feature).
- `client_auth` - Http authentication configuration for protected resources.
- `close_client_conn_on_config_update` - Close client connection on vs config update.
- `cloud_config_cksum` - Checksum of cloud configuration for vs.
- `cloud_ref` - It is a reference to an object of type cloud.
- `cloud_type` - Enum options - cloud_none, cloud_vcenter, cloud_openstack, cloud_aws, cloud_vca, cloud_apic, cloud_mesos, cloud_linuxserver, cloud_docker_ucp, cloud_rancher, cloud_oshift_k8s, cloud_azure, cloud_gcp.
- `connections_rate_limit` - Rate limit the incoming connections to this virtual service.
- `content_rewrite` - Profile used to match and rewrite strings in request and/or response body.
- `created_by` - Creator name.

- `delay_fairness` - Select the algorithm for qos fairness.
- `description` - User defined description for the object.
- `dns_info` - Service discovery specific data including fully qualified domain name, type and time-to-live of the dns record.
- `dns_policies` - Dns policies applied on the dns traffic of the virtual service.
- `east_west_placement` - Force placement on all se's in service group (mesos mode only).
- `enable_autogw` - Response traffic to clients will be sent back to the source mac address of the connection, rather than statically sent to a default gateway.
- `enable_rhi` - Enable route health injection using the bgp config in the vrf context.
- `enable_rhi_snat` - Enable route health injection for source nat'ted floating ip address using the bgp config in the vrf context.
- `enabled` - Enable or disable the virtual service.
- `error_page_profile_ref` - Error page profile to be used for this virtualservice.this profile is used to send the custom error page to the client generated by the proxy.
- `flow_dist` - Criteria for flow distribution among ses.
- `flow_label_type` - Criteria for flow labelling.
- `fqdn` - Dns resolvable, fully qualified domain name of the virtualservice.
- `host_name_xlate` - Translate the host name sent to the servers to this value.
- `http_policies` - Http policies applied on the data traffic of the virtual service.
- `ign_pool_net_reach` - Ignore pool servers network reachability constraints for virtual service placement.
- `l4_policies` - L4 policies applied to the data traffic of the virtual service.
- `limit_doser` - Limit potential dos attackers who exceed max_cps_per_client significantly to a fraction of max_cps_per_client for a while.
- `max_cps_per_client` - Maximum connections per second per client ip.
- `microservice_ref` - Microservice representing the virtual service.
- `min_pools_up` - Minimum number of up pools to mark vs up.
- `name` - Name for the virtual service.
- `network_profile_ref` - Determines network settings such as protocol, tcp or udp, and related options for the protocol.
- `network_security_policy_ref` - Network security policies for the virtual service.
- `nsx_securitygroup` - A list of nsx service groups representing the clients which can access the virtual ip of the virtual service.
- `performance_limits` - Optional settings that determine performance limits like max connections or bandwdith etc.
- `pool_group_ref` - The pool group is an object that contains pools.
- `pool_ref` - The pool is an object that contains destination servers and related attributes such as load-balancing and persistence.

- `remove_listening_port_on_vs_down` - Remove listening port if virtualservice is down.
- `requests_rate_limit` - Rate limit the incoming requests to this virtual service.
- `saml_sp_config` - Application-specific saml config.
- `scaleout_ecmp` - Disable re-distribution of flows across service engines for a virtual service.
- `se_group_ref` - The service engine group to use for this virtual service.
- `security_policy_ref` - Security policy applied on the traffic of the virtual service.
- `server_network_profile_ref` - Determines the network settings profile for the server side of tcp proxied connections.
- `service_metadata` - Metadata pertaining to the service provided by this virtual service.
- `service_pool_select` - Select pool based on destination port.
- `services` - List of services defined for this virtual service.
- `sideband_profile` - Sideband configuration to be used for this virtualservice.it can be used for sending traffic to sideband vips for external inspection etc.
- `snat_ip` - Nat'ted floating source ip address(es) for upstream connection to servers.
- `ssl_key_and_certificate_refs` - Select or create one or two certificates, ec and/or rsa, that will be presented to ssl/tls terminated connections.
- `ssl_profile_ref` - Determines the set of ssl versions and ciphers to accept for ssl/tls terminated connections.
- `ssl_profile_selectors` - Select ssl profile based on client ip address match.
- `ssl_sess_cache_avg_size` - Expected number of ssl session cache entries (may be exceeded).
- `sso_policy_ref` - The sso policy attached to the virtualservice.
- `static_dns_records` - List of static dns records applied to this virtual service.
- `tenant_ref` - It is a reference to an object of type tenant.
- `topology_policies` - Topology policies applied on the dns traffic of the virtual service based ongslb topology algorithm.
- `traffic_clone_profile_ref` - Server network or list of servers for cloning traffic.
- `traffic_enabled` - Knob to enable the virtual service traffic on its assigned service engines.
- `type` - Specify if this is a normal virtual service, or if it is the parent or child of an sni-enabled virtual hosted virtual service.
- `use_bridge_ip_as_vip` - Use bridge ip as vip on each host in mesos deployments.
- `use_vip_as_snat` - Use the virtual ip as the snat ip for health monitoring and sending traffic to the backend servers instead of the service engine interface ip.

- `uuid` - Uuid of the virtualservice.
- `vh_domain_name` - The exact name requested from the client's sni-enabled tls hello domain name field.
- `vh_parent_vs_uuid` - Specifies the virtual service acting as virtual hosting (sni) parent.
- `vip` - List of virtual service ips.
- `vrf_context_ref` - Virtual routing context that the virtual service is bound to.
- `vs_datascripts` - Datascripts applied on the data traffic of the virtual service.
- `vsvip_cloud_config_cksum` - Checksum of cloud configuration for vsvip.
- `vsvip_ref` - Mostly used during the creation of shared vs, this field refers to entities that can be shared across virtual services.
- `waf_policy_ref` - Waf policy for the virtual service.
- `weight` - The quality of service weight to assign to traffic transmitted from this virtual service.

» avi_vsvip

This data source is used to to get avi_vsvip objects.

» Example Usage

```
data "avi_vsvip" "foo_vsvip" {
    uuid = "vsvip-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
    cloud_ref = "/api/cloud/?tenant=admin&name=Default-Cloud"
  }
```

» Argument Reference

- `name` - (Optional) Search VsVip by name.
- `uuid` - (Optional) Search VsVip by uuid.
- `cloud_ref` - (Optional) Search VsVip by cloud_ref.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `cloud_ref` - It is a reference to an object of type cloud.
- `dns_info` - Service discovery specific data including fully qualified domain name, type and time-to-live of the dns record.

- `east_west_placement` - Force placement on all service engines in the service engine group (container clouds only).
- `name` - Name for the vsvip object.
- `tenant_ref` - It is a reference to an object of type tenant.
- `use_standard_alb` - This overrides the cloud level default and needs to match the se group value in which it will be used if the se group use_standard_alb value is set.
- `uuid` - Uuid of the vsvip object.
- `vip` - List of virtual service ips and other shareable entities.
- `vrf_context_ref` - Virtual routing context that the virtual service is bound to.
- `vsvip_cloud_config_cksum` - Checksum of cloud configuration for vsvip.

» avi_alertsyslogconfig

This data source is used to to get avi_alertsyslogconfig objects.

» Example Usage

```
data "avi_alertsyslogconfig" "foo_alertsyslogconfig" {
    uuid = "alertsyslogconfig-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search AlertSyslogConfig by name.
- `uuid` - (Optional) Search AlertSyslogConfig by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - User defined description for alert syslog config.
- `name` - A user-friendly name of the syslog notification.
- `syslog_servers` - The list of syslog servers.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.

» avi_alertscriptconfig

This data source is used to to get avi_alertscriptconfig objects.

» Example Usage

```
data "avi_alertscriptconfig" "foo_alertscriptconfig" {
    uuid = "alertscriptconfig-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search AlertScriptConfig by name.
- `uuid` - (Optional) Search AlertScriptConfig by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `action_script` - User defined alert action script.
- `name` - A user-friendly name of the script.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.

» avi_alertconfig

This data source is used to to get avi_alertconfig objects.

» Example Usage

```
data "avi_alertconfig" "foo_alertconfig" {
    uuid = "alertconfig-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search AlertConfig by name.
- `uuid` - (Optional) Search AlertConfig by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `action_group_ref` - The alert config will trigger the selected alert action, which can send notifications and execute a controlscript.
- `alert_rule` - List of filters matching on events or client logs used for triggering alerts.
- `autoscale_alert` - This alert config applies to auto scale alerts.
- `category` - Determines whether an alert is raised immediately when event occurs (realtime) or after specified number of events occurs within rolling time window.
- `description` - A custom description field.
- `enabled` - Enable or disable this alert config from generating new alerts.
- `expiry_time` - An alert is expired and deleted after the expiry time has elapsed.
- `name` - Name of the alert configuration.
- `obj_uuid` - Uuid of the resource for which alert was raised.
- `object_type` - The object type to which the alert config is associated with.
- `recommendation` - Placeholder for description of property recommendation of obj type alertconfig field type string type str.
- `rolling_window` - Only if the number of events is reached or exceeded within the time window will an alert be generated.
- `source` - Signifies system events or the type of client logsused in this alert configuration.
- `summary` - Summary of reason why alert is generated.
- `tenant_ref` - It is a reference to an object of type tenant.
- `threshold` - An alert is created only when the number of events meets or exceeds this number within the chosen time frame.
- `throttle` - Alerts are suppressed (throttled) for this duration of time since the last alert was raised for this alert config.
- `uuid` - Unique object identifier of the object.

» avi_actiongroupconfig

This data source is used to to get avi_actiongroupconfig objects.

» Example Usage

```
data "avi_actiongroupconfig" "foo_actiongroupconfig" {
    uuid = "actiongroupconfig-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search ActionGroupConfig by name.
- `uuid` - (Optional) Search ActionGroupConfig by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `action_script_config_ref` - Reference of the action script configuration to be used.
- `autoscale_trigger_notification` - Trigger notification to autoscale manager.
- `description` - User defined description for the object.
- `email_config_ref` - Select the email notification configuration to use when sending alerts via email.
- `external_only` - Generate alert only to external destinations.
- `level` - When an alert is generated, mark its priority via the alert level.
- `name` - Name of the object.
- `snmp_trap_profile_ref` - Select the snmp trap notification to use when sending alerts via snmp trap.
- `syslog_config_ref` - Select the syslog notification configuration to use when sending alerts via syslog.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.

» avi_alertemailconfig

This data source is used to to get avi_alertemailconfig objects.

» Example Usage

```
data "avi_alertemailconfig" "foo_alertemailconfig" {
    uuid = "alertemailconfig-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search AlertEmailConfig by name.
- `uuid` - (Optional) Search AlertEmailConfig by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `cc_emails` - Alerts are copied to the comma separated list of email recipients.
- `description` - User defined description for the object.
- `name` - A user-friendly name of the email notification service.
- `tenant_ref` - It is a reference to an object of type tenant.
- `to_emails` - Alerts are sent to the comma separated list of email recipients.
- `uuid` - Unique object identifier of the object.

» avi_vsdatascriptset

This data source is used to to get avi_vsdatascriptset objects.

» Example Usage

```
data "avi_vsdatascriptset" "foo_vsdatascriptset" {
    uuid = "vsdatascriptset-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search VSDataScriptSet by name.
- `uuid` - (Optional) Search VSDataScriptSet by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `created_by` - Creator name.
- `datascript` - Datascripts to execute.
- `description` - User defined description for the object.
- `ipgroup_refs` - Uuid of ip groups that could be referred by vsdatascriptset objects.
- `name` - Name for the virtual service datascript collection.
- `pool_group_refs` - Uuid of pool groups that could be referred by vsdatascriptset objects.
- `pool_refs` - Uuid of pools that could be referred by vsdatascriptset objects.

- `protocol_parser_refs` - List of protocol parsers that could be referred by vsdatascriptset objects.
- `string_group_refs` - Uuid of string groups that could be referred by vsdatascriptset objects.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the virtual service datascript collection.

» avi_customipamdnsprofile

This data source is used to to get avi_customipamdnsprofile objects.

» Example Usage

```
data "avi_customipamdnsprofile" "foo_customipamdnsprofile" {
    uuid = "customipamdnsprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search CustomIpamDnsProfile by name.
- `uuid` - (Optional) Search CustomIpamDnsProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `name` - Name of the custom ipam dns profile.
- `script_params` - Parameters that are always passed to the ipam/dns script.
- `script_uri` - Script uri of form controller //ipamdnsscripts/.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Field introduced in 17.1.1.

» avi_ipamdnsproviderprofile

This data source is used to to get avi_ipamdnsproviderprofile objects.

» Example Usage

```
data "avi_ipamdnsproviderprofile" "foo_ipamdnsproviderprofile" {
    uuid = "ipamdnsproviderprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search IpamDnsProviderProfile by name.
- `uuid` - (Optional) Search IpamDnsProviderProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `allocate_ip_in_vrf` - If this flag is set, only allocate ip from networks in the virtual service vrf.
- `aws_profile` - Provider details if type is aws.
- `azure_profile` - Provider details if type is microsoft azure.
- `custom_profile` - Provider details if type is custom.
- `gcp_profile` - Provider details if type is google cloud.
- `infoblox_profile` - Provider details if type is infoblox.
- `internal_profile` - Provider details if type is avi.
- `name` - Name for the ipam/dns provider profile.
- `oci_profile` - Provider details for oracle cloud.
- `openstack_profile` - Provider details if type is openstack.
- `proxy_configuration` - Field introduced in 17.1.1.
- `tenant_ref` - It is a reference to an object of type tenant.
- `tencent_profile` - Provider details for tencent cloud.
- `type` - Provider type for the ipam/dns provider profile.
- `uuid` - Uuid of the ipam/dns provider profile.

» avi_poolgroup

This data source is used to to get avi_poolgroup objects.

» Example Usage

```
data "avi_poolgroup" "foo_poolgroup" {
    uuid = "poolgroup-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
```

```
    cloud_ref = "/api/cloud/?tenant=admin&name=Default-Cloud"
  }
```

» Argument Reference

- `name` - (Optional) Search PoolGroup by name.
- `uuid` - (Optional) Search PoolGroup by uuid.
- `cloud_ref` - (Optional) Search PoolGroup by cloud_ref.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `cloud_config_cksum` - Checksum of cloud configuration for poolgroup.
- `cloud_ref` - It is a reference to an object of type cloud.
- `created_by` - Name of the user who created the object.
- `deployment_policy_ref` - When setup autoscale manager will automatically promote new pools into production when deployment goals are met.
- `description` - Description of pool group.
- `fail_action` - Enable an action - close connection, http redirect, or local http response - when a pool group failure happens.
- `implicit_priority_labels` - Whether an implicit set of priority labels is generated.
- `members` - List of pool group members object of type poolgroupmember.
- `min_servers` - The minimum number of servers to distribute traffic to.
- `name` - The name of the pool group.
- `priority_labels_ref` - Uuid of the priority labels.
- `service_metadata` - Metadata pertaining to the service provided by this poolgroup.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the pool group.

» avi_prioritylabels

This data source is used to to get avi_prioritylabels objects.

» Example Usage

```
data "avi_prioritylabels" "foo_prioritylabels" {
    uuid = "prioritylabels-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
    cloud_ref = "/api/cloud/?tenant=admin&name=Default-Cloud"
```

```
}
```

» Argument Reference

- `name` - (Optional) Search PriorityLabels by name.
- `uuid` - (Optional) Search PriorityLabels by uuid.
- `cloud_ref` - (Optional) Search PriorityLabels by cloud_ref.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `cloud_ref` - It is a reference to an object of type cloud.
- `description` - A description of the priority labels.
- `equivalent_labels` - Equivalent priority labels in descending order.
- `name` - The name of the priority labels.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the priority labels.

» avi_poolgroupdeploymentpolicy

This data source is used to to get avi_poolgroupdeploymentpolicy objects.

» Example Usage

```
data "avi_poolgroupdeploymentpolicy" "foo_poolgroupdeploymentpolicy" {
    uuid = "poolgroupdeploymentpolicy-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search PoolGroupDeploymentPolicy by name.
- `uuid` - (Optional) Search PoolGroupDeploymentPolicy by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `auto_disable_old_prod_pools` - It will automatically disable old production pools once there is a new production candidate.

- `description` - User defined description for the object.
- `evaluation_duration` - Duration of evaluation period for automatic deployment.
- `name` - The name of the pool group deployment policy.
- `rules` - List of list.
- `scheme` - Deployment scheme.
- `target_test_traffic_ratio` - Target traffic ratio before pool is made production.
- `tenant_ref` - It is a reference to an object of type tenant.
- `test_traffic_ratio_rampup` - Ratio of the traffic that is sent to the pool under test.
- `uuid` - Uuid of the pool group deployment policy.
- `webhook_ref` - Webhook configured with url that avi controller will pass back information about pool group, old and new pool information and current deployment rule results.

» avi_pool

This data source is used to to get avi_pool objects.

» Example Usage

```
data "Pool" "foo_Pool" {
    uuid = "Pool-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
    cloud_ref = "/api/cloud/?tenant=admin&name=Default-Cloud"
  }
```

» Argument Reference

- `name` - (Optional) Search Pool by name.
- `uuid` - (Optional) Search Pool by uuid.
- `cloud_ref` - (Optional) Search Pool by cloud_ref.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `apic_epg_name` - Synchronize cisco apic epg members with pool servers.
- `application_persistence_profile_ref` - Persistence will ensure the same user sticks to the same server for a desired duration of time.

- `autoscale_launch_config_ref` - If configured then avi will trigger orchestration of pool server creation and deletion.
- `autoscale_networks` - Network ids for the launch configuration.
- `autoscale_policy_ref` - Reference to server autoscale policy.
- `capacity_estimation` - Inline estimation of capacity of servers.
- `capacity_estimation_ttfb_thresh` - The maximum time-to-first-byte of a server.
- `cloud_config_cksum` - Checksum of cloud configuration for pool.
- `cloud_ref` - It is a reference to an object of type cloud.
- `connection_ramp_duration` - Duration for which new connections will be gradually ramped up to a server recently brought online.
- `created_by` - Creator name.
- `default_server_port` - Traffic sent to servers will use this destination server port unless overridden by the server's specific port attribute.
- `description` - A description of the pool.
- `domain_name` - Comma separated list of domain names which will be used to verify the common names or subject alternative names presented by server certificates.
- `east_west` - Inherited config from virtualservice.
- `enabled` - Enable or disable the pool.
- `external_autoscale_groups` - Names of external auto-scale groups for pool servers.
- `fail_action` - Enable an action - close connection, http redirect or local http response - when a pool failure happens.
- `fewest_tasks_feedback_delay` - Periodicity of feedback for fewest tasks server selection algorithm.
- `graceful_disable_timeout` - Used to gracefully disable a server.
- `health_monitor_refs` - Verify server health by applying one or more health monitors.
- `host_check_enabled` - Enable common name check for server certificate.
- `inline_health_monitor` - The passive monitor will monitor client to server connections and requests and adjust traffic load to servers based on successful responses.
- `ipaddrgroup_ref` - Use list of servers from ip address group.
- `lb_algorithm` - The load balancing algorithm will pick a server within the pool's list of available servers.
- `lb_algorithm_consistent_hash_hdr` - Http header name to be used for the hash key.
- `lb_algorithm_core_nonaffinity` - Degree of non-affinity for core afffinity based server selection.
- `lb_algorithm_hash` - Criteria used as a key for determining the hash between the client and server.
- `lookup_server_by_name` - Allow server lookup by name.
- `max_concurrent_connections_per_server` - The maximum number of concurrent connections allowed to each server within the pool.
- `max_conn_rate_per_server` - Rate limit connections to each server.

- `name` - The name of the pool.
- `networks` - (internal-use) networks designated as containing servers for this pool.
- `nsx_securitygroup` - A list of nsx service groups where the servers for the pool are created.
- `pki_profile_ref` - Avi will validate the ssl certificate present by a server against the selected pki profile.
- `placement_networks` - Manually select the networks and subnets used to provide reachability to the pool's servers.
- `request_queue_depth` - Minimum number of requests to be queued when pool is full.
- `request_queue_enabled` - Enable request queue when pool is full.
- `rewrite_host_header_to_server_name` - Rewrite incoming host header to server name of the server to which the request is proxied.
- `rewrite_host_header_to_sni` - If sni server name is specified, rewrite incoming host header to the sni server name.
- `server_count` - General description.
- `server_name` - Fully qualified dns hostname which will be used in the tls sni extension in server connections if sni is enabled.
- `server_reselect` - Server reselect configuration for http requests.
- `servers` - The pool directs load balanced traffic to this list of destination servers.
- `sni_enabled` - Enable tls sni for server connections.
- `ssl_key_and_certificate_ref` - Service engines will present a client ssl certificate to the server.
- `ssl_profile_ref` - When enabled, avi re-encrypts traffic to the backend servers.
- `tenant_ref` - It is a reference to an object of type tenant.
- `use_service_port` - Do not translate the client's destination port when sending the connection to the server.
- `uuid` - Uuid of the pool.
- `vrf_ref` - Virtual routing context that the pool is bound to.

» avi_network

This data source is used to to get avi_network objects.

» Example Usage

```
data "avi_network" "foo_network" {
    uuid = "network-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
    cloud_ref = "/api/cloud/?tenant=admin&name=Default-Cloud"
```

```
}
```

» Argument Reference

- `name` - (Optional) Search Network by name.
- `uuid` - (Optional) Search Network by uuid.
- `cloud_ref` - (Optional) Search Network by cloud_ref.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `cloud_ref` - It is a reference to an object of type cloud.
- `configured_subnets` - List of list.
- `dhcp_enabled` - Select the ip address management scheme for this network.
- `exclude_discovered_subnets` - When selected, excludes all discovered subnets in this network from consideration for virtual service placement.
- `ip6_autocfg_enabled` - Enable ipv6 auto configuration.
- `name` - Name of the object.
- `synced_from_se` - Boolean flag to set synced_from_se.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.
- `vcenter_dvs` - Boolean flag to set vcenter_dvs.
- `vrf_context_ref` - It is a reference to an object of type vrfcontext.

» avi_serverautoscalepolicy

This data source is used to to get avi_serverautoscalepolicy objects.

» Example Usage

```
data "avi_serverautoscalepolicy" "foo_serverautoscalepolicy" {
    uuid = "serverautoscalepolicy-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search ServerAutoScalePolicy by name.
- `uuid` - (Optional) Search ServerAutoScalePolicy by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - User defined description for the object.
- `intelligent_autoscale` - Use avi intelligent autoscale algorithm where autoscale is performed by comparing load on the pool against estimated capacity of all the servers.
- `intelligent_scalein_margin` - Maximum extra capacity as percentage of load used by the intelligent scheme.
- `intelligent_scaleout_margin` - Minimum extra capacity as percentage of load used by the intelligent scheme.
- `max_scalein_adjustment_step` - Maximum number of servers to scalein simultaneously.
- `max_scaleout_adjustment_step` - Maximum number of servers to scale-out simultaneously.
- `max_size` - Maximum number of servers after scaleout.
- `min_size` - No scale-in happens once number of operationally up servers reach min_servers.
- `name` - Name of the object.
- `scalein_alertconfig_refs` - Trigger scalein when alerts due to any of these alert configurations are raised.
- `scalein_cooldown` - Cooldown period during which no new scalein is triggered to allow previous scalein to successfully complete.
- `scaleout_alertconfig_refs` - Trigger scaleout when alerts due to any of these alert configurations are raised.
- `scaleout_cooldown` - Cooldown period during which no new scaleout is triggered to allow previous scaleout to successfully complete.
- `tenant_ref` - It is a reference to an object of type tenant.
- `use_predicted_load` - Use predicted load rather than current load.
- `uuid` - Unique object identifier of the object.

» avi_autoscalelaunchconfig

This data source is used to to get avi_autoscalelaunchconfig objects.

» Example Usage

```
data "avi_autoscalelaunchconfig" "foo_autoscalelaunchconfig" {
    uuid = "autoscalelaunchconfig-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search AutoScaleLaunchConfig by name.
- `uuid` - (Optional) Search AutoScaleLaunchConfig by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `description` - User defined description for the object.
- `image_id` - Unique id of the amazon machine image (ami) or openstack vm id.
- `mesos` - Dict settings for autoscalelaunchconfig.
- `name` - Name of the object.
- `openstack` - Dict settings for autoscalelaunchconfig.
- `tenant_ref` - It is a reference to an object of type tenant.
- `use_external_asg` - If set to true, serverautoscalepolicy will use the autoscaling group (external_autoscaling_groups) from pool to perform scale up and scale down.
- `uuid` - Unique object identifier of the object.

» avi_applicationprofile

This data source is used to to get avi_applicationprofile objects.

» Example Usage

```
data "avi_applicationprofile" "foo_applicationprofile" {
    uuid = "applicationprofile-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search ApplicationProfile by name.
- `uuid` - (Optional) Search ApplicationProfile by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `cloud_config_cksum` - Checksum of application profiles.

- `created_by` - Name of the application profile creator.
- `description` - User defined description for the object.
- `dns_service_profile` - Specifies various dns service related controls for virtual service.
- `dos_rl_profile` - Specifies various security related controls for virtual service.
- `http_profile` - Specifies the http application proxy profile parameters.
- `name` - The name of the application profile.
- `preserve_client_ip` - Specifies if client ip needs to be preserved for back-end connection.
- `preserve_client_port` - Specifies if we need to preserve client port while preserving client ip for backend connections.
- `sip_service_profile` - Specifies various sip service related controls for virtual service.
- `tcp_app_profile` - Specifies the tcp application proxy profile parameters.
- `tenant_ref` - It is a reference to an object of type tenant.
- `type` - Specifies which application layer proxy is enabled for the virtual service.
- `uuid` - Uuid of the application profile.

» avi_httppolicyset

This data source is used to to get avi_httppolicyset objects.

» Example Usage

```
data "avi_httppolicyset" "foo_httppolicyset" {
    uuid = "httppolicyset-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
}
```

» Argument Reference

- `name` - (Optional) Search HTTPPolicySet by name.
- `uuid` - (Optional) Search HTTPPolicySet by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `cloud_config_cksum` - Checksum of cloud configuration for pool.
- `created_by` - Creator name.

- `description` - User defined description for the object.
- `http_request_policy` - Http request policy for the virtual service.
- `http_response_policy` - Http response policy for the virtual service.
- `http_security_policy` - Http security policy for the virtual service.
- `is_internal_policy` - Boolean flag to set is_internal_policy.
- `name` - Name of the http policy set.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Uuid of the http policy set.

» avi_serviceengine

This data source is used to to get avi_serviceengine objects.

» Example Usage

```
data "avi_serviceengine" "foo_serviceengine" {
    uuid = "serviceengine-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    name = "foo"
    cloud_ref = "/api/cloud/?tenant=admin&name=Default-Cloud"
  }
```

» Argument Reference

- `name` - (Optional) Search ServiceEngine by name.
- `uuid` - (Optional) Search ServiceEngine by uuid.
- `cloud_ref` - (Optional) Search ServiceEngine by cloud_ref.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `availability_zone` - Placeholder for description of property availability_zone of obj type serviceengine field type string type str.
- `cloud_ref` - It is a reference to an object of type cloud.
- `container_mode` - Boolean flag to set container_mode.
- `container_type` - Enum options - container_type_bridge, container_type_host, container_type_host_dpdk.
- `controller_created` - Boolean flag to set controller_created.
- `controller_ip` - Placeholder for description of property controller_ip of obj type serviceengine field type string type str.
- `data_vnics` - List of list.
- `enable_state` - Inorder to disable se set this field appropriately.

- `flavor` - Placeholder for description of property flavor of obj type serviceengine field type string type str.
- `host_ref` - It is a reference to an object of type vimgrhostruntime.
- `hypervisor` - Enum options - default, vmware_esx, kvm, vmware_vsan, xen.
- `mgmt_vnic` - Dict settings for serviceengine.
- `name` - Name of the object.
- `resources` - Dict settings for serviceengine.
- `se_group_ref` - It is a reference to an object of type serviceenginegroup.
- `tenant_ref` - It is a reference to an object of type tenant.
- `uuid` - Unique object identifier of the object.

» avi_fileservice

This data source is used to to get fileservice objects.

» Example Usage

```
data "avi_fileservice" "foo_Fileservice" {
    uuid = "filename"
  }
```

» Argument Reference

- `uuid` - (Optional) Search fileservice object by uuid.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the fileservice.

» avi_server

This data source is used to to get avi_server objects.

» Example Usage

```
data "avi_server" "foo_Server" {
    uuid = "server-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    pool_ref = "/api/pool/pool-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    ip='10.0.0.3'
  }
```

» Argument Reference

- `pool_ref` - (Optional) Search Server by pool_ref.
- `uuid` - (Optional) Search Server by uuid.
- `ip` - (Optional) Search Server by ip.

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - (Optional) Search server object by uuid.
- `pool_ref` - The pool is an object that contains destination servers and related attributes such as load-balancing and persistence.
- `ip` - IP address of a destination servers.
- `port` - Port of a destination servers.
- `type` - Type of ip address (V4)
- `autoscaling_group_name` - Name of autoscaling group this server belongs to.
- `description` - A description of the server.
- `enabled` - Enable or disable the server.
- `external_orchestration_id` - UID of server in external orchestration systems.
- `external_uuid` - UUID identifying VM in OpenStack and other external compute.
- `hostname` - DNS resolvable name of the server. May be used in place of the IP address.
- `location` - Geographic location of the server.Currently only for internal usage.
- `nw_ref` - This field is used internally by Avi, not editable by the user. It is a reference to an object of type VIMgrNWRuntime.
- `prst_hdr_val` - Header value for custom header persistence.
- `rewrite_host_header` - Rewrite incoming Host Header to server name.
- `vm_ref` - This field is used internally by Avi, not editable by the user. It is a reference to an object of type VIMgrVMRuntime.

» avi_useraccountprofile

The UserAccountProfile resource allows the creation and management of Avi UserAccountProfile

» Example Usage

```
resource "avi_useraccountprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.
- `account_lock_timeout` - (Optional) Lock timeout period (in minutes).
- `credentials_timeout_threshold` - (Optional) The time period after which credentials expire.
- `max_concurrent_sessions` - (Optional) Maximum number of concurrent sessions allowed.
- `max_login_failure_count` - (Optional) Number of login attempts before lockout.
- `max_password_history_count` - (Optional) Maximum number of passwords to be maintained in the password history.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_role

The Role resource allows the creation and management of Avi Role

» Example Usage

```
resource "avi_role" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.
- `privileges` - (Optional) List of list.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_natpolicy

The NatPolicy resource allows the creation and management of Avi NatPolicy

» Example Usage

```
resource "avi_natpolicy" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `created_by` - (Optional) Creator name.
- `description` - (Optional) Field introduced in 18.2.3.
- `name` - (Optional) Name of the nat policy.
- `rules` - (Optional) Nat policy rules.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the nat policy.

» avi_ipaddrgroup

The IpAddrGroup resource allows the creation and management of Avi IpAddrGroup

» Example Usage

```
resource "avi_ipaddrgroup" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the ip address group.
- `addrs` - (Optional) Configure ip address(es).
- `apic_epg_name` - (Optional) Populate ip addresses from members of this cisco apic epg.
- `country_codes` - (Optional) Populate the ip address ranges from the geo database for this country.
- `description` - (Optional) User defined description for the object.
- `ip_ports` - (Optional) Configure (ip address, port) tuple(s).
- `marathon_app_name` - (Optional) Populate ip addresses from tasks of this marathon app.
- `marathon_service_port` - (Optional) Task port associated with marathon service port.
- `prefixes` - (Optional) Configure ip address prefix(es).
- `ranges` - (Optional) Configure ip address range(s).
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the ip address group.

» avi_microservicegroup

The MicroServiceGroup resource allows the creation and management of Avi MicroServiceGroup

» Example Usage

```
resource "avi_microservicegroup" "foo" {
    name = "terraform-example-foo"
```

```
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the microservice group.
- `created_by` - (Optional) Creator name.
- `description` - (Optional) User defined description for the object.
- `service_refs` - (Optional) Configure microservice(es).
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the microservice group.

» avi_stringgroup

The StringGroup resource allows the creation and management of Avi StringGroup

» Example Usage

```
resource "avi_stringgroup" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the string group.
- `type` - (Required) Type of stringgroup.
- `description` - (Optional) User defined description for the object.
- `kv` - (Optional) Configure key value in the string group.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the string group.

» avi_trafficcloneprofile

The TrafficCloneProfile resource allows the creation and management of Avi TrafficCloneProfile

» Example Usage

```
resource "avi_trafficcloneprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name for the traffic clone profile.
- `clone_servers` - (Optional) Field introduced in 17.1.1.
- `cloud_ref` - (Optional) It is a reference to an object of type cloud.

- `preserve_client_ip` - (Optional) Specifies if client ip needs to be preserved to clone destination.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the traffic clone profile.

» avi_webhook

The Webhook resource allows the creation and management of Avi Webhook

» Example Usage

```
resource "avi_webhook" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the webhook profile.
- `callback_url` - (Optional) Callback url for the webhook.
- `description` - (Optional) Field introduced in 17.1.1.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `verification_token` - (Optional) Verification token sent back with the callback asquery parameters.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the webhook profile.

» avi_authprofile

The AuthProfile resource allows the creation and management of Avi AuthProfile

» Example Usage

```
resource "avi_authprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the auth profile.
- `type` - (Required) Type of the auth profile.
- `description` - (Optional) User defined description for the object.
- `http` - (Optional) Http user authentication params.
- `ldap` - (Optional) Ldap server and directory settings.
- `pa_agent_ref` - (Optional) Pingaccessagent uuid.
- `saml` - (Optional) Saml settings.
- `tacacs_plus` - (Optional) Tacacs+ settings.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the auth profile.

» avi_sslkeyandcertificate

The SSLKeyAndCertificate resource allows the creation and management of Avi SSLKeyAndCertificate

» Example Usage

```
resource "avi_sslkeyandcertificate" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `certificate` - (Required) Dict settings for sslkeyandcertificate.
- `name` - (Required) Name of the object.
- `ca_certs` - (Optional) Ca certificates in certificate chain.
- `certificate_base64` - (Optional) States if the certificate is base64 encoded.
- `certificate_management_profile_ref` - (Optional) It is a reference to an object of type certificatemanagementprofile.
- `created_by` - (Optional) Creator name.
- `dynamic_params` - (Optional) Dynamic parameters needed for certificate management profile.
- `enckey_base64` - (Optional) Encrypted private key corresponding to the private key (e.g.
- `enckey_name` - (Optional) Name of the encrypted private key (e.g.

- `format` - (Optional) Format of the key/certificate file.
- `hardwaresecuritymodulegroup_ref` - (Optional) It is a reference to an object of type hardwaresecuritymodulegroup.
- `key` - (Optional) Private key.
- `key_base64` - (Optional) States if the private key is base64 encoded.
- `key_params` - (Optional) Dict settings for sslkeyandcertificate.
- `key_passphrase` - (Optional) Passphrase used to encrypt the private key.
- `status` - (Optional) Enum options - ssl_certificate_finished, ssl_certificate_pending.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `type` - (Optional) Enum options - ssl_certificate_type_virtualservice, ssl_certificate_type_system, ssl_certificate_type_ca.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_sslprofile

The SSLProfile resource allows the creation and management of Avi SSLProfile

» Example Usage

```
resource "avi_sslprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.

- `accepted_ciphers` - (Optional) Ciphers suites represented as defined by u(http //www.openssl.org/docs/apps/ciphers.html).
- `accepted_versions` - (Optional) Set of versions accepted by the server.
- `cipher_enums` - (Optional) Enum options - tls_ecdhe_ecdsa_with_aes_128_gcm_sha256, tls_ecdhe_ecdsa_with_aes_256_gcm_sha384, tls_ecdhe_rsa_with_aes_128_gcm_sha256, tls_ecdhe_rsa_with_aes_256_gcm_sha384, tls_ecdhe_ecdsa_with_aes_128_cbc_sha256, tls_ecdhe_ecdsa_with_aes_256_cbc_sha384, tls_ecdhe_rsa_with_aes_128_cbc_sha256, tls_ecdhe_rsa_with_aes_256_cbc_sha384, tls_rsa_with_aes_128_gcm_sha256, tls_rsa_with_aes_256_gcm_sha384, tls_rsa_with_aes_128_cbc_sha256, tls_rsa_with_aes_256_cbc_sha256, tls_ecdhe_ecdsa_with_aes_128_cbc_sha, tls_ecdhe_ecdsa_with_aes_256_cbc_sha, tls_ecdhe_rsa_with_aes_128_cbc_sha, tls_ecdhe_rsa_with_aes_256_cbc_sha, tls_rsa_with_aes_128_cbc_sha, tls_rsa_with_aes_256_cbc_sha, tls_rsa_with_3des_ede_cbc_sha, tls_rsa_with_rc4_128_sha.
- `description` - (Optional) User defined description for the object.
- `dhparam` - (Optional) Dh parameters used in ssl.
- `enable_ssl_session_reuse` - (Optional) Enable ssl session re-use.
- `prefer_client_cipher_ordering` - (Optional) Prefer the ssl cipher ordering presented by the client during the ssl handshake over the one specified in the ssl profile.
- `send_close_notify` - (Optional) Send 'close notify' alert message for a clean shutdown of the ssl connection.
- `ssl_rating` - (Optional) Dict settings for sslprofile.
- `ssl_session_timeout` - (Optional) The amount of time in seconds before an ssl session expires.
- `tags` - (Optional) List of list.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `type` - (Optional) Ssl profile type.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_pkiprofile

The PKIProfile resource allows the creation and management of Avi PKIProfile

» Example Usage

```
resource "avi_pkiprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the pki profile.
- `ca_certs` - (Optional) List of certificate authorities (root and intermediate) trusted that is used for certificate validation.
- `created_by` - (Optional) Creator name.
- `crl_check` - (Optional) When enabled, avi will verify via crl checks that certificates in the trust chain have not been revoked.
- `crls` - (Optional) Certificate revocation lists.
- `ignore_peer_chain` - (Optional) When enabled, avi will not trust intermediate and root certs presented by a client.
- `is_federated` - (Optional) This field describes the object's replication scope.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `validate_only_leaf_crl` - (Optional) When enabled, avi will only validate the revocation status of the leaf certificate using crl.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_certificatemanagementprofile

The CertificateManagementProfile resource allows the creation and management of Avi CertificateManagementProfile

» Example Usage

```
resource "avi_certificatemanagementprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the pki profile.
- `script_path` - (Required) Placeholder for description of property script_path of obj type certificatemanagementprofile field type string type str.
- `script_params` - (Optional) List of list.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_ssopolicy

The SSOPolicy resource allows the creation and management of Avi SSOPolicy

» Example Usage

```
resource "avi_ssopolicy" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `authentication_policy` - (Optional) Authentication policy settings.
- `authorization_policy` - (Optional) Authorization policy settings.
- `name` - (Optional) Name of the sso policy.
- `tenant_ref` - (Optional) Uuid of the tenant.
- `type` - (Optional) Sso policy type.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the sso policy.

» avi_l4policyset

The L4PolicySet resource allows the creation and management of Avi L4PolicySet

» Example Usage

```
resource "avi_l4policyset" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `created_by` - (Optional) Creator name.
- `description` - (Optional) Field introduced in 17.2.7.
- `is_internal_policy` - (Optional) Field introduced in 17.2.7.
- `l4_connection_policy` - (Optional) Policy to apply when a new transport connection is setup.
- `name` - (Optional) Name of the l4 policy set.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Id of the l4 policy set.

» avi_scheduler

The Scheduler resource allows the creation and management of Avi Scheduler

» Example Usage

```
resource "avi_scheduler" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of scheduler.

- `backup_config_ref` - (Optional) Backup configuration to be executed by this scheduler.
- `enabled` - (Optional) Boolean flag to set enabled.
- `end_date_time` - (Optional) Scheduler end date and time.
- `frequency` - (Optional) Frequency at which custom scheduler will run.
- `frequency_unit` - (Optional) Unit at which custom scheduler will run.
- `run_mode` - (Optional) Scheduler run mode.
- `run_script_ref` - (Optional) Control script to be executed by this scheduler.
- `scheduler_action` - (Optional) Define scheduler action.
- `start_date_time` - (Optional) Scheduler start date and time.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_backupconfiguration

The BackupConfiguration resource allows the creation and management of Avi BackupConfiguration

» Example Usage

```
resource "avi_backupconfiguration" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of backup configuration.
- `aws_access_key` - (Optional) Aws access key id.
- `aws_bucket_id` - (Optional) Aws bucket.
- `aws_secret_access` - (Optional) Aws secret access key.
- `backup_file_prefix` - (Optional) Prefix of the exported configuration file.
- `backup_passphrase` - (Optional) Passphrase of backup configuration.
- `maximum_backups_stored` - (Optional) Rotate the backup files based on this count.
- `remote_directory` - (Optional) Directory at remote destination with write permission for ssh user.
- `remote_hostname` - (Optional) Remote destination.
- `save_local` - (Optional) Local backup.
- `ssh_user_ref` - (Optional) Access credentials for remote destination.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `upload_to_remote_host` - (Optional) Remote backup.
- `upload_to_s3` - (Optional) Cloud backup.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_tenant

The Tenant resource allows the creation and management of Avi Tenant

» Example Usage

```
resource "avi_tenant" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.
- `config_settings` - (Optional) Dict settings for tenant.
- `created_by` - (Optional) Creator of this tenant.
- `description` - (Optional) User defined description for the object.
- `local` - (Optional) Boolean flag to set local.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_serviceenginegroup

The ServiceEngineGroup resource allows the creation and management of Avi ServiceEngineGroup

» Example Usage

```
resource "avi_serviceenginegroup" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.
- `accelerated_networking` - (Optional) Enable accelerated networking option for azure se.

- `active_standby` - (Optional) Service engines in active/standby mode for ha failover.
- `advertise_backend_networks` - (Optional) Advertise reach-ability of backend server networks via adc through bgp for default gateway feature.
- `aggressive_failure_detection` - (Optional) Enable aggressive failover configuration for ha.
- `algo` - (Optional) In compact placement, virtual services are placed on existing ses until max_vs_per_se limit is reached.
- `allow_burst` - (Optional) Allow ses to be created using burst license.
- `app_cache_percent` - (Optional) A percent value of total se memory reserved for applicationcaching.
- `app_learning_memory_percent` - (Optional) A percent value of total se memory reserved for application learning.
- `archive_shm_limit` - (Optional) Amount of se memory in gb until which shared memory is collected in core archive.
- `async_ssl` - (Optional) Ssl handshakes will be handled by dedicated ssl threads.requires se reboot.
- `async_ssl_threads` - (Optional) Number of async ssl threads per se_dp.requires se reboot.
- `auto_rebalance` - (Optional) If set, virtual services will be automatically migrated when load on an se is less than minimum or more than maximum thresholds.
- `auto_rebalance_capacity_per_se` - (Optional) Capacities of se for auto rebalance for each criteria.
- `auto_rebalance_criteria` - (Optional) Set of criteria for se auto rebalance.
- `auto_rebalance_interval` - (Optional) Frequency of rebalance, if 'auto rebalance' is enabled.
- `auto_redistribute_active_standby_load` - (Optional) Redistribution of virtual services from the takeover se to the replacement se can cause momentary traffic loss.
- `bgp_state_update_interval` - (Optional) Bgp peer state update interval.
- `buffer_se` - (Optional) Excess service engine capacity provisioned for ha failover.
- `cloud_ref` - (Optional) It is a reference to an object of type cloud.
- `config_debugs_on_all_cores` - (Optional) Enable config debugs on all cores of se.
- `connection_memory_percentage` - (Optional) Percentage of memory for connection state.
- `cpu_reserve` - (Optional) Boolean flag to set cpu_reserve.
- `cpu_socket_affinity` - (Optional) Allocate all the cpu cores for the service engine virtual machines on the same cpu socket.
- `custom_securitygroups_data` - (Optional) Custom security groups to be associated with data vnics for se instances in openstack and aws clouds.
- `custom_securitygroups_mgmt` - (Optional) Custom security groups to be associated with management vnic for se instances in openstack and aws

clouds.

- `custom_tag` - (Optional) Custom tag will be used to create the tags for se instance in aws.
- `data_network_id` - (Optional) Subnet used to spin up the data nic for service engines, used only for azure cloud.
- `datascript_timeout` - (Optional) Number of instructions before datascript times out.
- `dedicated_dispatcher_core` - (Optional) Dedicate the core that handles packet receive/transmit from the network to just the dispatching function.
- `description` - (Optional) User defined description for the object.
- `disable_avi_securitygroups` - (Optional) By default, avi creates and manages security groups along with custom sg provided by user.
- `disable_csum_offloads` - (Optional) Stop using tcp/udp and ip checksum offload features of nics.
- `disable_gro` - (Optional) Disable generic receive offload (gro) in dpdk poll-mode driver packet receive path.
- `disable_se_memory_check` - (Optional) If set, disable the config memory check done in service engine.
- `disable_tso` - (Optional) Disable tcp segmentation offload (tso) in dpdk poll-mode driver packet transmit path.
- `disk_per_se` - (Optional) Amount of disk space for each of the service engine virtual machines.
- `distribute_load_active_standby` - (Optional) Use both the active and standby service engines for virtual service placement in the legacy active standby ha mode.
- `distribute_queues` - (Optional) Distributes queue ownership among cores so multiple cores handle dispatcher duties.requires se reboot.
- `distribute_vnics` - (Optional) Distributes vnic ownership among cores so multiple cores handle dispatcher duties.requires se reboot.
- `enable_gratarp_permanent` - (Optional) Enable gratarp for vip_ip.
- `enable_hsm_priming` - (Optional) (this is a beta feature).
- `enable_multi_lb` - (Optional) Applicable only for azure cloud with basic sku lb.
- `enable_routing` - (Optional) Enable routing for this serviceenginegroup .
- `enable_vip_on_all_interfaces` - (Optional) Enable vip on all interfaces of se.
- `enable_vmac` - (Optional) Use virtual mac address for interfaces on which floating interface ips are placed.
- `ephemeral_portrange_end` - (Optional) End local ephemeral port number for outbound connections.
- `ephemeral_portrange_start` - (Optional) Start local ephemeral port number for outbound connections.
- `extra_config_multiplier` - (Optional) Multiplier for extra config to support large vs/pool config.
- `extra_shared_config_memory` - (Optional) Extra config memory to sup-

port large geo db configuration.

- `floating_intf_ip` - (Optional) If serviceenginegroup is configured for legacy 1+1 active standby ha mode, floating ip's will be advertised only by the active se in the pair.
- `floating_intf_ip_se_2` - (Optional) If serviceenginegroup is configured for legacy 1+1 active standby ha mode, floating ip's will be advertised only by the active se in the pair.
- `flow_table_new_syn_max_entries` - (Optional) Maximum number of flow table entries that have not completed tcp three-way handshake yet.
- `free_list_size` - (Optional) Number of entries in the free list.
- `gratarp_permanent_periodicity` - (Optional) Gratarp periodicity for vip-ip.
- `ha_mode` - (Optional) High availability mode for all the virtual services using this service engine group.
- `hardwaresecuritymodulegroup_ref` - (Optional) It is a reference to an object of type hardwaresecuritymodulegroup.
- `heap_minimum_config_memory` - (Optional) Minimum required heap memory to apply any configuration.
- `hm_on_standby` - (Optional) Enable active health monitoring from the standby se for all placed virtual services.
- `host_attribute_key` - (Optional) Key of a (key, value) pair identifying a label for a set of nodes usually in container clouds.
- `host_attribute_value` - (Optional) Value of a (key, value) pair identifying a label for a set of nodes usually in container clouds.
- `host_gateway_monitor` - (Optional) Enable the host gateway monitor when service engine is deployed as docker container.
- `hypervisor` - (Optional) Override default hypervisor.
- `ignore_rtt_threshold` - (Optional) Ignore rtt samples if it is above threshold.
- `ingress_access_data` - (Optional) Program se security group ingress rules to allow vip data access from remote cidr type.
- `ingress_access_mgmt` - (Optional) Program se security group ingress rules to allow ssh/icmp management access from remote cidr type.
- `instance_flavor` - (Optional) Instance/flavor name for se instance.
- `iptables` - (Optional) Iptable rules.
- `least_load_core_selection` - (Optional) Select core with least load for new flow.
- `license_tier` - (Optional) Specifies the license tier which would be used.
- `license_type` - (Optional) If no license type is specified then default license enforcement for the cloud type is chosen.
- `log_disksz` - (Optional) Maximum disk capacity (in mb) to be allocated to an se.
- `max_cpu_usage` - (Optional) When cpu usage on an se exceeds this threshold, virtual services hosted on this se may be rebalanced to other ses to reduce load.
- `max_memory_per_mempool` - (Optional) Max bytes that can be allocated

in a single mempool.

- `max_public_ips_per_lb` - (Optional) Applicable to azure platform only.
- `max_rules_per_lb` - (Optional) Applicable to azure platform only.
- `max_scaleout_per_vs` - (Optional) Maximum number of active service engines for the virtual service.
- `max_se` - (Optional) Maximum number of services engines in this group.
- `max_vs_per_se` - (Optional) Maximum number of virtual services that can be placed on a single service engine.
- `mem_reserve` - (Optional) Boolean flag to set mem_reserve.
- `memory_for_config_update` - (Optional) Indicates the percent of memory reserved for config updates.
- `memory_per_se` - (Optional) Amount of memory for each of the service engine virtual machines.
- `mgmt_network_ref` - (Optional) Management network to use for avi service engines.
- `mgmt_subnet` - (Optional) Management subnet to use for avi service engines.
- `min_cpu_usage` - (Optional) When cpu usage on an se falls below the minimum threshold, virtual services hosted on the se may be consolidated onto other underutilized ses.
- `min_scaleout_per_vs` - (Optional) Minimum number of active service engines for the virtual service.
- `min_se` - (Optional) Minimum number of services engines in this group (relevant for se autorebalance only).
- `minimum_connection_memory` - (Optional) Indicates the percent of memory reserved for connections.
- `n_log_streaming_threads` - (Optional) Number of threads to use for log streaming.
- `non_significant_log_throttle` - (Optional) This setting limits the number of non-significant logs generated per second per core on this se.
- `num_dispatcher_cores` - (Optional) Number of dispatcher cores (0,1,2,4,8 or 16).
- `num_flow_cores_sum_changes_to_ignore` - (Optional) Number of changes in num flow cores sum to ignore.
- `openstack_availability_zones` - (Optional) Field introduced in 17.1.1.
- `openstack_mgmt_network_name` - (Optional) Avi management network name.
- `openstack_mgmt_network_uuid` - (Optional) Management network uuid.
- `os_reserved_memory` - (Optional) Amount of extra memory to be reserved for use by the operating system on a service engine.
- `per_app` - (Optional) Per-app se mode is designed for deploying dedicated load balancers per app (vs).
- `placement_mode` - (Optional) If placement mode is 'auto', virtual services are automatically placed on service engines.
- `realtime_se_metrics` - (Optional) Enable or disable real time se metrics.
- `reboot_on_panic` - (Optional) Reboot the vm or host on kernel panic.

- `se_bandwidth_type` - (Optional) Select the se bandwidth for the bandwidth license.
- `se_deprovision_delay` - (Optional) Duration to preserve unused service engine virtual machines before deleting them.
- `se_dos_profile` - (Optional) Dict settings for serviceenginegroup.
- `se_dp_vnic_queue_stall_event_sleep` - (Optional) Time (in seconds) service engine waits for after generating a vnic transmit queue stall event before resetting thenic.
- `se_dp_vnic_queue_stall_threshold` - (Optional) Number of consecutive transmit failures to look for before generating a vnic transmit queue stall event.
- `se_dp_vnic_queue_stall_timeout` - (Optional) Time (in milliseconds) to wait for network/nic recovery on detecting a transmit queue stall after which service engine resets the nic.
- `se_dp_vnic_restart_on_queue_stall_count` - (Optional) Number of consecutive transmit queue stall events in se_dp_vnic_stall_se_restart_window to look for before restarting se.
- `se_dp_vnic_stall_se_restart_window` - (Optional) Window of time (in seconds) during which se_dp_vnic_restart_on_queue_stall_count number of consecutive stalls results in a se restart.
- `se_dpdk_pmd` - (Optional) Determines if dpdk pool mode driver should be used or not 0 automatically determine based on hypervisor/nic type 1 unconditionally use dpdk poll mode driver 2 don't use dpdk poll mode driver.requires se reboot.
- `se_flow_probe_retries` - (Optional) Flow probe retry count if no replies are received.requires se reboot.
- `se_flow_probe_retry_timer` - (Optional) Timeout in milliseconds for flow probe retries.requires se reboot.
- `se_ipc_udp_port` - (Optional) Udp port for se_dp ipc in docker bridge mode.
- `se_lro` - (Optional) Enable or disable large receive optimization for vnics.
- `se_name_prefix` - (Optional) Prefix to use for virtual machine name of service engines.
- `se_pcap_lookahead` - (Optional) Enables lookahead mode of packet receive in pcap mode.
- `se_pcap_pkt_count` - (Optional) Max number of packets the pcap interface can hold and if the value is 0 the optimum value will be chosen.
- `se_pcap_pkt_sz` - (Optional) Max size of each packet in the pcap interface.
- `se_pcap_reinit_frequency` - (Optional) Frequency in seconds at which periodically a pcap reinit check is triggered.
- `se_pcap_reinit_threshold` - (Optional) Threshold for input packet receive errors in pcap mode exceeding which a pcap reinit is triggered.
- `se_probe_port` - (Optional) Tcp port on se where echo service will be run.
- `se_remote_punt_udp_port` - (Optional) Udp port for punted packets in

docker bridge mode.
- `se_routing` - (Optional) Enable routing via service engine datapath.
- `se_sb_dedicated_core` - (Optional) Sideband traffic will be handled by a dedicated core.requires se reboot.
- `se_sb_threads` - (Optional) Number of sideband threads per se.requires se reboot.
- `se_thread_multiplier` - (Optional) Multiplier for se threads based on vcpu.
- `se_tracert_port_range` - (Optional) Traceroute port range.
- `se_tunnel_mode` - (Optional) Determines if dsr from secondary se is active or not 0 automatically determine based on hypervisor type.
- `se_tunnel_udp_port` - (Optional) Udp port for tunneled packets from secondary to primary se in docker bridge mode.requires se reboot.
- `se_tx_batch_size` - (Optional) Number of packets to batch for transmit to the nic.
- `se_udp_encap_ipc` - (Optional) Determines if se-se ipc messages are encapsulated in a udp header 0 automatically determine based on hypervisor type.
- `se_use_dpdk` - (Optional) Determines if dpdk library should be used or not 0 automatically determine based on hypervisor type 1 use dpdk if pcap is not enabled 2 don't use dpdk.
- `se_vs_hb_max_pkts_in_batch` - (Optional) Maximum number of aggregated vs heartbeat packets to send in a batch.
- `se_vs_hb_max_vs_in_pkt` - (Optional) Maximum number of virtualservices for which heartbeat messages are aggregated in one packet.
- `self_se_election` - (Optional) Enable ses to elect a primary amongst themselves in the absence of a connectivity to controller.
- `service_ip6_subnets` - (Optional) Ipv6 subnets assigned to the se group.
- `service_ip_subnets` - (Optional) Subnets assigned to the se group.
- `shm_minimum_config_memory` - (Optional) Minimum required shared memory to apply any configuration.
- `significant_log_throttle` - (Optional) This setting limits the number of significant logs generated per second per core on this se.
- `ssl_preprocess_sni_hostname` - (Optional) (beta) preprocess ssl client hello for sni hostname extension.if set to true, this will apply sni child's ssl protocol(s), if they are different from sni parent's allowed ssl protocol(s).
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `udf_log_throttle` - (Optional) This setting limits the number of udf logs generated per second per core on this se.
- `use_standard_alb` - (Optional) Use standard sku azure load balancer.
- `vcenter_clusters` - (Optional) Dict settings for serviceenginegroup.
- `vcenter_datastore_mode` - (Optional) Enum options - vcenter_datastore_any, vcenter_datastore_local, vcenter_datastore_shared.
- `vcenter_datastores` - (Optional) List of list.
- `vcenter_datastores_include` - (Optional) Boolean flag to set vcenter_datastores_include.

- `vcenter_folder` - (Optional) Folder to place all the service engine virtual machines in vcenter.
- `vcenter_hosts` - (Optional) Dict settings for serviceenginegroup.
- `vcpus_per_se` - (Optional) Number of vcpus for each of the service engine virtual machines.
- `vip_asg` - (Optional) When vip_asg is set, vip configuration will be managed by avi.user will be able to configure vip_asg or vips individually at the time of create.
- `vs_host_redundancy` - (Optional) Ensure primary and secondary service engines are deployed on different physical hosts.
- `vs_scalein_timeout` - (Optional) Time to wait for the scaled in se to drain existing flows before marking the scalein done.
- `vs_scalein_timeout_for_upgrade` - (Optional) During se upgrade, time to wait for the scaled-in se to drain existing flows before marking the scalein done.
- `vs_scaleout_timeout` - (Optional) Time to wait for the scaled out se to become ready before marking the scaleout done.
- `vs_se_scaleout_additional_wait_time` - (Optional) Wait time for sending scaleout ready notification after virtual service is marked up.
- `vs_se_scaleout_ready_timeout` - (Optional) Timeout in seconds for service engine to sendscaleout ready notification of a virtual service.
- `vs_switchover_timeout` - (Optional) During se upgrade in a legacy active/standby segroup, time to wait for the new primary se to accept flows before marking the switchover done.
- `vss_placement` - (Optional) Parameters to place virtual services on only a subset of the cores of an se.
- `vss_placement_enabled` - (Optional) If set, virtual services will be placed on only a subset of the cores of an se.
- `waf_mempool` - (Optional) Enable memory pool for waf.requires se reboot.
- `waf_mempool_size` - (Optional) Memory pool size used for waf.requires se reboot.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_networkservice

The NetworkService resource allows the creation and management of Avi NetworkService

» Example Usage

```
resource "avi_networkservice" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `cloud_ref` - (Optional) It is a reference to an object of type cloud.
- `name` - (Optional) Name of the networkservice.
- `routing_service` - (Optional) Routing information of the networkservice.
- `se_group_ref` - (Optional) Service engine group to which the service is applied.
- `service_type` - (Optional) Indicates the type of networkservice.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `vrf_ref` - (Optional) Vrf context to which the service is scoped.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the networkservice.

» avi_dnspolicy

The DnsPolicy resource allows the creation and management of Avi DnsPolicy

» Example Usage

```
resource "avi_dnspolicy" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `created_by` - (Optional) Creator name.
- `description` - (Optional) Field introduced in 17.1.1.
- `name` - (Optional) Name of the dns policy.
- `rule` - (Optional) Dns rules.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the dns policy.

» avi_hardwaresecuritymodulegroup

The HardwareSecurityModuleGroup resource allows the creation and management of Avi HardwareSecurityModuleGroup

» Example Usage

```
resource "avi_hardwaresecuritymodulegroup" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `hsm` - (Required) Hardware security module configuration.
- `name` - (Required) Name of the hsm group configuration object.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the hsm group configuration object.

» avi_vrfcontext

The VrfContext resource allows the creation and management of Avi VrfContext

» Example Usage

```
resource "avi_vrfcontext" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.
- `bgp_profile` - (Optional) Bgp local and peer info.
- `cloud_ref` - (Optional) It is a reference to an object of type cloud.
- `debugvrfcontext` - (Optional) Configure debug flags for vrf.
- `description` - (Optional) User defined description for the object.

- `gateway_mon` - (Optional) Configure ping based heartbeat check for gateway in service engines of vrf.
- `internal_gateway_monitor` - (Optional) Configure ping based heartbeat check for all default gateways in service engines of vrf.
- `static_routes` - (Optional) List of list.
- `system_default` - (Optional) Boolean flag to set system_default.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_securitypolicy

The SecurityPolicy resource allows the creation and management of Avi SecurityPolicy

» Example Usage

```
resource "avi_securitypolicy" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the security policy.
- `description` - (Optional) Security policy is used to specify various configuration information used to perform distributed denial of service (ddos) attacks detection and mitigation.

- `dns_attacks` - (Optional) Attacks utilizing the dns protocol operations.
- `dns_policy_index` - (Optional) Index of the dns policy to use for the mitigation rules applied to the dns attacks.
- `network_security_policy_index` - (Optional) Index of the network security policy to use for the mitigation rules applied to the attacks.
- `oper_mode` - (Optional) Mode of dealing with the attacks - perform detection only, or detect and mitigate the attacks.
- `tcp_attacks` - (Optional) Attacks utilizing the tcp protocol operations.
- `tenant_ref` - (Optional) Tenancy of the security policy.
- `udp_attacks` - (Optional) Attacks utilizing the udp protocol operations.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - The uuid of the security policy.

» avi_protocolparser

The ProtocolParser resource allows the creation and management of Avi ProtocolParser

» Example Usage

```
resource "avi_protocolparser" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `description` - (Optional) Description of the protocol parser.

- `name` - (Optional) Name of the protocol parser.
- `parser_code` - (Optional) Command script provided inline.
- `tenant_ref` - (Optional) Tenant uuid of the protocol parser.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the protocol parser.

» avi_cloudproperties

The CloudProperties resource allows the creation and management of Avi Cloud-Properties

» Example Usage

```
resource "avi_cloudproperties" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `cc_props` - (Optional) Cloudconnector properties.
- `cc_vtypes` - (Optional) Cloud types supported by cloudconnector.
- `hyp_props` - (Optional) Hypervisor properties.
- `info` - (Optional) Properties specific to a cloud type.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_applicationpersistenceprofile

The ApplicationPersistenceProfile resource allows the creation and management of Avi ApplicationPersistenceProfile

» Example Usage

```
resource "avi_applicationpersistenceprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) A user-friendly name for the persistence profile.
- `persistence_type` - (Required) Method used to persist clients to the same server for a duration of time or a session.
- `app_cookie_persistence_profile` - (Optional) Specifies the application cookie persistence profile parameters.
- `description` - (Optional) User defined description for the object.
- `hdr_persistence_profile` - (Optional) Specifies the custom http header persistence profile parameters.
- `http_cookie_persistence_profile` - (Optional) Specifies the http cookie persistence profile parameters.
- `ip_persistence_profile` - (Optional) Specifies the client ip persistence profile parameters.

- `is_federated` - (Optional) This field describes the object's replication scope.
- `server_hm_down_recovery` - (Optional) Specifies behavior when a persistent server has been marked down by a health monitor.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the persistence profile.

» avi_backup

The Backup resource allows the creation and management of Avi Backup

» Example Usage

```
resource "avi_backup" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `file_name` - (Required) The file name of backup.
- `backup_config_ref` - (Optional) Backupconfiguration information.
- `local_file_url` - (Optional) Url to download the backup file.
- `remote_file_url` - (Optional) Url to download the backup file.
- `scheduler_ref` - (Optional) Scheduler information.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `timestamp` - (Optional) Unix timestamp of when the backup file is created.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_networksecuritypolicy

The NetworkSecurityPolicy resource allows the creation and management of Avi NetworkSecurityPolicy

» Example Usage

```
resource "avi_networksecuritypolicy" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `cloud_config_cksum` - (Optional) Checksum of cloud configuration for network sec policy.
- `created_by` - (Optional) Creator name.
- `description` - (Optional) User defined description for the object.
- `name` - (Optional) Name of the object.
- `rules` - (Optional) List of list.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI

- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_seproperties

The SeProperties resource allows the creation and management of Avi SeProperties

» Example Usage

```
resource "avi_seproperties" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `se_agent_properties` - (Optional) Dict settings for seproperties.
- `se_bootup_properties` - (Optional) Dict settings for seproperties.
- `se_runtime_properties` - (Optional) Dict settings for seproperties.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_pingaccessagent

The PingAccessAgent resource allows the creation and management of Avi PingAccessAgent

» Example Usage

```
resource "avi_pingaccessagent" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `description` - (Optional) Field introduced in 18.2.3.
- `name` - (Optional) Name of the pingaccess agent.
- `pingaccess_pool_ref` - (Optional) Pool containing a primary pingaccess server, as well as any failover servers included in the agent.properties file.
- `primary_server` - (Optional) The ip and port of the primary pingaccess server.
- `properties_file_data` - (Optional) Pingaccessagent's agent.properties file generated by pingaccess server.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the pingaccess agent.

» avi_gslbgeodbprofile

The GslbGeoDbProfile resource allows the creation and management of Avi GslbGeoDbProfile

» Example Usage

```
resource "avi_gslbgeodbprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) A user-friendly name for the geodb profile.
- `description` - (Optional) Field introduced in 17.1.1.
- `entries` - (Optional) List of geodb entries.
- `is_federated` - (Optional) This field indicates that this object is replicated across gslb federation.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the geodb profile.

» avi_gslbservice

The GslbService resource allows the creation and management of Avi GslbService

» Example Usage

```
resource "avi_gslbservice" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name for the gslb service.
- `application_persistence_profile_ref` - (Optional) The federated application persistence associated with gslbservice site persistence functionality.
- `controller_health_status_enabled` - (Optional) Gs member's overall health status is derived based on a combination of controller and datapath health-status inputs.
- `created_by` - (Optional) Creator name.
- `description` - (Optional) User defined description for the object.
- `domain_names` - (Optional) Fully qualified domain name of the gslb service.
- `down_response` - (Optional) Response to the client query when the gslb service is down.
- `enabled` - (Optional) Enable or disable the gslb service.
- `groups` - (Optional) Select list of pools belonging to this gslb service.
- `health_monitor_refs` - (Optional) Verify vs health by applying one or more health monitors.
- `health_monitor_scope` - (Optional) Health monitor probe can be executed for all the members or it can be executed only for third-party members.
- `hm_off` - (Optional) This field is an internal field and is used in se.
- `is_federated` - (Optional) This field indicates that this object is replicated across gslb federation.
- `min_members` - (Optional) The minimum number of members to distribute traffic to.
- `num_dns_ip` - (Optional) Number of ip addresses of this gslb service to be returned by the dns service.
- `pool_algorithm` - (Optional) The load balancing algorithm will pick a gslb pool within the gslb service list of available pools.
- `site_persistence_enabled` - (Optional) Enable site-persistence for the gslbservice.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `ttl` - (Optional) Ttl value (in seconds) for records served for this gslb service by the dns service.

- `use_edns_client_subnet` - (Optional) Use the client ip subnet from the edns option as source ipaddress for client geo-location and consistent hash algorithm.
- `wildcard_match` - (Optional) Enable wild-card match of fqdn if an exact match is not found in the dns table, the longest match is chosen by wild-carding the fqdn in the dns request.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the gslb service.

» avi_gslb

The Gslb resource allows the creation and management of Avi Gslb

» Example Usage

```
resource "avi_gslb" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name for the gslb object.
- `async_interval` - (Optional) Frequency with which messages are propagated to vs mgr.
- `clear_on_max_retries` - (Optional) Max retries after which the remote site is treated as a fresh start.

- `client_ip_addr_group` - (Optional) Group to specify if the client ip addresses are public or private.
- `description` - (Optional) User defined description for the object.
- `dns_configs` - (Optional) Sub domain configuration for the gslb.
- `error_resync_interval` - (Optional) Frequency with which errored messages are resynced to follower sites.
- `is_federated` - (Optional) This field indicates that this object is replicated across gslb federation.
- `leader_cluster_uuid` - (Optional) Mark this site as leader of gslb configuration.
- `maintenance_mode` - (Optional) This field disables the configuration operations on the leader for all federated objects.
- `send_interval` - (Optional) Frequency with which group members communicate.
- `send_interval_prior_to_maintenance_mode` - (Optional) The user can specify a send-interval while entering maintenance mode.
- `sites` - (Optional) Select avi site member belonging to this gslb.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `third_party_sites` - (Optional) Third party site member belonging to this gslb.
- `view_id` - (Optional) The view-id is used in change-leader mode to differentiate partitioned groups while they have the same gslb namespace.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the gslb object.

» avi_cluster

The Cluster resource allows the creation and management of Avi Cluster

» Example Usage

```
resource "avi_cluster" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.
- `nodes` - (Optional) List of list.
- `rejoin_nodes_automatically` - (Optional) Re-join cluster nodes automatically in the event one of the node is reset to factory.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `virtual_ip` - (Optional) A virtual ip address.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_clusterclouddetails

The ClusterCloudDetails resource allows the creation and management of Avi ClusterCloudDetails

» Example Usage

```
resource "avi_clusterclouddetails" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
```

```
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Field introduced in 17.2.5.
- `azure_info` - (Optional) Azure info to configure cluster_vip on the controller.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Field introduced in 17.2.5.

» avi_wafpolicy

The WafPolicy resource allows the creation and management of Avi WafPolicy

» Example Usage

```
resource "avi_wafpolicy" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Field introduced in 17.2.1.

- `allow_mode_delegation` - (Optional) Allow rules to overwrite the policy mode.
- `created_by` - (Optional) Creator name.
- `crs_groups` - (Optional) Waf rules are categorized in to groups based on their characterization.
- `description` - (Optional) Field introduced in 17.2.1.
- `enable_app_learning` - (Optional) Enable application learning for this waf policy.
- `failure_mode` - (Optional) Waf policy failure mode.
- `mode` - (Optional) Waf policy mode.
- `paranoia_level` - (Optional) Waf ruleset paranoia mode.
- `positive_security_model` - (Optional) The positive security model.
- `post_crs_groups` - (Optional) Waf rules are categorized in to groups based on their characterization.
- `pre_crs_groups` - (Optional) Waf rules are categorized in to groups based on their characterization.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `waf_crs_ref` - (Optional) Waf core ruleset used for the crs part of this policy.
- `waf_profile_ref` - (Optional) Waf profile for waf policy.
- `whitelist` - (Optional) A set of rules which describe conditions under which the request will bypass the waf.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Field introduced in 17.2.1.

» avi_wafcrs

The WafCRS resource allows the creation and management of Avi WafCRS

» Example Usage

```
resource "avi_wafcrs" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `description` - (Optional) A short description of this ruleset.
- `groups` - (Optional) Waf rules are sorted in groups based on their characterization.
- `integrity` - (Optional) Integrity protection value.
- `name` - (Optional) The name of this ruleset object.
- `release_date` - (Optional) The release date of this version in rfc 3339 / iso 8601 format.
- `tenant_ref` - (Optional) Tenant that this object belongs to.
- `version` - (Optional) The version of this ruleset object.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Field introduced in 18.1.1.

» avi_wafprofile

The WafProfile resource allows the creation and management of Avi WafProfile

» Example Usage

```
resource "avi_wafprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Field introduced in 17.2.1.
- `config` - (Optional) Config params for waf.
- `description` - (Optional) Field introduced in 17.2.1.
- `files` - (Optional) List of data files used for waf rules.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Field introduced in 17.2.1.

» avi_wafpolicypsmgroup

The WafPolicyPSMGroup resource allows the creation and management of Avi WafPolicyPSMGroup

» Example Usage

```
resource "avi_wafpolicypsmgroup" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `description` - (Optional) Freetext comment about this group.
- `enable` - (Optional) Enable or disable this waf rule group.
- `hit_action` - (Optional) If a rule in this group matches the match_value pattern, this action will be executed.
- `is_learning_group` - (Optional) This field indicates that this group is used for learning.
- `locations` - (Optional) Positive security model locations.
- `miss_action` - (Optional) If a rule in this group does not match the match_value pattern, this action will be executed.
- `name` - (Optional) User defined name of the group.
- `tenant_ref` - (Optional) Tenant that this object belongs to.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of this object.

» avi_snmptrapprofile

The SnmpTrapProfile resource allows the creation and management of Avi SnmpTrapProfile

» Example Usage

```
resource "avi_snmptrapprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) A user-friendly name of the snmp trap configuration.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `trap_servers` - (Optional) The ip address or hostname of the snmp trap destination server.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the snmp trap profile object.

» avi_systemconfiguration

The SystemConfiguration resource allows the creation and management of Avi SystemConfiguration

» Example Usage

```
resource "avi_systemconfiguration" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `admin_auth_configuration` - (Optional) Dict settings for systemconfiguration.
- `default_license_tier` - (Optional) Specifies the default license tier which would be used by new clouds.

- `dns_configuration` - (Optional) Dict settings for systemconfiguration.
- `dns_virtualservice_refs` - (Optional) Dns virtualservices hosting fqdn records for applications across avi vantage.
- `docker_mode` - (Optional) Boolean flag to set docker_mode.
- `email_configuration` - (Optional) Dict settings for systemconfiguration.
- `global_tenant_config` - (Optional) Dict settings for systemconfiguration.
- `linux_configuration` - (Optional) Dict settings for systemconfiguration.
- `mgmt_ip_access_control` - (Optional) Configure ip access control for controller to restrict open access.
- `ntp_configuration` - (Optional) Dict settings for systemconfiguration.
- `portal_configuration` - (Optional) Dict settings for systemconfiguration.
- `proxy_configuration` - (Optional) Dict settings for systemconfiguration.
- `secure_channel_configuration` - (Optional) Configure secure channel properties.
- `snmp_configuration` - (Optional) Dict settings for systemconfiguration.
- `ssh_ciphers` - (Optional) Allowed ciphers list for ssh to the management interface on the controller and service engines.
- `ssh_hmacs` - (Optional) Allowed hmac list for ssh to the management interface on the controller and service engines.
- `welcome_workflow_complete` - (Optional) This flag is set once the initial controller setup workflow is complete.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_controllersite

The ControllerSite resource allows the creation and management of Avi ControllerSite

» Example Usage

```
resource "avi_controllersite" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `address` - (Optional) Ip address or a dns resolvable, fully qualified domain name of the site controller cluster.
- `name` - (Optional) Name for the site controller cluster.
- `port` - (Optional) The controller site cluster's rest api port number.
- `tenant_ref` - (Optional) Reference for the tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Reference for the site controller cluster.

» avi_networkprofile

The NetworkProfile resource allows the creation and management of Avi NetworkProfile

» Example Usage

```
resource "avi_networkprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the network profile.
- `profile` - (Required) Dict settings for networkprofile.
- `connection_mirror` - (Optional) When enabled, avi mirrors all tcp fast-path connections to standby.
- `description` - (Optional) User defined description for the object.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the network profile.

» avi_errorpagebody

The ErrorPageBody resource allows the creation and management of Avi ErrorPageBody

» Example Usage

```
resource "avi_errorpagebody" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `error_page_body` - (Optional) Error page body sent to client when match.
- `format` - (Optional) Format of an error page body html or json.

- `name` - (Optional) Field introduced in 17.2.4.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Field introduced in 17.2.4.

» avi_errorpageprofile

The ErrorPageProfile resource allows the creation and management of Avi ErrorPageProfile

» Example Usage

```
resource "avi_errorpageprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `error_pages` - (Optional) Defined error pages for http status codes.
- `name` - (Optional) Field introduced in 17.2.4.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI

- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Field introduced in 17.2.4.

» avi_controllerproperties

The ControllerProperties resource allows the creation and management of Avi ControllerProperties

» Example Usage

```
resource "avi_controllerproperties" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `allow_ip_forwarding` - (Optional) Field introduced in 17.1.1.
- `allow_unauthenticated_apis` - (Optional) Allow unauthenticated access for special apis.
- `allow_unauthenticated_nodes` - (Optional) Boolean flag to set allow_unauthenticated_nodes.
- `api_idle_timeout` - (Optional) Allowed values are 0-1440.
- `api_perf_logging_threshold` - (Optional) Threshold to log request timing in portal_performance.log and server-timing response header.
- `appviewx_compat_mode` - (Optional) Export configuration in appviewx compatibility mode.
- `attach_ip_retry_interval` - (Optional) Placeholder for description of property attach_ip_retry_interval of obj type controllerproperties field type integer type int.
- `attach_ip_retry_limit` - (Optional) Placeholder for description of property attach_ip_retry_limit of obj type controllerproperties field type integer type int.
- `bm_use_ansible` - (Optional) Use ansible for se creation in baremetal.

- `cleanup_expired_authtoken_timeout_period` - (Optional) Period for auth token cleanup job.
- `cleanup_sessions_timeout_period` - (Optional) Period for sessions cleanup job.
- `cloud_reconcile` - (Optional) Enable/disable periodic reconcile for all the clouds.
- `cluster_ip_gratuitous_arp_period` - (Optional) Period for cluster ip gratuitous arp job.
- `consistency_check_timeout_period` - (Optional) Period for consistency check job.
- `crashed_se_reboot` - (Optional) Placeholder for description of property crashed_se_reboot of obj type controllerproperties field type integer type int.
- `dead_se_detection_timer` - (Optional) Placeholder for description of property dead_se_detection_timer of obj type controllerproperties field type integer type int.
- `dns_refresh_period` - (Optional) Period for refresh pool and gslb dns job.
- `dummy` - (Optional) Placeholder for description of property dummy of obj type controllerproperties field type integer type int.
- `enable_api_sharding` - (Optional) This setting enables the controller leader to shard api requests to the followers (if any).
- `enable_memory_balancer` - (Optional) Enable/disable memory balancer.
- `fatal_error_lease_time` - (Optional) Placeholder for description of property fatal_error_lease_time of obj type controllerproperties field type integer type int.
- `max_dead_se_in_grp` - (Optional) Placeholder for description of property max_dead_se_in_grp of obj type controllerproperties field type integer type int.
- `max_pcap_per_tenant` - (Optional) Maximum number of pcap files stored per tenant.
- `max_seq_attach_ip_failures` - (Optional) Maximum number of consecutive attach ip failures that halts vs placement.
- `max_seq_vnic_failures` - (Optional) Placeholder for description of property max_seq_vnic_failures of obj type controllerproperties field type integer type int.
- `persistence_key_rotate_period` - (Optional) Period for rotate app persistence keys job.
- `portal_token` - (Optional) Token used for uploading tech-support to portal.
- `process_locked_useraccounts_timeout_period` - (Optional) Period for process locked user accounts job.
- `process_pki_profile_timeout_period` - (Optional) Period for process pki profile job.
- `query_host_fail` - (Optional) Placeholder for description of property query_host_fail of obj type controllerproperties field type integer type

int.

- `safenet_hsm_version` - (Optional) Version of the safenet package installed on the controller.
- `se_create_timeout` - (Optional) Placeholder for description of property se_create_timeout of obj type controllerproperties field type integer type int.
- `se_failover_attempt_interval` - (Optional) Interval between attempting failovers to an se.
- `se_from_marketplace` - (Optional) This setting decides whether se is to be deployed from the cloud marketplace or to be created by the controller.
- `se_offline_del` - (Optional) Placeholder for description of property se_offline_del of obj type controllerproperties field type integer type int.
- `se_vnic_cooldown` - (Optional) Placeholder for description of property se_vnic_cooldown of obj type controllerproperties field type integer type int.
- `secure_channel_cleanup_timeout` - (Optional) Period for secure channel cleanup job.
- `secure_channel_controller_token_timeout` - (Optional) Placeholder for description of property secure_channel_controller_token_timeout of obj type controllerproperties field type integer type int.
- `secure_channel_se_token_timeout` - (Optional) Placeholder for description of property secure_channel_se_token_timeout of obj type controllerproperties field type integer type int.
- `seupgrade_fabric_pool_size` - (Optional) Pool size used for all fabric commands during se upgrade.
- `seupgrade_segroup_min_dead_timeout` - (Optional) Time to wait before marking segroup upgrade as stuck.
- `ssl_certificate_expiry_warning_days` - (Optional) Number of days for ssl certificate expiry warning.
- `unresponsive_se_reboot` - (Optional) Placeholder for description of property unresponsive_se_reboot of obj type controllerproperties field type integer type int.
- `upgrade_dns_ttl` - (Optional) Time to account for dns ttl during upgrade.
- `upgrade_lease_time` - (Optional) Placeholder for description of property upgrade_lease_time of obj type controllerproperties field type integer type int.
- `vnic_op_fail_time` - (Optional) Placeholder for description of property vnic_op_fail_time of obj type controllerproperties field type integer type int.
- `vs_apic_scaleout_timeout` - (Optional) Time to wait for the scaled out se to become ready before marking the scaleout done, applies to apic configuration only.
- `vs_awaiting_se_timeout` - (Optional) Placeholder for description of property vs_awaiting_se_timeout of obj type controllerproperties field type integer type int.
- `vs_key_rotate_period` - (Optional) Period for rotate vs keys job.

- `vs_scaleout_ready_check_interval` - (Optional) Interval for checking scaleout_ready status while controller is waiting for scaleoutready rpc from the service engine.
- `vs_se_attach_ip_fail` - (Optional) Time to wait before marking attach ip operation on an se as failed.
- `vs_se_bootup_fail` - (Optional) Placeholder for description of property vs_se_bootup_fail of obj type controllerproperties field type integer type int.
- `vs_se_create_fail` - (Optional) Placeholder for description of property vs_se_create_fail of obj type controllerproperties field type integer type int.
- `vs_se_ping_fail` - (Optional) Placeholder for description of property vs_se_ping_fail of obj type controllerproperties field type integer type int.
- `vs_se_vnic_fail` - (Optional) Placeholder for description of property vs_se_vnic_fail of obj type controllerproperties field type integer type int.
- `vs_se_vnic_ip_fail` - (Optional) Placeholder for description of property vs_se_vnic_ip_fail of obj type controllerproperties field type integer type int.
- `warmstart_se_reconnect_wait_time` - (Optional) Placeholder for description of property warmstart_se_reconnect_wait_time of obj type controllerproperties field type integer type int.
- `warmstart_vs_resync_wait_time` - (Optional) Timeout for warmstart vs resync.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_healthmonitor

The HealthMonitor resource allows the creation and management of Avi Health-Monitor

» Example Usage

```
resource "avi_healthmonitor" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) A user friendly name for this health monitor.
- `type` - (Required) Type of the health monitor.
- `description` - (Optional) User defined description for the object.
- `dns_monitor` - (Optional) Dict settings for healthmonitor.
- `external_monitor` - (Optional) Dict settings for healthmonitor.
- `failed_checks` - (Optional) Number of continuous failed health checks before the server is marked down.
- `http_monitor` - (Optional) Dict settings for healthmonitor.
- `https_monitor` - (Optional) Dict settings for healthmonitor.
- `is_federated` - (Optional) This field describes the object's replication scope.
- `monitor_port` - (Optional) Use this port instead of the port defined for the server in the pool.
- `radius_monitor` - (Optional) Health monitor for radius.
- `receive_timeout` - (Optional) A valid response from the server is expected within the receive timeout window.
- `send_interval` - (Optional) Frequency, in seconds, that monitors are sent to a server.
- `sip_monitor` - (Optional) Health monitor for sip.
- `successful_checks` - (Optional) Number of continuous successful health checks before server is marked up.
- `tcp_monitor` - (Optional) Dict settings for healthmonitor.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `udp_monitor` - (Optional) Dict settings for healthmonitor.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the health monitor.

» avi_analyticsprofile

The AnalyticsProfile resource allows the creation and management of Avi AnalyticsProfile

» Example Usage

```
resource "avi_analyticsprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the analytics profile.
- `apdex_response_threshold` - (Optional) If a client receives an http response in less than the satisfactory latency threshold, the request is considered satisfied.
- `apdex_response_tolerated_factor` - (Optional) Client tolerated response latency factor.
- `apdex_rtt_threshold` - (Optional) Satisfactory client to avi round trip time(rtt).
- `apdex_rtt_tolerated_factor` - (Optional) Tolerated client to avi round trip time(rtt) factor.
- `apdex_rum_threshold` - (Optional) If a client is able to load a page in less than the satisfactory latency threshold, the pageload is considered satisfied.
- `apdex_rum_tolerated_factor` - (Optional) Virtual service threshold factor for tolerated page load time (plt) as multiple of apdex_rum_threshold.
- `apdex_server_response_threshold` - (Optional) A server http response is considered satisfied if latency is less than the satisfactory latency threshold.
- `apdex_server_response_tolerated_factor` - (Optional) Server tolerated response latency factor.

- `apdex_server_rtt_threshold` - (Optional) Satisfactory client to avi round trip time(rtt).
- `apdex_server_rtt_tolerated_factor` - (Optional) Tolerated client to avi round trip time(rtt) factor.
- `client_log_config` - (Optional) Configure which logs are sent to the avi controller from ses and how they are processed.
- `client_log_streaming_config` - (Optional) Configure to stream logs to an external server.
- `conn_lossy_ooo_threshold` - (Optional) A connection between client and avi is considered lossy when more than this percentage of out of order packets are received.
- `conn_lossy_timeo_rexmt_threshold` - (Optional) A connection between client and avi is considered lossy when more than this percentage of packets are retransmitted due to timeout.
- `conn_lossy_total_rexmt_threshold` - (Optional) A connection between client and avi is considered lossy when more than this percentage of packets are retransmitted.
- `conn_lossy_zero_win_size_event_threshold` - (Optional) A client connection is considered lossy when percentage of times a packet could not be trasmitted due to tcp zero window is above this threshold.
- `conn_server_lossy_ooo_threshold` - (Optional) A connection between avi and server is considered lossy when more than this percentage of out of order packets are received.
- `conn_server_lossy_timeo_rexmt_threshold` - (Optional) A connection between avi and server is considered lossy when more than this percentage of packets are retransmitted due to timeout.
- `conn_server_lossy_total_rexmt_threshold` - (Optional) A connection between avi and server is considered lossy when more than this percentage of packets are retransmitted.
- `conn_server_lossy_zero_win_size_event_threshold` - (Optional) A server connection is considered lossy when percentage of times a packet could not be trasmitted due to tcp zero window is above this threshold.
- `description` - (Optional) User defined description for the object.
- `disable_ondemand_metrics` - (Optional) Virtual service (vs) metrics are processed only when there is live data traffic on the vs.
- `disable_se_analytics` - (Optional) Disable node (service engine) level analytics forvs metrics.
- `disable_server_analytics` - (Optional) Disable analytics on backend servers.
- `disable_vs_analytics` - (Optional) Disable virtualservice (frontend) analytics.
- `enable_advanced_analytics` - (Optional) Enables advanced analytics features like anomaly detection.
- `exclude_client_close_before_request_as_error` - (Optional) Exclude client closed connection before an http request could be completed from being classified as an error.

- `exclude_dns_policy_drop_as_significant` - (Optional) Exclude dns policy drops from the list of errors.
- `exclude_gs_down_as_error` - (Optional) Exclude queries to gslb services that are operationally down from the list of errors.
- `exclude_http_error_codes` - (Optional) List of http status codes to be excluded from being classified as an error.
- `exclude_invalid_dns_domain_as_error` - (Optional) Exclude dns queries to domains outside the domains configured in the dns application profile from the list of errors.
- `exclude_invalid_dns_query_as_error` - (Optional) Exclude invalid dns queries from the list of errors.
- `exclude_no_dns_record_as_error` - (Optional) Exclude queries to domains that did not have configured services/records from the list of errors.
- `exclude_no_valid_gs_member_as_error` - (Optional) Exclude queries to gslb services that have no available members from the list of errors.
- `exclude_persistence_change_as_error` - (Optional) Exclude persistence server changed while load balancing' from the list of errors.
- `exclude_server_dns_error_as_error` - (Optional) Exclude server dns error response from the list of errors.
- `exclude_server_tcp_reset_as_error` - (Optional) Exclude server tcp reset from errors.
- `exclude_sip_error_codes` - (Optional) List of sip status codes to be excluded from being classified as an error.
- `exclude_syn_retransmit_as_error` - (Optional) Exclude 'server unanswered syns' from the list of errors.
- `exclude_tcp_reset_as_error` - (Optional) Exclude tcp resets by client from the list of potential errors.
- `exclude_unsupported_dns_query_as_error` - (Optional) Exclude unsupported dns queries from the list of errors.
- `healthscore_max_server_limit` - (Optional) Skips health score computation of pool servers when number of servers in a pool is more than this setting.
- `hs_event_throttle_window` - (Optional) Time window (in secs) within which only unique health change events should occur.
- `hs_max_anomaly_penalty` - (Optional) Maximum penalty that may be deducted from health score for anomalies.
- `hs_max_resources_penalty` - (Optional) Maximum penalty that may be deducted from health score for high resource utilization.
- `hs_max_security_penalty` - (Optional) Maximum penalty that may be deducted from health score based on security assessment.
- `hs_min_dos_rate` - (Optional) Dos connection rate below which the dos security assessment will not kick in.
- `hs_performance_boost` - (Optional) Adds free performance score credits to health score.
- `hs_pscore_traffic_threshold_l4_client` - (Optional) Threshold number of connections in 5min, below which apdexr, apdexc, rum_apdex, and

other network quality metrics are not computed.
- `hs_pscore_traffic_threshold_l4_server` - (Optional) Threshold number of connections in 5min, below which apdexr, apdexc, rum_apdex, and other network quality metrics are not computed.
- `hs_security_certscore_expired` - (Optional) Score assigned when the certificate has expired.
- `hs_security_certscore_gt30d` - (Optional) Score assigned when the certificate expires in more than 30 days.
- `hs_security_certscore_le07d` - (Optional) Score assigned when the certificate expires in less than or equal to 7 days.
- `hs_security_certscore_le30d` - (Optional) Score assigned when the certificate expires in less than or equal to 30 days.
- `hs_security_chain_invalidity_penalty` - (Optional) Penalty for allowing certificates with invalid chain.
- `hs_security_cipherscore_eq000b` - (Optional) Score assigned when the minimum cipher strength is 0 bits.
- `hs_security_cipherscore_ge128b` - (Optional) Score assigned when the minimum cipher strength is greater than equal to 128 bits.
- `hs_security_cipherscore_lt128b` - (Optional) Score assigned when the minimum cipher strength is less than 128 bits.
- `hs_security_encalgo_score_none` - (Optional) Score assigned when no algorithm is used for encryption.
- `hs_security_encalgo_score_rc4` - (Optional) Score assigned when rc4 algorithm is used for encryption.
- `hs_security_hsts_penalty` - (Optional) Penalty for not enabling hsts.
- `hs_security_nonpfs_penalty` - (Optional) Penalty for allowing non-pfs handshakes.
- `hs_security_selfsignedcert_penalty` - (Optional) Deprecated.
- `hs_security_ssl30_score` - (Optional) Score assigned when supporting ssl3.0 encryption protocol.
- `hs_security_tls10_score` - (Optional) Score assigned when supporting tls1.0 encryption protocol.
- `hs_security_tls11_score` - (Optional) Score assigned when supporting tls1.1 encryption protocol.
- `hs_security_tls12_score` - (Optional) Score assigned when supporting tls1.2 encryption protocol.
- `hs_security_weak_signature_algo_penalty` - (Optional) Penalty for allowing weak signature algorithm(s).
- `ondemand_metrics_idle_timeout` - (Optional) This flag sets the time duration of no live data traffic after which virtual service metrics processing is suspended.
- `ranges` - (Optional) List of http status code ranges to be excluded from being classified as an error.
- `resp_code_block` - (Optional) Block of http response codes to be excluded from being classified as an error.
- `sensitive_log_profile` - (Optional) Rules applied to the http applica-

tion log for filtering sensitive information.
- `sip_log_depth` - (Optional) Maximum number of sip messages added in logs for a sip transaction.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the analytics profile.

» avi_cloud

The Cloud resource allows the creation and management of Avi Cloud

» Example Usage

```
resource "avi_cloud" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.
- `vtype` - (Required) Cloud type.
- `apic_configuration` - (Optional) Dict settings for cloud.
- `apic_mode` - (Optional) Boolean flag to set apic_mode.
- `autoscale_polling_interval` - (Optional) Cloudconnector polling interval in seconds for external autoscale groups, minimum 60 seconds.
- `aws_configuration` - (Optional) Dict settings for cloud.
- `azure_configuration` - (Optional) Field introduced in 17.2.1.

- `cloudstack_configuration` - (Optional) Dict settings for cloud.
- `custom_tags` - (Optional) Custom tags for all avi created resources in the cloud infrastructure.
- `dhcp_enabled` - (Optional) Select the ip address management scheme.
- `dns_provider_ref` - (Optional) Dns profile for the cloud.
- `docker_configuration` - (Optional) Dict settings for cloud.
- `east_west_dns_provider_ref` - (Optional) Dns profile for east-west services.
- `east_west_ipam_provider_ref` - (Optional) Ipam profile for east-west services.
- `enable_vip_static_routes` - (Optional) Use static routes for vip side network resolution during virtualservice placement.
- `gcp_configuration` - (Optional) Google cloud platform configuration.
- `ip6_autocfg_enabled` - (Optional) Enable ipv6 auto configuration.
- `ipam_provider_ref` - (Optional) Ipam profile for the cloud.
- `license_tier` - (Optional) Specifies the default license tier which would be used by new se groups.
- `license_type` - (Optional) If no license type is specified then default license enforcement for the cloud type is chosen.
- `linuxserver_configuration` - (Optional) Dict settings for cloud.
- `mtu` - (Optional) Mtu setting for the cloud.
- `nsx_configuration` - (Optional) Configuration parameters for nsx manager.
- `obj_name_prefix` - (Optional) Default prefix for all automatically created objects in this cloud.
- `openstack_configuration` - (Optional) Dict settings for cloud.
- `oshiftk8s_configuration` - (Optional) Dict settings for cloud.
- `prefer_static_routes` - (Optional) Prefer static routes over interface routes during virtualservice placement.
- `proxy_configuration` - (Optional) Dict settings for cloud.
- `rancher_configuration` - (Optional) Dict settings for cloud.
- `se_group_template_ref` - (Optional) The service engine group to use as template.
- `state_based_dns_registration` - (Optional) Dns records for vips are added/deleted based on the operational state of the vips.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `vca_configuration` - (Optional) Dict settings for cloud.
- `vcenter_configuration` - (Optional) Dict settings for cloud.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI

- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_cloudconnectoruser

The CloudConnectorUser resource allows the creation and management of Avi CloudConnectorUser

» Example Usage

```
resource "avi_cloudconnectoruser" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.
- `azure_serviceprincipal` - (Optional) Field introduced in 17.2.1.
- `azure_userpass` - (Optional) Field introduced in 17.2.1.
- `gcp_credentials` - (Optional) Credentials for google cloud platform.
- `oci_credentials` - (Optional) Credentials for oracle cloud infrastructure.
- `password` - (Optional) Placeholder for description of property password of obj type cloudconnectoruser field type string type str.
- `private_key` - (Optional) Placeholder for description of property private_key of obj type cloudconnectoruser field type string type str.
- `public_key` - (Optional) Placeholder for description of property public_key of obj type cloudconnectoruser field type string type str.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `tencent_credentials` - (Optional) Credentials for tencent cloud.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_virtualservice

The VirtualService resource allows the creation and management of Avi VirtualService

» Example Usage

```
resource "avi_virtualservice" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name for the virtual service.
- `active_standby_se_tag` - (Optional) This configuration only applies if the virtualservice is in legacy active standby ha mode and load distribution among active standby is enabled.
- `allow_invalid_client_cert` - (Optional) Process request even if invalid client certificate is presented.
- `analytics_policy` - (Optional) Determines analytics settings for the application.
- `analytics_profile_ref` - (Optional) Specifies settings related to analytics.
- `apic_contract_graph` - (Optional) The name of the contract/graph associated with the virtual service.
- `application_profile_ref` - (Optional) Enable application layer specific features for the virtual service.
- `bulk_sync_kvcache` - (Optional) (this is a beta feature).

- `client_auth` - (Optional) Http authentication configuration for protected resources.
- `close_client_conn_on_config_update` - (Optional) Close client connection on vs config update.
- `cloud_config_cksum` - (Optional) Checksum of cloud configuration for vs.
- `cloud_ref` - (Optional) It is a reference to an object of type cloud.
- `cloud_type` - (Optional) Enum options - cloud_none, cloud_vcenter, cloud_openstack, cloud_aws, cloud_vca, cloud_apic, cloud_mesos, cloud_linuxserver, cloud_docker_ucp, cloud_rancher, cloud_oshift_k8s, cloud_azure, cloud_gcp.
- `connections_rate_limit` - (Optional) Rate limit the incoming connections to this virtual service.
- `content_rewrite` - (Optional) Profile used to match and rewrite strings in request and/or response body.
- `created_by` - (Optional) Creator name.
- `delay_fairness` - (Optional) Select the algorithm for qos fairness.
- `description` - (Optional) User defined description for the object.
- `dns_info` - (Optional) Service discovery specific data including fully qualified domain name, type and time-to-live of the dns record.
- `dns_policies` - (Optional) Dns policies applied on the dns traffic of the virtual service.
- `east_west_placement` - (Optional) Force placement on all se's in service group (mesos mode only).
- `enable_autogw` - (Optional) Response traffic to clients will be sent back to the source mac address of the connection, rather than statically sent to a default gateway.
- `enable_rhi` - (Optional) Enable route health injection using the bgp config in the vrf context.
- `enable_rhi_snat` - (Optional) Enable route health injection for source nat'ted floating ip address using the bgp config in the vrf context.
- `enabled` - (Optional) Enable or disable the virtual service.
- `error_page_profile_ref` - (Optional) Error page profile to be used for this virtualservice.this profile is used to send the custom error page to the client generated by the proxy.
- `flow_dist` - (Optional) Criteria for flow distribution among ses.
- `flow_label_type` - (Optional) Criteria for flow labelling.
- `fqdn` - (Optional) Dns resolvable, fully qualified domain name of the virtualservice.
- `host_name_xlate` - (Optional) Translate the host name sent to the servers to this value.
- `http_policies` - (Optional) Http policies applied on the data traffic of the virtual service.
- `ign_pool_net_reach` - (Optional) Ignore pool servers network reachability constraints for virtual service placement.
- `l4_policies` - (Optional) L4 policies applied to the data traffic of the

virtual service.

- `limit_doser` - (Optional) Limit potential dos attackers who exceed max_cps_per_client significantly to a fraction of max_cps_per_client for a while.
- `max_cps_per_client` - (Optional) Maximum connections per second per client ip.
- `microservice_ref` - (Optional) Microservice representing the virtual service.
- `min_pools_up` - (Optional) Minimum number of up pools to mark vs up.
- `network_profile_ref` - (Optional) Determines network settings such as protocol, tcp or udp, and related options for the protocol.
- `network_security_policy_ref` - (Optional) Network security policies for the virtual service.
- `nsx_securitygroup` - (Optional) A list of nsx service groups representing the clients which can access the virtual ip of the virtual service.
- `performance_limits` - (Optional) Optional settings that determine performance limits like max connections or bandwdith etc.
- `pool_group_ref` - (Optional) The pool group is an object that contains pools.
- `pool_ref` - (Optional) The pool is an object that contains destination servers and related attributes such as load-balancing and persistence.
- `remove_listening_port_on_vs_down` - (Optional) Remove listening port if virtualservice is down.
- `requests_rate_limit` - (Optional) Rate limit the incoming requests to this virtual service.
- `saml_sp_config` - (Optional) Application-specific saml config.
- `scaleout_ecmp` - (Optional) Disable re-distribution of flows across service engines for a virtual service.
- `se_group_ref` - (Optional) The service engine group to use for this virtual service.
- `security_policy_ref` - (Optional) Security policy applied on the traffic of the virtual service.
- `server_network_profile_ref` - (Optional) Determines the network settings profile for the server side of tcp proxied connections.
- `service_metadata` - (Optional) Metadata pertaining to the service provided by this virtual service.
- `service_pool_select` - (Optional) Select pool based on destination port.
- `services` - (Optional) List of services defined for this virtual service.
- `sideband_profile` - (Optional) Sideband configuration to be used for this virtualservice.it can be used for sending traffic to sideband vips for external inspection etc.
- `snat_ip` - (Optional) Nat'ted floating source ip address(es) for upstream connection to servers.
- `ssl_key_and_certificate_refs` - (Optional) Select or create one or two certificates, ec and/or rsa, that will be presented to ssl/tls terminated connections.

- `ssl_profile_ref` - (Optional) Determines the set of ssl versions and ciphers to accept for ssl/tls terminated connections.
- `ssl_profile_selectors` - (Optional) Select ssl profile based on client ip address match.
- `ssl_sess_cache_avg_size` - (Optional) Expected number of ssl session cache entries (may be exceeded).
- `sso_policy_ref` - (Optional) The sso policy attached to the virtualservice.
- `static_dns_records` - (Optional) List of static dns records applied to this virtual service.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `topology_policies` - (Optional) Topology policies applied on the dns traffic of the virtual service based ongslb topology algorithm.
- `traffic_clone_profile_ref` - (Optional) Server network or list of servers for cloning traffic.
- `traffic_enabled` - (Optional) Knob to enable the virtual service traffic on its assigned service engines.
- `type` - (Optional) Specify if this is a normal virtual service, or if it is the parent or child of an sni-enabled virtual hosted virtual service.
- `use_bridge_ip_as_vip` - (Optional) Use bridge ip as vip on each host in mesos deployments.
- `use_vip_as_snat` - (Optional) Use the virtual ip as the snat ip for health monitoring and sending traffic to the backend servers instead of the service engine interface ip.
- `vh_domain_name` - (Optional) The exact name requested from the client's sni-enabled tls hello domain name field.
- `vh_parent_vs_uuid` - (Optional) Specifies the virtual service acting as virtual hosting (sni) parent.
- `vip` - (Optional) List of virtual service ips.
- `vrf_context_ref` - (Optional) Virtual routing context that the virtual service is bound to.
- `vs_datascripts` - (Optional) Datascripts applied on the data traffic of the virtual service.
- `vsvip_cloud_config_cksum` - (Optional) Checksum of cloud configuration for vsvip.
- `vsvip_ref` - (Optional) Mostly used during the creation of shared vs, this field refers to entities that can be shared across virtual services.
- `waf_policy_ref` - (Optional) Waf policy for the virtual service.
- `weight` - (Optional) The quality of service weight to assign to traffic transmitted from this virtual service.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the virtualservice.

» avi_vsvip

The VsVip resource allows the creation and management of Avi VsVip

» Example Usage

```
resource "avi_vsvip" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name for the vsvip object.
- `cloud_ref` - (Optional) It is a reference to an object of type cloud.
- `dns_info` - (Optional) Service discovery specific data including fully qualified domain name, type and time-to-live of the dns record.
- `east_west_placement` - (Optional) Force placement on all service engines in the service engine group (container clouds only).
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `use_standard_alb` - (Optional) This overrides the cloud level default and needs to match the se group value in which it will be used if the se group use_standard_alb value is set.
- `vip` - (Optional) List of virtual service ips and other shareable entities.
- `vrf_context_ref` - (Optional) Virtual routing context that the virtual service is bound to.
- `vsvip_cloud_config_cksum` - (Optional) Checksum of cloud configuration for vsvip.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the vsvip object.

» avi_alertsyslogconfig

The AlertSyslogConfig resource allows the creation and management of Avi AlertSyslogConfig

» Example Usage

```
resource "avi_alertsyslogconfig" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) A user-friendly name of the syslog notification.
- `description` - (Optional) User defined description for alert syslog config.
- `syslog_servers` - (Optional) The list of syslog servers.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_alertscriptconfig

The AlertScriptConfig resource allows the creation and management of Avi AlertScriptConfig

» Example Usage

```
resource "avi_alertscriptconfig" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) A user-friendly name of the script.
- `action_script` - (Optional) User defined alert action script.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_alertconfig

The AlertConfig resource allows the creation and management of Avi AlertConfig

» Example Usage

```
resource "avi_alertconfig" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `alert_rule` - (Required) List of filters matching on events or client logs used for triggering alerts.
- `category` - (Required) Determines whether an alert is raised immediately when event occurs (realtime) or after specified number of events occurs within rolling time window.
- `name` - (Required) Name of the alert configuration.
- `source` - (Required) Signifies system events or the type of client logsused in this alert configuration.
- `action_group_ref` - (Optional) The alert config will trigger the selected alert action, which can send notifications and execute a controlscript.
- `autoscale_alert` - (Optional) This alert config applies to auto scale alerts.
- `description` - (Optional) A custom description field.
- `enabled` - (Optional) Enable or disable this alert config from generating new alerts.
- `expiry_time` - (Optional) An alert is expired and deleted after the expiry time has elapsed.
- `obj_uuid` - (Optional) Uuid of the resource for which alert was raised.
- `object_type` - (Optional) The object type to which the alert config is associated with.
- `recommendation` - (Optional) Placeholder for description of property recommendation of obj type alertconfig field type string type str.
- `rolling_window` - (Optional) Only if the number of events is reached or exceeded within the time window will an alert be generated.
- `summary` - (Optional) Summary of reason why alert is generated.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `threshold` - (Optional) An alert is created only when the number of events meets or exceeds this number within the chosen time frame.

- `throttle` - (Optional) Alerts are suppressed (throttled) for this duration of time since the last alert was raised for this alert config.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_actiongroupconfig

The ActionGroupConfig resource allows the creation and management of Avi ActionGroupConfig

» Example Usage

```
resource "avi_actiongroupconfig" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `external_only` - (Required) Generate alert only to external destinations.
- `level` - (Required) When an alert is generated, mark its priority via the alert level.
- `name` - (Required) Name of the object.
- `action_script_config_ref` - (Optional) Reference of the action script configuration to be used.
- `autoscale_trigger_notification` - (Optional) Trigger notification to autoscale manager.
- `description` - (Optional) User defined description for the object.

- `email_config_ref` - (Optional) Select the email notification configuration to use when sending alerts via email.
- `snmp_trap_profile_ref` - (Optional) Select the snmp trap notification to use when sending alerts via snmp trap.
- `syslog_config_ref` - (Optional) Select the syslog notification configuration to use when sending alerts via syslog.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_alertemailconfig

The AlertEmailConfig resource allows the creation and management of Avi AlertEmailConfig

» Example Usage

```
resource "avi_alertemailconfig" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) A user-friendly name of the email notification service.
- `to_emails` - (Required) Alerts are sent to the comma separated list of email recipients.

- `cc_emails` - (Optional) Alerts are copied to the comma separated list of email recipients.
- `description` - (Optional) User defined description for the object.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_vsdatascriptset

The VSDataScriptSet resource allows the creation and management of Avi VSDataScriptSet

» Example Usage

```
resource "avi_vsdatascriptset" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name for the virtual service datascript collection.
- `created_by` - (Optional) Creator name.
- `datascript` - (Optional) Datascripts to execute.
- `description` - (Optional) User defined description for the object.
- `ipgroup_refs` - (Optional) Uuid of ip groups that could be referred by vsdatascriptset objects.

- `pool_group_refs` - (Optional) Uuid of pool groups that could be referred by vsdatascriptset objects.
- `pool_refs` - (Optional) Uuid of pools that could be referred by vsdatascriptset objects.
- `protocol_parser_refs` - (Optional) List of protocol parsers that could be referred by vsdatascriptset objects.
- `string_group_refs` - (Optional) Uuid of string groups that could be referred by vsdatascriptset objects.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the virtual service datascript collection.

» avi_customipamdnsprofile

The CustomIpamDnsProfile resource allows the creation and management of Avi CustomIpamDnsProfile

» Example Usage

```
resource "avi_customipamdnsprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Optional) Name of the custom ipam dns profile.

- `script_params` - (Optional) Parameters that are always passed to the ipam/dns script.
- `script_uri` - (Optional) Script uri of form controller //ipamdnsscripts/.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Field introduced in 17.1.1.

» avi_ipamdnsproviderprofile

The IpamDnsProviderProfile resource allows the creation and management of Avi IpamDnsProviderProfile

» Example Usage

```
resource "avi_ipamdnsproviderprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name for the ipam/dns provider profile.
- `type` - (Required) Provider type for the ipam/dns provider profile.
- `allocate_ip_in_vrf` - (Optional) If this flag is set, only allocate ip from networks in the virtual service vrf.
- `aws_profile` - (Optional) Provider details if type is aws.
- `azure_profile` - (Optional) Provider details if type is microsoft azure.
- `custom_profile` - (Optional) Provider details if type is custom.

- `gcp_profile` - (Optional) Provider details if type is google cloud.
- `infoblox_profile` - (Optional) Provider details if type is infoblox.
- `internal_profile` - (Optional) Provider details if type is avi.
- `oci_profile` - (Optional) Provider details for oracle cloud.
- `openstack_profile` - (Optional) Provider details if type is openstack.
- `proxy_configuration` - (Optional) Field introduced in 17.1.1.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `tencent_profile` - (Optional) Provider details for tencent cloud.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the ipam/dns provider profile.

» avi_poolgroup

The PoolGroup resource allows the creation and management of Avi PoolGroup

» Example Usage

```
resource "avi_poolgroup" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the pool group.
- `cloud_config_cksum` - (Optional) Checksum of cloud configuration for poolgroup.
- `cloud_ref` - (Optional) It is a reference to an object of type cloud.

- `created_by` - (Optional) Name of the user who created the object.
- `deployment_policy_ref` - (Optional) When setup autoscale manager will automatically promote new pools into production when deployment goals are met.
- `description` - (Optional) Description of pool group.
- `fail_action` - (Optional) Enable an action - close connection, http redirect, or local http response - when a pool group failure happens.
- `implicit_priority_labels` - (Optional) Whether an implicit set of priority labels is generated.
- `members` - (Optional) List of pool group members object of type poolgroupmember.
- `min_servers` - (Optional) The minimum number of servers to distribute traffic to.
- `priority_labels_ref` - (Optional) Uuid of the priority labels.
- `service_metadata` - (Optional) Metadata pertaining to the service provided by this poolgroup.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the pool group.

» avi_prioritylabels

The PriorityLabels resource allows the creation and management of Avi PriorityLabels

» Example Usage

```
resource "avi_prioritylabels" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the priority labels.
- `cloud_ref` - (Optional) It is a reference to an object of type cloud.
- `description` - (Optional) A description of the priority labels.
- `equivalent_labels` - (Optional) Equivalent priority labels in descending order.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the priority labels.

» avi_poolgroupdeploymentpolicy

The PoolGroupDeploymentPolicy resource allows the creation and management of Avi PoolGroupDeploymentPolicy

» Example Usage

```
resource "avi_poolgroupdeploymentpolicy" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the pool group deployment policy.

- `auto_disable_old_prod_pools` - (Optional) It will automatically disable old production pools once there is a new production candidate.
- `description` - (Optional) User defined description for the object.
- `evaluation_duration` - (Optional) Duration of evaluation period for automatic deployment.
- `rules` - (Optional) List of list.
- `scheme` - (Optional) Deployment scheme.
- `target_test_traffic_ratio` - (Optional) Target traffic ratio before pool is made production.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `test_traffic_ratio_rampup` - (Optional) Ratio of the traffic that is sent to the pool under test.
- `webhook_ref` - (Optional) Webhook configured with url that avi controller will pass back information about pool group, old and new pool information and current deployment rule results.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the pool group deployment policy.

» avi_pool

The Pool resource allows the creation and management of Avi Pool

» Example Usage

```
resource "Pool" "foo" {
    name = "terraform-example-foo"
    tenant = "admin"
}
```

» Argument Reference

The following arguments are supported:

```
* `apic_epg_name` - (Optional ) argument_description.
    * `application_persistence_profile_ref` - (Optional ) argument_description.
    * `autoscale_launch_config_ref` - (Optional ) argument_description.
    * `autoscale_networks` - (Optional ) argument_description.
    * `autoscale_policy_ref` - (Optional ) argument_description.
    * `capacity_estimation` - (Optional ) argument_description.
    * `capacity_estimation_ttfb_thresh` - (Optional ) argument_description.
    * `cloud_config_cksum` - (Optional ) argument_description.
    * `cloud_ref` - (Optional ) argument_description.
    * `connection_ramp_duration` - (Optional ) argument_description.
    * `created_by` - (Optional ) argument_description.
    * `default_server_port` - (Optional ) argument_description.
    * `description` - (Optional ) argument_description.
    * `domain_name` - (Optional ) argument_description.
    * `east_west` - (Optional ) argument_description.
    * `enabled` - (Optional ) argument_description.
    * `external_autoscale_groups` - (Optional ) argument_description.
    * `fail_action` - (Optional ) argument_description.
    * `fewest_tasks_feedback_delay` - (Optional ) argument_description.
    * `graceful_disable_timeout` - (Optional ) argument_description.
    * `health_monitor_refs` - (Optional ) argument_description.
    * `host_check_enabled` - (Optional ) argument_description.
    * `inline_health_monitor` - (Optional ) argument_description.
    * `ipaddrgroup_ref` - (Optional ) argument_description.
    * `lb_algorithm` - (Optional ) argument_description.
    * `lb_algorithm_consistent_hash_hdr` - (Optional ) argument_description.
    * `lb_algorithm_core_nonaffinity` - (Optional ) argument_description.
    * `lb_algorithm_hash` - (Optional ) argument_description.
    * `lookup_server_by_name` - (Optional ) argument_description.
    * `max_concurrent_connections_per_server` - (Optional ) argument_description.
    * `max_conn_rate_per_server` - (Optional ) argument_description.
    * `name` - (Required) argument_description.
    * `networks` - (Optional ) argument_description.
    * `nsx_securitygroup` - (Optional ) argument_description.
    * `pki_profile_ref` - (Optional ) argument_description.
    * `placement_networks` - (Optional ) argument_description.
    * `request_queue_depth` - (Optional ) argument_description.
    * `request_queue_enabled` - (Optional ) argument_description.
    * `rewrite_host_header_to_server_name` - (Optional ) argument_description.
    * `rewrite_host_header_to_sni` - (Optional ) argument_description.
    * `server_count` - (Optional ) argument_description.
    * `server_name` - (Optional ) argument_description.
```

```
    * `server_reselect` - (Optional ) argument_description.
    * `servers` - (Optional ) argument_description.
    * `sni_enabled` - (Optional ) argument_description.
    * `ssl_key_and_certificate_ref` - (Optional ) argument_description.
    * `ssl_profile_ref` - (Optional ) argument_description.
    * `tenant_ref` - (Optional ) argument_description.
    * `use_service_port` - (Optional ) argument_description.
        * `vrf_ref` - (Optional ) argument_description.
```

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

» avi_network

The Network resource allows the creation and management of Avi Network

» Example Usage

```
resource "avi_network" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.
- `cloud_ref` - (Optional) It is a reference to an object of type cloud.
- `configured_subnets` - (Optional) List of list.

- `dhcp_enabled` - (Optional) Select the ip address management scheme for this network.
- `exclude_discovered_subnets` - (Optional) When selected, excludes all discovered subnets in this network from consideration for virtual service placement.
- `ip6_autocfg_enabled` - (Optional) Enable ipv6 auto configuration.
- `synced_from_se` - (Optional) Boolean flag to set synced_from_se.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `vcenter_dvs` - (Optional) Boolean flag to set vcenter_dvs.
- `vrf_context_ref` - (Optional) It is a reference to an object of type vrf-context.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_serverautoscalepolicy

The ServerAutoScalePolicy resource allows the creation and management of Avi ServerAutoScalePolicy

» Example Usage

```
resource "avi_serverautoscalepolicy" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.
- `description` - (Optional) User defined description for the object.
- `intelligent_autoscale` - (Optional) Use avi intelligent autoscale algorithm where autoscale is performed by comparing load on the pool against estimated capacity of all the servers.
- `intelligent_scalein_margin` - (Optional) Maximum extra capacity as percentage of load used by the intelligent scheme.
- `intelligent_scaleout_margin` - (Optional) Minimum extra capacity as percentage of load used by the intelligent scheme.
- `max_scalein_adjustment_step` - (Optional) Maximum number of servers to scalein simultaneously.
- `max_scaleout_adjustment_step` - (Optional) Maximum number of servers to scaleout simultaneously.
- `max_size` - (Optional) Maximum number of servers after scaleout.
- `min_size` - (Optional) No scale-in happens once number of operationally up servers reach min_servers.
- `scalein_alertconfig_refs` - (Optional) Trigger scalein when alerts due to any of these alert configurations are raised.
- `scalein_cooldown` - (Optional) Cooldown period during which no new scalein is triggered to allow previous scalein to successfully complete.
- `scaleout_alertconfig_refs` - (Optional) Trigger scaleout when alerts due to any of these alert configurations are raised.
- `scaleout_cooldown` - (Optional) Cooldown period during which no new scaleout is triggered to allow previous scaleout to successfully complete.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `use_predicted_load` - (Optional) Use predicted load rather than current load.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_autoscalelaunchconfig

The AutoScaleLaunchConfig resource allows the creation and management of Avi AutoScaleLaunchConfig

» Example Usage

```
resource "avi_autoscalelaunchconfig" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the object.
- `description` - (Optional) User defined description for the object.
- `image_id` - (Optional) Unique id of the amazon machine image (ami) or openstack vm id.
- `mesos` - (Optional) Dict settings for autoscalelaunchconfig.
- `openstack` - (Optional) Dict settings for autoscalelaunchconfig.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.
- `use_external_asg` - (Optional) If set to true, serverautoscalepolicy will use the autoscaling group (external_autoscaling_groups) from pool to perform scale up and scale down.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_applicationprofile

The ApplicationProfile resource allows the creation and management of Avi ApplicationProfile

» Example Usage

```
resource "avi_applicationprofile" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the application profile.
- `type` - (Required) Specifies which application layer proxy is enabled for the virtual service.
- `cloud_config_cksum` - (Optional) Checksum of application profiles.
- `created_by` - (Optional) Name of the application profile creator.
- `description` - (Optional) User defined description for the object.
- `dns_service_profile` - (Optional) Specifies various dns service related controls for virtual service.
- `dos_rl_profile` - (Optional) Specifies various security related controls for virtual service.
- `http_profile` - (Optional) Specifies the http application proxy profile parameters.
- `preserve_client_ip` - (Optional) Specifies if client ip needs to be preserved for backend connection.
- `preserve_client_port` - (Optional) Specifies if we need to preserve client port while preserving client ip for backend connections.
- `sip_service_profile` - (Optional) Specifies various sip service related controls for virtual service.
- `tcp_app_profile` - (Optional) Specifies the tcp application proxy profile parameters.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI

- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the application profile.

» avi_httppolicyset

The HTTPPolicySet resource allows the creation and management of Avi HTTP-PolicySet

» Example Usage

```
resource "avi_httppolicyset" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Name of the http policy set.
- `cloud_config_cksum` - (Optional) Checksum of cloud configuration for pool.
- `created_by` - (Optional) Creator name.
- `description` - (Optional) User defined description for the object.
- `http_request_policy` - (Optional) Http request policy for the virtual service.
- `http_response_policy` - (Optional) Http response policy for the virtual service.
- `http_security_policy` - (Optional) Http security policy for the virtual service.
- `is_internal_policy` - (Optional) Boolean flag to set is_internal_policy.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Uuid of the http policy set.

» avi_serviceengine

The ServiceEngine resource allows the creation and management of Avi ServiceEngine

» Example Usage

```
resource "avi_serviceengine" "foo" {
    name = "terraform-example-foo"
    tenant_ref = "/api/tenant/?name=admin"
}
```

» Argument Reference

The following arguments are supported:

- `availability_zone` - (Optional) Placeholder for description of property availability_zone of obj type serviceengine field type string type str.
- `cloud_ref` - (Optional) It is a reference to an object of type cloud.
- `container_mode` - (Optional) Boolean flag to set container_mode.
- `container_type` - (Optional) Enum options - container_type_bridge, container_type_host, container_type_host_dpdk.
- `controller_created` - (Optional) Boolean flag to set controller_created.
- `controller_ip` - (Optional) Placeholder for description of property controller_ip of obj type serviceengine field type string type str.
- `data_vnics` - (Optional) List of list.
- `enable_state` - (Optional) Inorder to disable se set this field appropriately.
- `flavor` - (Optional) Placeholder for description of property flavor of obj type serviceengine field type string type str.
- `host_ref` - (Optional) It is a reference to an object of type vimgrhostruntime.

- `hypervisor` - (Optional) Enum options - default, vmware_esx, kvm, vmware_vsan, xen.
- `mgmt_vnic` - (Optional) Dict settings for serviceengine.
- `name` - (Optional) Name of the object.
- `resources` - (Optional) Dict settings for serviceengine.
- `se_group_ref` - (Optional) It is a reference to an object of type serviceenginegroup.
- `tenant_ref` - (Optional) It is a reference to an object of type tenant.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

- `uuid` - Unique object identifier of the object.

» avi_fileservice

The Fileservice resource allows the download and upload of files

» Example Usage

```
resource "avi_fileservice" "foo" {
    uri = "/uploads"
    local_file = "/file/path"
    upload = True
}
```

» Argument Reference

The following arguments are supported:

```
* `uri` - (Required) argument_description.
* `local_file` - (Required) argument_description.
* `upload` - (Optional ) argument_description.
```

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported:

```
* `uuid` - argument_description.
```

» avi_server

The Server resource allows the creation and management of Avi Server

» Example Usage

```
resource "avi_server" "foo" {
    pool_ref = "/api/pool/Pool-f9cf6b3e-a411-436f-95e2-2982ba2b217b"
    ip = "10.0.0.3"
}
```

» Argument Reference

The following arguments are supported:

```
* `pool_ref` - (Required) argument_description.
* `ip` - (Required) argument_description.
* `port` - (Optional ) argument_description.
* `type` - (Optional ) argument_description.
* `autoscaling_group_name` - (Optional ) argument_description.
* `description` - (Optional ) argument_description.
* `enabled` - (Optional ) argument_description.
* `external_orchestration_id` - (Optional ) argument_description.
* `external_uuid` - (Optional ) argument_description.
* `hostname` - (Optional ) argument_description.
* `location` - (Optional ) argument_description.
* `nw_ref` - (Optional ) argument_description.
* `prst_hdr_val` - (Optional ) argument_description.
```

* `rewrite_host_header` - (Optional) argument_description.
* `vm_ref` - (Optional) argument_description.

» Timeouts

The `timeouts` block allows you to specify timeouts for certain actions:

- `create` - (Defaults to 40 mins) Used when creating the AMI
- `update` - (Defaults to 40 mins) Used when updating the AMI
- `delete` - (Defaults to 90 mins) Used when deregistering the AMI

» Attributes Reference

In addition to all arguments above, the following attributes are exported: