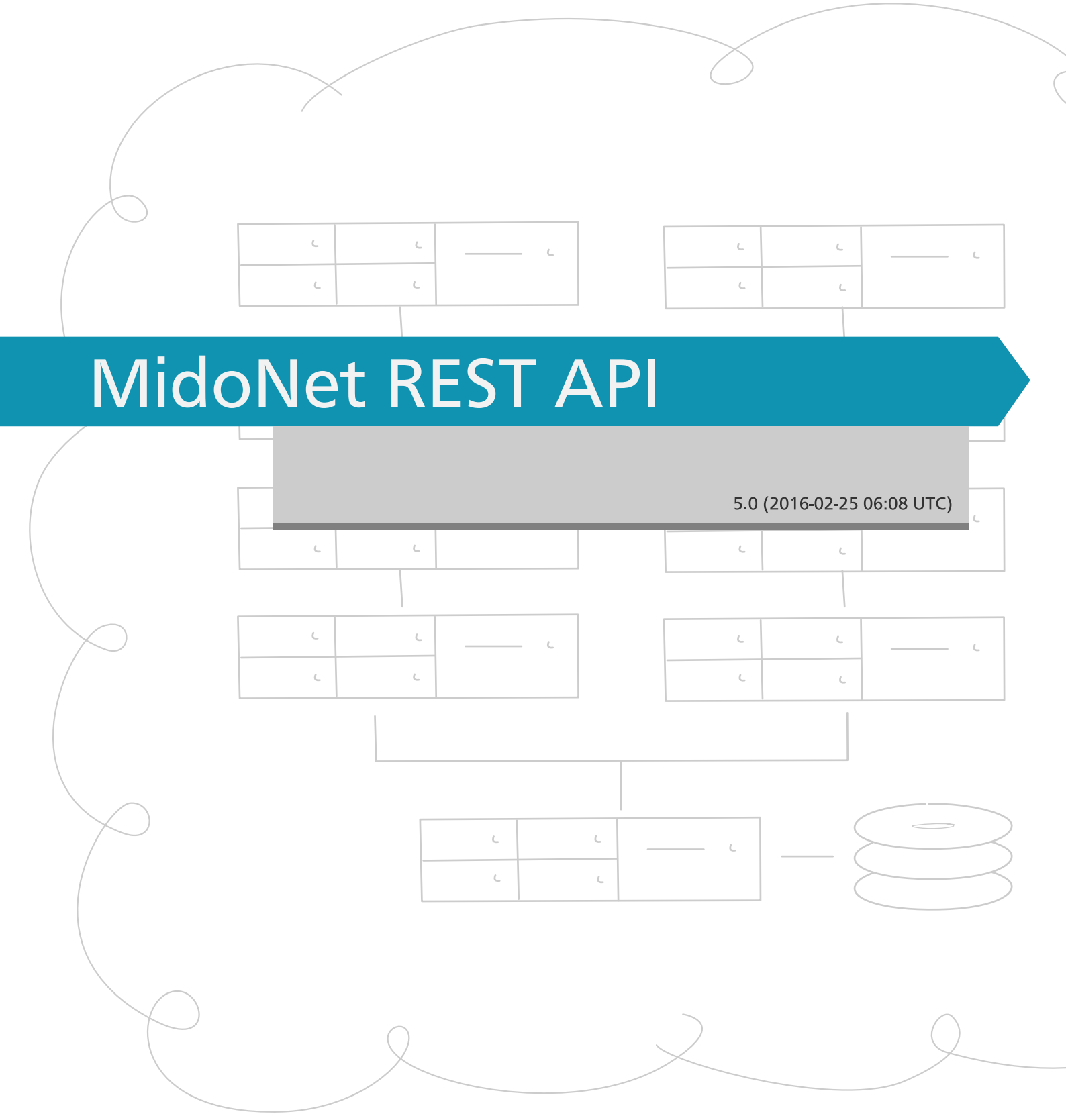

MidoNet REST API

5.0 (2016-02-25 06:08 UTC)

midonet

docs.midonet.org

MidoNet REST API

5.0 (2016-02-25 06:08 UTC)

MidoNet is a network virtualization software for Infrastructure-as-a-Service (IaaS) clouds.

It decouples your IaaS cloud from your network hardware, creating an intelligent software abstraction layer between your end hosts and your physical network.

This document describes the MidoNet REST API.

Note

Please consult the MidoNet Mailing Lists or Chat if you need assistance.

Table of Contents

1. Introduction

This document specifies a RESTful API for creating and managing MidoNet resources. The API uses JSON as its format.

2. Getting Started

This section is intended to help users get started on using the API. It assumes that the MidoNet Management REST API host is known. This host is represented as `example.org` in this document. The following GET request to the base URL of the API reveals the locations of the available resources :

```
GET /
Host: example.org
Accept: application/vnd.org.midonet.Application-v5+json
```

The request above may yield the following output :

```
HTTP/1.1 200 OK
Content-Type: application/vnd.org.midonet.Application-v5+json
{
  "uri":"http://example.com/midonet-api/",
  "version":"v5.0",
  "bgpNetworkTemplate":"http://example.com/midonet-api/bgp_networks/{id}",
  "bgpPeerTemplate":"http://example.com/midonet-api/bgp_peers/{id}",
  "bridges":"http://example.com/midonet-api/bridges",
  "bridgeTemplate":"http://example.com/midonet-api/bridges/{id}",
  "chains":"http://example.com/midonet-api/chains",
  "chainTemplate":"http://example.com/midonet-api/chains/{id}",
  "healthMonitors":"http://example.com/midonet-api/health_monitors",
  "healthMonitorTemplate":"http://example.com/midonet-api/health_monitors/
{id}",
  "hosts":"http://example.com/midonet-api/hosts",
  "hostTemplate":"http://example.com/midonet-api/hosts/{id}",
  "ipAddrGroups":"http://example.com/midonet-api/ip_addr_groups",
  "ipAddrGroupTemplate":"http://example.com/midonet-api/ip_addr_groups/
{id}",
  "l2insertions":"http://example.com/midonet-api/l2insertions",
  "l2InsertionTemplate":"http://example.com/midonet-api/l2insertions/{id}",
  "loadBalancers":"http://example.com/midonet-api/load_balancers",
  "loadBalancerTemplate":"http://example.com/midonet-api/load_balancers/
{id}",
  "mirrors":"http://example.com/midonet-api/mirrors",
  "mirrorTemplate":"http://example.com/midonet-api/mirrors/{id}",
  "neutron":"http://example.com/midonet-api/neutron",
  "poolMembers":"http://example.com/midonet-api/pool_members",
  "poolMemberTemplate":"http://example.com/midonet-api/pool_members/{id}",
  "pools":"http://example.com/midonet-api/pools",
  "poolTemplate":"http://example.com/midonet-api/pools/{id}",
  "portGroups":"http://example.com/midonet-api/port_groups",
  "portGroupTemplate":"http://example.com/midonet-api/port_groups/{id}",
  "ports":"http://example.com/midonet-api/ports",
  "portTemplate":"http://example.com/midonet-api/ports/{id}",
  "routers":"http://example.com/midonet-api/routers",
  "routerTemplate":"http://example.com/midonet-api/routers/{id}",
  "routeTemplate":"http://example.com/midonet-api/routes/{id}",
  "ruleTemplate":"http://example.com/midonet-api/rules/{id}",
  "systemState":"http://example.com/midonet-api/system_state",
  "tenants":"http://example.com/midonet-api/tenants",
  "tenantTemplate":"http://example.com/midonet-api/tenants/{id}",
  "traceRequests":"http://example.com/midonet-api/traces",
  "traceRequestTemplate":"http://example.com/midonet-api/traces/{id}",
  "tunnelZones":"http://example.com/midonet-api/tunnel_zones",
  "tunnelZoneTemplate":"http://example.com/midonet-api/tunnel_zones/{id}",
  "vips":"http://example.com/midonet-api/vips",
  "vipTemplate":"http://example.com/midonet-api/vips/{id}",
  "vteps":"http://example.com/midonet-api/vteps",
  "vtepTemplate":"http://example.com/midonet-api/vteps/{id}"
```

```
}
```

This reveals that users can access the router resources using the URI `/routers`. Host resources are accessible with the URI `/hosts`. The response also includes information about the API version. The URIs with `{id}` in them are *uri-templates*, and they are explained later in this document.

3. Common Behaviors

Table of Contents

This section specifies the common constraints that apply to all the requests and responses that occur in the MidoNet Management REST API.

Media Types

In MidoNet REST API, the resources are encoded in JSON, as specified in RFC 4267. Each type of resource has its own media-type, which matches the pattern:

application/vnd.org.midonet.xxxxx-v#+json

where *xxxxx* represents the unique resource identifier and # is the media type's version number. For most media types the version number will be 1, but several media types have additional versions. See the sections on individual media types for available versions. Starting with MidoNet v5.0, the REST API discontinued support for old media type versions, and in general you must use the newest available version.

When doing a *GET* on a particular resource, specify the media type in the `Accept` header field. When doing a *POST* or *PUT* on a particular resource, specify the media type in the `Content-Type` header field. This also applies when you are operating on collections as well.

Request Headers

The following HTTP request headers are relevant to MidoNet REST API:

Header	Supported Values	Description	Required
Accept	Comma-delimited list of media types or media type patterns	Indicates to the server what media type(s) this client is prepared to accept	No, but recommended
Content Type	Media type describing the request message body	Describes the representation and syntax of the request message body	Yes

Response Headers

The following HTTP response headers exist in MidoNet REST API:

Header	Supported Values	Description	Required
Content Type	Media type describing the response message body	Describes the representation and syntax of the response message body	Yes

Header	Supported Values	Description	Required
Location	Canonical URI of a newly created resource	A new URI that can be used to request a representation of the newly created resource	Yes, on responsed that create new server side resources which are accessible via a URI

HTTP Status Codes

The following HTTP status codes are returned from MidoNet REST API:

HTTP Status	Description
200 OK	The request was successfully completed, and the response body contains the resource data.
201 Created	A new resource was successfully created. A Location header contains the URI of the resource.
204 No Content	The server fulfilled the request, but does not need to return anything.
400 Bad Request	The request could not be processed because it contained missing or invalid information.
401 Unauthorized	The authentication credentials included with the request are missing or invalid.
403 Forbidden	The server recognized the credentials, but the user is not authorized to perform this request.
404 Not Found	The requested URI does not exist.
405 Method Not Allowed	The HTTP verb specified in the request (GET, POST, PUT, DELETE, HEAD) is not supported for this URI.
406 Not Acceptable	The resource identified by this request is not capable of generating a representation corresponding to one of the media types in the Accept header.
409 Conflict	A creation or update request could not be completed because it would cause a conflict in the current state of the resources. One example is when a request attempts to create a resource with an identifier that already exists.
500 Internal Server Error	The server encountered an unexpected condition which prevented the request to be completed.
503 Service Unavailable	The server is currently unable to handle the request due to temporary overloading or maintenance of the server.

URI Templates

A URI may contain a part that is left out to the client to fill. These parts are enclosed inside { and }.

For example, given a URI template, `http://example.org/routers/{id}` and a router identifier `d7435bb0-3bc8-11e2-81c1-0800200c9a66`, after doing the replacement, the final URI becomes: `http://example.org/routers/d7435bb0-3bc8-11e2-81c1-0800200c9a66`.

The following table lists the existing expressions in the URI templates and what they should be replaced with:

Expression	Replace with
id	Unique identifier of resource
ipAddr	IP address
macAddress	MAC address
portId	Port UUID
portName	Port name
vlanId	VLAN identifier

Methods

POST

Used to create a new resource. The `Location` header field in the response contains the URI of the newly created resource.

PUT

Used to update an existing resource.

GET

Used to retrieve one more more resources. It could either return a single object or a collection of objects in the response.

DELETE

In MidoNet API, the *DELETE* operation means cascade delete unless noted otherwise. When a resource is deleted, all of its child resources are also deleted.

4. Resource Models

Table of Contents

This section specifies the representations of the MidoNet REST API resources. Each type of resource has its own Internet Media Type. The media type for each resource is included in square brackets in the corresponding section header.

The 'POST/PUT' column indicates whether the field can be included in the request with these verbs. If they are not specified, the field should not be included in the request.

The Required column indicates is only relevant for POST/PUT operations. You should not see any entry for 'Required' if the 'POST/PUT' column is empty. When the Required value is set, it will have indicate whether the field is relevant for POST, PUT or both. Required fields need to be included in the request to create/update the object. Note that fields may be required for PUT but not POST, and viceversa. In this case it will be indicated in the specific cell for the field.

Application

Media Type `[application/vnd.org.midonet.Application-v5+json]`

```
GET     /
```

This is the root object in MidoNet REST API. From this object, clients can traverse the URIs to discover all the available services. The Neutron endpoint was added in version 5.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
version	String			The version of MidoNet REST API.
bridges	URI			A GET against this URI returns the list of bridges.
chains	URI			A GET against this URI returns the list of chains.
healthMonitors	URI			A GET against this URI returns the list of health monitors.
hosts	URI			A GET against this URI returns the list of hosts.
ipAddrGroups	URI			A GET against this URI returns the list of IP address groups.
loadBalancers	URI			A GET against this URI returns the list of load balancers.
mirrors	URI			A GET against this URI returns the list of port mirrors.
neutron	URI			A GET against this URI returns the available Neutron resources.
poolMembers	URI			A GET against this URI returns the list of pool members.
pools	URI			A GET against this URI returns the list of pools.
portGroups	URI			A GET against this URI returns the list of port groups.
ports	URI			A GET against this URI returns the list of ports.
routers	URI			A GET against this URI returns the list of routers.
systemState	URI			A GET against this URI returns the system state.

Field Name	Type	POST/PUT	Required	Description
tenants	URI			A GET against this URI returns the list of tenants.
traceRequests	URI			A GET against this URI returns the list of trace requests.
tunnelZones	URI			A GET against this URI return the list of tunnel zones.
vips	URI			A GET against this URI returns the list of VIPs.
vteps	URI			A GET against this URI returns the list of VTEPs.
bgpNetworkTemplate	String			Template of the URI that represents the location of the BGP network with the provided identifier.
bgpPeerTemplate	String			Template of the URI that represents the location of the BGP peer with the provided identifier.
bridgeTemplate	String			Template of the URI that represents the location of the bridge with the provided identifier.
chainTemplate	String			Template of the URI that represents the location of the chain with the provided identifier.
healthMonitorTemplate	String			Template of the URI that represents the location of the health monitor with the provided identifier.
hostTemplate	String			Template of the URI that represents the location of the host with the provided identifier.
ipAddrGroupTemplate	String			Template of the URI that represents the location of the IP address group with the provided identifier.
loadBalancerTemplate	String			Template of the URI that represents the location of the load balancer with the provided identifier.
mirrorTemplate	String			Template of the URI that represents the location of the port mirror with the provided identifier.
poolMemberTemplate	String			Template of the URI that represents the location of the pool member with the provided identifier.
poolTemplate	String			Template of the URI that represents the location of the pool with the provided identifier.
portGroupTemplate	String			Template of the URI that represents the location of the port group with the provided identifier.
portTemplate	String			Template of the URI that represents the location of the port with the provided identifier.
routerTemplate	String			Template of the URI that represents the location of the router with the provided identifier.
routeTemplate	String			Template of the URI that represents the location of the route with the provided identifier.
ruleTemplate	String			Template of the URI that represents the location of the rule with the provided identifier.

Field Name	Type	POST/PUT	Required	Description
tenantTemplate	String			Template of the URI that represents the location of the tenant with the provided identifier.
traceRequestTemplate	String			Template of the URI that represents the location of the trace request with the provided identifier.
tunnelZoneTemplate	String			Template of the URI that represents the location of the tunnel zone with the provided identifier.
vipTemplate	String			Template of the URI that represents the location of the VIP with the provided identifier.
vtepTemplate	String			Template of the URI that represents the location of the VTEP with the provided identifier.

The following media types have been removed from the API: `[application/vnd.org.midonet.Application-v1+json]` `[application/vnd.org.midonet.Application-v2+json]` `[application/vnd.org.midonet.Application-v3+json]` `[application/vnd.org.midonet.Application-v4+json]`

BGP Network

Media Type `[application/vnd.org.midonet.BgpNetwork-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.BgpNetwork-v1+json]`

```
GET     /routers/:routerId/bgp_networks
GET     /bgp_networks/:bgpNetworkId
POST    /routers/:routerId/bgp_networks
DELETE  /bgp_networks/:bgpNetworksId
```

BGP Network is an entity that represents an IP network adevertised to a BGP peer. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
router	URI			A GET against this URI returns the router resource to which the BGP network belongs.
id	UUID	POST	No	A unique identifier of the resource. If the field is omitted in the POST request a random UUID is generated.
subnetAddress	String	POST	Yes	The IPv4 subnet prefix address.
subnetLength	Integer	POST	Yes	The IPv4 subnet prefix length. The value must belong to the interval [0, 32].

BGP Peer

Media Type `[application/vnd.org.midonet.BgpPeer-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.BgpPeer-v1+json]`

```
GET     /routers/:routerId/bgp_peers
GET     /bgp_peers/:bgpPeerId
POST    /routers/:routerId/bgp_peers
PUT     /bgp_peers/:bgpPeerId
DELETE  /bgp_peers/:bgpPeerId
```

BGP Peer is an entity that represents a the BGP endpoint of a neighboring autonomous system. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
router	URI			A GET against this URI returns the router resource to which the BGP network belongs.
id	UUID	POST	No	A unique identifier of the resource. If the field is omitted in the POST request a random UUID is generated.
address	String	POST/PUT	Yes	The IPv4 address of the BGP peer to which the local router will connect.
asNumber	Integer	POST/PUT	Yes	The autonomous system number (ASN) assigned to the BGP neighbor. The value must be greater than zero.
connectRetry	Integer	POST/PUT	No	The connection retry timer in seconds for the BGP session. The value must belong to the interval [5, 3600], and if present it will override the value specified in the MidoNet Agent configuration.
holdTime	Integer	POST/PUT	No	The hold timer in seconds for the BGP session. The value must belong to the interval [5, 7200], and if present it will override the value specified in the MidoNet Agent configuration.
keepAlive	Integer	POST/PUT	No	The keep alive timer in seconds for the BGP session. The value must belong to the interval [5, 3600], and if present it will override the value specified in the MidoNet Agent configuration.

Bridge

Media Type `[application/vnd.org.midonet.Bridge-v4+json]`

Collection Media Type `[application/vnd.org.midonet.collection.Bridge-v4+json]`

```
GET     /bridges
GET     /bridges?tenant_id=:tenantId
GET     /bridges/:bridgeId
POST    /bridges
PUT     /bridges/:bridgeId
DELETE  /bridges/:bridgeId
```

Bridge is an entity that represents a virtual bridge device in MidoNet. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
arpTable	URI			A GET against this URI returns the ARP table for this bridge. See the section called "IPv4-MAC Pair" [18].

Field Name	Type	POST/PUT	Required	Description
dhcpSubnets	URI			A GET against this URI returns the DHCP subnets for this bridge. See the section called "DHCP Subnet" [14].
dhcpSubnets6	URI			A GET against this URI returns the DHCPv6 subnets for this bridge. See the section called "DHCPv6 Subnet" [15].
inboundFilter	URI			A GET against this URI returns the inbound filter chain.
macTable	URI			A GET against this URI returns the MAC table for this bridge. See the section called "MAC-Port" [21].
outboundFilter	URI			A GET against this URI returns the outbound filter chain.
ports	URI			A GET against this URI returns the ports for this bridge. See the section called "Port" [35].
peerPorts	URI			A GET against this URI returns the interior ports connected to this bridge.
vxLanPorts	URI			A GET against this URI returns the VXLAN ports for this bridge.
macPortTemplate	String			Template of the URI that represents the location of the MAC-port entry in the MAC table for this bridge.
vlanMacPortTemplate	String			Template of the URI that represents the location of the MAC-port entry for a specific VLAN in the MAC table of this bridge.
vlanMacTableTemplate	String			Template of the URI that represents the location of the MAC table for a specific VLAN of this bridge. See the section called "MAC-Port" [21].
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
adminStateUp	Boolean	POST/PUT	No	The administrative state of the bridge, and if *false* (down), the bridge stops forwarding packets. The default is *true* (up).
inboundFilterId	UUID	POST/PUT	No	The identifier of the filter chain to be applied for ingress packets.
inboundMirrorIds	Array of UUID	POST/PUT	No	The list of IDs for the mirrors applied to ingress packets.
name	String	POST/PUT	No	The name of the bridge. The maximum length is 255 characters.
outboundFilterId	UUID	POST/PUT	No	The identifier of the filter chain to be applied for egress packets.
outboundMirrorIds	Array of UUID	POST/PUT	No	The list of IDs for the mirrors applied to egress packets.
tenantId	String	POST/PUT	No	The identifier of the tenant that owns the bridge.
vxlanPortIds	Array of UUID			The list of IDs for the bridge VXLAN ports, which contains the bindings from this bridge to hardware VTEPs. It is read-only.

Query Parameters

Name	Description
tenant_id	The identifier of the tenant to filter the search.

Chain

Media Type `[application/vnd.org.midonet.Chain-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.Chain-v1+json]`

```
GET     /chains
GET     /chains?tenant_id=:tenantId
GET     /chains/:chainId
POST    /chains
DELETE  /chains/:chainId
```

Chain is an entity that represents a rule chain on a virtual router in MidoNet. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
rules	URI			A GET against this URI returns the rules belonging to this chain.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
name	String	POST	No	The name of the chain. The maximum length is 255 characters.
tenantId	String	POST	No	The identifier of the tenant that owns the chain.

Query Parameters

Name	Description
tenant_id	The identifier of the tenant to filter the search.

DHCP Host

Media Type `[application/vnd.org.midonet.DhcpHost-v2+json]`

Collection Media Type `[application/vnd.org.midonet.collection.DhcpHost-v2+json]`

```
GET     /bridges/:bridgeId/dhcp/:subnetAddr/hosts
GET     /bridges/:bridgeId/dhcp/:subnetAddr/hosts/:mac_address
POST    /bridges/:bridgeId/dhcp/:subnetAddr/hosts
PUT     /bridges/:bridgeId/dhcp/:subnetAddr/hosts/:mac_address
DELETE  /bridges/:bridgeId/dhcp/:subnetAddr/hosts/:mac_address
```

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this source.
ipAddr	String	POST/PUT	Yes	The IPv4 address of the host.
macAddr	String	POST/PUT	Yes	The MAC Address of the host.
name	String	POST/PUT	Yes	The name of the host.
extraDhcpOpts	Array of (String, String)	POST/PUT	No	List of DHCP options where an option is composed of two key-value pairs with the key fields, `optName` and `optValue`. For "optName", use the DHCP option code listed here: http://www.iana.org/assignments/bootp-dhcp-parameters/bootp-dhcp-parameters.xhtml#options For example, to set the interface MTU: `[{"optName": "26", "optValue": "9000"}]`

DHCP Subnet

Media Type `[application/vnd.org.midonet.DhcpSubnet-v2+json]`

Collection Media Type `[application/vnd.org.midonet.collection.DhcpSubnet-v2+json]`

```
GET     /bridges/:bridgeId/dhcp
GET     /bridges/:bridgeId/dhcp/:subnetAddr
POST    /bridges/:bridgeId/dhcp
PUT     /bridges/:bridgeId/dhcp/:subnetAddr
DELETE  /bridges/:bridgeId/dhcp/:subnetAddr
```

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this source.
hosts	URI			A GET against this URI returns the DHCP hosts for this subnet.
subnetPrefix	String	POST/PUT	Yes	The IPv4 subnet prefix address.
subnetLength	Integer	POST/PUT	Yes	The IPv4 subnet prefix length. The value must belong to the interval [0, 32].
defaultGateway	String	POST/PUT	No	The IPv4 address of the default gateway.
dnsServerAddrs	Array of String	POST/PUT	No	The list of DNS server IPv4 addresses.
enabled	Boolean	POST/PUT	No	Indicates whether the DHCP service is enabled. The default value is true.
interfaceMTU	Integer	POST/PUT	No	The interface Maximum Transmission Unit. The value must belong to the interval [0, 65536].
opt121Routes	Array of (String, Integer, String)	POST/PUT	No	The list of DHCP option 121 routes, each of which consists of the following fields: `destinationPrefix` as an IPv4 subnet address; `destinationLength` as an IPv4 subnet prefix length; and `gatewayAddr` as the gateway IPv4 address.
serverAddr	String	POST/PUT	No	The IPv4 address of the DHCP server.

DHCPv6 Host

Media Type `[application/vnd.org.midonet.DhcpV6Host-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.DhcpV6Host-v1+json]`

```
GET     /bridges/:bridgeId/dhcpV6/:subnetAddr/hosts
GET     /bridges/:bridgeId/dhcpV6/:subnetAddr/hosts/:mac_address
POST    /bridges/:bridgeId/dhcpV6/:subnetAddr/hosts
PUT     /bridges/:bridgeId/dhcpV6/:subnetAddr/hosts/:mac_address
DELETE  /bridges/:bridgeId/dhcpV6/:subnetAddr/hosts/:mac_address
```

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this source.
clientId	String	POST/PUT	Yes	The client identifier.

Field Name	Type	POST/PUT	Required	Description
fixedAddress	String	POST/PUT	Yes	The IPv6 address assigned to the specified client.
name	String	POST/PUT	Yes	The name of the host.

DHCPv6 Subnet

Media Type `[application/vnd.org.midonet.DhcpV6Subnet-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.DhcpV6Subnet-v1+json]`

```
GET     /bridges/:bridgeId/dhcpV6
GET     /bridges/:bridgeId/dhcpV6/:subnetAddr
POST    /bridges/:bridgeId/dhcpV6
PUT     /bridges/:bridgeId/dhcpV6/:subnetAddr
DELETE  /bridges/:bridgeId/dhcpV6/:subnetAddr
```

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this source.
hosts	URI			A GET against this URI returns the DHCP hosts for this subnet.
prefix	String	POST/PUT	Yes	The IPv6 subnet prefix.
prefixLength	Integer	POST/PUT	Yes	The IPv6 subnet prefix length. The value must belong to the interval [0, 128].

Health Monitor

Media Type `[application/vnd.org.midonet.HealthMonitor-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.HealthMonitor-v1+json]`

```
GET     /health_monitors
GET     /health_monitors/:healthMonitorId
POST    /health_monitors
PUT     /health_monitors/:healthMonitorId
DELETE  /health_monitors/:healthMonitorId
```

Note

To use this feature, please make sure that health monitoring is activated in the MidoNet Agent configuration. See *HAProxy configuration* in the Operation Guide for details.

A HealthMonitor is an entity that represents a virtual health monitor device for use with load balancers in MidoNet. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.

Field Name	Type	POST/PUT	Required	Description
pools	URI			A GET against this URI returns the pools monitored by this healh monitor.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
type	String	POST/PUT	Yes	The type of the health monitor checking protocol. The following type is supported: TCP.
adminStateUp	Boolean	POST/PUT	No	The administrative state of the health monitor. The default is true (up).
delay	Integer	POST/PUT	No	The delay for the health check interval in seconds. The default is zero.
maxRetries	Integer	POST/PUT	No	The number of times to retry for health check. The defaults is zero.
timeout	Integer	POST/PUT	No	The timeout value for the health check in seconds. The defaults is zero.

Host

Media Type `[application/vnd.org.midonet.Host-v3+json]`

Collection Media Type `[application/vnd.org.midonet.collection.Host-v3+json]`

```
GET     /hosts
GET     /hosts/:hostId
PUT     /hosts/:hostId
DELETE  /hosts/:hostId
```

Host is an entity that provides some information about a MidoNet Agent node. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
interfaces	URI			A GET against this URI returns the interfaces of this host.
ports	URI			A GET against this URI returns the virtual ports bound to the interfaces of this host.
id	UUID			A unique identifier of the resource. It is generated by the MidoNet Agent running on the host.
alive	Boolean			Returns true if the MidoNet Agent is running on the physical host.
name	String			The host name.
addresses	Array of String			The list of IP addresses assigned to the interfaces of this host.
hostInterfaces	Array of Interface			The list of interfaces belonging to this host. See the section called "Interface" [17].
floodingProxyWeight	Integer	PUT	No	The weight assigned to the host for becoming a flooding proxy for a L2 VXLAN gateway. For more information see the *L2 VXLAN Gateway* in the Operation Guide. The value must belong to the interval [0, 65535].

Host Interface Port

Media Type `[application/vnd.org.midonet.HostInterfacePort-v1+json]`

```
GET     /hosts/:hostId/ports
GET     /hosts/:hostId/ports/:portId
POST    /hosts/:hostId/ports
DELETE  /hosts/:hostId/ports/:portId
```

The `HostInterfacePort` binding allows mapping a virtual network port to an interface (virtual or physical) of a physical host where the MidoNet Agent is running. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
host	URI			A GET against this URI returns the host corresponding to this binding.
port	URI			A GET against this URI returns the virtual port corresponding to this binding.
hostId	UUID	POST	Yes	The identifier of the physical host.
interfaceName	String	POST	Yes	The name of the interface that is mapped to the virtual port.
portId	UUID	POST	Yes	The identifier of the virtual port mapped to the host interface.

Interface

Media Type `[application/vnd.org.midonet.Interface-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.Interface-v1+json]`

```
GET     /hosts/:hostId/interfaces
GET     /hosts/:hostId/interfaces/:interfaceName
```

The interface is an entity abstracting information about a physical interface associated with a host.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
addresses	Array of String			The list of IP addresses bound to this interface.
addresses	Array of String			The list of IP addresses bound to this interface.
hostId	UUID			The identifier of the host that owns this interface.
mac	String			The interface physical address (MAC).
mtu	Integer			The interface maximum transmission unit (MTU) value.
name	String			The physical interface name.
portType	String			The datapath port type. It can be one of the following: `NetDev`, `Internal`, `Gre`, `VXLan`, `Gre64`, `Lisp`
status	Integer			A bitmask representing the status flags. Currently it provides information about *UP* (0x1) status and *CARRIER* (0x2) status.

Field Name	Type	POST/PUT	Required	Description
type	String			The interface type. It can be one of the following: `Unknown`, `Physical`, `Virtual`, `Tunnel`.

IPv4-MAC Pair

Media Type `[application/vnd.org.midonet.IP4Mac-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.IP4Mac-v1+json]`

```
GET     /bridges/:bridgeId/arp_table
GET     /bridges/:bridgeId/arp_table/:ip4MacPair
POST    /bridges/:bridgeId/arp_table
DELETE  /bridges/:bridgeId/arp_table/:ip4MacPair
```

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
ip	String	POST	Yes	The IP version 4 address.
mac	String	POST	Yes	The MAC address. If ARP replies are enabled on the bridge, the IP will resolve to this MAC.

IP Address Group

Media Type `[application/vnd.org.midonet.IpAddrGroup-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.IpAddrGroup-v1+json]`

```
GET     /ip_addr_groups
GET     /ip_addr_groups/:ipAddrGroupId
POST    /ip_addr_groups
DELETE  /ip_addr_groups/:ipAddrGroupId
```

IP address group is a group of IP addresss. Currently only IPv4 is supported. An IP address group can be specified in the chain rule to filter the traffic coming from all the addresses belonging to that the specified group.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
addrs	URI			A GET against this URI returns the members of this address group.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
name	String	POST	Yes	The name of the address group. The name length must be between 1 and 255 characters.

IP Address Group Address

Media Type `[application/vnd.org.midonet.IpAddrGroupAddr-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.IpAddrGroupAddr-v1+json]`

```
GET     /ip_addr_groups/:ipAddrGroupId/ip_addrs
GET     /ip_addr_groups/:ipAddrGroupId/versions/:version/ip_addrs/:ip_addr
POST    /ip_addr_groups/:ipAddrGroupId/ip_addrs
DELETE  /ip_addr_groups/:ipAddrGroupId/versions/:version/ip_addrs/:ip_addr
```

IP address group address represents the membership of an IP address in an IP address group.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
ipAddrGroup	URI			A GET against this URI returns the IP address group.
addr	String	POST	Yes	The IPv4 or IPv6 address.
ipAddrGroupId	UUID	POST	Yes	The identifier of the IP address group of which this IP address is a member.
version	Integer			The IP address version. The value is 4 or 6.

L2 Insertion

Media Type `[application/vnd.org.midonet.L2Insertion-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.L2Insertion-v1+json]`

```
GET     /l2insertions
GET     /l2insertions/:l2insertionId
POST    /l2insertions
PUT     /l2insertions/:l2insertionId
DELETE  /l2insertions/:l2insertionId
```

L2 insertion represents a service insertion redirecting the packets to a specified service port. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
mac	String	POST/PUT	Yes	The MAC address for which the corresponding traffic will be redirected by this service insertion.
portId	UUID	POST/PUT	Yes	The identifier of the virtual port on which the traffic is inspected.
position	Integer	POST/PUT	Yes	The position of the service insertion in a service insertions chain.
srvPortId	UUID	POST/PUT	Yes	The identifier of the virtual port toward which the traffic will be redirected.
failOpen	Boolean	POST/PUT	No	If true, it allows the traffic when the chain is down or not ready. The default is false.
vlan	Integer	POST/PUT	No	The VLAN identifier applied to the redirected traffic.

Port Link

Media Type `[application/vnd.org.midonet.PortLink-v1+json]`

```
POST     /ports/:portId/link
DELETE   /ports/:portId/link
```

It represents a link between two interior ports. Links are possible between:

- Two router ports.
- A router port and a bridge port
- Two bridge ports, as long as just one of the two peers has a VLAN ID assigned. The bridge owning this port will act as a VLAN-aware bridge, pushing and poping VLAN IDs as frames traverse this port.

It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
port	URI			A GET against this URI returns the port.
peer	URI			A GET against this URI returns the peer port.
portId	UUID	POST	Yes	The identifier of the port.
peerId	UUID	POST	Yes	The identifier of the peer port.

Load Balancer

Media Type `[application/vnd.org.midonet.LoadBalancer-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.LoadBalancer-v1+json]`

```
GET     /load_balancers
GET     /load_balancers/:loadBalancerId
POST    /load_balancers
PUT     /load_balancers/:loadBalancerId
DELETE  /load_balancers/:loadBalancerId
```

A load balancer is an entity that represents a layer 4 virtual load balancer device. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
router	URI			A GET against this URI returns the router for this load balancer.
pools	URI			A GET against this URI returns the list of pools associated with the load balancer.
vips	URI			A GET against this URI returns the list of VIPs associated with the load balancer.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.

Field Name	Type	POST/PUT	Required	Description
routerId	UUID	POST/PUT	Yes	The identifier of the associated router.
adminStateUp	Boolean	POST/PUT	No	The administrative state of the load balancer. The default is *true* (up).

MAC-Port

Media Type `[application/vnd.org.midonet.MacPort-v2+json]`

Collection Media Type `[application/vnd.org.midonet.collection.MacPort-v2+json]`

```
GET     /bridges/:bridgeId/mac_table
GET     /bridges/:bridgeId/vlans/:vlanId/mac_table
GET     /bridges/:bridgeId/mac_table/:macPortPair
GET     /bridges/:bridgeId/vlans/:vlanId/mac_table/:macPortPair
POST    /bridges/:bridgeId/mac_table
POST    /bridges/:bridgeId/vlans/:vlanId/mac_table
DELETE  /bridges/:bridgeId/mac_table/:macPortPair
DELETE  /bridges/:bridgeId/vlans/:vlanId/mac_table/:macPortPair
```

It represents the mapping between a MAC address and a corresponding virtual port identifier.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
macAddr	String	POST	Yes	The physical (MAC) address.
portId	UUID	POST	Yes	The identifier of the virtual port corresponding to the MAC address.
vlanId	Integer			The VLAN to which the port belongs. The field is ignored in POST requests.

Mirror

Media Type `[application/vnd.org.midonet.Mirror-v1+json]`

Collection Media Type `["application/vnd.org.midonet.collection.Mirror-v1+json]`

```
GET     /mirrors
GET     /mirrors/:mirrorId
POST    /mirrors
PUT     /mirrors/:mirrorId
DELETE  /mirrors/:mirrorId
```

A mirror is an entity that indicates whether the traffic flowing through a particular virtual port and matching a set of conditions should be mirrored to another virtual port.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
conditions	Array of Condition	POST/PUT	No	A list of matching conditions against which the mirrored traffic should be matched. See the section called "Condition" [22].

Condition

A mirror condition uses the same fields as a chain rule. See the section called "Rule" [40].

Field Name	Type	POST/PUT	Required	Description
condInvert	Boolean	POST/PUT	No	Inverts the conjunction of all the other predicates.
dlDst	String	POST/PUT	No	Matches the destination physical (MAC) address.
dlSrc	String	POST/PUT	No	Matches the source physical (MAC) address.
dlDstMask	String	POST/PUT	No	Destination physical (MAC) address mask in the format *xxxx.xxxx.xxxx* where each *x* is a hexadecimal digit.
dlSrcMask	String	POST/PUT	No	Source physical (MAC) address mask in the format *xxxx.xxxx.xxxx* where each *x* is a hexadecimal digit.
dlType	Integer	POST/PUT	No	Matches the ethertype provided by the data link layer. The value must be in the interval [0x800, 0xFFFF].
fragmentPolicy	String	POST/PUT	No	Matches the datagram fragmentation. The value can be one of the following: *any* (matches any fragment), *header* (matches the first fragment, *nonheader* (matches subsequent fragments), *unfragmented* (matches unfragmented datagrams).
inPorts	Array of UUID	POST/PUT	No	Matches the list of (interior or exterior) ingress ports.
ipAddrGroupDst	UUID	POST/PUT	No	Matches the destination IP address with an IP address from the specified IP address group.
ipAddrGroupSrc	UUID	POST/PUT	No	Matches the source IP address with an IP address from the specified IP address group.
invDlDst	Boolean	POST/PUT	No	Inverts the destination data link (MAC) address predicate. It has no effect unless the `dlDst` field is also set.
invDlSrc	Boolean	POST/PUT	No	Inverts the source data link (MAC) address predicate. It has no effect unless the `dlSrc` field is also set.
invDlType	Boolean	POST/PUT	No	Inverts the data link ethertype predicate. It has no effect unless the `dlType` field is also set.
invInPorts	Boolean	POST/PUT	No	Inverts the ingress ports predicate.
invIpAddrGroupDst	Boolean	POST/PUT	No	Inverts the destination IP address group predicate.
invIpAddrGroupSrc	Boolean	POST/PUT	No	Inverts the source IP address group predicate.
invNwDst	Boolean	POST/PUT	No	Inverts the network layer destination address predicate. It has no effect unless the `nwDst` field is also set.
invNwProto	Boolean	POST/PUT	No	Inverts the network layer protocol number predicate. It has no effect unless the `nwProto` field is also set.
invNwSrc	Boolean	POST/PUT	No	Inverts the network layer source address predicate. It has no effect unless the `nwSrc` field is also set.
invNwTos	Boolean	POST/PUT	No	Inverts the network layer type-of-service (ToS) predicate. It has no effect unless the `nwTos` field is also set.
invOutPorts	Boolean	POST/PUT	No	Inverts the egress ports predicate.
invPortGroup	Boolean	POST/PUT	No	Inverts the port group predicate.
invTpDst	Boolean	POST/PUT	No	Inverts the destination TCP/UDP port range predicate.
invTpSrc	Boolean	POST/PUT	No	Inverts the source TCP/UDP port range predicate.
invTraversedDevice	Boolean	POST/PUT	No	Inverts the traversed device predicate.
matchForwardFlow	Boolean	POST/PUT	No	Matches a forward flow.

Field Name	Type	POST/PUT	Required	Description
matchReturnFlow	Boolean	POST/PUT	No	Matches a return flow.
noVlan	Boolean	POST/PUT	No	Matches if the traffic does not belong to a VLAN.
nwDstAddress	String	POST/PUT	No	Matches the network layer destination address.
nwDstLength	Integer	POST/PUT	No	Matches the network layer destination address `nwDstAddress` for the specified prefix length.
nwProto	Integer	POST/PUT	No	Matches the network layer protocol number.
nwSrcAddress	String	POST/PUT	No	Matches the network layer source address.
nwSrcLength	Integer	POST/PUT	No	Matches the network layer source address `nwSrcAddress` for the specified prefix length.
nwTos	Integer	POST/PUT	No	Matches the value of the IP datagram type-of-service (ToS) field.
outPorts	Array of UUID	POST/PUT	No	Matches the list of (interior or exterior) egress ports.
portGroup	UUID	POST/PUT	No	Matches the traffic originated from an exterior port from the specified port group.
tpDst	(Integer, Integer)	POST/PUT	No	Matches the range of the TCP/UDP destination ports. It is a JSON object with two integer fields `start` and `end` defining the boundaries of the port range interval. See the section called "Transport Layer Port Range" [23].
tpSrc	(Integer, Integer)	POST/PUT	No	Matches the range of the TCP/UDP source ports. It is a JSON object with two integer fields `start` and `end` defining the boundaries of the port range interval. See the section called "Transport Layer Port Range" [23].
traversedDevice	UUID	POST/PUT	No	Matches that the traffic traverses the device with the specified identifier.
vlan	Boolean	POST/PUT	No	Matches the VLAN identifier.

Data Link Layer Address Masking

The data link address masking helps to reduce the number of L2 address match conditions.

For example, if you specify `dlDstMask` to be `ffff.0000.0000`, and if `dlDst` is `abcd.0000.0000`, all traffic with the destination MAC address that starts with `abcd` will be matched, regardless of the value of the least significant 32 bits.

Transport Layer Port Range

The port range is a JSON object defining the boundaries of the port number interval to match. The `start` boundary must be smaller than the `end` boundary.

```
{ "start": 80, "end": 400 }
```

The range may be open-ended where, one of the range boundaries but not both may be missing.

```
{ "start": 80 }
{ "end": 400 }
```

Neutron

Media Type `[application/vnd.org.midonet.neutron.Neutron-v3+json]`

```
GET     /neutron
```

This is the root object of the Neutron resource in MidoNet REST API. From this object, clients can discover the URIs for all the Neutron services provided by MidoNet REST API.

Field Name	Type	POST/ PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
firewalls	URI			A GET against this URI returns the list of Neutron firewalls.
floating_ips	URI			A GET against this URI returns the list of Neutron floating IP addresses.
networks	URI			A GET against this URI returns the list of Neutron networks.
ports	URI			A GET against this URI returns the list of Neutron ports.
routers	URI			A GET against this URI returns the list of Neutron routers.
security_groups	URI			A GET against this URI returns the list of Neutron security groups.
security_group_rules	URI			A GET against this URI returns the list of Neutron security group rules.
subnets	URI			A GET against this URI returns the list of Neutron subnets.
add_router_interface_template	String			A PUT against the URI constructed from this template adds a Neutron router interface.
firewall_template	String			URI Template that represents the location of a Neutron firewall.
floating_ip_template	String			URI Template that represents the location of a Neutron floating IP address.
network_template	String			URI template that represents the location of a Neutron network.
port_template	String			URI Template that represents the location of a Neutron port.
remove_router_interface_template	String			A PUT against the URI constructed from this template removes a Neutron router interface.
router_template	String			URI Template that represents the location of a Neutron router.
security_group_template	String			URI Template that represents the location of a Neutron security group.
security_group_rule_template	String			URI template that represents the location of a Neutron security group rule.

Field Name	Type	POST/ PUT	Required	Description
subnet_template	String			URI Template that represents the location of a Neutron subnet.
load_balancer	Object			Object that has the URIs of the load balancer objects: `pools`, `vips`, `members` and `health_monitors`.

Neutron Floating IP

Media Type `[application/vnd.org.midonet.neutron.FloatingIp-v1+json]`

Collection Media Type `[application/vnd.org.midonet.neutron.FloatingIps-v1+json]`

```
GET     /neutron/floating_ips
GET     /neutron/floating_ips/:floatingIpId
POST    /neutron/floating_ips
PUT     /neutron/floating_ips/:floatingIpId
DELETE  /neutron/floating_ips/:floatingIpid
```

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
fixed_ip_address	String	POST/PUT	Yes	The private IP address that the floating IP is associated with in the format `x.x.x.x/y`, such as `10.0.0.100/24`.
floating_ip_address	String	POST/PUT	Yes	The IP address in the format `x.x.x.x/y`, such as `200.0.0.100/24`.
floating_network_id	UUID	POST/PUT	Yes	The identifier of the external network from which the floating IP address was allocated.
router_id	UUID	POST/PUT	Yes	The identifier of the router where the floating IP is NATed.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the floating IP address.
port_id	UUID	POST/PUT	No	ID of the port to which the floating IP is associated with

Neutron HealthMonitor

Media Type `[application/vnd.org.midonet.neutron.lb.HealthMonitor-v1+json]`

Collection Media Type `[application/vnd.org.midonet.neutron.lb.HealthMonitors-v1+json]`

```
GET     /neutron/lb/health_monitors
GET     /neutron/lb/health_monitors/:healthMonitorId
POST    /neutron/lb/health_monitors
```

```
DELETE  /neutron/lb/health_monitors/:healthMonitorId
```

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the health monitor.
admin_state_up	Boolean	POST/PUT	No	The administrative state of the health monitor. The default is *true* (up).
delay	Integer	POST/PUT	No	The minimum time in seconds between regular pings of member.
max_retries	Integer	POST/PUT	No	The number of permissible ping failures before changing the member's status to `INACTIVE`.
pools	Array of (UUID, String, String)	POST/PUT	No	The list of pools associated with this health monitor. Each element is a JSON including the following fields: `pool_id` the identifier of the pool, `status` the pool status, and `status_description` the status description.
timeout	Integer	POST/PUT	No	The maximum number of seconds for a monitor to wait for a ping reply before it times out.
type	String	POST	No	The health monitor type. Allowed values are `PING`, `TCP`, `HTTP`, `HTTPS`. This determines the type of packet sent for the health check.

Neutron Member

Media Type `[application/vnd.org.midonet.neutron.lb.Member-v1+json]`

Collection Media Type `[application/vnd.org.midonet.neutron.lb.Members-v1+json]`

```
GET     /neutron/lb/members
GET     /neutron/lb/members/:memberId
POST    /neutron/lb/members
DELETE  /neutron/lb/members/:memberId
```

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the pool member.
address	String	POST/PUT	No	The IPv4 address of the pool member.
admin_state_up	Boolean	POST/PUT	No	The administrative state of the pool member. The default is `true` (up).
pool_id	UUID	POST/PUT	No	The identifier of the pool resource associated with this member.
protocol_port	Integer	POST/PUT	No	The port on which the traffic will be load balanced.
status	String	POST/PUT	No	The pool member status. Values are `ACTIVE` or `INACTIVE`. It is currently unused.
status_description	String	POST/PUT	No	The status description.
weight	Integer	POST/PUT	No	The proportion of traffic that this member will receive.

Neutron Network

Media Type `[application/vnd.org.midonet.neutron.Network-v1+json]`

Collection Media Type `[application/vnd.org.midonet.neutron.Networks-v1+json]`

```
GET     /neutron/networks
GET     /neutron/networks/:networkId
POST    /neutron/networks
PUT     /neutron/networks/:networkId
DELETE  /neutron/networks/:networkid
```

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the network.
admin_state_up	Boolean	POST/PUT	No	The administrative state of the network. Default is *true* (up).
router:external	Boolean	POST/PUT	No	It indicates whether this network is external, that is administratively owned. The default is *false*.
name	String	POST/PUT	No	The network name.
provider:network_type	String	POST/PUT	No	The network type. The value must be one of the following: `FLAT`, `GRE`, `LOCAL`, `UPLINK`, `VLAN`.
shared	Boolean	POST/PUT	No	Indicates whether this resource is shared among tenants.
status	String			Status of this resource. This field is currently unused.

If a network is created and marked as *external*, MidoNet API also creates an administratively owned router called Provider Router. Provider router is a MidoNet virtual router that serves as the gateway router for the OpenStack Neutron deployment. This router is responsible for forwarding traffic between the Internet and the OpenStack cloud. It is up to the network operator to configure this router. There can be at most one instance of provider router at any time. To locate this router, search for the router with the name *'MidoNet Provider Router'*.

Neutron Pool

Media Type `[application/vnd.org.midonet.neutron.lb.Pool-v1+json]`

Collection Media Type `[application/vnd.org.midonet.neutron.lb.Pools-v1+json]`

```
GET     /neutron/lb/pools
GET     /neutron/lb/pools/:poolId
POST    /neutron/lb/pools
DELETE  /neutron/lb/pools/:poolId
POST    /neutron/lb/pools/:poolId/health_monitors
DELETE  /neutron/lb/pools/:poolId/health_monitors/:healthMonitorId
```

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the pool.
admin_state_up	Boolean	POST/PUT	No	The administrative state of the pool. Default is `true` (up).
description	String	POST/PUT	No	The pool description.
health_monitors	Array of UUID	POST/PUT	No	The list of identifiers representing the health monitors associated with this pool.
lb_method	String	POST/PUT	No	The load balancing method. Only `ROUND_ROBIN` is supported at this time.
members	Array of UUID	POST/PUT	No	The list of identifiers representing the members associated with this pool.
name	String	POST/PUT	No	The pool name.
protocol	String	POST/PUT	No	The protocol for which the pool will load balance. Only `TCP` is currently supported.
provider	String	POST/PUT	No	The provider name of load balancer service.
router_id	UUID	POST/PUT	No	The identifier of the router resource associated with this pool.
status	String	POST/PUT	No	The pool status. The values are `ACTIVE` or `INACTIVE`. It is currently unused.
status_description	String	POST/PUT	No	The pool status description.
subnet_id	UUID	POST/PUT	No	The identifier of the subnet associated with this pool.
vip_id	UUID	POST/PUT	No	The identifier of the VIP resource associated with this pool.

Neutron Port

Media Type `[application/vnd.org.midonet.neutron.Port-v1+json]`

Collection Media Type `[application/vnd.org.midonet.neutron.Ports-v1+json]`

```
GET     /neutron/ports
GET     /neutron/ports/:portId
POST    /neutron/ports
PUT     /neutron/ports/:portId
DELETE  /neutron/ports/:portid
```

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated
mac_address	String	POST/PUT	Yes	The physical (MAC() address of the instance attached to this port.
network_id	UUID	POST	Yes	The identifier of the Neutron network to whcih this port belongs.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the port.
name	String	POST/PUT	No	The name of the port.
admin_state_up	Boolean	POST/PUT	No	The administrative state of the port. The default is *true* (up).
allowed_address_pairs	Array of	POST/PUT	No	The list of address pairs that are allowed to send packets through this port. Each array ele-

Field Name	Type	POST/PUT	Required	Description
	(String, String)			ment is a JSON specifying the `ip_address` and `mac_address`, such as `{ "ip_address": "10.0.0.100", "mac_address": "00:11:22:33:44:55" }`.
binding:profile	(String)	POST	No	The binding information for this port. Currently is a JSON with an `interface_name` field indicating the physical interface to which the port is bound.
binding:host_id	String	POST	No	The identifier of the compute host where the port is bound.
device_id	String	POST	No	The identifier of the device that owns the port.
device_owner	String	POST	No	The device owner.
extra_dhcp_opts	Array of (String, String)	POST	No	The list of additional DHCP options. Each array element is a JSON object which includes the `opt_name` and `opt_value`.
fixed_ips	Array of (String, UUID)	POST/PUT	No	The list of IP addresses assigned to this port. Each array element is a JSON indicating the `ip_address` and `subnet_id`, such as `{ "ip_address": "10.0.0.100", "subnet_id": "00000000-0000-0000-0000-000000000000" }`.
port_security_enabled	Boolean	POST	No	Indicates whether the port security is enabled. The default is `true`.
security_groups	Array of UUID	POST	No	The list of security groups applied to this port.
status	String			The status of this resource. This field is currently unused.

Neutron Router

Media Type `[application/vnd.org.midonet.neutron.Router-v1+json]`

Collection Media Type `[application/vnd.org.midonet.neutron.Routers-v1+json]`

```
GET     /neutron/routers
GET     /neutron/routers/:routerId
POST    /neutron/routers
PUT     /neutron/routers/:routerId
DELETE  /neutron/routers/:routerid
```

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the router.
admin_state_up	Boolean	POST/PUT	No	The administrative state of the router. The default is *true* (up).
external_gateway_info	(UUID, Boolean))	POST/PUT	No	The external gateway information. It is a JSON that includes the `network_id` and `enable_snat`.
gw_port_id	UUID	POST/PUT	No	The identifier of the gateway port on the external network.
name	String	POST/PUT	No	The router name.

Field Name	Type	POST/PUT	Required	Description
routes	Array of (String, String)	POST/PUT	No	The list of routes of this router. Each array element is a JSON indicating the `destination` and `nexthop` IP addresses for each route, such as `{ "destination": "192.168.0.100", "nexthop": "10.0.0.100" }`
status	String			Status of this resource. This field is currently unused.

`external_gateway_info` consists of the following fields:

- `network_id`: ID of the external network. This field is required.
- `enable_snat`: Enabling SNAT allows VMs to reach the Internet. This field is optional and is defaulted to True.

Neutron Router Interface

Media Type `[application/vnd.org.midonet.neutron.RouterInterface-v1+json]`

```
PUT     /neutron/routers/:routerId/add_router_interface
PUT     /neutron/routers/:routerId/remove_router_interface
```

Field Name	Type	POST/PUT	Required	Description
id	UUID			The identifier of the router to which the interface is added or from which the interface is removed.
port_id	UUID	POST/PUT	Yes	The identifier of the interface port.
subnet_id	UUID	POST/PUT	Yes	The identifier of the subnet to which the interface port is allocated.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the router interface.

Neutron Security Group

Media Type `[application/vnd.org.midonet.neutron.SecurityGroup-v1+json]`

Collection Media Type `[application/vnd.org.midonet.neutron.SecurityGroups-v1+json]`

```
GET     /neutron/security_groups
GET     /neutron/security_groups/:securityGroupId
POST    /neutron/security_groups
PUT     /neutron/security_groups/:securityGroupId
DELETE  /neutron/security_groups/:securityGroupId
```

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the security group.
description	String	POST/PUT	No	The description of the security group.
name	String	POST/PUT	No	The security group name.

Field Name	Type	POST/PUT	Required	Description
security_group_rules	Array of Object	POST/PUT	No	The list of security group rules that belong to this security group. See the section called "Neutron Security Group Rule" [31].

Neutron Security Group Rule

Media Type `[application/vnd.org.midonet.neutron.SecurityGroupRule-v1+json]`

Collection Media Type `[application/vnd.org.midonet.neutron.SecurityGroupRules-v1+json]`

```
GET     /neutron/security_group_rules
GET     /neutron/security_group_rules/:securityGroupRuleId
POST    /neutron/security_group_rules
DELETE  /neutron/security_group_rules/:securityGroupRuleId
```

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
direction	String	POST	Yes	The traffic direction to match. The value can be `ingress` or `egress`.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the security group rule.
security_group_id	UUID	POST	Yes	The identifier of the security group to which the rule belongs.
ethertype	String	POST	No	The ethertype to match. Supported types are `ipv4`, `ipv6` and `arp`.
name	String	POST	No	The security group rule name.
port_range_min	Integer	POST	No	The start protocol port number to match.
port_range_max	Integer	POST	No	The end protocol port number to match.
protocol	String	POST	No	The protocol to match. It could be specified in either string or numerical value. Supported protocols are `ICMP` (1), `ICMPv6` (58), `TCP` (6) and `UDP` (17).
remote_group_id	UUID	POST	No	The identifier of the security group against which to match.
remote_ip_prefix	String	POST	No	The IP address in the CIDR format `x.x.x.x/y` to match.

If you want to match on a particular port number, specify that number for both `port_range_min` and `port_range_max`.

Neutron Subnet

Media Type `[application/vnd.org.midonet.neutron.Subnet-v1+json]`

Collection Media Type `[application/vnd.org.midonet.neutron.Subnets-v1+json]`

```
GET     /neutron/subnets
```

```
GET     /neutron/subnets/:subnetId
POST    /neutron/subnets
PUT     /neutron/subnets/:subnetId
DELETE  /neutron/subnets/:subnetid
```

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
cidr	String	POST	Yes	The subnet address in CIDR Format should be x.x.x.x/y, such as 10.0.0.0/24.
network_id	String	POST	Yes	The identifier of the Neutron network.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the subnet.
allocation_pools	Array of (String, String)	POST	No	The IP addresses allocation pools for DHCP. Each array element is a JSON indicating the `start` and `end` of the allocation pool address range, such as `{ "start": "10.0.0.100", "end": "10.0.0.200" }`.
enable_dhcp	Boolean	POST/PUT	No	Indicates whether DHCP is enabled on this subnet. Default is *true* (enabled).
dns_nameservers	Array of String	POST/PUT	No	The IP addresses for the DNS servers.
host_routes	Array of (String, String)	POST/PUT	No	The host routes for this subnet. Each array element is a JSON indicating the `destination` and `nexthop` IP addresses for each route, such as `{ "destination: "192.168.0.100", "nexthop": "10.0.0.100" }`
gateway_ip	String	POST/PUT	No	The IP address for the gateway of this subnet.
ip_version	Integer	POST/PUT	No	The version of IP address (4 or 6). Currently only 4 is supported.
name	String	POST/PUT	No	The subnet name.
shared	Boolean	POST/PUT	No	Indicates whether this resource is shared among tenants.

Neutron VIP

Media Type `[application/vnd.org.midonet.neutron.lb.Vip-v1+json]`

Collection Media Type `[application/vnd.org.midonet.neutron.lb.Vips-v1+json]`

```
GET     /neutron/lb/vips
GET     /neutron/lb/vips/:vipId
POST    /neutron/lb/vips
DELETE  /neutron/lb/vips/:vipId
```

Field Name	Type	POST/PUT	Required	Description
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
tenant_id	String	POST	Yes	The identifier of the tenant that owns the VIP.
address	String	POST/PUT	No	The IPv4 destination address of the traffic to be load balanced.
admin_state_up	Boolean	POST/PUT	No	The administrative state of the resource. Default is `true` (up).

Field Name	Type	POST/PUT	Required	Description
connection_limit	Integer	POST/PUT	No	The maximum amount of open connections using this VIP at any given time.
description	String	POST/PUT	No	The VIP description.
name	String	POST/PUT	No	The VIP name.
pool_id	UUID	POST/PUT	No	The identifier of the pool resource associated with this VIP.
port_id	UUID	POST/PUT	No	The identifier of the port resource associated with this VIP.
protocol	String	POST/PUT	No	The protocol used for load balancing at this VIP. The possible values are HTTP, HTTPS, and TCP. Currently only TCP is supported.
protocol_port	Integer	POST/PUT	No	The TCP port of the traffic to be load balanced. The value must belong to the interval [0, 65535].
session_persistence	(String, String)	POST/PUT	No	The session persistence settings. It is a JSON object with two fields: `type` with possible values APP_COOKIE, HTTP_COOKIE, SOURCE_IP, and `cookie_name` indicating the cookie name.
status	String	POST/PUT	No	The VIP status. Values are ACTIVE or INACTIVE. It is currently unused.
status_description	String	POST/PUT	No	The status description.
subnet_id	UUID	POST/PUT	No	The identifier of the subnet associated with this pool.

Pool

Media Type `[application/vnd.org.midonet.Pool-v1+json]`

Condition Media Type `[application/vnd.org.midonet.collection.Pool-v1+json]`

```
GET     /pools
GET     /load_balancers/:loadBalancerId/pools
GET     /healh_monitors/:healthMonitorId/pools
GET     /pools/:poolId
POST    /pools
POST    /load_balancers/:loadBalancerId/pools
PUT     /pools/:poolId
DELETE  /pools/:poolId
```

A pool is an entity that represents a group of backend load balancer addresses in MidoNet. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
healthMonitor	URI			A GET against this URI returns the health monitor for this pool.
loadBalancer	URI			A GET against this URI returns the load balancer for this pool.
poolMembers	URI			A GET against this URI returns the list of pool members.
vips	URI			A GET against this URI returns the list of VIPs associated with the pool.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.

Field Name	Type	POST/PUT	Required	Description
loadBalancerId	UUID	POST/PUT	Yes	The identifier of the load balancer corresponding to the pool. When using the `/pools` API end-point to create a pool, this field is used to determine the load balancer to which the pool belongs. When using the `/load_balancers/:loadBalancerId/pools` API end-point, this field is mandatory but its value will be ignored.
lbMethod	String	POST/PUT	Yes	The load balancing algorithm. Only `ROUND_ROBIN` is supported.
protocol	String	POST/PUT	No	The protocol used in the load balancing. Only `TCP` is supported.
adminStateUp	Boolean	POST/PUT	No	The administrative state of the pool. The default is *true* (up).
healthMonitorId	UUID	POST/PUT	No	The identifier of the health monitor to monitor the members of the pool.
status	String			The pool status. It can be one of the following: `ACTIVE` or `INACTIVE`.

PoolMember

Media Type `[application/vnd.org.midonet.PoolMember-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.PoolMember-v1+json]`

```
GET      /pool_members
GET      /load_balancers/:loadBalancerId/pools/:poolId/pool_members
GET      /pool_members/:poolMemberId
POST     /pool_members
POST     /load_balancers/:loadBalancerId/pools/:poolId/pool_members
PUT      /pool_members/:poolMemberId
DELETE   /pool_members/:poolMemberId
```

A pool member is an entity that represents a backend load balancer address in MidoNet. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
pool	URI			A GET against this URI returns the pool.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
address	String	POST/PUT	Yes	The IP address of the pool member.
poolId	UUID	POST/PUT	Yes	The identifier of the pool. When using the `/pool_members` API end-point to create a pool member, this field is used to determine the pool to which the member belongs. When using the `/load_balancers/:loadBalancerId/pools/:poolId/pool_members` API end-point, this field is mandatory but its value will be ignored.
protocolPort	Integer	POST/PUT	Yes	The protocol port of the pool member. The value must belong to the interval [0, 65535].
adminStateUp	Boolean	POST/PUT	No	The administrative state of the pool member. The default is *true* (up).
weight	Integer	POST/PUT	No	The weight used for random algorithm. The default it 1.

Field Name	Type	POST/PUT	Required	Description
status	String			The pool member status. It can be one of the following: `ACTIVE` or `INACTIVE`.

Port

Media Type `[application/vnd.org.midonet.Port-v3+json]`

Collection Media Type `[application/vnd.org.midonet.collection.Port-v3+json]`

```
GET     /ports
GET     /ports/:portId
GET     /bridges/:bridgeId/ports
GET     /bridges/:bridgeId/peer_ports
GET     /routers/:routerId/ports
GET     /routers/:routerId/peer_ports
POST    /routers/:routerId/ports
POST    /bridges/:bridgeId/ports
PUT     /ports/:portId
DELETE  /ports/:portId
```

Port is an entity that represents a port on a virtual device (bridge or router) in MidoNet. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
device	URI			A GET against this URI retrieves the device resource to which this port belongs. If the port is a bridge port, it returns a bridge resource. If it is a router port, it returns a router resource.
host	URI			A GET against this URI returns the host where the port is bound. The request succeeds only if the port is an exterior port, bound to a host.
hostInterfacePort	URI			A GET against this URI returns the interface-binding information for this port.
inboundFilter	URI			A GET against this URI returns the inbound filter chain.
link	URI			A POST against this URI links two interior ports. A DELETE against this URI removes the link. The body of the request must contain a link resource. See the section called "Port Link" [20].
outboundFilter	URI			A GET against this URI returns the outbound filter chain.
peer	URI			A GET against this URI returns the peer port. It requires a port to be linked to another port.
portGroups	URI			A GET against this URI returns the port groups of which this port is a member.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
type	String	POST	Yes	Type of port. It must be one of the following: * `Bridge` * `Router` A new router or bridge port is unplugged. Depending on what it is later attached to, it is referred to as an *exterior* or *interior* port.

Field Name	Type	POST/PUT	Required	Description
				An exterior router port is a virtual port that plugs into the VIF of an entity, such as a VM. It can also be a virtual port connected to a host physical port, directly or after implementing tunnel encapsulation. Access to exterior ports is managed by OpenVSwitch (OpenFlow switch). Exterior bridge port is the same as exterior router port but it is a port on a virtual bridge. Upon being bound to an interface, the port becomes exterior and will have the `hostId`, `host`, and `interfaceName` fields be non-null. The `peer` and `peerId` fields will be null. An interior router port is a virtual port that only exists in the MidoNet virtual router network abstraction. It refers to a logical connection to another virtual networking device such as another router. An interior bridge port is the equivalent on a virtual bridge. Upon being linked to a peer, a port will become interior and will have the `peer` and `peerId` fields be non-null. The `hostId`, `host`, and `interfaceName` fields will be null. There is a third type of port, `Vxlan`, which is created automatically when binding a VTEP to a Neutron network. The only operations supported on a port of this type are GET.
adminStateUp	Boolean	POST/PUT	No	The administrative state of the port. If *false* (down), the port stops forwarding packets. If it is a router port, it additionally replies with a *Communication administratively prohibited* ICMP response. The default is *true* (up).
inboundFilterId	UUID	POST/PUT	No	The identifier of the filter chain to be applied for ingress packets.
inboundMirrorIds	Array of UUID	POST/PUT	No	The list of identifiers for the port mirrors to be applied for ingress packets.
insertionIds	Array of UUID	POST/PUT	No	The list of identifiers for the service insertions to be applied to this port.
outboundFilterId	UUID	POST/PUT	No	The identifier of the filter chain to be applied for egress packets.
outboundMirrorIds	Array of UUID	POST/PUT	No	The list of identifiers for the port mirrors to be applied for egress packets.
deviceId	UUID			The identifier of the device (bridge or router) to which this port belongs.
interfaceName	String			The interface name for a bound port. This will be set when binding a port to a host, becoming an exterior port. See the section called "Host Interface Port" [17].
hostId	UUID		No	The identifier of the host where this port is bound. This will be set when binding a port to a host, becoming an exterior port.
peerId	UUID			The identifier of the peer port to which this port is linked. This will be set when linking a port to another peer, becoming an interior port. See the section called "Port Link" [20].
tunnelKey	Integer			The port tunnel key.
vifId	UUID			The identifier of the VIF plugged into the port.

The ports of type `Bridge` include the following additional fields:

Field Name	Type	POST/PUT	Required	Description
vlanId	Integer	POST/PUT	No	The VLAN identifier assigned to this port. On a given bridge, the VLAN identifier can be present at most in one interior port.

The ports of type `Router` include the following additional fields:

Field Name	Type	POST/PUT	Required	Description
networkAddress	String	POST/PUT	Yes	The IP address of the network attached to this port.
networkLength	Integer	POST/PUT	Yes	The network prefix length of the network attached to this port. The value must belong to the interval [0, 32].
portAddress	String	POST/PUT	Yes	The IP address assigned to the port.
portMac	String	POST/PUT	No	The port physical (MAC) address.
bgpStatus	String			If the port is used to advertise routes to one or more BGP neighbors, this field includes the status of the BGP sessions.

The ports of type `Vxlan` include the following additional fields:

Field Name	Type	POST/PUT	Required	Description
vtepId	UUID			The identifier of the VTEP corresponding to this port.

Port Group

Media Type `[application/vnd.org.midonet.PortGroup-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.PortGroup-v1+json]`

```
GET     /port_groups
GET     /port_groups?tenant_id=:tenantId
GET     /ports/:portId/port_groups
GET     /port_groups/:portGroupId
POST    /port_groups
PUT     /port_groups/:portGroupId
DELETE  /port_groups/:portGroupId
```

A port group is a group of ports. Port groups are owned by tenants. A port could belong to multiple port groups as long as they belong to the same tenant. A port group can be specified in the chain rule to filter the traffic coming from all the ports belonging to that the specified group.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
ports	URI			A GET against this URI returns the list of ports in the port group.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
name	String	POST	No	The name of the port group. The maximum length is 255 characters.
stateful	Boolean	POST/PUT	No	Indicates whether the port group is stateful.
tenantId	UUID	POST/PUT	No	The identifier of the tenant that owns the port group.

Query Parameters

Name	Description
tenant_id	The identifier of the tenant to filter the search.

Port Group Port

Media Type `[application/vnd.org.midonet.PortGroupPort-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.PortGroupPort-v1+json]`

```
GET     /port_groups/:portGroupId/ports
GET     /port_groups/:portGroupId/ports/:portId
POST    /port_groups/:portGroupId/ports
DELETE  /port_groups/:portGroupId/ports/:portId
```

A port group port represents a port membership in a port group.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
port	URI			A GET against this URI returns the port for this port membership.
portGroup	URI			A GET against this URI returns the corresponding port group.
portGroupId	UUID	POST	Yes	The identifier of the port group.
portId	UUID	POST	Yes	The identifier of the port in a port group membership.

Route

Media Type: `[application/vnd.org.midonet.Route-v1+json]`

```
GET     /routers/:routerId/routes
GET     /routes/:routeId
POST    /routers/:routerId/routes
DELETE  /routes/:routeId
```

Route is an entity that represents a route on a virtual router in MidoNet. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
router	URI			A GET against this URI returns the router resource.
id	UUID			A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
dstNetworkAddr	String	POST	Yes	The destination IP network address.
dstNetworkLength	Integer	POST	Yes	The destination IP network prefix length. The value must belong to the interval [0, 32].
nextHopGateway	String	POST	Yes	The IP address of the gateway router to which the traffic is forwarded. This field must be present only for `Normal` routes.
nextHopPort	UUID	POST	Yes	The identifier of the next hop port. This field must be present only for `Normal` routes.
srcNetworkAddr	String	POST	Yes	The source IP network address.
srcNetworkLength	Integer	POST	Yes	The source IP network prefix length. The value must belong to the interval [0, 32].
type	String	POST	Yes	The route type. It can be one of the following: `Normal`, `BlackHole`, `Reject`, `Local`.

Field Name	Type	POST/PUT	Required	Description
weight	Integer	POST	Yes	The priority weight of the route. Lower weights take precedence over higher weights. The value must be greater or equal to zero.
learned	Boolean			Indicates whether the route was learned dynamically using a routing protocol.
routerId	UUID			The identifier of the router to which the route belongs.

Router

Media Type `[application/vnd.org.midonet.Router-v3+json]`

Collection Media Type `[application/vnd.org.midonet.collection.Router-v3+json]`

```
GET     /routers
GET     /routers?tenant_id=:tenantId
GET     /routers/:routerId
POST    /routers
PUT     /routers/:routerId
DELETE  /routers/:routerId
```

Router is an entity that represents a virtual router device in MidoNet. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
bgpNetworks	URI			A GET against this URI returns the BGP networks advertised by this router.
bgpPeers	URI			A GET against this URI returns the BGP neighbors for this router.
inboundFilter	URI			A GET against this URI returns the inbound filter chain.
loadBalancer	URI			A GET against this URI returns the load balancer for this router.
outboundFilter	URI			A GET against this URI returns the outbound filter chain.
peerPorts	URI			A GET against this URI returns the interior ports attached to this router.
ports	URI			A GET against this URI returns the list of ports for this router.
routes	URI			A GET against this URI returns the routes for this router.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
adminStateUp	Boolean	POST/PUT	No	The administrative state of the router. If *false* (down), the router replies with a *Communication administratively prohibited* ICMP response and stops forwarding packets. The default is *true* (up).
asNumber	Integer	POST/PUT	No	The Autonomous System Number (ASN) used for BGP routing.
inboundFilterId	UUID	POST/PUT	No	The identifier of the filter chain to be applied to ingress packets before routing.

Field Name	Type	POST/PUT	Required	Description
inboundMirrorIds	Array of UUID	POST/PUT	No	The list of IDs for the mirrors applied to ingress packets.
loadBalancerId	UUID			The layer 4 load balancer for this router.
name	String	POST/PUT	No	The name of the router. The maximum length is 255 characters.
outboundFilterId	UUID	POST/PUT	No	The identifier of the filter chain to be applied to egress packets after routing.
outboundMirrorIds	Array of UUID	POST/PUT	No	The list of IDs for the mirrors applied to egress packets.
tenantId	String	POST/PUT	No	The identifier of the tenant that owns the router.

Query Parameters

Name	Description
tenant_id	The identifier of the tenant to filter the search.

Rule

Media Type `[application/vnd.org.midonet.Rule-v2+json]`

```
GET     /chains/:chainId/rules
GET     /rules/:ruleId
POST    /chains/:chainId/rules
DELETE  /rules/:ruleId
```

Rule is an entity that represents a rule on a virtual router chain in MidoNet. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
type	String	POST	Yes	The rule type. It must be one of the following: `accept`, `continue`, `dnat`, `drop`, `jump`, `rev_dnat`, `rev_snat`, `reject`, `return`, `snat`, `trace`.
condInvert	Boolean	POST	No	Inverts the conjunction of all the other predicates.
dlDst	String	POST	No	Matches the destination physical (MAC) address.
dlSrc	String	POST	No	Matches the source physical (MAC) address.
dlDstMask	String	POST	No	Destination physical (MAC) address mask in the format *xxxx.xxxx.xxxx* where each *x* is a hexadecimal digit.
dlSrcMask	String	POST	No	Source physical (MAC) address mask in the format *xxxx.xxxx.xxxx* where each *x* is a hexadecimal digit.
dlType	Integer	POST	No	Matches the ethertype provided by the data link layer. The value must be in the interval [0x800, 0xFFFF].
fragmentPolicy	String	POST	No	Matches the datagram fragmentation. The value can be one of the following: *any* (matches any fragment), *header* (matches the first fragment, *nonheader* (matches subsequent fragments), *unfragmented* (matches unfragmented datagrams).
inPorts	Array of UUID	POST	No	Matches the list of (interior or exterior) ingress ports.
ipAddrGroupDst	UUID	POST	No	Matches the destination IP address with an IP address from the specified IP address group.

Field Name	Type	POST/PUT	Required	Description
ipAddrGroupSrc	UUID	POST	No	Matches the source IP address with an IP address from the specified IP address group.
invDlDst	Boolean	POST	No	Inverts the destination data link (MAC) address predicate. It has no effect unless the `dlDst` field is also set.
invDlSrc	Boolean	POST/PUT	No	Inverts the source data link (MAC) address predicate. It has no effect unless the `dlSrc` field is also set.
invDlType	Boolean	POST	No	Inverts the data link ethertype predicate. It has no effect unless the `dlType` field is also set.
invInPorts	Boolean	POST	No	Inverts the ingress ports predicate.
invIpAddrGroupDst	Boolean	POST	No	Inverts the destination IP address group predicate.
invIpAddrGroupSrc	Boolean	POST	No	Inverts the source IP address group predicate.
invNwDst	Boolean	POST	No	Inverts the network layer destination address predicate. It has no effect unless the `nwDst` field is also set.
invNwProto	Boolean	POST	No	Inverts the network layer protocol number predicate. It has no effect unless the `nwProto` field is also set.
invNwSrc	Boolean	POST	No	Inverts the network layer source address predicate. It has no effect unless the `nwSrc` field is also set.
invNwTos	Boolean	POST	No	Inverts the network layer type-of-service (ToS) predicate. It has no effect unless the `nwTos` field is also set.
invOutPorts	Boolean	POST	No	Inverts the egress ports predicate.
invPortGroup	Boolean	POST	No	Inverts the port group predicate.
invTpDst	Boolean	POST	No	Inverts the destination TCP/UDP port range predicate.
invTpSrc	Boolean	POST	No	Inverts the source TCP/UDP port range predicate.
invTraversedDevice	Boolean	POST	No	Inverts the traversed device predicate.
matchForwardFlow	Boolean	POST	No	Matches a forward flow.
matchReturnFlow	Boolean	POST	No	Matches a return flow.
noVlan	Boolean	POST	No	Matches if the traffic does not belong to a VLAN.
nwDstAddress	String	POST	No	Matches the network layer destination address.
nwDstLength	Integer	POST	No	Matches the network layer destination address `nwDstAddress` for the specified prefix length.
nwProto	Integer	POST	No	Matches the network layer protocol number.
nwSrcAddress	String	POST	No	Matches the network layer source address.
nwSrcLength	Integer	POST	No	Matches the network layer source address `nwSrcAddress` for the specified prefix length.
nwTos	Integer	POST	No	Matches the value of the IP datagram type-of-service (ToS) field.
outPorts	Array of UUID	POST	No	Matches the list of (interior or exterior) egress ports.
position	Integer	POST	No	The position at which this rule should be inserted. The value must be greater than or equal to 1 and less than or equal to the greatest position in the chain. The default is one (1).
portGroup	UUID	POST	No	Matches the traffic originated from an exterior port from the specified port group.
tpDst	(Integer, Integer)	POST	No	Matches the range of the TCP/UDP destination ports. It is a JSON object with two integer fields `start` and `end` defining the boundaries of the port range interval. See the section called "Transport Layer Port Range" [42].
tpSrc	(Integer, Integer)	POST	No	Matches the range of the TCP/UDP source ports. It is a JSON object with two integer fields `start` and `end` defining the boundaries of the port range interval. See the section called "Transport Layer Port Range" [42].
traversedDevice	UUID	POST	No	Matches that the traffic traverses the device with the specified identifier.

Field Name	Type	POST/PUT	Required	Description
vlan	Boolean	POST	No	Matches the VLAN identifier.
action	String			The action applied by this rule. It can be one of the following: `accept`, `continue`, `drop`, `jump`, `reject`, `return`.
chainId	UUID			The identifier of the chain to which the rule belongs.

The rules of type `jump` include the following fields.

Field Name	Type	POST/PUT	Required	Description
jumpChainId	UUID	POST	Yes	The identifier of the chain where to jump.
jumpChainName	String			The name of the jump chain.

The rules of type `dnat`, `snat`, `rev_dnat` and `rev_dnat` include the following fields.

Field Name	Type	POST/PUT	Required	Description
flowAction	String	POST	Yes	The action to take on each flow. The value must be one of the following: `accept`, `continue`, `return`.

The rules of type `dnat` and `snat` include the following fields.

Field Name	Type	POST/PUT	Required	Description
natTargets	Array of (String, String, Integer, Integer)	POST	No	The list of NAT targets for a forward NAT rule. See: the section called "NAT Targets" [42].

Data Link Layer Address Masking

The data link address masking helps to reduce the number of L2 address match conditions.

For example, if you specify `dlDstMask` to be `ffff.0000.0000`, and if `dlDst` is `abcd.0000.0000`, all traffic with the destination MAC address that starts with `abcd` will be matched, regardless of the value of the least significant 32 bits.

Transport Layer Port Range

The port range is a JSON object defining the boundaries of the port number interval to match. The `start` boundary must be smaller than the `end` boundary.

```
{ "start": 80, "end": 400 }
```

The range may be open-ended where, one of the range boundaries but not both may be missing.

```
{ "start": 80 }
{ "end": 400 }
```

NAT Targets

The NAT targets specify the range of IP addresses and transports ports to use with a forward NAT rule.

```
{
  "addressFrom": "10.0.0.10"
  "addressTo": "10.0.0.100"
  "portFrom": 80
```

```
  "portTo": 400
}
```

Tenant

Media Type `[application/vnd.org.midonet.Tenant-v2+json]`

Collection Media Type `[application/vnd.org.midonet.collection.Tenant-v2+json]`

```
GET     /tenants
GET     /tenants/:tenantId
```

It represents a tenant, or a group of users, in the identity services.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
bridges	URI			A GET against this URI returns the bridges for this tenant.
chains	URI			A GET against this URI returns the chains for this tenant.
port_groups	URI			A GET against this URI returns the port groups for this tenant.
routers	URI			A GET against this URI returns the routers for this tenant.
id	String			The identifier of the tenant unique in the identity system.
description	String			The description of the tenant in the identity system.
enabled	Boolean			Indicates whether the tenant is enabled in the identity system.
name	String			The name of the tenant in the identity system.

Token

Media Type `[application/vnd.org.midonet.Token-v1+json]`

A token represents the info required for the *token authentication* method. It can NOT be retrieved through a GET request, but instead must be retrieved in the body or the header of a login request.

Field Name	Type	POST/PUT	Required	Description
key	String			The authentication token
expires	String			The expiration date for the authentication token.

Trace Request

Media Type `[application/vnd.org.midonet.TraceRequest-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.TraceRequest-v1+json]`

```
GET     /traces
GET     /traces/:traceId
```

```
POST    /traces
PUT     /traces/:traceId
DELETE  /traces/:traceId
```

It represents a request for trace information.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
condition	Condition	POST/PUT	Yes	The match condition for this trace request. See the section called "Condition" [22].
deviceId	UUID	POST/PUT	Yes	The identifier of the device where the traffic is traced.
deviceType	String	POST/PUT	Yes	The device type. It must be one of the following: `BRIDGE`, `PORT` or `ROUTER`.
enabled	String	POST/PUT	Yes	Indicates whether the trace request is enabled. The default is `false` (no).
name	String	POST/PUT	Yes	The name of the trace request.
creationTimeStampMs	Integer	POST/PUT	No	Sets or indicates the creation timestamp for this trace request. If not present, it is set to the current system time.
limit	Integer	POST/PUT	No	The trace request limit. The default is 2^63-1.

Tunnel Zone

Media Type `[application/vnd.org.midonet.TunnelZone-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.TunnelZone-v1+json]`

```
GET     /tunnel_zones
GET     /tunnel_zones/:tunnelZoneId
POST    /tunnel_zones
PUT     /tunnel_zones/:tunnelZoneId
DELETE  /tunnel_zones/:tunnelZoneId
```

A tunnel zone represents a group in which hosts can be included to form an isolated zone for tunneling. They must have unique, case insensitive names per type. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
hosts	URI			A GET against this URI returns the host members for this tunnel zone. See the section called "Tunnel Zone Host" [45].
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
name	String	POST/PUT	Yes	The tunnel zone name. The length must be between 1 and 255 characters.
type	String	POST/PUT	Yes	The tunnel zone type, indicating the protocol and the type of end-points supported by the tunnel. It can be one of the following: * `gre` used between agents with GRE tunnelling * `vxlan` used between agents with VXLAN tunnelling * `vtep` used between an agent and a hardware VTEP with VXLAN tunnelling

Tunnel Zone Host

Media Type `[application/vnd.org.midonet.TunnelZoneHost-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.TunnelZoneHost-v1+json]`

```
GET     /tunnel_zones/:tunnelZoneId/hosts
GET     /tunnel_zones/:tunnelZoneId/hosts/:hostId
POST    /tunnel_zones/:tunnelZoneId/hosts
DELETE  /tunnel_zones/:tunnelZoneId/hosts/:hostId
```

GET requests support in addition the following media types to filter the responses by tunnel zone type.

Media Type `[application/vnd.org.midonet.GreTunnelZoneHost-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.GreTunnelZoneHost-v1+json]`

Hosts in the same tunnel zone share the same tunnel configurations, and they are allowed to create tunnels among themselves.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
host	URI			A GET against this URI returns the host corresponding to this tunnel zone member.
tunnelZone	URI			A GET against this URI returns the tunnel zone.
hostId	UUID	POST	Yes	The identifier of the host member.
ipAddress	String	POST	Yes	The IP address uses by the host as an endpoint for the tunnel in this tunnel zone.
tunnelZoneId	UUID			The identifier of the tunnel zone.

VIP

Media Type `[application/vnd.org.midonet.VIP-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.VIP-v1+json]`

```
GET     /vips
GET     /vips/:vipId
GET     /pools/:poolId/vips
GET     /load_balancers/:loadBalancerId/vips
POST    /vips
POST    /pools/:poolId/vips
PUT     /vips/:vipId
DELETE  /vips/:vipId
```

A VIP is an entity that represents a virtual IP address device for use with load balancers in MidoNet. It contains the following fields:

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
loadBalancer	URI			A GET against this URI returns the load balancer.
pool	URI			A GET against this URI returns the pool.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
address	String	POST/PUT	Yes	The IP address of the VIP.
poolId	UUID	POST/PUT	Yes	The identifier of the pool. When using the `/vips` API end-point to create a pool, this field is used to determine the pool to which the VIP belongs. When using the `/pools/:poolId/vips` API end-point, this field is mandatory but its value will be ignored.
protocolPort	Integer	POST/PUT	Yes	The transport protocol port of the VIP. The value must belong to the interval [0, 65535].
adminStateUp	Boolean	POST/PUT	No	The administrative state of the VIP. The default it *true* (up).
sessionPersistence	String	POST/PUT	No	Indicates the session persistence of the VIP. The allowed values are SOURCE_IP or null.
loadBalancerId	UUID			The identifier of the load balancer object to which the VIP is associated with.

VTEP

Media Type `[application/vnd.org.midonet.VTEP-v2+json]`

Collection Media Type `[application/vnd.org.midonet.collection.VTEP-v2+json]`

```
GET     /vteps
GET     /vteps/:vtepId
POST    /vteps
DELETE  /vteps/:vtepId
```

It is the representation of a hardware VXLAN Tunnel EndPoint, or VTEP, which allows you to merge a Midonet L2 network with physical L2 network over an IP tunnel. Once you create the Midonet VTEP representation of your external VTEP, you can bind Neutron networks to the VTEP's ports.

All properties other than those required in POST are obtained from the external VTEP configuration and not controlled by MidoNet.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
bindings	URI			A GET against this URI returns the list of VTEP bindings to Neutron networks. See the section called "VTEP Binding" [47].
ports	URI			A GET against this URI returns a list of VTEP ports. See the section called "VTEP Port" [47].

Field Name	Type	POST/PUT	Required	Description
vtepBindingTemplate	String			Template for the URI to the VTEP individual bindings.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
managementIp	IP Address	POST	Yes	The VTEP management IP address.
managementPort	Integer	POST	Yes	The VTEP management TCP port. The value must belong to the interval [1, 65535].
tunnelZoneId	UUID	POST	Yes	The identifier of the tunnel zone of type `vtep` used by Midonet to send and receive tunneled traffic to and from the VTEP.
connectionState	String			Indicates whether Midonet could successfully connect to the VTEP. The allowed values are `disconnected`, `connected` and `error`.
name	String			The VTEP name as configured in the OVSDB database.
description	String			The VTEP description as configured in the OVSDB database.
tunnelIpAddrs	Array			The list of IP addresses used as end-points for the VXLAN tunnels, as configured in the OVSDB database.

VTEP Binding

Media Type `[application/vnd.org.midonet.VTEPBinding-v2+json]`

Collection Media Type `[application/vnd.org.midonet.collection.VTEPBinding-v2+json]`

```
GET     /vteps/:vtepId/bindings
GET     /vteps/:vtepId/bindings/:portName/:vlanId
POST    /vteps/:vtepId/bindings
DELETE  /vteps/:vtepId/bindings/:portName/:vlanId
```

Bindings between a VTEP port/VLAN and a Neutron network. Creating a binding creates an IP tunnel through which L2 traffic can pass between the VTEP and Neutron network.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
portName	String	POST	Yes	The name of the VTEP port to be bound to the Neutron network.
networkId	UUID	POST	Yes	The identifier of the Neutron network.
vlanId	Integer	POST	Yes	The VLAN ID with which traffic from the VTEP to Midonet will be tagged. The value must belong to the interval [0, 4095]. If 0, then traffic will not be tagged with a VLAN ID.
vtepId	UUID	POST	No	The identifier of the VTEP.

VTEP Port

Media Type `[application/vnd.org.midonet.VTEPPort-v1+json]`

Collection Media Type `[application/vnd.org.midonet.collection.VTEPPort-v1+json]`

```
GET     /vteps/:vtepId/ports
GET     /vteps/:vtepId/ports/:portName
```

Gets the name and description of all ports on the specified VTEP.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource.
vtep	URI			A GET against this URI returns the VTEP corresponding to this port.
name	String			The port name as configured in the OVSDB database.
description	String			The port description as configured in the OVSDB database.

5. Resource Collection

A collection of a resource is represented by inserting 'collection' right before the resource name in the media type. For example, to get a collection of Tenants V1 you would represent:

```
vnd.org.midonet.Tenant-v1+json
```

as:

```
vnd.org.midonet.collection.Tenant-v1+json
```

See the Query Parameters section of each resource type whether the collection can be filtered.

6. Bulk Creation

The following resources support bulk creation where multiple objects can be created atomically:

- Neutron Network
- Neutron Subnet
- Neutron Port

The URI for the bulk creation is the same as one used to do single object creation. It also expects POST method. The only difference is that the Content-Type must be set to the Collection Media Type specified in each of the resource section above. These special media types indicate to the API server that multiple resource objects are being submitted in the request body.

7. Authentication/Authorization

MidoNet API provides two ways to authenticate: username/password and token. MidoNet uses Basic Access Authentication [1] scheme for username/password authentication. From the client with username 'foo' and password 'bar', the following HTTP POST request should be sent to '/login' path appended to the base URI:

```
POST    /login
Authorization: Basic Zm9vOmJhcg==
```

where `Zm9vOmJhcg==` is the base64 encoded value of `foo:bar`.

If the API sever is configured to use OpenStack Keystone as its authentication service, then the tenant name given in the web.xml file will be used in the request sent to the keystone authentication service. However, you can override this tenant name by specifying it in the request header. :

```
X-Auth-Project: example_tenant_name
```

The server returns 401 Unauthorized if the authentication fails, and 200 if succeeds. When the login succeeds, the server sets 'Set-Cookie' header with the generated token and its expiration data as such:

```
Set-Cookie: sessionId=baz; Expires=Fri, 02 July 2014 1:00:00 GMT
```

where 'baz' is the token and 'Wed, 09 Jun 2021 10:18:14 GM' is the expiration date. The token can be used for all the subsequent requests until it expires. Additionally, the content type is set to a Token json type as such:

```
Content-Type: application/vnd.org.midonet.Token-v1+json;charset=UTF-8
```

with the body of the response set to the token information:

```
{"key":"baz","expires":"Fri, 02 July 2014 1:00:00 GMT"}
```

To send a token instead for authentication, the client needs to set it in `X-Auth-Token` HTTP header:

```
X-Auth-Token: baz
```

The server returns 200 if the token is validated successfully, 401 if the token was invalid, and 500 if there was a server error.

For authorization, if the requesting user attempts to perform operations or access resources that it does not have permission to, the API returns 403 Forbidden in the response. Currently there are only three roles in MidoNet:

- Admin: Superuser that has access to everything
- Tenant Admin: Admin of a tenant that has access to everything that belongs to the tenant
- Tenant User: User of a tenant that only has read-only access to resources belonging to the tenant

Roles and credentials are set up in the auth service used by the API.

[1] http://tools.ietf.org/html/rfc2617

8. List of Acronyms

- API: Application Programmable Interface
- BGP: Border Gateway Protocol
- HTTP: HyperText Transfer Protocol
- ICMP: Internet Control Message Protocol
- JSON: JavaScript Object Notation
- REST: REpresentational State Transfer
- TOS: Type Of Service
- URI: Uniform Resource Identifier
- URL: Uniform Resource Locator
- VIF: Virtual Interface