

» **logicmonitor__collectors**

Use this datasource to get the ID of an available collector.

» **Example Usage**

```
# Look up a LogicMonitor collector id
data "logicmonitor_collectors" "collectors" {
  filters {
    "property" = "hostname"
    "operator" = "~"
    "value" = "test"
  },
  "most_recent" = true
}
```

» **Argument Reference**

The following arguments are supported:

- **size** - (Optional) The number of results to display. Max is 1000. Default is 50
- **offset** - (Optional) The number of results to offset the displayed results by. Default is 0
- **most_recent** - (Optional) The most recent collector installed that is online
- **filters** - (Optional) Filters the response according to the operator and value specified. Note that you can use * to match on more than one character. More Info: <https://www.logicmonitor.com/support/rest-api-developers-guide/device-groups/get-device-groups/>

» **Nested filters blocks**

Nested **filters** blocks have the following structure: **property{operator}value**
* **property** - (Required if using filters) The name of filtered property. Currently the properties supported are **hostname** and **description**
* **operator** - (Required if using filters) The type of operator. Currently the operators supported are : ~ !: !~ *
* **value** - (Required if using filters) The value of the filtered property.

» **logicmonitor__device__group**

Use this datasource to get the ID of an available device group.

» Example Usage

```
# Look up a LogicMonitor device group id
data "logicmonitor_device_group" "devicegroups" {
  filters {
    "property" = "name"
    "operator" = ":"
    "value" = "Production"
  },

  filters {
    "custom_property_name" = "app.user"
    "operator" = ":"
    "custom_property_value" = "api"
  }
}
```

» Argument Reference

The following arguments are supported:

- **size** - (Optional) The number of results to display. Max is 1000. Default is 50
- **offset** - (Optional) The number of results to offset the displayed results by. Default is 0
- **filters** - (Optional) Filters the response according to the operator and value specified. Note that you can use * to match on more than one character. More Info: <https://www.logicmonitor.com/support/rest-api-developers-guide/device-groups/get-device-groups/>

» Nested filters blocks

Nested **filters** blocks have the following structure: **property{operator}value**
* **property** - (Required if using filters) The name of filtered property. * **operator** - (Required if using filters) The type of operator. * **value** - (Required if using filters) The value of the filtered property.

You can also do custom properties * **custom_property_name** - (Required if using filters and custom properties) The name of filtered custom property. * **operator** - (Required if using filters) The type of operator. * **custom_property_value** - (Required if using filters and custom properties) The value of the filtered custom property.

» **logicmonitor__collector**

Provides a LogicMonitor collector resource. This can be used to create and manage LogicMonitor collectors.

Note: This resource will only create the collector device in your account. See Downloading a Collector Installer for information on how to download and install an existing collector.

» **Example Usage**

```
# Create a new LogicMonitor collector
resource "logicmonitor__collector" "collector1" {
  description      = "my terraformed collector"
  enable_failback = true
}
```

» **Argument Reference**

The following arguments are supported:

- **backup_collector_id** - (Optional) The Id of the failover Collector configured for this Collector
- **collector_group_id** - (Optional) The Id of the group the Collector is in
- **description** - (Optional) The Collector's description
- **enable_failback** - (Optional) Whether or not automatic failback is enabled for the Collector
- **enable_collector_device_failover** - (Optional) Whether or not the device the Collector is installed on is enabled for fail over
- **escalation_chain_id** - (Optional) The Id of the escalation chain associated with this Collector
- **resend_interval** - (Optional) The interval, in minutes, after which alert notifications for the Collector will be resent
- **suppress_alert_clear** - (Optional) Whether alert clear notifications are suppressed for the Collector

» **logicmonitor__collector__group**

Provides a LogicMonitor collector group resource. This can be used to create and manage LogicMonitor collector groups

» Example Usage

```
# Create a new LogicMonitor collector group
resource "logicmonitor_collector_group" "group1" {
  name = "collector_group_1"
  description = "a new test group"
}
```

» Argument Reference

The following arguments are supported:

- **name** - (Required) Name of collector group
- **description** - (Optional) Set description of collector group

» logicmonitor_device

Provides a LogicMonitor device resource. This can be used to create and manage LogicMonitor devices

» Example Usage

```
# Create a new LogicMonitor device
resource "logicmonitor_device" "host" {
  ip_addr = "10.32.12.18"
  disable_alerting = true
  collector = "2"
  properties {
    "app" = "haproxy"
    "system.categories" = "a,b,c,d"
  }
}

# Create a new LogicMonitor device and device group with some data source lookups and computed values
resource "logicmonitor_device" "host" {
  ip_addr = "10.32.12.18"
  disable_alerting = true
  collector = "${data.logicmonitor_collectors.collectors.id}"
  hostgroup_id = "${logicmonitor_device_group.group1.id}"
  properties {
    "app" = "haproxy"
    "system.categories" = "a,b,c,d"
  }
}
```

```

}

resource "logicmonitor_device_group" "group1" {
  name = "newgroup"
  properties {
    "system.categories" = "ec2"
    "jmx.port" = "3008"
    "snmp.version" = "v2c"
  }
}

data "logicmonitor_collectors" "collectors" {
  most_recent = true
}

```

» Argument Reference

The following arguments are supported:

- **ip_addr** - (Required) Ip Address/Hostname of device
- **collector** - (required) The id of the collector that will monitoring the device
- **display_name** - (Optional) Display name of device, (default is ip_addr)
- **disable_alerting** - (Optional) The host is created with alerting disabled (default is true)
- **hostgroup_id** - (Optional) The host group id that specifies which group the device belongs to (multiple host group ids can be added, represented by a comma separated string)
- **properties** - (Optional) The properties associated with this device group. Any string value pair will work (see example).

» Import

Devices can be imported using their device id or ip address/dns name

```

$ terraform import logicmonitor_device.host 751
$ terraform import logicmonitor_device.host server01.us-east-1.logicmonitor.net

```

» logicmonitor__device__group

Provides a LogicMonitor device group resource. This can be used to create and manage LogicMonitor device groups

» Example Usage

```
# Create a new LogicMonitor device group
resource "logicmonitor_device_group" "group" {
  name = "NewTestGroup"
  description = "new test group"
  properties {
    "group" = "test"
    "system.categories" = "a,b,c,d"
  }
}

# Create a new LogicMonitor dynamic device group
resource "logicmonitor_device_group" "group1" {
  name = "NewDynamicGroup"
  description = "new dynamic group"
  applies_to = "system.displayname =~ \"Prod\""
}
```

» Argument Reference

The following arguments are supported:

- **name** - (Required) Name of device group
- **description** - (Optional) Description of device group
- **parent_id** - (Optional) The id of the parent group for this device group (the root device group has an Id of 1)
- **applies_to** - (Optional) The Applies to custom query for this group. Setting this field will make this a dynamic group.
- **disable_alerting** - (Optional) Indicates whether alerting is disabled (true) or enabled (false) for this device group
- **properties** - (Optional) The properties associated with this device group. Any string value pair will work (see example).

» Import

Device Groups can be imported using their group id or full path

```
$ terraform import logicmonitor_device_group.group1 451
$ terraform import logicmonitor_device_group.group1 Production/SBA/Linux
```