» rancher2_app

Use this data source to retrieve information about a Rancher v2 app.

» Example Usage

```
data "rancher2_app" "rancher2" {
    name = "foo"
    project_id = "<project_id>"
    target_namespace = "<namespace_name>"
}
```

» Argument Reference

- `name` - (Required) The app name (string)
- `project_id` - (Required) The id of the project where the app is deployed (string)
- `target_namespace` - (Optional/Computed) The namespace name where the app is deployed (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `catalog_name` - (Computed) Catalog name of the app (string)
- `answers` - (Computed) Answers for the app (map)
- `description` - (Computed) Description for the app (string)
- `external_id` - (Computed) The URL of the helm catalog app (string)
- `revision_id` - (Computed) Current revision id for the app (string)
- `template_name` - (Computed) Template name of the app (string)
- `template_version` - (Computed) Template version of the app (string)
- `values_yaml` - (Computed) values.yaml base64 encoded file content for the app (string)
- `annotations` - (Computed) Annotations for the catalog (map)
- `labels` - (Computed) Labels for the catalog (map)

» rancher2_catalog

Use this data source to retrieve information about a Rancher v2 catalog.

» Example Usage

```
data "rancher2_catalog" "library" {
    name = "catalog"
}
```

» Argument Reference

- `name` - (Required) The catalog name.
- `scope` - (Optional) The scope of the catalog. `cluster`, `global`, and `project` are supported. Default `global` (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `branch` - (Computed) The branch of the catalog repo to use (string)
- `cluster_id` - (Computed) The cluster id of the catalog (string)
- `description` - (Computed) A catalog description (string)
- `kind` - (Computed) The kind of the catalog. Just helm by the moment (string)
- `password` - (Computed/Sensitive) The password to access the catalog if needed (string)
- `project_id` - (Computed) The project id of the catalog (string)
- `username` - (Computed/Sensitive) The username to access the catalog if needed (string)
- `version` - (Computed) Helm version for the catalog (string)
- `url` - (Computed) The url of the catalog repo (string)
- `annotations` - (Computed) Annotations for the catalog (map)
- `labels` - (Computed) Labels for the catalog (map)

» rancher2_certificate

Use this data source to retrieve information about a Rancher v2 certificate.

Depending of the availability, there are 2 types of Rancher v2 certificates: - Project certificate: Available to all namespaces in the `project_id` - Namespaced certificate: Available to just `namespace_id` in the `project_id`

» Example Usage

```
# Retrieve a rancher2 Project Certificate
data "rancher2_certificate" "foo" {
```

```
  name = "<name>"
  project_id = "<project_id>"
}

# Retrieve a rancher2 Namespaced Certificate
data "rancher2_certificate" "foo" {
  name = "<name>"
  project_id = "<project_id>"
  namespace_id = "<namespace_id>"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the certificate (string)
- `project_id` - (Required) The project id where to assign the certificate (string)
- `namespace_id` - (Optional) The namespace id where to assign the namespaced certificate (string)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `certs` - (Computed) Base64 encoded certs (string)
- `description` - (Computed) A certificate description (string)
- `annotations` - (Computed) Annotations for certificate object (map)
- `labels` - (Computed) Labels for certificate object (map)

» rancher2_cloud_credential

Use this data source to retrieve information about a Rancher v2 Cloud Credential.

» Example Usage

```
data "rancher2_cloud_credential" "test" {
    name = "test"
}
```

» Argument Reference

- `name` - (Required) The Cloud Credential name.

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `annotations` - (Computed) Annotations for the Cloud Credential (map)
- `labels` - (Computed) Labels for the Cloud Credential (map)

» rancher2_cluster

Use this data source to retrieve information about a Rancher v2 cluster.

» Example Usage

```
data "rancher2_cluster" "foo-custom" {
  name = "foo-custom"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the Cluster (string)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `cluster_registration_token` - (Computed) Cluster Registration Token generated for the cluster (list maxitems:1)
- `default_project_id` - (Computed) Default project ID for the cluster (string)
- `driver` - (Computed) The driver used for the Cluster. `imported`, `azurekubernetesservice`, `amazonelasticcontainerservice`, `googlekubernetesengine` and `rancherKubernetesEngine` are supported (string)
- `kube_config` - (Computed) Kube Config generated for the cluster (string)
- `system_project_id` - (Computed) System project ID for the cluster (string)

- `rke_config` - (Computed) The RKE configuration for `rke` Clusters. Conflicts with `aks_config`, `eks_config` and `gke_config` (list maxitems:1)
- `k3s_config` - (Computed) The K3S configuration for `k3s` imported Clusters. Conflicts with `aks_config`, `eks_config`, `gke_config` and `rke_config` (list maxitems:1)
- `aks_config` - (Computed) The Azure aks configuration for `aks` Clusters. Conflicts with `eks_config`, `gke_config` and `rke_config` (list maxitems:1)
- `eks_config` - (Computed) The Amazon eks configuration for `eks` Clusters. Conflicts with `aks_config`, `gke_config` and `rke_config` (list maxitems:1)
- `gke_config` - (Computed) The Google gke configuration for `gke` Clusters. Conflicts with `aks_config`, `eks_config` and `rke_config` (list maxitems:1)
- `description` - (Computed) The description for Cluster (string)
- `cluster_auth_endpoint` - (Computed) Enabling the local cluster authorized endpoint allows direct communication with the cluster, bypassing the Rancher API proxy. (list maxitems:1)
- `cluster_monitoring_input` - (Computed) Cluster monitoring config (list maxitems:1)
- `cluster_template_answers` - (Computed) Cluster template answers (list maxitems:1)
- `cluster_template_id` - (Computed) Cluster template ID (string)
- `cluster_template_questions` - (Computed) Cluster template questions (list)
- `cluster_template_revision_id` - (Computed) Cluster template revision ID (string)
- `default_pod_security_policy_template_id` - (Optional/Computed) Default pod security policy template id (string)
- `enable_cluster_monitoring` - (Computed) Enable built-in cluster monitoring. Default `false` (bool)
- `enable_network_policy` - (Computed) Enable project network isolation. Default `false` (bool)
- `enable_cluster_istio` - (Computed) Enable built-in cluster istio. Default `false` (bool)
- `annotations` - (Computed) Annotations for Node Pool object (map)
- `labels` - (Computed) Labels for Node Pool object (map)

» rancher2_cluster_alert_group

Use this data source to retrieve information about a Rancher v2 cluster alert group.

» Example Usage

```
data "rancher2_cluster_alert_group" "foo" {
  cluster_id = "<cluster_id>"
  name = "<cluster_alert_group_name>"
}
```

» Argument Reference

- `cluster_id` - (Required) The cluster id where create cluster alert group (string)
- `name` - (Required) The cluster alert group name (string)

» Attributes Reference

- `description` - (Computed) The cluster alert group description (string)
- `group_interval_seconds` - (Computed) The cluster alert group interval seconds. Default: `180` (int)
- `group_wait_seconds` - (Computed) The cluster alert group wait seconds. Default: `180` (int)
- `recipients` - (Computed) The cluster alert group recipients (list)
- `repeat_interval_seconds` - (Computed) The cluster alert group wait seconds. Default: `3600` (int)
- `annotations` - (Computed) The cluster alert group annotations (map)
- `labels` - (Computed) The cluster alert group labels (map)

» rancher2_cluster_alert_rule

Use this data source to retrieve information about a Rancher v2 cluster alert rule.

» Example Usage

```
data "rancher2_cluster_alert_rule" "foo" {
  cluster_id = "<cluster_id>"
  name = "<cluster_alert_rule_name>"
}
```

» Argument Reference

- `cluster_id` - (Required) The cluster id where create cluster alert rule (string)
- `name` - (Required) The cluster alert rule name (string)

» Attributes Reference

- `group_id` - (Computed) The cluster alert rule alert group ID (string)
- `event_rule` - (Computed) The cluster alert rule event rule. ConflictsWith: `"metric_rule", "node_rule", "system_service_rule"`‘ (list Maxitems:1)
- `group_interval_seconds` - (Computed) The cluster alert rule group interval seconds. Default: `180` (int)
- `group_wait_seconds` - (Computed) The cluster alert rule group wait seconds. Default: `180` (int)
- `inherited` - (Computed) The cluster alert rule inherited. Default: `true` (bool)
- `metric_rule` - (Computed) The cluster alert rule metric rule. ConflictsWith: `"event_rule", "node_rule", "system_service_rule"`‘ (list Maxitems:1)
- `node_rule` - (Computed) The cluster alert rule node rule. ConflictsWith: `"event_rule", "metric_rule", "system_service_rule"`‘ (list Maxitems:1)
- `repeat_interval_seconds` - (Optional) The cluster alert rule wait seconds. Default: `3600` (int)
- `severity` - (Computed) The cluster alert rule severity. Supported values : `"critical" | "info" | "warning"`. Default: `critical` (string)
- `system_service_rule` - (Computed) The cluster alert rule system service rule. ConflictsWith: `"event_rule", "metric_rule", "node_rule"`‘ (list Maxitems:1)
- `annotations` - (Computed) The cluster alert rule annotations (map)
- `labels` - (Computed) The cluster alert rule labels (map)

» rancher2_cluster_driver

Use this data source to retrieve information about a Rancher v2 Cluster Driver resource.

» Example Usage

```
data "rancher2_cluster_driver" "foo" {
```

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

» Argument Reference

- `name` - (Required) Name of the cluster driver (string)
- `url` - (Optional/Computed) The URL to download the machine driver binary for 64-bit Linux (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `active` - (Computed) Specify if the cluster driver state (bool)
- `builtin` - (Computed) Specify whether the cluster driver is an internal cluster driver or not (bool)
- `actual_url` - (Computed) Actual url of the cluster driver (string)
- `checksum` - (Computed) Verify that the downloaded driver matches the expected checksum (string)
- `ui_url` - (Computed) The URL to load for customized Add Clusters screen for this driver (string)
- `whitelist_domains` - (Computed) Domains to whitelist for the ui (list)
- `annotations` - (Computed) Annotations of the resource (map)
- `labels` - (Computed) Labels of the resource (map)

» rancher2_cluster_logging

Use this data source to retrieve information about a Rancher v2 Cluster Logging.

» Example Usage

```
data "rancher2_cluster_logging" "foo" {
  cluster_id = "<cluster_id>"
}
```

» Argument Reference

The following arguments are supported:

- `cluster_id` - (Required) The cluster id to configure logging (string)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `kind` - (Computed) The kind of the Cluster Logging. `elasticsearch`, `fluentd`, `kafka`, `splunk` and `syslog` are supported (string)
- `elasticsearch_config` - (Computed) The elasticsearch config for Cluster Logging. For `kind = elasticsearch` (list maxitems:1)
- `fluentd_config` - (Computed) The fluentd config for Cluster Logging. For `kind = fluentd` (list maxitems:1)
- `kafka_config` - (Computed) The kafka config for Cluster Logging. For `kind = kafka` (list maxitems:1)
- `name` - (Computed) The name of the cluster logging config (string)
- `namespace_id` - (Computed) The namespace id from cluster logging (string)
- `output_flush_interval` - (Computed) How often buffered logs would be flushed. Default: `3` seconds (int)
- `output_tags` - (computed) The output tags for Cluster Logging (map)
- `splunk_config` - (Computed) The splunk config for Cluster Logging. For `kind = splunk` (list maxitems:1)
- `syslog_config` - (Computed) The syslog config for Cluster Logging. For `kind = syslog` (list maxitems:1)
- `annotations` - (Computed) Annotations for Cluster Logging object (map)
- `labels` - (Computed) Labels for Cluster Logging object (map)

» rancher2_cluster_role_template_binding

Use this data source to retrieve information about a Rancher v2 cluster role template binding.

» Example Usage

```
data "rancher2_cluster_role_template_binding" "foo" {
    name = "foo"
    cluster_id = "foo_id"
}
```

» Argument Reference

- `name` - (Required) The name of the cluster role template binding (string)
- `cluster_id` - (Required) The cluster id where bind cluster role template (string)

- `role_template_id` - (Optional/Computed) The role template id from create cluster role template binding (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `group_id` - (Computed) The group ID to assign cluster role template binding (string)
- `group_principal_id` - (Computed) The group_principal ID to assign cluster role template binding (string)
- `user_id` - (Computed) The user ID to assign cluster role template binding (string)
- `user_principal_id` - (Computed) The user_principal ID to assign cluster role template binding (string)
- `annotations` - (Computed) Annotations of the resource (map)
- `labels` - (Computed) Labels of the resource (map)

» rancher2_cluster_scan

Use this data source to retrieve information about a Rancher v2 Cluster CIS Scan resource.

» Example Usage

```
data "rancher2_cluster_scan" "foo" {
    cluster_id = <clusterID>
    name = "foo"
}
```

» Argument Reference

- `cluster_id` - (Required) Cluster ID for CIS Scan (string)
- `name` - (Optional/Computed) Name of the cluster Scan (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `run_type` - (Computed) Cluster Scan run type (string)
- `scan_config` - (Computed) Cluster Scan config (bool)
- `scan_type` - (Computed) Cluster Scan type (string)
- `status` - (Computed) Cluster Scan status (string)

- `annotations` - (Computed) Annotations of the resource (map)
- `labels` - (Computed) Labels of the resource (map)

» Nested blocks

» `scan_config`

» Arguments

- `cis_scan_config` - (Computed) Cluster Cis Scan config (List maxitems:1)

» `cis_scan_config`

» Arguments

- `debug_master` - (Computed) Debug master. Default: `false` (bool)
- `debug_worker` - (Computed) Debug worker. Default: `false` (bool)
- `override_benchmark_version` - (Computed) Override benchmark version (string)
- `override_skip` - (Computed) Override skip (string)
- `profile` - (Computed) Cis scan profile. Allowed values: `"permissive" (default) || "hardened"` (string)

» rancher2_cluster_template

Use this data source to retrieve information about a Rancher v2 cluster template.

Cluster Templates are available from Rancher v2.3.x and above.

» Example Usage

```
data "rancher2_cluster_template" "foo" {
    name = "foo"
}
```

» Argument Reference

- `name` - (Required) The cluster template name (string)
- `decription` - (Optional/Computed) The cluster template description (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `default_revision_id` - (Computed) Default cluster template revision ID (string)
- `members` - (Computed) Cluster template members (list)
- `template_revisions` - (Computed) Cluster template revisions (list)
- `annotations` - (Computed) Annotations for the cluster template (map)
- `labels` - (Computed) Labels for the cluster template (map)

» rancher2_etcd_backup

Use this data source to retrieve information about a Rancher v2 etcd backup.

» Example Usage

```
data "rancher2_etcd_backup" "foo" {
  cluster_id = "<CLUSTER_ID>"
  name = "foo"
}
```

» Argument Reference

- `cluster_id` - (Required) Cluster ID to config Etcd Backup (string)
- `name` - (Required) The name of the Etcd Backup (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `backup_config` - (Computed) Backup config for etcd backup (list max-items:1)
- `filename` - (Computed) Filename of the Etcd Backup (string)
- `manual` - (Computed) Manual execution of the Etcd Backup. Default `false` (bool)
- `namespace_id` - (Computed) Description for the Etcd Backup (string)
- `annotations` - (Computed) Annotations for Etcd Backup object (map)
- `labels` - (Computed) Labels for Etcd Backup object (map)

» rancher2_global_role_binding

Use this data source to retrieve information about a Rancher v2 global role binding.

» Example Usage

```
data "rancher2_global_role_binding" "foo" {
    name = "foo"
    global_role_id = "foo_id"
}
```

» Argument Reference

- `name` - (Required) The name of the global role binding (string)
- `global_role_id` - (Optional/Computed) The global role id (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `group_principal_id` - (Computed) The group principal ID to assign global role binding. Rancher v2.4.0 or higher is required (string)
- `user_id` - (Computed) The user ID to assign global role binding (string)
- `annotations` - (Computed) Annotations of the resource (map)
- `labels` - (Computed) Labels of the resource (map)

» rancher2_multi_cluster_app

Use this data source to retrieve information about a Rancher v2 multi cluster app.

» Example Usage

```
data "rancher2_multi_cluster_app" "foo" {
    name = "foo"
}
```

» Argument Reference

- `name` - (Required) The multi cluster app name (string)

» Attributes Reference

- `catalog_name` - (Computed) The multi cluster app catalog name (string)
- `id` - (Computed) The ID of the resource (string)
- `roles` - (Computed) The multi cluster app roles (list)
- `targets` - (Computed) The multi cluster app target projects (list)
- `template_name` - (Computed) The multi cluster app template name (string)
- `template_version` - (Computed) The multi cluster app template version (string)
- `template_version_id` - (Computed) The multi cluster app template version ID (string)
- `answers` - (Computed) The multi cluster app answers (list)
- `members` - (Computed) The multi cluster app members (list)
- `revision_history_limit` - (Computed) The multi cluster app revision history limit (int)
- `revision_id` - (Computed) Current revision id for the multi cluster app (string)
- `upgrade_strategy` - (Computed) The multi cluster app upgrade strategy (list)
- `annotations` - (Computed) Annotations for multi cluster app object (map)
- `labels` - (Computed) Labels for multi cluster app object (map)

» rancher2_namespace

Use this data source to retrieve information about a Rancher v2 namespace.

» Example Usage

```
data "rancher2_namespace" "foo" {
  name = "foo"
  project_id = "${rancher2_cluster.foo-custom.default_project_id}"
}
```

» Argument Reference

- `name` - (Required) The name of the namespace (string)

- `project_id` - (Required) The project id where namespace is assigned (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `container_resource_limit` - (Computed) Default containers resource limits on namespace (List maxitem:1)
- `description` - (Computed) A namespace description (string)
- `resource_quota` - (Computed) Resource quota for namespace. Rancher v2.1.x or higher (list maxitems:1)
- `annotations` - (Computed) Annotations for Node Pool object (map)
- `labels` - (Computed) Labels for Node Pool object (map)

» rancher2_node_driver

Use this data source to retrieve information about a Rancher v2 Node Driver resource.

» Example Usage

```
data "rancher2_node_driver" "foo" {
    name = "foo"
}
```

» Argument Reference

- `name` - (Required) Name of the node driver (string)
- `url` - (Optional/Computed) The URL to download the machine driver binary for 64-bit Linux (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `active` - (Computed) Specify if the node driver state (bool)
- `builtin` - (Computed) Specify wheter the node driver is an internal cluster driver or not (bool)
- `checksum` - (Computed) Verify that the downloaded driver matches the expected checksum (string)
- `description` - (Computed) Description of the node driver (string)
- `external_id` - (Computed) External ID (string)

- `ui_url` - (Computed) The URL to load for customized Add Node screen for this driver (string)
- `whitelist_domains` - (Computed) Domains to whitelist for the ui (list)
- `annotations` - (Computed) Annotations of the resource (map)
- `labels` - (Computed) Labels of the resource (map)

» rancher2_node_pool

Use this data source to retrieve information about a Rancher v2 Node Pool resource.

» Example Usage

```
data "rancher2_node_pool" "foo" {
  cluster_id =  "${rancher2_cluster.foo-custom.id}"
  name = "foo"
}
```

» Argument Reference

- `cluster_id` - (Required) The RKE cluster id to use Node Pool (string)
- `name` - (Required) The name of the Node Pool (string)
- `node_template_id` - (Optional/Computed) The Node Template ID to use for node creation (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `hostname_prefix` - (Computed) The prefix for created nodes of the Node Pool (string)
- `delete_not_ready_after_secs` - (Computed) Delete not ready node after secs. Default `0` (int)
- `node_taints` - (Computed) Node taints (List)
- `quantity` - (Computed) The number of nodes to create on Node Pool (int)
- `control_plane` - (Computed) RKE control plane role for created nodes (bool)
- `etcd` - (Computed) RKE etcd role for created nodes (bool)
- `worker` - (Computed) RKE role role for created nodes (bool)
- `annotations` - (Computed) Annotations for Node Pool object (map)
- `labels` - (Computed) Labels for Node Pool object (map)

» rancher2_node_template

Use this data source to retrieve information about a Rancher v2 Node Template resource.

» Example Usage

```
data "rancher2_node_template" "foo" {
  name = "foo"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the Node Template (string)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `cloud_credential_id` - (Computed) Cloud credential ID for the Node Template. Required from Rancher v2.2.x (string)
- `description` - (Computed) Description for the Node Template (string)
- `driver` - (Computed) The driver of the node template (string)
- `engine_env` - (Computed) Engine environment for the node template (string)
- `engine_insecure_registry` - (Computed) Insecure registry for the node template (list)
- `engine_install_url` - (Computed) Docker engine install URL for the node template (string)
- `engine_label` - (Computed) Engine label for the node template (string)
- `engine_opt` - (Computed) Engine options for the node template (map)
- `engine_registry_mirror` - (Computed) Engine registry mirror for the node template (list)
- `engine_storage_driver` - (Computed) Engine storage driver for the node template (string)
- `use_internal_ip_address` - (Computed) Engine storage driver for the node template (bool)
- `annotations` - (Computed) Annotations for Node Template object (map)
- `labels` - (Computed) Labels for Node Template object (map)

» rancher2_notifier

Use this data source to retrieve information about a Rancher v2 notifier.

» Example Usage

```
data "rancher2_notifier" "foo" {
    name = "foo"
    cluster_id = "<cluster_id>"
}
```

» Argument Reference

- `name` - (Required) The name of the notifier (string)
- `cluster_id` - (Required) The cluster id where create notifier (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `description` - (Computed) The notifier description (string)
- `send_resolved` - (Computed) If the notifier sends resolved notifications (bool)
- `pagerduty_config` - (Computed) Pagerduty config for notifier (list maxitems:1)
- `slack_config` - (Computed) Slack config for notifier (list maxitems:1)
- `smtp_config` - (Computed) SMTP config for notifier (list maxitems:1)
- `webhook_config` - (Computed) Webhook config for notifier (list maxitems:1)
- `wechat_config` - (Computed) Wechat config for notifier (list maxitems:1)
- `annotations` - (Computed) Annotations for notifier object (map)
- `labels` - (Computed) Labels for notifier object (map)

» rancher2_project

Use this data source to retrieve information about a Rancher v2 project. This data source can be used in conjunction with the Terraform Kubernetes provider to associate Namespaces with projects.

» Example Usage

```
data "rancher2_project" "system" {
    cluster_id = "${var.my_cluster_id}"
    name = "System"
}

resource "kubernetes_namespace" "my_namespace" {
  metadata {
    annotations {
      "field.cattle.io/projectId" = "${data.rancher2_project.system.id}"
    }
    name = "my-namespace"
  }
}
```

» Argument Reference

- `cluster_id` - (Required) ID of the Rancher 2 cluster (string)
- `name` - (Required) The project name (string)

» Attributes Reference

- `id` - (Computed) Cluster-wide unique ID of the Rancher 2 project (string)
- `container_resource_limit` - (Computed) Default containers resource limits on project (List maxitem:1)
- `enable_project_monitoring` - (Computed) Enable built-in project monitoring. Default `false` (bool)
- `pod_security_policy_template_id` - (Computed) Default Pod Security Policy ID for the project (string)
- `resource_quota` - (Computed) Resource quota for project. Rancher v2.1.x or higher (list maxitems:1)
- `uuid` - (Computed) UUID of the project as stored by Rancher 2 (string)
- `description` - (Computed) The project's description (string)
- `annotations` - (Computed) Annotations of the rancher2 project (map)
- `labels` - (Computed) Labels of the rancher2 project (map)

» rancher2_project_logging

Use this data source to retrieve information about a Rancher v2 Project Logging.

» Example Usage

```
data "rancher2_project_logging" "foo" {
  project_id = "<project_id>"
}
```

» Argument Reference

The following arguments are supported:

- `project_id` - (Required) The project id to configure logging (string)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `kind` - (Computed) The kind of the Cluster Logging. `elasticsearch`, `fluentd`, `kafka`, `splunk` and `syslog` are supported (string)
- `elasticsearch_config` - (Computed) The elasticsearch config for Cluster Logging. For `kind = elasticsearch` (list maxitems:1)
- `fluentd_config` - (Computed) The fluentd config for Cluster Logging. For `kind = fluentd` (list maxitems:1)
- `kafka_config` - (Computed) The kafka config for Cluster Logging. For `kind = kafka` (list maxitems:1)
- `name` - (Computed) The name of the cluster logging config (string)
- `namespace_id` - (Computed) The namespace id from cluster logging (string)
- `output_flush_interval` - (Computed) How often buffered logs would be flushed. Default: `3` seconds (int)
- `output_tags` - (computed) The output tags for Cluster Logging (map)
- `splunk_config` - (Computed) The splunk config for Cluster Logging. For `kind = splunk` (list maxitems:1)
- `syslog_config` - (Computed) The syslog config for Cluster Logging. For `kind = syslog` (list maxitems:1)
- `annotations` - (Computed) Annotations for Cluster Logging object (map)
- `labels` - (Computed) Labels for Cluster Logging object (map)

» rancher2_project_role_template_binding

Use this data source to retrieve information about a Rancher v2 project role template binding.

» Example Usage

```
data "rancher2_project_role_template_binding" "foo" {
    name = "foo"
    project_id = "foo_id"
}
```

» Argument Reference

- `name` - (Required) The name of the project role template binding (string)
- `project_id` - (Required) The project id where bind project role template (string)
- `role_template_id` - (Optional/Computed) The role template id from create project role template binding (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `group_id` - (Computed) The group ID to assign project role template binding (string)
- `group_principal_id` - (Computed) The group_principal ID to assign project role template binding (string)
- `user_id` - (Computed) The user ID to assign project role template binding (string)
- `user_principal_id` - (Computed) The user_principal ID to assign project role template binding (string)
- `annotations` - (Computed) Annotations of the resource (map)
- `labels` - (Computed) Labels of the resource (map)

» rancher2_registry

Use this data source to retrieve information about a Rancher v2 docker registry.

Depending of the availability, there are 2 types of Rancher v2 docker registries: - Project registry: Available to all namespaces in the `project_id` - Namespaced registry: Available to just `namespace_id` in the `project_id`

» Example Usage

```
# Retrieve a rancher2 Project Registry
data "rancher2_registry" "foo" {
  name = "<name>"
```

```
  project_id = "<project_id>"
}

# Retrieve a rancher2 Namespaced Registry
data "rancher2_registry" "foo" {
  name = "<name>"
  project_id = "<project_id>"
  namespace_id = "<namespace_id>"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the registry (string)
- `project_id` - (Required) The project id where to assign the registry (string)
- `namespace_id` - (Optional) The namespace id where to assign the namespaced registry (string)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `registries` - (Computed) Registries data for registry (list)
- `description` - (Computed) A registry description (string)
- `annotations` - (Computed) Annotations for Registry object (map)
- `labels` - (Computed) Labels for Registry object (map)

» rancher2_role_template

Use this data source to retrieve information about a Rancher v2 role template resource.

» Example Usage

```
data "rancher2_role_template" "foo" {
  name = "foo"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the Node Template (string)
- `context` - (Optional/Computed) Role template context. `cluster` and `project` values are supported (string)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `builtin` - (Computed) Builtin role template (string)
- `administrative` - (Computed) Administrative role template (bool)
- `default_role` - (Computed) Default role template for new created cluster or project (bool)
- `description` - (Computed) Role template description (string)
- `external` - (Computed) External role template (bool)
- `hidden` - (Computed) Hidden role template (bool)
- `locked` - (Computed) Locked role template (bool)
- `role_template_ids` - (Computed) Inherit role template IDs (list)
- `rules` - (Computed) Role template policy rules (list)
- `annotations` - (Computed) Annotations for role template object (map)
- `labels` - (Computed) Labels for role template object (map)

» rancher2_secret

Use this data source to retrieve information about a Rancher v2 secret.

Depending of the availability, there are 2 types of Rancher v2 secrets: - Project secret: Available to all namespaces in the `project_id` - Namespaced secret: Available to just `namespace_id` in the `project_id`

» Example Usage

```
# Retrieve a rancher2 Project Secret
data "rancher2_secret" "foo" {
  name = "<name>"
  project_id = "<project_id>"
}

# Retrieve a rancher2 Namespaced Secret
data "rancher2_secret" "foo" {
```

```
  name = "<name>"
  project_id = "<project_id>"
  namespace_id = "<namespace_id>"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the secret (string)
- `project_id` - (Required) The project id where to assign the secret (string)
- `namespace_id` - (Optional) The namespace id where to assign the namespaced secret (string)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `data` - (Computed) Secret key/value data. Base64 encoding required for values (map)
- `description` - (Computed) A secret description (string)
- `annotations` - (Computed) Annotations for secret object (map)
- `labels` - (Computed) Labels for secret object (map)

» rancher2_setting

Use this data source to retrieve information about a Rancher v2 setting.

» Example Usage

```
data "rancher2_setting" "server-image" {
    name = "server-image"
}
```

» Argument Reference

- `name` - (Required) The setting name.

» Attributes Reference

- `value` - the settting's value.

» rancher2_user

Use this data source to retrieve information about a Rancher v2 user

» Example Usage

```
data "rancher2_user" "foo" {
    username = "foo"
}
```

» Argument Reference

- `is_external` - (Optional) Set is the user if the user is external. Default: `false` (bool)
- `name` - (Optional) The name of the user (string)
- `username` - (Optional) The username of the user (string)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `name` - (Computed) The user common name (string)
- `annotations` - (Computed) Annotations of the resource (map)
- `enabled` - (Computed) The user is enabled (bool)
- `principal_ids` - (Computed) The user principal IDs (list)
- `labels` - (Computed) Labels of the resource (map)

» rancher2_app

Provides a Rancher v2 app resource. This can be used to deploy apps within Rancher v2 projects.

This resource can also modify Rancher v2 apps in 3 ways: - `Update`: If `description`, `annotations` or `labels` arguments are modified the app will be updated. No new `revision_id` will be generated in Rancher. - `Upgrade`: If `answers`, `catalog_name`, `template_name`, `template_version` or `values_yaml` arguments are modified, the app will be upgraded. A new `revision_id` will

be generated in Rancher. - `Rollback`: If `revision_id` argument is provided or modified the app will be rolled back accordingly. A new `revision_id` will be generated in Rancher. It will also generate a non-empty terraform plan that will require manual .tf file intervention. Use carefully.

Note: In case of multiple resource modifications in a row, `rollback` has preference over `upgrade`.

» Example Usage

```
# Create a new rancher2 App
resource "rancher2_app" "foo" {
  catalog_name = "<catalog_name>"
  name = "foo"
  description = "Foo app"
  project_id = "<project_id>"
  template_name = "<template_name>"
  template_version = "<template_version>"
  target_namespace = "<namespace_name>"
  answers = {
    "ingress_host" = "test.xip.io"
    "foo" = "bar"
    "ingress.annotations.nginx.ingress.kubernetes.io/force-ssl-redirect" = true
  }
}
```

```
# Create a new rancher2 App in a new namespace
resource "rancher2_namespace" "foo" {
  name = "foo"
  description = "Foo namespace"
  project_id = "<project_id>"
  resource_quota {
    limit {
      limits_cpu = "100m"
      limits_memory = "100Mi"
      requests_storage = "1Gi"
    }
  }
}

resource "rancher2_app" "foo" {
  catalog_name = "<catalog_name>"
  name = "foo"
  description = "Foo app"
  project_id = "<project_id>"
  template_name = "<template_name>"
```

```
  template_version = "<template_version>"
  target_namespace = "${rancher2_namespace.foo.name}"
  answers = {
    "ingress_host" = "test.xip.io"
    "foo" = "bar"
    "ingress.annotations.nginx.ingress.kubernetes.io/force-ssl-redirect" = true
  }
}
```

» Argument Reference

The following arguments are supported:

- `catalog_name` - (Required) Catalog name of the app. If modified, app will be upgraded. For use scoped catalogs:
  - add cluster ID before name, `c-XXXXX:<name>`
  - add project ID before name, `p-XXXXX:<name>`
- `name` - (Required/ForceNew) The name of the app (string)
- `project_id` - (Required/ForceNew) The project id where the app will be installed (string)
- `target_namespace` - (Required/ForceNew) The namespace name where the app will be installed (string)
- `template_name` - (Required) Template name of the app. If modified, app will be upgraded (string)
- `answers` - (Optional) Answers for the app template. If modified, app will be upgraded (map)
- `description` - (Optional/Computed) Description for the app (string)
- `force_upgrade` - (Optional) Force app upgrade (string)
- `revision_id` - (Optional/Computed) Current revision id for the app. If modified, If this argument is provided or modified, app will be rollbacked to `revision_id` (string)
- `template_version` - (Optional/Computed) Template version of the app. If modified, app will be upgraded. Default: `latest` (string)
- `values_yaml` - (Optional) values.yaml base64 encoded file content for the app template. If modified, app will be upgraded (string)
- `annotations` - (Optional/Computed) Annotations for App object (map)
- `labels` - (Optional/Computed) Labels for App object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `external_id` - (Computed) The url of the app template on a catalog (string)

» Timeouts

`rancher2_app` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating apps.
- `update` - (Default `10 minutes`) Used for app modifications.
- `delete` - (Default `10 minutes`) Used for deleting apps.

» Import

Apps can be imported using the app ID in the format `<project_id>:<app_name>`

```
$ terraform import rancher2_app.foo <project_id>:<app_name>
```

» rancher2_auth_config_activedirectory

Provides a Rancher v2 Auth Config ActiveDirectory resource. This can be used to configure and enable Auth Config ActiveDirectory for Rancher v2 RKE clusters and retrieve their information.

In addition to the built-in local auth, only one external auth config provider can be enabled at a time.

» Example Usage

```
# Create a new rancher2 Auth Config ActiveDirectory
resource "rancher2_auth_config_activedirectory" "activedirectory" {
  servers = ["<ACTIVEDIRECTORY_SERVER>"]
  service_account_username = "<SERVICE_DN>"
  service_account_password = "<SERVICE_PASSWORD>"
  user_search_base = "<SEARCH_BASE>"
  port = <ACTIVEDIRECTORY_PORT>
}
```

» Argument Reference

The following arguments are supported:

- `servers` - (Required) ActiveDirectory servers list (list)
- `service_account_username` - (Required/Sensitive) Service account DN for access ActiveDirectory service (string)
- `service_account_password` - (Required/Sensitive) Service account password for access ActiveDirectory service (string)

- `user_search_base` - (Required) User search base DN (string)
- `access_mode` - (Optional) Access mode for auth. `required`, `restricted`, `unrestricted` are supported. Default `unrestricted` (string)
- `allowed_principal_ids` - (Optional) Allowed principal ids for auth. Required if `access_mode` is `required` or `restricted`. Ex: `activedirectory_user://<DN> activedirectory_group://<DN>` (list)
- `certificate` - (Optional/Sensitive) CA certificate for TLS if selfsigned (string)
- `connection_timeout` - (Optional) ActiveDirectory connection timeout. Default `5000` (int)
- `default_login_domain` - (Optional) ActiveDirectory defult lgoin domain (string)
- `enabled` - (Optional) Enable auth config provider. Default `true` (bool)
- `group_dn_attribute` - (Optional/Computed) Group DN attribute. Default `distinguishedName` (string)
- `group_member_mapping_attribute` - (Optional/Computed) Group member mapping attribute. Default `member` (string)
- `group_member_user_attribute` - (Optional/Computed) Group member user attribute. Default `distinguishedName` (string)
- `group_name_attribute` - (Optional/Computed) Group name attribute. Default `name` (string)
- `group_object_class` - (Optional/Computed) Group object class. Default `group` (string)
- `group_search_attribute` - (Optional/Computed) Group search attribute. Default `sAMAccountName` (string)
- `group_search_base` - (Optional/Computed) Group search base (string)
- `group_search_filter` - (Optional/Computed) Group search filter (string)
- `nested_group_membership_enabled` - (Optional/Computed) Nested group membership enable. Default `false` (bool)
- `port` - (Optional) ActiveDirectory port. Default `389` (int)
- `user_disabled_bit_mask` - (Optional) User disabled bit mask. Default `2` (int)
- `user_enabled_attribute` - (Optional/Computed) User enable attribute (string)
- `user_login_attribute` - (Optional/Computed) User login attribute. Default `sAMAccountName` (string)
- `user_name_attribute` - (Optional/Computed) User name attribute. Default `name` (string)
- `user_object_class` - (Optional/Computed) User object class. Default `person` (string)
- `user_search_attribute` - (Optional/Computed) User search attribute. Default `sAMAccountName|sn|givenName` (string)
- `user_search_filter` - (Optional/Computed) User search filter (string)
- `tls` - (Optional/Computed) Enable TLS connection (bool)
- `annotations` - (Optional/Computed) Annotations of the resource (map)

- `labels` - (Optional/Computed) Labels of the resource (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `name` - (Computed) The name of the resource (string)
- `type` - (Computed) The type of the resource (string)

» rancher2_auth_config_adfs

Provides a Rancher v2 Auth Config ADFS resource. This can be used to configure and enable Auth Config ADFS for Rancher v2 RKE clusters and retrieve their information.

In addition to the built-in local auth, only one external auth config provider can be enabled at a time.

» Example Usage

```
# Create a new rancher2 Auth Config ADFS
resource "rancher2_auth_config_adfs" "adfs" {
  display_name_field = "<DISPLAY_NAME_FIELD>"
  groups_field = "<GROUPS_FIELD>"
  idp_metadata_content = "<IDP_METADATA_CONTENT>"
  rancher_api_host = "https://<RANCHER_API_HOST>"
  sp_cert = "<SP_CERT>"
  sp_key = "<SP_KEY>"
  uid_field = "<UID_FIELD>"
  user_name_field = "<USER_NAME_FIELD>"
}
```

» Argument Reference

The following arguments are supported:

- `display_name_field` - (Required) ADFS display name field (string)
- `groups_field` - (Required) ADFS group field (string)
- `idp_metadata_content` - (Required/Sensitive) ADFS IDP metadata content (string)
- `rancher_api_host` - (Required) Rancher url. Schema needs to be specified, `https://<RANCHER_API_HOST>` (string)

- `sp_cert` - (Required/Sensitive) ADFS SP cert (string)
- `sp_key` - (Required/Sensitive) ADFS SP key (string)
- `uid_field` - (Required) ADFS UID field (string)
- `user_name_field` - (Required) ADFS user name field (string)
- `access_mode` - (Optional) Access mode for auth. `required`, `restricted`, `unrestricted` are supported. Default `unrestricted` (string)
- `allowed_principal_ids` - (Optional) Allowed principal ids for auth. Required if `access_mode` is `required` or `restricted`. Ex: `adfs_user://<USER_ID>` `adfs_group://<GROUP_ID>` (list)
- `enabled` - (Optional) Enable auth config provider. Default `true` (bool)
- `annotations` - (Optional/Computed) Annotations of the resource (map)
- `labels` - (Optional/Computed) Labels of the resource (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `name` - (Computed) The name of the resource (string)
- `type` - (Computed) The type of the resource (string)

» rancher2_auth_config_azuread

Provides a Rancher v2 Auth Config AzureAD resource. This can be used to configure and enable Auth Config AzureAD for Rancher v2 RKE clusters and retrieve their information.

In addition to the built-in local auth, only one external auth config provider can be enabled at a time.

» Example Usage

```
# Create a new rancher2 Auth Config AzureAD
resource "rancher2_auth_config_azuread" "azuread" {
  application_id = "<AZUREAD_APP_ID>"
  application_secret = "<AZUREAD_APP_SECRET>"
  auth_endpoint = "<AZUREAD_AUTH_ENDPOINT>"
  graph_endpoint = "<AZUREAD_GRAPH_ENDPOINT>"
  rancher_url = "<RANCHER_URL>"
  tenant_id = "<AZUREAD_TENANT_ID>"
  token_endpoint = "<AZUREAD_TOKEN_ENDPOINT>"
}
```

» Argument Reference

The following arguments are supported:

- `application_id` - (Required/Sensitive) AzureAD auth application ID (string)
- `application_secret` - (Required/Sensitive) AzureAD auth application secret (string)
- `auth_endpoint` - (Required) AzureAD auth endpoint (string)
- `graph_endpoint` - (Required) AzureAD graph endpoint (string)
- `rancher_url` - (Required) Rancher URL (string). "/verify-auth-azure"
- `tenant_id` - (Required) AzureAD tenant ID (string)
- `token_endpoint` - (Required) AzureAD token endpoint (string)
- `endpoint` - (Optional) AzureAD endpoint. Default `https://login.microsoftonline.com/` (string)
- `access_mode` - (Optional) Access mode for auth. `required`, `restricted`, `unrestricted` are supported. Default `unrestricted` (string)
- `allowed_principal_ids` - (Optional) Allowed principal ids for auth. Required if `access_mode` is `required` or `restricted`. Ex: `azuread_user://<USER_ID> azuread_group://<GROUP_ID>` (list)
- `enabled` - (Optional) Enable auth config provider. Default `true` (bool)
- `tls` - (Optional) Enable TLS connection. Default `true` (bool)
- `annotations` - (Optional/Computed) Annotations of the resource (map)
- `labels` - (Optional/Computed) Labels of the resource (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `name` - (Computed) The name of the resource (string)
- `type` - (Computed) The type of the resource (string)

» rancher2_auth_config_freeipa

Provides a Rancher v2 Auth Config FreeIpa resource. This can be used to configure and enable Auth Config FreeIpa for Rancher v2 RKE clusters and retrieve their information.

In addition to the built-in local auth, only one external auth config provider can be enabled at a time.

» Example Usage

```
# Create a new rancher2 Auth Config FreeIpa
resource "rancher2_auth_config_freeipa" "freeipa" {
  servers = ["<FREEIPA_SERVER>"]
  service_account_distinguished_name = "<SERVICE_DN>"
  service_account_password = "<SERVICE_PASSWORD>"
  user_search_base = "<SEARCH_BASE>"
  port = <FREEIPA_PORT>
}
```

» Argument Reference

The following arguments are supported:

- `servers` - (Required) FreeIpa servers list (list)
- `service_account_distinguished_name` - (Required/Sensitive) Service account DN for access FreeIpa service (string)
- `service_account_password` - (Required/Sensitive) Service account password for access FreeIpa service (string)
- `user_search_base` - (Required) User search base DN (string)
- `access_mode` - (Optional) Access mode for auth. `required`, `restricted`, `unrestricted` are supported. Default `unrestricted` (string)
- `allowed_principal_ids` - (Optional) Allowed principal ids for auth. Required if `access_mode` is `required` or `restricted`. Ex: `freeipa_user://<DN>` `freeipa_group://<DN>` (list)
- `certificate` - (Optional/Sensitive) Base64 encoded CA certificate for TLS if self-signed. Use filebase64() for encoding file (string)
- `connection_timeout` - (Optional) FreeIpa connection timeout. Default `5000` (int)
- `enabled` - (Optional) Enable auth config provider. Default `true` (bool)
- `group_dn_attribute` - (Optional/Computed) Group DN attribute. Default `entryDN` (string)
- `group_member_mapping_attribute` - (Optional/Computed) Group member mapping attribute. Default `member` (string)
- `group_member_user_attribute` - (Optional/Computed) Group member user attribute. Default `entryDN` (string)
- `group_name_attribute` - (Optional/Computed) Group name attribute. Default `cn` (string)
- `group_object_class` - (Optional/Computed) Group object class. Default `groupOfNames` (string)
- `group_search_attribute` - (Optional/Computed) Group search attribute. Default `cn` (string)
- `group_search_base` - (Optional/Computed) Group search base (string)

- `nested_group_membership_enabled` - (Optional/Computed) Nested group membership enable. Default `false` (bool)
- `port` - (Optional) FreeIpa port. Default `389` (int)
- `user_disabled_bit_mask` - (Optional/Computed) User disabled bit mask (int)
- `user_enabled_attribute` - (Optional/Computed) User enable attribute (string)
- `user_login_attribute` - (Optional/Computed) User login attribute. Default `uid` (string)
- `user_member_attribute` - (Optional/Computed) User member attribute. Default `memberOf` (string)
- `user_name_attribute` - (Optional/Computed) User name attribute. Default `givenName` (string)
- `user_object_class` - (Optional/Computed) User object class. Default `inetorgperson` (string)
- `user_search_attribute` - (Optional/Computed) User search attribute. Default `uid|sn|givenName` (string)
- `tls` - (Optional/Computed) Enable TLS connection (bool)
- `annotations` - (Optional/Computed) Annotations of the resource (map)
- `labels` - (Optional/Computed) Labels of the resource (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `name` - (Computed) The name of the resource (string)
- `type` - (Computed) The type of the resource (string)

» rancher2_auth_config_github

Provides a Rancher v2 Auth Config Github resource. This can be used to configure and enable Auth Config Github for Rancher v2 RKE clusters and retrieve their information.

In addition to the built-in local auth, only one external auth config provider can be enabled at a time.

» Example Usage

```
# Create a new rancher2 Auth Config Github
resource "rancher2_auth_config_github" "github" {
  client_id = "<GITHUB_CLIENT_ID>"
```

```
  client_secret = "<GITHUB_CLIENT_SECRET>"
}
```

» Argument Reference

The following arguments are supported:

- `client_id` - (Required/Sensitive) Github auth Client ID (string)
- `client_secret` - (Required/Sensitive) Github auth Client secret (string)
- `hostname` - (Optional) Github hostname to connect. Default `github.com` (string)
- `access_mode` - (Optional) Access mode for auth. `required`, `restricted`, `unrestricted` are supported. Default `unrestricted` (string)
- `allowed_principal_ids` - (Optional) Allowed principal ids for auth. Required if `access_mode` is `required` or `restricted`. Ex: `github_user://<USER_ID>` `github_team://<GROUP_ID>` `github_org://<ORG_ID>` (list)
- `enabled` - (Optional) Enable auth config provider. Default `true` (bool)
- `tls` - (Optional) Enable TLS connection. Default `true` (bool)
- `annotations` - (Optional/Computed) Annotations of the resource (map)
- `labels` - (Optional/Computed) Labels of the resource (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `name` - (Computed) The name of the resource (string)
- `type` - (Computed) The type of the resource (string)

» rancher2_auth_config_keycloak

Provides a Rancher v2 Auth Config KeyCloak resource. This can be used to configure and enable Auth Config KeyCloak for Rancher v2 RKE clusters and retrieve their information.

In addition to the built-in local auth, only one external auth config provider can be enabled at a time.

» Example Usage

```
# Create a new rancher2 Auth Config KeyCloak
resource "rancher2_auth_config_keycloak" "keycloak" {
```

```
  display_name_field = "<DISPLAY_NAME_FIELD>"
  groups_field = "<GROUPS_FIELD>"
  idp_metadata_content = "<IDP_METADATA_CONTENT>"
  rancher_api_host = "https://<RANCHER_API_HOST>"
  sp_cert = "<SP_CERT>"
  sp_key = "<SP_KEY>"
  uid_field = "<UID_FIELD>"
  user_name_field = "<USER_NAME_FIELD>"
}
```

» Argument Reference

The following arguments are supported:

- `display_name_field` - (Required) KeyCloak display name field (string)
- `groups_field` - (Required) KeyCloak group field (string)
- `idp_metadata_content` - (Required/Sensitive) KeyCloak IDP metadata content (string)
- `rancher_api_host` - (Required) Rancher url. Schema needs to be specified, `https://<RANCHER_API_HOST>` (string)
- `sp_cert` - (Required/Sensitive) KeyCloak SP cert (string)
- `sp_key` - (Required/Sensitive) KeyCloak SP key (string)
- `uid_field` - (Required) KeyCloak UID field (string)
- `user_name_field` - (Required) KeyCloak user name field (string)
- `access_mode` - (Optional) Access mode for auth. `required`, `restricted`, `unrestricted` are supported. Default `unrestricted` (string)
- `allowed_principal_ids` - (Optional) Allowed principal ids for auth. Required if `access_mode` is `required` or `restricted`. Ex: `keycloak_user://<USER_ID>` `keycloak_group://<GROUP_ID>` (list)
- `enabled` - (Optional) Enable auth config provider. Default `true` (bool)
- `annotations` - (Optional/Computed) Annotations of the resource (map)
- `labels` - (Optional/Computed) Labels of the resource (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `name` - (Computed) The name of the resource (string)
- `type` - (Computed) The type of the resource (string)

» rancher2_auth_config_okta

Provides a Rancher v2 Auth Config OKTA resource. This can be used to configure and enable Auth Config OKTA for Rancher v2 RKE clusters and retrieve their information.

In addition to the built-in local auth, only one external auth config provider can be enabled at a time.

» Example Usage

```
# Create a new rancher2 Auth Config OKTA
resource "rancher2_auth_config_okta" "okta" {
  display_name_field = "<DISPLAY_NAME_FIELD>"
  groups_field = "<GROUPS_FIELD>"
  idp_metadata_content = "<IDP_METADATA_CONTENT>"
  rancher_api_host = "https://<RANCHER_API_HOST>"
  sp_cert = "<SP_CERT>"
  sp_key = "<SP_KEY>"
  uid_field = "<UID_FIELD>"
  user_name_field = "<USER_NAME_FIELD>"
}
```

» Argument Reference

The following arguments are supported:

- `display_name_field` - (Required) OKTA display name field (string)
- `groups_field` - (Required) OKTA group field (string)
- `idp_metadata_content` - (Required/Sensitive) OKTA IDP metadata content (string)
- `rancher_api_host` - (Required) Rancher url. Schema needs to be specified, `https://<RANCHER_API_HOST>` (string)
- `sp_cert` - (Required/Sensitive) OKTA SP cert (string)
- `sp_key` - (Required/Sensitive) OKTA SP key (string)
- `uid_field` - (Required) OKTA UID field (string)
- `user_name_field` - (Required) OKTA user name field (string)
- `access_mode` - (Optional) Access mode for auth. `required`, `restricted`, `unrestricted` are supported. Default `unrestricted` (string)
- `allowed_principal_ids` - (Optional) Allowed principal ids for auth. Required if `access_mode` is `required` or `restricted`. Ex: `okta_user://<USER_ID>` `okta_group://<GROUP_ID>` (list)
- `enabled` - (Optional) Enable auth config provider. Default `true` (bool)
- `annotations` - (Optional/Computed) Annotations of the resource (map)

- `labels` - (Optional/Computed) Labels of the resource (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `name` - (Computed) The name of the resource (string)
- `type` - (Computed) The type of the resource (string)

» rancher2_auth_config_openldap

Provides a Rancher v2 Auth Config OpenLdap resource. This can be used to configure and enable Auth Config OpenLdap for Rancher v2 RKE clusters and retrieve their information.

In addition to the built-in local auth, only one external auth config provider can be enabled at a time.

» Example Usage

```
# Create a new rancher2 Auth Config OpenLdap
resource "rancher2_auth_config_openldap" "openldap" {
  servers = ["<OPENLDAP_SERVER>"]
  service_account_distinguished_name = "<SERVICE_DN>"
  service_account_password = "<SERVICE_PASSWORD>"
  user_search_base = "<SEARCH_BASE>"
  port = <OPENLDAP_PORT>
}
```

» Argument Reference

The following arguments are supported:

- `servers` - (Required) OpenLdap servers list (list)
- `service_account_distinguished_name` - (Required/Sensitive) Service account DN for access OpenLdap service (string)
- `service_account_password` - (Required/Sensitive) Service account password for access OpenLdap service (string)
- `user_search_base` - (Required) User search base DN (string)
- `access_mode` - (Optional) Access mode for auth. `required`, `restricted`, `unrestricted` are supported. Default `unrestricted` (string)

- `allowed_principal_ids` - (Optional) Allowed principal ids for auth. Required if `access_mode` is `required` or `restricted`. Ex: `openldap_user://<DN> openldap_group://<DN>` (list)
- `certificate` - (Optional/Sensitive) Base64 encoded CA certificate for TLS if self-signed. Use filebase64() for encoding file (string)
- `connection_timeout` - (Optional) OpenLdap connection timeout. Default `5000` (int)
- `enabled` - (Optional) Enable auth config provider. Default `true` (bool)
- `group_dn_attribute` - (Optional/Computed) Group DN attribute. Default `entryDN` (string)
- `group_member_mapping_attribute` - (Optional/Computed) Group member mapping attribute. Default `member` (string)
- `group_member_user_attribute` - (Optional/Computed) Group member user attribute. Default `entryDN` (string)
- `group_name_attribute` - (Optional/Computed) Group name attribute. Default `cn` (string)
- `group_object_class` - (Optional/Computed) Group object class. Default `groupOfNames` (string)
- `group_search_attribute` - (Optional/Computed) Group search attribute. Default `cn` (string)
- `group_search_base` - (Optional/Computed) Group search base (string)
- `nested_group_membership_enabled` - (Optional/Computed) Nested group membership enable. Default `false` (bool)
- `port` - (Optional) OpenLdap port. Default `389` (int)
- `user_disabled_bit_mask` - (Optional/Computed) User disabled bit mask (int)
- `user_enabled_attribute` - (Optional/Computed) User enable attribute (string)
- `user_login_attribute` - (Optional/Computed) User login attribute. Default `uid` (string)
- `user_member_attribute` - (Optional/Computed) User member attribute. Default `memberOf` (string)
- `user_name_attribute` - (Optional/Computed) User name attribute. Default `givenName` (string)
- `user_object_class` - (Optional/Computed) User object class. Default `inetorgperson` (string)
- `user_search_attribute` - (Optional/Computed) User search attribute. Default `uid|sn|givenName` (string)
- `tls` - (Optional/Computed) Enable TLS connection (bool)
- `annotations` - (Optional/Computed) Annotations of the resource (map)
- `labels` - (Optional/Computed) Labels of the resource (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `name` - (Computed) The name of the resource (string)
- `type` - (Computed) The type of the resource (string)

» rancher2_auth_config_ping

Provides a Rancher v2 Auth Config Ping resource. This can be used to configure and enable Auth Config Ping for Rancher v2 RKE clusters and retrieve their information.

In addition to the built-in local auth, only one external auth config provider can be enabled at a time.

» Example Usage

```
# Create a new rancher2 Auth Config Ping
resource "rancher2_auth_config_ping" "ping" {
  display_name_field = "<DISPLAY_NAME_FIELD>"
  groups_field = "<GROUPS_FIELD>"
  idp_metadata_content = "<IDP_METADATA_CONTENT>"
  rancher_api_host = "https://<RANCHER_API_HOST>"
  sp_cert = "<SP_CERT>"
  sp_key = "<SP_KEY>"
  uid_field = "<UID_FIELD>"
  user_name_field = "<USER_NAME_FIELD>"
}
```

» Argument Reference

The following arguments are supported:

- `display_name_field` - (Required) Ping display name field (string)
- `groups_field` - (Required) Ping group field (string)
- `idp_metadata_content` - (Required/Sensitive) Ping IDP metadata content (string)
- `rancher_api_host` - (Required) Rancher url. Schema needs to be specified, `https://<RANCHER_API_HOST>` (string)
- `sp_cert` - (Required/Sensitive) Ping SP cert (string)
- `sp_key` - (Required/Sensitive) Ping SP key (string)
- `uid_field` - (Required) Ping UID field (string)
- `user_name_field` - (Required) Ping user name field (string)
- `access_mode` - (Optional) Access mode for auth. `required`, `restricted`, `unrestricted` are supported. Default `unrestricted` (string)

- `allowed_principal_ids` - (Optional) Allowed principal ids for auth. Required if `access_mode` is `required` or `restricted`. Ex: `ping_user://<USER_ID>` `ping_group://<GROUP_ID>` (list)
- `enabled` - (Optional) Enable auth config provider. Default `true` (bool)
- `annotations` - (Optional/Computed) Annotations of the resource (map)
- `labels` - (Optional/Computed) Labels of the resource (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `name` - (Computed) The name of the resource (string)
- `type` - (Computed) The type of the resource (string)

» rancher2_bootstrap

Provides a Rancher v2 bootstrap resource. This can be used to bootstrap Rancher v2 environments and output information. It just works if `bootstrap` provider config is added to the .tf file. More info at rancher2 provider

This resource bootstraps a Rancher system by performing the following tasks: - Updates the default admin password, provided by setting `password` or generating a random one. - Sets `server-url` setting, based on `api_url`. - Sets `telemetry-opt` setting. - Creates a token for admin user with concrete TTL.

Rancher2 admin password can be updated after the initial run of terraform by setting `password` field and applying this resource again.

Rancher2 admin `token` can also be regenerated if `token_update` is set to true. Refresh resource function will check if token is expired. If it is expired, `token_update` will be set to true to force token regeneration on next `terraform apply`.

Login to Rancher2 is done by trying to use `token` first. If it fails, it uses admin `current_password`. If admin password has been changed outside of terraform and the terraform `token` is expired, `current_password` field can be specified to allow terraform to manage admin password and token again.

» Example Usage

```
# Provider bootstrap config
provider "rancher2" {
  api_url   = "https://rancher.my-domain.com"
  bootstrap = true
```

```
}

# Create a new rancher2_bootstrap
resource "rancher2_bootstrap" "admin" {
  password = "blahblah"
  telemetry = true
}

# Provider bootstrap config with alias
provider "rancher2" {
  alias = "bootstrap"

  api_url   = "https://rancher.my-domain.com"
  bootstrap = true
}

# Create a new rancher2_bootstrap using bootstrap provider config
resource "rancher2_bootstrap" "admin" {
  provider = "rancher2.bootstrap"

  password = "blahblah"
  telemetry = true
}
```

» Argument Reference

The following arguments are supported:

- `current_password` - (Optional/computed/sensitive) Current password for Admin user. Just needed for recover if admin password has been changed from other resources and token is expired (string)
- `password` - (Optional/computed/sensitive) Password for Admin user or random generated if empty (string)
- `telemetry` - (Optional) Send telemetry anonymous data. Default: `false` (bool)
- `token_ttl` - (Optional) TTL in seconds for generated admin token. Default: `0` (int)
- `token_update` - (Optional) Regenerate admin token. Default: `false` (bool)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

- `token` - (Computed) Generated API token for Admin User (string)
- `token_id` - (Computed) Generated API token id for Admin User (string)
- `url` - (Computed) URL set as server-url (string)
- `user` - (Computed) Admin username (string)
- `temp_token` - (Computed) Generated API temporary token as helper. Should be empty (string)
- `temp_token_id` - (Computed) Generated API temporary token id as helper. Should be empty (string)

» rancher2_catalog

Provides a Rancher v2 Catalog resource. This can be used to create cluster, global and/or project catalogs for Rancher v2 environments and retrieve their information.

» Example Usage

```
# Create a new Rancher2 Global Catalog
resource "rancher2_catalog" "foo-global" {
  name = "foo-global"
  url = "https://<CATALOG_URL>"
}
# Create a new Rancher2 Cluster Catalog
resource "rancher2_catalog" "foo-cluster" {
  name = "foo-cluster"
  url = "https://<CATALOG_URL>"
  scope = "cluster"
}
# Create a new Rancher2 Project Catalog
resource "rancher2_catalog" "foo-project" {
  name = "foo-project"
  url = "https://<CATALOG_URL>"
  scope = "project"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the catalog (string)
- `url` - (Required) The url of the catalog repo (string)
- `branch` - (Optional) The branch of the catalog repo to use. Default `master` (string)

- `cluster_id` - (Optional/ForceNew) The cluster id of the catalog. Mandatory if `scope = cluster` (string)
- `description` - (Optional) A catalog description (string)
- `kind` - (Optional) The kind of the catalog. Just helm by the moment (string)
- `password` - (Optional/Sensitive) The password to access the catalog if needed (string)
- `project_id` - (Optional/ForceNew) The project id of the catalog. Mandatory if `scope = project` (string)
- `refresh` - (Optional) Catalog will wait for refresh after tf creation and on every tf read. Default `false` (bool)
- `scope` - (Optional) The scope of the catalog. `cluster`, `global`, and `project` are supported. Default `global` (string)
- `username` - (Optional/Sensitive) The username to access the catalog if needed (string)
- `version` - (Optional/ForceNew/Computed) Helm version for the catalog. Available options: `helm_v2` and `helm_v3` (string)
- `annotations` - (Optional/Computed) Annotations for the catalog (map)
- `labels` - (Optional/Computed) Labels for the catalog (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Timeouts

`rancher2_catalog` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating catalogs.
- `update` - (Default `10 minutes`) Used for catalog modifications.
- `delete` - (Default `10 minutes`) Used for deleting catalogs.

» Import

Catalogs can be imported using the Rancher Catalog ID and its scope.

```
$ terraform import rancher2_catalog.foo <scope>.<catalog_id>
```

» rancher2_certificate

Provides a Rancher v2 certificate resource. This can be used to create certificates for Rancher v2 environments and retrieve their information.

There are 2 types of Rancher v2 certificates: - Project certificate: Available to all namespaces in the `project_id` - Namespaced certificate: Available to just `namespace_id` in the `project_id`

» Example Usage

```
# Create a new rancher2 Project Certificate
resource "rancher2_certificate" "foo" {
  certs = base64encode(<PUBLIC_CERTS>)
  key = base64encode(<PRIVATE_KEY>)
  name = "foo"
  description = "Terraform certificate foo"
  project_id = "<project_id>"
}

# Create a new rancher2 Namespaced Certificate
resource "rancher2_certificate" "foo" {
  certs = base64encode(<PUBLIC_CERTS>)
  key = base64encode(<PRIVATE_KEY>)
  name = "foo"
  description = "Terraform certificate foo"
  project_id = "<project_id>"
  namespace_id = "<namespace_id>"
}
```

» Argument Reference

The following arguments are supported:

- `certs` - (Required) Base64 encoded public certs (string)
- `key` - (Required/Sensitive) Base64 encoded private key (string)
- `project_id` - (Required/ForceNew) The project id where the certificate should be created (string)
- `description` - (Optional) A certificate description (string)
- `name` - (Optional/ForceNew) The name of the certificate (string)
- `namespace_id` - (Optional/ForceNew) The namespace id where the namespaced certificate should be created (string)
- `annotations` - (Optional/Computed) Annotations for certificate object (map)
- `labels` - (Optional/Computed) Labels for certificate object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Timeouts

`rancher2_certificate` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating registries.
- `update` - (Default `10 minutes`) Used for certificate modifications.
- `delete` - (Default `10 minutes`) Used for deleting registries.

» rancher2_cloud_credential

Provides a Rancher v2 Cloud Credential resource. This can be used to create Cloud Credential for Rancher v2.2.x and retrieve their information.

amazonec2, azure, digitalocean, openstack and vsphere credentials config are supported for Cloud Credential.

» Example Usage

```
# Create a new rancher2 Cloud Credential
resource "rancher2_cloud_credential" "foo" {
  name = "foo"
  description = "foo test"
  amazonec2_credential_config {
    access_key = "<AWS_ACCESS_KEY>"
    secret_key = "<AWS_SECRET_KEY>"
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the Cloud Credential (string)
- `amazonec2_credential_config` - (Optional) AWS config for the Cloud Credential (list maxitems:1)
- `azure_credential_config` - (Optional) Azure config for the Cloud Credential (list maxitems:1)

- `description` - (Optional) Description for the Cloud Credential (string)
- `digitalocean_credential_config` - (Optional) DigitalOcean config for the Cloud Credential (list maxitems:1)
- `openstack_credential_config` - (Optional) OpenStack config for the Cloud Credential (list maxitems:1)
- `vsphere_credential_config` - (Optional) vSphere config for the Cloud Credential (list maxitems:1)
- `annotations` - (Optional) Annotations for Cloud Credential object (map)
- `labels` - (Optional/Computed) Labels for Cloud Credential object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `driver` - (Computed) The driver of the Cloud Credential (string)

» Nested blocks

» `amazonec2_credential_config`

» Arguments

- `access_key` - (Required/Sensitive) AWS access key (string)
- `secret_key` - (Required/Sensitive) AWS secret key (string)

» `azure_credential_config`

» Arguments

- `client_id` - (Required/Sensitive) Azure Service Principal Account ID (string)
- `client_secret` - (Required/Sensitive) Azure Service Principal Account password (string)
- `subscription_id` - (Required/Sensitive) Azure Subscription ID (string)

» `digitalocean_credential_config`

» Arguments

- `access_token` - (Required/Sensitive) DigitalOcean access token (string)

» `openstack_credential_config`

» Arguments

- `password` - (Required/Sensitive) OpenStack password (string)

» `vsphere_credential_config`

» Arguments

- `password` - (Required/Sensitive) vSphere password (string)
- `username` - (Required) vSphere username (string)
- `vcenter` - (Required) vSphere IP/hostname for vCenter (string)
- `vcenter_port` - (Optional) vSphere Port for vCenter. Default `443` (string)

» Timeouts

`rancher2_cloud_credential` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating cloud credentials.
- `update` - (Default `10 minutes`) Used for cloud credential modifications.
- `delete` - (Default `10 minutes`) Used for deleting cloud credentials.

» rancher2_cluster

Provides a Rancher v2 Cluster resource. This can be used to create Clusters for Rancher v2 environments and retrieve their information.

» Example Usage

Creating Rancher v2 imported cluster

```
# Create a new rancher2 imported Cluster
resource "rancher2_cluster" "foo-imported" {
  name = "foo-imported"
  description = "Foo rancher2 imported cluster"
}
```

Creating Rancher v2 RKE cluster

```
# Create auditlog policy yaml file
auditlog_policy.yaml
apiVersion: audit.k8s.io/v1
```

```
kind: Policy
rules:
  - level: RequestResponse
    resources:
    - group: ""
      resources: ["pods"]

# Create a new rancher2 RKE Cluster
resource "rancher2_cluster" "foo-custom" {
  name = "foo-custom"
  description = "Foo rancher2 custom cluster"
  rke_config {
    network {
      plugin = "canal"
    }
    services {
      kube_api {
        audit_log {
          enabled = true
          configuration {
            max_age = 5
            max_backup = 5
            max_size = 100
            path = "-"
            format = "json"
            policy = file("auditlog_policy.yaml")
          }
        }
      }
    }
  }
}
```

Creating Rancher v2 RKE cluster enabling and customizing monitoring

```
# Create a new rancher2 RKE Cluster
resource "rancher2_cluster" "foo-custom" {
  name = "foo-custom"
  description = "Foo rancher2 custom cluster"
  rke_config {
    network {
      plugin = "canal"
    }
  }
  enable_cluster_monitoring = true
  cluster_monitoring_input {
    answers = {
```

```
      "exporter-kubelets.https" = true
      "exporter-node.enabled" = true
      "exporter-node.ports.metrics.port" = 9796
      "exporter-node.resources.limits.cpu" = "200m"
      "exporter-node.resources.limits.memory" = "200Mi"
      "grafana.persistence.enabled" = false
      "grafana.persistence.size" = "10Gi"
      "grafana.persistence.storageClass" = "default"
      "operator.resources.limits.memory" = "500Mi"
      "prometheus.persistence.enabled" = "false"
      "prometheus.persistence.size" = "50Gi"
      "prometheus.persistence.storageClass" = "default"
      "prometheus.persistent.useReleaseName" = "true"
      "prometheus.resources.core.limits.cpu" = "1000m",
      "prometheus.resources.core.limits.memory" = "1500Mi"
      "prometheus.resources.core.requests.cpu" = "750m"
      "prometheus.resources.core.requests.memory" = "750Mi"
      "prometheus.retention" = "12h"
    }
  }
}
```

Creating Rancher v2 RKE cluster assigning a node pool (overlapped planes)

```
# Create a new rancher2 RKE Cluster
resource "rancher2_cluster" "foo-custom" {
  name = "foo-custom"
  description = "Foo rancher2 custom cluster"
  rke_config {
    network {
      plugin = "canal"
    }
  }
}
# Create a new rancher2 Node Template
resource "rancher2_node_template" "foo" {
  name = "foo"
  description = "foo test"
  amazonec2_config {
    access_key = "AWS_ACCESS_KEY"
    secret_key = "<AWS_SECRET_KEY>"
    ami =  "<AMI_ID>"
    region = "<REGION>"
    security_group = ["<AWS_SECURITY_GROUP>"]
    subnet_id = "<SUBNET_ID>"
    vpc_id = "<VPC_ID>"
    zone = "<ZONE>"
```

```
  }
}
# Create a new rancher2 Node Pool
resource "rancher2_node_pool" "foo" {
  cluster_id =  "${rancher2_cluster.foo-custom.id}"
  name = "foo"
  hostname_prefix =  "foo-cluster-0"
  node_template_id = "${rancher2_node_template.foo.id}"
  quantity = 3
  control_plane = true
  etcd = true
  worker = true
}
```

Creating Rancher v2 RKE cluster from template. For Rancher v2.3.x or above.

```
# Create a new rancher2 cluster template
resource "rancher2_cluster_template" "foo" {
  name = "foo"
  members {
    access_type = "owner"
    user_principal_id = "local://user-XXXXX"
  }
  template_revisions {
    name = "V1"
    cluster_config {
      rke_config {
        network {
          plugin = "canal"
        }
        services {
          etcd {
            creation = "6h"
            retention = "24h"
          }
        }
      }
    }
    default = true
  }
  description = "Test cluster template v2"
}
# Create a new rancher2 RKE Cluster from template
resource "rancher2_cluster" "foo" {
  name = "foo"
  cluster_template_id = "${rancher2_cluster_template.foo.id}"
  cluster_template_revision_id = "${rancher2_cluster_template.foo.template_revisions.0.id}"
```

```
}
```

Creating Rancher v2 RKE cluster with upgrade strategy. For Rancher v2.4.x or above.

```
resource "rancher2_cluster" "foo" {
  name = "foo"
  description = "Terraform custom cluster"
  rke_config {
    network {
      plugin = "canal"
    }
    services {
      etcd {
        creation = "6h"
        retention = "24h"
      }
      kube_api {
        audit_log {
          enabled = true
          configuration {
            max_age = 5
            max_backup = 5
            max_size = 100
            path = "-"
            format = "json"
            policy = "apiVersion: audit.k8s.io/v1\nkind: Policy\nmetadata:\n  creationTimest
          }
        }
      }
    }
    upgrade_strategy {
      drain = true
      max_unavailable_worker = "20%"
    }
  }
}
```

Creating Rancher v2 RKE cluster with scheduled cluster scan. For Rancher v2.4.x or above.

```
resource "rancher2_cluster" "foo" {
  name = "foo"
  description = "Terraform custom cluster"
  rke_config {
    network {
      plugin = "canal"
    }
```

```
    services {
      etcd {
        creation = "6h"
        retention = "24h"
      }
    }
  }
  scheduled_cluster_scan {
    enabled = true
    scan_config {
      cis_scan_config {
        debug_master = true
        debug_worker = true
      }
    }
    schedule_config {
      cron_schedule = "30 * * * *"
      retention = 5
    }
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the Cluster (string)
- `rke_config` - (Optional/Computed) The RKE configuration for `rke` Clusters. Conflicts with `aks_config`, `eks_config`, `gke_config` and `k3s_config` (list maxitems:1)
- `k3s_config` - (Optional/Computed) The K3S configuration for `k3s` imported Clusters. Conflicts with `aks_config`, `eks_config`, `gke_config` and `rke_config` (list maxitems:1)
- `aks_config` - (Optional) The Azure AKS configuration for `aks` Clusters. Conflicts with `eks_config`, `gke_config`, `k3s_config` and `rke_config` (list maxitems:1)
- `eks_config` - (Optional) The Amazon EKS configuration for `eks` Clusters. Conflicts with `aks_config`, `gke_config`, `k3s_config` and `rke_config` (list maxitems:1)
- `gke_config` - (Optional) The Google GKE configuration for `gke` Clusters. Conflicts with `aks_config`, `eks_config`, `k3s_config` and `rke_config` (list maxitems:1)
- `description` - (Optional) The description for Cluster (string)
- `cluster_auth_endpoint` - (Optional/Computed) Enabling the local cluster authorized endpoint allows direct communication with the cluster, by-

passing the Rancher API proxy. (list maxitems:1)

- `cluster_monitoring_input` - (Optional/Computed) Cluster monitoring config. Any parameter defined in rancher-monitoring charts could be configured (list maxitems:1)
- `cluster_template_answers` - (Optional/Computed) Cluster template answers. Just for Rancher v2.3.x and above (list maxitems:1)
- `cluster_template_id` - (Optional) Cluster template ID. Just for Rancher v2.3.x and above (string)
- `cluster_template_questions` - (Optional) Cluster template questions. Just for Rancher v2.3.x and above (list)
- `cluster_template_revision_id` - (Optional) Cluster template revision ID. Just for Rancher v2.3.x and above (string)
- `default_pod_security_policy_template_id` - (Optional/Computed) Default pod security policy template id (string)
- `desired_agent_image` - (Optional/Computed) Desired agent image. Just for Rancher v2.3.x and above (string)
- `desired_auth_image` - (Optional/Computed) Desired auth image. Just for Rancher v2.3.x and above (string)
- `docker_root_dir` - (Optional/Computed) Desired auth image. Just for Rancher v2.3.x and above (string)
- `enable_cluster_alerting` - (Optional) Enable built-in cluster alerting. Default `false` (bool)
- `enable_cluster_monitoring` - (Optional) Enable built-in cluster monitoring. Default `false` (bool)
- `enable_cluster_istio` - (Optional) Enable built-in cluster istio. Default `false`. Just for Rancher v2.3.x and above (bool)
- `enable_network_policy` - (Optional) Enable project network isolation. Default `false` (bool)
- `scheduled_cluster_scan`- (Optional) Cluster scheduled cis scan. For Rancher v2.4.0 or above (List maxitems:1)
- `annotations` - (Optional/Computed) Annotations for Node Pool object (map)
- `labels` - (Optional/Computed) Labels for Node Pool object (map)
- `windows_prefered_cluster` - (Optional) Windows preferred cluster. Default: `false` (bool)

» `schedule_config`

» Arguments

- `cron_schedule` - (Required) Crontab schedule. It should contains 5 fields `"<min> <hour> <month_day> <month> <week_day>"` (string)
- `retention` - (Optional/Computed) Cluster scan retention (int)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `cluster_registration_token` - (Computed) Cluster Registration Token generated for the cluster (list maxitems:1)
- `default_project_id` - (Computed) Default project ID for the cluster (string)
- `driver` - (Computed) The driver used for the Cluster. `imported`, `azurekubernetesservice`, `amazonelasticcontainerservice`, `googlekubernetesengine` and `rancherKubernetesEngine` are supported (string)
- `kube_config` - (Computed/Sensitive) Kube Config generated for the cluster (string)
- `system_project_id` - (Computed) System project ID for the cluster (string)

» Nested blocks

» `rke_config`

» Arguments

- `addon_job_timeout` - (Optional/Computed) Duration in seconds of addon job (int)
- `addons` - (Optional) Addons descripton to deploy on RKE cluster.
- `addons_include` - (Optional) Addons yaml manifests to deploy on RKE cluster (list)
- `authentication` - (Optional/Computed) Kubernetes cluster authentication (list maxitems:1)
- `authorization` - (Optional/Computed) Kubernetes cluster authorization (list maxitems:1)
- `bastion_host` - (Optional/Computed) RKE bastion host (list maxitems:1)
- `cloud_provider` - (Optional/Computed) RKE cloud provider rke-cloud-providers (list maxitems:1)
- `dns` - (Optional/Computed) RKE dns add-on. Just for Rancher v2.2.x (list maxitems:1)
- `ignore_docker_version` - (Optional) Ignore docker version. Default `true` (bool)
- `ingress` - (Optional/Computed) Kubernetes ingress configuration (list maxitems:1)
- `kubernetes_version` - (Optional/Computed) K8s version to deploy. Default: `Rancher default` (string)
- `monitoring` - (Optional/Computed) Kubernetes cluster monitoring (list maxitems:1)

- `network` - (Optional/Computed) Kubernetes cluster networking (list maxitems:1)
- `nodes` - (Optional) RKE cluster nodes (list)
- `prefix_path` - (Optional/Computed) Prefix to customize Kubernetes path (string)
- `private_registries` - (Optional) private registries for docker images (list)
- `services` - (Optional/Computed) Kubernetes cluster services (list maxitems:1)
- `ssh_agent_auth` - (Optional) Use ssh agent auth. Default `false`
- `ssh_cert_path` - (Optional/Computed) Cluster level SSH certificate path (string)
- `ssh_key_path` - (Optional/Computed) Cluster level SSH private key path (string)
- `upgrade_strategy` - (Optional/Computed) RKE upgrade strategy (list maxitems:1)

» `authentication`

» Arguments

- `sans` - (Optional/Computed) RKE sans for authentication ([]string)
- `strategy` - (Optional/Computed) RKE strategy for authentication (string)

» `authorization`

» Arguments

- `mode` - (Optional) RKE mode for authorization. `rbac` and `none` modes are available. Default `rbac` (string)
- `options` - (Optional/Computed) RKE options for authorization (map)

» `bastion_host`

» Arguments

- `address` - (Required) Address ip for the bastion host (string)
- `user` - (Required) User to connect bastion host (string)
- `port` - (Optional) Port for bastion host. Default `22` (string)
- `ssh_agent_auth` - (Optional) Use ssh agent auth. Default `false` (bool)
- `ssh_key` - (Optional/Computed/Sensitive) Bastion host SSH private key (string)

- `ssh_key_path` - (Optional/Computed) Bastion host SSH private key path (string)

» `cloud_provider`

» Arguments

- `aws_cloud_provider` - (Optional/Computed) RKE AWS Cloud Provider config for Cloud Provider rke-aws-cloud-provider (list maxitems:1)
- `azure_cloud_provider` - (Optional/Computed) RKE Azure Cloud Provider config for Cloud Provider rke-azure-cloud-provider (list maxitems:1)
- `custom_cloud_provider` - (Optional/Computed) RKE Custom Cloud Provider config for Cloud Provider (string) (string)
- `name` - (Optional/Computed) RKE sans for Cloud Provider. `aws`, `azure`, `custom`, `openstack`, `vsphere` are supported. (string)
- `openstack_cloud_provider` - (Optional/Computed) RKE Openstack Cloud Provider config for Cloud Provider rke-openstack-cloud-provider (list maxitems:1)
- `vsphere_cloud_provider` - (Optional/Computed) RKE Vsphere Cloud Provider config for Cloud Provider rke-vsphere-cloud-provider Extra argument `name` is required on `virtual_center` configuration. (list maxitems:1)

» `aws_cloud_provider`

» Arguments

- `global` - (Optional/Computed) (list maxitems:1)
- `service_override` - (Optional) (list)

» `global`

» Arguments

- `disable_security_group_ingress` - (Optional) Default `false` (bool)
- `disable_strict_zone_check` - (Optional) Default `false` (bool)
- `elb_security_group` - (Optional/Computed) (string)
- `kubernetes_cluster_id` - (Optional/Computed) (string)
- `kubernetes_cluster_tag` - (Optional/Computed) (string)
- `role_arn` - (Optional/Computed) (string)
- `route_table_id` - (Optional/Computed/Sensitive) (string)
- `subnet_id` - (Optional/Computed) (string)
- `vpc` - (Optional/Computed) (string)
- `zone` - (Optional/Computed) (string)

» `service_override`

» Arguments

- `service` - (Required) (string)
- `region` - (Optional/Computed) (string)
- `signing_method` - (Optional/Computed) (string)
- `signing_name` - (Optional/Computed) (string)
- `signing_region` - (Optional/Computed) (string)
- `url` - (Optional/Computed) (string)

» `azure_cloud_provider`

» Arguments

- `aad_client_id` - (Required/Sensitive) (string)
- `aad_client_secret` - (Required/Sensitive) (string)
- `subscription_id` - (Required/Sensitive) (string)
- `tenant_id` - (Required/Sensitive) (string)
- `aad_client_cert_password` - (Optional/Computed/Sensitive) (string)
- `aad_client_cert_path` - (Optional/Computed) (string)
- `cloud` - (Optional/Computed) (string)
- `cloud_provider_backoff` - (Optional/Computed) (bool)
- `cloud_provider_backoff_duration` - (Optional/Computed) (int)
- `cloud_provider_backoff_exponent` - (Optional/Computed) (int)
- `cloud_provider_backoff_jitter` - (Optional/Computed) (int)
- `cloud_provider_backoff_retries` - (Optional/Computed) (int)
- `cloud_provider_rate_limit` - (Optional/Computed) (bool)
- `cloud_provider_rate_limit_bucket` - (Optional/Computed) (int)
- `cloud_provider_rate_limit_qps` - (Optional/Computed) (int)
- `location` - (Optional/Computed) (string)
- `maximum_load_balancer_rule_count` - (Optional/Computed) (int)
- `primary_availability_set_name` - (Optional/Computed) (string)
- `primary_scale_set_name` - (Optional/Computed) (string)
- `resource_group` - (Optional/Computed) (string)
- `route_table_name` - (Optional/Computed) (string)
- `security_group_name` - (Optional/Computed) (string)
- `subnet_name` - (Optional/Computed) (string)
- `use_instance_metadata` - (Optional/Computed) (bool)
- `use_managed_identity_extension` - (Optional/Computed) (bool)
- `vm_type` - (Optional/Computed) (string)
- `vnet_name` - (Optional/Computed) (string)
- `vnet_resource_group` - (Optional/Computed) (string)

» `openstack_cloud_provider`

» Arguments

- `global` - (Required) (list maxitems:1)
- `block_storage` - (Optional/Computed) (list maxitems:1)
- `load_balancer` - (Optional/Computed) (list maxitems:1)
- `metadata` - (Optional/Computed) (list maxitems:1)
- `route` - (Optional/Computed) (list maxitems:1)

» `global`

» Arguments

- `auth_url` - (Required) (string)
- `password` - (Required/Sensitive) (string)
- `username` - (Required/Sensitive) (string)
- `ca_file` - (Optional/Computed) (string)
- `domain_id` - (Optional/Computed/Sensitive) Required if `domain_name` not provided. (string)
- `domain_name` - (Optional/Computed) Required if `domain_id` not provided. (string)
- `region` - (Optional/Computed) (string)
- `tenant_id` - (Optional/Computed/Sensitive) Required if `tenant_name` not provided. (string)
- `tenant_name` - (Optional/Computed) Required if `tenant_id` not provided. (string)
- `trust_id` - (Optional/Computed/Sensitive) (string)

» `block_storage`

» Arguments

- `bs_version` - (Optional/Computed) (string)
- `ignore_volume_az` - (Optional/Computed) (string)
- `trust_device_path` - (Optional/Computed) (string)

» `load_balancer`

» Arguments

- `create_monitor` - (Optional/Computed) (bool)
- `floating_network_id` - (Optional/Computed) (string)
- `lb_method` - (Optional/Computed) (string)
- `lb_provider` - (Optional/Computed) (string)
- `lb_version` - (Optional/Computed) (string)
- `manage_security_groups` - (Optional/Computed) (bool)
- `monitor_delay` - (Optional/Computed) Default `60s` (string)
- `monitor_max_retries` - (Optional/Computed) Default 5 (int)
- `monitor_timeout` - (Optional/Computed) Default `30s` (string)
- `subnet_id` - (Optional/Computed) (string)
- `use_octavia` - (Optional/Computed) (bool)

» `metadata`

» Arguments

- `request_timeout` - (Optional/Computed) (int)
- `search_order` - (Optional/Computed) (string)

» `route`

» Arguments

- `router_id` - (Optional/Computed) (string)

» `vsphere_cloud_provider`

» Arguments

- `virtual_center` - (Required) (List)
- `workspace` - (Required) (list maxitems:1)
- `disk` - (Optional/Computed) (list maxitems:1)
- `global` - (Optional/Computed) (list maxitems:1)
- `network` - (Optional/Computed) (list maxitems:1)

» `virtual_center`

» Arguments

- `datacenters` - (Required) (string)
- `name` - (Required) Name of virtualcenter config for Vsphere Cloud Provider config (string)
- `password` - (Required/Sensitive) (string)
- `user` - (Required/Sensitive) (string)
- `port` - (Optional/Computed) (string)
- `soap_roundtrip_count` - (Optional/Computed) (int)

» `workspace`

» Arguments

- `datacenter` - (Required) (string)
- `folder` - (Required) (string)
- `server` - (Required) (string)
- `default_datastore` - (Optional/Computed) (string)
- `resourcepool_path` - (Optional/Computed) (string)

» `disk`

» Arguments

- `scsi_controller_type` - (Optional/Computed) (string)

» `global`

» Arguments

- `datacenters` - (Optional/Computed) (string)

- `insecure_flag` - (Optional/Computed) (bool)
- `password` - (Optional/Computed) (string)
- `user` - (Optional/Computed) (string)
- `port` - (Optional/Computed) (string)
- `soap_roundtrip_count` - (Optional/Computed) (int)

» `network`

» Arguments

- `public_network` - (Optional/Computed) (string)

» `dns`

» Arguments

- `node_selector` - (Optional/Computed) DNS add-on node selector (map)
- `provider` - (Optional) DNS add-on provider. `kube-dns`, `coredns` (default), and `none` are supported (string)
- `reverse_cidrs` - (Optional/Computed) DNS add-on reverse cidr (list)
- `upstream_nameservers` - (Optional/Computed) DNS add-on upstream nameservers (list)

» `ingress`

» Arguments

- `dns_policy` - (Optional/Computed) Ingress controller DNS policy. `ClusterFirstWithHostNet`, `ClusterFirst`, `Default`, and `None` are supported. K8S dns Policy (string)
- `extra_args` - (Optional/Computed) Extra arguments for RKE Ingress (map)
- `node_selector` - (Optional/Computed) Node selector for RKE Ingress (map)
- `options` - (Optional/Computed) RKE options for Ingress (map)
- `provider` - (Optional/Computed) Provider for RKE Ingress (string)

» `monitoring`

» Arguments

- `options` - (Optional/Computed) RKE options for monitoring (map)
- `provider` - (Optional/Computed) Provider for RKE monitoring (string)

» network

» Arguments

- `calico_network_provider` - (Optional/Computed) Calico provider config for RKE network (list maxitems:1)
- `canal_network_provider` - (Optional/Computed) Canal provider config for RKE network (list maxitems:1)
- `flannel_network_provider` - (Optional/Computed) Flannel provider config for RKE network (list maxitems:1)
- `weave_network_provider` - (Optional/Computed) Weave provider config for RKE network (list maxitems:1)
- `mtu` - (Optional) Network provider MTU. Default `0` (int)
- `options` - (Optional/Computed) RKE options for network (map)
- `plugin` - (Optional/Computed) Plugin for RKE network. `canal` (default), `flannel`, `calico`, `none` and `weave` are supported. (string)

» calico_network_provider

» Arguments

- `cloud_provider` - (Optional/Computed) RKE options for Calico network provider (string)

» canal_network_provider

» Arguments

- `iface` - (Optional/Computed) Iface config Canal network provider (string)

» flannel_network_provider

» Arguments

- `iface` - (Optional/Computed) Iface config Flannel network provider (string)

» weave_network_provider

» Arguments

- `password` - (Optional/Computed) Password config Weave network provider (string)

» nodes

» Arguments

- `address` - (Required) Address ip for node (string)
- `role` - (Requires) Roles for the node. `controlplane`, `etcd` and `worker` are supported. (list)
- `user` - (Required/Sensitive) User to connect node (string)
- `docker_socket` - (Optional/Computed) Docker socket for node (string)
- `hostname_override` - (Optional) Hostname override for node (string)
- `internal_address` - (Optional) Internal ip for node (string)
- `labels` - (Optional) Labels for the node (map)
- `node_id` - (Optional) Id for the node (string)
- `port` - (Optional) Port for node. Default `22` (string)
- `ssh_agent_auth` - (Optional) Use ssh agent auth. Default `false` (bool)
- `ssh_key` - (Optional/Computed/Sensitive) Node SSH private key (string)
- `ssh_key_path` - (Optional/Computed) Node SSH private key path (string)

» `private_registries`

» Arguments

- `url` - (Required) Registry URL (string)
- `is_default` - (Optional) Set as default registry. Default `false` (bool)
- `password` - (Optional/Sensitive) Registry password (string)
- `user` - (Optional/Sensitive) Registry user (string)

» `services`

» Arguments

- `etcd` - (Optional/Computed) Etcd options for RKE services (list maxitems:1)
- `kube_api` - (Optional/Computed) Kube API options for RKE services (list maxitems:1)
- `kube_controller` - (Optional/Computed) Kube Controller options for RKE services (list maxitems:1)
- `kubelet` - (Optional/Computed) Kubelet options for RKE services (list maxitems:1)
- `kubeproxy` - (Optional/Computed) Kubeproxy options for RKE services (list maxitems:1)
- `scheduler` - (Optional/Computed) Scheduler options for RKE services (list maxitems:1)

» `etcd`

» Arguments

- `backup_config` - (Optional/Computed) Backup options for etcd service. Just for Rancher v2.2.x (list maxitems:1)
- `ca_cert` - (Optional/Computed) TLS CA certificate for etcd service (string)
- `cert` - (Optional/Computed/Sensitive) TLS certificate for etcd service (string)
- `creation` - (Optional/Computed) Creation option for etcd service (string)
- `external_urls` - (Optional) External urls for etcd service (list)
- `extra_args` - (Optional/Computed) Extra arguments for etcd service (map)
- `extra_binds` - (Optional) Extra binds for etcd service (list)
- `extra_env` - (Optional) Extra environment for etcd service (list)
- `gid` - (Optional) Etcd service GID. Default: `0`. For Rancher v2.3.x or above (int)
- `image` - (Optional/Computed) Docker image for etcd service (string)
- `key` - (Optional/Computed/Sensitive) TLS key for etcd service (string)
- `path` - (Optional/Computed) Path for etcd service (string)
- `retention` - (Optional/Computed) Retention option for etcd service (string)
- `snapshot` - (Optional/Computed) Snapshot option for etcd service (bool)
- `uid` - (Optional) Etcd service UID. Default: `0`. For Rancher v2.3.x or above (int)

» `backup_config`

» Arguments

- `enabled` - (Optional) Enable etcd backup (bool)
- `interval_hours` - (Optional) Interval hours for etcd backup. Default `12` (int)
- `retention` - (Optional) Retention for etcd backup. Default `6` (int)
- `s3_backup_config` - (Optional) S3 config options for etcd backup (list maxitems:1)
- `safe_timestamp` - (Optional) Safe timestamp for etcd backup. Default: `false` (bool)

» `s3_backup_config`

» Arguments

- `access_key` - (Optional/Sensitive) Access key for S3 service (string)
- `bucket_name` - (Required) Bucket name for S3 service (string)
- `custom_ca` - (Optional) Base64 encoded custom CA for S3 service. Use filebase64() for encoding file. Available from Rancher v2.2.5 (string)
- `endpoint` - (Required) Endpoint for S3 service (string)

- `folder` - (Optional) Folder for S3 service. Available from Rancher v2.2.7 (string)
- `region` - (Optional) Region for S3 service (string)
- `secret_key` - (Optional/Sensitive) Secret key for S3 service (string)

» kube_api

» Arguments

- `admission_configuration` - (Optional) Admission configuration (map)
- `always_pull_images` - (Optional) Enable AlwaysPullImages Admission controller plugin. Rancher docs Default: `false` (bool)
- `audit_log` - (Optional) K8s audit log configuration. (list maxitems: 1)
- `event_rate_limit` - (Optional) K8s event rate limit configuration. (list maxitems: 1)
- `extra_args` - (Optional/Computed) Extra arguments for kube API service (map)
- `extra_binds` - (Optional) Extra binds for kube API service (list)
- `extra_env` - (Optional) Extra environment for kube API service (list)
- `image` - (Optional/Computed) Docker image for kube API service (string)
- `pod_security_policy` - (Optional) Pod Security Policy option for kube API service. Default `false` (bool)
- `secrets_encryption_config` - (Optional) Encrypt k8s secret data configration. (list maxitem: 1)
- `service_cluster_ip_range` - (Optional/Computed) Service Cluster IP Range option for kube API service (string)
- `service_node_port_range` - (Optional/Computed) Service Node Port Range option for kube API service (string)

» `audit_log`

» Arguments

- `configuration` - (Optional) Audit log configuration. (list maxitems: 1)
- `enabled` - (Optional) Enable audit log. Default: `false` (bool)

» `configuration`

» Arguments

- `format` - (Optional) Audit log format. Default: 'json' (string)
- `max_age` - (Optional) Audit log max age. Default: `30` (int)
- `max_backup` - (Optional) Audit log max backup. Default: `10` (int)
- `max_size` - (Optional) Audit log max size. Default: `100` (int)
- `path` - (Optional) (Optional) Audit log path. Default: `/var/log/kube-audit/audit-log.json` (string)
- `policy` - (Optional/Computed) Audit policy yaml encoded definition. `apiVersion` and `kind: Policy\nrules:"` fields are required in the yaml. Ex. `"apiVersion: audit.k8s.io/v1\nkind: Policy\nrules:\n-`

`level: RequestResponse\n  resources:\n  - resources:\n    - pods\n"` More info (string)

» `event_rate_limit`

» Arguments

- `configuration` - (Optional) Event rate limit configuration. (map)
- `enabled` - (Optional) Enable event rate limit. Default: `false` (bool)

» `secrets_encryption_config`

» Arguments

- `custom_config` - (Optional) Secrets encryption configuration. (map)
- `enabled` - (Optional) Enable secrets encryption. Default: `false` (bool)

» `kube_controller`

» Arguments

- `cluster_cidr` - (Optional/Computed) Cluster CIDR option for kube controller service (string)
- `extra_args` - (Optional/Computed) Extra arguments for kube controller service (map)
- `extra_binds` - (Optional) Extra binds for kube controller service (list)
- `extra_env` - (Optional) Extra environment for kube controller service (list)
- `image` - (Optional/Computed) Docker image for kube controller service (string)
- `service_cluster_ip_range` - (Optional/Computed) Service Cluster ip Range option for kube controller service (string)

» `kubelet`

» Arguments

- `cluster_dns_server` - (Optional/Computed) Cluster DNS Server option for kubelet service (string)
- `cluster_domain` - (Optional/Computed) Cluster Domain option for kubelet service (string)
- `extra_args` - (Optional/Computed) Extra arguments for kubelet service (map)
- `extra_binds` - (Optional) Extra binds for kubelet service (list)
- `extra_env` - (Optional) Extra environment for kubelet service (list)
- `fail_swap_on` - (Optional/Computed) Enable or disable failing when swap on is not supported (bool)
- `generate_serving_certificate` Generate a certificate signed by the kube-ca. Default `false` (bool)

- `image` - (Optional/Computed) Docker image for kubelet service (string)
- `infra_container_image` - (Optional/Computed) Infra container image for kubelet service (string)

» `kubeproxy`

» Arguments

- `extra_args` - (Optional/Computed) Extra arguments for kubeproxy service (map)
- `extra_binds` - (Optional) Extra binds for kubeproxy service (list)
- `extra_env` - (Optional) Extra environment for kubeproxy service (list)
- `image` - (Optional/Computed) Docker image for kubeproxy service (string)

» `scheduler`

» Arguments

- `extra_args` - (Optional/Computed) Extra arguments for scheduler service (map)
- `extra_binds` - (Optional) Extra binds for scheduler service (list)
- `extra_env` - (Optional) Extra environment for scheduler service (list)
- `image` - (Optional/Computed) Docker image for scheduler service (string)

» `upgrade_strategy`

» Arguments

- `drain` - (Optional) RKE drain nodes. Default: `false` (bool)
- `drain_input` - (Optional/Computed) RKE drain node input (list Maxitems: 1)
- `max_unavailable_controlplane` - (Optional) RKE max unavailable controlplane nodes. Default: `1` (string)
- `max_unavailable_worker` - (Optional) RKE max unavailable worker nodes. Default: `10%` (string)

» `drain_input`

» Arguments

- `delete_local_data` - Delete RKE node local data. Default: `false` (bool)
- `force` - Force RKE node drain. Default: `false` (bool)
- `grace_period` - RKE node drain grace period. Default: `-1` (int)
- `ignore_daemon_sets` - Ignore RKE daemon sets. Default: `true` (bool)
- `timeout` - RKE node drain timeout. Default: `60` (int)

» `k3s_config`

» Arguments

The following arguments are supported:

- `upgrade_strategy` - (Optional/Computed) K3S upgrade strategy (List maxitems: 1)
- `version` - (Optional/Computed) K3S kubernetes version (string)

» `upgrade_strategy`

» Arguments

- `drain_server_nodes` - (Optional) Drain server nodes. Default: `false` (bool)
- `drain_worker_nodes` - (Optional) Drain worker nodes. Default: `false` (bool)
- `server_concurrency` - (Optional) Server concurrency. Default: `1` (int)
- `worker_concurrency` - (Optional) Worker concurrency. Default: `1` (int)

» `aks_config`

» Arguments

The following arguments are supported:

- `agent_dns_prefix` - (Required) DNS prefix to be used to create the FQDN for the agent pool (string)
- `client_id` - (Required/Sensitive) Azure client ID to use (string)
- `client_secret` - (Required/Sensitive) Azure client secret associated with the \"client id\" (string)
- `kubernetes_version` - (Required) Specify the version of Kubernetes. To check available versions exec `az aks get-versions -l eastus -o table` (string)
- `master_dns_prefix` - (Required) DNS prefix to use the Kubernetes cluster control pane (string)
- `resource_group` - (Required) The name of the Cluster resource group (string)
- `ssh_public_key_contents` - (Required) Contents of the SSH public key used to authenticate with Linux hosts (string)
- `subnet` - (Required) The name of an existing Azure Virtual Subnet. Composite of agent virtual network subnet ID (string)
- `subscription_id` - (Required) Subscription credentials which uniquely identify Microsoft Azure subscription (string)
- `tenant_id` - (Required) Azure tenant ID to use (string)

- `virtual_network` - (Required) The name of an existing Azure Virtual Network. Composite of agent virtual network subnet ID (string)
- `virtual_network_resource_group` - (Required) The resource group of an existing Azure Virtual Network. Composite of agent virtual network subnet ID (string)
- `add_client_app_id` - (Optional/Sensitive) The ID of an Azure Active Directory client application of type \"Native\". This application is for user login via kubectl (string)
- `add_server_app_id` - (Optional/Sensitive) The ID of an Azure Active Directory server application of type \"Web app/API\". This application represents the managed cluster's apiserver (Server application) (string)
- `aad_server_app_secret` - (Optional/Sensitive) The secret of an Azure Active Directory server application (string)
- `aad_tenant_id` - (Optional/Sensitive) The ID of an Azure Active Directory tenant (string)
- `admin_username` - (Optional) The administrator username to use for Linux hosts. Default `azureuser` (string)
- `agent_os_disk_size` - (Optional) GB size to be used to specify the disk for every machine in the agent pool. If you specify 0, it will apply the default according to the \"agent vm size\" specified. Default `0` (int)
- `agent_pool_name` - (Optional) Name for the agent pool, upto 12 alphanumeric characters. Default `agentpool0` (string)
- `agent_storage_profile` - (Optional) Storage profile specifies what kind of storage used on machine in the agent pool. Chooses from [ManagedDisks StorageAccount]. Default `ManagedDisks` (string)
- `agent_vm_size` - (Optional) Size of machine in the agent pool. Default `Standard_D1_v2` (string)
- `auth_base_url` - (Optional) Different authentication API url to use. Default `https://login.microsoftonline.com/` (string)
- `base_url` - (Optional) Different resource management API url to use. Default `https://management.azure.com/` (string)
- `count` - (Optional) Number of machines (VMs) in the agent pool. Allowed values must be in the range of 1 to 100 (inclusive). Default `1` (int)
- `dns_service_ip` - (Optional) An IP address assigned to the Kubernetes DNS service. It must be within the Kubernetes Service address range specified in \"service cidr\". Default `10.0.0.10` (string)
- `docker_bridge_cidr` - (Required) A CIDR notation IP range assigned to the Docker bridge network. It must not overlap with any Subnet IP ranges or the Kubernetes Service address range specified in \"service cidr\". Default `172.17.0.1/16` (string)
- `enable_http_application_routing` - (Optional) Enable the Kubernetes ingress with automatic public DNS name creation. Default `false` (bool)
- `enable_monitoring` - (Optional) Turn on Azure Log Analytics monitoring. Uses the Log Analytics \"Default\" workspace if it exists, else creates one. if using an existing workspace, specifies \"log analytics workspace resource id\". Default `true` (bool)

- `location` - (Optional) Azure Kubernetes cluster location. Default `eastus` (string)
- `log_analytics_workspace` - (Optional) The name of an existing Azure Log Analytics Workspace to use for storing monitoring data. If not specified, uses '{resource group}-{subscription id}-{location code}' (string)
- `log_analytics_workspace_resource_group` - (Optional) The resource group of an existing Azure Log Analytics Workspace to use for storing monitoring data. If not specified, uses the 'Cluster' resource group (string)
- `max_pods` - (Optional) Maximum number of pods that can run on a node. Default `110` (int)
- `network_plugin` - (Optional) Network plugin used for building Kubernetes network. Chooses from `azure` or `kubenet`. Default `azure` (string)
- `network_policy` - (Optional) Network policy used for building Kubernetes network. Chooses from `calico` (string)
- `pod_cidr` - (Optional) A CIDR notation IP range from which to assign Kubernetes Pod IPs when \"network plugin\" is specified in \"kubenet\". Default `172.244.0.0/16` (string)
- `service_cidr` - (Optional) A CIDR notation IP range from which to assign Kubernetes Service cluster IPs. It must not overlap with any Subnet IP ranges. Default `10.0.0.0/16` (string)
- `tag` - (Optional/Computed) Tags for Kubernetes cluster. For example, foo=bar (map)

» `eks_config`

» Arguments

The following arguments are supported:

- `access_key` - (Required/Sensitive) The AWS Client ID to use (string)
- `kubernetes_version` - (Required) The Kubernetes master version (string)
- `secret_key` - (Required/Sensitive) The AWS Client Secret associated with the Client ID (string)
- `ami` - (Optional) AMI ID to use for the worker nodes instead of the default (string)
- `associate_worker_node_public_ip` - (Optional) Associate public ip EKS worker nodes. Default `true` (bool)
- `desired_nodes` - (Optional) The desired number of worker nodes. Just for Rancher v2.3.x and above. Default `3` (int)
- `instance_type` - (Optional) The type of machine to use for worker nodes. Default `t2.medium` (string)
- `key_pair_name` - (Optional) Allow user to specify key name to use. Just for Rancher v2.2.7 and above (string)
- `maximum_nodes` - (Optional) The maximum number of worker nodes. Default `3` (int)

- `minimum_nodes` - (Optional) The minimum number of worker nodes. Default `1` (int)
- `node_volume_size` - (Optional) The volume size for each node. Default `20` (int)
- `region` - (Optional) The AWS Region to create the EKS cluster in. Default `us-west-2` (string)
- `security_groups` - (Optional) List of security groups to use for the cluster. If it's not specified Rancher will create a new security group (list)
- `service_role` - (Optional) The service role to use to perform the cluster operations in AWS. If it's not specified Rancher will create a new service role (string)
- `session_token` - (Optional/Sensitive) A session token to use with the client key and secret if applicable (string)
- `subnets` - (Optional) List of subnets in the virtual network to use. If it's not specified Rancher will create 3 news subnets (list)
- `user_data` - (Optional/Computed) Pass user-data to the nodes to perform automated configuration tasks (string)
- `virtual_network` - (Optional) The name of the virtual network to use. If it's not specified Rancher will create a new VPC (string)

» `gke_config`

» Arguments

The following arguments are supported:

- `cluster_ipv4_cidr` - (Required) The IP address range of the container pods (string)
- `credential` - (Required/Sensitive) The contents of the GC credential file (string)
- `disk_type` - (Required) Type of the disk attached to each node (string)
- `image_type` - (Required) The image to use for the worker nodes (string)
- `ip_policy_cluster_ipv4_cidr_block` - (Required) The IP address range for the cluster pod IPs (string)
- `ip_policy_cluster_secondary_range_name` - (Required) The name of the secondary range to be used for the cluster CIDR block (string)
- `ip_policy_node_ipv4_cidr_block` - (Required) The IP address range of the instance IPs in this cluster (string)
- `ip_policy_services_ipv4_cidr_block` - (Required) The IP address range of the services IPs in this cluster (string)
- `ip_policy_services_secondary_range_name` - (Required) The name of the secondary range to be used for the services CIDR block (string)
- `ip_policy_subnetwork_name` - (Required) A custom subnetwork name to be used if createSubnetwork is true (string)
- `locations` - (Required) Locations for GKE cluster (list)
- `machine_type` - (Required) Machine type for GKE cluster (string)

- `maintenance_window` - (Required) Maintenance window for GKE cluster (string)
- `master_ipv4_cidr_block` - (Required) The IP range in CIDR notation to use for the hosted master network (string)
- `master_version` - (Required) Master version for GKE cluster (string)
- `network` - (Required) Network for GKE cluster (string)
- `node_pool` - (Required) The ID of the cluster node pool (string)
- `node_version` - (Required) Node version for GKE cluster (string)
- `oauth_scopes` - (Required) The set of Google API scopes to be made available on all of the node VMs under the default service account (list)
- `project_id` - (Required) Project ID for GKE cluster (string)
- `service_account` - (Required) The Google Cloud Platform Service Account to be used by the node VMs (string)
- `sub_network` - (Required) Subnetwork for GKE cluster (string)
- `description` - (Optional) An optional description of this cluster (string)
- `disk_size_gb` - (Optional) Size of the disk attached to each node. Default `100` (int)
- `enable_alpha_feature` - (Optional) To enable Kubernetes alpha feature. Default `true` (bool)
- `enable_auto_repair` - (Optional) Specifies whether the node auto-repair is enabled for the node pool. Default `false` (bool)
- `enable_auto_upgrade` - (Optional) Specifies whether node auto-upgrade is enabled for the node pool. Default `false` (bool)
- `enable_horizontal_pod_autoscaling` - (Optional) Enable horizontal pod autoscaling for the cluster. Default `true` (bool)
- `enable_http_load_balancing` - (Optional) Enable HTTP load balancing on GKE cluster. Default `true` (bool)
- `enable_kubernetes_dashboard` - (Optional) Whether to enable the Kubernetes dashboard. Default `false` (bool)
- `enable_legacy_abac` - (Optional) Whether to enable legacy abac on the cluster. Default `false` (bool)
- `enable_network_policy_config` - (Optional) Enable stackdriver logging. Default `true` (bool)
- `enable_nodepool_autoscaling` - (Optional) Enable nodepool autoscaling. Default `false` (bool)
- `enable_private_endpoint` - (Optional) Whether the master's internal IP address is used as the cluster endpoint. Default `false` (bool)
- `enable_private_nodes` - (Optional) Whether nodes have internal IP address only. Default `false` (bool)
- `enable_stackdriver_logging` - (Optional) Enable stackdriver monitoring. Default `true` (bool)
- `enable_stackdriver_monitoring` - (Optional) Enable stackdriver monitoring on GKE cluster (bool)
- `ip_policy_create_subnetwork` - (Optional) Whether a new subnetwork will be created automatically for the cluster. Default `false` (bool)
- `issue_client_certificate` - (Optional) Issue a client certificate. De-

fault `false` (bool)

- `kubernetes_dashboard` - (Optional) Enable the Kubernetes dashboard. Default `false` (bool)
- `labels` - (Optional/Computed) The map of Kubernetes labels to be applied to each node (map)
- `local_ssd_count` - (Optional) The number of local SSD disks to be attached to the node. Default `0` (int)
- `master_authorized_network_cidr_blocks` - (Optional) Define up to 10 external networks that could access Kubernetes master through HTTPS (list)
- `max_node_count` - (Optional) Maximum number of nodes in the NodePool. Must be >= minNodeCount. There has to enough quota to scale up the cluster. Default `0` (int)
- `min_node_count` - (Optional) Minimmum number of nodes in the NodePool. Must be >= 1 and <= maxNodeCount. Default `0` (int)
- `node_count` - (Optional) Node count for GKE cluster. Default `3` (int)
- `preemptible` - (Optional) Whether the nodes are created as preemptible VM instances. Default `false` (bool)
- `resource_labels` - (Optional/Computed) The map of Kubernetes labels to be applied to each cluster (map)
- `use_ip_aliases` - (Optional) Whether alias IPs will be used for pod IPs in the cluster. Default `false` (bool)
- `taints` - (Required) List of Kubernetes taints to be applied to each node (list)
- `zone` - (Required) Zone GKE cluster (string)

» `cluster_auth_endpoint`

» Arguments

- `ca_certs` - (Optional) CA certs for the authorized cluster endpoint (string)
- `enabled` - (Optional) Enable the authorized cluster endpoint. Default `true` (bool)
- `fqdn` - (Optional) FQDN for the authorized cluster endpoint (string)

» `cluster_monitoring_input`

» Arguments

- `answers` - (Optional/Computed) Key/value answers for monitor input (map)

» `cluster_template_answers`

» **Arguments**

- `cluster_id` - (Optional) Cluster ID to apply answer (string)
- `project_id` - (Optional) Project ID to apply answer (string)
- `values` - (Optional) Key/values for answer (map)

» `cluster_template_questions`

» **Arguments**

- `default` - (Required) Default variable value (string)
- `required` - (Optional) Required variable. Default `false` (bool)
- `type` - (Optional) Variable type. `boolean`, `int` and `string` are allowed. Default `string` (string)
- `variable` - (Optional) Variable name (string)

» `cluster_registration_token`

» **Attributes**

- `cluster_id` - (Computed) Cluster ID (string)
- `name` - (Computed) Name of cluster registration token (string)
- `command` - (Computed) Command to execute in a imported k8s cluster (string)
- `insecure_command` - (Computed) Insecure command to execute in a imported k8s cluster (string)
- `manifest_url` - (Computed) K8s manifest url to execute with `kubectl` to import an existing k8s cluster (string)
- `node_command` - (Computed) Node command to execute in linux nodes for custom k8s cluster (string)
- `token` - (Computed) Token for cluster registration token object (string)
- `windows_node_command` - (Computed) Node command to execute in windows nodes for custom k8s cluster (string)
- `annotations` - (Computed) Annotations for cluster registration token object (map)
- `labels` - (Computed) Labels for cluster registration token object (map)

» `scheduled_cluster_scan`

» **Arguments**

- `scan_config` - (Required) Cluster scan config (List maxitems:1)

- `schedule_config` - (Required) Cluster scan schedule config (list maxitems:1)
- `enabled` - (Optional) Enable scheduled cluster scan. Default: `false` (bool)

» `scan_config`

» **Arguments**

- `cis_scan_config` - (Optional/computed) Cluster Cis Scan config (List maxitems:1)

» `cis_scan_config`

» Arguments

- `debug_master` - (Optional) Debug master. Default: `false` (bool)
- `debug_worker` - (Optional) Debug worker. Default: `false` (bool)
- `override_benchmark_version` - (Optional) Override benchmark version (string)
- `override_skip` - (Optional) Override skip (string)
- `profile` - (Optional) Cis scan profile. Allowed values: `"permissive" (default) || "hardened"` (string)

» Timeouts

`rancher2_cluster` provides the following Timeouts configuration options:

- `create` - (Default `30 minutes`) Used for creating clusters.
- `update` - (Default `30 minutes`) Used for cluster modifications.
- `delete` - (Default `30 minutes`) Used for deleting clusters.

» Import

Clusters can be imported using the Rancher Cluster ID

```
$ terraform import rancher2_cluster.foo <cluster>
```

» rancher2_cluster_alert_group

Provides a Rancher v2 Cluster Alert Group resource. This can be used to create Cluster Alert Group for Rancher v2 environments and retrieve their information.

» Example Usage

```
# Create a new Rancher2 Cluster Alert Group
resource "rancher2_cluster_alert_group" "foo" {
  cluster_id = "<cluster_id>"
  name = "foo"
  description = "Terraform cluster alert group"
  group_interval_seconds = 300
  repeat_interval_seconds = 3600
}
```

» Argument Reference

The following arguments are supported:

- `cluster_id` - (Required) The cluster id where create cluster alert group (string)
- `name` - (Required) The cluster alert group name (string)
- `description` - (Optional) The cluster alert group description (string)
- `group_interval_seconds` - (Optional) The cluster alert group interval seconds. Default: `180` (int)
- `group_wait_seconds` - (Optional) The cluster alert group wait seconds. Default: `180` (int)
- `recipients` - (Optional) The cluster alert group recipients (list)
- `repeat_interval_seconds` - (Optional) The cluster alert group wait seconds. Default: `3600` (int)
- `annotations` - (Optional/Computed) The cluster alert group annotations (map)
- `labels` - (Optional/Computed) The cluster alert group labels (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Nested blocks

» `recipients`

» Arguments

- `notifier_id` - (Required) Recipient notifier ID (string)
- `recipient` - (Optional/Computed) Recipient (string)

- `default_recipient` - (Optional) Use notifier default recipient, overriding `recipient` argument if set. Default: `false` (bool)

» Attributes

- `notifier_type` - (Computed) Recipient notifier ID. Supported values : `"pagerduty" | "slack" | "email" | "webhook" | "wechat"` (string)

» Timeouts

`rancher2_cluster_alert_group` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating cluster alert groups.
- `update` - (Default `10 minutes`) Used for cluster alert group modifications.
- `delete` - (Default `10 minutes`) Used for deleting cluster alert groups.

» Import

Cluster Alert Group can be imported using the Rancher cluster alert group ID

```
$ terraform import rancher2_cluster_alert_group.foo <rancher2_cluster_alert_group_id>
```

» rancher2_cluster_alert_rule

Provides a Rancher v2 Cluster Alert Rule resource. This can be used to create Cluster Alert Rule for Rancher v2 environments and retrieve their information.

» Example Usage

```
# Create a new Rancher2 Cluster Alert Group
resource "rancher2_cluster_alert_group" "foo" {
  cluster_id = "<cluster_id>"
  name = "foo"
  description = "Terraform cluster alert group"
  group_interval_seconds = 300
  repeat_interval_seconds = 3600
}
# Create a new Rancher2 Cluster Alert Rule
resource "rancher2_cluster_alert_rule" "foo" {
  cluster_id = "${rancher2_cluster_alert_group.foo.cluster_id}"
  group_id = "${rancher2_cluster_alert_group.foo.id}"
```

```
  name = "foo"
  group_interval_seconds = 600
  repeat_interval_seconds = 6000
}
```

» Argument Reference

The following arguments are supported:

- `cluster_id` - (Required) The cluster id where create cluster alert rule (string)
- `group_id` - (Required) The cluster alert rule alert group ID (string)
- `name` - (Required) The cluster alert rule name (string)
- `event_rule` - (Optional) The cluster alert rule event rule. ConflictsWith: `"metric_rule", "node_rule", "system_service_rule"`‘ (list Maxitems:1)
- `group_interval_seconds` - (Optional) The cluster alert rule group interval seconds. Default: `180` (int)
- `group_wait_seconds` - (Optional) The cluster alert rule group wait seconds. Default: `180` (int)
- `inherited` - (Optional) The cluster alert rule inherited. Default: `true` (bool)
- `metric_rule` - (Optional) The cluster alert rule metric rule. ConflictsWith: `"event_rule", "node_rule", "system_service_rule"`‘ (list Maxitems:1)
- `node_rule` - (Optional) The cluster alert rule node rule. ConflictsWith: `"event_rule", "metric_rule", "system_service_rule"`‘ (list Maxitems:1)
- `repeat_interval_seconds` - (Optional) The cluster alert rule wait seconds. Default: `3600` (int)
- `severity` - (Optional) The cluster alert rule severity. Supported values : `"critical" | "info" | "warning"`. Default: `critical` (string)
- `system_service_rule` - (Optional) The cluster alert rule system service rule. ConflictsWith: `"event_rule", "metric_rule", "node_rule"`‘ (list Maxitems:1)
- `annotations` - (Optional/Computed) The cluster alert rule annotations (map)
- `labels` - (Optional/Computed) The cluster alert rule labels (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Nested blocks

» `event_rule`

» Arguments

- `resource_kind` - (Required) Resource kind. Supported values : `"DaemonSet" | "Deployment" | "Node" | "Pod" | "StatefulSet"` (string)
- `event_type` - (Optional) Event type. Supported values : `"Warning" | "Normal"`. Default: `Warning` (string)

» `metric_rule`

» Arguments

- `duration` - (Required) Metric rule duration (string)
- `expression` - (Required) Metric rule expression (string)
- `threshold_value` - (Required) Metric rule threshold value (float64)
- `comparison` - (Optional) Metric rule comparison. Supported values : `"equal" | "greater-or-equal" | "greater-than" | "less-or-equal" | "less-than" | "not-equal" | "has-value"`. Default: `equal` (string)
- `description` - (Optional) Metric rule description (string)

» `node_rule`

» Arguments

- `cpu_threshold` - (Optional) Node rule cpu threshold. Default: `70` (int)
- `condition` - (Optional) Node rule condition. Supported values : `"cpu" | "mem" | "notready"`. Default: `notready` (string)
- `mem_threshold` - (Optional) Node rule mem threshold. Default: `70` (int)
- `node_id` - (Optional) Node ID (string)
- `selector` - (Optional) Node rule selector (map)

» `system_service_rule`

» Arguments

- `condition` - (Optional) System service rule condition. Supported values : `"controller-manager" | "etcd" | "scheduler"`. Default: `scheduler` (string)

» Timeouts

`rancher2_cluster_alert_rule` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating cluster alert rules.
- `update` - (Default `10 minutes`) Used for cluster alert rule modifications.
- `delete` - (Default `10 minutes`) Used for deleting cluster alert rules.

» Import

Cluster Alert Rule can be imported using the Rancher cluster alert rule ID

```
$ terraform import rancher2_cluster_alert_rule.foo <rancher2_cluster_alert_rule_id>
```

» rancher2_cluster_driver

Provides a Rancher v2 Cluster Driver resource. This can be used to create Cluster Driver for Rancher v2.2.x Kontainer Engine clusters and retrieve their information.

» Example Usage

```
# Create a new Rancher2 Cluster Driver
resource "rancher2_cluster_driver" "foo" {
    active = true
    builtin = false
    checksum = "0x0"
    description = "Foo description"
    external_id = "foo_external"
    name = "foo"
    ui_url = "local://ui"
    url = "local://"
    whitelist_domains = ["*.foo.com"]
}
```

» Argument Reference

The following arguments are supported:

- `active` - (Required) Specify the cluster driver state (bool)
- `builtin` - (Required) Specify whether the cluster driver is an internal cluster driver or not (bool)

- `name` - (Required) Name of the cluster driver (string)
- `url` - (Required) The URL to download the machine driver binary for 64-bit Linux (string)
- `actual_url` - (Optional) Actual url of the cluster driver (string)
- `checksum` - (Optional) Verify that the downloaded driver matches the expected checksum (string)
- `ui_url` - (Optional) The URL to load for customized Add Clusters screen for this driver (string)
- `whitelist_domains` - (Optional) Domains to whitelist for the ui (list)
- `annotations` - (Optional/Computed) Annotations of the resource (map)
- `labels` - (Optional/Computed) Labels of the resource (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Timeouts

`rancher2_cluster_driver` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating cluster drivers.
- `update` - (Default `10 minutes`) Used for cluster driver modifications.
- `delete` - (Default `10 minutes`) Used for deleting cluster drivers.

» Import

Cluster Driver can be imported using the Rancher Cluster Driver ID

```
$ terraform import rancher2_cluster_driver.foo <cluster_driver_id>
```

» rancher2_cluster_logging

Provides a Rancher v2 Cluster Logging resource. This can be used to configure Cluster Logging for Rancher v2 environments and retrieve their information.

» Example Usage

```
# Create a new Rancher2 Cluster Logging
resource "rancher2_cluster_logging" "foo" {
```

```
  name = "foo"
  cluster_id = "<cluster_id>"
  kind = "syslog"
  syslog_config {
    endpoint = "<syslog_endpoint>"
    protocol = "udp"
    severity = "notice"
    ssl_verify = false
  }
}
```

» Argument Reference

The following arguments are supported:

- `cluster_id` - (Required) The cluster id to configure logging (string)
- `name` - (Required) The name of the cluster logging config (string)
- `kind` - (Required) The kind of the Cluster Logging. `elasticsearch`, `fluentd`, `kafka`, `splunk` and `syslog` are supported (string)
- `custom_target_config` - (Optional) The custom target config for Cluster Logging. For `kind = custom`. Conflicts with `elasticsearch_config`, `fluentd_config`, `kafka_config`, `splunk_config` and `syslog_config` (list maxitems:1)
- `enable_json_parsing` - (Optional) Enable json log parsing. Default: `false` (bool)
- `elasticsearch_config` - (Optional) The elasticsearch config for Cluster Logging. For `kind = elasticsearch`. Conflicts with `custom_target_config`, `fluentd_config`, `kafka_config`, `splunk_config` and `syslog_config` (list maxitems:1)
- `fluentd_config` - (Optional) The fluentd config for Cluster Logging. For `kind = fluentd`. Conflicts with `custom_target_config`, `elasticsearch_config`, `kafka_config`, `splunk_config` and `syslog_config` (list maxitems:1)
- `kafka_config` - (Optional) The kafka config for Cluster Logging. For `kind = kafka`. Conflicts with `custom_target_config`, `elasticsearch_config`, `fluentd_config`, `splunk_config` and `syslog_config` (list maxitems:1)
- `namespace_id` - (Optional) The namespace id from cluster logging (string)
- `output_flush_interval` - (Optional) How often buffered logs would be flushed. Default: `3` seconds (int)
- `output_tags` - (Optional/computed) The output tags for Cluster Logging (map)
- `splunk_config` - (Optional) The splunk config for Cluster Logging. For `kind = splunk`. Conflicts with `custom_target_config`, `elasticsearch_config`, `fluentd_config`, `kafka_config`, and

`syslog_config` (list maxitems:1)

- `syslog_config` - (Optional) The syslog config for Cluster Logging. For `kind = syslog`. Conflicts with `custom_target_config`, `elasticsearch_config`, `fluentd_config`, `kafka_config`, and `splunk_config` (list maxitems:1)
- `annotations` - (Optional/Computed) Annotations for Cluster Logging object (map)
- `labels` - (Optional/Computed) Labels for Cluster Logging object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Nested blocks

» `custom_target_config`

» Arguments

- `content` - (Required) Custom target config content (string)
- `certificate` - (Required/Sensitive) SSL CA certificate for the custom target service (string)
- `client_cert` - (Optional/Sensitive) SSL client certificate for the custom target service (string)
- `client_key` - (Optional/Sensitive) SSL client key for the custom target service (string)

» `elasticsearch_config`

» Arguments

- `endpoint` - (Required) Endpoint of the elascticsearch service. Must include protocol, `http://` or `https://` (string)
- `auth_password` - (Optional/Sensitive) User password for the elascticsearch service (string)
- `auth_username` - (Optional/Sensitive) Username for the elascticsearch service (string)
- `certificate` - (Optional/Sensitive) SSL certificate for the elascticsearch service (string)
- `client_cert` - (Optional/Sensitive) SSL client certificate for the elascticsearch service (string)
- `client_key` - (Optional/Sensitive) SSL client key for the elascticsearch service (string)

- `client_key_pass` - (Optional/Sensitive) SSL client key password for the elascticsearch service (string)
- `date_format` - (Optional) Date format for the elascticsearch logs. Default: `YYYY-MM-DD` (string)
- `index_prefix` - (Optional) Index prefix for the elascticsearch logs. Default: `local` (string)
- `ssl_verify` - (Optional) SSL verify for the elascticsearch service (bool)
- `ssl_version` - (Optional) SSL version for the elascticsearch service (string)

» `fluentd_config`

» Arguments

- `fluent_servers` - (Required) Servers for the fluentd service (list)
- `certificate` - (Optional/Sensitive) SSL certificate for the fluentd service (string)
- `compress` - (Optional) Compress data for the fluentd service (bool)
- `enable_tls` - (Optional) Enable TLS for the fluentd service (bool)

» `fluent_servers`

» Arguments

- `endpoint` - (Required) Endpoint of the fluentd service (string)
- `hostname` - (Optional) Hostname of the fluentd service (string)
- `password` - (Optional/Sensitive) User password of the fluentd service (string)
- `shared_key` - (Optional/Sensitive) Shared key of the fluentd service (string)
- `standby` - (Optional) Standby server of the fluentd service (bool)
- `username` - (Optional/Sensitive) Username of the fluentd service (string)
- `weight` - (Optional) Weight of the fluentd server (int)

» `kafka_config`

» Arguments

- `topic` - (Required) Topic to publish on the kafka service (string)
- `broker_endpoints` - (Optional) Kafka endpoints for kafka service. Conflicts with `zookeeper_endpoint` (list)
- `certificate` - (Optional/Sensitive) SSL certificate for the kafka service (string)

- `client_cert` - (Optional/Sensitive) SSL client certificate for the kafka service (string)
- `client_key` - (Optional/Sensitive) SSL client key for the kafka service (string)
- `zookeeper_endpoint` - (Optional) Zookeeper endpoint for kafka service. Conflicts with `broker_endpoints` (string)

» `splunk_config`

» Arguments

- `endpoint` - (Required) Endpoint of the splunk service. Must include protocol, `http://` or `https://` (string)
- `token` - (Required/Sensitive) Token for the splunk service (string)
- `certificate` - (Optional/Sensitive) SSL certificate for the splunk service (string)
- `client_cert` - (Optional/Sensitive) SSL client certificate for the splunk service (string)
- `client_key` - (Optional/Sensitive) SSL client key for the splunk service (string)
- `client_key_pass` - (Optional/Sensitive) SSL client key password for the splunk service (string)
- `index` - (Optional) Index prefix for the splunk logs (string)
- `source` - (Optional) Date format for the splunk logs (string)
- `ssl_verify` - (Optional) SSL verify for the splunk service (bool)

» `syslog_config`

» Arguments

- `endpoint` - (Required) Endpoint of the syslog service (string)
- `certificate` - (Optional/Sensitive) SSL certificate for the syslog service (string)
- `client_cert` - (Optional/Sensitive) SSL client certificate for the syslog service (string)
- `client_key` - (Optional/Sensitive) SSL client key for the syslog service (string)
- `program` - (Optional) Program for the syslog service (string)
- `protocol` - (Optional) Protocol for the syslog service. `tcp` and `udp` are supported. Default: `udp` (string)
- `severity` - (Optional) Date format for the syslog logs. `emergency`, `alert`, `critical`, `error`, `warning`, `notice`, `info` and `debug` are supported. Default: `notice` (string)
- `ssl_verify` - (Optional) SSL verify for the syslog service (bool)
- `token` - (Optional/Sensitive) Token for the syslog service (string)

» Timeouts

`rancher2_cluster_logging` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating cluster logging configurations.
- `update` - (Default `10 minutes`) Used for cluster logging configuration modifications.
- `delete` - (Default `10 minutes`) Used for deleting cluster logging configurations.

» Import

Cluster Logging can be imported using the Rancher Cluster Logging ID

```
$ terraform import rancher2_cluster_logging.foo <cluster_logging_id>
```

» rancher2_cluster_role_template_binding

Provides a Rancher v2 Cluster Role Template Binding resource. This can be used to create Cluster Role Template Bindings for Rancher v2 environments and retrieve their information.

» Example Usage

```
# Create a new Rancher2 Cluster Role Template Binding
resource "rancher2_cluster_role_template_binding" "foo" {
  name = "foo"
  cluster_id = "<cluster_id>"
  role_template_id = "<role_template_id>"
  user_id = "<user_id>"
}
```

» Argument Reference

The following arguments are supported:

- `cluster_id` - (Required) The cluster id where bind cluster role template binding (string)
- `role_template_id` - (Required) The role template id from create cluster role template binding (string)
- `name` - (Required) The name of the cluster role template binding (string)

- `group_id` - (Optional) The group ID to assign cluster role template binding (string)
- `group_principal_id` - (Optional) The group_principal ID to assign cluster role template binding (string)
- `user_id` - (Optional) The user ID to assign cluster role template binding (string)
- `user_principal_id` - (Optional) The user_principal ID to assign cluster role template binding (string)
- `annotations` - (Optional/Computed) Annotations for cluster role template binding (map)
- `labels` - (Optional/Computed) Labels for cluster role template binding (map)

Note user `user_id | user_principal_id` OR group `group_id | group_principal_id` must be defined

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Timeouts

`rancher2_cluster_role_template_binding` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating cluster role template bindings.
- `update` - (Default `10 minutes`) Used for cluster role template binding modifications.
- `delete` - (Default `10 minutes`) Used for deleting cluster role template bindings.

» Import

Cluster Role Template Bindings can be imported using the Rancher cluster Role Template Binding ID

```
$ terraform import rancher2_cluster_role_template_binding.foo <cluster_role_template_binding
```

» rancher2_cluster_sync

Provides a Rancher v2 Cluster Sync dummy resource. This can be used to create a Cluster Sync to wait for a Rancher v2 Cluster resource `active` state.

This dummy resource doesn't create anything at Rancher side. It's used to sync terraform resources that depends of Rancher v2 Cluster resource in `active` state. This resource will wait until `cluster_id` is `active` on `terraform apply`. It also helps to sync `terraform destroy` dependencies, specially useful if cluster is using node pools.

This resource will also compute attributes with useful cluster related data (see Attributes Reference section).

» Example Usage

```
# Create a new rancher2 rke Cluster
resource "rancher2_cluster" "foo-custom" {
  name = "foo-custom"
  description = "Foo rancher2 custom cluster"
  rke_config {
    network {
      plugin = "canal"
    }
  }
}
# Create a new rancher2 Node Template
resource "rancher2_node_template" "foo" {
  name = "foo"
  description = "foo test"
  amazonec2_config {
    access_key = "AWS_ACCESS_KEY"
    secret_key = "<AWS_SECRET_KEY>"
    ami =  "<AMI_ID>"
    region = "<REGION>"
    security_group = ["<AWS_SECURITY_GROUP>"]
    subnet_id = "<SUBNET_ID>"
    vpc_id = "<VPC_ID>"
    zone = "<ZONE>"
  }
}
# Create a new rancher2 Node Pool
resource "rancher2_node_pool" "foo" {
  cluster_id =  "${rancher2_cluster.foo-custom.id}"
  name = "foo"
```

```
  hostname_prefix =  "foo-cluster-0"
  node_template_id = "${rancher2_node_template.foo.id}"
  quantity = 3
  control_plane = true
  etcd = true
  worker = true
}
# Create a new rancher2 Cluster Sync
resource "rancher2_cluster_sync" "foo-custom" {
  cluster_id =  "${rancher2_cluster.foo-custom.id}"
  node_pool_ids = ["${rancher2_node_pool.foo.id}"]
}
# Create a new rancher2 Project
resource "rancher2_project" "foo" {
  name = "foo"
  cluster_id = "${rancher2_cluster_sync.foo-custom.id}"
  description = "Terraform namespace acceptance test"
  resource_quota {
    project_limit {
      limits_cpu = "2000m"
      limits_memory = "2000Mi"
      requests_storage = "2Gi"
    }
    namespace_default_limit {
      limits_cpu = "500m"
      limits_memory = "500Mi"
      requests_storage = "1Gi"
    }
  }
  container_resource_limit {
    limits_cpu = "20m"
    limits_memory = "20Mi"
    requests_cpu = "1m"
    requests_memory = "1Mi"
  }
}
```

» Argument Reference

The following arguments are supported:

- `cluster_id` - (Required/ForceNew) The cluster ID that is syncing (string)
- `node_pool_ids` - (Optional) The node pool IDs used by the cluster id (list)
- `wait_monitoring` - (Optional) Wait until monitoring is up and running.

Default: `false` (bool)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource. Same as `cluster_id` (string)
- `default_project_id` - (Computed) Default project ID for the cluster sync (string)
- `kube_config` - (Computed/Sensitive) Kube Config generated for the cluster sync (string)
- `system_project_id` - (Computed) System project ID for the cluster sync (string)

» Timeouts

`rancher2_cluster_sync` provides the following Timeouts configuration options:

- `create` - (Default `30 minutes`) Used for creating cluster sync.
- `update` - (Default `30 minutes`) Used for cluster sync modifications.
- `delete` - (Default `30 minutes`) Used for deleting cluster sync.

» rancher2_cluster_template

Provides a Rancher v2 Cluster Template resource. This can be used to create Cluster Templates for Rancher v2 RKE clusters and retrieve their information.

Cluster Templates are available from Rancher v2.3.x and above.

» Example Usage

```
# Create a new rancher2 Cluster Template
resource "rancher2_cluster_template" "foo" {
  name = "foo"
  members {
    access_type = "owner"
    user_principal_id = "local://user-XXXXX"
  }
  template_revisions {
    name = "V1"
    cluster_config {
      rke_config {
```

```
        network {
          plugin = "canal"
        }
        services {
          etcd {
            creation = "6h"
            retention = "24h"
          }
        }
      }
    }
    default = true
  }
  description = "Terraform cluster template foo"
}
```

Creating Rancher v2 RKE cluster template with upgrade strategy. For Rancher v2.4.x or above.

```
# Create a new rancher2 Cluster Template
resource "rancher2_cluster_template" "foo" {
  name = "foo"
  members {
    access_type = "owner"
    user_principal_id = "local://user-XXXXX"
  }
  template_revisions {
    name = "V1"
    cluster_config {
      rke_config {
        network {
          plugin = "canal"
        }
        services {
          etcd {
            creation = "6h"
            retention = "24h"
          }
        }
        upgrade_strategy {
          drain = true
          max_unavailable_worker = "20%"
        }
      }
    }
    default = true
  }
```

```
  description = "Terraform cluster template foo"
}
```

Creating Rancher v2 RKE cluster template with scheduled cluster scan. For Rancher v2.4.x or above.

```
# Create a new rancher2 Cluster Template
resource "rancher2_cluster_template" "foo" {
  name = "foo"
  members {
    access_type = "owner"
    user_principal_id = "local://user-XXXXX"
  }
  template_revisions {
    name = "V1"
    cluster_config {
      rke_config {
        network {
          plugin = "canal"
        }
        services {
          etcd {
            creation = "6h"
            retention = "24h"
          }
        }
      }
      scheduled_cluster_scan {
        enabled = true
        scan_config {
          cis_scan_config {
            debug_master = true
            debug_worker = true
          }
        }
        schedule_config {
          cron_schedule = "30 * * * *"
          retention = 5
        }
      }
    }
    default = true
  }
  description = "Terraform cluster template foo"
}
```

» Argument Reference

- `name` - (Required) The cluster template name (string)
- `decription` - (Optional) The cluster template description (string)
- `members` - (Optional) Cluster template members (list)
- `template_revisions` - (Optional/Computed) Cluster template revisions (list)
- `annotations` - (Optional/Computed) Annotations for the cluster template (map)
- `labels` - (Optional/Computed) Labels for the cluster template (map)

» Attributes Reference

- `id` - (Computed) The ID of the resource (string)
- `default_revision_id` - (Computed) Default cluster template revision ID (string)

» Nested blocks

» `members`

» Arguments

- `access_type` - (Optional) Member access type. Valid values: `["read-only" | "owner"]` (string)
- `group_principal_id` - (Optional) Member group principal id (string)
- `user_principal_id` - (Optional) Member user principal id (string)

» `template_revisions`

» Arguments

- `name` - (Required) The cluster template revision name (string)
- `cluster_config` - (Optional) Cluster configuration (list maxitem: 1)
- `default` - (Optional) Default cluster template revision. Default `false` (bool)
- `enabled` - (Optional) Enable cluster template revision. Default `true` (bool)
- `questions` - (Optional) Cluster template questions (list)
- `annotations` - (Optional/Computed) Annotations for the cluster template revision (map)
- `labels` - (Optional/Computed) Labels for the cluster template revision (map)

» Attributes

- `id` - (Computed) The cluster template revision ID (string)
- `cluster_template_id` - (Computed) Cluster template ID (string)

» `cluster_config`

» Arguments

- `cluster_auth_endpoint` - (Optional/Computed) Local cluster auth endpoint (list maxitems: 1)
- `default_cluster_role_for_project_members` - (Optional/Computed) Default cluster role for project members (string)
- `default_pod_security_policy_template_id` - (Optional/Computed) Default pod security policy template ID (string)
- `desired_agent_image` - (Optional/Computed) Desired agent image (string)
- `desired_auth_image` - (Optional/Computed) Desired auth image (string)
- `docker_root_dir` - (Optional/Computed) Desired auth image (string)
- `enable_cluster_alerting` - (Optional) Enable built-in cluster alerting. Default: `false` (bool)
- `enable_cluster_monitoring` - (Optional) Enable built-in cluster monitoring. Default: `false` (bool)
- `enable_network_policy` - (Optional) Enable project network isolation. Default: `false` (bool)
- `rke_config` - (Optional/Computed) Rancher Kubernetes Engine Config (list maxitems: 1)
- `scheduled_cluster_scan`- (Optional) Cluster scheduled cis scan. For Rancher v2.4.0 or above (List MaxItem:1)
- `windows_prefered_cluster` - (Optional) Windows prefered cluster. Default: `false` (bool)

» `questions`

» Arguments

- `default` - (Required) Default variable value (string)
- `required` - (Optional) Required variable. Default `false` (bool)
- `type` - (Optional) Variable type. `boolean`, `int` and `string` are allowed. Default `string` (string)
- `variable` - (Optional) Variable name (string)

» Timeouts

`rancher2_cluster_template` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating cluster templates.
- `update` - (Default `10 minutes`) Used for cluster template modifications.
- `delete` - (Default `10 minutes`) Used for deleting cluster templates.

» Import

Cluster Template can be imported using the rancher Cluster Template ID

```
$ terraform import rancher2_cluster_template.foo <cluster_template_id>
```

» rancher2_etcd_backup

Provides a Rancher v2 Etcd Backup resource. This can be used to create Etcd Backup for Rancher v2.2.x and retrieve their information.

`rancher2_etcd_backup` resource is used to define extra etcd backups for `rancher2_cluster`. The main etcd backup config for the cluster should be set on the cluster config

» Example Usage

```
# Create a new rancher2 Etcd Backup
resource "rancher2_etcd_backup" "foo" {
  backup_config {
    enabled = true
    interval_hours = 20
    retention = 10
    s3_backup_config {
      access_key = "access_key"
      bucket_name = "bucket_name"
      endpoint = "endpoint"
      folder = "/folder"
      region = "region"
      secret_key = "secret_key"
    }
  }
  cluster_id = "<CLUSTER_ID>"
  name = "foo"
  filename = "<FILENAME>"
```

```
}
```

» Argument Reference

The following arguments are supported:

- `cluster_id` - (Required) Cluster ID to config Etcd Backup (string)
- `backup_config` - (Optional/Computed) Backup config for etcd backup (list maxitems:1)
- `filename` - (Optional/Computed) Filename of the Etcd Backup (string)
- `manual` - (Optional) Manual execution of the Etcd Backup. Default `false` (bool)
- `name` - (Required) The name of the Etcd Backup (string)
- `namespace_id` - (Optional/Computed) Description for the Etcd Backup (string)
- `annotations` - (Optional) Annotations for Etcd Backup object (map)
- `labels` - (Optional/Computed) Labels for Etcd Backup object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Nested blocks

» `backup_config`

» Arguments

- `enabled` - (Optional) Enable etcd backup (bool)
- `interval_hours` - (Optional) Interval hours for etcd backup. Default `12` (int)
- `retention` - (Optional) Retention for etcd backup. Default `6` (int)
- `s3_backup_config` - (Optional) S3 config options for etcd backup. Valid for `imported` and `rke` clusters. (list maxitems:1)

» `s3_backup_config`

» Arguments

- `access_key` - (Optional/Sensitive) Access key for S3 service (string)
- `bucket_name` - (Required) Bucket name for S3 service (string)

- `custom_ca` - (Optional) Base64 encoded custom CA for S3 service. Use filebase64() for encoding file. Available from Rancher v2.2.5 (string)
- `endpoint` - (Required) Endpoint for S3 service (string)
- `folder` - (Optional) Folder for S3 service. Available from Rancher v2.2.7 (string)
- `region` - (Optional) Region for S3 service (string)
- `secret_key` - (Optional/Sensitive) Secret key for S3 service (string)

» Timeouts

`rancher2_etcd_backup` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating cloud credentials.
- `update` - (Default `10 minutes`) Used for cloud credential modifications.
- `delete` - (Default `10 minutes`) Used for deleting cloud credentials.

» Import

Etcd Backup can be imported using the Rancher etcd backup ID

```
$ terraform import rancher2_etcd_backup.foo <etcd_backup_id>
```

» rancher2_global_role_binding

Provides a Rancher v2 Global Role Binding resource. This can be used to create Global Role Bindings for Rancher v2 environments and retrieve their information.

» Example Usage

```
# Create a new rancher2 Global Role Binding using user_id
resource "rancher2_global_role_binding" "foo" {
  name = "foo"
  global_role_id = "admin"
  user_id = "user-XXXXX"
}
# Create a new rancher2 Global Role Binding using group_principal_id
resource "rancher2_global_role_binding" "foo2" {
  name = "foo2"
  global_role_id = "admin"
  group_principal_id = "local://g-XXXXX"
}
```

» Argument Reference

The following arguments are supported:

- `global_role_id` - (Required/ForceNew) The role id from create global role binding (string)
- `group_principal_id` - (Optional/Computed/ForceNew) The group principal ID to assign global role binding (only works with external auth providers that support groups). Rancher v2.4.0 or higher is required (string)
- `user_id` - (Optional/Computed/ForceNew) The user ID to assign global role binding (string)
- `name` - (Optional/Computed/ForceNew) The name of the global role binding (string)
- `annotations` - (Optional/Computed) Annotations for global role binding (map)
- `labels` - (Optional/Computed) Labels for global role binding (map)

Note user `user_id` OR group `group_principal_id` must be defined

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Timeouts

`rancher2_global_role_binding` provides the following Timeouts configuration options:

- `create` - (Default `5 minutes`) Used for creating global role bindings.
- `update` - (Default `5 minutes`) Used for global role binding modifications.
- `delete` - (Default `5 minutes`) Used for deleting global role bindings.

» Import

Global Role Bindings can be imported using the Rancher Global Role Binding ID

```
$ terraform import rancher2_global_role_binding.foo <global_role_binding_id>
```

» rancher2_multi_cluster_app

Provides a Rancher v2 multi_cluster_app resource. This can be used to deploy multi_cluster_app on Rancher v2.

This resource can also modify Rancher v2 multi cluster apps in 3 ways: - `Add/Remove targets`: If `targets` arguments is modified, the multi cluster app targets will be updated. - `Rollback`: If `revision_id` argument is provided or modified the app will be rolled back accordingly. A new `revision_id` will be generated in Rancher. It will also generate a non-empty terraform plan that will require manual .tf file intervention. Use carefully. - `Update`: If any other argument is modified the app will be upgraded.

Note: In case of multiple resource modification in a row, `rollback` has preference.

» Example Usage

```
# Create a new rancher2 Multi Cluster App
resource "rancher2_multi_cluster_app" "foo" {
  catalog_name = "<catalog_name>"
  name = "foo"
  targets {
    project_id = "<project_id>"
  }
  template_name = "<template_name>"
  template_version = "<template_version>"
  answers {
    values = {
      "ingress_host" = "test.xip.io"
    }
  }
  roles = ["project-member"]
}

# Create a new rancher2 Multi Cluster App overriding answers
resource "rancher2_multi_cluster_app" "foo" {
  catalog_name = "<catalog_name>"
  name = "foo"
  targets {
    project_id = "<project_id1>"
  }
  targets {
    project_id = "<project_id2>"
  }
  template_name = "<template_name>"
```

```
  template_version = "<template_version>"
  answers {
    values = {
      "ingress_host" = "test.xip.io"
    }
  }
  answers {
    project_id = "<project_id2>"
    values = {
      "ingress_host" = "test2.xip.io"
    }
  }
  roles = ["project-member"]
}
```

» Argument Reference

The following arguments are supported:

- `catalog_name` - (Required) The multi cluster app catalog name (string)
- `name` - (Required/ForceNew) The multi cluster app name (string)
- `roles` - (Required) The multi cluster app roles (list)
- `targets` - (Required) The multi cluster app target projects (list)
- `template_name` - (Required) The multi cluster app template name (string)
- `answers` - (Optional/Computed) The multi cluster app answers (list)
- `members` - (Optional) The multi cluster app answers (list)
- `revision_history_limit` - (Computed) The multi cluster app revision history limit. Default `10` (int)
- `revision_id` - (Optional/Computed) Current revision id for the multi cluster app (string)
- `template_version` - (Optional/Computed) The multi cluster app template version. Default: `latest` (string)
- `upgrade_strategy` - (Optional/Computed) The multi cluster app upgrade strategy (list MaxItems:1)
- `wait` - (Optional) Wait until the multi cluster app is active. Default `true` (bool)
- `annotations` - (Optional/Computed) Annotations for multi cluster app object (map)
- `labels` - (Optional/Computed) Labels for multi cluster app object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

- `template_version_id` - (Computed) The multi cluster app template version ID (string)

» Nested blocks

» `targets`

» Arguments

- `project_id` - (Required) Project ID for target (string)
- `app_id` - (Computed) App ID for target (string)
- `health_state` - (Computed) App health state for target (string)
- `state` - (Computed) App state for target (string)

» `answers`

» Arguments

- `cluster_id` - (Optional) Cluster ID for answer (string)
- `project_id` - (Optional) Project ID for target (string)
- `values` - (Optional) Key/values for answer (map)

» `members`

» Arguments

- `access_type` - (Optional) Member access type. Valid values: `["member" | "owner" | "read-only"]` (string)
- `group_principal_id` - (Optional) Member group principal id (string)
- `user_principal_id` - (Optional) Member user principal id (string)

» `upgrade_strategy`

» Arguments

- `rolling_update` - (Optional) Upgrade strategy rolling update (list MaxItems:1)

» `rolling_update`

» Arguments

- `batch_size` - (Optional) Rolling update batch size. Default `1` (int)
- `interval` - (Optional) Rolling update interval. Default `1` (int)

» Timeouts

`rancher2_app` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating apps.
- `update` - (Default `10 minutes`) Used for app modifications.
- `delete` - (Default `10 minutes`) Used for deleting apps.

» Import

Multi cluster app can be imported using the multi cluster app ID in the format `<multi_cluster_app_name>`

```
$ terraform import rancher2_multi_cluster_app.foo <multi_cluster_app_name>
```

» rancher2_namespace

Provides a Rancher v2 Namespace resource. This can be used to create namespaces for Rancher v2 environments and retrieve their information.

» Example Usage

```
# Create a new rancher2 Namespace
resource "rancher2_namespace" "foo" {
  name = "foo"
  project_id = "<PROJECT_ID>"
  description = "foo namespace"
  resource_quota {
    limit {
      limits_cpu = "100m"
      limits_memory = "100Mi"
      requests_storage = "1Gi"
    }
  }
  container_resource_limit {
    limits_cpu = "20m"
    limits_memory = "20Mi"
    requests_cpu = "1m"
    requests_memory = "1Mi"
  }
}

# Create a new rancher2 Cluster
resource "rancher2_cluster" "foo-custom" {
```

```
  name = "foo-custom"
  description = "Foo rancher2 custom cluster"
  rke_config {
    network {
      plugin = "canal"
    }
  }
}
# Create a new rancher2 Namespace assigned to default cluster project
resource "rancher2_namespace" "foo" {
  name = "foo"
  project_id = "${rancher2_cluster.foo-custom.default_project_id}"
  description = "foo namespace"
  resource_quota {
    limit {
      limits_cpu = "100m"
      limits_memory = "100Mi"
      requests_storage = "1Gi"
    }
  }
  container_resource_limit {
    limits_cpu = "20m"
    limits_memory = "20Mi"
    requests_cpu = "1m"
    requests_memory = "1Mi"
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the namespace (string)
- `project_id` - (Required) The project id where assign namespace. It's on the form `project_id=<cluster_id>:<id>`. Updating `<id>` part on same `<cluster_id>` namespace will be moved between projects (string)
- `container_resource_limit` - (Optional) Default containers resource limits on namespace (List maxitem:1)
- `description` - (Optional) A namespace description (string)
- `resource_quota` - (Optional) Resource quota for namespace. Rancher v2.1.x or higher (list maxitems:1)
- `wait_for_cluster` - (Optional) Wait for cluster becomes active. Default `false` (bool)
- `annotations` - (Optional/Computed) Annotations for Node Pool object (map)

- `labels` - (Optional/Computed) Labels for Node Pool object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Nested blocks

» `container_resource_limit`

» Arguments

- `limits_cpu` - (Optional) CPU limit for containers (string)
- `limits_memory` - (Optional) Memory limit for containers (string)
- `requests_cpu` - (Optional) CPU reservation for containers (string)
- `requests_memory` - (Optional) Memory reservation for containers (string)

» `resource_quota`

» Arguments

- `limit` - (Required) Resource quota limit for namespace (list maxitems:1)

» `limit`

» Arguments

- `config_maps` - (Optional) Limit for config maps in namespace (string)
- `limits_cpu` - (Optional) Limit for limits cpu in namespace (string)
- `limits_memory` - (Optional) Limit for limits memory in namespace (string)
- `persistent_volume_claims` - (Optional) Limit for persistent volume claims in namespace (string)
- `pods` - (Optional) Limit for pods in namespace (string)
- `replication_controllers` - (Optional) Limit for replication controllers in namespace (string)
- `requests_cpu` - (Optional) Limit for requests cpu in namespace (string)
- `requests_memory` - (Optional) Limit for requests memory in namespace (string)
- `requests_storage` - (Optional) Limit for requests storage in namespace (string)
- `secrets` - (Optional) Limit for secrets in namespace (string)

- `services_load_balancers` - (Optional) Limit for services load balancers in namespace (string)
- `services_node_ports` - (Optional) Limit for services node ports in namespace (string)

More info at resource-quotas

» Timeouts

`rancher2_namespace` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating namespaces.
- `update` - (Default `10 minutes`) Used for namespace modifications.
- `delete` - (Default `10 minutes`) Used for deleting namespaces.

» Import

Namespaces can be imported using the namespace ID in the format `<project_id>.<namespace_id>`

```
$ terraform import rancher2_namespace.foo <project_id>.<namespace_id>
```

`<project_id>` is in the format `<cluster_id>:<id>`, but part is optional:

- If full project_id is provided, `<project_id>=<cluster_id>:<id>`, the namespace'll be assigned to corresponding cluster project once it's imported.
- If `<id>` part is omitted `<project_id>=<cluster_id>`, the namespace'll not be assigned to any project. To move it into a project, `<project_id>=<cluster_id>:<id>` needs to be updated in tf file. Namespace movement is only supported inside same `cluster_id`.

» rancher2_node_driver

Provides a Rancher v2 Node Driver resource. This can be used to create Node Driver for Rancher v2 RKE clusters and retrieve their information.

» Example Usage

```
# Create a new rancher2 Node Driver
resource "rancher2_node_driver" "foo" {
    active = true
    builtin = false
    checksum = "0x0"
```

```
    description = "Foo description"
    external_id = "foo_external"
    name = "foo"
    ui_url = "local://ui"
    url = "local://"
    whitelist_domains = ["*.foo.com"]
}
```

» Argument Reference

The following arguments are supported:

- `active` - (Required) Specify if the node driver state (bool)
- `builtin` - (Required) Specify wheter the node driver is an internal node driver or not (bool)
- `name` - (Required) Name of the node driver (string)
- `url` - (Required) The URL to download the machine driver binary for 64-bit Linux (string)
- `checksum` - (Optional) Verify that the downloaded driver matches the expected checksum (string)
- `description` - (Optional) Description of the node driver (string)
- `external_id` - (Optional) External ID (string)
- `ui_url` - (Optional) The URL to load for customized Add Nodes screen for this driver (string)
- `whitelist_domains` - (Optional) Domains to whitelist for the ui (list)
- `annotations` - (Optional/Computed) Annotations of the resource (map)
- `labels` - (Optional/Computed) Labels of the resource (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Timeouts

`rancher2_node_driver` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating node drivers.
- `update` - (Default `10 minutes`) Used for node driver modifications.
- `delete` - (Default `10 minutes`) Used for deleting node drivers.

» Import

Node Driver can be imported using the Rancher Node Driver ID

```
$ terraform import rancher2_node_driver.foo <node_driver_id>
```

» rancher2_node_pool

Provides a Rancher v2 Node Pool resource. This can be used to create Node Pool, using Node template for Rancher v2 RKE clusters and retrieve their information.

» Example Usage

```
# Create a new rancher2 RKE Cluster
resource "rancher2_cluster" "foo-custom" {
  name = "foo-custom"
  description = "Foo rancher2 custom cluster"
  kind = "rke"
  rke_config {
    network {
      plugin = "canal"
    }
  }
}
# Create a new rancher2 Cloud Credential
resource "rancher2_cloud_credential" "foo" {
  name = "foo"
  description= "Terraform cloudCredential acceptance test"
  amazonec2_credential_config {
    access_key = "XXXXXXXXXXXXXXXXXXX"
    secret_key = "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
  }
}
# Create a new rancher2 Node Template
resource "rancher2_node_template" "foo" {
  name = "foo"
  description = "foo test"
  cloud_credential_id = "${rancher2_cloud_credential.foo.id}"
  amazonec2_config {
    ami =  "<AMI_ID>"
    region = "<REGION>"
    security_group = ["<AWS_SECURITY_GROUP>"]
    subnet_id = "<SUBNET_ID>"
```

```
    vpc_id = "<VPC_ID>"
    zone = "<ZONE>"
  }
}
# Create a new rancher2 Node Pool
resource "rancher2_node_pool" "foo" {
  cluster_id =  "${rancher2_cluster.foo-custom.id}"
  name = "foo"
  hostname_prefix =  "foo-cluster-0"
  node_template_id = "${rancher2_node_template.foo.id}"
  quantity = 1
  control_plane = true
  etcd = true
  worker = true
}
```

» Argument Reference

The following arguments are supported:

- `cluster_id` - (Required) The RKE cluster id to use Node Pool (string)
- `name` - (Required) The name of the Node Pool (string)
- `hostname_prefix` - (Required) The prefix for created nodes of the Node Pool (string)
- `node_template_id` - (Required) The Node Template ID to use for node creation (string)
- `delete_not_ready_after_secs` - (Optional) Delete not ready node after secs. For Rancher v2.3.3 or above. Default `0` (int)
- `node_taints` - (Required) Node taints. For Rancher v2.3.3 or above (List)
- `control_plane` - (Optional) RKE control plane role for created nodes (bool)
- `etcd` - (Optional) RKE etcd role for created nodes (bool)
- `quantity` - (Optional) The number of nodes to create on Node Pool. Default `1`. Only values >= 1 allowed (int)
- `worker` - (Optional) RKE role role for created nodes (bool)
- `annotations` - (Optional/Computed) Annotations for Node Pool object (map)
- `labels` - (Optional/Computed) Labels for Node Pool object (map)

» Nested blocks

» `node_taints`

» Arguments

- `key` - (Required) Taint key (string)
- `value` - (Required) Taint value (string)
- `effect` - (Optional) Taint effect. Supported values : `"NoExecute" | "NoSchedule" | "PreferNoSchedule"` (string)
- `time_added` - (Optional) Taint time added (string)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Timeouts

`rancher2_node_pool` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating node pools.
- `update` - (Default `10 minutes`) Used for node pool modifications.
- `delete` - (Default `10 minutes`) Used for deleting node pools.

» Import

Node Pool can be imported using the Rancher Node Pool ID

```
$ terraform import rancher2_node_pool.foo <node_pool_id>
```

» rancher2_node_template

Provides a Rancher v2 Node Template resource. This can be used to create Node Template for Rancher v2 and retrieve their information.

amazonec2, azure, digitalocean, opennebula, openstack, and vsphere drivers are supported for node templates.

Note If you are upgrading to Rancher v2.3.3, please take a look to final section

» Example Usage

```
# Create a new rancher2 Node Template up to Rancher 2.1.x
resource "rancher2_node_template" "foo" {
  name = "foo"
  description = "foo test"
  amazonec2_config {
```

```
    access_key = "AWS_ACCESS_KEY"
    secret_key = "<AWS_SECRET_KEY>"
    ami =  "<AMI_ID>"
    region = "<REGION>"
    security_group = ["<AWS_SECURITY_GROUP>"]
    subnet_id = "<SUBNET_ID>"
    vpc_id = "<VPC_ID>"
    zone = "<ZONE>"
  }
}
```

```
# Create a new rancher2 Node Template from Rancher 2.2.x
resource "rancher2_cloud_credential" "foo" {
  name = "foo"
  description = "foo test"
  amazonec2_credential_config {
    access_key = "<AWS_ACCESS_KEY>"
    secret_key = "<AWS_SECRET_KEY>"
  }
}
resource "rancher2_node_template" "foo" {
  name = "foo"
  description = "foo test"
  cloud_credential_id = "${rancher2_cloud_credential.foo.id}"
  amazonec2_config {
    ami =  "<AMI_ID>"
    region = "<REGION>"
    security_group = ["<AWS_SECURITY_GROUP>"]
    subnet_id = "<SUBNET_ID>"
    vpc_id = "<VPC_ID>"
    zone = "<ZONE>"
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the Node Template (string)
- `amazonec2_config` - (Optional) AWS config for the Node Template (list maxitems:1)
- `auth_certificate_authority` - (Optional/Sensitive) Auth certificate authority for the Node Template (string)
- `auth_key` - (Optional/Sensitive) Auth key for the Node Template (string)
- `azure_config` - (Optional) Azure config for the Node Template (list maxitems:1)

- `cloud_credential_id` - (Optional) Cloud credential ID for the Node Template. Required from Rancher v2.2.x (string)
- `description` - (Optional) Description for the Node Template (string)
- `digitalocean_config` - (Optional) Digitalocean config for the Node Template (list maxitems:1)
- `driver_id` - (Optional/Computed) The node driver id used by the node template. It's required if the node driver isn't built in Rancher (string)
- `engine_env` - (Optional) Engine environment for the node template (string)
- `engine_insecure_registry` - (Optional) Insecure registry for the node template (list)
- `engine_install_url` - (Optional) Docker engine install URL for the node template. Default `https://releases.rancher.com/install-docker/18.09.sh`. Available install docker versions at `https://github.com/rancher/install-docker` (string)
- `engine_label` - (Optional) Engine label for the node template (string)
- `engine_opt` - (Optional) Engine options for the node template (map)
- `engine_registry_mirror` - (Optional) Engine registry mirror for the node template (list)
- `engine_storage_driver` - (Optional) Engine storage driver for the node template (string)
- `opennebula_config` - (Optional) Opennebula config for the Node Template (list maxitems:1)
- `openstack_config` - (Optional) Openstack config for the Node Template (list maxitems:1)
- `use_internal_ip_address` - (Optional) Engine storage driver for the node template (bool)
- `vsphere_config` - (Optional) vSphere config for the Node Template (list maxitems:1)
- `annotations` - (Optional) Annotations for Node Template object (map)
- `labels` - (Optional/Computed) Labels for Node Template object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `driver` - (Computed) The driver of the node template (string)

» Nested blocks

» `amazonec2_config`

» Arguments

- `ami` - (Required) AWS machine image (string)
- `region` - (Required) AWS region. (string)
- `security_group` - (Required) AWS VPC security group. (list)
- `subnet_id` - (Required) AWS VPC subnet id (string)
- `vpc_id` - (Required) AWS VPC id. (string)
- `zone` - (Required) AWS zone for instance (i.e. a,b,c,d,e) (string)
- `access_key` - (Optional/Sensitive) AWS access key. Required on Rancher v2.0.x and v2.1.x. Use `rancher2_cloud_credential` from Rancher v2.2.x (string)
- `block_duration_minutes` - (Optional) AWS spot instance duration in minutes (60, 120, 180, 240, 300, or 360). Default `0` (string)
- `device_name` - (Optional) AWS root device name. Default `/dev/sda1` (string)
- `endpoint` - (Optional) Optional endpoint URL (hostname only or fully qualified URI) (string)
- `iam_instance_profile` - (Optional) AWS IAM Instance Profile (string)
- `insecure_transport` - (Optional) Disable SSL when sending requests (bool)
- `instance_type` - (Optional) AWS instance type. Default `t2.micro` (string)
- `keypair_name` - (Optional) AWS keypair to use; requires --amazonec2-ssh-keypath (string)
- `monitoring` - (Optional) Set this flag to enable CloudWatch monitoring. Deafult `false` (bool)
- `open_port` - (Optional) Make the specified port number accessible from the Internet. (list)
- `private_address_only` - (Optional) Only use a private IP address. Default `false` (bool)
- `request_spot_instance` - (Optional) Set this flag to request spot instance. Default `false` (bool)
- `retries` - (Optional) Set retry count for recoverable failures (use -1 to disable). Default `5` (string)
- `root_size` - (Optional) AWS root disk size (in GB). Default `16` (string)
- `secret_key` - (Optional/Sensitive) AWS secret key. Required on Rancher v2.0.x and v2.1.x. Use `rancher2_cloud_credential` from Rancher v2.2.x (string)
- `security_group_readonly` - (Optional) Skip adding default rules to security groups (bool)
- `session_token` - (Optional/Sensitive) AWS Session Token (string)
- `spot_price` - (Optional) AWS spot instance bid price (in dollar). Default `0.50` (string)
- `ssh_keypath` - (Optional) SSH Key for Instance (string)
- `ssh_user` - (Optional) Set the name of the ssh user (string)
- `tags` - (Optional) AWS Tags (e.g. key1,value1,key2,value2) (string)
- `use_ebs_optimized_instance` - (Optional) Create an EBS optimized instance. Default `false` (bool)

- `use_private_address` - (Optional) Force the usage of private IP address. Default `false` (bool)
- `userdata` - (Optional) Path to file with cloud-init user data (string)
- `volume_type` - (Optional) Amazon EBS volume type. Default `gp2` (string)

» `azure_config`

» Arguments

- `client_id` - (Optional/Sensitive) Azure Service Principal Account ID. Mandatory on Rancher v2.0.x and v2.1.x. Use `rancher2_cloud_credential` from Rancher v2.2.x (string)
- `client_secret` - (Optional/Sensitive) Azure Service Principal Account password. Mandatory on Rancher v2.0.x and v2.1.x. Use `rancher2_cloud_credential` from Rancher v2.2.x (string)
- `subscription_id` - (Optional/Sensitive) Azure Subscription ID. Mandatory on Rancher v2.0.x and v2.1.x. Use `rancher2_cloud_credential` from Rancher v2.2.x (string)
- `availability_set` - (Optional) Azure Availability Set to place the virtual machine into. Default `docker-machine` (string)
- `custom_data` - (Optional) Path to file with custom-data (string)
- `disk_size` - (Optional) Disk size if using managed disk. Just for Rancher v2.3.x and above. Default `30` (string)
- `dns` - (Optional) A unique DNS label for the public IP adddress (string)
- `docker_port` - (Optional) Port number for Docker engine. Default `2376` (string)
- `environment` - (Optional) Azure environment (e.g. AzurePublicCloud, AzureChinaCloud). Default `AzurePublicCloud` (string) `fault_domain_count` - (Optional) Fault domain count to use for availability set. Default `3` (string)
- `image` - (Optional) Azure virtual machine OS image. Default `canonical:UbuntuServer:18.04-LTS:latest` (string)
- `location` - (Optional) Azure region to create the virtual machine. Default `westus` (string)
- `managed_disks` - (Optional) Configures VM and availability set for managed disks. Just for Rancher v2.3.x and above. Default `false` (bool)
- `no_public_ip` - (Optional) Do not create a public IP address for the machine. Default `false` (bool)
- `open_port` - (Optional) Make the specified port number accessible from the Internet. (list)
- `private_ip_address` - (Optional) Specify a static private IP address for the machine. (string)
- `resource_group` - (Optional) Azure Resource Group name (will be created if missing). Default `docker-machine` (string)
- `size` - (Optional) Size for Azure Virtual Machine. Default `Standard_A2`

(string)
- `ssh_user` - (Optional) Username for SSH login (string)
- `static_public_ip` - (Optional) Assign a static public IP address to the machine. Default `false` (bool)
- `storage_type` - (Optional) Type of Storage Account to host the OS Disk for the machine. Default `Standard_LRS` (string)
- `subnet` - (Optional) Azure Subnet Name to be used within the Virtual Network. Default `docker-machine` (string)
- `subnet_prefix` - (Optional) Private CIDR block to be used for the new subnet, should comply RFC 1918. Default `192.168.0.0/16` (string)
- `update_domain_count` - (Optional) Update domain count to use for availability set. Default `5` (string)
- `use_private_ip` - (Optional) Use private IP address of the machine to connect. Default `false` (bool)
- `vnet` - (Optional) Azure Virtual Network name to connect the virtual machine (in [resourcegroup:]name format). Default `docker-machine-vnet` (string)

» `digitalocean_config`

» Arguments

- `access_token` - (Optional/Sensitive) Digital Ocean access token. Mandatory on Rancher v2.0.x and v2.1.x. Use `rancher2_cloud_credential` from Rancher v2.2.x (string)
- `backups` - (Optional) Enable backups for droplet. Default `false` (bool)
- `image` - (Optional) Digital Ocean Image. Default `ubuntu-16-04-x64` (string)
- `ipv6` - (Optional) Enable ipv6 for droplet. Default `false` (bool)
- `monitoring` - (Optional) Enable monitoring for droplet. Default `false` (bool)
- `private_networking` - (Optional) Enable private networking for droplet. Default `false` (bool)
- `region` - (Optional) Digital Ocean region. Default `nyc3` (string)
- `size` - (Optional) Digital Ocean size. Default `s-1vcpu-1gb` (string)
- `ssh_key_fingerprint` - (Optional/Sensitive) SSH key fingerprint (string)
- `ssh_key_path` - (Optional) SSH private key path (string)
- `ssh_port` - (Optional) SSH port. Default `22` (string)
- `ssh_user` - (Optional) SSH username. Default `root` (string)
- `tags` - (Optional) Comma-separated list of tags to apply to the Droplet (string)
- `userdata` - (Optional) Path to file with cloud-init user-data (string)

» `opennebula_config`

» **Arguments**

- `image_id` - (Required*) Image ID to use as the VM OS. Conflicts with `image_name` (string)
- `image_name` - (Required*) Opennebula image to use as the VM OS. Conflicts with `image_id` (string)
- `template_id` - (Required*) Opennebula template ID to use. Conflicts with `template_name` (string)
- `template_name` - (Required*) Name of the Opennbula template to use. Conflicts with `template_id` (string)
- `password` - (Required/Sensitive) Set the password for the XML-RPC API authentication (string)
- `user` - (Required) Set the user for the XML-RPC API authentication (string)
- `xml_rpc_url` - (Required) Set the url for the Opennebula XML-RPC API (string)
- `b2d_size` - (Optional) Size of the Volatile disk in MB - only for b2d (string)
- `cpu` - (Optional) CPU value for the VM (string)
- `dev_prefix` - (Optional) Dev prefix to use for the images. E.g.: 'vd', 'sd', 'hd' (string)
- `disable_vnc` - (Optional) VNC is enabled by default. Disable it with this flag (bool)
- `disk_resize` - (Optional) Size of the disk for the VM in MB (string)
- `image_owner` - (Optional) Owner of the image to use as the VM OS (string)
- `memory`- (Optional) Size of the memory for the VM in MB (string)
- `network_id` - (Optional) Opennebula network ID to connect the machine to. Conflicts with `network_name` (string)
- `network_name` - (Optional) Opennebula network to connect the machine to. Conflicts with `network_id` (string)
- `network_owner` - (Optional) Opennebula user ID of the Network to connect the machine to (string)
- `ssh_user` - (Optional) Set the name of the SSH user. Defaults to docker (string)
- `vcpu` - (Optional) VCPUs for the VM (string)

  Note: `Required*` denotes that one of image_name / image_id or template_name / template_id is required but you cannot combine them.

» `openstack_config`

» **Arguments**

- `auth_url` - (Required) OpenStack authentication URL (string)
- `availability_zone` - (Required) OpenStack availability zone (string)
- `region` - (Required) OpenStack region name (string)
- `username` - (Required) OpenStack username (string)
- `active_timeout`- (Optional) OpenStack active timeout Default `200` (string)
- `cacert` - (Optional) CA certificate bundle to verify against (string)
- `config_drive` - (Optional) Enables the OpenStack config drive for the instance. Default `false` (bool)
- `domain_id` - (Required*) OpenStack domain ID. Identity v3 only. Conflicts with `domain_name` (string)
- `domain_name` - (Required*) OpenStack domain name. Identity v3 only. Conflicts with `domain_id` (string)
- `endpoint_type` - (Optional) OpenStack endpoint type. adminURL, internalURL or publicURL (string)
- `flavor_id` - (Required*) OpenStack flavor id to use for the instance. Conflicts with `flavor_name` (string)
- `flavor_name` - (Required*) OpenStack flavor name to use for the instance. Conflicts with `flavor_id` (string)
- `floating_ip_pool` - (Optional) OpenStack floating IP pool to get an IP from to assign to the instance (string)
- `image_id` - (Required*) OpenStack image id to use for the instance. Conflicts with `image_name` (string)
- `image_name` - (Required*) OpenStack image name to use for the instance. Conflicts with `image_id` (string)
- `insecure` - (Optional) Disable TLS credential checking. Default `false` (bool)
- `ip_version` - (Optional) OpenStack version of IP address assigned for the machine Default `4` (string)
- `keypair_name` - (Optional) OpenStack keypair to use to SSH to the instance (string)
- `net_id` - (Required*) OpenStack network id the machine will be connected on. Conflicts with `net_name` (string)
- `net_name` - (Required*) OpenStack network name the machine will be connected on. Conflicts with `net_id` (string)
- `nova_network` - (Optional) Use the nova networking services instead of neutron (string)
- `password` - (Optional/Sensitive) OpenStack password. Mandatory on Rancher v2.0.x and v2.1.x. Use `rancher2_cloud_credential` from Rancher v2.2.x (string)
- `private_key_file` - (Optional/Sensitive) Private key content to use for SSH (string)
- `sec_groups` - (Optional) OpenStack comma separated security groups for the machine (string)
- `ssh_port` - (Optional) OpenStack SSH port * Default `22` (string)
- `ssh_user` - (Optional) OpenStack SSH user * Default: `root` (string)

- `tenant_id` - (Required*) OpenStack tenant id. Conflicts with `tenant_name` (string)
- `tenant_name` - (Required*) OpenStack tenant name. Conflicts with `tenant_id` (string)
- `user_data_file` - (Optional) File containing an openstack userdata script (string)

  Note: `Required*` denotes that either the _name or _id is required but you cannot use both.

» `vsphere_config`

» Arguments

- `boot2docker_url` - (Optional) vSphere URL for boot2docker iso image. Default `https://releases.rancher.com/os/latest/rancheros-vmware.iso` (string)
- `cfgparam` - (Optional) vSphere vm configuration parameters (used for guestinfo) (list)
- `clone_from` - (Optional) If you choose creation type clone a name of what you want to clone is required. From Rancher v2.3.3 (string)
- `cloud_config` - (Optional) Filepath to a cloud-config yaml file to put into the ISO user-data. From Rancher v2.3.3 (string)
- `cloudinit` - (Optional) vSphere cloud-init file or url to set in the guestinfo (string)
- `content_library` - (Optional) If you choose to clone from a content library template specify the name of the library. From Rancher v2.3.3 (string)
- `cpu_count` - (Optional) vSphere CPU number for docker VM. Default `2` (string)
- `creation_type` - (Optional) Creation type when creating a new virtual machine. Supported values: vm, template, library, legacy. Default `legacy`. From Rancher v2.3.3 (string)
- `custom_attributes` - (Optional) vSphere custom attributes, format key/value e.g. `200=my custom value`. From Rancher v2.3.3 (List)
- `datacenter` - (Optional) vSphere datacenter for docker VM (string)
- `datastore` - (Optional) vSphere datastore for docker VM (string)
- `datastore_cluster` - (Optional) vSphere datastore cluster for virtual machine. From Rancher v2.3.3 (string)
- `disk_size` - (Optional) vSphere size of disk for docker VM (in MB). Default `20480` (string)
- `folder` - (Optional) vSphere folder for the docker VM. This folder must already exist in the datacenter (string)
- `hostsystem` - (Optional) vSphere compute resource where the docker VM will be instantiated. This can be omitted if using a cluster with DRS (string)

- `memory_size` - (Optional) vSphere size of memory for docker VM (in MB). Default `2048` (string)
- `network` - (Optional) vSphere network where the docker VM will be attached (list)
- `password` - (Optional/Sensitive) vSphere password. Mandatory on Rancher v2.0.x and v2.1.x. Use `rancher2_cloud_credential` from Rancher v2.2.x (string)
- `pool` - (Optional) vSphere resource pool for docker VM (string)
- `ssh_password` - (Optional) If using a non-B2D image you can specify the ssh password. Default `tcuser`. From Rancher v2.3.3 (string)
- `ssh_port` - (Optional) If using a non-B2D image you can specify the ssh port. Default `22`. From Rancher v2.3.3 (string)
- `ssh_user` - (Optional) If using a non-B2D image you can specify the ssh user. Default `docker`. From Rancher v2.3.3 (string)
- `ssh_user_group` - (Optional) If using a non-B2D image the uploaded keys will need chown'ed. Default `staff`. From Rancher v2.3.3 (string)
- `tags` - (Optional) vSphere tags id e.g. `urn:xxx`. From Rancher v2.3.3 (list)
- `username` - (Optional/Sensitive) vSphere username. Mandatory on Rancher v2.0.x and v2.1.x. Use `rancher2_cloud_credential` from Rancher v2.2.x (string)
- `vapp_ip_allocation_policy` - (Optional) vSphere vApp IP allocation policy. Supported values are: `dhcp`, `fixed`, `transient` and `fixedAllocated` (string)
- `vapp_ip_protocol` - (Optional) vSphere vApp IP protocol for this deployment. Supported values are: `IPv4` and `IPv6` (string)
- `vapp_property` - (Optional) vSphere vApp properties (list)
- `vapp_transport` - (Optional) vSphere OVF environment transports to use for properties. Supported values are: `iso` and `com.vmware.guestInfo` (string)
- `vcenter` - (Optional/Sensitive) vSphere IP/hostname for vCenter. Mandatory on Rancher v2.0.x and v2.1.x. Use `rancher2_cloud_credential` from Rancher v2.2.x (string)
- `vcenter_port` - (Optional/Sensitive) vSphere Port for vCenter. Mandatory on Rancher v2.0.x and v2.1.x. Use `rancher2_cloud_credential` from Rancher v2.2.x. Default `443` (string)

» Timeouts

`rancher2_node_template` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating node templates.
- `update` - (Default `10 minutes`) Used for node template modifications.
- `delete` - (Default `10 minutes`) Used for deleting node templates.

» Import

Node Template can be imported using the Rancher Node Template ID

```
$ terraform import rancher2_node_template.foo <node_template_id>
```

» Upgrading to Rancher v2.3.3

Important This process could update `rancher2_node_template` data on tfstate file. Be sure to save a copy of tfstate file before proceed

Due to this feature included on Rancher v2.3.3, `rancher2_node_template` are now global scoped objects with RBAC around them, instead of user scoped objects as they were. This means that existing node templates `id` field is changing on upgrade. Provider implements `fixNodeTemplateID()` that will update tfstate with proper id. “‘

» rancher2_notifier

Provides a Rancher v2 Notifier resource. This can be used to create notifiers for Rancher v2 environments and retrieve their information.

» Example Usage

```
# Create a new rancher2 Notifier
resource "rancher2_notifier" "foo" {
  name = "foo"
  cluster_id = "<cluster_id>"
  description = "Terraform notifier acceptance test"
  send_resolved = "true"
  pagerduty_config {
    service_key = "XXXXXXXX"
    proxy_url = "http://proxy.test.io"
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the notifier (string)
- `cluster_id` - (Required/ForceNew) The cluster id where create notifier (string)

- `description` - (Optional) The notifier description (string)
- `send_resolved` = (Optional) Enable the notifier to send resolved notifications. Default `false` (bool)
- `pagerduty_config` - (Optional) Pagerduty config for notifier (list maxitems:1)
- `slack_config` - (Optional) Slack config for notifier (list maxitems:1)
- `smtp_config` - (Optional) SMTP config for notifier (list maxitems:1)
- `webhook_config` - (Optional) Webhook config for notifier (list maxitems:1)
- `wechat_config` - (Optional) Wechat config for notifier (list maxitems:1)
- `annotations` - (Optional/Computed) Annotations for notifier object (map)
- `labels` - (Optional/Computed) Labels for notifier object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Nested blocks

» `pagerduty_config`

» Arguments

- `service_key` - (Required) Pagerduty service key (string)
- `proxy_url` - (Optional) Pagerduty proxy url (string)

» `slack_config`

» Arguments

- `default_recipient` - (Required) Slack default recipient (string)
- `url` - (Required) Slack url (string)
- `proxy_url` - (Optional) Slack proxy url (string)

» `smtp_config`

- `default_recipient` - (Required) SMTP default recipient (string)
- `host` - (Required) SMTP host (string)
- `port` - (Required) SMTP port (int)
- `sender` - (Required) SMTP sender (string)
- `password` - (Optional/Sensitive) SMTP password (string)

- `tls` - (Optional/Sensitive) SMTP tls. Default `true` (bool)
- `username` - (Optional/Sensitive) SMTP username (string)

» Arguments

» `webhook_config`

- `url` - (Required) Webhook url (string)
- `proxy_url` - (Optional) Webhook proxy url (string)

» Arguments

» `wechat_config`

» Arguments

- `agent` - (Required) Wechat agent ID (string)
- `corp` - (Required) Wechat corporation ID (string)
- `default_recipient` - (Required) Wechat default recipient (string)
- `secret` - (Required/Sensitive) Wechat agent ID (string)
- `proxy_url` - (Optional) Wechat proxy url (string)
- `recipient_type` - (Optional) Wechat recipient type. Allowed values: `party` | `tag` | `user` (string)

» Timeouts

`rancher2_notifier` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating notifiers.
- `update` - (Default `10 minutes`) Used for notifier modifications.
- `delete` - (Default `10 minutes`) Used for deleting notifiers.

» Import

Notifiers can be imported using the Rancher nNtifier ID

```
$ terraform import rancher2_notifier.foo <notifier_id>
```

» rancher2_project

Provides a Rancher v2 Project resource. This can be used to create projects for Rancher v2 environments and retrieve their information.

» Example Usage

```
# Create a new rancher2 Project
resource "rancher2_project" "foo" {
  name = "foo"
  cluster_id = "<CLUSTER_ID>"
  resource_quota {
    project_limit {
      limits_cpu = "2000m"
      limits_memory = "2000Mi"
      requests_storage = "2Gi"
    }
    namespace_default_limit {
      limits_cpu = "2000m"
      limits_memory = "500Mi"
      requests_storage = "1Gi"
    }
  }
  container_resource_limit {
    limits_cpu = "20m"
    limits_memory = "20Mi"
    requests_cpu = "1m"
    requests_memory = "1Mi"
  }
}

# Create a new rancher2 Project enabling and customizing monitoring
resource "rancher2_project" "foo" {
  name = "foo"
  cluster_id = "<CLUSTER_ID>"
  resource_quota {
    project_limit {
      limits_cpu = "2000m"
      limits_memory = "2000Mi"
      requests_storage = "2Gi"
    }
    namespace_default_limit {
      limits_cpu = "2000m"
      limits_memory = "500Mi"
      requests_storage = "1Gi"
```

```
    }
  }
  container_resource_limit {
    limits_cpu = "20m"
    limits_memory = "20Mi"
    requests_cpu = "1m"
    requests_memory = "1Mi"
  }
  enable_project_monitoring = true
  project_monitoring_input {
    answers = {
      "exporter-kubelets.https" = true
      "exporter-node.enabled" = true
      "exporter-node.ports.metrics.port" = 9796
      "exporter-node.resources.limits.cpu" = "200m"
      "exporter-node.resources.limits.memory" = "200Mi"
      "grafana.persistence.enabled" = false
      "grafana.persistence.size" = "10Gi"
      "grafana.persistence.storageClass" = "default"
      "operator.resources.limits.memory" = "500Mi"
      "prometheus.persistence.enabled" = "false"
      "prometheus.persistence.size" = "50Gi"
      "prometheus.persistence.storageClass" = "default"
      "prometheus.persistent.useReleaseName" = "true"
      "prometheus.resources.core.limits.cpu" = "1000m",
      "prometheus.resources.core.limits.memory" = "1500Mi"
      "prometheus.resources.core.requests.cpu" = "750m"
      "prometheus.resources.core.requests.memory" = "750Mi"
      "prometheus.retention" = "12h"
    }
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the project (string)
- `cluster_id` - (Required) The cluster id where create project (string)
- `container_resource_limit` - (Optional) Default containers resource limits on project (List maxitem:1)
- `description` - (Optional) A project description (string)
- `enable_project_monitoring` - (Optional) Enable built-in project monitoring. Default `false` (bool)
- `pod_security_policy_template_id` - (Optional) Default Pod Security

Policy ID for the project (string)
- `project_monitoring_input` - (Optional/Computed) Project monitoring config. Any parameter defined in rancher-monitoring charts could be configured (list maxitems:1)
- `resource_quota` - (Optional) Resource quota for project. Rancher v2.1.x or higher (list maxitems:1)
- `wait_for_cluster` - (Optional) Wait for cluster becomes active. Default `false` (bool)
- `annotations` - (Optional/Computed) Annotations for Node Pool object (map)
- `labels` - (Optional/Computed) Labels for Node Pool object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Nested blocks

» `container_resource_limit`

» Arguments

- `limits_cpu` - (Optional) CPU limit for containers (string)
- `limits_memory` - (Optional) Memory limit for containers (string)
- `requests_cpu` - (Optional) CPU reservation for containers (string)
- `requests_memory` - (Optional) Memory reservation for containers (string)

» `project_monitoring_input`

» Arguments

- `answers` - (Optional/Computed) Key/value answers for monitor input (map)

» `resource_quota`

» Arguments

- `project_limit` - (Required) Resource quota limit for project (list maxitems:1)
- `namespace_default_limit` - (Required) Default resource quota limit for namespaces in project (list maxitems:1)

» `project_limit` and `namespace_default_limit`

» Arguments

The following arguments are supported:

- `config_maps` - (Optional) Limit for config maps in project (string)
- `limits_cpu` - (Optional) Limit for limits cpu in project (string)
- `limits_memory` - (Optional) Limit for limits memory in project (string)
- `persistent_volume_claims` - (Optional) Limit for persistent volume claims in project (string)
- `pods` - (Optional) Limit for pods in project (string)
- `replication_controllers` - (Optional) Limit for replication controllers in project (string)
- `requests_cpu` - (Optional) Limit for requests cpu in project (string)
- `requests_memory` - (Optional) Limit for requests memory in project (string)
- `requests_storage` - (Optional) Limit for requests storage in project (string)
- `secrets` - (Optional) Limit for secrets in project (string)
- `services_load_balancers` - (Optional) Limit for services load balancers in project (string)
- `services_node_ports` - (Optional) Limit for services node ports in project (string)

More info at resource-quotas

» Timeouts

`rancher2_project` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating projects.
- `update` - (Default `10 minutes`) Used for project modifications.
- `delete` - (Default `10 minutes`) Used for deleting projects.

» Import

Projects can be imported using the Rancher Project ID

```
$ terraform import rancher2_project.foo <project_id>
```

» rancher2_project_logging

Provides a Rancher v2 Project Logging resource. This can be used to create Project Logging for Rancher v2 environments and retrieve their information.

» Example Usage

```
# Create a new rancher2 Project Logging
resource "rancher2_project_logging" "foo" {
  name = "foo"
  project_id = "<project_id>"
  kind = "syslog"
  syslog_config {
    endpoint = "<syslog_endpoint>"
    protocol = "udp"
    severity = "notice"
    ssl_verify = false
  }
}
```

» Argument Reference

The following arguments are supported:

- `project_id` - (Required) The project id to configure logging (string)
- `name` - (Required) The name of the Project Logging config (string)
- `kind` - (Required) The kind of the Project Logging. `elasticsearch`, `fluentd`, `kafka`, `splunk` and `syslog` are supported (string)
- `custom_target_config` - (Optional) The custom target config for Cluster Logging. For `kind = custom`. Conflicts with `elasticsearch_config`, `fluentd_config`, `kafka_config`, `splunk_config` and `syslog_config` (list maxitems:1)
- `enable_json_parsing` - (Optional) Enable json log parsing. Default: `false` (bool)
- `elasticsearch_config` - (Optional) The elasticsearch config for Project Logging. For `kind = elasticsearch`. Conflicts with `custom_target_config`, `fluentd_config`, `kafka_config`, `splunk_config` and `syslog_config` (list maxitems:1)
- `fluentd_config` - (Optional) The fluentd config for Project Logging. For `kind = fluentd`. Conflicts with `custom_target_config`, `elasticsearch_config`, `kafka_config`, `splunk_config` and `syslog_config` (list maxitems:1)
- `kafka_config` - (Optional) The kafka config for Project Logging. For `kind = kafka`. Conflicts with `custom_target_config`,

`elasticsearch_config`, `fluentd_config`, `splunk_config` and `syslog_config` (list maxitems:1)

- `namespace_id` - (Optional) The namespace id from Project logging (string)
- `output_flush_interval` - (Optional) How often buffered logs would be flushed. Default: `3` seconds (int)
- `output_tags` - (Optional/computed) The output tags for Project Logging (map)
- `splunk_config` - (Optional) The splunk config for Project Logging. For `kind = splunk`. Conflicts with `custom_target_config`, `elasticsearch_config`, `fluentd_config`, `kafka_config`, and `syslog_config` (list maxitems:1)
- `syslog_config` - (Optional) The syslog config for Project Logging. For `kind = syslog`. Conflicts with `custom_target_config`, `elasticsearch_config`, `fluentd_config`, `kafka_config`, and `splunk_config` (list maxitems:1)
- `annotations` - (Optional/Computed) Annotations for Project Logging object (map)
- `labels` - (Optional/Computed) Labels for Project Logging object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Nested blocks

» `custom_target_config`

» Arguments

- `content` - (Required) Custom target config content (string)
- `certificate` - (Required/Sensitive) SSL CA certificate for the custom target service (string)
- `client_cert` - (Optional/Sensitive) SSL client certificate for the custom target service (string)
- `client_key` - (Optional/Sensitive) SSL client key for the custom target service (string)

» `elasticsearch_config`

» Arguments

- `endpoint` - (Required) Endpoint of the elascticsearch service. Must include protocol, `http://` or `https://` (string)
- `auth_password` - (Optional/Sensitive) User password for the elascticsearch service (string)
- `auth_username` - (Optional/Sensitive) Username for the elascticsearch service (string)
- `certificate` - (Optional/Sensitive) SSL certificate for the elascticsearch service (string)
- `client_cert` - (Optional/Sensitive) SSL client certificate for the elascticsearch service (string)
- `client_key` - (Optional/Sensitive) SSL client key for the elascticsearch service (string)
- `client_key_pass` - (Optional/Sensitive) SSL client key password for the elascticsearch service (string)
- `date_format` - (Optional) Date format for the elascticsearch logs. Default: `YYYY-MM-DD` (string)
- `index_prefix` - (Optional) Index prefix for the elascticsearch logs. Default: `local` (string)
- `ssl_verify` - (Optional) SSL verify for the elascticsearch service (bool)
- `ssl_version` - (Optional) SSL version for the elascticsearch service (string)

» `fluentd_config`

» Arguments

- `fluent_servers` - (Required) Servers for the fluentd service (list)
- `certificate` - (Optional/Sensitive) SSL certificate for the fluentd service (string)
- `compress` - (Optional) Compress data for the fluentd service (bool)
- `enable_tls` - (Optional) Enable TLS for the fluentd service (bool)

» `fluent_servers`

» Arguments

- `endpoint` - (Required) Endpoint of the fluentd service (string)
- `hostname` - (Optional) Hostname of the fluentd service (string)
- `password` - (Optional/Sensitive) User password of the fluentd service (string)
- `shared_key` - (Optional/Sensitive) Shared key of the fluentd service (string)
- `standby` - (Optional) Standby server of the fluentd service (bool)
- `username` - (Optional/Sensitive) Username of the fluentd service (string)
- `weight` - (Optional) Weight of the fluentd server (int)

» `kafka_config`

» **Arguments**

- `topic` - (Required) Topic to publish on the kafka service (string)
- `broker_endpoints` - (Optional) Kafka endpoints for kafka service. Conflicts with `zookeeper_endpoint` (list)
- `certificate` - (Optional/Sensitive) SSL certificate for the kafka service (string)
- `client_cert` - (Optional/Sensitive) SSL client certificate for the kafka service (string)
- `client_key` - (Optional/Sensitive) SSL client key for the kafka service (string)
- `zookeeper_endpoint` - (Optional) Zookeeper endpoint for kafka service. Conflicts with `broker_endpoints` (string)

» `splunk_config`

» **Arguments**

- `endpoint` - (Required) Endpoint of the splunk service. Must include protocol, `http://` or `https://` (string)
- `token` - (Required/Sensitive) Token for the splunk service (string)
- `certificate` - (Optional/Sensitive) SSL certificate for the splunk service (string)
- `client_cert` - (Optional/Sensitive) SSL client certificate for the splunk service (string)
- `client_key` - (Optional/Sensitive) SSL client key for the splunk service (string)
- `client_key_pass` - (Optional/Sensitive) SSL client key password for the splunk service (string)
- `index` - (Optional) Index prefix for the splunk logs (string)
- `source` - (Optional) Date format for the splunk logs (string)
- `ssl_verify` - (Optional) SSL verify for the splunk service (bool)

» `syslog_config`

» **Arguments**

- `endpoint` - (Required) Endpoint of the syslog service (string)
- `certificate` - (Optional/Sensitive) SSL certificate for the syslog service (string)
- `client_cert` - (Optional/Sensitive) SSL client certificate for the syslog service (string)
- `client_key` - (Optional/Sensitive) SSL client key for the syslog service (string)

- `program` - (Optional) Program for the syslog service (string)
- `protocol` - (Optional) Protocol for the syslog service. `tcp` and `udp` are supported. Default: `udp` (string)
- `severity` - (Optional) Date format for the syslog logs. `emergency`, `alert`, `critical`, `error`, `warning`, `notice`, `info` and `debug` are supported. Default: `notice` (string)
- `ssl_verify` - (Optional) SSL verify for the syslog service (bool)
- `token` - (Optional/Sensitive) Token for the syslog service (string)

» Timeouts

`rancher2_project_logging` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating project logging configurations.
- `update` - (Default `10 minutes`) Used for project logging configuration modifications.
- `delete` - (Default `10 minutes`) Used for deleting project logging configurations.

» Import

Project Logging can be imported using the Rancher Project Logging ID

```
$ terraform import rancher2_project_logging.foo <project_logging_id>
```

» rancher2_project_role_template_binding

Provides a Rancher v2 Project Role Template Binding resource. This can be used to create Project Role Template Bindings for Rancher v2 environments and retrieve their information.

» Example Usage

```
# Create a new rancher2 Project Role Template Binding
resource "rancher2_project_role_template_binding" "foo" {
  name = "foo"
  project_id = "<project_id>"
  role_template_id = "<role_template_id>"
  user_id = "<user_id>"
}
```

» Argument Reference

The following arguments are supported:

- `project_id` - (Required/ForceNew) The project id where bind project role template (string)
- `role_template_id` - (Required/ForceNew) The role template id from create project role template binding (string)
- `name` - (Required) The name of the project role template binding (string)
- `group_id` - (Optional) The group ID to assign project role template binding (string)
- `group_principal_id` - (Optional/Computed) The group_principal ID to assign project role template binding (string)
- `user_id` - (Optional) The user ID to assign project role template binding (string)
- `user_principal_id` - (Optional/Computed) The user_principal ID to assign project role template binding (string)
- `annotations` - (Optional/Computed) Annotations of the resource (map)
- `labels` - (Optional/Computed) Labels of the resource (map)

Note user `user_id | user_principal_id` OR group `group_id | group_principal_id` must be defined

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Timeouts

`rancher2_project_role_template_binding` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating project role template bindings.
- `update` - (Default `10 minutes`) Used for project role template binding modifications.
- `delete` - (Default `10 minutes`) Used for deleting project role template bindings.

» Import

Project Role Template Bindings can be imported using the Rancher Project Role Template Binding ID

```
$ terraform import rancher2_project_role_template_binding.foo <project_role_template_binding
```

» rancher2_registry

Provides a Rancher v2 Registry resource. This can be used to create docker registries for Rancher v2 environments and retrieve their information.

Depending of the availability, there are 2 types of Rancher v2 docker registries: - Project registry: Available to all namespaces in the `project_id` - Namespaced regitry: Available to just `namespace_id` in the `project_id`

» Example Usage

```
# Create a new rancher2 Project Registry
resource "rancher2_registry" "foo" {
  name = "foo"
  description = "Terraform registry foo"
  project_id = "<project_id>"
  registries {
    address = "test.io"
    username = "user"
    password = "pass"
  }
}
```

```
# Create a new rancher2 Namespaced Registry
resource "rancher2_registry" "foo" {
  name = "foo"
  description = "Terraform registry foo"
  project_id = "<project_id>"
  namespace_id = "<namespace_id>"
  registries {
    address = "test.io"
    username = "user2"
    password = "pass"
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required/ForceNew) The name of the registry (string)

- `project_id` - (Required/ForceNew) The project id where to assign the registry (string)
- `registries` - (Required) Registries data for registry (list)
- `description` - (Optional) A registry description (string)
- `namespace_id` - (Optional) The namespace id where to assign the namespaced registry (string)
- `annotations` - (Optional/Computed) Annotations for Registry object (map)
- `labels` - (Optional/Computed) Labels for Registry object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Nested blocks

» `registries`

» Arguments

- `address` - (Required) Address for registry.
- `password` - (Optional) Password for the registry (string)
- `username` - (Optional) Username for the registry (string)

» Timeouts

`rancher2_registry` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating registries.
- `update` - (Default `10 minutes`) Used for registry modifications.
- `delete` - (Default `10 minutes`) Used for deleting registries.

» Import

Registries can be imported using the registry ID in the format `<namespace_id>.<project_id>.<registry_id>`

```
$ terraform import rancher2_registry.foo <namespace_id>.<project_id>.<registry_id>
```

`<namespace_id>` is optional, just needed for namespaced registry.

» rancher2_role_template

Provides a Rancher v2 Role Template resource. This can be used to create Role Template for Rancher v2 and retrieve their information.

`cluster` and `project` scopes are supported for role templates.

» Example Usage

```
# Create a new rancher2 cluster Role Template
resource "rancher2_role_template" "foo" {
  name = "foo"
  context = "cluster"
  default_role = true
  description = "Terraform role template acceptance test"
  rules {
    api_groups = ["*"]
    resources = ["secrets"]
    verbs = ["create"]
  }
}
```

```
# Create a new rancher2 project Role Template
resource "rancher2_role_template" "foo" {
  name = "foo"
  context = "project"
  default_role = true
  description = "Terraform role template acceptance test"
  rules {
    api_groups = ["*"]
    resources = ["secrets"]
    verbs = ["create"]
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Role template name (string)
- `administrative` - (Optional) Administrative role template. Default `false` (bool)
- `context` - (Optional) Role template context. `cluster` and `project` values are supported. Default: `cluster` (string)

- `default_role` - (Optional) Default role template for new created cluster or project. Default `false` (bool)
- `description` - (Optional/Computed) Role template description (string)
- `external` - (Optional) External role template. Default `false` (bool)
- `hidden` - (Optional) Hidden role template. Default `false` (bool)
- `locked` - (Optional) Locked role template. Default `false` (bool)
- `role_template_ids` - (Optional/Computed) Inherit role template IDs (list)
- `rules` - (Optional/Computed) Role template policy rules (list)
- `annotations` - (Optional/Computed) Annotations for role template object (map)
- `labels` - (Optional/Computed) Labels for role template object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `builtin` - (Computed) Builtin role template (string)

» Nested blocks

» `rules`

» Arguments

- `api_groups` - (Optional) Policy rule api groups (list)
- `non_resource_urls` - (Optional) Policy rule non resource urls (list)
- `resource_names` - (Optional) Policy rule resource names (list)
- `resources` - (Optional) Policy rule resources (list)
- `verbs` - (Optional) Policy rule verbs. `create`, `delete`, `get`, `list`, `patch`, `update`, `watch` and `*` values are supported (list)

» Timeouts

`rancher2_role_template` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating role templates.
- `update` - (Default `10 minutes`) Used for role template modifications.
- `delete` - (Default `10 minutes`) Used for deleting role templates.

» Import

Role Template can be imported using the Rancher Role Template ID

```
$ terraform import rancher2_role_template.foo <role_template_id>
```

» rancher2_secret

Provides a Rancher v2 Secret resource. This can be used to create secrets for Rancher v2 environments and retrieve their information.

Depending of the availability, there are 2 types of Rancher v2 secrets: - Project secret: Available to all namespaces in the `project_id` - Namespaced secret: Available to just `namespace_id` in the `project_id`

» Example Usage

```
# Create a new rancher2 Project Secret
resource "rancher2_secret" "foo" {
  name = "foo"
  description = "Terraform secret foo"
  project_id = "<project_id>"
  data = {
    address = base64encode("test.io")
    username = base64encode("user2")
    password = base64encode("pass")
  }
}

# Create a new rancher2 Namespaced Secret
resource "rancher2_secret" "foo" {
  name = "foo"
  description = "Terraform secret foo"
  project_id = "<project_id>"
  namespace_id = "<namespace_id>"
  data = {
    address = base64encode("test.io")
    username = base64encode("user2")
    password = base64encode("pass")
  }
}
```

» Argument Reference

The following arguments are supported:

- `data` - (Required/Sensitive) Secret key/value data. Base64 encoding required for values (map)
- `project_id` - (Required/ForceNew) The project id where to assign the secret (string)
- `description` - (Optional) A secret description (string)
- `name` - (Optional/ForceNew) The name of the secret (string)
- `namespace_id` - (Optional/ForceNew) The namespace id where to assign the namespaced secret (string)
- `annotations` - (Optional/Computed) Annotations for secret object (map)
- `labels` - (Optional/Computed) Labels for secret object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Timeouts

`rancher2_secret` provides the following Timeouts configuration options:

- `create` - (Default `10 minutes`) Used for creating registries.
- `update` - (Default `10 minutes`) Used for secret modifications.
- `delete` - (Default `10 minutes`) Used for deleting registries.

» Import

Secrets can be imported using the secret ID in the format `<namespace_id>.<project_id>.<secret_id>`

```
$ terraform import rancher2_secret.foo <namespace_id>.<project_id>.<secret_id>
```

`<namespace_id>` is optional, just needed for namespaced secret.

» rancher2_setting

Provides a Rancher v2 Setting resource. This can be used to create settings for Rancher v2 environments and retrieve their information.

On create, if setting already exists, provider will import it and update its value.

On destroy, if setting is a system setting like `server-url`, provider'll not delete it from Rancher, it'll just update setting value to default and remove it from tfstate.

» Example Usage

```
# Create a new rancher2 Setting
resource "rancher2_setting" "foo" {
  name = "foo"
  value = "<VALUE>"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the setting (string)
- `value` - (Required) The value of the setting (string)
- `annotations` - (Optional/Computed) Annotations for setting object (map)
- `labels` - (Optional/Computed) Labels for setting object (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)

» Import

Setting can be imported using the Rancher setting ID.

```
$ terraform import rancher2_setting.foo <setting_id>
```

» rancher2_token

Provides a Rancher v2 Token resource. This can be used to create Tokens for Rancher v2 provider user and retrieve their information.

There are 2 kind of tokens: - no scoped: valid for global system. - scoped: valid for just a specific cluster (`cluster_id` should be provided).

Tokens can't be updated once created. Any diff in token data will recreate the token. If any token expire, Rancher2 provider will generate a diff to regenerate it.

» Example Usage

```
# Create a new rancher2 Token
resource "rancher2_token" "foo" {
  description = "foo token"
  ttl = 1200
}
# Create a new rancher2 Token scoped
resource "rancher2_token" "foo" {
  cluster_id = "<cluster-id>"
  description = "foo token"
  ttl = 1200
}
```

» Argument Reference

The following arguments are supported:

- `cluster_id` - (Optional/ForceNew) Cluster ID for scoped token (string)
- `description` - (Optional/ForceNew) Token description (string)
- `renew` - (Optional/ForceNew) Renew token if expired or disabled. If `true`, a terraform diff would be generated to renew the token if it's disabled or expired. If `false`, the token will not be renewed. Default `true` (bool)
- `ttl` - (Optional/ForceNew) Token time to live in seconds. Default `0` (int)
- `annotations` - (Optional/Computed) Annotations of the token (map)
- `labels` - (Optional/Computed) Labels of the token (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `access_key` - (Computed) Token access key part (string)
- `enabled` - (Computed) Token is enabled (bool)
- `expired` - (Computed) Token is expired (bool)
- `name` - (Computed) Token name (string)
- `secret_key` - (Computed/Sensitive) Token secret key part (string)
- `token` - (Computed/Sensitive) Token value (string)
- `user_id` - (Computed) Token user ID (string)

» Timeouts

`rancher2_token` provides the following Timeouts configuration options:

- `create` - (Default `5 minutes`) Used for creating tokens.
- `update` - (Default `5 minutes`) Used for token modifications.
- `delete` - (Default `5 minutes`) Used for deleting tokens.

» rancher2_user

Provides a Rancher v2 User resource. This can be used to create Users for Rancher v2 environments and retrieve their information.

When a Rancher User is created, it doesn't have a global role binding. At least, `user-base` global role binding in needed in order to enable user login.

» Example Usage

```
# Create a new rancher2 User
resource "rancher2_user" "foo" {
  name = "Foo user"
  username = "foo"
  password = "changeme"
  enabled = true
}
# Create a new rancher2 global_role_binding for User
resource "rancher2_global_role_binding" "foo" {
  name = "foo"
  global_role_id = "user-base"
  user_id = "${rancher2_user.foo.id}"
}
```

» Argument Reference

The following arguments are supported:

- `username` - (Required/ForceNew) The user username (string)
- `password` - (Required/ForceNew) The user password (string)
- `name` - (Optional) The user full name (string)
- `annotations` - (Optional/Computed) Annotations for global role binding (map)
- `labels` - (Optional/Computed) Labels for global role binding (map)

» Attributes Reference

The following attributes are exported:

- `id` - (Computed) The ID of the resource (string)
- `principal_ids` - (Computed) The user principal IDs (list)

» Timeouts

`rancher2_user` provides the following Timeouts configuration options:

- `create` - (Default `5 minutes`) Used for creating users.
- `update` - (Default `5 minutes`) Used for user modifications.
- `delete` - (Default `5 minutes`) Used for deleting users.

» Import

Users can be imported using the Rancher User ID

```
$ terraform import rancher2_user.foo <user_id>
```