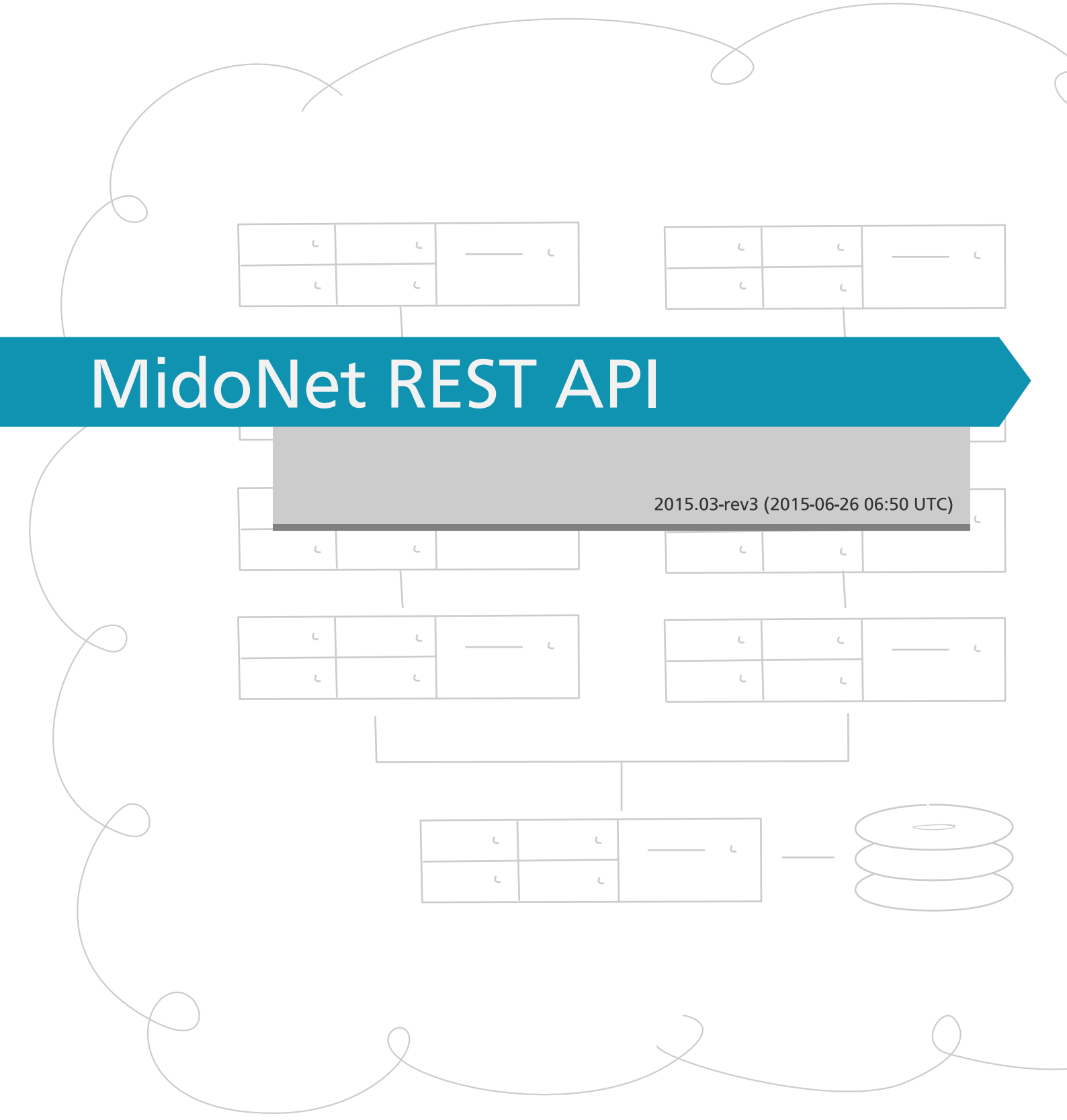

MidoNet REST API

2015.03-rev3 (2015-06-26 06:50 UTC)

midonet

docs.midonet.org

MidoNet REST API

2015.03-rev3 (2015-06-26 06:50 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-v1+json
```

The request above may yield the following output :

```
HTTP/1.1 200 OK
Content-Type: application/vnd.org.midonet.Application-v1+json
{
    "uri": "http://example.org/",
    "version": "1",
    "bridges": "http://example.org/bridges",
    "chains": "http://example.org/chains",
    "hosts": "http://example.org/hosts",
    "portGroups": "http://example.org/port_groups",
    "routers": "http://example.org/routers",
    "bgpTemplate": "http://example.org/bgps/{id}",
    "adRouteTemplate": "http://example.org/ad_routes/{id}",
    "bridgeTemplate": "http://example.org/bridges/{id}",
    "chainTemplate": "http://example.org/chains/{id}",
    "hostTemplate": "http://example.org/hosts/{id}",
    "portTemplate": "http://example.org/ports/{id}",
    "portGroupTemplate": "http://example.org/port_groups/{id}",
    "routeTemplate": "http://example.org/routes/{id}",
    "routerTemplate": "http://example.org/routers/{id}",
    "ruleTemplate": "http://example.org/rules/{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. Old versions are provided for backwards compatibility; in general you should use the newest version available.

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 response that create new server side resources 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 ID 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 ID `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 ID

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, 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.

neutron was added in v5.

Field Name	Type	POST/PUT	Required	Description
tenants	URI			A GET against this URI gets a list of tenants
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 gets a list of bridges
chains	URI			A GET against this URI gets a list of chains
healthMonitors	URI			A GET against this URI gets a list of health monitors
hosts	URI			A GET against this URI gets a list of hosts
loadBalancers	URI			A GET against this URI gets a list of load balancers
portGroups	URI			A GET against this URI gets a list of port groups
poolMembers	URI			A GET against this URI gets a list of pool members
pools	URI			A GET against this URI gets a list of pools
ports	URI			A GET against this URI gets a list of ports

Field Name	Type	POST/PUT	Required	Description
ipAddrGroups	URI			A GET against this URI gets a list of IP address groups
routers	URI			A GET against this URI gets a list of routers
tunnelZones	URI			A GET against this URI gets a list of tunnel zones
vips	URI			A GET against this URI gets a list of VIPs
vteps	URI			A GET against this URI gets a list of VTEPs.
neutron	URI			A GET against this URI gets a available Neutron resources
licenses	URI			A GET against this URI gets a list of available License resources.
licenseStatus	URI			A GET against this URI gets a list of available License Status resources.
adRouteTemplate	String			Template of the URI that represents the location of ad route with the provided ID
bgpTemplate	String			Template of the URI that represents the location of BGP with the provided ID
bridgeTemplate	String			Template of the URI that represents the location of bridge with the provided ID
chainTemplate	String			Template of the URI that represents the location of chain with the provided ID
healthMonitorTemplate	String			Template of the URI that represents the location of the health monitor with the provided ID
hostTemplate	String			Template of the URI that represents the location of host with the provided ID
loadBalancerTemplate	String			Template of the URI that represents the location of the health monitor with the provided ID
portTemplate	String			Template of the URI that represents the location of port with the provided ID
portGroupTemplate	String			Template of the URI that represents the location of port group with the provided ID
poolMemberTemplate	String			Template of the URI that represents the location of the pool member with the provided ID
poolTemplate	String			Template of the URI that represents the location of the pool with the provided ID
ipAddrGroupTemplate	String			Template of the URI that represents the location of port port group with the provided ID
routeTemplate	String			Template of the URI that represents the location of route with the provided ID
routerTemplate	String			Template of the URI that represents the location of router with the provided ID
ruleTemplate	String			Template of the URI that represents the location of rule with the provided ID
tenantTemplate	String			Template of the URI that represents the location of tenant with the provided ID
tunnelZoneTemplate	String			Template of the URI that represents the location of tunnel zone with the provided ID
vipTemplate	String			Template of the URI that represents the location of the vip with the provided ID
vtepTemplate	String			Template of the URI that represents the location of the VTEP with the provided IP address
licenseTemplate	String			Template of the URI that represents the location of the license with the provided id

Application - v4, Deprecated

Media Type: `[application/vnd.org.midonet.Application-v4+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.

LoadBalancers, vips, healthMonitors, pools, poolMembers were added in v4.

Field Name	Type	POST/PUT	Required	Description
Tenants	URI			A GET against this URI gets a list of tenants
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 gets a list of bridges
chains	URI			A GET against this URI gets a list of chains
healthMonitors	URI			A GET against this URI gets a list of health monitors
hosts	URI			A GET against this URI gets a list of hosts
loadBalancers	URI			A GET against this URI gets a list of load balancers
portGroups	URI			A GET against this URI gets a list of port groups
poolMembers	URI			A GET against this URI gets a list of pool members
pools	URI			A GET against this URI gets a list of pools
ipAddrGroups	URI			A GET against this URI gets a list of IP address groups
routers	URI			A GET against this URI gets a list of routers
tunnelZones	URI			A GET against this URI gets a list of tunnel zones
vips	URI			A GET against this URI gets a list of vips
adRouteTemplate	String			Template of the URI that represents the location of ad route with the provided ID
bgpTemplate	String			Template of the URI that represents the location of BGP with the provided ID
bridgeTemplate	String			Template of the URI that represents the location of bridge with the provided ID
chainTemplate	String			Template of the URI that represents the location of chain with the provided ID
healthMonitorTemplate	String			Template of the URI that represents the location of the health monitor with the provided ID
hostTemplate	String			Template of the URI that represents the location of host with the provided ID
loadBalancerTemplate	String			Template of the URI that represents the location of the health monitor with the provided ID
portTemplate	String			Template of the URI that represents the location of port with the provided ID
portGroupTemplate	String			Template of the URI that represents the location of port group with the provided ID
poolMemberTemplate	String			Template of the URI that represents the location of the pool member with the provided ID

Field Name	Type	POST/PUT	Required	Description
poolTemplate	String			Template of the URI that represents the location of the pool with the provided ID
ipAddrGroupTemplate	String			Template of the URI that represents the location of port port group with the provided ID
routeTemplate	String			Template of the URI that represents the location of route with the provided ID
routerTemplate	String			Template of the URI that represents the location of router with the provided ID
ruleTemplate	String			Template of the URI that represents the location of rule with the provided ID
tenantTemplate	String			Template of the URI that represents the location of tenant with the provided ID
tunnelZoneTemplate	String			Template of the URI that represents the location of tunnel zone with the provided ID
vipTemplate	String			Template of the URI that represents the location of the vip with the provided ID

Application v1 has been removed from the API. Application v2 has been removed from the API. Application v3 has been removed from the API.

Neutron - v2

Media Type: `[application/vnd.org.midonet.neutron.Neutron-v1+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.

The load_balancer field was added in version 2.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource
networks	URI			
subnets	URI			
ports	URI			
routers	URI			
floating_ips	URI			
security_groups	URI			
security_group_rules	URI			
network_template	String			URI Template that represents the location of a Neutron network
subnet_template	String			URI Template that represents the location of a Neutron subnet
port_template	String			URI Template that represents the location of a Neutron port
router_template	String			URI Template that represents the location of a Neutron router
add_router_interface_template	String			A PUT against the URI constructed from this template

Field Name	Type	POST/ PUT	Required	Description
				adds a Neutron router interface
remove_router_interface_template	String			A PUT against the URI constructed from this template removes a Neutron router interface
floating_ip_template	String			URI Template that represents the location of a Neutron floating IP
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
load_balancer	Object			Object that has the URIs of the load balancer objects. pools, vips, members and health_monitors.

Neutron - v1, Deprecated

Media Type: `[application/vnd.org.midonet.neutron.Neutron-v1+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
networks	URI			
subnets	URI			
ports	URI			
routers	URI			
floating_ips	URI			
security_groups	URI			
security_group_rules	URI			
network_template	String			URI Template that represents the location of a Neutron network
subnet_template	String			URI Template that represents the location of a Neutron subnet
port_template	String			URI Template that represents the location of a Neutron port
router_template	String			URI Template that represents the location of a Neutron router
add_router_interface_template	String			A PUT against the URI constructed from this template adds a Neutron router interface

Field Name	Type	POST/ PUT	Required	Description
remove_router_interface_template	String			A PUT against the URI constructed from this template removes a Neutron router interface
floating_ip_template	String			URI Template that represents the location of a Neutron floating IP
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

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
name	String	POST/PUT	No	Name of the resource
tenant_id	String	POST	Yes	ID of the tenant that owns the resource
admin_state_up	Bool	POST/PUT	No	The administrative state of the resource. Default is true (up)
external	Bool	POST/PUT	No	Indicates whether this network is external - administraively owned. Default is false
shared	Bool	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 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
name	String	POST/PUT	No	Name of the resource
tenant_id	String	POST	Yes	ID of the tenant that owns the resource
admin_state_up	Bool	POST/PUT	No	The administrative state of the resource. Default is true (up)
ip_version	int	POST/PUT	No	Version of IP (4 or 6) Currently only 4 is supported
shared	Bool	POST/PUT	No	Indicates whether this resource is shared among tenants.
cidr	String	POST	Yes	CIDR of the subnet Format should be x.x.x.x/y, such as 10.0.0.0/24
gateway_ip	String	POST/PUT	No	Gateway IP address of this subnet
enable_dhcp	Bool	POST/PUT	No	Enable/disable DHCP on this subnet. Default is true (enabled)
allocation_pools	Array	POST	No	
host_routes	Array	POST/PUT	No	
dns_nameservers	Array	POST/PUT	No	

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
name	String	POST/PUT	No	Name of the resource
tenant_id	String	POST	Yes	ID of the tenant that owns the resource
admin_state_up	Bool	POST/PUT	No	The administrative state of the resource. Default is true (up)
network_id	UUID	POST	Yes	ID of the network this port belongs to.
mac_address	String	POST/PUT	Yes	MAC address of the instance attached to this port.
fixed_ips	Array	POST/PUT	No	
device_id	String	POST	No	ID of the device that owns the port.
device_owner	String	POST	No	
status	String			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
name	String	POST/PUT	No	Name of the resource
tenant_id	String	POST	Yes	ID of the tenant that owns the resource
admin_state_up	Bool	POST/PUT	No	The administrative state of the resource. Default is true (up)
gw_port_id	UUID	POST/PUT	No	ID of the gateway port on the external network
external_gateway_info	UUID	POST/PUT	No	
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			ID of the router to which an interface is added to or removed from
tenant_id	String	POST	Yes	ID of the tenant that owns the resource
port_id	UUID	POST/PUT	Yes	ID of the interface port
subnet_id	UUID	POST/PUT	Yes	ID of the subnet to which the interface port is allocated in

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
tenant_id	String	POST	Yes	ID of the tenant that owns the resource
floating_ip_address	String	POST/PUT	Yes	IP address in the format x.x.x.x/y, such as 200.0.0.0/24
floating_network_id	UUID	POST/PUT	Yes	ID of the externa network from which the floating IP address was allocated from
router_id	UUID	POST/PUT	Yes	ID of the router where the floating IP is NAT-ed
port_id	UUID	POST/PUT	No	ID of the port to which the floating IP is associated with
fixed_ip_address	String	POST/PUT	Yes	Private IP address that the floating IP is associated with in the format x.x.x.x/y, such as 10.0.0.3/24

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.
name	String	POST/PUT	No	Name of the resource
tenant_id	String	POST	Yes	ID of the tenant that owns the resource
description	String	POST/PUT	No	Description of the resource.
security_group_rules	Array	POST/PUT	No	

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.

Field Name	Type	POST/PUT	Required	Description
name	String	POST	No	Name of the resource
tenant_id	String	POST	Yes	ID of the tenant that owns the resource
security_group_id	UUID	POST	Yes	ID of the security group that the rule belongs to
remote_group_id	UUID	POST	No	ID of the security group to match against
direction	String	POST	Yes	Traffic direction to match: 'ingress' or 'egress'
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".
port_range_min	Integer	POST	No	Start protocol port number to match on
port_range_max	Integer	POST	No	End protocol port number to match on
ethertype	String	POST	No	ethertype to match on. Supported types are "ipv4", "ipv6" and "arp"
remote_ip_prefix	String	POST	No	IP address in the CIDR format (x.x.x.x/y) to match on

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

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
admin_state_up	Bool	POST/PUT	No	The administrative state of the resource. Default is true (up)
description	String	POST/PUT	No	description of the pool resource.
health_monitors	List	POST/PUT	No	List of UUIDs representing health monitors associated with this 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.
lb_method	String	POST/PUT	No	The load balancing method. Only ROUND_ROBIN is supported at this time.
members	List	POST/PUT	No	List of UUIDs representing the members associated with this pool
name	String	POST/PUT	No	Name of the resource
protocol	String	POST/PUT	No	protocol for which the pool will load balance. Only TCP is currently supported.
provider	String	POST/PUT	No	Provider name of loadbalancer service.
router_id	UUID	POST/PUT	No	The identifier of the router resource associated with this pool.
status	String	POST/PUT	No	Values are "ACTIVE" or "INACTIVE" Currently unused.
status_description	String	POST/PUT	No	Description of the status.
subnet_id	UUID	POST/PUT	No	UUID of the subnet associated with this pool.
tenant_id	String	POST	Yes	ID of the tenant that owns the resource

Field Name	Type	POST/PUT	Required	Description
vip_id	UUID	POST/PUT	No	UUID of the VIP resource associated with this Pool.

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
address	String	POST/PUT	No	The IPv4 destination address of the traffic to be load balanced.
admin_state_up	Bool	POST/PUT	No	The administrative state of the resource. Default is true (up)
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 description of this Vip 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.
name	String	POST/PUT	No	Name of the resource.
pool_id	UUID	POST/PUT	No	UUID of the Pool resource associated with this vip.
port_id	UUID	POST/PUT	No	UUID of the Port resource associated with this vip.
protocol	String	POST/PUT	No	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. Must be between 0 and 65535.
session_persistence	Object	POST/PUT	No	Object representing the session persistence settings. It has only two fields: type, a string, with possible values of "APP_COOKIE", "HTTP_COOKIE", "SOURCE_IP", and cookie_name of type String.
status	String	POST/PUT	No	Values are "ACTIVE" or "INACTIVE" Currently unused.
status_description	String	POST/PUT	No	Description of the status.
subnet_id	UUID	POST/PUT	No	UUID of the subnet associated with this resource.
tenant_id	String	POST	Yes	ID of the tenant that owns the resource

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
address	String	POST/PUT	No	The IPv4 address
admin_state_up	Bool	POST/PUT	No	The administrative state of the resource. Default is true (up)
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
pool_id	UUID	POST/PUT	No	UUID of the Pool resource associated with this member.
protocol_port	Integer	POST/PUT	No	The port that the traffic will be load balanced to.
status	String	POST/PUT	No	Values are "ACTIVE" or "INACTIVE" Currently unused.
status_description	String	POST/PUT	No	Description of the status.
tenant_id	String	POST	Yes	ID of the tenant that owns the resource
weight	Integer	POST/PUT	No	The proportion of traffic that this member will receive

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
admin_state_up	Bool	POST/PUT	No	The administrative state of the resource. Default is true (up)
delay	Integer	POST/PUT	No	This is the minimm time in seconds between regular pings of 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.
max_retries	Integer	POST/PUT	No	Number of permissible ping failures before changing the member's status to INACTIVE.
pools	List	POST/PUT	No	List of pools associated with this health monitor.
tenant_id	String	POST	Yes	ID of the tenant that owns the resource
timeout	Integer	POST/PUT	No	Maximum number of seconds for a monitor to wait for a ping reply before it times out.
type	String	POST	No	Valid values are 'PING', 'TCP', 'HTTP', 'HTTPS'. This determines the type of packet sent for the health check.

Router

Media Type: `[application/vnd.org.midonet.Router-v2+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
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	Name of the router. Must be unique within each tenant
tenantId	String			ID of the tenant that owns the router
adminStateUp	Bool	POST/PUT	No	The administrative state of the router. If false (down), the router replies with a 'Communication administratively prohibited' ICMP error and stops forwarding packets. Default is true (up).
loadBalancerId	UUID			Load balancer object to which it is associated with.
loadBalancer	URI			A GET against this URI gets the load balancer object.
ports	URI			A GET against this URI retrieves ports on this router
chains	URI			A GET against this URI retrieves the rule chains on this router
routes	URI			A GET against this URI retrieves the routes on this router
bridges	URI			A GET against this URI retrieves the bridges on this router
peerPorts	URI			A GET against this URI retrieves the interior ports attached to this router
inboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for incoming packets before routing
inboundFilter	URI			A GET against this URI retrieves the inbound filter chain
outboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for outgoing packets after routing
outboundFilter	URI			A GET against this URI retreives the outbound filter chain

Query Parameters

Name	Description
tenant_id	ID of the tenant to filter the search with

Router - v1, Deprecated

Media Type: `[application/vnd.org.midonet.Router-v1+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
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	Name of the router. Must be unique within each tenant
tenantId	String			ID of the tenant that owns the router
adminStateUp	Bool	POST/PUT	No	The administrative state of the router. If false (down), the router replies with a 'Communication administratively prohibited' ICMP error and stops forwarding packets. Default is true (up).
ports	URI			A GET against this URI retrieves ports on this router
chains	URI			A GET against this URI retrieves the rule chains on this router
routes	URI			A GET against this URI retrieves the routes on this router
bridges	URI			A GET against this URI retrieves the bridges on this router
peerPorts	URI			A GET against this URI retrieves the interior ports attached to this router
inboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for incoming packets before routing
inboundFilter	URI			A GET against this URI retrieves the inbound filter chain
outboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for outgoing packets after routing
outboundFilter	URI			A GET against this URI retreives the outbound filter chain

Query Parameters

Name	Description
tenant_id	ID of the tenant to filter the search with

Bridge - v2

Media Type: `[application/vnd.org.midonet.Bridge-v1+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
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	Name of the bridge. Must be unique within each tenant
tenantId	String			ID of the tenant that owns the bridge

Field Name	Type	POST/PUT	Required	Description
adminStateUp	Bool	POST/PUT	No	The administrative state of the bridge. If false (down), the bridge stops forwarding packets. Default is true (up).
ports	URI			A GET against this URI retrieves ports on this bridge
dhcpSubnets	URI			A GET against this URI retrieves dhcpSubnets on this bridge
routers	URI			A GET against this URI retrieves routers on this bridge
macTable	URI			A GET against this URI retrieves the bridge's MAC table
peerPorts	URI			A GET against this URI retrieves the interior ports attached to this bridge
inboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for incoming packes
inboundFilter	URI			A GET against this URI retrieves the inbound filter chain
outboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for outgoing packets
outboundFilter	URI			A GET against this URI retreives the outbound filter chain
vxlanPortId	UUID			ID of the bridge's VXLAN port, which contains the bridge's bindings to a VTEP. Will be null if the bridge has no bindings to a VTEP. Read-only.
vxlanPort	URI			A GET against this URI retrieves the VXLAN port.

Query Parameters

Name	Description
tenant_id	ID of the tenant to filter the search with

Bridge - v1, Deprecated

Media Type: `[application/vnd.org.midonet.Bridge-v1+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
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	Name of the bridge. Must be unique within each tenant
tenantId	String			ID of the tenant that owns the bridge
adminStateUp	Bool	POST/PUT	No	The administrative state of the bridge. If false (down), the bridge stops forwarding packets. Default is true (up).
ports	URI			A GET against this URI retrieves ports on this bridge

Field Name	Type	POST/PUT	Required	Description
dhcpSubnets	URI			A GET against this URI retrieves dhcpSubnets on this bridge
routers	URI			A GET against this URI retrieves routers on this bridge
macTable	URI			A GET against this URI retrieves the bridge's MAC table
peerPorts	URI			A GET against this URI retrieves the interior ports attached to this bridge
inboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for incoming packes
inboundFilter	URI			A GET against this URI retrieves the inbound filter chain
outboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for outgoing packets
outboundFilter	URI			A GET against this URI retreives the outbound filter chain

Query Parameters

Name	Description
tenant_id	ID of the tenant to filter the search with

MacPort

Media Type: `[application/vnd.org.midonet.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
```

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource
vlanId	short			ID of the VLAN to which the port with ID portId belongs. This field is used only in responses to GET requests and will be ignored in POST requests
macAddr	String		Yes	A MAC address in the form `aa:bb:cc:dd:ee:ff`
portId	UUID		Yes	ID of the port to which the packets destined to the macAddr will be emitted

Path Parameters

Name	Description
bridgeId	UUID of the bridge owning the MAC table to query or modify
vlanId	ID of the VLAN owning the MAC table to query or modify
macPortPair	Consists of a MAC address in the form `12-34-56-78-9a-bc` and the destination port's ID, separated by an underscore. For example: `12-34-56-78-9a-bc_01234567-89ab-cdef-0123-4567890abcdef`

MacPort - v1, Deprecated

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

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

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource
macAddr	String		Yes	A MAC address in the form `aa:bb:cc:dd:ee:ff`
portId	UUID		Yes	ID of the port to which packets destined to macAddr will be emitted

Path Parameters

Name	Description
bridgeId	UUID of the bridge owning the MAC table to query or modify
vlanId	ID of the VLAN owning the MAC table to query or modify
macPortPair	Consists of a MAC address in the form `12-34-56-78-9a-bc` and the destination port's ID, separated by an underscore. For example: `12-34-56-78-9a-bc_01234567-89ab-cdef-0123-4567890abcdef`

IP4MacPair

Media Type: `[application/vnd.org.midonet.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		Yes	IP version 4 address in the form `1.2.3.4`
mac	String		Yes	A MAC address in the form `aa:bb:cc:dd:ee:ff`. If ARP replies are enabled on the bridge, the ip will resolve to this MAC

DHCP Subnet

Media Type: `[application/vnd.org.midonet.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 returns or refreshes the representation of this source
subnetPrefix	String	POST/PUT	No	Subnet Prefix in the form of `1.2.3.4`
subnetLength	Integer	POST/PUT	No	Subnet Length (0-32)
defaultGateway	String	POST/PUT	No	Default Gateway in the form `1.2.3.4`
serverAddr	String	POST/PUT	No	DHCP Server Address in the form of `1.2.3.4`
dnsServerAddrs	List(String)	POST/PUT	No	List of DNS Server Addresses in the form of `1.2.3.4`
interfaceMTU	Integer	POST/PUT	No	Interface Maximum Transmission Unit advertised by DHCP

Field Name	Type	POST/PUT	Required	Description
opt121Routes	List(String, Integer, String)	POST/PUT	No	List of DHCP Option 121 routes, each of which consists of \{destination prefix (String, `1.2.3.4` form), destination prefix length (Integer, 0-32), gateway address (String, `1.2.3.4` form)}
hosts	URI			A GET against this URI returns the IP:MAC mappings of this DHCP Host.
enabled	Boolean	POST/PUT	No	Indicates whether the DHCP service is enabled. The default value is True.

DHCPSubnet - v1, Deprecated

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

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

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI returns or refreshes the representation of this source
subnetPrefix	String	POST	No	Subnet Prefix in the form of `1.2.3.4`
subnetLength	Integer	POST	No	Subnet Length (0-32)
defaultGateway	String	POST	No	Default Gateway in the form `1.2.3.4`
serverAddr	String	POST	No	DHCP Server Address in the form of `1.2.3.4`
dnsServerAddrs	List(String)	POST	No	List of DNS Server Addresses in the form of `1.2.3.4`
interfaceMTU	Integer	POST	No	Interface Maximum Transmission Unit advertised by DHCP
opt121Routes	List(String, Integer, String)	POST	No	List of DHCP Option 121 routes, each of which consists of \{destination prefix (String, `1.2.3.4` form), destination prefix length (Integer, 0-32), gateway address (String, `1.2.3.4` form)}
hosts	URI			A GET against this URI returns the IP:MAC mappings of this DHCP Host.

DHCP Host

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

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

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI returns or refreshes the representation of this source
ipAddress	String	POST	Yes	IPv4 address of the host in the form `1.2.3.4`
macAddress	String	POST	Yes	MAC Address of the host in the form `AA.BB.CC.DD.EE.FF`
extraDhcpOpts	List((String, String), (String, String))	POST	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,

Field Name	Type	POST/PUT	Required	Description
				to set the interface MTU: `[{"optName": "26", "optValue": "9000"}]`

DHCP Host - v1, Deprecated

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

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI returns or refreshes the representation of this source
ipAddress	String	POST	Yes	IPv4 address of the host in the form `1.2.3.4`
macAddress	String	POST	Yes	MAC Address of the host in the form `AA.BB.CC.DD.EE.FF`

Port

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

```
GET     /ports
GET     /ports/:portId
GET     /routers/:routerId/ports
GET     /routers/:routerId/peer_ports
GET     /bridges/:bridgeId/ports
GET     /bridges/:bridgeId/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
id	UUID			A unique identifier of the resource
adminStateUp	Bool	POST/PUT	No	The administrative state of the port. If false (down), the port stops forwarding packets. If it is a router port, it adittionally replies with a 'Communication administratively prohibited' ICMP Default is true (up).
deviceId	UUID			ID of the device (bridge or router) that this port belongs to
device	URI			A GET against this URI retrieves the device resource that the port belongs to. If the port is a router port, it gets a router resource, and if it's a bridge port, it gets a bridge resource
type	String	POST	Yes	Type of device port. It must be one of: * Router * Bridge 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. 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 Open-

Field Name	Type	POST/PUT	Required	Description
				VSwitch (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 and DELETE. Deleting a VXLAN port will delete all associated VTEP bindings.
peerId	UUID			ID of the peer port that this port is linked to. This will be set when linking a port to another peer (becoming an interior port)
peer	URI			A GET against this URI retrieves the peer port resource. Requires a port to be linked to another port
networkAddress (Router only)	String	POST	Yes	IP address of the network attached to this port. For example, `192.168.10.32`
networkLength (Router only)	Int	POST	Yes	Prefix length of the network attached to this port (number of fixed network bits)
portAddress (Router only)	String	POST	Yes	IP address assigned to the port
portMac (Router only)	String	POST		Port MAC address
vifId	UUID			ID of the VIF plugged into the port
hostId	UUID		No	ID of the port's host. This will be set when binding a port to a host (becoming an exterior port)
host	URI			The port host's URI. Requires a port to be bound to a host
interfaceName	String			Interface name of a bound port. This will be set when binding a port to a host (becoming an exterior port)
bgps (Router only)	URI			A GET against this URI retrieves BGP configurations for this port
link	URI			Location of the port link resource. A POST against this URI links two interior ports. In the body of the request, 'peerId' must be specified to indicate the peer interior port ID. A DELETE against this URI removes the link
inboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for incoming packets
inboundFilter	URI			A GET against this URI retrieves the inbound filter chain
outboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for outgoing packets
outboundFilter	URI			A GET against this URI retrieves the outbound filter chain
portGroups	URI			A GET against this URI retrieves the port groups that this port is a member of

Field Name	Type	POST/PUT	Required	Description
hostInterfacePort	URI			A GET against this URI retrieves the interface-binding information of this port
vlanId (Bridge only)	Short	POST	No	The VLAN ID assigned to this port. On a given bridge, each VLAN ID can be present at most in one interior port
bindings (Vxlan only)	URI			A GET against this URI retrieves the list of bindings between this port's bridge and its VTEP.
mgmtIpAddr (Vxlan only)	IP Address			The management IP address of the VTEP whose bindings this port contains
mgmtPort (Vxlan only)	Integer			The TCP port used in combination with mgmtIpAddr to manage the VTEP whose bindings this port contains
vni	Integer			The VXLAN network identifier used by the VTEP to identify this port's bridge.

Port - v1, Deprecated

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

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

This port type has been deprecated. Please use the updated v2 Port api described above.

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
id	UUID			A unique identifier of the resource
deviceId	UUID			ID of the device (bridge or router) that this port belongs to
device	URI			A GET against this URI retrieves the device resource that the port belongs to. If the port is a router port, it gets a router resource, and if it's a bridge port, it gets a bridge resource
type	String	POST	Yes	Type of device port. It must be one of: * ExteriorRouter * InteriorRouter * ExteriorBridge * InteriorBridge 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 Open-

Field Name	Type	POST/PUT	Required	Description
				VSwitch (OpenFlow switch). Exterior bridge port is the same as exterior router port but it is a port on a virtual bridge. 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. Interior bridge is the equivalent port type on a virtual bridge.
peerId (Interior)	UUID			ID of the peer port that this port is linked to
peer (Interior)	URI			A GET against this URI retrieves the peer port resource
networkAddress (Router only)	String	POST	Yes	IP address of the network attached to this port. For example `192.168.10.32/27`
networkLength (Router only)	Int	POST	Yes	Prefix length of the network attached to this port (number of fixed network bits)
portAddress (Router only)	String	POST	Yes	IP address assigned to the port
vifId (Exterior and Trunk only)	UUID			ID of the VIF plugged into the port
bgps (Exterior router only)	URI			A GET against this BGP configurations for this port.
link (Interior only)	URI			Location of the port link resource. A POST against this URI links two interior ports. In the body of the request, 'peerId' must be specified to indicate the peer interior port ID. A DELETE against this URI removes the link
inboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for incoming packets
inboundFilter	URI			A GET against this URI retrieves the inbound filter chain
outboundFilterId	UUID	POST/PUT	No	ID of the filter chain to be applied for outgoing packets
outboundFilter	URI			A GET against this URI retrieves the outbound filter chain
portGroups	URI			A GET against this URI retrieves the port groups that this port is a member of
hostInterfacePort	URI			A GET against this URI retrieves the interface-binding information of this port
vlanId (Interior Bridge only)	Short	POST	No	The VLAN ID assigned to this port. On a given bridge, each VLAN ID can be present at most in one interior port

Port Link

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

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

Represents a link between two interior ports. Links are possible between:

- Two router ports.
- A router port and a bridge port
- A router port and a bridge
- A bridge port and a bridge port

- Two Bridges, 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, PUSH'ing and POP'ing 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
portId	UUID			A unique identifier of the port
port	URI			A GET against this URI retrieves the port
peerId	UUID	POST	yes	A unique identifier of the peer port
peer	URI			A GET against this URI retrieves the peer port

Route

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

```
GET     /routes/:routeId
GET     /routers/:routerId/routes
POST    /routers/:routerId/routes
PUT     /routers/:routerId/routes/:routeId
DELETE  /routers/:routerId/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
id	UUID			A unique identifier of the resource
routerId	UUID			ID of the router that this route belongs to
router	URI			A GET against this URI gets the router resource
type	String	POST	Yes	
srcNetworkAddr	String	POST	Yes	Source IP address
srcNetworkLength	Int	POST	Yes	Source network IP address length
dstNetworkAddr	String	POST	Yes	Destination IP address
dstNetworkLength	Int	POST	Yes	Destination network IP address length
weight	Int	POST	Yes	The priority weight of the route. Lower weights take precedence over higher weights
nextHopPort (Normal type only)	UUID	POST	Yes	The ID of the next hop port
nextHopGateway (Normal type only)	String	POST	Yes	IP address of the gateway router to forward the traffic to

Pool

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

```
GET     /load_balancers/:loadBalancerId/pools
POST    /load_balancers/:loadBalancerId/pools
GET     /pools/:poolId
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.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
loadBalancerId	UUID	POST	Yes	Load balancer object to which it is associated with.
loadBalancer	URI			A GET against this URI gets the load balancer object.
protocol	String	POST	No	The read-only value represents the protocol used in the load balancing. Only "TCP" is supported.
lbMethod	String	POST	Yes	Load balancing algorithm. Only “ROUND_ROBIN” is supported.
healthMonitorId	UUID	POST	No	ID of the health monitor object to assign to the pool.
healthMonitor	URI			A GET against this URI gets the health monitor object.
poolMembers	URI			A GET against this URI gets the list of URLs for the member objects.
adminStateUp	Bool	POST/PUT	No	Administrative state of the object.
vips	URI			A GET against this URI gets the list of VIPs associated with the pool.

PoolMember

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

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

A PoolMember 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.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
poolId	UUID	POST	Yes	ID of the pool.
pool	URI			A GET against this URI retrieves the Pool.
address	String	POST/PUT	Yes	IP address of the member.
protocolPort	Int	POST/PUT	Yes	Protocol port of the member.
weight	Int	POST/PUT	No	Weight used for random algorithm. Defaults to 1.
adminStateUp	Bool	POST/PUT	No	Administrative state of the object.
status	String			The status of the object. Values are: UP, DOWN

Port Group

Media Type: `[application/vnd.org.midonet.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
```

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
id	UUID			A unique identifier of the resource
tenantId	UUID			ID of the tenant that this chain belongs to
name	String	POST	Yes	Name of the port group. Unique per tenant
ports	URI			URI for port membership operations

Query Parameters

Name	Description
tenant_id	ID of the tenant to filter the search with
port_id	ID of the port to filter the search with

Port Group Port

Media Type: `[application/vnd.org.midonet.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
```

PortGroupPort represents membership of ports in port groups.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource
portGroupId	UUID			ID of the port group that a port is a member of
portGroup	URI			URI to fetch the port group
portId	UUID	POST	Yes	ID of the port in a port group membership
port	URI			URI to fetch the port

IP Address Group

Media Type: `[application/vnd.org.midonet.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
id	UUID			A unique identifier of the resource
name	String	POST	Yes	Name of the group.
addrs	URI			URI for address membership operations

IP Address Group Address

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

```
GET     /ip_addr_groups/:ipAddrGroupId/versions/4/ip_addrs
GET     /ip_addr_groups/:ipAddrGroupId/ip_addrs/:ip_addr
POST    /ip_addr_groups/:ipAddrGroupId/ip_addrs
DELETE  /ip_addr_groups/:ipAddrGroupId/ip_addrs/:ip_addr
```

IpAddrGroupAddr represents membership of IP address in IP address groups.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource
ipAddrGroupId	UUID			ID of the IP address group that this IP address is a member of
ipAddrGroup	URI			URI to fetch the IP address group
addr	String	POST	Yes	IP Address member in an IP address group

Chain

Media Type: `[application/vnd.org.midonet.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
id	UUID			A unique identifier of the resource
tenantId	UUID			ID of the tenant that this chain belongs to
name	String	POST	Yes	Name of the chain. Unique per tenant
rules	URI			A GET against this URI retrieves the representation of the rules set for this chain

Query Parameters

Name	Description
tenant_id	ID of the tenant to filter the search with

HealthMonitor

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

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

Note

To use this feature, please make sure that health monitoring is activated in the MidoNet Host 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.
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated
delay	Int	POST/PUT	No	Delay for the health check interval in seconds. Defaults to `0`.
timeout	Int	POST/PUT	No	Timeout value for the health check in seconds. Defaults to `0`.
maxRetries	int	POST/PUT	No	Number of times to retry for health check. Defaults to `0`.
type	String	POST/PUT	Yes	A type of the health monitor checking protocol. Only "TCP" is supported in the current version. Read-only property.
adminStateUp	Bool	POST/PUT	No	Administrative state of the object.
status	String			
pools	URI			A GET against this URI retrieves the pools

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:

In this version, dlSrcMask, dlDstMask and fragmentPolicy were added.

Field Name	Type	POST/PUT	Required	Description
chainId	UUID			ID of the chain that this chain belongs to
condInvert	Bool	POST	No	Invert the conjunction of all the other predicates
dlDst	String	POST	No	The data link layer destination that this rule matches on. A MAC address in the form `aa:bb:cc:dd:ee:ff`
dlSrc	String	POST	No	The data link layer source that this rule matches on. A MAC address in the form `aa:bb:cc:dd:ee:ff`
dlType	Short	POST	No	Set the data link layer type (ethertype) of packets matched by this rule. The type provided is not check for validity
dlSrcMask	String	POST	No	Source MAC address mask in the format xxxx.xxxx.xxxx where each x is a hexadecimal digit.
dlDstMask	String	POST	No	Destination MAC address mask in the format xxxx.xxxx.xxxx where each x is a hexadecimal digit.
flowAction	String	POST	No	Action to take on each flow. If the type is snat, dnat, rev_snat and rev_dnat then this field is required. Must be one of accept, continue, return
id	UUID			A unique identifier of the resource
inPorts	UUID	POST	No	The list of (interior or exterior) ingress port UUIDs to match
invDlDst	Bool	POST	No	Set whether the match on the data link layer destination should be inverted (match packets whose data link layer destination is NOT equal to dlDst). Will be stored, but ignored until dlDst is set
invDlSrc	Bool	POST	No	Set whether the match on the data link layer source should be inverted (match packets whose data layer link source is NOT equal to dlSrc). Will be stored, but ignored until dlSrc is set
invDlType	Bool	POST	No	Set whether the match on the data link layer type should be inverted (match packets whose data link layer type is NOT equal to the Ethertype set by dlType. Will be stored, but ignored until dlType is set
invInPorts	Bool	POST	No	Inverts the in_ports predicate. Match if the packet's ingress is NOT in in_ports
invNwDst	Bool	POST	No	Invert the IP dest prefix predicate. Match packets whose destination is NOT in the prefix
invNwProto	Bool	POST	No	Invert the nwProto predicate. Match if the packet's protocol number is not nwProto
invNwSrc	Bool	POST	No	Invert the IP source prefix predicate. Match packets whose source is NOT in the prefix
invNwTos	Bool	POST	No	Invert the nwTos predicate. Match if the packet's protocol number is not nwTos
invOutPorts	Bool	POST	No	Inverts the out_ports predicate. Match if the packet's egress is NOT in out_ports
invTpDst	Bool	POST	No	Invert the destination TCP/UDP port range predicate. Match packets whose dest port is NOT in the range
invTpSrc	Bool	POST	No	Invert the source TCP/UDP port range predicate. Match packets whose source port is NOT in the range
jumpChainId	UUID	POST	No	ID of the jump chain. If the type == jump then this field is required
jumpChainName	String			Name of the jump chain
natTargets	Array of JSON objects	POST	No	
nwDstAddress	String	POST	No	The address part of the IP destination prefix to match
nwDstLength	Int	POST	No	The length of the IP destination prefix to match

Field Name	Type	POST/PUT	Required	Description
nwProto	Int	POST	No	The Network protocol number to match (0-255)
nwSrcAddress	String	POST	No	The IP address of the IP source prefix to match
nwSrcLength	Int	POST	No	The length of the source IP prefix to match (number of fixed network bits)
nwTos	Int	POST	No	The value of the IP packet TOS field to match (0-255)
outPorts	Array of UUID	POST	No	The list of (interior or exterior) egress port UUIDs to match
portGroup	UUID	POST	No	ID of the port group that you want to filter traffic from. If matched, the filter action is applied to any packet coming from ports belonging to the specified port group
position	Int	POST	No	The position at which this rule should be inserted >= 1 and # the greatest position in the chain + 1. If not specified, it is assumed to be 1
tpSrc	Range	POST	No	A JSON representation of the Range object representing the tcp/udp source port range to match, like \{"start":80,"end":400}. When creating an ICMP rule, this field should be set to the ICMP type value. The absence of a Range will be interpreted as "any"
tpDst	Range	POST	No	A JSON representation of the Range object representing the tcp/udp source port range to match, like \{"start":80,"end":400}. When creating an ICMP rule, this field should be set to the ICMP code value. A null value in this field will be interpreted as "any"
fragmentPolicy	String	POST/ PUT		
type	String	POST	Yes	Must be one of these strings: accept, dnat, drop, jump, rev_dnat, rev_snat, reject, return, snat
uri	URI			A GET against this URI refreshes the representation of this resource

How L2 Address masking works

dlDstMask and dlSrcMask help reduce the number of L2 address match rules.

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.

Rule - v1, Deprecated

Media Type: `[application/vnd.org.midonet.Rule-v1+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
chainId	UUID			ID of the chain that this chain belongs to
condInvert	Bool	POST	No	Invert the conjunction of all the other predicates
dlDst	String	POST	No	The data link layer destination that this rule matches on. A MAC address in the form `aa:bb:cc:dd:ee:ff`
dlSrc	String	POST	No	The data link layer source that this rule matches on. A MAC address in the form `aa:bb:cc:dd:ee:ff`

Field Name	Type	POST/PUT	Required	Description
dlType	Short	POST	No	Set the data link layer type (ethertype) of packets matched by this rule. The type provided is not check for validity
flowAction	String	POST	No	Action to take on each flow. If the type is snat, dnat, rev_snat and rev_dnat then this field is required. Must be one of accept, continue, return
id	UUID			A unique identifier of the resource
inPorts	UUID	POST	No	The list of (interior or exterior) ingress port UUIDs to match
invDlDst	Bool	POST	No	Set whether the match on the data link layer destination should be inverted (match packets whose data link layer destination is NOT equal to dlDst). Will be stored, but ignored until dlDst is set
invDlSrc	Bool	POST	No	Set whether the match on the data link layer source should be inverted (match packets whose data layer link source is NOT equal to dlSrc). Will be stored, but ignored until dlSrc is set
invDlType	Bool	POST	No	Set whether the match on the data link layer type should be inverted (match packets whose data link layer type is NOT equal to the Ethertype set by dlType. Will be stored, but ignored until dlType is set
invInPorts	Bool	POST	No	Inverts the in_ports predicate. Match if the packet's ingress is NOT in in_ports
invNwDst	Bool	POST	No	Invert the IP dest prefix predicate. Match packets whose destination is NOT in the prefix
invNwProto	Bool	POST	No	Invert the nwProto predicate. Match if the packet's protocol number is not nwProto
invNwSrc	Bool	POST	No	Invert the IP source prefix predicate. Match packets whose source is NOT in the prefix
invNwTos	Bool	POST	No	Invert the nwTos predicate. Match if the packet's protocol number is not nwTos
invOutPorts	Bool	POST	No	Inverts the out_ports predicate. Match if the packet's egress is NOT in out_ports
invTpDst	Bool	POST	No	Invert the destination TCP/UDP port range predicate. Match packets whose dest port is NOT in the range
invTpSrc	Bool	POST	No	Invert the source TCP/UDP port range predicate. Match packets whose source port is NOT in the range
jumpChainId	UUID	POST	No	ID of the jump chain. If the type == jump then this field is required
jumpChainName	String			Name of the jump chain
natTargets	Array of JSON objects	POST	No	
nwDstAddress	String	POST	No	The address part of the IP destination prefix to match
nwDstLength	Int	POST	No	The length of the IP destination prefix to match
nwProto	Int	POST	No	The Network protocol number to match (0-255)
nwSrcAddress	String	POST	No	The IP address of the IP source prefix to match
nwSrcLength	Int	POST	No	The length of the source IP prefix to match (number of fixed network bits)
nwTos	Int	POST	No	The value of the IP packet TOS field to match (0-255)
outPorts	Array of UUID	POST	No	The list of (interior or exterior) egress port UUIDs to match
portGroup	UUID	POST	No	ID of the port group that you want to filter traffic from. If matched, the filter action is applied to any packet coming from ports belonging to the specified port group

Field Name	Type	POST/PUT	Required	Description
position	Int	POST	No	The position at which this rule should be inserted >= 1 and # the greatest position in the chain + 1. If not specified, it is assumed to be 1
tpSrc	Range	POST	No	A JSON representation of the Range object representing the tcp/udp source port range to match, like \{"start":80,"end":400}. When creating an ICMP rule, this field should be set to the ICMP type value. The absence of a Range will be interpreted as "any"
tpDst	Range	POST	No	A JSON representation of the Range object representing the tcp/udp source port range to match, like \{"start":80,"end":400}. When creating an ICMP rule, this field should be set to the ICMP code value. A null value in this field will be intepreted as "any"
type	String	POST	Yes	Must be one of these strings: accept, dnat, drop, jump, rev_dnat, rev_snat, reject, return, snat
uri	URI			A GET against this URI refreshes the representation of this resource

BGP

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

```
GET     /ports/:portId/bgps
GET     /bgps/:bgpId
POST    /ports/:portId/bgps
DELETE  /bgps/:bgpId
```

BGP is an entity that represents a single set of BGP configurations. 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			A unique identifier of the resource
portId	UUID			ID of the port to set the BGP confgurations on
port	URI			A GET against this URI gets the port resource
localAS	Int	POST	Yes	Local AS number
peerAS	Int	POST	Yes	Peer BGP speaker's AS number
peerAddr	String	POST	Yes	The address of the peer to connect to
adRoutes	URI			A GET against this URi retrieves the advertised routes of this BGP speaker

Route Advertisement

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

```
GET     /bgps/:bgpId/ad_routes
GET     /ad_routes/:adRouteId
POST    /bgps/:bgpId/ad_routes
DELETE  /ad_routes/:adRouteId
```

Advertised Route is an entity that represents an advertising route of BGP. 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
id	UUID			A unique identifier of the resource
bgpId	UUID			ID of the BGP configuration that this route advertisement is configured for
bgp	URI			A GET agains this URI gets the BGP resource
nwPrefix	String	POST	Yes	The prefix address of the advertising route
prefixLength	Int	POST	Yes	The prefix length of the advertising route

Host

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

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

Host is an entity that provides some information about a cluster 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
id	UUID			A unique identifier of the resource. It is usually autogenerated by the daemon running on the host
name	String			The last seen host name
alive	bool			Return true if the node-agent running on the host is connected to ZK
addresses	MultiArray			The of last seen ip addresses visible on the host
interfaces	URI			A GET against this URI gets the interface names on this host
hostInterfaces	MultiArray			List of HostInterface objects belonging to this host
ports	URI			A GET against this URI gets virtual ports bound to the interfaces on this host
floodingProxyWeight	Integer			A non-negative integer whose default value is 1 used to select the proxy for flooding in vxlan gateway

Host - v2, Deprecated

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

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

Host is an entity that provides some information about a cluster 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

Field Name	Type	POST/PUT	Required	Description
id	UUID			A unique identifier of the resource. It is usually autogenerated by the daemon running on the host
name	String			The last seen host name
alive	bool			Return true if the node-agent running on the host is connected to ZK
addresses	MultiArray			The of last seen ip addresses visible on the host
interfaces	URI			A GET against this URI gets the interface names on this host
ports	URI			A GET against this URI gets virtual ports bound to the interfaces on this host
floodingProxyWeight	Integer			A non-negative integer whose default value is 1 used to select the proxy for flooding in vxlan gateway

LoadBalancer

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

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

A LoadBalancer is an entity that represents a virtual load balancer 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
id	UUID	POST	No	A unique identifier of the resource. If this field is omitted in the POST request, a random UUID is generated.
routerId	UUID	No		A unique identifier of the associated router. This property is readonly and not allowed to be updated by users. Please assign load balancers to routers through routers.
router	URI		No	A URI of the associated router.
adminStateUp	Bool	POST/PUT	No	Administrative state of the object.
vips	URI			A GET against this URI gets the list of VIPs associated with the load balancer.
pools	URI			A GET against this URI gets the list pools associated with the load balancer.

Interface

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

```
GET     /hosts/:hostId/interfaces
GET     /hosts/:hostId/interfaces/:interfaceName
POST    /hosts/:hostId/interfaces
PUT     /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
hostId	UUID	PUT		The unique identifier of the host that owns this interface
name	string	PUT		The interface physical name
mac	string	POST / PUT		The interface physical address (MAX)
mtu	integer	POST / PUT		The interface MTU value
status	integer	POST / PUT		Bitmask of status flags. Currently we provide information about UP status and Carrier status (0x01, 0x02 respectively)
type	string			Interface type (the best information that we have been able to infer). Can be: Unknown
Physical	Virtual	Tunnel	addresses	multiArray of InetAddress

Host Command

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

```
GET     /hosts/:hostId/commands
GET     /hosts/:hostId/commands/:hostCommandId
DELETE  /hosts/:hostId/commands/:hostCommandId
```

This is the description of the command generated by an Interface PUT operation. For each host there is going to be a list of HostCommand objects intended to be executed sequentially to make sure that the local host configuration is kept up to date.

Field Name	Type	POST/PUT	Required	Description
uri	URI			A GET against this URI refreshes the representation of this resource
id	UUID			A unique identifier of the resource. It is usually auto-generated by the daemon running on the host
hostId	UUID			The unique identifier of the host that is the target of this command
interfaceName	string			The name of the interface targeted by this command
commands	array of Command			Each Command has three properties: [operation, property, value]. The operation can be one of: SET, DELETE, CLEAR. The property can be one of: mtu, address, mac, interface, midonet_port_id. The value is the value of the operation as a string
logEntries	array of LogEntry			A log entry contains a timestamp (which is a unix time long) and a string which is the error message that was generated at the moment

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 Midolman 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
hostId	UUID	POST/PUT	Yes	A unique identifier of the host resource. It is usually auto-generated by the daemon running on the host
interfaceName	String	POST/PUT	Yes	The interface physical name
portId	UUID	POST/PUT	Yes	A unique identifier of the port resource

Tunnel Zone

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

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

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
id	UUID			A unique identifier of the resource
name	String	POST/PUT	Yes	The name of the resource
type	String	POST	Yes	Tunnel type. Currently this value can only be 'GRE'

Tunnel Zone Host

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

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

The following two GET requests are allowed to specify the media types to filter the responses:

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

The media types below are available for each URI:

- `application/vnd.org.midonet.collection.CapwapTunnelZoneHost-v1+json`
- `application/vnd.org.midonet.collection.GreTunnelZoneHost-v1+json`
- `application/vnd.org.midonet.collection.IpsecTunnelZoneHost-v1+json`

and

- `application/vnd.org.midonet.CapwapTunnelZoneHost-v1+json`

- `application/vnd.org.midonet.GreTunnelZoneHost-v1+json"`
- `application/vnd.org.midonet.IpsecTunnelZoneHost-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
id	UUID			A unique identifier of the resource
tunnelZoneId	UUID			ID of the tunnel zone that the host is a member of
tunnelZone	URI			A GET against this URI retrieves the tunnel zone
hostId	UUID	POST	Yes	ID of the host that you want to add to a tunnel zone
host	URI			A GET against this URI retrieves the host
ipAddress	String	POST/PUT	Yes for GRE tunnel zone type	IP address to use for the GRE tunnels from this host

Tenant

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

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

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

Field Name	Type	POST/PUT	Required	Description
id	String			ID of the tenant unique in the identity system
name	String			Name of the tenant in the identity system
uri	URI			A GET against this URI refreshes the representation of this resource
bridges	URI			A GET against this URI retrieves tenant's bridges
chains	URI			A GET against this URI retrieves tenant's chains
port_groups	URI			A GET against this URI retrieves tenant's port groups
routers	URI			A GET against this URI retrieves tenant's routers

VIP

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

```
GET     /load_balancers/:loadBalancerId/vips
POST    /load_balancers/:loadBalancerId/vips
GET     /vips/:vipId
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.

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.
loadBalancerId	UUID			Load balancer object to which it is associated with. This is deduced from the pool that it is associated with.
loadBalancer	URI			A GET against this URI gets the load balancer object.
poolId	UUID	POST	Yes	ID of the pool.
pool	URI			A GET against this URI gets the pool object.
address	String	POST	Yes	IP address of the VIP.
protocolPort	Int	POST	Yes	Port of the VIP.
sessionPersistence	String	POST	No	Session persistence of the VIP (Only "SOURCE_IP" allowed). This property can be null.
adminStateUp	Bool	POST/PUT	No	Administrative state of the object.

Query Parameters

Query strings for Tenant may vary based on the Authtentication Service used.

Keystone:

Name	Description
marker	ID of the last tenant in the previous search. If this is specified, the GET returns a list of Tenants starting the next item after this ID
limit	Number of items to fetch

VTEP

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

```
GET     /vteps
GET     /vteps/:managementIp
POST    /vteps
```

Midonet representation of a 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.
managementIp	IP Address	POST	yes	The VTEP's IP address.
managementPort	Integer	POST	yes	The TCP port used for management connections to the VTEP.
tunnelZoneId	UUID	POST	yes	ID of the tunnel zone 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. Possible values are CONNECTED and ERROR.
name	String			VTEP's name.

Field Name	Type	POST/PUT	Required	Description
description	String			VTEP's description.
tunnelIpAddrs	List (IP Address)			List of IP addresses available to Midonet to tunnel to the VTEP.
bindings	URI			A GET on this URI retrieves a list of the VTEP's bindings to Neutron networks.
bindingTemplate	String			Template for the URI to the VTEP's individual bindings.
ports	URI			A GET on this URI retrieves a list of the VTEP's ports.

VTEP Binding

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

```
GET     /vteps/:managementIp/bindings
GET     /vteps/:managementIp/bindings/:portName/:vlanId
GET     /ports/:vxLanPortId/bindings
GET     /ports/:vxLanPortId/bindings/:portName/:vlanId
POST    /vteps/:managementIp/bindings
DELETE  /vteps/:ManagementIp/bindings/:portName/:vlanId
```

Bindings between a VTEP port/vlanId 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.
vlanId	Short	POST	Yes	The VLAN ID with which traffic from the VTEP to Midonet will be tagged. Must be between 0 and 4095 inclusive. If 0, then traffic will not be tagged with a VLAN ID.
networkId	UUID	POST	Yes	

VTEP Port

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

```
GET     /vteps/:managementIp/ports
```

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

Field Name	Type	POST/PUT	Required	Description
name	String			The port's name.
description	String			The port's description

System State - v2

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

```
GET     /system_state
PUT     /system_state
```

System State specifies parameters for the various states the deployment might be in. You may modify the system state to make limited changes to the behavior of midonet. For example, changing the "state" field to "UPGRADE" will cause the spawning of new midolman agents to abort.

Field Name	Type	POST/PUT	Required	Description
state	String	PUT	yes	Setting the state field to "UPGRADE" will put the midolman into 'upgrade mode', which will cause all new midolman agents starting up in the deployment to abort the start up process. This is used during deployment wide upgrades to prevent unexpected startups of any midolman agent that might have the wrong version. This state can be reversed by setting the upgrade field to "ACTIVE". The deployment is not in upgrade state by default.
availability	String	PUT	yes	Setting the availability to "READONLY" will cause most API requests to be rejected. The exceptions are only administrative APIs that don't affect the topology: system_state and write_version. This is meant to let the operator stop REST API requests while performing maintenance or upgrades. Setting the availability to "READWRITE" (the default value) allows both GETs and PUT/POST API requests
write_version	string	PUT	yes	The version field determines the version of the topology data that the midolman agents will be writing. This matters during upgrade operations where we will change the write version only after all midolman agents are upgraded. The format of the version field is 'major.minor', where 'major' is the Major version, and 'minor' is the minor version. For example, '1.2'.

SystemState - v1, Deprecated

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

```
GET     /system_state
PUT     /system_state
```

System State specifies parameters for the various states the deployment might be in. You may modify the system state to make limited changes to the behavior of midonet. For example, changing the "state" field to "UPGRADE" will cause the spawning of new midolman agents to abort.

Field Name	Type	POST/PUT	Required	Description
state	String	PUT	yes	Setting the state field to "UPGRADE" will put the midolman into 'upgrade mode', which will cause all new midolman agents starting up in the deployment to abort the start up process. This is used during deployment wide upgrades to prevent unexpected startups of any midolman agent that might have the wrong version. This state can be reversed by setting the upgrade field to "ACTIVE". The deployment is not in upgrade state by default
availability	String	PUT	yes	Setting the availability to "READONLY" will cause most API requests to be rejected. The exceptions are only administrative APIs that don't affect the topology: system_state and write_version. This is meant to let the operator stop REST API requests while performing maintenance or upgrades. Setting the availability to "READWRITE" (the default value) allows both GETs and PUT/POST API requests

Write Version

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

```
GET     /write_version
PUT     /write_version
```

Write Version specifies the version information that is relevant to the midonet deployment as a whole. For example, the "version" field specifies the version of the topology information that all midolman agents must write to, regardless of that midolman agent's version.

Field Name	Type	POST/PUT	Required	Description
version	string	PUT	yes	The version field determines the version of the topology data that the midolman agents will be writing. This matters during upgrade operations where we will change the write version only after all midolman agents are upgraded. The format of the version field is '[major].[minor]', where 'major' is the Major version, and 'minor' is the minor version. For example '1.2'

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

Host Version

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

```
GET     /versions
```

The Host Version specifies version information for each host running in the Midonet deployment.

Field Name	Type	POST/PUT	Required	Description
version	string		yes	the version of Midolman agent running on the host
hostId	string		yes	The the UUID of the host that the Midolman agent is running on
host	string		yes	The URI of the host that the Midolman agent is running on

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