FortiOS Provider

The FortiOS provider is used to interact with the resources supported by FortiOS. We need to configure the provider with with the proper credentials before it can be used.

Example Usage

```
# Configure the FortiOS Provider
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

# Create a Static Route Item
resource "fortios_networking_route_static" "test1" {
    dst = "110.2.2.122/32"
    gateway = "2.2.2.2"
    # ...
}
```

Authentication

The FortiOS provider offers a means of providing credentials for authentication. The following methods are supported:

- Static credentials
- Environment variables

Static credentials

Static credentials can be provided by adding a `token` key in-line in the FortiOS provider block.

Usage: `hcl provider "fortios" { hostname = "54.226.179.231" token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb" }`

Environment variables

You can provide your credentials via the `FORTIOS_ACCESS_HOSTNAME` and `FORTIOS_ACCESS_TOKEN` environment variables. Note that setting your FortiOS credentials using static credentials variables will override the environment variables.

Usage:

```
$ export FORTIOS_ACCESS_HOSTNAME=192.168.52.177
$ export FORTIOS_ACCESS_TOKEN=q3Hs49jxts195gkd9Hjsxnjtmr6k39
```

Then configure the FortiOS Provider as following:

```
provider "fortios" { }

# Create a Static Route Item
resource "fortios_networking_route_static" "test1" {
    dst = "110.2.2.122/32"
    gateway = "2.2.2.2"
    blackhole = "disable"
    distance = "22"
    weight = "3"
    # …
}
```

VDOM

If the FortiGate unit is running in VDOM mode, the `vdom` configuration needs to be added.

Usage:

```
provider "fortios" {
    hostname = "192.168.52.177"
    token = "q3Hs49jxts195gkd9Hjsxnjtmr6k39"
    vdom = "vdomtest"
}

resource "fortios_networking_route_static" "test1" {
    dst = "120.2.2.122/32"
    gateway = "2.2.2.2"
    blackhole = "disable"
    distance = "22"
    weight = "3"
    priority = "3"
    device = "lbforvdomtest"
    comment = "Terraform test"
}
```

Argument Reference

The following arguments are supported:

- `hostname` - (Optional) This is the hostname or IP address of FortiOS unit. It must be provided, but it can also be sourced from the `FORTIOS_ACCESS_HOSTNAME` environment variable.
- `token` - (Optional) This is the token of FortiOS unit. It must be provided, but it can also be sourced from the `FORTIOS_ACCESS_TOKEN` environment variable.
- `insecure` - (Optional) This is used to control whether the Provider to perform insecure SSL requests. If omitted, the `FORTIOS_INSECURE` environment variable is used. If neither is set, default value is `false` .
- `cabundlefile` - (Optional) The path of a custom CA bundle file. You can specify a path to the file, or you can specify it by the `FORTIOS_CA_CABUNDLE` environment variable.

- `vdom` - (Optional) If the FortiGate unit is running in VDOM mode, you can use this argument to specify the name of the vdom to be set .

Versioning

The provider can cover both FortiOS 6.0 and 6.2 versions.

fortios_firewall_object_addressgroup

Provides a resource to configure firewall address group used in firewall policies of FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_addressgroup" "s1" {
    name = "s1"
    member = ["google-play","swscan.apple.com"]
    comment = "dfdsad"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) Address group name.
- `member` - (Required) Address objects contained within the group.
- `comment` - Comment.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the firewall address group item.
- `name` - Address group name.
- `member` - Address objects contained within the group.
- `comment` - Comment.

fortios_firewall_object_address

Provides a resource to configure firewall addresses used in firewall policies of FortiOS.

Example Usage for Iprange Address

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_address" "s1" {
    name = "s1"
    type = "iprange"
    start_ip = "1.0.0.0"
    end_ip = "2.0.0.0"
    comment = "dd"
}
```

Example Usage for Geography Address

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_address" "s2" {
    name = "s2"
    type = "geography"
    country = "AO"
    comment = "dd"
}
```

Example Usage for Fqdn Address

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_address" "s3" {
    name = "s3"
    type = "fqdn"
    fqdn = "baid.com"
    comment = "dd"
}
```

Example Usage for Ipmask Address

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_address" "s4" {
    name = "s4"
    type = "ipmask"
    subnet = "0.0.0.0 0.0.0.0"
    comment = "dd"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) Address name.
- `type` - (Required) Type of address(Support ipmask, iprange, fqdn and geography).
- `subnet` - IP address and subnet mask of address.
- `start_ip` - First IP address (inclusive) in the range for the address.
- `end_ip` - Final IP address (inclusive) in the range for the address.
- `fqdn` - Fully Qualified Domain Name address.
- `country` - IP addresses associated to a specific country.
- `comment` - Comment.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the address item.
- `name` - Address name.
- `type` - Type of address(Support ipmask, iprange, fqdn and geography).
- `subnet` - IP address and subnet mask of address.
- `start_ip` - First IP address (inclusive) in the range for the address.
- `end_ip` - Final IP address (inclusive) in the range for the address.
- `fqdn` - Fully Qualified Domain Name address.
- `country` - IP addresses associated to a specific country.
- `comment` - Comment.

fortios_firewall_object_ippool

Provides a resource to configure IPv4 IP address pools of FortiOS.

Example Usage for Overload Ippool

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_ippool" "s1" {
    name = "ddd"
    type = "overload"
    startip = "11.0.0.0"
    endip = "22.0.0.0"
    arp_reply = "enable"
    comments = "fdsaf"
}
```

Example Usage for One-to-one Ippool

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_ippool" "s2" {
    name = "dd22d"
    type = "one-to-one"
    startip = "121.0.0.0"
    endip = "222.0.0.0"
    arp_reply = "enable"
    comments = "fdsaf"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) IP pool name.
- `type` - (Required) IP pool type(Support overload and one-to-one).
- `startip` - (Required) First IPv4 address (inclusive) in the range for the address pool (format xxx.xxx.xxx.xxx).
- `endip` - (Required) Final IPv4 address (inclusive) in the range for the address pool (format xxx.xxx.xxx.xxx).

- `arp_reply` - Enable/disable replying to ARP requests when an IP Pool is added to a policy.
- `comments` - Comment.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the IP pool item.
- `name` - IP pool name.
- `type` - IP pool type(Support overload and one-to-one).
- `startip` - First IPv4 address (inclusive) in the range for the address pool (format xxx.xxx.xxx.xxx).
- `endip` - Final IPv4 address (inclusive) in the range for the address pool (format xxx.xxx.xxx.xxx).
- `arp_reply` - Enable/disable replying to ARP requests when an IP Pool is added to a policy.
- `comments` - Comment.

fortios_firewall_object_servicegroup

Provides a resource to configure firewall service group of FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_servicegroup" "v11" {
    name = "1fdsafd11a"
    comment = "fdsafdsa"
    member = ["DCE-RPC", "DNS", "HTTPS"]
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) Service group name.
- `member` - (Required) Service objects contained within the group.
- `comment` - Comment.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the firewall service group item.
- `name` - Service group name.
- `member` - Service objects contained within the group.
- `comment` - Comment.

fortios_firewall_object_service

Provides a resource to configure firewall service of FortiOS.

Example Usage for Fqdn Service

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_service" "v11" {
    name = "servicetest1"
    category = "General"
    protocol = "TCP/UDP/SCTP"
    fqdn = "abc.com"
    comment = "comment"
}
```

Example Usage for Iprange Service

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_service" "v13" {
    name = "servicetest2"
    category = "General"
    protocol = "TCP/UDP/SCTP"
    iprange = "1.1.1.1-2.2.2.2"
    tcp_portrange = "22-33"
    udp_portrange = "44-55"
    sctp_portrange = "66-88"
    comment = "comment"
}
```

Example Usage for ICMP Service

```
resource "fortios_firewall_object_service" "ICMP" {
    name = "ICMPService"
    category = "General"
    protocol = "ICMP"
    icmptype = "2"
    icmpcode = "3"
    protocol_number = "1"
    comment = "comment"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) Number of minutes before an idle administrator session time out.
- `category` - (Required) Service category.
- `protocol` - Protocol type based on IANA numbers.
- `fqdn` - Fully qualified domain name.
- `iprange` - Start and end of the IP range associated with service.
- `tcp_portrange` - Multiple TCP port ranges.
- `udp_portrange` - Multiple UDP port ranges.
- `sctp_portrange` - Multiple SCTP port ranges.
- `icmptype` - ICMP type.
- `icmpcode` - ICMP code.
- `protocol_number` - IP protocol number.
- `comment` - Comment.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the firewall service item.
- `name` - Number of minutes before an idle administrator session time out.
- `category` - Service category.
- `protocol` - Protocol type based on IANA numbers.
- `fqdn` - Fully qualified domain name.
- `iprange` - Start and end of the IP range associated with service.

- `tcp_portrange` - Multiple TCP port ranges.
- `udp_portrange` - Multiple UDP port ranges.
- `sctp_portrange` - Multiple SCTP port ranges.
- `icmptype` - ICMP type.
- `icmpcode` - ICMP code.
- `protocol_number` - IP protocol number.
- `comment` - Comment.

fortios_firewall_object_vipgroup

Provides a resource to configure virtual IP groups of FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_vipgroup" "v11" {
    name = "1fdsafd11a"
    interface = "port3"
    comments = "comments"
    member = ["vip1", "vip3"]
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) VIP group name.
- `interface` - Interface name.
- `member` - (Required) Member VIP objects of the group.
- `comments` - Comment.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the virtual IP groups item.
- `name` - VIP group name.
- `interface` - Interface name.
- `member` - Member VIP objects of the group.
- `comments` - Comment.

fortios_firewall_object_vip

Provides a resource to configure firewall virtual IPs (VIPs) of FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_object_vip" "v11" {
    name = "dfa"
    comment = "fdsafdsafds"
    extip = "11.1.1.1-21.1.1.1"
    mappedip = ["22.2.2.2-32.2.2.2"]
    extintf = "port3"
    portforward = "enable"
    protocol = "tcp"
    extport = "2-3"
    mappedport = "4-5"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) Virtual IP name.
- `extip` - (Required) IP address or address range on the external interface that you want to map to an address or address range on the destination network.
- `mappedip` - (Required) IP address or address range on the destination network to which the external IP address is mapped.
- `extintf` - Interface connected to the source network that receives the packets that will be forwarded to the destination network.
- `portforward` - Enable/disable port forwarding.
- `protocol` - Protocol to use when forwarding packets.
- `extport` - Incoming port number range that you want to map to a port number range on the destination network.
- `mappedport` - Port number range on the destination network to which the external port number range is mapped.
- `comment` - Comment.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the firewall virtual IPs item.
- `name` - Virtual IP name.
- `extip` - IP address or address range on the external interface that you want to map to an address or address range on the destination network.
- `mappedip` - IP address or address range on the destination network to which the external IP address is mapped.
- `extintf` - Interface connected to the source network that receives the packets that will be forwarded to the destination network.
- `portforward` - Enable/disable port forwarding.
- `protocol` - Protocol to use when forwarding packets.
- `extport` - Incoming port number range that you want to map to a port number range on the destination network.
- `mappedport` - Port number range on the destination network to which the external port number range is mapped.
- `comment` - Comment.

fortios_firewall_security_policy

Provides a resource to configure firewall policies of FortiOS.

Example Usage 1

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_security_policy" "test1" {
    name = "ap11"
    srcintf = ["port2"]
    dstintf = ["port1"]
    srcaddr = ["swscan.apple.com", "google-play"]
    dstaddr = ["swscan.apple.com", "update.microsoft.com"]
    internet_service = "disable"
    internet_service_id = []
    schedule = "always"
    service = ["ALL_ICMP", "FTP"]
    action = "accept"
    utm_status = "enable"
    logtraffic = "all"
    logtraffic_start = "enable"
    capture_packet = "enable"
    ippool = "enable"
    poolname = ["rewq", "rbb"]
    groups = ["Guest-group", "SSO_Guest_Users"]
    devices = ["android-phone", "android-tablet"]
    comments = "security policy"
    av_profile = "wifi-default"
    webfilter_profile = "monitor-all"
    dnsfilter_profile = "default"
    ips_sensor = "protect_client"
    application_list = "block-high-risk"
    ssl_ssh_profile = "certificate-inspection"
    nat = "enable"
}
```

Example Usage 2

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_firewall_security_policy" "test2" {
    name = "ap21"
    srcintf = ["port2"]
    dstintf = ["port1"]
    srcaddr = ["swscan.apple.com", "google-play"]
    dstaddr = ["swscan.apple.com", "update.microsoft.com"]
    internet_service = "enable"
    internet_service_id = [917520, 6881402, 393219]
    schedule = "always"
    service = []
    action = "accept"
    utm_status = "enable"
    logtraffic = "all"
    logtraffic_start = "enable"
    capture_packet = "enable"
    ippool = "enable"
    poolname = ["rewq", "rbb"]
    groups = ["Guest-group", "SSO_Guest_Users"]
    devices = ["android-phone", "android-tablet"]
    comments = "security policy"
    av_profile = "wifi-default"
    webfilter_profile = "monitor-all"
    dnsfilter_profile = "default"
    ips_sensor = "protect_client"
    application_list = "block-high-risk"
    ssl_ssh_profile = "certificate-inspection"
    nat = "enable"
}
```

Example Usage 3

```
resource "fortios_firewall_security_policy" "test1" {
    name = "ap12221"
    srcintf = ["port3"]
    dstintf = ["port4"]
    srcaddr = []
    dstaddr = []
    internet_service = "enable"
    internet_service_id = [5242880]
    internet_service_src = "enable"
    internet_service_src_id = [65643]
    users = ["guest"]
    status = "enable"
    schedule = "always"
    service = []
    action = "accept"
    utm_status = "enable"
    logtraffic = "all"
    logtraffic_start = "enable"
    capture_packet = "enable"
    ippool = "disable"
    poolname = []
    groups = ["Guest-group", "SSO_Guest_Users"]
    devices = []
    comments = "security policy"
    av_profile = "wifi-default"
    webfilter_profile = "monitor-all"
    dnsfilter_profile = "default"
    ips_sensor = "protect_client"
    application_list = "block-high-risk"
    ssl_ssh_profile = "certificate-inspection"
    nat = "enable"
    profile_protocol_options = "default"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) Policy name.
- `srcintf` - (Required) Incoming (ingress) interface.
- `dstintf` - (Required) Outgoing (egress) interface.
- `srcaddr` - (Required) Source address and address group names.
- `dstaddr` - (Required) Destination address and address group names.
- `internet_service` - Enable/disable use of Internet Services for this policy. If enabled, destination address and service are not used.
- `internet_service_id` - Internet Service ID.
- `action` - (Required) Policy action.

- `schedule` - (Required) Schedule name.
- `service` - (Required) Service and service group names..
- `utm_status` - Enable to add one or more security profiles (AV, IPS, etc.) to the firewall policy.
- `logtraffic` - Enable or disable logging. Log all sessions or security profile sessions.
- `logtraffic_start` - Record logs when a session starts and ends.
- `capture_packet` - Enable/disable capture packets.
- `ippool` - Enable to use IP Pools for source NAT.
- `poolname` - IP Pool names.
- `groups` - Names of user groups that can authenticate with this policy.
- `devices` - Device type category.
- `comments` - Comment.
- `av_profile` - Name of an existing Antivirus profile.
- `webfilter_profile` - Name of an existing Web filter profile.
- `dnsfilter_profile` - Name of an existing DNS filter profile.
- `ips_sensor` - Name of an existing IPS sensor.
- `application_list` - Name of an existing Application list.
- `ssl_ssh_profile` - Name of an existing SSL SSH profile.
- `nat` - Enable/disable source NAT.
- `internet_service_src` - Enable/disable use of Internet Services in source for this policy. If enabled, source address is not used.
- `internet_service_src_id` - Internet Service source ID.
- `users` - Names of individual users that can authenticate with this policy.
- `status` - Enable or disable this policy.
- `profile_protocol_options` - Name of an existing Protocol options profile.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the firewall policy item.
- `name` - Policy name.
- `srcintf` - Incoming (ingress) interface.
- `dstintf` - Outgoing (egress) interface.

- `srcaddr` - Source address and address group names.
- `dstaddr` - Destination address and address group names.
- `internet_service` - Enable/disable use of Internet Services for this policy. If enabled, destination address and service are not used.
- `internet_service_id` - Internet Service ID.
- `action` - Policy action.
- `schedule` - Schedule name.
- `service` - Service and service group names..
- `utm_status` - Enable to add one or more security profiles (AV, IPS, etc.) to the firewall policy.
- `logtraffic` - Enable or disable logging. Log all sessions or security profile sessions.
- `logtraffic_start` - Record logs when a session starts and ends.
- `capture_packet` - Enable/disable capture packets.
- `ippool` - Enable to use IP Pools for source NAT.
- `poolname` - IP Pool names.
- `groups` - Names of user groups that can authenticate with this policy.
- `devices` - Device type category.
- `comments` - Comment.
- `av_profile` - Name of an existing Antivirus profile.
- `webfilter_profile` - Name of an existing Web filter profile.
- `dnsfilter_profile` - Name of an existing DNS filter profile.
- `ips_sensor` - Name of an existing IPS sensor.
- `application_list` - Name of an existing Application list.
- `ssl_ssh_profile` - Name of an existing SSL SSH profile.
- `nat` - Enable/disable source NAT.
- `internet_service_src` - Enable/disable use of Internet Services in source for this policy. If enabled, source address is not used.
- `internet_service_src_id` - Internet Service source ID.
- `users` - Names of individual users that can authenticate with this policy.
- `status` - Enable or disable this policy.
- `profile_protocol_options` - Name of an existing Protocol options profile.

fortios_log_fortianalyzer_setting

Provides a resource to configure configure logging to FortiAnalyzer log management devices.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_log_fortianalyzer_setting" "test1" {
    status = "enable"
    server = "10.2.2.99"
    source_ip = "10.2.2.99"
    upload_option = "realtime"
    reliable = "enable"
    hmac_algorithm = "sha256"
    enc_algorithm = "high-medium"
}
```

Argument Reference

The following arguments are supported:

- `status` - (Required) Enable/disable logging to FortiAnalyzer.
- `server` - The remote FortiAnalyzer.
- `source_ip` - Source IPv4 or IPv6 address used to communicate with FortiAnalyzer.
- `upload_option` - Enable/disable logging to hard disk and then uploading to FortiAnalyzer.
- `reliable` - Enable/disable reliable logging to FortiAnalyzer.
- `hmac_algorithm` - FortiAnalyzer IPsec tunnel HMAC algorithm.
- `enc_algorithm` - Enable/disable sending FortiAnalyzer log data with SSL encryption.

Attributes Reference

The following attributes are exported:

- `status` - Enable/disable logging to FortiAnalyzer.
- `server` - The remote FortiAnalyzer.
- `source_ip` - Source IPv4 or IPv6 address used to communicate with FortiAnalyzer.
- `upload_option` - Enable/disable logging to hard disk and then uploading to FortiAnalyzer.

- `reliable` - Enable/disable reliable logging to FortiAnalyzer.
- `hmac_algorithm` - FortiAnalyzer IPsec tunnel HMAC algorithm.
- `enc_algorithm` - Enable/disable sending FortiAnalyzer log data with SSL encryption.

fortios_log_syslog_setting

Provides a resource to configure logging to remote Syslog logging servers.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_log_syslog_setting" "test2" {
    status = "enable"
    server = "2.2.2.2"
    mode = "udp"
    port = "514"
    facility = "local7"
    source_ip = "10.2.2.199"
    format = "csv"
}
```

Argument Reference

The following arguments are supported:

- `status` - (Required) Enable/disable remote syslog logging.
- `server` - Address of remote syslog server.
- `mode` - Remote syslog logging over UDP/Reliable TCP.
- `port` - Server listen port.
- `facility` - Remote syslog facility.
- `source_ip` - Source IP address of syslog.
- `format` - Log format.

Attributes Reference

The following attributes are exported:

- `status` - Enable/disable remote syslog logging.
- `server` - Address of remote syslog server.
- `mode` - Remote syslog logging over UDP/Reliable TCP.
- `port` - Server listen port.

- `facility` - Remote syslog facility.
- `source_ip` - Source IP address of syslog.
- `format` - Log format.

fortios_networking_interface_port

Provides a resource to configure interface settings of FortiOS.

Example Usage for Loopback Interface

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_networking_interface_port" "loopback1" {
    ip = "23.123.33.10/24"
    allowaccess = "ping http"
    alias = "cc1"
    description = "description"
    status = "up"
    role = "lan"
    name = "myinterface1"
    vdom = "root"
    type = "loopback"
    mode = "static"
}
```

Example Usage for VLAN Interface

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_networking_interface_port" "vlan1" {
    role = "lan"
    mode = "static"
    defaultgw = "enable"
    distance = "33"
    type = "vlan"
    vlanid = "3"
    name = "myinterface2"
    vdom = "root"
    ip = "3.123.33.10/24"
    interface = "port2"
    allowaccess = "ping"
}
```

Example Usage for Physical Interface

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_networking_interface_port" "test1" {
    name = "port2"
    ip = "93.133.133.110/24"
    alias = "dkeeew"
    description = "description"
    status = "up"
    device_identification = "enable"
    tcp_mss = "3232"
    speed = "auto"
    mtu_override = "enable"
    mtu = "2933"
    role = "lan"
    allowaccess = "ping https"
    mode = "static"
    dns_server_override = "enable"
    defaultgw = "enable"
    distance = "33"
    type = "physical"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) If the interface is physical, the argument is the name of the interface.
- `type` - (Required) Interface type (support physical, vlan, loopback).
- `ip` - Interface IPv4 address and subnet mask, syntax` - X.X.X.X/24.
- `alias` - Alias will be displayed with the interface name to make it easier to distinguish.
- `status` - Bring the interface up or shut the interface down.
- `device_identification` - Enable/disable passively gathering of device identity information about the devices on the network connected to this interface.
- `tcp_mss` - TCP maximum segment size. 0 means do not change segment size.
- `speed` - Interface speed. The default setting and the options available depend on the interface hardware.
- `mtu_override` - Enable to set a custom MTU for this interface.
- `mtu` - MTU value for this interface.
- `role` - Interface role.
- `allowaccess` - Permitted types of management access to this interface.
- `mode` - (Required) Addressing mode.

- `dns_server_override` - Enable/disable use DNS acquired by DHCP or PPPoE.
- `defaultgw` - Enable to get the gateway IP from the DHCP or PPPoE server.
- `distance` - Distance for routes learned through PPPoE or DHCP, lower distance indicates preferred route.
- `description` - Description.
- `interface` - Interface name.
- `name` - Name.
- `vdom` - Interface is in this virtual domain (VDOM).
- `vlanid` - VLAN ID.

Attributes Reference

The following attributes are exported:

- `id` - The Name of the interface.
- `ip` - Interface IPv4 address and subnet mask, syntax` - X.X.X.X/24.
- `alias` - Alias will be displayed with the interface name to make it easier to distinguish.
- `status` - Bring the interface up or shut the interface down.
- `device_identification` - Enable/disable passively gathering of device identity information about the devices on the network connected to this interface.
- `tcp_mss` - TCP maximum segment size. 0 means do not change segment size.
- `speed` - Interface speed. The default setting and the options available depend on the interface hardware.
- `mtu_override` - Enable to set a custom MTU for this interface.
- `mtu` - MTU value for this interface.
- `role` - Interface role.
- `allowaccess` - Permitted types of management access to this interface.
- `mode` - Addressing mode.
- `dns_server_override` - Enable/disable use DNS acquired by DHCP or PPPoE.
- `defaultgw` - Enable to get the gateway IP from the DHCP or PPPoE server.
- `distance` - Distance for routes learned through PPPoE or DHCP, lower distance indicates preferred route.
- `description` - Description.
- `type` - Interface type (support physical, vlan, loopback).
- `interface` - Interface name.
- `name` - Name.

- `vdom` - Interface is in this virtual domain (VDOM).
- `vlanid` - VLAN ID.

fortios_networking_route_static

Provides a resource to configure static route of FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_networking_route_static" "test1" {
    dst = "110.2.2.122/32"
    gateway = "2.2.2.2"
    blackhole = "disable"
    distance = "22"
    weight = "3"
    priority = "3"
    device = "port2"
    comment = "Terraform test"
}
```

Argument Reference

The following arguments are supported:

- `dst` - (Required) Destination IP and mask for this route.
- `gateway` - (Required) Gateway IP for this route.
- `blackhole` - Enable/disable black hole.
- `distance` - Administrative distance.
- `weight` - Administrative weight.
- `priority` - Administrative priority.
- `device` - (Required) Gateway out interface or tunnel.
- `comment` - Optional comments.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the static route item.
- `dst` - Destination IP and mask for this route.

- `gateway` - Gateway IP for this route.
- `blackhole` - Enable/disable black hole.
- `distance` - Administrative distance.
- `weight` - Administrative weight.
- `priority` - Administrative priority.
- `device` - Gateway out interface or tunnel.
- `comment` - Optional comments.

fortios_system_admin_administrator

Provides a resource to configure administrator accounts of FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_system_admin_administrator" "admintest" {
    name = "testadminacc"
    password = "cc37331AC1"
    trusthost1 = "1.1.1.0 255.255.255.0"
    trusthost2 = "2.2.2.0 255.255.255.0"
    accprofile = "3d3"
    vdom = ["root"]
    comments = "comments"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) User name.
- `password` - (Required) Admin user password.
- `trusthostN` - Any IPv4 address or subnet address and netmask from which the administrator can connect to the FortiGate unit.
- `vdom` - Virtual domain(s) that the administrator can access.
- `accprofile` - Access profile for this administrator. Access profiles control administrator access to FortiGate features.
- `comments` - Comment.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the administrator account item.
- `name` - User name.
- `password` - Admin user password.
- `trusthostN` - Any IPv4 address or subnet address and netmask from which the administrator can connect to the

FortiGate unit.

- `vdom` - Virtual domain(s) that the administrator can access.
- `accprofile` - Access profile for this administrator. Access profiles control administrator access to FortiGate features.
- `comments` - Comment.

fortios_system_admin_profiles

Provides a resource to configure access profiles of FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_system_admin_profiles" "test1" {
    name = "223d3"
    scope = "vdom"
    comments = "test"
    secfabgrp = "read-write"
    ftviewgrp = "read"
    authgrp = "none"
    sysgrp = "read"
    netgrp = "none"
    loggrp = "none"
    fwgrp = "none"
    vpngrp = "none"
    utmgrp = "none"
    wanoptgrp = "none"
    wifi = "none"
    admintimeout_override = "disable"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) Profile name.
- `scope` - Scope of admin access.
- `secfabgrp` - Security Fabric.
- `ftviewgrp` - FortiView.
- `authgrp` - Administrator access to Users and Devices.
- `sysgrp` - System Configuration.
- `netgrp` - Network Configuration.
- `loggrp` - Administrator access to Logging and Reporting including viewing log messages.
- `fwgrp` - Administrator access to the Firewall configuration.
- `vpngrp` - Administrator access to IPsec, SSL, PPTP, and L2TP VPN.

- `utmgrp` - Administrator access to Security Profiles.
- `wanoptgrp` - Administrator access to WAN Opt & Cache.
- `wifi` - Administrator access to the WiFi controller and Switch controller.
- `admintimeout_override` - Enable/disable overriding the global administrator idle timeout.
- `comments` - Comment.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the access profile item.
- `name` - Profile name.
- `scope` - Scope of admin access.
- `secfabgrp` - Security Fabric.
- `ftviewgrp` - FortiView.
- `authgrp` - Administrator access to Users and Devices.
- `sysgrp` - System Configuration.
- `netgrp` - Network Configuration.
- `loggrp` - Administrator access to Logging and Reporting including viewing log messages.
- `fwgrp` - Administrator access to the Firewall configuration.
- `vpngrp` - Administrator access to IPsec, SSL, PPTP, and L2TP VPN.
- `utmgrp` - Administrator access to Security Profiles.
- `wanoptgrp` - Administrator access to WAN Opt & Cache.
- `wifi` - Administrator access to the WiFi controller and Switch controller.
- `admintimeout_override` - Enable/disable overriding the global administrator idle timeout.
- `comments` - Comment.

fortios_system_apiuser_setting

Provides a resource to configure API users of FortiOS. The API user of the token for this feature should have a super admin profile, It can be set in CLI while GUI does not allow.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_system_apiuser_setting" "test2" {
    name = "testapiuser"
    accprofile = "restAPIprofile"
    vdom = ["root"]
    trusthost {
        type = "ipv4-trusthost"
        ipv4_trusthost = "61.149.0.0 255.255.0.0"
    }

    trusthost {
        type = "ipv4-trusthost"
        ipv4_trusthost = "22.22.0.0 255.255.0.0"
    }
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) User name.
- `accprofile` - (Required) Admin user access profile.
- `vdom` - (Required) Virtual domains.
- `trusthost-Type` - (Required) Trusthost type.
- `trusthost-ipv4_trusthost` - (Required) IPv4 trusted host address.
- `comments` - Comment.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the API user.

- `name` - User name.
- `accprofile` - Admin user access profile.
- `vdom` - Virtual domains.
- `trusthost-Type` - Trusthost type.
- `trusthost-ipv4_trusthost` - IPv4 trusted host address.
- `comments` - Comment.

fortios_system_license_forticare

Provides a resource to add a FortiCare license for FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_system_license_forticare" "test2" {
    registration_code = "license"
}
```

Argument Reference

The following arguments are supported:

- `registration_code` - (Required) Registration code.

fortios_system_license_vdom

Provides a resource to add a VDOM license for FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_system_license_vdom" "test2" {
    license = "license"
}
```

Argument Reference

The following arguments are supported:

- `license` - (Required) Registration code.

fortios_system_license_vm

Provides a resource to update VM license using uploaded file for FortiOS. Reboots immediately if successful.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_system_license_vm" "test2" {
    file_content = "LS0tLS1CRUdJTiBGR1QgVk0gTElDRU5TRS0tLS0tDQpRQUFBQUxXaTdCVnVkV2x3QXJZcC92S2J2Yk5zME5YN
WluUW9sVldmcFoxWldJQi9pL2g4c01oR0psWWc5Vkl1DQorSlBJRis1aFphMWwyNm9yNHdiEQE3RnJDeVZnQUFBQWhxWjliWHFLK1hGN2
o3dnB3WTB6QXRTaTdOMVM1ZWNxDQpWYmRRREZyYklUdnRvUWNyRU1jV0ltQzFqWWs5dmVoeGlYTG1OV0MwN25BSitYTTJFNmh2b29DMjE
1YUwxK2wrDQovUHl5M0VLVnNTNjJDT2hMZHc3UndXajB3V3RqMmZiWg0KLS0tLS1FTkQgRkdUIFZNIExJQ0VOU0UtLS0tLQ0K"
}
```

Argument Reference

The following arguments are supported:

- `file_content` - (Required) The license file, it needs to be base64 encoded, must not contain whitespace or other invalid base64 characters, and must be included in HTTP body.

fortios_system_setting_dns

Provides a resource to configure DNS of FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_system_setting_dns" "test1" {
    primary = "208.91.112.53"
    secondary = "208.91.112.22"
}
```

Argument Reference

The following arguments are supported:

- `primary` - Primary DNS server IP address.
- `secondary` - Secondary DNS server IP address.

Attributes Reference

The following attributes are exported:

- `primary` - Primary DNS server IP address.
- `secondary` - Secondary DNS server IP address.

fortios_system_setting_global

Provides a resource to configure options related to the overall operation of FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_system_setting_global" "test1" {
    admintimeout = 65
    timezone = "04"
    hostname = "mytestFortiGate"
    admin_sport = 443
    admin_ssh_port = 22
}
```

Argument Reference

The following arguments are supported:

- `hostname` - (Required) FortiGate unit's hostname.
- `admintimeout` - Number of minutes before an idle administrator session time out.
- `timezone` - Number corresponding to your time zone from 00 to 86.
- `admin_sport` - Administrative access port for HTTPS.
- `admin_ssh_port` - Administrative access port for SSH.

Attributes Reference

The following attributes are exported:

- `admintimeout` - Number of minutes before an idle administrator session time out.
- `timezone` - Number corresponding to your time zone from 00 to 86.
- `hostname` - FortiGate unit's hostname.
- `admin_sport` - Administrative access port for HTTPS.
- `admin_ssh_port` - Administrative access port for SSH.

fortios_system_setting_ntp

Provides a resource to configure Network Time Protocol (NTP) servers of FortiOS.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_system_setting_ntp" "test2" {
    type = "custom"
    ntpserver = ["1.1.1.1", "3.3.3.3"]
    ntpsync = "disable"
}
```

Argument Reference

The following arguments are supported:

- `type` - (Required) Use the FortiGuard NTP server or any other available NTP Server.
- `ntpserver` - Configure the FortiGate to connect to any available third-party NTP server.
- `ntpsync` - Enable/disable setting the FortiGate system time by synchronizing with an NTP Server.

Attributes Reference

The following attributes are exported:

- `type` - Use the FortiGuard NTP server or any other available NTP Server.
- `ntpserver` - Configure the FortiGate to connect to any available third-party NTP server.
- `ntpsync` - Enable/disable setting the FortiGate system time by synchronizing with an NTP Server.

fortios_system_vdom_setting

Provides a resource to configure VDOM of FortiOS. The API user of the token for this feature should have a super admin profile, It can be set in CLI while GUI does not allow.

Example Usage

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_system_vdom_setting" "test2" {
    name = "aa1122"
    short_name = "aa1122"
    temporary = 0
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) VDOM name.
- `short_name` - VDOM short name.
- `temporary` - Temporary.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the VDOM.
- `name` - VDOM name.
- `short_name` - VDOM short name.
- `temporary` - Temporary.

fortios_vpn_ipsec_phase1interface

Provides a resource to use phase1-interface to define a phase 1 definition for a route-based (interface mode) IPsec VPN tunnel that generates authentication and encryption keys automatically.

Example Usage

fortios_vpn_ipsec_phase1interface needs to be set with fortios_vpn_ipsec_phase2interface. See section fortios_vpn_ipsec_phase2interface.

Argument Reference

The following arguments are supported:

- `name` - (Required) IPsec remote gateway name.
- `type` - (Required) Remote gateway type.
- `interface` - (Required) Local physical, aggregate, or VLAN outgoing interface.
- `peertype` - Accept this peer type.
- `proposal` - Phase1 proposal.
- `comments` - Comment.
- `wizard_type` - GUI VPN Wizard Type.
- `remote_gw` - (Required) IPv4 address of the remote gateway's external interface.
- `psksecret` - (Required) Pre-shared secret for PSK authentication.
- `certificate` - Names of signed personal certificates.
- `peerid` - Accept this peer identity.
- `peer` - Accept this peer certificate.
- `peergrp` - Accept this peer certificate group.
- `ipv4_split_include` - IPv4 split-include subnets.
- `split_include_service` - Split-include services.
- `ipv4_split_exclude` - IPv4 subnets that should not be sent over the IPsec tunnel.
- `authmethod` - Authentication method.
- `authmethod_remote` - Authentication method (remote side).
- `mode_cfg` - Enable/disable configuration method.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the phase1-interface item.
- `name` - IPsec remote gateway name.
- `type` - Remote gateway type.
- `interface` - Local physical, aggregate, or VLAN outgoing interface.
- `peertype` - Accept this peer type.
- `proposal` - Phase1 proposal.
- `comments` - Comment.
- `wizard_type` - GUI VPN Wizard Type.
- `remote_gw` - IPv4 address of the remote gateway's external interface.
- `psksecret` - Pre-shared secret for PSK authentication.
- `certificate` - Names of signed personal certificates.
- `peerid` - Accept this peer identity.
- `peer` - Accept this peer certificate.
- `peergrp` - Accept this peer certificate group.
- `ipv4_split_include` - IPv4 split-include subnets.
- `split_include_service` - Split-include services.
- `ipv4_split_exclude` - IPv4 subnets that should not be sent over the IPsec tunnel.
- `authmethod` - Authentication method.
- `authmethod_remote` - Authentication method (remote side).
- `mode_cfg` - Enable/disable configuration method.

fortios_vpn_ipsec_phase2interface

Provides a resource to use phase2-interface to add or edit a phase 2 configuration on a route-based (interface mode) IPsec tunnel.

Example Usage for Site to Site/Pre-shared Key

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_vpn_ipsec_phase1interface" "test1" {
    name = "001Test"
    type = "static"
    interface = "port2"
    peertype = "any"
    proposal = "aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1"
    comments = "VPN 001Test P1"
    wizard_type = "static-fortigate"
    remote_gw = "1.2.2.2"
    psksecret = "testscecret112233445566778899"
    authmethod = "psk"
    authmethod_remote = ""
    mode_cfg = "disable"

}
```

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_vpn_ipsec_phase2interface" "test2" {
    name = "001Test"
    phase1name = "001Test"
    proposal = "aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm chacha20poly1305"
    comments = "VPN 001Test P2"
    src_addr_type = "name"
    dst_addr_type = "name"
    src_name = "HQ-toBranch_local"
    dst_name = "HQ-toBranch_remote"
}
```

Example Usage for Site to Site/Signature

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_vpn_ipsec_phase1interface" "test1" {
    name = "001Test"
    type = "static"
    interface = "port2"
    proposal = "aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1"
    comments = "VPN 001Test P1"
    wizard_type = "static-fortigate"
    remote_gw = "1.2.2.2"
    psksecret = "testscecret112233445566778899"
    certificate = ["Fortinet_SSL_ECDSA384"]
    peertype = "peer"
    peerid = ""
    peer = "2b_peer"
    peergrp = ""
}
```

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_vpn_ipsec_phase2interface" "test2" {
    name = "001Test"
    phase1name = "001Test"
    proposal = "aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm chacha20poly1305"
    comments = "VPN 001Test P2"
    src_addr_type = "range"
    dst_addr_type = "subnet"
    src_start_ip = "1.1.1.0"
    src_end_ip = "1.1.1.1"
    dst_subnet = "2.2.2.2/24"
}
```

Example Usage for Remote Access/Pre-shared Key

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_vpn_ipsec_phase1interface" "test1" {
    name = "001Test"
    type = "dynamic"
    interface = "port2"
    peertype = "any"
    proposal = "aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1"
    comments = "VPN 001Test P1"
    wizard_type = "dialup-forticlient"
    remote_gw = "0.0.0.0"
    psksecret = "testscecret1122334455667788 99"
    ipv4_split_include = "d_split"
    split_include_service = ""
    ipv4_split_exclude = ""
}
```

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_vpn_ipsec_phase2interface" "test2" {
    name = "001Test"
    phase1name = "001Test"
    proposal = "aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm chacha20poly1305"
    comments = "VPN 001Test P2"
    src_addr_type = "subnet"
    src_start_ip = "0.0.0.0"
    src_end_ip = "0.0.0.0"
    src_subnet = "0.0.0.0 0.0.0.0"
    dst_addr_type = "subnet"
    dst_start_ip = "0.0.0.0"
    dst_end_ip = "0.0.0.0"
    dst_subnet = "0.0.0.0 0.0.0.0"
}
```

Example Usage for Remote Access/Signature

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_vpn_ipsec_phase1interface" "test1" {
    name = "001Test"
    type = "dynamic"
    interface = "port2"
    peertype = "any"
    proposal = "aes128-sha256 aes256-sha256 aes128-sha1 aes256-sha1"
    comments = "VPN 001Test P1"
    wizard_type = "dialup-forticlient"
    remote_gw = "1.2.2.2"
    psksecret = "testscecret1122334455667788 99"
    certificate = ["Fortinet_SSL_ECDSA384"]
    peertype = "peer"
    peerid = ""
    peer = "2b_peer"
    peergrp = "",
    ipv4_split_include = "d_split"
    split_include_service = ""
    ipv4_split_exclude = ""
}
```

```
provider "fortios" {
    hostname = "54.226.179.231"
    token = "jn3t3Nw7qckQzt955Htkfj5hwQ6jdb"
}

resource "fortios_vpn_ipsec_phase2interface" "test2" {
    name = "001Test"
    phase1name = "001Test"
    proposal = "aes128-sha1 aes256-sha1 aes128-sha256 aes256-sha256 aes128gcm aes256gcm chacha20poly1305"
    comments = "VPN 001Test P2"
    src_addr_type = "subnet"
    src_start_ip = "0.0.0.0"
    src_end_ip = "0.0.0.0"
    src_subnet = "0.0.0.0 0.0.0.0"
    dst_addr_type = "subnet"
    dst_start_ip = "0.0.0.0"
    dst_end_ip = "0.0.0.0"
    dst_subnet = "0.0.0.0 0.0.0.0"
}
```

Argument Reference

The following arguments are supported:

- `name` - (Required) IPsec tunnel name.
- `phase1name` - (Required) Phase 1 determines the options required for phase 2.
- `proposal` - Phase2 proposal.

- `src_addr_type` - Local proxy ID type.
- `src_start_ip` - Local proxy ID start.
- `src_end_ip` - Local proxy ID end.
- `src_subnet` - Local proxy ID subnet.
- `dst_addr_type` - Local proxy ID type.
- `src_name` - Local proxy ID name.
- `dst_name` - Remote proxy ID name.
- `dst_start_ip` - Remote proxy ID IPv4 start.
- `dst_end_ip` - Remote proxy ID IPv4 end.
- `dst_subnet` - Remote proxy ID IPv4 subnet.
- `comments` - Comment.

Attributes Reference

The following attributes are exported:

- `id` - The ID of the phase2-interface.
- `name` - IPsec tunnel name.
- `phase1name` - Phase 1 determines the options required for phase 2.
- `proposal` - Phase2 proposal.
- `src_addr_type` - Local proxy ID type.
- `src_start_ip` - Local proxy ID start.
- `src_end_ip` - Local proxy ID end.
- `src_subnet` - Local proxy ID subnet.
- `dst_addr_type` - Local proxy ID type.
- `src_name` - Local proxy ID name.
- `dst_name` - Remote proxy ID name.
- `dst_start_ip` - Remote proxy ID IPv4 start.
- `dst_end_ip` - Remote proxy ID IPv4 end.
- `dst_subnet` - Remote proxy ID IPv4 subnet.
- `comments` - Comment.