» Data Source: dome9_cloudaccount_AWS

Use this data source to get information about an AWS cloud account onboarded to Dome9.

» Example Usage

```
data "dome9_cloudaccount_AWS" "test" {
  id = "d9-AWS-cloud-account-id"
}
```

» Argument Reference

The following arguments are supported:

- `id` - (Required) The Dome9 id for the AWS account

» Attributes Reference

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

- `vendor` - The cloud provider ("AWS").
- `name` - The cloud account name in Dome9.
- `external_account_number` - The AWS account number.
- `error` - Credentials error status.
- `is_fetching_suspended` - Fetching suspending status.
- `creation_date` - Date account was onboarded to Dome9.
- `full_protection` - The tamper Protection mode for current security groups.
- `allow_read_only` - The AWS cloud account operation mode. true for "Manage", false for "Readonly".
- `net_sec` - The network security configuration for the AWS cloud account.
- `organizational_unit_id` - Organizational unit id.
- `IAM_safe` - IAM safe entity details
  - `AWS_group_ARN` - AWS group ARN
  - `AWS_policy_ARN` - AWS policy ARN
  - `mode` - Mode
  - `restricted_IAM_entities` - Restricted IAM safe entities which has the following:
    - `roles_ARNs` - Restricted IAM safe entities roles ARNs

* `users_ARNs` - Restricted IAM safe entities users ARNs

» Data Source: dome9_cloudaccount_azure

Use this data source to get information about an Azure cloud account onboarded to Dome9.

» Example Usage

```
data "dome9_cloudaccount_azure" "test" {
  id = "d9-azure-cloud-account-id"
}
```

» Argument Reference

The following arguments are supported:

- `id` - (Required) The Dome9 id for the Azure account.

» Attributes Reference

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

- `name` - Account name (in Dome9).
- `subscription_id` - Azure subscription id for account.
- `tenant_id` - Azure tenant id.
- `operation_mode` - Dome9 operation mode for the Azure account (Read-Only or Managed).
- `vendor` - The cloud provider (Azure).
- `creation_date` - Date Azure account was onboarded to a Dome9 account.
- `organizational_unit_id` - Organizational unit id.
- `organizational_unit_path` - Organizational unit path.
- `organizational_unit_name` - Organizational unit name.

» Data Source: dome9_cloudaccount_gcp

Use this data source to get information about a GCP cloud account onboarded to Dome9.

» Example Usage

```
data "dome9_cloudaccount_gcp" "test" {
  id = "d9-gcp-cloud-account-id"
}
```

» Argument Reference

The following arguments are supported:

- `id` - (Required) The Dome9 id for the GCP account.

» Attributes Reference

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

- `name` - GCP account name in Dome9.
- `project_id` - the Google project id (that will be onboarded).
- `creation_date` - creation date for project in Google.
- `organizational_unit_id` - Organizational unit id.
- `organizational_unit_path` - Organizational unit path.
- `organizational_unit_name` - Organizational unit name.
- `gsuite_user` - Gsuite user.
- `domain_name` - Domain name.
- `domain_name` - Azure tenant id.
- `vendor` - The cloud provider (gcp).

» Data Source: dome9_continuous_compliance_policy

Use this data source to get information about a Dome9 continuous compliance policy.

» Example Usage

```
data "dome9_continuous_compliance_policy" "test" {
  id = "d9-continuous-compliance-policy-id"
}
```

» Argument Reference

The following arguments are supported:

- `id` - (Required) The id for the cloud account in Dome9.

» Attributes Reference

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

- `cloud_account_id` - GCP account name in Dome9.
- `external_account_id` - The account number.
- `cloud_account_type` - creation date for project in Google.
- `bundle_id` - Organizational unit id.
- `notification_ids` - Organizational unit path.

» Data Source: dome9_continuous_compliance_notification

Use this data source to get information about a Dome9 continuous compliance notification policy.

» Example Usage

```
data "dome9_continuous_compliance_notification" "test" {
  id = "d9-continuous-compliance-notification-id"
}
```

» Argument Reference

The following arguments are supported:

- `id` - (Required) The id for the continuous compliance notification policy in Dome9.

» Attributes Reference

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

- `name` - Notification policy name.
- `description` - Description of the notification.
- `alerts_console` - Include in the alerts console.
- `scheduled_report` - Scheduled report details.
- `change_detection` - Change detection options.
- `gcp_security_command_center_integration` - GCP security command center details

» Data Source: dome9_iplist

Use this data source to get information about an IP list in Dome9.

» Example Usage

```
data "dome9_iplist" "test" {
  id = "d9-ip-list-id"
}
```

» Argument Reference

The following arguments are supported:

- `id` - (Required) The id of the IP list in Dome9.

» Attributes Reference

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

- `name` - IP list name.
- `description` - IP list description.
- `items` - Items (IP addresses) in the IP list.

» Data Source: dome9_ruleset

Use this data source to get information about a ruleset in Dome9.

» Example Usage

```
data "dome9_ruleset" "test" {
  id        = "d9-rule-set-id"
}
```

» Argument Reference

The following arguments are supported:

- `id` - (Required) The id of the ruleset in Dome9.

» Attributes Reference

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

- `name` - The ruleset name.
- `description` - The ruleset description.
- `cloud_vendor` - Cloud vendor that the ruleset is associated with.
- `created_time` - Rule set creation time.
- `updated_time` - Rule set last update time.
- `rules` - List of rules in the ruleset.

» Data Source: dome9_aws_security_group

Use this data source to get information about an AWS Security Group onboarded to Dome9.

» Example Usage

Basic usage:

```
data "dome9_aws_security_group" "aws_sg_ds" {
  id = "SECURITY_GROUP_ID"
}
```

» Argument Reference

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

- `dome9_security_group_name` - Name of the Security Group.
- `dome9_cloud_account_id` - Cloud account id in Dome9.
- `description` - Security Group description.
- `aws_region_id` - AWS region; in AWS format (e.g., "us-east-1").
- `is_protected` - indicates whether the Security Group is protected.
- `vpc_id` - Security Group id.
- `vpc_name` - name of VPC containing the Security Group.
- `tags` - Security Group tags.
- `services` - Security Group services.
- `cloud_account_name` - AWS cloud account name.
- `external_id` - Security Group external id.

» Data Source: dome9_azure_security_group

Use this data source to get information about an Azure Security Group onboarded to Dome9.

» Example Usage

Basic usage:

```
data "dome9_azure_security_group" "azure_sg_ds" {
  id = "SECURITY_GROUP_ID"
}
```

» Argument Reference

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

- `dome9_security_group_name` - (Required) Name of the Security Group.
- `region` - (Required) Security Group region.
- `resource_group` - (Required) Azure resource group name.
- `dome9_cloud_account_id` - (Required) Cloud account id in Dome9.
- `description` - (Optional) Security Group description.
- `is_tamper_protected` - (Optional) Is Security Group tamper protected.
- `tags` - (Optional) Security Group tags.
- `inbound` - (Optional) Security Group services.
- `outbound` - (Optional) Security Group services.

» Data Source: dome9_role

Use this data source to get information about a role in Dome9.

» Example Usage

```
data "dome9_role" "test" {
  id = "d9-role-id"
}
```

» Argument Reference

The following arguments are supported:

- `id` - (Required) The id of the role list in Dome9.

» Attributes Reference

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

- `name` - (Required) Dome9 role name.
- `description` - (Required) Dome9 role description.
- `permit_rulesets` - Is permitted permit rulesets (Optional) .
- `permit_notifications` - Is permitted permit notifications (Optional) .
- `permit_policies` - Is permitted permit policies (Optional) .
- `permit_alert_actions` - Is permitted permit alert actions (Optional) .
- `permit_on_boarding` - Is permitted permit on boarding (Optional) .
- `cross_account_access` - (Optional) Cross account access.
- `create` - (Optional) Create permission list.
- `access` - (Optional) Access permission list (SRL Type).
- `view` - (Optional) View permission list (SRL Type).
- `manage` - (Optional) Manage permission list (SRL Type).

» SRL

- `type` - (Optional) Accepted values: AWS, Azure, GCP, OrganizationalUnit.
- `main_id` - (Optional) Cloud Account or Organizational Unit ID.
- `region` - (Optional) Accepted values: "us_east_1", "us_west_1", "eu_west_1", "ap_southeast_1", "ap_northeast_1", "us_west_2", "sa_east_1", "ap_southeast_2", "eu_central_1", "ap_northeast_2", "ap_south_1", "us_east_2", "ca_central_1", "eu_west_2", "eu_west_3", "eu_north_1".
- `security_group_id` - (Optional) AWS Security Group ID.
- `traffic` - (Optional) Accepted values: "All Traffic", "All Services".

» Data Source: dome9_organizational_unit

Use this data source to get information about a Organizational Unit in Dome9.

» Example Usage

```
data "dome9_organizational_unit" "test" {
  id = "d9-organizational-unit-id"
}
```

» Argument Reference

The following arguments are supported:

- `id` - (Required) The ID of the organizational unit in Dome9.

» Attributes Reference

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

- `name` - The name of the Organizational Unit in Dome9.
- `parent_id` - The Organizational Unit parent ID.
- `account_id` - Dome9 internal account ID.
- `path` - Organizational Unit full path (IDs).
- `path_str` - Organizational Unit full path (names).
- `created` - Organizational Unit creation time.
- `updated` - Organizational Unit update time.
- `aws_cloud_accounts_count` - Number of AWS cloud accounts in the Organizational Unit.
- `azure_cloud_accounts_count` - Number of Azure cloud accounts in the Organizational Unit.
- `google_cloud_accounts_count` - Number of GCP cloud accounts in the Organizational Unit.
- `aws_aggregated_cloud_accounts_count` - Number of AWS cloud accounts in the Organizational Unit and its children.
- `azure_aggregate_cloud_accounts_count` - Number of Azure cloud accounts in the Organizational Unit and its children.
- `google_aggregate_cloud_accounts_count` - Number of GCP cloud accounts in the Organizational Unit and its children.
- `is_root` - Is Organizational Unit root.
- `is_parent_root` - Is the parent of Organizational Unit root.

» dome9_cloudaccount_AWS

This resource is used to onboard AWS cloud accounts to Dome9. This is the first and pre-requisite step in order to apply Dome9 features, such as compliance testing, on the account.

» Example Usage

Basic usage:

```
resource "dome9_cloudaccount_AWS" "test" {
  name  = "ACCOUNT NAME"
```

```
credentials  {
  ARN    = "ARN"
  secret = "SECRET"
  type   = "RoleBased"
}

organizational_unit_id = "ORGANIZATIONAL UNIT ID"

net_sec {
  regions {
    new_group_behavior = "ReadOnly"
    region             = "us_east_1"
  }
  regions {
    new_group_behavior = "ReadOnly"
    region             = "us_west_1"
  }
  regions {
    new_group_behavior = "ReadOnly"
    region             = "eu_west_1"
  }
  regions {
    new_group_behavior = "ReadOnly"
    region             = "ap_southeast_1"
  }
  regions {
    new_group_behavior = "ReadOnly"
    region             = "ap_northeast_1"
  }
  regions {
    new_group_behavior = "ReadOnly"
    region             = "us_west_2"
  }
  regions {
    new_group_behavior = "ReadOnly"
    region             = "sa_east_1"
  }
  regions {
    new_group_behavior = "ReadOnly"
    region             = "ap_southeast_2"
  }
  regions {
    new_group_behavior = "ReadOnly"
    region             = "eu_central_1"
  }
```

```
    regions {
      new_group_behavior = "ReadOnly"
      region             = "ap_northeast_2"
    }
    regions {
      new_group_behavior = "ReadOnly"
      region             = "ap_south_1"
    }
    regions {
      new_group_behavior = "ReadOnly"
      region             = "us_east_2"
    }
    regions {
      new_group_behavior = "ReadOnly"
      region             = "ca_central_1"
    }
    regions {
      new_group_behavior = "ReadOnly"
      region             = "eu_west_2"
    }
    regions {
      new_group_behavior = "ReadOnly"
      region             = "eu_west_3"
    }
    regions {
      new_group_behavior = "ReadOnly"
      region             = "eu_north_1"
    }
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of AWS account in Dome9
- `credentials` - (Required) The information needed for Dome9 System in order to connect to the AWS cloud account
- `organizational_unit_id` - (Optional) The Organizational Unit that this cloud account will be attached to

» Credentials

`credentials` has the following arguments:

- `arn` - (Required) AWS Role ARN (to be assumed by Dome9)
- `secret` - (Required) The AWS role External ID (Dome9 will have to use this secret in order to assume the role)
- `type` - (Required) The cloud account onboarding method. Set to "Role-Based".

» Network security configuration

`net_sec` has the these arguments:

- `Regions` - (Required) list of the supported regions, and their configuration:
  - `new_group_behavior` - (Required) The network security configuration. Select "ReadOnly", "FullManage", or "Reset".
  - `region` - (Required) AWS region, in AWS format (e.g., "us-east-1")

» Attributes Reference

- `id` - The id of the account in Dome9.
- `vendor` - The cloud provider ("AWS").
- `external_account_number` - The AWS account number.
- `is_fetching_suspended` - Fetching suspending status.
- `creation_date` - Date the account was onboarded to Dome9.
- `full_protection` - The protection mode for existing security groups in the account.
- `allow_read_only` - The AWS cloud account operation mode. true for "Full-Manage", false for "Readonly".
- `net_sec` - The network security configuration for the AWS cloud account. If not given, sets to default value.
- `IAM_safe` - IAM safe entity details
  - `AWS_group_ARN` - AWS group ARN
  - `AWS_policy_ARN` - AWS policy ARN
  - `mode` - Mode
  - `restricted_IAM_entities` - Restricted IAM safe entities, which have the following fields:
    - `roles_ARNs` - Restricted IAM safe entities roles ARNs
    - `users_ARNs` - Restricted IAM safe entities users ARNs

» Import

AWS cloud account can be imported; use `<AWS CLOUD ACCOUNT ID>` as the import ID.

For example:

```
terraform import dome9_cloudaccount_AWS.test 00000000-0000-0000-0000-000000000000
```

» dome9_cloudaccount_azure

This resource is used to onboard Azure cloud accounts to Dome9. This is the first and pre-requisite step in order to apply Dome9 features, such as compliance testing, on the account.

» Example Usage

Basic usage:

```
resource "dome9_cloudaccount_azure" "test" {
  name                  = "NAME"
  operation_mode        = "OPERATION MODE"
  subscription_id       = "SUBSCRIPTION ID"
  tenant_id             = "TENANT ID"
  client_id             = "CLIENT ID"
  client_password       = "CLIENT PASSWORD"
  organizational_unit_id = "ORGANIZATIONAL UNIT ID"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the Azure account in Dome9
- `operation_mode` - (Required) Dome9 operation mode for the Azure account ("Read-Only" or "Managed")
- `subscription_id` - (Required) The Azure subscription id for account
- `tenant_id` - (Required) The Azure tenant id
- `client_id` - (Required) Azure account id
- `client_password` - (Required) Password for account*
- `organizational_unit_id` - (Optional) Organizational Unit that this cloud account will be attached to

» Attributes Reference

- `id` - The ID of the Azure cloud account
- `vendor` - The cloud provider ("Azure")
- `creation_date` - Date the account was onboarded to Dome9
- `organizational_unit_path` - Organizational unit path
- `organizational_unit_name` - Organizational unit name

» Import

Azure cloud account can be imported; use `<Azure CLOUD ACCOUNT ID>` as the import ID.

For example:

```
terraform import dome9_cloudaccount_Azure.test 00000000-0000-0000-0000-000000000000
```

» dome9_cloudaccount_gcp

This resource is used to onboard GCP cloud accounts to Dome9. This is the first and pre-requisite step in order to apply Dome9 features, such as compliance testing, on the account.

» Example Usage

Basic usage:

```
resource "dome9_cloudaccount_gcp" "gcp_ca" {
  name                = "sandbox"
  project_id          = "ID"
  private_key_id      = "PRIVATE"
  private_key         = "KEY"
  client_email        = "EMAIL@ADDRESS.COM"
  client_id           = "CID"
  client_x509_cert_url = "https://www.googleapis.com/oauth2/v1/certs"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Google account name in Dome9
- `project_id` - (Required) Project ID

- `private_key_id` - (Required) Private key ID
- `private_key` - (Required) Private key
- `client_email` - (Required) GCP client email
- `client_id` - (Required) Client id
- `client_x509_cert_url` - (Required) client x509 certificate URL
- `gsuite_user` - (Optional) The Gsuite user
- `domain_name` - (Optional) The domain name
- `organizational_unit_id` - (Optional) Organizational Unit that this cloud account will be attached to

» Attributes Reference

- `id` - The ID of the GCP cloud account
- `creation_date` - creation date for project in Google.
- `vendor` - The cloud provider (gcp).
- `organizational_unit_path` - Organizational unit path.
- `organizational_unit_name` - Organizational unit name.

» Import

GCP cloud account can be imported; use `<GCP CLOUD ACCOUNT ID>` as the import ID.

For example:

```
terraform import dome9_cloudaccount_gcp.test 00000000-0000-0000-0000-000000000000
```

» dome9_continuous_compliance_policy

This resource is used to create and modify compliance policies in Dome9 for continuous compliance assessments. A continuous compliance policy is the combination of a Rule Bundle applied to a specific cloud account.

» Example Usage

Basic usage:

```
resource "dome9_continuous_compliance_policy" "test_policy" {
  cloud_account_id    = "CLOUD ACCOUNT ID"
  external_account_id = "EXTERNAL ACCOUNT ID"
  bundle_id           = 00000
  cloud_account_type  = "CLOUD ACCOUNT TYPE"
  notification_ids    = ["NOTIFICATION IDS"]
```

```
}
```

» Argument Reference

The following arguments are supported:

- `cloud_account_id` - (Required) The cloud account id.
- `external_account_id` - (Required) The account number.
- `bundle_id` - (Required) The bundle id for the bundle that will be used in the policy.
- `cloud_account_type` - (Required) The cloud account provider ("Aws", "Azure", "Google").
- `notification_ids` - (Required) The notification policy id's for the policy [list].

» Attributes Reference

- `id` - Id of the compliance policy.

» Import

The policy can be imported; use `<POLICY ID>` as the import ID.

For example:

```
terraform import dome9_continuous_compliance_policy.test 00000000-0000-0000-0000-0000000000
```

» dome9_continuous_compliance_notification

This resource is used to create and modify Dome9 notification policies for Continuous Compliance assessments of cloud accounts. Continuous assessments apply bundles of compliance rules to your cloud account continuously, and send notifications of issues according to the Notification Policy.

» Example Usage

Basic usage:

```
resource "dome9_continuous_compliance_notification" "test_notification" {
  name           = "NAME"
  description    = "DESCRIPTION"
  alerts_console = "ALERTS_CONSOLE"
```

```
  change_detection {
    email_sending_state                = "EMAIL_SENDING_STATE"
    email_per_finding_sending_state    = "EMAIL_PER_FINDING_SENDING_STATE"
    sns_sending_state                  = "SNS_SENDING_STATE"
    external_ticket_creating_state     = "EXTERNAL_TICKET_CREATING_STATE"
    aws_security_hub_integration_state = "AWS_SECURITY_HUB_INTEGRATION_STATE"
    webhook_integration_state          = "WEBHOOK_INTEGRATION_STATE"

    email_data {
      recipients = ["RECIPIENTS"]
    }

    email_per_finding_data {
      recipients                 = ["RECIPIENTS"]
      notification_output_format = "NOTIFICATION_OUTPUT_FORMAT"
    }
  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The cloud account id in Dome9.
- `description` - (Optional) Description of the notification.

at least one of `alerts_console`, `scheduled_report`, or `change_detection` must be included

- `alerts_console` - (Optional) send findings (also) to the Dome9 web app alerts console (Boolean); default is False.
- `scheduled_report` - Scheduled email report notification block:
  - `email_sending_state` - send schedule report of findings by email; can be ”Enabled” or ”Disabled”.

    if `email_sending_state` is Enabled, the following must be included:

    * `schedule_data` - Schedule details:
      · `cron_expression` - the schedule to issue the email report (in cron expression format)
      · `type` - type of report; can be ”Detailed”, ”Summary”, ”FullCsv” or ”FullCsvZip”
      · `recipients` - comma-separated list of email recipients
- `change_detection` - Send changes in findings options:

- `email_sending_stat` - send email report of changes in findings; can be "Enabled" or "Disabled".

  if `email_sending_stat` is Enabled, the following must be included:

  * `email_data` - Email notification details:
    - `recipients` - comma-separated list of email recipients

- `email_per_finding_sending_state` - send separate email notification for each finding; can be "Enabled" or "Disabled"

  if `email_per_finding_sending_state` is Enabled, the following must be included:

  * `email_per_finding_data` - Email per finding notification details:
    - `recipients` - comma-separated list of email recipients
    - `notification_output_format` - (Required) format of JSON block for finding; can be "JsonWithFullEntity", "JsonWithBasicEntity", or "PlainText".

- `sns_sending_state` - send by AWS SNS for each new finding; can be "Enabled" or "Disabled".

  if `sns_sending_state` is Enabled, the following must be included:

  * `sns_data` - SNS notification details:
    - `sns_topic_arn` - SNS topic ARN
    - `sns_output_format` - SNS output format; can be "JsonWithFullEntity", "JsonWithBasicEntity", or "PlainText".

- `external_ticket_creating_state` - send each finding to an external ticketing system; can be "Enabled" or "Disabled".

  if `external_ticket_creating_state` is Enabled, the following must be included:

  * `ticketing_system_data` - Ticketing system details:
    - `system_type` - system type; can be "ServiceOne", "Jira", or "PagerDuty"
    - `should_close_tickets` - ticketing system should close tickets when resolved (bool)
    - `domain` - ServiceNow domain name (ServiceNow only)
    - `user` - user name (ServiceNow only)
    - `pass` - password (ServiceNow only)
    - `project_key` - project key (Jira) or API Key (PagerDuty)
    - `issue_type` - issue type (Jira)

- `webhook_integration_state` - send findings to an HTTP endpoint (webhook); can be "Enabled" or "Disabled".

if `webhook_integration_state` is Enabled, the following must be included:

* `webhook_data` - Webhook data block supports:
  - `url` - HTTP endpoint URL
  - `http_method` - HTTP method, "Post" by default.
  - `auth_method` - authentication method; "NoAuth" by default
  - `username` - username in endpoint system
  - `password` - password in endpoint system
  - `format_type` - format for JSON block for finding; can be "Basic" or "ServiceNow"

– `aws_security_hub_integration_state` - send findings to AWS Secure Hub; can be "Enabled" or "Disabled".

if `aws_security_hub_integration_state` is Enabled, the following must be included:

* `aws_security_hub_integration` - AWS security hub integration block supports:
  - `external_account_id` - (Required) external account id
  - `region` - (Required) AWS region

`gcp_security_command_center_integration` is a change_detection option

- `gcp_security_command_center_integration` - GCP security command center details

  – `state` - send findings to the GCP Security Command Center; can be "Enabled" or "Disabled"

  if `state` is Enabled, the following must be included:

  * `project_id` - GCP Project id
  * `source_id` - GCP Source id

» Import

The notification can be imported; use `<NOTIFICATION ID>` as the import ID.

For example:

```
terraform import dome9_continuouscompliance_notification.test 00000000-0000-0000-0000-000000
```

» dome9_iplist

This resource is used to create and manage IP lists in Dome9. IP lists are groups of IP addresses (typically in customer cloud environments), on which common

actions are applied. For example, a Security Group could be applied to a list, instead of applying it to each IP address in the list individually.

» Example Usage

Basic usage:

```
resource "dome9_iplist" "iplist" {
  name        = "NAME"
  description = "DESCRIPTION"

  items = [
    {
      ip      = "1.1.1.1"
      comment = "COMMENT1"
    },
    {
      ip      = "2.2.2.2"
      comment = "COMMENT2"
    },
  ]
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the IP list in Dome9
- `description` - (Optional) A description of the list (what it represents); defaults to empty string
- `items` - (Optional) the individual IP addresses for the list; defaults to empty list

» Items

The `items` supports the following arguments:

- `ip` - (Optional) IP address
- `comment` - (Optional) Comment

» Attributes Reference

- `id` - IP list Id

» Import

IP list can be imported; use `<IP LIST ID>` as the import ID.

For example:

```
terraform import dome9_iplist.test 00000
```

» dome9_ruleset

This resource is used to create and manage rulesets in Dome9. Rulesets are sets of compliance rules.

» Example Usage

Basic usage:

```
resource "dome9_ruleset" "ruleset" {
  name        = "some_ruleset"
  description = "this is the descrption of my ruleset"
  cloud_vendor = "aws"
  language = "en"
  hide_in_compliance = false
  is_template = false
  rules {
    name = "some_rule2"
    logic = "EC2 should x"
    severity = "High"
    description = "rule description here"
    compliance_tag = "ct"

  }
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the ruleset in Dome9.
- `description` - (Optional) A description of the ruleset (what it represents); defaults to empty string.
- `cloud_vendor` - (Required) Cloud vendor that the ruleset is associated with, can be one of the following: `aws`, `azure` or `google`.
- `language` - (Optional) Language of the rules; defaults to 'en' (English).

» Rules

The `rules` supports the following arguments:

- `name` - (Required) Rule name
- `logic` - (Optional) Rule GSL logic. This is the text of the rule, using Dome9 GSL syntax
- `severity` - (Optional) Rule severity
- `description` - (Optional) Rule description
- `compliance_tag` - (Optional) A reference to a compliance standard

» Attributes Reference

- `id` - Ruleset Id

» Import

Ruleset can be imported; use `<RULE SET ID>` as the import ID.

For example:

```
terraform import dome9_rule_set.test 00000
```

» dome9_aws_security_group

This resource has methods to add and manage Security Groups in a cloud account that is managed by Dome9.

» Example Usage

Basic usage:

```
resource "dome9_aws_security_group" "aws_sg" {
  dome9_security_group_name = "dome9_security_group_name"
  description               = "description"
  aws_region_id             = "aws_region_id"
  dome9_cloud_account_id    = "dome9_cloud_account_id"

  services {
    inbound {
      name          = "FIRST_INBOUND_SERVICE_NAME"
      description   = "DESCRIPTION"
      protocol_type = "PROTOCOL_TYPE"
```

```
      port          = "PORT"
      open_for_all  = false
      scope {
        type = "TYPE"
        data = {
          cidr = "CIDR"
          note = "NOTE"
        }
      }
    }
    outbound {
      name          = "NAME"
      description   = "DESCRIPTION"
      protocol_type = "PROTOCOL_TYPE"
      port          = ""
      open_for_all  = true
    }
  }

  tags = {
    tag-key = "TAG-VALUE"
  }
}
```

» Argument Reference

The following arguments are supported:

- `dome9_security_group_name` - (Required) Name of the Security Group.
- `dome9_cloud_account_id` - (Required) Cloud account id in Dome9.
- `description` - (Optional) Security Group description.
- `aws_region_id` - (Optional) AWS region, in AWS format (e.g., "us-east-1"); default is us_east_1.
- `is_protected` - (Optional) Indicates the Security Group is in Protected mode.
  - Note: to set the protection mode, first create the Security Group, then update it with the desired protection mode value ('true' for Protected).
- `vpc_id` - (Optional) VPC id for VPC containing the Security Group.
- `vpc_name` - (Optional) Security Group VPC name.
- `tags` - (Optional) Security Group tags.
- `services` - (Optional) Security Group services.

» Security Group services

`services` has the these arguments:

- `inbound` - (Required) inbound service.
- `outbound` - (Required) outbound service.

The configuration of inbound and outbound is: * `name` - (Required) Service name. * `description` - (Optional) Service description. * `protocol_type` - (Required) Service protocol type. Select from "ALL", "HOPOPT", "ICMP", "IGMP", "GGP", "IPV4", "ST", "TCP", "CBT", "EGP", "IGP", "BBN_RCC_MON", "NVP2", "PUP", "ARGUS", "EMCON", "XNET", "CHAOS", "UDP", "MUX", "DCN_MEAS", "HMP", "PRM", "XNS_IDP", "TRUNK1", "TRUNK2", "LEAF1", "LEAF2", "RDP", "IRTP", "ISO_TP4", "NETBLT", "MFE_NSP", "MERIT_INP", "DCCP", "ThreePC", "IDPR", "XTP", "DDP", "IDPR_CMTP", "TPplusplus", "IL", "IPV6", "SDRP", "IPV6_ROUTE", "IPV6_FRAG", "IDRP", "RSVP", "GRE", "DSR", "BNA", "ESP", "AH", "I_NLSP", "SWIPE", "NARP", "MOBILE", "TLSP", "SKIP", "ICMPV6", "IPV6_NONXT", "IPV6_OPTS", "CFTP", "SAT_EXPAK", "KRYPTOLAN", "RVD", "IPPC", "SAT_MON", "VISA", "IPCV", "CPNX", "CPHB", "WSN", "PVP", "BR_SAT_MON", "SUN_ND", "WB_MON", "WB_EXPAK", "ISO_IP", "VMTP", "SECURE_VMTP", "VINES", "TTP", "NSFNET_IGP", "DGP", "TCF", "EIGRP", "OSPFIGP", "SPRITE_RPC", "LARP", "MTP", "AX25", "IPIP", "MICP", "SCC_SP", "ETHERIP", "ENCAP", "GMTP", "IFMP", "PNNI", "PIM", "ARIS", "SCPS", "QNX", "AN", "IPCOMP", "SNP", "COMPAQ_PEER", "IPX_IN_IP", "VRRP", "PGM", "L2TP", "DDX", "IATP", "STP", "SRP", "UTI", "SMP", "SM", "PTP", "ISIS", "FIRE", "CRTP", "CRUDP", "SSCOPMCE", "IPLT", "SPS", "PIPE", "SCTP", "FC", "RSVP_E2E_IGNORE", "MOBILITY_HEADER", "UDPLITE", "MPLS_IN_IP", "MANET", "HIP", "SHIM6", "WESP" or "ROHC". * `port` - (Optional) Service type (port). * `open_for_all` - (Optional) Is open for all. * `scope` - (Optional) Service scope which has the following configuration: * `type` - (Required) scope type. * `data` - (Required) scope data.

» Attributes Reference

- `cloud_account_name` - AWS cloud account name.
- `external_id` - Security Group external id.
- Note: Just the following fields can be updated: services (inbound / outbound), tags and protection mode.

» Import

The security group can be imported; use `<SESCURITY GROUP ID>` as the import

ID.

For example:

```
terraform import dome9_aws_security_group.test 00000000-0000-0000-0000-000000000000
```

» dome9_azure_security_group

The Azure Security Group resource has methods to add and manage Azure Security Group policies for Azure cloud accounts that are managed by Dome9.

» Example Usage

Basic usage:

```
resource "dome9_azure_security_group" "azure_sg" {
  dome9_security_group_name = "dome9_security_group_name"
  region                    = "australiaeast"
  resource_group            = "resource_group"
  dome9_cloud_account_id    = "dome9_cloud_account_id"

  description         = "description"
  is_tamper_protected = false

  inbound {
    name               = "name"
    description        = "description"
    priority           = 1000
    access             = "Allow"
    protocol           = "TCP"
    source_port_ranges = ["*"]

    source_scopes {
      type = "Tag"
      data = {
        name = "VirtualNetwork"
      }
    }
    destination_port_ranges = ["20-90"]

    destination_scopes {
      type = "CIDR"
      data = {
        cidr = "0.0.0.0/0"
        note = "Any"
```

```
      }
    }
    is_default = false
  }
}
```

» Argument Reference

The following arguments are supported:

- `dome9_security_group_name` - (Required) Name of the security group.
- `region` - (Required) Region can be one of the following: `centralus`, `eastus`, `eastus2`, `usgovlowa`, `usgovvirginia`, `northcentralus`, `southcentralus`, `westus`, `westus2`, `westcentralus`, `northeurope`, `westeurope`, `eastasia`, `southeastasia`, `japaneast`, `japanwest`, `brazilsouth`, `australiaeast`, `australiasoutheast`, `centralindia`, `southindia`, `westindia`, `chinaeast`, `chinanorth`, `canadacentral`, `canadaeast`, `germanycentral`, `germanynortheast`, `koreacentral`, `uksouth`, `ukwest`, `koreasout`
- `resource_group` - (Required) Azure resource group name.
- `dome9_cloud_account_id` - (Required) Cloud account id in Dome9.
- `description` - (Optional) Security group description.
- `is_tamper_protected` - (Optional) Is security group tamper protected.
- `tags` - (Optional) Security group tags list of `key`, `value`:
  - `key` - (Required) Tag key.
  - `value` - (Required) Tag value.
- `inbound` - (Optional) Security group services.
- `outbound` - (Optional) Security group services.

The configuration of inbound and outbound is: * `name` - (Required) Service name. * `description` - (Optional) Service description. * `priority` - (Required) Service priority (a number between 100 and 4096) * `access` - (Optional) Service access (Allow / Deny). * `protocol` - (Required) Service protocol (UDP / TCP / ANY). * `source_port_ranges` - (Required) Source port ranges. * `destination_port_ranges` - (Required) Destination port ranges. * `source_scopes` - (Required) List of source scopes for the service (CIDR / IP List / Tag): * `type` - (Required) scope type. * `data` - (Required) scope data. * `destination_scopes` - (Required) List of destination scopes for the service (CIDR / IP List / Tag). * `type` - (Required) scope type. * `data` - (Required) scope data. * `is_default` - Gets or sets the default security rules of network security group.

» Attributes Reference

- `external_security_group_id` - Azure external security group id.

- `cloud_account_name` - Azure cloud account name.
- `last_updated_by_dome9` - Last updated by dome9.

» Import

The security group can be imported; use `<SESCURITY GROUP ID>` as the import ID.

For example:

```
terraform import dome9_azure_security_group.test 00000000-0000-0000-0000-000000000000
```

» dome9_role

The Role resource is used to create and manage Dome9 roles. Roles are used to manage access permissions for Dome9 users.

» Example Usage

Basic usage:

```
resource "dome9_role" "role_rs" {
  name        = "ROLE_NAME"
  description = "ROLE_DESC"
  access {
    type              = "AWS"
    main_id           = "MAIN_ID"
    region            = "us_east_1"
    security_group_id = "SECURITY_GROUP_ID"
    traffic           = "All Traffic"
  }
  access {
    type    = "OrganizationalUnit"
    main_id = "00000000-0000-0000-0000-000000000000"
  }

  permit_notifications = false
  permit_rulesets      = false
  permit_policies      = false
  permit_alert_actions = false
  permit_on_boarding   = false
  create               = []
  cross_account_access = []
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) Dome9 role name.
- `description` - (Required) Dome9 role description.
- `permit_rulesets` - Is permitted permit rulesets (Optional) .
- `permit_notifications` - Is permitted permit notifications (Optional) .
- `permit_policies` - Is permitted permit policies (Optional) .
- `permit_alert_actions` - Is permitted permit alert actions (Optional) .
- `permit_on_boarding` - Is permitted permit on boarding (Optional) .
- `cross_account_access` - (Optional) Cross account access.
- `create` - (Optional) Create permission list.
- `access` - (Optional) Access permission list (SRL Type).
- `view` - (Optional) View permission list (SRL Type).
- `manage` - (Optional) Manage permission list (SRL Type).

» SRL

- `type` - (Optional) Accepted values: AWS, Azure, GCP, OrganizationalUnit.
- `main_id` - (Optional) Cloud Account or Organizational Unit ID.
- `region` - (Optional) Accepted values: "us_east_1", "us_west_1", "eu_west_1", "ap_southeast_1", "ap_northeast_1", "us_west_2", "sa_east_1", "ap_southeast_2", "eu_central_1", "ap_northeast_2", "ap_south_1", "us_east_2", "ca_central_1", "eu_west_2", "eu_west_3", "eu_north_1".
- `security_group_id` - (Optional) AWS Security Group ID.
- `traffic` - (Optional) Accepted values: "All Traffic", "All Services".
- Note: to create a role, create it with no permissions, then updated it with the desired permissions.

  To understand the roles/permissions CLICK HERE.

» Import

IP role can be imported; use `<ROLE ID>` as the import ID.

For example:

```
terraform import dome9_role.role_rs 00000
```

» dome9_ruleset

This resource is used to create and manage Organizational Unit in Dome9. An Organizational Unit is a group of cloud accounts representing, for example, a business unit or geographical region.

» Example Usage

Basic usage:

```
resource "dome9_organizational_unit" "test_ou" {
  name      = "some_organizational_unit"
  parent_id = "00000000-0000-0000-0000-000000000000"
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the organizational unit in Dome9.
- `parent_id` - (Optional) The organizational unit parent ID.

» Attributes Reference

- `id` - Organizational unit Id
- `account_id` - Dome9 internal account ID.
- `path` - Organizational Unit full path (IDs).
- `path_str` - Organizational Unit full path (names).
- `created` - Organizational Unit creation time.
- `updated` - Organizational Unit update time.
- `aws_cloud_accounts_count` - Number of AWS cloud accounts in the Organizational Unit.
- `azure_cloud_accounts_count` - Number of Azure cloud accounts in the Organizational Unit.
- `google_cloud_accounts_count` - Number of GCP cloud accounts in the Organizational Unit.
- `aws_aggregated_cloud_accounts_count` - Number of AWS cloud accounts in the Organizational Unit and its children.
- `azure_aggregate_cloud_accounts_count` - Number of Azure cloud accounts in the Organizational Unit and its children.
- `google_aggregate_cloud_accounts_count` - Number of GCP cloud accounts in the Organizational Unit and its children.
- `is_root` - Is Organizational Unit root.
- `is_parent_root` - Is the parent of Organizational Unit root.

» Import

Organizational unit can be imported; use `<ORGANIZATIONAL UNIT ID>` as the import ID.

For example:

```
terraform import dome9_organizational_unit.test 00000
```

» dome9_attach_iam_safe

Attach IAM safe to AWS cloud account.

» Example Usage

Basic usage:

```
resource "dome9_attach_iam_safe" "test" {
  aws_cloud_account_id = "00000000-0000-0000-0000-000000000000"
  aws_group_arn        = "AWS_GROUP_ARN"
  aws_policy_arn       = "AWS_POLICY_ARN"
}
```

» Argument Reference

The following arguments are supported:

- `aws_cloud_account_id` - (Required) AWS cloud account to attach IAM safe to it.
- `aws_group_arn` - (Required) AWS group arn.
- `aws_policy_arn` - (Required) AWS policy arn.

» Attributes Reference

- `mode` - Mode.

» Import

Cloud account IAM safe can be imported; use `<AWS CLOUD ACCOUNT ID>` as the import ID.

For example:

```
terraform import dome9_attach_iam_safe_re.test 00000000-0000-0000-0000-000000000000
```

» dome9_iam_safe_entity

Protect cloud accounts that are managed by Dome9. Control access to them with targeted short-term authorizations (involving the Dome9 mobile app).

» Example Usage

Basic usage:

```
resource "dome9_iam_safe_entity" "dome9_iam_safe_entity_re" {
  protection_mode           = "ProtectWithElevation"
  entity_type               = "User"
  entity_name               = "ENTITY_NAME"
  aws_cloud_account_id      = "00000000-0000-0000-0000-000000000000"
  dome9_users_id_to_protect = ["000000", "111111"]
}
```

» Argument Reference

The following arguments are supported:

- `protection_mode` - (Required) Protection mode; can be "Protect", "ProtectWithElevation".
- `entity_type` - (Required) Entity type to protect; can be "User", "Role".
- `aws_cloud_account_id` - (Required) AWS cloud account id to protect.
- `entity_name` - (Required) AWS IAM user or role name to protect.
- `dome9_users_id_to_protect` - (Optional) When ProtectWithElevation mode selected, dome9 users ids must be provided.
- Note: To following filed can be updated:
  - `protection_mode`: Switch between `Protect` to `ProtectWithElecation` mode.
  - `dome9_users_id_to_protect`: Update the dome9 users list that can evaluate the aws users or roles. Empty list with switch it from `Protect` to `ProtectWithElevation` mode.

» Attributes Reference

- `state` - Can be one of the following: `Unattached`, `Attached` or `Restricted`.
- `attached_dome9_users` - List of users in protect with elevation mode.
- `exists_in_aws` - Is exist in aws.

- `arn` - Role or User arn.