» oneandone_instance_size

Fetches a predefined instance type for 1&1 servers

» Example Usage

```
data "oneandone_instance_size" "sizeByName" {
  name = "L"
}

data "oneandone_instance_size" "sizeByHardware" {
  vcores = 2
  ram = 4
}

resource "oneandone_server" "server" {
  name                = "Example"
  image               = "debian8-64min"
  datacenter          = "DE"
  fixed_instance_size = "${data.oneandone_instance_size.sizeByName.id}"
  ...
}
```

» Argument Reference

The following arguments are supported, at least one is required:

- `name` -(Optional) Number of cores per processor
- `ram` - (Optional) Size of ram in GB
- `vcores` - (Optional) Number of vcores

It exposes the following attributes

- `coresPerProcessor` - (Computed) The number of vcores per processor
- `id` - (Computed) The ID of the instance type
- `name` - (Computed) The Name of the instance type
- `ram` - (Computed) The size of the ram in GB
- `vcores` - (Computed) The number of vcores

» oneandone_server

Manages a Firewall Policy on 1&1

» Example Usage

```
resource "oneandone_firewall_policy" "fw" {
  name = "test_fw_011"
  rules = [
    {
      "protocol" = "TCP"
      "port_from" = 80
      "port_to" = 80
      "source_ip" = "0.0.0.0"
    },
    {
      "protocol" = "ICMP"
      "source_ip" = "0.0.0.0"
    },
    {
      "protocol" = "TCP"
      "port_from" = 43
      "port_to" = 43
      "source_ip" = "0.0.0.0"
    },
    {
      "protocol" = "TCP"
      "port_from" = 22
      "port_to" = 22
      "source_ip" = "0.0.0.0"
    }
  ]
}
```

» Argument Reference

The following arguments are supported:

- `description` - (Optional) Description for the VPN
- `name` - (Required) The name of the VPN.

Firewall Policy Rules (`rules`) support the follwing:

- `protocol` - (Required) The protocol for the rule. Allowed values are `TCP`, `UDP`, `TCP/UDP`, `ICMP` and `IPSEC`.
- `port_from` - (Optional) Defines the start range of the allowed port
- `port_to` - (Optional) Defines the end range of the allowed port
- `source_ip` - (Optional) Only traffic directed to the respective IP address

» oneandone_server

Manages a Load Balancer on 1&1

» Example Usage

```
resource "oneandone_loadbalancer" "lb" {
  name = "test_lb"
  method = "ROUND_ROBIN"
  persistence = true
  persistence_time = 60
  health_check_test = "TCP"
  health_check_interval = 300
  datacenter = "GB"
  rules = [
    {
      protocol = "TCP"
      port_balancer = 8080
      port_server = 8089
      source_ip = "0.0.0.0"
    },
    {
      protocol = "TCP"
      port_balancer = 9090
      port_server = 9099
      source_ip = "0.0.0.0"
    }
  ]
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the load balancer.
- `description` - (Optional) Description for the load balancer
- `method` - (Required) Balancing procedure Can be `ROUND_ROBIN` or `LEAST_CONNECTIONS`
- `datacenter` - (Optional) Location of desired 1and1 datacenter. Can be `DE`, `GB`, `US` or `ES`
- `persistence` - (Optional) True/false defines whether persistence should be turned on/off
- `persistence_time` - (Optional) Persistence duration in seconds
- `health_check_test` - (Optional) Can be `TCP` or`ICMP`.

- `health_check_interval` - (Optional)
- `health_check_path` - (Optional)
- `health_check_path_parser` - (Optional)

Loadbalancer rules (`rules`) support the following

- `protocol` - (Required) The protocol for the rule. Allowed values are `TCP`, `UDP`, `TCP/UDP`, `ICMP` and `IPSEC`.
- `port_balancer` - (Required)
- `port_server` - (Required)
- `source_ip` - (Required)

» oneandone_server

Manages a Monitoring Policy on 1&1

» Example Usage

```
resource "oneandone_monitoring_policy" "mp" {
  name = "test_mp"
  agent = true
  email = "jasmin@stackpointcloud.com"

  thresholds = {
    cpu = {
      warning = {
        value = 50,
        alert = false
      }
      critical = {
        value = 66,
        alert = false
      }

    }
    ram = {
      warning = {
        value = 70,
        alert = true
      }
      critical = {
        value = 80,
        alert = true
      }
```

```
  },
  ram = {
    warning = {
      value = 85,
      alert = true
    }
    critical = {
      value = 95,
      alert = true
    }
  },
  disk = {
    warning = {
      value = 84,
      alert = true
    }
    critical = {
      value = 94,
      alert = true
    }
  },
  transfer = {
    warning = {
      value = 1000,
      alert = true
    }
    critical = {
      value = 2000,
      alert = true
    }
  },
  internal_ping = {
    warning = {
      value = 3000,
      alert = true
    }
    critical = {
      value = 4000,
      alert = true
    }
  }
}
ports = [
  {
    email_notification = true
    port = 443
```

```
      protocol = "TCP"
      alert_if = "NOT_RESPONDING"
    },
    {
      email_notification = false
      port = 80
      protocol = "TCP"
      alert_if = "NOT_RESPONDING"
    },
    {
      email_notification = true
      port = 21
      protocol = "TCP"
      alert_if = "NOT_RESPONDING"
    }
  ]

  processes = [
    {
      email_notification = false
      process = "httpdeamon"
      alert_if = "RUNNING"
    },
    {
      process = "iexplorer",
      alert_if = "NOT_RUNNING"
      email_notification = true
    }]
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the VPN.
- `description` - (Optional) Description for the VPN
- `email` - (Optional) Email address to which notifications monitoring system will send
- `agent` (Required) Indicates which monitoring type will be used. True: To use this monitoring type, you must install an agent on the server. False: Monitor a server without installing an agent. Note: If you do not install an agent, you cannot retrieve information such as free hard disk space or ongoing processes.

Monitoring Policy Thresholds (`thresholds`) support the following:

- ‘cpu - (Required) CPU thresholds
  - `warning - (Required)Warning alert *`value - (Required) Warning to be issued when the threshold is reached. from 1 to 100 * ‘alert - (Required) If set true warning will be issued.
    - ‘critical - (Required) Critical alert
      - ‘value - (Required) Warning to be issued when the threshold is reached. from 1 to 100
      - ‘alert - (Required) If set true warning will be issued.
- ‘ram - (Required) RAM threshold
  - ‘warning - (Required) Warning alert
    - ‘value - (Required) Warning to be issued when the threshold is reached. from 1 to 100
    - ‘alert - (Required) If set true warning will be issued.
  - ‘critical - (Required) Critical alert
    - ‘value - (Required) Warning to be issued when the threshold is reached. from 1 to 100
    - ‘alert - (Required) If set true warning will be issued.
- ‘disk - (Required) Hard Disk threshold
  - ‘warning - (Required) Warning alert
    - ‘value - (Required) Warning to be issued when the threshold is reached. from 1 to 100
    - ‘alert - (Required) If set true warning will be issued.
  - ‘critical - (Required) Critical alert
    - ‘value - (Required) Warning to be issued when the threshold is reached. from 1 to 100
    - ‘alert - (Required) If set true warning will be issued.
- ‘transfer - (Required) Data transfer threshold
  - ‘warning - (Required) Warning alert
    - ‘value - (Required) Warning to be issued when the threshold is reached. from 1 to 100
    - ‘alert - (Required) If set true warning will be issued.
  - ‘critical - (Required) Critical alert
    - ‘value - (Required) Warning to be issued when the threshold is reached. from 1 to 100
    - ‘alert - (Required) If set true warning will be issued.
- ‘internal_ping - (Required) Ping threshold
  - ‘warning - (Required) Warning alert
    - ‘value - (Required) Warning to be issued when the threshold is reached. from 1 to 100
    - ‘alert - (Required) If set true warning will be issued.
  - ‘critical - (Required) Critical alert
    - ‘value - (Required) Warning to be issued when the threshold is reached. from 1 to 100
    - ‘alert - (Required) If set true warning will be issued.

Monitoring Policy Ports (`ports`) support the following:

- `email_notification` - (Required) If set true email will be sent.
- `port` - (Required) Port number.
- `protocol` - (Required) The protocol of the port. Allowed values are `TCP`, `UDP`, `TCP/UDP`, `ICMP` and `IPSEC`.
- `alert_if` - (Required) Condition for the alert to be issued.

Monitoring Policy Ports (`processes`) support the following:

- `email_notification` - (Required) If set true email will be sent.
- `process` - (Required) Process name.
- `alert_if` - (Required) Condition for the alert to be issued.

» oneandone_server

Manages a Private Network on 1&1

» Example Usage

```
resource "oneandone_private_network" "pn" {
  name = "pn_test",
  description = "new stuff001"
  datacenter = "GB"
  network_address = "192.168.7.0"
  subnet_mask = "255.255.255.0"
  server_ids = [
    "${oneandone_server.server.id}",
    "${oneandone_server.server02.id}",
  ]
}
```

» Argument Reference

The following arguments are supported:

- `datacenter` - (Optional) Location of desired 1and1 datacenter. Can be `DE`, `GB`, `US` or `ES`.
- `description` - (Optional) Description for the shared storage
- `name` - (Required) The name of the private network
- `network_address` - (Optional) Network address for the private network
- `subnet_mask` - (Optional) Subnet mask for the private network
- `server_ids` (Optional) List of servers that are to be associated with the private network

» oneandone_ip

Manages a Public IP on 1&1

» Example Usage

```
resource "oneandone_public_ip" "ip" {
    "ip_type"     = "IPV4"
    "reverse_dns" = "%s"
    "datacenter"  = "GB"
}
```

» Argument Reference

The following arguments are supported:

- `ip_type` - (Required) IP type. Can be `IPV4` or `IPV6`
- `reverse_dns` - (Optional)
- `datacenter` - (Optional) Location of desired 1and1 datacenter. Can be `DE`, `GB`, `US` or `ES`.
- `ip_address` - (Computed) The IP address.

» oneandone_server

Manages a Server on 1&1

» Example Usage

```
resource "oneandone_server" "server" {
  name = "Example"
  description = "Terraform 1and1 tutorial"
  image = "ubuntu"
  datacenter = "GB"
  vcores = 1
  cores_per_processor = 1
  ram = 2
  ssh_key_path = "/path/to/private/ssh_key"
  ssh_key_public = "${file("/path/to/public/key.pub")}"
  hdds = [
    {
      disk_size = 60
      is_main = true
```

```
    }
  ]

  provisioner "remote-exec" {
    inline = [
      "apt-get update",
      "apt-get -y install nginx",
    ]
  }
}
```

» Argument Reference

The following arguments are supported:

- `cores_per_processor` -(Optional) Number of cores per processor
- `datacenter` - (Optional) Location of desired 1and1 datacenter. Can be `DE`, `GB`, `US` or `ES`
- `description` - (Optional) Description of the server
- `firewall_policy_id` - (Optional) ID of firewall policy
- `fixed_instance_size` - (Optional) ID of a fixed instance size
- `hdds` - (Optional) List of HDDs. One HDD must be main.
- `*disk_size` -(Required) The size of HDD
- `*is_main` - (Optional) Indicates if HDD is to be used as main hard disk of the server
- `image` -(Required) The name of a desired image to be provisioned with the server
- `ip` - (Optional) IP address for the server
- `loadbalancer_id` - (Optional) ID of the load balancer
- `monitoring_policy_id` - (Optional) ID of monitoring policy
- `name` -(Required) The name of the server.
- `password` - (Optional) Desired password.
- `ram` -(Optional) Size of ram.
- `ssh_key_path` - (Optional) Path to private ssh key
- `ssh_key_public` - (Optional) The public key data in OpenSSH authorized_keys format.
- `vcores` -(Optional) Number of virtual cores.

Either `fixed_instance_size` or all of `vcores`, `cores_per_processor`, `ram` and `hdds` are required.

IPs (`ips`) expose the following attributes

- `id` - (Computed) The ID of the attached IP
- `ip` - (Computed) The IP
- `firewall_policy_id` - (Computed) The attached firewall policy

» oneandone_server

Manages a Shared Storage on 1&1

» Example Usage

```
resource "oneandone_shared_storage" "storage" {
  name = "test_storage1"
  description = "1234"
  size = 50

  storage_servers = [
    {
      id = "${oneandone_server.server.id}"
      rights = "RW"
    },
    {
      id = "${oneandone_server.server02.id}"
      rights = "RW"
    }
  ]
}
```

» Argument Reference

The following arguments are supported:

- `name` - (Required) The name of the storage
- `datacenter` - (Optional) Location of desired 1and1 datacenter. Can be `DE`, `GB`, `US` or `ES`
- `description` - (Optional) Description for the shared storage
- `size` - (Required) Size of the shared storage
- `storage_servers` (Optional) List of servers that will have access to the stored storage
  - `id` - (Required) ID of the server
  - `rights` - (Required) Access rights to be assigned to the server. Can be `RW` or `R`

» oneandone_vpn

Manages a VPN on 1&1

» Example Usage

```
resource "oneandone_vpn" "vpn" {
  datacenter  = "GB"
  name        = "%s"
  description = "ttest descr"
}
```

» Argument Reference

The following arguments are supported:

- `datacenter` - (Optional) Location of desired 1and1 datacenter. Can be `DE`, `GB`, `US` or `ES`.
- `name` - (Required) The name of the VPN
- `description` - (Optional)
- `download_path` - (Optional)
- `file_name` - (Optional)