

开发手册

产品版本 : ZStack 2.3.1

文档版本 : V2.3.1

网络资源

版权声明

版权所有©上海云轴信息科技有限公司 2018。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标说明

ZStack商标和其他云轴商标均为上海云轴信息科技有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受上海云轴公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，上海云轴公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

目录

版权声明.....	1
1 引言.....	1
2 系列索引.....	5
3 API规范概述.....	6
3.1 HTTP方法 (HTTP Verbs).....	6
3.2 传参方式.....	6
3.3 HTTP Headers.....	7
3.4 HTTP返回码 (HTTP Status Code).....	8
3.5 API种类.....	8
3.6 API操作.....	9
3.7 基本流程示例.....	11
3.8 Webhook.....	14
3.9 查询API.....	15
4 网络资源.....	22
4.1 二层网络资源相关接口.....	22
4.1.1 创建VXLAN网络池(CreateL2VxlanNetworkPool).....	22
4.1.2 查询VXLAN网络池(QueryL2VxlanNetworkPool).....	26
4.1.3 创建VXLAN网络(CreateL2VxlanNetwork).....	30
4.1.4 查询VXLAN网络(QueryL2VxlanNetwork)	33
4.1.5 创建普通二层网络(CreateL2NoVlanNetwork).....	36
4.1.6 创建二层VLAN网络(CreateL2VlanNetwork).....	38
4.1.7 查询二层VLAN网络(QueryL2VlanNetwork).....	40
4.1.8 删除二层网络(DeleteL2Network).....	43
4.1.9 查询二层网络(QueryL2Network).....	44
4.1.10 更新二层网络(UpdateL2Network).....	46
4.1.11 获取二层网络类型(GetL2NetworkTypes).....	49
4.1.12 挂载二层网络到集群(AttachL2NetworkToCluster).....	51
4.1.13 从集群上卸载二层网络(DetachL2NetworkFromCluster).....	53
4.1.14 创建Vni Range(CreateVniRange).....	56
4.1.15 查询Vni Range(QueryVniRange).....	59
4.1.16 删除Vni Range>DeleteVniRange).....	61
4.2 三层网络相关接口.....	62
4.2.1 创建三层网络(CreateL3Network).....	62
4.2.2 删除三层网络>DeleteL3Network).....	67
4.2.3 查询三层网络(QueryL3Network).....	68
4.2.4 更新三层网络(UpdateL3Network).....	72
4.2.5 获取三层网络类型(GetL3NetworkTypes).....	76
4.2.6 改变三层网络状态(ChangeL3NetworkState).....	77
4.2.7 获取网络DHCP服务所用地址(GetL3NetworkDhcpIpAddress).....	81
4.2.8 从三层网络移除DNS(RemoveDnsFromL3Network).....	83
4.2.9 向三层网络添加DNS(AddDnsToL3Network).....	86
4.2.10 获取空闲IP(GetFreeIp).....	90
4.2.11 检查IP可用性(CheckIpAvailability).....	93
4.2.12 获取IP网络地址容量(GetIpAddressCapacity).....	95
4.2.13 添加IP地址范围(AddIpRange).....	97

4.2.14 删除IP地址范围(DeleteIpRange).....	100
4.2.15 查询IP地址范围(QueryIpRange).....	102
4.2.16 更新IP地址范围(UpdateIpRange).....	104
4.2.17 通过网络CIDR添加IP地址范围(AddIpRangeByNetworkCidr).....	107
4.2.18 获取三层网络Mtu值(GetL3NetworkMtu).....	110
4.2.19 设置三层网络Mtu值(SetL3NetworkMtu).....	112
4.2.20 获取三层网络上路由器的接口地址 (GetL3NetworkRouterInterfaceIp)	114
4.2.21 设置三层网络路由器接口IP(SetL3NetworkRouterInterfaceIp).....	116
4.3 路由资源相关接口.....	117
4.3.1 创建云路由器(CreateVirtualRouterVm).....	118
4.3.2 重连云路由器(ReconnectVirtualRouter).....	125
4.3.3 查询云路由器 (QueryVirtualRouterVm).....	129
4.3.4 查询服务云主机(QueryApplianceVm).....	134
4.3.5 创建云路由规格(CreateVirtualRouterOffering).....	140
4.3.6 查询云路由规格(QueryVirtualRouterOffering).....	144
4.3.7 更新云路由规格(UpdateVirtualRouterOffering).....	147
4.3.8 获取云路由器可加载外部网络(GetAttachablePublicL3ForVRouter).....	150
4.3.9 云路由路由表相关接口.....	154
4.3.9.1 创建云路由路由表(CreateVRouterRouteTable).....	154
4.3.9.2 删除云路由路由表>DeleteVRouterRouteTable).....	158
4.3.9.3 查询云路由路由表(QueryVRouterRouteTable).....	159
4.3.9.4 获取路由器实时路由表(GetVRouterRouteTable).....	160
4.3.9.5 添加云路由路由条目(AddVRouterRouteEntry).....	164
4.3.9.6 删除云路由路由条目>DeleteVRouterRouteEntry).....	168
4.3.9.7 查询云路由路由条目(QueryVRouterRouteEntry).....	169
4.3.9.8 绑定云路由路由表到云路由器(AttachVRouterRouteTableToVRouter).....	172
4.3.9.9 解绑云路由路由表(DetachVRouterRouteTableFromVRouter).....	176
4.3.9.10 查询绑定关系(QueryVirtualRouterVRouterRouteTableRef).....	179
4.4 VPC相关接口.....	182
4.4.1 创建VPC云路由(CreateVpcVRouter).....	182
4.4.2 获取VPC云路由可加载的三层网络(GetAttachableVpcL3Network).....	188
4.4.3 获取实时流量状态(GetVpcVRouterDistributedRoutingConnections).....	192
4.4.4 获取分布式路由是否打开(GetVpcVRouterDistributedRoutingEnabled).....	194
4.4.5 设置分布式路由开关(SetVpcVRouterDistributedRoutingEnabled).....	196

术语表..... 199

1 引言

产品版本

目前与本文档相对应的产品版本为：ZStack 2.3.1

读者对象

本文档详述了ZStack 2.3.1 RESTful API的使用规范，并提供所有API的详细定义。本文档主要适用于以下读者：

- 架构设计师
- 开发工程师
- 测试工程师
- 项目实施人员
- 对ZStack有兴趣研究的相关人员

版本更新

2.3.1

2018/04/03主要更新：

1. 网络拓扑
2. 新版菜单导航、新版首页
3. ZWatch：全新监控报警系统
4. ZStack定制版ISO新增：基于CentOS 7.4深度定制版本
5. 亲和组
6. 增强vCenter接管功能：接管vCenter云盘、基于vCenter云路由网络提供网络服务
7. 一个云主机加载多个ISO
8. 多种策略创建云主机
9. 一个二层网络可用于创建多个三层网络
10. 操作日志/审计全新改版
11. HTTPS安全访问UI管理界面
12. 内部访问业务流量的负载均衡
13. 优化自定义UI
14. 多个场景新增进度条、操作助手和帮助文档，优化UI交互

15.优化部分业务逻辑

2.3.0

2018/02/08主要更新：

1. 专有网络VPC
2. 混合云灾备（混合云版支持）
3. 大屏监控
4. 用户自定义UI
5. ImageStore类型镜像服务器支持Ceph类型主存储
6. 支持vSwitch
7. 支持vCenter资源同步
8. ESXi云主机支持扁平网络
9. 云主机更换操作系统
- 10.跨NFS存储数据迁移
- 11.虚拟IP支持QoS
- 12.支持AD认证
- 13.云主机自定义MAC地址
- 14.强化浏览器上传镜像功能
- 15.新增云盘镜像资源
- 16.数据云盘扩容
- 17.数据云盘规格支持QoS
- 18.停止NeverStop状态的云主机
- 19.开放云路由公网IP，并支持同一虚拟IP多网络服务复用
- 20.支持USB设备透传，强化外接设备透传功能
- 21.增加VDI SPICE流量优化选项
- 22.支持修改已设置的存储网络
- 23.支持设置VXLAN对普通账户的配额
- 24.支持ImageStore类型镜像服务器间的数据同步
- 25.管理节点数据库自动备份到远程服务器
- 26.多个场景新增进度条、操作助手和帮助文档，优化UI交互
- 27.优化部分业务逻辑

2.2

2017/10/16主要更新：

1. 公有网络创建云主机
2. 自定义DHCP模式
3. 新增系统网络
4. 云主机根云盘扩容
5. 浏览器添加镜像（目前支持ImageStore类型镜像服务器）
6. 管理节点高可用：多网络配置
7. 跨Ceph存储数据迁移
8. 增强Ceph存储功能
9. 增强VDI功能
10. LDAP自定义过滤规则
11. 增强裸机管理
12. 单集群支持多类型主存储（目前支持本地存储+NFS/SMP类型）
13. 更换License支持本地上传
14. 共享存储指定存储网络，增强云主机高可用
15. 多个场景新增进度条、操作助手和帮助文档，优化UI交互
16. 优化部分业务逻辑

2.1

2017/08/14主要更新：

1. VDI
2. 裸机管理
3. GPU透传
4. 智能报警
5. 集群挂载多个主存储
6. 新版定时器
7. 静态路由
8. User Data导入
9. 云路由加载多个公有网络
10. 增量升级

11. 优化部分业务逻辑

2 系列索引

ZStack 2.3.1 开发手册系列索引如下：

- 《ZStack 2.3.1 开发手册 云资源池》
- 《ZStack 2.3.1 开发手册 硬件设施》
- 《ZStack 2.3.1 开发手册 网络资源》
- 《ZStack 2.3.1 开发手册 网络服务》
- 《ZStack 2.3.1 开发手册 vCenter》
- 《ZStack 2.3.1 开发手册 平台运维》
- 《ZStack 2.3.1 开发手册 平台管理》
- 《ZStack 2.3.1 开发手册 设置》
- 《ZStack 2.3.1 开发手册 系统全局相关》

3 API规范概述

ZStack 2.3.1提供原生RESTful支持。您可以通过REST定义的架构设计原则和约束条件，并使用支持HTTP的编程语言进行开发。

3.1 HTTP方法 (HTTP Verbs)

当前API支持如下操作资源的方法：

方法名	描述
GET	获取资源信息。 所有的查询API以及读API均使用该方法。
POST	创建一个资源。
PUT	修改一个资源。 所有对资源的修改操作，以及类RPC调用的操作，例如启动虚拟机，均使用该方法。
DELETE	删除一个资源。

3.2 传参方式

URL、Query String、HTTP body三种方式均可用于传参。每种方式可以单独使用，也可以混合使用，具体使用哪种传参方式由具体API决定。

URL传参

当对某具体资源进行操作时，资源的UUID通过编码到URL的方式进行传参。

例如启动一个UUID为 `f97143d60f1042c9badd9a1336d3c105`的虚拟机，URL格式为：

```
zstack/v1/vm-instances/f97143d60f1042c9badd9a1336d3c105/actions
```

这里UUID编码到URL路径当中。

Query String传参

所有使用HTTP GET方法的API均使用Query String传参。

例如查询所有状态为Running的虚拟机，URL格式为：

```
zstack/v1/vm-instances?condition=state=Running
```

HTTP Body传参

当使用POST方法创建一个资源，或PUT方法修改一个资源时，除通过URL传参的部分外，剩余参数均通过HTTP Body传参。

例如在指定物理机上启动一个虚拟机：

```
PUT zstack/v1/vm-instances/f97143d60f1042c9badd9a1336d3c105/actions
{
  "startVmInstance": {
    "hostUuid": "8aef7e3a53b34eedaa05027a919156d9"
  }
}
```

这里虚拟机的UUID通过URL传参，参数`hostUuid`则通过HTTP Body传递。

3.3 HTTP Headers

当前API使用如下自定义HTTP Headers：

Authorization

除了少数API外（例如登录API），使用ZStack API前都需要一个会话(session)，在调用API时通过Authorization HTTP Header传递会话UUID。该Header的格式为：

```
Authorization: OAuth 会话UUID
```

举例：

```
Authorization: OAuth 34cbfddd470a47d8bdb0727cd2182618
```



注：OAuth和会话UUID之间用空格分隔。

X-Job-UUID

对于异步API，可以通过X-Job-UUID HTTP Header来指定该API Job的UUID，例如：

```
X-Job-UUID: d825b1a26f4e474b8c59306081920ff2
```

如果未指定该HTTP Header，ZStack会自动为API Job生成一个UUID。



注：

X-Job-UUID必须为一个v4版本的UUID（即随机UUID）字符串去掉连接符-。ZStack会验证X-Job-UUID格式的合法性，并对非法的字符串返回一个400 Bad Request的错误。

X-Web-Hook

对于异步API，可以通过X-Web-Hook HTTP Header指定一个回调URL用于接收API 返回。通过使用回调URL的方法，调用者可以避免使用轮询去查询一个异步API的执行结果。举例：

```
X-Web-Hook: http://localhost:5000/api-callback
```

X-Job-Success

当使用了X-Web-Hook回调的方式获取异步API结果时，ZStack推送给回调URL的HTTP Post请求中会包含X-Job-Success HTTP Header指明该异步API的执行结果是成功还是失败。例如：

```
X-Job-Success: true
```

当值为true时执行成功，为false时执行失败。

3.4 HTTP返回码 (HTTP Status Code)

ZStack使用如下返回码：

返回码	意义
200	API执行成功。
202	API请求已被ZStack接受，用户需要通过轮询或Web Hook的方式获取API结果。该返回码只在调用异步API时出现。
400	API请求未包含必要的参数或包含了非法的参数。具体信息可以从HTTP Response Body获得。
404	URL不存在，通常是指定了错误的API URL。如果访问的URL是异步API返回的轮询地址，表示该轮询地址已经过期。
405	API调用使用了错误的HTTP方法，例如在创建一个资源的时候用了GET方法而不是POST方法。
500	ZStack RESTful终端遭遇了一个内部错误。
503	API所执行的操作引发了一个错误，例如资源不足无法创建虚拟机。错误的具体信息可以从HTTP Response Body。

3.5 API种类

ZStack的API分为同步API和异步API两种：

同步API

所有使用GET方法的API都是同步API，调用方收到的HTTP Response中直接包含了API的结果。例如：

```
GET zstack/v1/zones/f3fa7671894a40f6a73f5bfc7d90c126

{
  "inventory": {
    "uuid": "f3fa7671894a40f6a73f5bfc7d90c126",
    "name": "zone1",
    "description": "test",
    "state": "Enabled",
    "type": "zstack",
    "createDate": "Jan 6, 2017 3:51:16 AM",
    "lastOpDate": "Jan 6, 2017 3:51:16 AM"
  }
}
```

异步API

除了登录相关的API外，所有不使用GET方法的API都为异步API。用户调用一个异步API成功后会收到202返回码以及 Body中包含的一个轮询地址（location字段），用户需要周期性的GET该轮询地址以获得API的执行结果。例如：

```
Status Code: 202

Body:

{
  "location": "http://localhost:8989/v1/api-jobs/967a26b7431c49c0b1d50d709ef1aef3"
}
```

通常情况下GET一个轮询地址可以得到四种返回：

1. 202返回码表示该API仍在处理中，用户需要继续轮询。
2. 200返回码表示API执行成功，Body中包含API结果。
3. 503返回码表示API执行失败，Body中包含错误码。
4. 404返回码，则表示轮询地址已经过期，产生这种结果的原因可能是用户访问了一个错误的轮询地址，或者太久没有访问该轮询地址（例如超过2天没有访问），该轮询地址已经被删除。

异步API也可以用Web Hook的方式获得结果，具体方法见后面章节。

3.6 API操作

跟所有的RESTful API类似，绝大多数ZStack API执行的是CRUD（Create, Read, Update, Delete）操作，以及类RPC操作。

创建资源

所有资源的创建都使用POST方法，参数通过HTTP Body传递，例如创建一个虚拟机：

```
POST zstack/v1/vm-instances
```

```
Authorization : OAuth 0c234e29a2ad4ff4b0d97d4f3b47c6cf
```

```
{
  "params": {
    "l3NetworkUuids": ["37a701c7fe4a40758da15593aedd8aff"],
    "defaultL3NetworkUuid": "37a701c7fe4a40758da15593aedd8aff",
    "dataDiskOfferingUuids": [],
    "name": "TestVm",
    "description": "Test",
    "systemTags": [],
    "instanceOfferingUuid": "dd53f94b58924510b0122e40799a4114",
    "type": "UserVm",
    "imageUuid": "cc7b56780879409f98c1f992b75a12b0"
  }
}
```

查询资源

资源的查询使用GET方法，查询条件通过Query String传参，例如查询集群`cluster1`中名字**不等于**`web-vm`的虚拟机：

```
GET zstack/v1/vm-instances?condition=name!=web-vm&condition=cluster.name=cluster1
```

```
Authorization : OAuth 0c234e29a2ad4ff4b0d97d4f3b47c6cf
```

如果已知资源的UUID，要直接获取该资源的信息，直接使用GET方法不加任何查询条件，例如：

```
GET zstack/v1/vm-instances/56f0fd314a2647ffb4f9565f6d05858e
```

```
Authorization : OAuth 0c234e29a2ad4ff4b0d97d4f3b47c6cf
```

返回UUID为`56f0fd314a2647ffb4f9565f6d05858e`虚拟机的信息。

删除资源

删除资源使用DELETE方法，被删除资源的UUID编码在URL中，例如：

```
DELETE zstack/v1/vm-instances/56f0fd314a2647ffb4f9565f6d05858e
```

```
Authorization : OAuth 0c234e29a2ad4ff4b0d97d4f3b47c6cf
```

删除UUID为`56f0fd314a2647ffb4f9565f6d05858e`的虚拟机。

修改资源与类RPC操作

但由于IaaS本身业务的性质，一部分操作更类似于RPC（远程调用）而非CRUD操作，例如启动虚拟机。根据RESTful API的一些最佳实践，ZStack将这些操作都归为资源的actions子资源，例如启动虚拟机、停止虚拟机都是对虚拟机的actions子资源进行操作。举个例子：

启动虚拟机：

```
PUT zstack/v1/vm-instances/d46841bd4ebd47f8bf0bed85c3bdf0db/actions
{
  "startVmInstance": {}
}
```

停止虚拟机：

```
PUT zstack/v1/vm-instances/d46841bd4ebd47f8bf0bed85c3bdf0db/actions
{
  "stopVmInstance": {}
}
```

在上面的例子中，两个操作都访问的是相同的URL/*v1/vminstances/d46841bd4ebd47f8bf0bed85c3bdf0db/actions*，具体的操作类型由包含在Body中的字段名表示，例如stopVmInstance，如果该API包含额外参数，则包含在操作字段名对应的map中。

资源操作的具体字段名和例子参考每个API的详细文档。

3.7 基本流程示例

在下例中，我们会创建一个Zone，以展示API使用的基本流程：

登录

使用API的第一步是登录以获取一个Session UUID，以供后续API调用使用。

```
PUT zstack/v1/accounts/login
body:
{
  "loginByAccount": {
    "password": "b109f3bbbc244eb82441917ed06d618b9008dd09b3befd1b5e07394c
706a8bb980b1d7785e5976ec049b46df5f1326af5a2ea6d103fd07c95385ffab0cacbc86",
    "accountName": "admin"
  }
}
```

这里的密码是用sha512哈希后的结果。

API返回如下：

```
status code: 200

body:
{
  "inventory": {
    "uuid": "00d038b699b74e76a01705918d48d939",
    "accountUuid": "36c27e8ff05c4780bf6d2fa65700f22e",
    "userUuid": "36c27e8ff05c4780bf6d2fa65700f22e",
    "expiredDate": "Jan 1, 2017 11:31:06 AM",
    "createDate": "Jan 1, 2017 9:31:06 AM"
  }
}
```

返回内容中包含账户UUID等其他字段，我们需要的session UUID包含在字段uuid中：*00d038b699b74e76a01705918d48d939*。

创建Zone

```
POST zstack/v1/zones

headers:

Authorization: OAuth 00d038b699b74e76a01705918d48d939

body:
{
  "params": {
    "name": "Zone1",
    "description": "Test"
  }
}
```

由于创建Zone操作是一个异步API，API返回不是直接的结果，而是一个轮询地址：

```
status code: 202

body:
{
  "location": "http://localhost:8989/v1/api-jobs/d0345d3ddcae485f8170572b15a2b581"
}
```

用户需要周期性的轮询API结果：

```
GET http://localhost:8989/v1/api-jobs/d0345d3ddcae485f8170572b15a2b581
```

```
Authorization: OAuth 00d038b699b74e76a01705918d48d939
```

如果API还未执行完成，上述GET操作得到的仍然是202返回码和上述轮询地址。当操作完成时，得到的结果如下：

```
status code: 200

body:

{
  "inventory": {
    "uuid": "f52fe55b64094ceb99b3893a238c4931",
    "name": "Zone1",
    "description": "Test",
    "state": "Enabled",
    "type": "zstack",
    "createDate": "Jan 1, 2017 9:31:07 AM",
    "lastOpDate": "Jan 1, 2017 9:31:07 AM"
  }
}
```

查询Zone

要获取创建的Zone的信息，可以用GET查询：

```
GET zstack/v1/zones/f52fe55b64094ceb99b3893a238c4931

Authorization: OAuth 00d038b699b74e76a01705918d48d939
```

返回：

```
status code: 200

body:

{
  "inventory": {
    "uuid": "f52fe55b64094ceb99b3893a238c4931",
    "name": "Zone1",
    "description": "Test",
    "state": "Enabled",
    "type": "zstack",
    "createDate": "Jan 1, 2017 9:31:07 AM",
    "lastOpDate": "Jan 1, 2017 9:31:07 AM"
  }
}
```

```
}
```

登出

当所有API调用完毕，我们需要对已登录的session进行登出操作：

```
DELETE zstack/v1/accounts/sessions/00d038b699b74e76a01705918d48d939
```

返回

```
status code: 200
```

3.8 Webhook

对于异步API使用轮询的方式查询操作结果是一种低效的方式，为此ZStack提供Webhook的方式主动推送异步API结果给调用者。

要使用Webhook功能，调用者只需在HTTP Headers中指定X-Job-UUID和X-Web-Hook即可。以上面创建Zone为例，使用Webhook的API版本为：

```
POST zstack/v1/zones
```

headers:

```
Authorization: OAuth 00d038b699b74e76a01705918d48d939
X-Job-UUID: d0345d3ddcae485f8170572b15a2b581
X-Web-Hook: http://127.0.0.1:8989/rest-webhook
```

body:

```
{
  "params": {
    "name": "Zone1",
    "description": "Test"
  }
}
```

API返回仍然是202返回码和一个轮询地址，但调用者无需再轮询。API执行成功后，结果会被推送到

<http://127.0.0.1:8989/rest-webhook>

```
POST http://127.0.0.1:8989/rest-webhook
```

headers:

```
X-Job-Success: true
X-Job-UUID: d0345d3ddcae485f8170572b15a2b581
```

body:

```
{
  "inventory": {
    "uuid": "f52fe55b64094ceb99b3893a238c4931",
  }
}
```

```
{
  "name": "Zone1",
  "description": "Test",
  "state": "Enabled",
  "type": "zstack",
  "createDate": "Jan 1, 2017 9:31:07 AM",
  "lastOpDate": "Jan 1, 2017 9:31:07 AM"
}
```

推送的结果之中，X-Job-Success指明了API执行成功与否，X-Job-UUID包含值跟API调用时的X-Job-UUID相同，调用者可以对应结果和API。

3.9 查询API

用户可以用GET方法对一个资源进行查询，并且可以像MySQL一样指定多个查询条件、排序方式、选择字段、以及进行跨表查询等等。

支持超过400万个单项查询条件，以及400万阶乘的组合查询条件。

单表查询

例如：

```
GET zstack/v1/vm-instances?q=name=vm1
```

查询名字为`vm1`的虚拟机。

例如：

```
GET zstack/v1/vm-instances?q=name=vm1&q=state=Running
```

查询名字为`vm1`并且状态为`Running`的虚拟机。

这两个例子都是对虚拟机资源本身查询，反应到数据库层面还属于单表查询。

跨表查询

我们可以通过.进行跨表查询。

例如：

```
GET zstack/v1/vm-instances?q=vmNics.ip=192.168.10.100
```

查询IP地址为`192.168.10.100`的虚拟机，这里对虚拟机和网卡两张表进行了跨表查询。

例如：

```
GET zstack/v1/vm-instances?q=host.managementIp=10.10.20.3
```

查询IP为`10.10.20.3`上运行的所有虚拟机。这里对虚拟机和物理机两张表进行了跨表查询。

所有资源的查询API都支持下列参数

名字	类型	位置	描述	可选值	起始版本
q (可选)	List	query	见 查询条件 。省略该字段将返回所有记录，返回记录数的上限受限于limit字段		0.6
limit (可选)	Integer	query	最多返回的记录数，类似MySQL的limit，默认值1000		0.6
start (可选)	Integer	query	起始查询记录位置，类似MySQL的offset。跟limit配合使用可以实现分页		0.6
count (可选)	Boolean	query	计数查询，相当于MySQL中的count()函数。当设置成true时，API只返回的是满足查询条件的记录数		0.6
groupBy (可选)	String	query	以字段分组，相当于MySQL中的group by关键字。例如groupBy=type		1.9
replyWithCount (可选)	Boolean	query	见 分页查询		0.6
sort (可选)	String	query	以字段排序，等同于MySQL中的sort by关键字，例如sort=+ip。必须跟+或者-配合使用，+表示升序，-表示降序,后面跟排序字段名	`+`字段名, `-`字段名	0.6
sortDirection (可选)	String	query	字段排序方向，必须跟sortBy配合使用	ascdesc	0.6
fields (可选)	List	query	指定返回的字段，等同于MySQL中的select字段功能。例如fields=name,uuid，则只返回满足条件记录的name和uuid字段		0.6

查询条件

的查询条件类似于MySQL数据库。

例如：

```
uuid=bfa67f956afb430890aa49db14b85153
totalCapacity>2000
```

```
vmInstanceUuid not null
```



注:

- 字段名、查询操作符、匹配值三者之间不能有任何空格。
- 例如`uuid = 25506342d1384c07b7342373a57475b9`就是一个错误的查询条件，必须写为`uuid=25506342d1384c07b7342373a57475b9`。

多个查询条件之间是与关系。总共支持10个查询操作符：

1. `=`: 等于，例如：

```
vmInstanceUuid=c4981689088b40f98d2ade2548c323da
```

2. `!=`: 不等于，例如：

```
vmInstanceUuid!=c4981689088b40f98d2ade2548c323da
```

3. `>`: 大于

4. `<`: 小于

5. `>=`: 大于等于

6. `<=`: 小于等于

7. `?= in`操作符，测试字段值是否在一个集合。集合中的值以,分隔。例如测试`uuid`是否属于某个集合：

```
uuid?=25506342d1384c07b7342373a57475b9,bc58d68090ac42358c0cb0fe72e3287f
```

8. `!?= not in`操作符，测试字段值是否**不属于**一个集合。集合中的值以,分隔，例如测试`name`是否不等于VM1和VM2：

```
name!?=VM1,VM2
```

9. `~=`: 字符串模糊匹配，相当于MySQL中的`like`操作。使用`%`匹配一个或多个字符，使用`_`匹配一个字符。例如查询一个名字是以`IntelCore`开头的：

```
name~=IntelCore%
```

10. 或者查询一个名字是以`IntelCore`开头，以`7`结尾，中间模糊匹配一个字符：

```
name~=IntelCore_7
```

这样名字是`IntelCoreI7`，`IntelCoreM7`的记录都会匹配上。

11. `!~=`: 模糊匹配非操作。查询一个字段不能模糊匹配到某个字符串，匹配条件与`~=`相同。

12.is null: 字段为null :

```
name is null
```

13.not null: 字段不为null :

```
name not null
```

分页查询

start、**limit**、**replyWithCount**三个字段可以配合使用实现分页查询。

- **start**指定起始查询位置。
- **limit**指定查询返回的最大记录数。
- **replyWithCount**被设置成true后，查询返回中会包含满足查询条件的记录总数，跟**start**值比较就可以得知还需几次分页。

例如：

总共有1000记录满足查询条件，使用如下组合：

```
start=0 limit=100 replyWithCount=true
```

则API返回将包含头100条记录，以及total字段等于1000，表示总共满足条件的记录为1000。

获取资源可查询字段

由于支持的查询条件数非常巨大，我们无法在文档中枚举所有的查询条件。

用户可以使用命令行工具-**cli**的自动补全功能来查看一个资源可查询的字段以及可跨表查询的字段。

- 以查询虚拟机为例，在-**cli**里输入QueryVmInstance并按Tab键补全，可以看到提示页面：

```
- >>>QueryVmInstance
[Query Conditions:]
allVolumes.      cluster.      host.      image.      instanceOffering.
  rootVolume.
  vmNics.        zone.

__systemTag__=   __userTag__=   allocatorStrategy=   clusterUuid=
cpuNum=         cpuSpeed=
createDate=     defaultL3NetworkUuid=   description=   groupBy=   hostUuid
=               hypervisorType=
imageUuid=      instanceOfferingUuid=   lastHostUuid=   lastOpDate=
memorySize=    name=
platform=     rootVolumeUuid=   state=   type=
uuid=         zoneUuid=

[Parameters:]
count=   fields=   limit=   replyWithCount=   sortBy=
sortDirection=
```

```
start=          timeout=
```

- 这里中间行：

```
__systemTag__=    __userTag__=    allocatorStrategy=    clusterUuid=
cpuNum=          cpuSpeed=
createDate=      defaultL3NetworkUuid=    description=          groupBy=          hostUuid
=              hypervisorType=
imageUuid=       instanceOfferingUuid=    lastHostUuid=        lastOpDate=
memorySize=      name=
platform=       rootVolumeUuid=    state=              type=
uuid=           zoneUuid=
```

除__systemTag__和__userTag__两个特殊查询条件外，其余均为虚拟机表的原生字段，用户可以在API的查询条件里面指定它们，并且可以在**fields**参数中指定这些字段来过滤其它不希望API返回的字段。

例如：

```
GET zstack/v1/vm-instances?q=cpuNum>5
```

返回CPU数量多于5的虚拟机。

```
GET zstack/v1/vm-instances?q=hypervisorType=KVM&fields=uuid&fields=name
```

返回虚拟化类型为KVM的虚拟机，由于在**fields**指定了uuid和name两个字段，API返回中只会包含虚拟机的name和uuid。



注：
只有资源的原生字段可以被**fields**选取，__userTag__、__systemTag__以及下面讲到的跨表字段均不能出现在**fields**参数中。

- 提示的第一行：

```
allVolumes.      cluster.      host.      image.      instanceOffering.
rootVolume.
vmNics.          zone.
```

指明了虚拟机资源可以跟哪些资源做跨表查询，例如：**allVolumes**代表云盘，**cluster**代表集群，**vmNics**代表网卡等。

如需查看这些资源的具体字段，只需输入资源名加.号，并按Tab键补全。

例如：

```
- >>>QueryVmInstance vmNics.
[Query Conditions:]
```

```

vmNics.eip.          vmNics.I3Network.      vmNics.loadBalancerListener.  vmNics
.portForwarding.    vmNics.securityGroup.
vmNics.vmInstance.

vmNics.__systemTag__=  vmNics.__userTag__=      vmNics.createDate=
vmNics.deviceId=      vmNics.gateway=
vmNics.ip=            vmNics.I3NetworkUuid=  vmNics.lastOpDate=      vmNics
.mac=                vmNics.metaData=
vmNics.netmask=       vmNics.uuid=          vmNics.vmInstanceUuid=

```

这里我们输入了资源**vmNics**并用.号表示我们要做一个跨表查询，Tab键为我们补全了**vmNics**资源的原生字段以及可跨表查询的其它资源。

- 例如**vmNics.ip**表示网卡的原生字段**ip**：

```
GET zstack/v1/vm-instances?q=vmNics.ip=192.168.0.100
```

进行了一个跨表查询，条件是网卡表的**ip**字段，返回的结果是**ip**为**192.168.0.100**的虚拟机。

- 网卡资源同样可以跟其它资源进行跨表查询，例如**vmNics.eip**。

将网卡表和EIP表进行跨表：

```
GET zstack/v1/vm-instances?q=vmNics.eip.ip=192.168.0.100
```

进行了跨3表查询，返回的是EIP为**192.168.0.100**的虚拟机。

- 通过资源间连续跨表，一个资源几乎跟系统中多个有逻辑关系的资源进行跨表，例如：

```

- >>>QueryVmInstance zone.cluster.I2Network.I3Network.
[Query Conditions:]
zone.cluster.I2Network.I3Network.ipRanges.      zone.cluster.I2Network.I3Network.I2Netwo
rk.        zone.cluster.I2Network.I3Network.networkServices.
zone.cluster.I2Network.I3Network.serviceProvider. zone.cluster.I2Network.I3Network.vmNic.
zone.cluster.I2Network.I3Network.zone.

zone.cluster.I2Network.I3Network.__systemTag__=  zone.cluster.I2Network.I3Network
.__userTag__=  zone.cluster.I2Network.I3Network.createDate=
zone.cluster.I2Network.I3Network.description=  zone.cluster.I2Network.I3Network
.dnsDomain=  zone.cluster.I2Network.I3Network.I2NetworkUuid=
zone.cluster.I2Network.I3Network.lastOpDate=  zone.cluster.I2Network.I3Network.name=
zone.cluster.I2Network.I3Network.state=
zone.cluster.I2Network.I3Network.system=  zone.cluster.I2Network.I3Network.type=
zone.cluster.I2Network.I3Network.uuid=
zone.cluster.I2Network.I3Network.zoneUuid=

```

分别跟**zone**、**cluster**、**I2Network**、**I3Network**多个资源进行跨表。



注:

- 由于一个资源的逻辑关系存在环路，例如以虚拟机为主体可以跟网卡进行跨表，例如QueryVmInstance vmNics.，同时以网卡为查询主体也可以跟虚拟机进行跨表QueryVmNic vmInstance.，这样就会存在环路路径。
- 例如 QueryVmInstance vmNics.vmInstance.name=vm1通过跨表查询了name=vm1的虚拟机，它的实际效果跟QueryVmInstance name=vm1完全等同。这里的跨表是无意义的，只会生产复杂的SQL语句导致低效的数据库查询。在使用中应该避免这种环路跨表查询。
- **__systemTag__**和**__userTag__**是两个特殊的查询条件，允许用户通过tag查询资源。

例如：

```
QueryVmInstance __systemTag__=staticip:10.10.1.20
```

查询具有staticip:10.10.1.20这个tag的虚拟机。

4 网络资源

4.1 二层网络资源相关接口

4.1.1 创建VXLAN网络池(CreateL2VxlanNetworkPool)

API请求

URLs

```
POST zstack/v1/l2-networks/vxlan-pool
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "name": "Test-NetPool",
    "description": "Test",
    "zoneUuid": "ec6cc473d1c345528b92da754cd8fa00"
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"name":"Test-NetPool","description":"Test","zoneUuid":"2de2048daac33068aecac1f5b40d1343"}}' \
http://localhost:8080/zstack/v1/l2-networks/vxlan-pool
```

参数列表

名字	类型	位置	描述	可选值	起始版本
name	String	body(包含在 params 结构中)	资源名称		0.6

名字	类型	位置	描述	可选值	起始版本
description (可选)	String	body(包含在 params 结构中)	资源的详细描述		0.6
zoneUuid	String	body(包含在 params 结构中)	区域UUID		0.6
physicalInterface	String	body(包含在 params 结构中)			0.6
type (可选)	String	body(包含在 params 结构中)			0.6
resourceUuid (可选)	String	body(包含在 params 结构中)			0.6
systemTags (可选)	List	body			0.6
userTags (可选)	List	body			0.6

API返回

返回示例

```
{
  "inventory": {
    "name": "Test-NetPool",
    "description": "Test",
    "zoneUuid": "3f35bd50dab14ababe3d85736b63aba7",
    "type": "L2VxlanNetwork"
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6

名字	类型	描述	起始版本
inventory	L2VxlanNetwork PoolInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
attachedCidrs	Map		0.6
uuid	String	资源的UUID，唯一标识该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
physicalInterface	String		0.6
type	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
attachedClusterUuids	List		0.6
attachedVtepRefs	List	详情参考 attachedVtepRefs	0.6

名字	类型	描述	起始版本
attachedVxlanNetworkRefs	List	详情参考 attachedVxlanNetworkRefs	0.6
attachedVniRanges	List	详情参考 attachedVniRanges	0.6

#attachedVtepRefs

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
hostUuid	String	物理机UUID	0.6
vteplp	String		0.6
port	Integer		0.6
type	String		0.6
poolUuid	String		0.6

#attachedVxlanNetworkRefs

名字	类型	描述	起始版本
vni	Integer		0.6
poolUuid	String		0.6
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
physicalInterface	String		0.6
type	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
attachedClusterUuids	List		0.6

#attachedVniRanges

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startVni	Integer		0.6
endVni	Integer		0.6
l2NetworkUuid	String	二层网络UUID	0.6

SDK示例

Java SDK

```
CreateL2VxlanNetworkPoolAction action = new CreateL2VxlanNetworkPoolAction();
action.name = "Test-NetPool";
action.description = "Test";
action.zoneUuid = "b54ad1e316ea43028b7042018a376aa8";
action.sessionId = "d584d552f0a34d78866f5a7757d7adf0";
CreateL2VxlanNetworkPoolAction.Result res = action.call();
```

Python SDK

```
CreateL2VxlanNetworkPoolAction action = CreateL2VxlanNetworkPoolAction()
action.name = "Test-NetPool"
action.description = "Test"
action.zoneUuid = "081d5f9914f44d8da30dda37448d0c99"
action.sessionId = "2f51609112064da98d1720c28c72843c"
CreateL2VxlanNetworkPoolAction.Result res = action.call()
```

4.1.2 查询VXLAN网络池(QueryL2VxlanNetworkPool)

API请求

URLs

```
GET zstack/v1/l2-networks/vxlan-pool
GET zstack/v1/l2-networks/vxlan-pool/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b2132aefec76492cb00aff0a2d97a202" \
```

```
-X GET http://localhost:8080/zstack/v1/l2-networks/vxlan-pool
```

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 07c093f70f10467e826adc1743a022e6" \
-X GET http://localhost:8080/zstack/v1/l2-networks/vxlan-pool/0576843440c74069b541f
aa856f71d39
```

可查询字段

运行 `zstack-cli` 命令行工具，输入 `QueryL2VxlanNetworkPool` 并按 Tab 键查看所有可查询字段以及可跨表查询的资源名。

API返回

返回示例

```
{
  "inventories": [
    {
      "name": "Test-Net",
      "description": "Test",
      "zoneUuid": "6199e247f6bf4157a41e51ed7ff2f78c",
      "type": "L2VxlanNetworkPool"
    }
  ]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	0.6
inventories	List	详情参考 inventories	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6

名字	类型	描述	起始版本
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventories

名字	类型	描述	起始版本
attachedCidrs	Map		0.6
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
physicalInterface	String		0.6
type	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
attachedClusterUuids	List		0.6
attachedVtepRefs	List	详情参考 attachedVtepRefs	0.6
attachedVxlanNetworkRefs	List	详情参考 attachedVxlanNetworkRefs	0.6
attachedVniRanges	List	详情参考 attachedVniRanges	0.6

#attachedVtepRefs

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
hostUuid	String	物理机UUID	0.6
vtepIp	String		0.6
port	Integer		0.6

名字	类型	描述	起始版本
type	String		0.6
poolUuid	String		0.6

#attachedVxlanNetworkRefs

名字	类型	描述	起始版本
vni	Integer		0.6
poolUuid	String		0.6
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
physicalInterface	String		0.6
type	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
attachedClusterUuids	List		0.6

#attachedVniRanges

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startVni	Integer		0.6
endVni	Integer		0.6
l2NetworkUuid	String	二层网络UUID	0.6

SDK示例

Java SDK

```
QueryL2VxlanNetworkPoolAction action = new QueryL2VxlanNetworkPoolAction();
action.conditions = asList();
action.sessionId = "773c52887a0945c4b5dee525e71f5287";
QueryL2VxlanNetworkPoolAction.Result res = action.call();
```

Python SDK

```
QueryL2VxlanNetworkPoolAction action = QueryL2VxlanNetworkPoolAction()
action.conditions = []
action.sessionId = "0fa3249e9dbd48f3951a93a772882f67"
QueryL2VxlanNetworkPoolAction.Result res = action.call()
```

4.1.3 创建VXLAN网络(CreateL2VxlanNetwork)

API请求

URLs

```
POST zstack/v1/l2-networks/vxlan
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "vni": 10.0,
    "poolUuid": "517752a193074a92807df368892851cc",
    "name": "Test-Net",
    "description": "Test",
    "zoneUuid": "1966127fc98444b1812fde2382e96cfe"
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中systemTags、userTags字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"vni":10.0,"poolUuid":"c27a329718ed3585a27d227174ed1ace"},"name":"Test-Net","description":"Test","zoneUuid":"8dd66847fcf930178acc687cb4c1438c"}' \
```

http://localhost:8080/zstack/v1/l2-networks/vxlan

参数列表

名字	类型	位置	描述	可选值	起始版本
vni (可选)	Integer	body(包含在params结构中)			0.6
poolUuid	String	body(包含在params结构中)	Vxlan网络资源池uuid		0.6
name	String	body(包含在params结构中)	资源名称		0.6
description (可选)	String	body(包含在params结构中)	资源的详细描述		0.6
zoneUuid	String	body(包含在params结构中)	区域UUID		0.6
physicalInterface	String	body(包含在params结构中)			0.6
type (可选)	String	body(包含在params结构中)			0.6
resourceUuid (可选)	String	body(包含在params结构中)			0.6
systemTags (可选)	List	body			0.6
userTags (可选)	List	body			0.6

API返回

返回示例

```
{
  "inventory": {
    "vni": 10.0,
    "poolUuid": "3f84a7f8be86430592a999375b0da922",
    "name": "Test-Net",
    "description": "Test",
    "zoneUuid": "ef5d57db14a447cea80882a9d0b08dec",
    "type": "L2VxlanNetwork"
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventory	L2VxlanNetwork PoolInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
vni	Integer		0.6

名字	类型	描述	起始版本
poolUuid	String	Vxlan网络资源池uuid	0.6
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
physicalInterface	String		0.6
type	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
attachedClusterUuids	List		0.6

SDK示例

Java SDK

```
CreateL2VxlanNetworkAction action = new CreateL2VxlanNetworkAction();
action.vni = 10.0;
action.poolUuid = "94522f4f06ad4741b6c129844f5d9cb9";
action.name = "Test-Net";
action.description = "Test";
action.zoneUuid = "cc13465255d24ede858b4b293ca538a2";
action.sessionId = "c4afaffac3ac4233a56f5d03387675dd";
CreateL2VxlanNetworkAction.Result res = action.call();
```

Python SDK

```
CreateL2VxlanNetworkAction action = CreateL2VxlanNetworkAction()
action.vni = 10.0
action.poolUuid = "38302b5f00b24ae48fd55f22546e7543"
action.name = "Test-Net"
action.description = "Test"
action.zoneUuid = "04f0b5941c8040918412a562269cccbc"
action.sessionId = "53b39e7d9da047e98931dde2e94eae05"
CreateL2VxlanNetworkAction.Result res = action.call()
```

4.1.4 查询VXLAN网络(QueryL2VxlanNetwork)

API请求

URLs

```
GET zstack/v1/l2-networks/vxlan
```

```
GET zstack/v1/l2-networks/vxlan/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth eff53f15d4ad4f17b3dc13bc15d7f0f6" \
-X GET http://localhost:8080/zstack/v1/l2-networks/vxlan
```

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 801edfc2a7f244198ed2f13743c1b232" \
-X GET http://localhost:8080/zstack/v1/l2-networks/vxlan/8dcdd0a8c57a45a5a884887f5a0681c7
```

可查询字段

运行 **zstack-cli** 命令行工具，输入 **QueryL2VxlanNetwork** 并按 Tab 键查看所有可查询字段以及可跨表查询的资源名。

API返回

返回示例

```
{
  "inventories": [
    {
      "name": "Test-Net",
      "description": "Test",
      "zoneUuid": "ef52a71330384e3080c88b243bf3fae2",
      "type": "L2VxlanNetwork"
    }
  ]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	0.6
inventories	List	详情参考 inventories	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventories

名字	类型	描述	起始版本
vni	Integer		0.6
poolUuid	String		0.6
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
physicalInterface	String		0.6
type	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
attachedClusterUuids	List		0.6

SDK示例

Java SDK

```
QueryL2VxlanNetworkAction action = new QueryL2VxlanNetworkAction();
action.conditions = asList();
action.sessionId = "4a008984bc2844e59b6fa8acf188d587";
```

```
QueryL2VxlanNetworkAction.Result res = action.call();
```

Python SDK

```
QueryL2VxlanNetworkAction action = QueryL2VxlanNetworkAction()
action.conditions = []
action.sessionId = "c530e08c8ad24dbeb986647100eff212"
QueryL2VxlanNetworkAction.Result res = action.call()
```

4.1.5 创建普通二层网络(CreateL2NoVlanNetwork)

API请求

URLs

```
POST zstack/v1/l2-networks/no-vlan
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "name": "Test-Net",
    "description": "Test",
    "zoneUuid": "b582ad84692f447d827c91b94cf295e7",
    "physicalInterface": "eth0"
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"name":"Test-Net","description":"Test","zoneUuid":"c23ba5023fcb3b4b8c3e76c443005e3d","physicalInterface":"eth0"}}' \
http://localhost:8080/zstack/v1/l2-networks/no-vlan
```

参数列表

名字	类型	位置	描述	可选值	起始版本
name	String	body(包含在 params 结构中)	普通二层网络名称		0.6
description (可选)	String	body(包含在 params 结构中)	普通二层网络的详细描述		0.6
zoneUuid	String	body(包含在 params 结构中)	区域UUID		0.6
physicalInterface	String	body(包含在 params 结构中)	物理网卡		0.6
type (可选)	String	body(包含在 params 结构中)	二层网络类型		0.6
resourceUuid (可选)	String	body(包含在 params 结构中)	资源UUID。若指定，二层网络会使用该字段值作为UUID		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

该API成功时返回一个空的JSON结构`{}`，出错时返回的JSON结构包含一个error字段，例如：

```
{
  "error": {
    "code": "SYS.1001",
    "description": "A message or a operation timeout",
    "details": "Create VM on KVM timeout after 300s"
  }
}
```

```
}
```

SDK示例

Java SDK

```
CreateL2NoVlanNetworkAction action = new CreateL2NoVlanNetworkAction();
action.name = "Test-Net";
action.description = "Test";
action.zoneUuid = "c8ff026642ed477a9ceac2e014329518";
action.physicalInterface = "eth0";
action.sessionId = "3272547ade654ae2876fa230fa348025";
CreateL2NoVlanNetworkAction.Result res = action.call();
```

Python SDK

```
CreateL2NoVlanNetworkAction action = CreateL2NoVlanNetworkAction()
action.name = "Test-Net"
action.description = "Test"
action.zoneUuid = "003dc8013ef84d9fb0e80be8dba11028"
action.physicalInterface = "eth0"
action.sessionId = "999cbaccd7e140059ee277f795b40469"
CreateL2NoVlanNetworkAction.Result res = action.call()
```

4.1.6 创建二层VLAN网络(CreateL2VlanNetwork)

API请求

URLs

```
POST zstack/v1/l2-networks/vlan
```

Headers

```
OAuth: the-session-uuid
```

Body

```
{
  "params": {
    "vlan": 10.0,
    "name": "Test-Net",
    "description": "Test",
    "zoneUuid": "94ece16f613e43fe9bdc6bfad5e2bb3c",
    "physicalInterface": "eth0"
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"vlan":10.0,"name":"Test-Net","description":"Test","zoneUuid":"43af355e5eae3ab0b74ee76ca9c19005","physicalInterface":"eth0"}}' \
http://localhost:8080/zstack/v1/l2-networks/vlan
```

参数列表

名字	类型	位置	描述	可选值	起始版本
vlan	Integer	body(包含在 params 结构中)			0.6
name	String	body(包含在 params 结构中)	资源名称		0.6
description (可选)	String	body(包含在 params 结构中)	资源的详细描述		0.6
zoneUuid	String	body(包含在 params 结构中)	区域UUID		0.6
physicalInterface	String	body(包含在 params 结构中)			0.6
type (可选)	String	body(包含在 params 结构中)			0.6
resourceUuid (可选)	String	body(包含在 params 结构中)			0.6
systemTags (可选)	List	body			0.6
userTags (可选)	List	body			0.6

API返回

该API成功时返回一个空的JSON结构{}，出错时返回的JSON结构包含一个error字段，例如：

```
{
  "error": {
    "code": "SYS.1001",
    "description": "A message or a operation timeout",
    "details": "Create VM on KVM timeout after 300s"
  }
}
```

SDK示例

Java SDK

```
CreateL2VlanNetworkAction action = new CreateL2VlanNetworkAction();
action.vlan = 10.0;
action.name = "Test-Net";
action.description = "Test";
action.zoneUuid = "5f5272ef22b445e7aee01f051bc6f43c";
action.physicalInterface = "eth0";
action.sessionId = "a51511d455b94fc09b91d793d704f845";
CreateL2VlanNetworkAction.Result res = action.call();
```

Python SDK

```
CreateL2VlanNetworkAction action = CreateL2VlanNetworkAction()
action.vlan = 10.0
action.name = "Test-Net"
action.description = "Test"
action.zoneUuid = "62e4c09ca5bd4878a636a8605b358d43"
action.physicalInterface = "eth0"
action.sessionId = "a3fe3a2b518c4e88ba42407003db7cc7"
CreateL2VlanNetworkAction.Result res = action.call()
```

4.1.7 查询二层VLAN网络(QueryL2VlanNetwork)

API请求

URLs

```
GET zstack/v1/l2-networks/vlan
GET zstack/v1/l2-networks/vlan/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 1456f654cfc46889ff24bd83da741f4" \
```

```
-X GET http://localhost:8080/zstack/v1/l2-networks/vlan
```

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 3ad0ef4e609e43829cde6fcaac60d90d" \
-X GET http://localhost:8080/zstack/v1/l2-networks/vlan/d1eefa849cc84978bacd10ef5e1da12f
```

可查询字段

运行 `zstack-cli` 命令行工具，输入 `QueryL2VlanNetwork` 并按 Tab 键查看所有可查询字段以及可跨表查询的资源名。

API返回

返回示例

```
{
  "inventories": [
    {
      "vlan": 10.0,
      "name": "Test-Net",
      "description": "Test",
      "zoneUuid": "02681829852848e6b25d6a11aa673f0c",
      "physicalInterface": "eth0",
      "type": "L2VlanNetwork"
    }
  ]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventories	List	详情参考 inventories	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6

名字	类型	描述	起始版本
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventories

名字	类型	描述	起始版本
vlan	Integer		0.6
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
physicalInterface	String		0.6
type	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
attachedClusterUuids	List		0.6

SDK示例

Java SDK

```
QueryL2VlanNetworkAction action = new QueryL2VlanNetworkAction();
action.conditions = asList();
action.sessionId = "9ae1447a4dce41f9835c2a89ed7e7eba";
QueryL2VlanNetworkAction.Result res = action.call();
```

Python SDK

```
QueryL2VlanNetworkAction action = QueryL2VlanNetworkAction()
action.conditions = []
action.sessionId = "3e20ec335a82447e9e39c10b371de1b3"
```

```
QueryL2VlanNetworkAction.Result res = action.call()
```

4.1.8 删除二层网络>DeleteL2Network

API请求

URLs

```
DELETE zstack/v1/l2-networks/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth ff3445cd125b4f67b6fc551133852726" \
-X DELETE http://localhost:8080/zstack/v1/l2-networks/254811ac54a04835a8f82871f96ab86f?
参数列表
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID ，唯一标示该资源		0.6
deleteMode (可选)	String	body	删除模式		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可 选)	List	body	用户标签		0.6

API返回

该API成功时返回一个空的JSON结构{}，出错时返回的JSON结构包含一个error字段，例如：

```
{
  "error": {
    "code": "SYS.1001",
    "description": "A message or a operation timeout",
    "details": "Create VM on KVM timeout after 300s"
  }
}
```

```
} /9
```

SDK示例

Java SDK

```
DeleteL2NetworkAction action = new DeleteL2NetworkAction();
action.uuid = "8952ecc352fc49b9a0b22286772e74e2";
action.deleteMode = "Permissive";
action.sessionId = "1bece9d1092e47e7ba176ac2e3b03793";
DeleteL2NetworkAction.Result res = action.call();
```

Python SDK

```
DeleteL2NetworkAction action = DeleteL2NetworkAction()
action.uuid = "f1f6deb480d44b77b53b74784c17a1d9"
action.deleteMode = "Permissive"
action.sessionId = "9b914d6dc4d64c8f8f45452bad140f9f"
DeleteL2NetworkAction.Result res = action.call()
```

4.1.9 查询二层网络(QueryL2Network)

API请求

URLs

```
GET zstack/v1/l2-networks
GET zstack/v1/l2-networks/{uuid}
```

Headers

```
OAuth: the-session-uuid
```

Curl示例

```
curl
-H "Content-Type: application/json" \
-H "OAuth: 9bdaea2355c6417b9e6f3acff94fe731" \
-X GET http://localhost:8080/zstack/v1/l2-networks
```

```
curl
-H "Content-Type: application/json" \
-H "OAuth: 0469a0dfcb7343abb2eb08d876ad3209" \
-X GET http://localhost:8080/zstack/v1/l2-networks/a523215c60b14236869d249176fb328d
```

可查询字段

运行**zstack-cli**命令行工具，输入QueryL2Network并按Tab键查看所有可查询字段以及可跨表查询的资源名。

API返回

API返回返回示例

```
{
  "inventories": [
    {
      "vlan": 10.0,
      "name": "Test-Net",
      "description": "Test",
      "zoneUuid": "a5af85a0db3a4433b58ff429eea1e9a3",
      "physicalInterface": "eth0",
      "type": "L2VlanNetwork"
    }
  ]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventories	List	详情参考 inventories	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventories

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
physicalInterface	String		0.6
type	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
attachedClusterUuids	List		0.6

SDK示例

Java SDK

```
QueryL2NetworkAction action = new QueryL2NetworkAction();
action.conditions = asList();
action.sessionId = "c9c0aa502c3d4bdd8cf480364a17fe67";
QueryL2NetworkAction.Result res = action.call();
```

Python SDK

```
QueryL2NetworkAction action = QueryL2NetworkAction()
action.conditions = []
action.sessionId = "2170549725174f5f9eb390d314f86f8b"
QueryL2NetworkAction.Result res = action.call()
```

4.1.10 更新二层网络(UpdateL2Network)

API请求

URLs

```
PUT zstack/v1/l2-networks/{uuid}/actions
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "updateL2Network": {
```

```
"name": "Test-Net",
"description": "Test"
},
"systemTags": [],
"userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X PUT -d '{"updateL2Network":{"name":"Test-Net","description":"Test"}}' \
http://localhost:8080/zstack/v1/l2-networks/8a44ece25f973842b841591a554f5fee/actions
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID ，唯一标示该资源		0.6
name (可选)	String	body(包含在 updateL2Network 结构中)	普通二层网络名称		0.6
description (可选)	String	body(包含在 updateL2Network 结构中)	普通二层网络的详细描述		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

返回示例

```
{
  "inventory": {
    "vlan": 10.0,
    "name": "Test-Net",
    "description": "Test",
    "zoneUuid": "95bb5b1156ff49caacaf9acc2eda0058",

```

```
"physicalInterface": "eth0",
"type": "L2VlanNetwork"
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventory	L2NetworkInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
physicalInterface	String		0.6
type	String		0.6

名字	类型	描述	起始版本
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
attachedClusterUuids	List		0.6

SDK示例

Java SDK

```
UpdateL2NetworkAction action = new UpdateL2NetworkAction();
action.uuid = "f7ea5335fcab4d438069879d62384b0e";
action.name = "Test-Net";
action.description = "Test";
action.sessionId = "05627ec42486449cb6b422ef927829f1";
UpdateL2NetworkAction.Result res = action.call();
```

Python SDK

```
UpdateL2NetworkAction action = UpdateL2NetworkAction()
action.uuid = "7dc08ad01b8a4d578aab2ae0ad750f1b"
action.name = "Test-Net"
action.description = "Test"
action.sessionId = "130b064e63204a479857ced03f550cf2"
UpdateL2NetworkAction.Result res = action.call()
```

4.1.11 获取二层网络类型(GetL2NetworkTypes)

API请求

URLs

```
GET zstack/v1/l2-networks/types
```

Headers

```
OAuth: the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "OAuth: 76514d7cdcf44b57bc26ff7b4ae0be63" \
-X GET http://localhost:8080/zstack/v1/l2-networks/types
```

参数列表

名字	类型	位置	描述	可选值	起始版本
systemTags (可选)	List	query			0.6
userTags (可选)	List	query			0.6

API返回

返回示例

```
{
  "l2NetworkTypes": [
    "L2VlanNetwork",
    "L2NoVlanNetwork"
  ]
}
```

名字	类型	描述	起始版本
types	List		0.6
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

SDK示例

Java SDK

```
GetL2NetworkTypesAction action = new GetL2NetworkTypesAction();
action.sessionId = "485005865e744e19a12ad2f9af2ea997";
GetL2NetworkTypesAction.Result res = action.call();
```

Python SDK

```
GetL2NetworkTypesAction action = GetL2NetworkTypesAction()
action.sessionId = "b34f3c756aa542789f35bc1774845bc1"
GetL2NetworkTypesAction.Result res = action.call()
```

4.1.12 挂载二层网络到集群(AttachL2NetworkToCluster)

API请求

URLs

```
POST zstack/v1/l2-networks/{l2NetworkUuid}/clusters/{clusterUuid}
```

Headers

```
OAuth: the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST http://localhost:8080/zstack/v1/l2-networks/e852ba6eb4c03f2fafdb73ce838d8980/
clusters/112816423a1a3b50a58d54fd1f6a8665
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l2NetworkUuid	String	url	二层网络UUID		0.6
clusterUuid	String	url	集群UUID		0.6
systemTags (可选)	List	body			0.6
userTags (可 选)	List	body			0.6

API返回

返回示例

```
{
  "inventory": {
    "vlan": 10.0,
    "name": "Test-Net",
    "description": "Test",
    "zoneUuid": "5e0f88c760854e47b9cfdb8fcdcd89c1",
    "physicalInterface": "eth0",
    "type": "L2VlanNetwork"
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventory	L2NetworkInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6

名字	类型	描述	起始版本
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
physicalInterface	String		0.6
type	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
attachedClusterUuids	List		0.6

SDK示例

Java SDK

```
AttachL2NetworkToClusterAction action = new AttachL2NetworkToClusterAction();
action.l2NetworkUuid = "81e7dc92f36a4796bd17575bb3a57564";
action.clusterUuid = "ec6d68c52a0f4da0895d378038e9cff8";
action.sessionId = "1062e8b6e1dd4fb5a869e3fe9d0593a3";
AttachL2NetworkToClusterAction.Result res = action.call();
```

Python SDK

```
AttachL2NetworkToClusterAction action = AttachL2NetworkToClusterAction()
action.l2NetworkUuid = "005556aab00c470781709712bd158d25"
action.clusterUuid = "a1a505d9e9094dc1b47d158b64e19c31"
action.sessionId = "061bcb396e584c06b8848b2d4f9e516a"
AttachL2NetworkToClusterAction.Result res = action.call()
```

4.1.13 从集群上卸载二层网络(DetachL2NetworkFromCluster)

API请求

URLs

```
DELETE/v1/l2-networks/{l2NetworkUuid}/clusters/{clusterUuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 7e281cfe40094b6e8dee0c449d58b027" \
```

```
-X DELETE http://localhost:8080/zstack/v1/l2-networks/51ad692ac9cf442bbdb9fede20cd3c32/
clusters/8a1c4126279a4478b281c6d2e5cb9766?
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l2NetworkUuid	String	url	二层网络UUID		0.6
clusterUuid	String	url	集群UUID		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可 选)	List	body	用户标签		0.6

API返回

返回示例

```
{
  "inventory": {
    "vlan": 10.0,
    "name": "Test-Net",
    "description": "Test",
    "zoneUuid": "13f35f160b6f404ab29ccaa1d772a69c",
    "physicalInterface": "eth0",
    "type": "L2VlanNetwork"
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventory	L2NetworkInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6

名字	类型	描述	起始版本
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
physicalInterface	String		0.6
type	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
attachedClusterUuids	List		0.6

SDK示例

Java SDK

```
DetachL2NetworkFromClusterAction action = new DetachL2NetworkFromClusterAction();
action.l2NetworkUuid = "5ce4c9b306a646a89d720fecc3907974";
action.clusterUuid = "67d24ddc7a074fcf994a14a905333834";
action.sessionId = "a308fc78ddba4fa5b4b70b169b4caf93";
DetachL2NetworkFromClusterAction.Result res = action.call();
```

Python SDK

```
DetachL2NetworkFromClusterAction action = DetachL2NetworkFromClusterAction()
action.l2NetworkUuid = "c6144d30e2e34a978720b71d4a7916bb"
action.clusterUuid = "6a90edf14fa74397bc84f49643d7e916"
action.sessionId = "4673052cb35f4eef83135c95630cfbb9"
```

```
DetachL2NetworkFromClusterAction.Result res = action.call()
```

4.1.14 创建Vni Range(CreateVniRange)

API请求

URLs

```
POST zstack/v1/l2-networks/vxlan-pool/{l2NetworkUuid}/vni-ranges
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "name": "TestVniRange",
    "description": "Here is a Vni Range",
    "startVni": 10.0,
    "endVni": 5000.0
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"name":"TestVniRange","description":"Here is a Vni Range","startVni":10.0,"endVni":5000.0}}' \
http://localhost:8080/zstack/v1/l2-networks/vxlan-pool/66b7ddcebeea33eaaa39a2b4667267ab/vni-ranges
```

参数列表

名字	类型	位置	描述	可选值	起始版本
name	String	body(包含在 params 结构中)	资源名称		0.6
description (可选)	String	body(包含在 params 结构中)	资源的详细描述		0.6

名字	类型	位置	描述	可选值	起始版本
startVni	Integer	body(包含在 params 结构中)			0.6
endVni	Integer	body(包含在 params 结构中)			0.6
l2NetworkUuid	String	url	二层Vxlan网络资源池uuid		0.6
resourceUuid (可选)	String	body(包含在 params 结构中)			0.6
systemTags (可选)	List	body			0.6
userTags (可选)	List	body			0.6

API返回

返回示例

```
{
  "inventory": {
    "name": "TestVniRange",
    "description": "Here is a Vni Range",
    "startVni": 10.0,
    "endVni": 5000.0,
    "l2NetworkUuid": "252d938da43a48c98ce93fe7163f14df"
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	0.6
inventory	VniRangeInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startVni	Integer		0.6
endVni	Integer		0.6
I2NetworkUuid	String	二层网络UUID	0.6

SDK示例

Java SDK

```

CreateVniRangeAction action = new CreateVniRangeAction();
action.name = "TestVniRange";
action.description = "Here is a Vni Range";
action.startVni = 10.0;
action.endVni = 5000.0;
action.I2NetworkUuid = "9548ed5fc8434782890d5c6ae24d87a7";
action.sessionId = "977eac0795364c5995c20120e9e35ca1";
CreateVniRangeAction.Result res = action.call();

```

Python SDK

```

CreateVniRangeAction action = CreateVniRangeAction()
action.name = "TestVniRange"
action.description = "Here is a Vni Range"

```

```

action.startVni = 10.0
action.endVni = 5000.0
action.l2NetworkUuid = "d55172d6e90e403da46074d53c680838"
action.sessionId = "dfacd10f07444a99bca68426cc19f079"
CreateVniRangeAction.Result res = action.call()

```

4.1.15 查询Vni Range(QueryVniRange)

API请求

URLs

```

GET zstack/v1/l2-networks/vxlan-pool/vni-range
GET zstack/v1/l2-networks/vxlan-pool/vni-range/{uuid}

```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```

curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 337c51002a0f4fec9b9ced4cdd9ca08d" \
-X GET http://localhost:8080/zstack/v1/l2-networks/vxlan-pool/vni-range

```

```

curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 69245fbe344f4adf95454deb1bfe74fb" \
-X GET http://localhost:8080/zstack/v1/l2-networks/vxlan-pool/vni-range/4397921aaa5d45c7ba9bbde764b89389

```

可查询字段

运行 **zstack-cli** 命令行工具，输入 **QueryVniRange** 并按 Tab 键查看所有可查询字段以及可跨表查询的资源名。

API返回

返回示例

```

{
  "inventories": [
    {
      "name": "Test-Range",
      "description": "Test",
      "startVni": 10.0,
      "endVni": 10000.0,
      "l2NetworkUuid": "32e0539245504a22a474ea2f34db16d0"
    }
  ]
}

```

}

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventories	List	详情参考 inventories	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventories

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startVni	Integer		0.6
endVni	Integer		0.6
l2NetworkUuid	String	二层网络UUID	0.6

SDK示例

Java SDK

```
QueryVniRangeAction action = new QueryVniRangeAction();
action.conditions = asList();
action.sessionId = "eb1c573c6d4240ecb746321b12f9e1d3";
QueryVniRangeAction.Result res = action.call();
```

Python SDK

```
QueryVniRangeAction action = QueryVniRangeAction()
action.conditions = []
action.sessionId = "89f7a520087049dcbd87f91aa20994fa"
QueryVniRangeAction.Result res = action.call()
```

4.1.16 删除Vni Range>DeleteVniRange)

API请求

URLs

```
DELETE/v1/l2-networks/vxlan-pool/vni-ranges/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 49bf51476d4f49e3b6322688a4c02027" \
-X DELETE http://localhost:8080/zstack/v1/l2-networks/vxlan-pool/vni-ranges/57aae81f5ed248e3bea4509843ab919f?
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID ，唯一标示该资源		0.6
deleteMode (可选)	String	body			0.6
systemTags (可选)	List	body			0.6
userTags (可选)	List	body			0.6

API返回

该API成功时返回一个空的JSON结构{}，出错时返回的JSON结构包含一个error字段，例如：

```
{
  "error": {
    "code": "SYS.1001",
    "description": "A message or a operation timeout",
    "details": "Create VM on KVM timeout after 300s"
  }
}
```

SDK示例

Java SDK

```
DeleteVniRangeAction action = new DeleteVniRangeAction();
action.uuid = "8ad0e38ae63d4f2d86b9cd2283c5df03";
action.deleteMode = "Permissive";
action.sessionId = "85100440f4f94c68ba0884e09c7429dc";
DeleteVniRangeAction.Result res = action.call();
```

Python SDK

```
DeleteVniRangeAction action = DeleteVniRangeAction()
action.uuid = "4aef50ec1f784e66ab79ef021216389d"
action.deleteMode = "Permissive"
action.sessionId = "dfbae4e4897f497a9803a2a5dc23f848"
DeleteVniRangeAction.Result res = action.call()
```

4.2 三层网络相关接口

4.2.1 创建三层网络(CreateL3Network)

API请求

URLs

```
POST zstack/v1/l3-networks
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "name": "Test-L3Network",
    "type": "L3BasicNetwork",
    "l2NetworkUuid": "f29d1122065540c9a495e70893f813b9",
    "category": "Private",
    "system": false
  }
}
```

```

    },
    "systemTags": [],
    "userTags": []
  }

```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```

curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"name":"Test-L3Network","type":"L3BasicNetwork","l2NetworkUuid":"bb059728e72f3ff1b3ae2fc9caba3dce","category":"Private","system":false}}' \
http://localhost:8080/zstack/v1/l3-networks

```

参数列表

名字	类型	位置	描述	可选值	起始版本
name	String	body(包含在 params 结构中)	三层网络的名称		0.6
description (可选)	String	body(包含在 params 结构中)	三层网络的详细描述		0.6
type (可选)	String	body(包含在 params 结构中)	三层网络类型		0.6
l2NetworkUuid	String	body(包含在 params 结构中)	二层网络UUID		0.6
system (可选)	boolean	body(包含在 params 结构中)	是否用于系统云主机		0.6
dnsDomain (可选)	String	body(包含在 params 结构中)	DNS域		0.6
resourceUuid (可选)	String	body(包含在 params 结构中)	资源UUID。若指定，三层网络会使用该字段值作为UUID		0.6

名字	类型	位置	描述	可选值	起始版本
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6
category (可选)	String	body(包含在params结构中)	网络类型，需要与system标签搭配使用，system为true时可设置为Public、Private	- Public - Private - System	2.2

API返回

返回示例

```
{
  "inventory": {
    "name": "Test-L3Network",
    "l2NetworkUuid": "1bfca1892b674df28d26d9a3c0f609f9",
    "category": "Private"
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventory	L3NetworkInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6

名字	类型	描述	起始版本
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
type	String	三层网络类型	0.6
zoneUuid	String	区域UUID	0.6
l2NetworkUuid	String	二层网络UUID	0.6
state	String	三层网络的可用状态	0.6
dnsDomain	String	DNS域	0.6
system	Boolean	是否用于系统云主机	0.6
category	String	网络类型，需要与system标签搭配使用，system为true时可设置为Public、Private	0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
dns	List	域名解析服务器	0.6
ipRanges	List	详情参考 ipRanges	0.6
networkServices	List	详情参考 networkServices	0.6

#ipRanges

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
I3NetworkUuid	String	三层网络UUID	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startIp	String		0.6
endIp	String		0.6
netmask	String		0.6
gateway	String		0.6
networkCidr	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

#networkServices

名字	类型	描述	起始版本
I3NetworkUuid	String	三层网络UUID	0.6
networkServiceProviderUuid	String	网络服务提供模块UUID	0.6
networkServiceType	String		0.6

SDK示例

Java SDK

```
AddIpRangeAction action = new AddIpRangeAction();
action.I3NetworkUuid = "471f0bcd100a40bda0a2366106f97a2c";
action.name = "Test-IP-Range";
action.startIp = "192.168.100.10";
action.endIp = "192.168.100.250";
action.netmask = "255.255.255.0";
action.gateway = "192.168.100.1";
action.sessionId = "2a858b6bfa6542d8be18df36fe6d1603";
AddIpRangeAction.Result res = action.call();
```

Python SDK

```
AddIpRangeAction action = AddIpRangeAction()
action.I3NetworkUuid = "7529cacdba2e463783b0345842770b3c"
```

```

action.name = "Test-IP-Range"
action.startIp = "192.168.100.10"
action.endIp = "192.168.100.250"
action.netmask = "255.255.255.0"
action.gateway = "192.168.100.1"
action.sessionId = "8643a901b9ef4a6fbbfc599c62924f41"
AddIpRangeAction.Result res = action.call()

```

4.2.2 删除三层网络>DeleteL3Network)

API请求

URLs

```
DELETE zstack/v1/l3-networks/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```

curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 9e78fa4cfc8a49baadb763cd62ac6a3f" \
-X DELETE http://localhost:8080/zstack/v1/l3-networks/76490f9cd26d4c8a9501cc5a5312aa99?

```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID，唯一标示该资源		0.6
deleteMode (可选)	String	body	删除模式		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

该API成功时返回一个空的JSON结构{}，出错时返回的JSON结构包含一个error字段，例如：

```

{
  "error": {
    "code": "SYS.1001",
    "description": "A message or a operation timeout",
    "details": "Create VM on KVM timeout after 300s"
  }
}

```

```
}
```

SDK示例

Java SDK

```
DeleteL3NetworkAction action = new DeleteL3NetworkAction();
action.uuid = "c19ee03770c6486b85abc58ea10f017a";
action.deleteMode = "Permissive";
action.sessionId = "b8b93037f8064e219bdcbe9dd9a9f366";
DeleteL3NetworkAction.Result res = action.call();
```

Python SDK

```
DeleteL3NetworkAction action = DeleteL3NetworkAction()
action.uuid = "8c4a39676be64072b20e7b7dc4b0ccc7"
action.deleteMode = "Permissive"
action.sessionId = "e0e4140009fc4b95af7197bffc64f7c0"
DeleteL3NetworkAction.Result res = action.call()
```

4.2.3 查询三层网络(QueryL3Network)

API请求

URLs

```
GET zstack/v1/l3-networks
GET zstack/v1/l3-networks/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 98858a4b613f46a4815c34c1c930b34e" \
-X GET http://localhost:8080/zstack/v1/l3-networks
```



注:

例如：查询三层私有网络

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 98858a4b613f46a4815c34c1c930b34e" \
-X GET http://localhost:8080/zstack/v1/l3-networks?q=category=Private
```

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth c1148725103644bc8e2879d72fa47389" \
-X GET http://localhost:8080/zstack/v1/l3-networks/9d119c003bb54d4781443bbc7febcd7f
```

可查询字段

运行`zstack-cli`命令行工具，输入`QueryL3Network`并按Tab键查看所有可查询字段以及可跨表查询的资源名。

API返回

返回示例

```
{
  "inventories": [
    {
      "name": "Test-L3Network",
      "l2NetworkUuid": "8bcf5b824378406494d81ffdae56bde3"
    }
  ]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventories	List	详情参考 inventories	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventories

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
type	String	三层网络类型	0.6
zoneUuid	String	区域UUID	0.6
l2NetworkUuid	String	二层网络UUID	0.6
state	String	三层网络的可用状态	0.6
dnsDomain	String	DNS域	0.6
system	Boolean	是否用于系统云主机	0.6
category	String	网络类型，需要与system标签搭配使用，system为true时可设置为Public、Private	0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
dns	List	域名解析服务器	0.6
ipRanges	List	详情参考 ipRanges	0.6
networkServices	List	详情参考 networkServices	0.6

#ipRanges

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
l3NetworkUuid	String	三层网络UUID	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startIp	String		0.6
endIp	String		0.6
netmask	String		0.6

名字	类型	描述	起始版本
gateway	String		0.6
networkCidr	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

#networkServices

名字	类型	描述	起始版本
l3NetworkUuid	String	三层网络UUID	0.6
networkService ProviderUuid	String	网络服务提供模 块UUID	0.6
networkServiceType	String		0.6

SDK示例

Java SDK

```
QueryL3NetworkAction action = new QueryL3NetworkAction();
action.conditions = asList();
action.sessionId = "58f4bcc9b20f46bfaba379ca0c739535";
QueryL3NetworkAction.Result res = action.call();
```

Python SDK

```
QueryL3NetworkAction action = QueryL3NetworkAction()
action.conditions = []
action.sessionId = "2f99e245b0f04712941da401ecab7173"
```

```
QueryL3NetworkAction.Result res = action.call()
```

4.2.4 更新三层网络(UpdateL3Network)

API请求

URLs

```
PUT zstack/v1/l3-networks/{uuid}/actions
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "updateL3Network": {
    "name": "Test-L3Network"
  },
  "systemTags": [],
  "userTags": []
}
```

上述示例中`systemTags`、`userTags`字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X PUT -d '{"updateL3Network":{"name":"Test-L3Network"}}' \
http://localhost:8080/zstack/v1/l3-networks/a1a3a4bea5e73628b591081fde9f019a/actions
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID ，唯一标示该资源		0.6
name (可选)	String	body(包含在 updateL3Network 结构中)	三层网络的名称		0.6
description (可选)	String	body(包含在 updateL3Network 结构中)	三层网络的详细描述		0.6

名字	类型	位置	描述	可选值	起始版本
system (可选)	Boolean	body(包含在 updateL3Network 结构中)	是否用于系统云主机		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6
category (可选)	String	body(包含在 updateL3Network 结构中)	网络类型，需要与system标签搭配使用，system为true时可设置为Public、Private	- Public - Private - System	2.2

API返回

返回示例

```
{
  "inventory": {
    "uuid": "dd370da0d6fa49288fc119b35db36b7c",
    "name": "Test-L3Network"
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	0.6
inventory	L3NetworkInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
type	String		0.6
zoneUuid	String	区域UUID	0.6
l2NetworkUuid	String	二层网络UUID	0.6
state	String		0.6
dnsDomain	String		0.6
system	Boolean		0.6
category	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
dns	List		0.6
ipRanges	List	详情参考 ipRanges	0.6
networkServices	List	详情参考 networkServices	0.6

#ipRanges

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
l3NetworkUuid	String	三层网络UUID	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startIp	String		0.6
endIp	String		0.6
netmask	String		0.6
gateway	String		0.6
networkCidr	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

#networkServices

名字	类型	描述	起始版本
l3NetworkUuid	String	三层网络UUID	0.6
networkServiceProviderUuid	String	网络服务提供模块UUID	0.6
networkServiceType	String		0.6

SDK示例

Java SDK

```
UpdateL3NetworkAction action = new UpdateL3NetworkAction();
action.uuid = "88d2a1d687974c0ea0e102c276739e39";
action.name = "Test-L3Network";
action.sessionId = "dae0914beced4293bd336fe443ed0d2b";
UpdateL3NetworkAction.Result res = action.call();
```

Python SDK

```
UpdateL3NetworkAction action = UpdateL3NetworkAction()
action.uuid = "3210b12187464e35bfdc50d8d4e2d1d0"
action.name = "Test-L3Network"
action.sessionId = "9e0b1d4240ef4421804612b01c98d7b8"
```

```
UpdateL3NetworkAction.Result res = action.call()
```

4.2.5 获取三层网络类型(GetL3NetworkTypes)

API请求

URLs

```
GET zstack/v1/l3-networks/types
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 3a142b201e9342abbb5864a0ea48992e" \
-X GET http://localhost:8080/zstack/v1/l3-networks/types
```

参数列表

名字	类型	位置	描述	可选值	起始版本
systemTags (可选)	List	query	系统标签		0.6
userTags (可选)	List	query	用户标签		0.6

API返回

返回示例

```
{
  "l3NetworkTypes": [
    "L3BasicNetwork"
  ]
}
```

名字	类型	描述	起始版本
types	List		0.6
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

SDK示例

Java SDK

```
GetL3NetworkTypesAction action = new GetL3NetworkTypesAction();
action.sessionId = "9105437ebb27461fbae36ecb9336ceba";
GetL3NetworkTypesAction.Result res = action.call();
```

Python SDK

```
GetL3NetworkTypesAction action = GetL3NetworkTypesAction()
action.sessionId = "a69bae9c3cad4209b042956b23c70af7"
GetL3NetworkTypesAction.Result res = action.call()
```

4.2.6 改变三层网络状态(ChangeL3NetworkState)

API请求

URLs

```
PUT zstack/v1/l3-networks/{uuid}/actions
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "changeL3NetworkState": {
    "stateEvent": "enable"
  }
}
```

```

},
"systemTags": [],
"userTags": []
}

```

上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```

curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X PUT -d '{"changeL3NetworkState":{"stateEvent":"enable"}}' \
http://localhost:8080/zstack/v1/l3-networks/a8744f1ba60c33f3a2c912c077f41fe5/actions

```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID，唯一标示该资源		0.6
stateEvent	String	body(包含在 changeL3NetworkState 结构中)	状态，可选 enable 与 disable	- enable - disable	0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

返回示例

```

{
  "inventory": {
    "name": "Test-L3Network",
    "l2NetworkUuid": "dfae4bce43694f4d9fc0ba5cdf4e38bd"
  }
}

```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6

名字	类型	描述	起始版本
inventory	L3NetworkInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
type	String		0.6
zoneUuid	String	区域UUID	0.6
l2NetworkUuid	String	二层网络UUID	0.6
state	String		0.6
dnsDomain	String		0.6
system	Boolean		0.6
category	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

名字	类型	描述	起始版本
dns	List		0.6
ipRanges	List	详情参考 ipRanges	0.6
networkServices	List	详情参考 networkServices	0.6

#ipRange

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
l3NetworkUuid	String	三层网络UUID	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startIp	String		0.6
endIp	String		0.6
netmask	String		0.6
gateway	String		0.6
networkCidr	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

#networkServices

名字	类型	描述	起始版本
l3NetworkUuid	String	三层网络UUID	0.6
networkServiceProviderUuid	String	网络服务提供模块UUID	0.6
networkServiceType	String		0.6

SDK示例

Java SDK

```
ChangeL3NetworkStateAction action = new ChangeL3NetworkStateAction();
action.uuid = "eb0849bf0e57400da650a09f2e852260";
```

```
action.stateEvent = "enable";
action.sessionId = "be8ec965ebd94169ba713c42dc079af9";
ChangeL3NetworkStateAction.Result res = action.call();
```

Python SDK

```
ChangeL3NetworkStateAction action = ChangeL3NetworkStateAction()
action.uuid = "d8a1565df89748a2a21ef87b9c9f5be7"
action.stateEvent = "enable"
action.sessionId = "7b8417b9ee3b4df8a21acc8e7c41a7f9"
ChangeL3NetworkStateAction.Result res = action.call()
```

4.2.7 获取网络DHCP服务所用地址(GetL3NetworkDhcpIpAddress)

API请求

URLs

```
GET zstack/v1/l3-networks/{l3NetworkUuid}/dhcp-ip
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth c2c2525cafb64e95bf5210a3549b6895" \
-X GET http://localhost:8080/zstack/v1/l3-networks/82cae6542d3b3e7ea50dab8681f3785c/
dhcp-ip
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l3NetworkUuid	String	url	三层网络UUID		0.6
systemTags (可选)	List	query	系统标签		0.6
userTags (可选)	List	query	用户标签		0.6

API返回

返回示例

```
{
  "ip": "192.168.100.3"
```

}

名字	类型	描述	起始版本
ip	String	IP地址	0.6
success	boolean	成功	0.6
error	ErrorCode	错误码，若不 为null，则表示操作失 败，操作成功时该字段 为null。 详情参考 error	0.6

error

名字	类型	描述	起始版本
code	String	错误码号，错误的全 局唯一标识，例如SYS .1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错 误的源错误，若无原错 误，该字段为null	0.6

SDK示例

Java SDK

```
GetL3NetworkDhcpIpAddressAction action = new GetL3NetworkDhcpIpAddressAction();
action.l3NetworkUuid = "1c311c313859434f8586d4288a31952e";
action.sessionId = "dd1a8993de0e41e18d8718ea2e2d8259";
GetL3NetworkDhcpIpAddressAction.Result res = action.call();
```

Python SDK

```
GetL3NetworkDhcpIpAddressAction action = GetL3NetworkDhcpIpAddressAction()
action.l3NetworkUuid = "4970a85c53a24e35a55f44830058e719"
action.sessionId = "50fd78e250d94c5d944a5fa1a5d6bde0"
```

```
GetL3NetworkDhcpIpAddressAction.Result res = action.call()
```

4.2.8 从三层网络移除DNS(RemoveDnsFromL3Network)

API请求

URLs

```
DELETE/v1/l3-networks/{l3NetworkUuid}/dns/{dns}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 209fa412351946a6bef267d423f4e53c" \
-X DELETE http://localhost:8080/zstack/v1/l3-networks/d96f7d4c0506435a9a1aa7394976c0b2/dns/8.8.4.4?
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l3NetworkUuid	String	url	三层网络UUID		0.6
dns	String	url	DNS地址		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可 选)	List	body	用户标签		0.6

API返回

返回示例

```
{
  "inventory": {
    "name": "Test-L3Network",
    "l2NetworkUuid": "535aea9c1be24fb9b172fa1955341685",
    "dns": []
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失	0.6

名字	类型	描述	起始版本
		败, 操作成功时该字段为null。 详情参考 error	
inventory	L3NetworkInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号, 错误的全局唯一标识, 例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段, 默认为null	0.6
opaque	LinkedHashMap	保留字段, 默认为null	0.6
cause	ErrorCode	根错误, 引发当前错误的源错误, 若无原错误, 该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID, 唯一标识该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
type	String		0.6
zoneUuid	String	区域UUID	0.6
l2NetworkUuid	String	二层网络UUID	0.6
state	String		0.6
dnsDomain	String		0.6
system	Boolean		0.6
category	String		0.6
createDate	Timestamp	创建时间	0.6

名字	类型	描述	起始版本
lastOpDate	Timestamp	最后一次修改时间	0.6
dns	List		0.6
ipRanges	List	详情参考 ipRanges	0.6
networkServices	List	详情参考 networkServices	0.6

#ipRanges

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
I3NetworkUuid	String	三层网络UUID	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startIp	String		0.6
endIp	String		0.6
netmask	String		0.6
gateway	String		0.6
networkCidr	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

#networkServices

名字	类型	描述	起始版本
I3NetworkUuid	String	三层网络UUID	0.6
networkServiceProviderUuid	String	网络服务提供模块UUID	0.6
networkServiceType	String		0.6

SDK示例

Java SDK

```
RemoveDnsFromL3NetworkAction action = new RemoveDnsFromL3NetworkAction();
action.l3NetworkUuid = "3e9227b9ecbd42bba0f3a4620f5655d2";
action.dns = "8.8.4.4";
action.sessionId = "f3e42d2a37ed4c42a6ce7d192af94e05";
RemoveDnsFromL3NetworkAction.Result res = action.call();
```

Python SDK

```
RemoveDnsFromL3NetworkAction action = RemoveDnsFromL3NetworkAction()
action.l3NetworkUuid = "4046beb7d6164cb3843fb5fac2f86ac3"
action.dns = "8.8.4.4"
action.sessionId = "e9fd57376378422e9b58ada9be59185c"
RemoveDnsFromL3NetworkAction.Result res = action.call()
```

4.2.9 向三层网络添加DNS(AddDnsToL3Network)

API请求

URLs

```
POST zstack/v1/l3-networks/{l3NetworkUuid}/dns
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "dns": "8.8.8.8"
  },
  "systemTags": [],
  "userTags": []
}
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"dns":"8.8.8.8"}}' \
http://localhost:8080/zstack/v1/l3-networks/0b8da8c298ed3960925abbcc4f913da8/dns
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l3NetworkUuid	String	url	三层网络UUID		0.6

名字	类型	位置	描述	可选值	起始版本
dns	String	url	DNS地址		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

返回示例

```
{
  "inventory": {
    "name": "Test-L3Network",
    "l2NetworkUuid": "08f7b0b577a84c28968ddce5a9377c5d",
    "dns": [
      "8.8.8.8"
    ]
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	0.6
inventory	L3NetworkInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6

名字	类型	描述	起始版本
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
type	String		0.6
zoneUuid	String	区域UUID	0.6
l2NetworkUuid	String	二层网络UUID	0.6
state	String		0.6
dnsDomain	String		0.6
system	Boolean		0.6
category	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
dns	List		0.6
ipRanges	List	详情参考 ipRanges	0.6
networkServices	List	详情参考 networkServices	0.6

#ipRanges

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
l3NetworkUuid	String	三层网络UUID	0.6
name	String	资源名称	0.6

名字	类型	描述	起始版本
description	String	资源的详细描述	0.6
startIp	String		0.6
endIp	String		0.6
netmask	String		0.6
gateway	String		0.6
networkCidr	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

#networkServices

名字	类型	描述	起始版本
I3NetworkUuid	String	三层网络UUID	0.6
networkService ProviderUuid	String	网络服务提供模 块UUID	0.6
networkServiceType	String		0.6

SDK示例

Java SDK

```
AddDnsToL3NetworkAction action = new AddDnsToL3NetworkAction();
action.I3NetworkUuid = "80bfafb9c4534a2593fcda534ee6414f";
action.dns = "8.8.8.8";
action.sessionId = "0a24af3791134b81bba960844b3aa899";
AddDnsToL3NetworkAction.Result res = action.call();
```

Python SDK

```
AddDnsToL3NetworkAction action = AddDnsToL3NetworkAction()
action.I3NetworkUuid = "17ecd7a47dc34f1ba01df5180480e264"
action.dns = "8.8.8.8"
action.sessionId = "30e00bf1296244cd98c93dec6182ab72"
```

```
AddDnsToL3NetworkAction.Result res = action.call()
```

4.2.10 获取空闲IP(GetFreelp)

API请求

URLs

```
GET zstack/v1/l3-networks/{l3NetworkUuid}/ip/free
GET zstack/v1/l3-networks/ip-ranges/{ipRangeUuid}/ip/free
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 75eb7b9fa3b44161bd57ec5329df4a51" \
-X GET http://localhost:8080/zstack/v1/l3-networks/d5690be5ea634f17b2cc654890cdf254/ip\
/free?l3NetworkUuid=d5690be5ea634f17b2cc654890cdf254&ipRangeUuid=6015f753b00841
0c9fb39344eb31f9e8&start=0.0.0.0&limit=100.0
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l3NetworkUuid (可选)	String	query	三层网络UUID  注: l3NetworkUuid 与ipRange Uuid至少一个 必选		0.6
ipRangeUuid (可选)	String	query	IP段UUID  注: l3NetworkUuid 与ipRange Uuid至少一个 必选		0.6
start (可选)	String	query	起始值		0.6

名字	类型	位置	描述	可选值	起始版本
limit (可选)	int	query	数量限制		0.6
systemTags (可选)	List	query	系统标签		0.6
userTags (可选)	List	query	用户标签		0.6

API返回

返回示例

```
{
  "inventories": [
    {
      "ipRangeUuid": "662d600a4b644f98a0c49df2e5430d69",
      "ip": "10.20.10.5",
      "netmask": "255.255.255.0",
      "gateway": "10.20.10.1"
    },
    {
      "ipRangeUuid": "662d600a4b644f98a0c49df2e5430d69",
      "ip": "10.20.10.6",
      "netmask": "255.255.255.0",
      "gateway": "10.20.10.1"
    },
    {
      "ipRangeUuid": "662d600a4b644f98a0c49df2e5430d69",
      "ip": "10.20.10.10",
      "netmask": "255.255.255.0",
      "gateway": "10.20.10.1"
    }
  ]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventory	L3NetworkInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
ipRangeUuid	String	IP段UUID	0.6
ip	String		0.6
netmask	String		0.6
gateway	String		0.6

SDK示例

Java SDK

```
GetFreelpAction action = new GetFreelpAction();
action.l3NetworkUuid = "80ffbdcfb3f447abb33400614e08d47a";
action.ipRangeUuid = "26a2dc70d5854a8daa7343b2a01bb0a6";
action.start = "0.0.0.0";
action.limit = 100.0;
action.sessionId = "a5fbab39bc1849988b9e421a4678d9ab";
GetFreelpAction.Result res = action.call();
```

Python SDK

```
GetFreelpAction action = GetFreelpAction()
action.l3NetworkUuid = "f722ea2e93f24a1cb020ce0aa714f75c"
action.ipRangeUuid = "3a21991e6b0744d095bd6aa145c23c7b"
action.start = "0.0.0.0"
action.limit = 100.0
action.sessionId = "4bec31f898944daeb3b358232e757ae6"
```

```
GetFreelpAction.Result res = action.call()
```

4.2.11 检查IP可用性(CheckIpAvailability)

API请求

URLs

```
GET zstack/v1/l3-networks/{l3NetworkUuid}/ip/{ip}/availability
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 642bc388bc3040ce9ed0c477bd6b3874" \
-X GET http://localhost:8080/zstack/v1/l3-networks/42ab1436d38245a78a9995e732aa5aad/ip/192.168.10.100/availability?
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l3NetworkUuid	String	url	三层网络UUID		0.6
ip	String	url	IP地址		0.6
systemTags (可选)	List	query	系统标签		0.6
userTags (可 选)	List	query	用户标签		0.6

API返回

返回示例

```
{
  "available": true
}
```

名字	类型	描述	起始版本
available	boolean	可用	0.6
success	boolean	成功	0.6
error	ErrorCode	错误码，若不 为null，则表示操作失	0.6

名字	类型	描述	起始版本
		败, 操作成功时该字段为null。 详情参考 error	

#error

名字	类型	描述	起始版本
code	String	错误码号, 错误的全局唯一标识, 例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段, 默认为null	0.6
opaque	LinkedHashMap	保留字段, 默认为null	0.6
cause	ErrorCode	根错误, 引发当前错误的源错误, 若无原错误, 该字段为null	0.6

SDK示例

Java SDK

```

CheckIpAvailabilityAction action = new CheckIpAvailabilityAction();
action.l3NetworkUuid = "8cb0703274b44a389ffc2f24935235fe";
action.ip = "192.168.10.100";
action.sessionId = "399dfce0d08f460c92a2c2660d80b187";
CheckIpAvailabilityAction.Result res = action.call();

```

Python SDK

```

CheckIpAvailabilityAction action = CheckIpAvailabilityAction()
action.l3NetworkUuid = "19ff4c2f7aab415cae3a6abb8512d05d"
action.ip = "192.168.10.100"
action.sessionId = "c872c583479743009ba56a65d2c7dd22"

```

```
CheckIpAvailabilityAction.Result res = action.call()
```

4.2.12 获取IP网络地址容量(GetIpAddressCapacity)

API请求

URLs

```
GET zstack/v1/ip-capacity
```

Headers

```
Authorization: OAuth the-session-uuid s
```

Curl示例

```
curl -H "Content-Type: application/json" \  
-H "Authorization: OAuth 6ebda54b0c594e68aa8b1e6b97529e88" \  
-X GET http://localhost:8080/zstack/v1/ip-capacity?ipRangeUids=8d45c343b1314986aa3d413b6dc4c013&all=false
```

参数列表

名字	类型	位置	描述	起始版本	
zoneUids (可选)	List	query	区域UUID		0.6
l3NetworkUids (可选)	List	query	三层网络UUID		0.6
ipRangeUids (可选)	List	query	IP地址范围UUID		0.6
all (可选)	boolean	query	系统全局		0.6
systemTags (可选)	List	query	系统标签		0.6
userTags (可选)	List	query	用户标签		0.6

API返回

返回示例

```
{  
  "totalCapacity": 245.0,  
  "availableCapacity": 229.0
```

}

名字	类型	描述	起始版本
totalCapacity	long	IP地址容量	0.6
availableCapacity	long	可用IP地址容量	0.6
success	boolean	成功	0.6
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

SDK示例

Java SDK

```
GetIpAddressCapacityAction action = new GetIpAddressCapacityAction();
action.ipRangeUuids = asList("454d69e78a7849b487ce2266c64cc6e0");
action.all = false;
action.sessionId = "58287d81bc0d441cac85579ed46cb4dd";
GetIpAddressCapacityAction.Result res = action.call();
```

Python SDK

```
GetIpAddressCapacityAction action = GetIpAddressCapacityAction()
action.ipRangeUuids = [2ef2089aa3ba4929bfde743c9d1cb674]
action.all = false
action.sessionId = "349872e1c8ad445bbc0b024ac6ac0fbb"
```

```
GetIpAddressCapacityAction.Result res = action.call()
```

4.2.13 添加IP地址范围(AddIpRange)

API请求

URLs

```
POST zstack/v1/l3-networks/{l3NetworkUuid}/ip-ranges
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "name": "Test-IP-Range",
    "startIp": "192.168.100.10",
    "endIp": "192.168.100.250",
    "netmask": "255.255.255.0",
    "gateway": "192.168.100.1"
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"name":"Test-IP-Range","startIp":"192.168.100.10","endIp":"192.168.100.250","netmask":"255.255.255.0","gateway":"192.168.100.1"}}' \
http://localhost:8080/zstack/v1/l3-networks/acc0573590db3297814610cab774b60b/ip-ranges
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l3NetworkUuid	String	url	三层网络UUID		0.6
name	String	body(包含在params结构中)	三层网络的名称		0.6

名字	类型	位置	描述	可选值	起始版本
description (可选)	String	body(包含在params结构中)	三层网络的详细描述		0.6
startIp	String	body(包含在params结构中)	起始地址		0.6
endIp	String	body(包含在params结构中)	结束地址		0.6
netmask	String	body(包含在params结构中)	网络掩码		0.6
gateway	String	body(包含在params结构中)	网关		0.6
resourceUuid (可选)	String	body(包含在params结构中)	资源UUID。若指定，三层网络会使用该字段值作为UUID		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

返回示例

```
{
  "inventory": {
    "I3NetworkUuid": "4cbec970df744e258726f71de79e6497",
    "name": "Test-IP-Range",
    "startIp": "192.168.100.10",
    "endIp": "192.168.100.250",
    "netmask": "255.255.255.0",
    "gateway": "192.168.100.1"
  }
}
```

}

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventory	IpRangeInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6
l3NetworkUuid	String	三层网络UUID	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startIp	String		0.6
endIp	String		0.6
netmask	String		0.6
gateway	String		0.6

名字	类型	描述	起始版本
networkCidr	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

SDK示例

Java SDK

```
AddIpRangeAction action = new AddIpRangeAction();
action.l3NetworkUuid = "470ba00c2a70496d85294f47ef25c802";
action.name = "Test-IP-Range";
action.startIp = "192.168.100.10";
action.endIp = "192.168.100.250";
action.netmask = "255.255.255.0";
action.gateway = "192.168.100.1";
action.sessionId = "52f08437bbfb43d89f8afbf5af4abd56";
AddIpRangeAction.Result res = action.call();
```

Python SDK

```
AddIpRangeAction action = AddIpRangeAction()
action.l3NetworkUuid = "f92c5975520f415288156e5a887598ed"
action.name = "Test-IP-Range"
action.startIp = "192.168.100.10"
action.endIp = "192.168.100.250"
action.netmask = "255.255.255.0"
action.gateway = "192.168.100.1"
action.sessionId = "b8705737cc3647c78573729c9fcee1d0"
AddIpRangeAction.Result res = action.call()
```

4.2.14 删除IP地址范围(DeleteIpRange)

API请求

URLs

```
DELETE/v1/l3-networks/ip-ranges/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 1872261fb9824d9f9dd73a991cf05a44" \
```

```
-X DELETE http://localhost:8080/zstack/v1/l3-networks/ip-ranges/a2a9f85d22914afe95a195b63143830d?
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID ，唯一标示该资源		0.6
deleteMode (可选)	String	body	删除模式		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

该API成功时返回一个空的JSON结构{}，出错时返回的JSON结构包含一个error字段，例如：

```
{
  "error": {
    "code": "SYS.1001",
    "description": "A message or a operation timeout",
    "details": "Create VM on KVM timeout after 300s"
  }
}
```

SDK示例

Java SDK

```
DeletelpRangeAction action = new DeletelpRangeAction();
action.uuid = "6d1f44d0fe1440189eb293f18d7bf682";
action.deleteMode = "Permissive";
action.sessionId = "4a3745cb42f5430f874c6f6c4b298e95";
DeletelpRangeAction.Result res = action.call();
```

Python SDK

```
DeletelpRangeAction action = DeletelpRangeAction()
action.uuid = "0f1fcf2e779543baa416deb89eeeb2db"
action.deleteMode = "Permissive"
action.sessionId = "d07fb42a181d43dda2b6992536b02804"
```

```
DeletelpRangeAction.Result res = action.call()
```

4.2.15 查询IP地址范围(QueryIpRange)

API请求

URLs

```
GET zstack/v1/l3-networks/ip-ranges
GET /v1/l3-networks/ip-ranges/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 8f7b1e0396c54646a4b37affdfdda079" \
-X GET http://localhost:8080/zstack/v1/l3-networks/ip-ranges?q=uuid=887227aa5d5c49e2a8137b79dad3738a
```

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 60a5dfda1a9843809415eb28f2550f94" \
-X GET http://localhost:8080/zstack/v1/l3-networks/ip-ranges/17684784fcc04571a47eb24dc2eb9638
```

可查询字段

运行 **zstack-cli** 命令行工具，输入 **QueryIpRange** 并按 Tab 键查看所有可查询字段以及可跨表查询的源名。

API返回

返回示例

```
{
  "inventories": [
    {
      "l3NetworkUuid": "9a72daf0034443aa81864e593f0f5b32",
      "name": "Test-IPRange",
      "networkCidr": "192.168.10.0/24"
    }
  ]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失	0.6

名字	类型	描述	起始版本
		败, 操作成功时该字段为null。 详情参考 error	
inventories	List	详情参考 inventories	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号, 错误的全局唯一标识, 例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段, 默认为null	0.6
opaque	LinkedHashMap	保留字段, 默认为null	0.6
cause	ErrorCode	根错误, 引发当前错误的源错误, 若无原错误, 该字段为null	0.6

#inventories

名字	类型	描述	起始版本
uuid	String	资源的UUID, 唯一标识该资源	0.6
l3NetworkUuid	String	三层网络UUID	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startIp	String		0.6
endIp	String		0.6
netmask	String		0.6
gateway	String		0.6
networkCidr	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

SDK示例

Java SDK

```
QueryIpRangeAction action = new QueryIpRangeAction();
action.conditions = asList("uuid=d5764d7ab9174e5486e55ba805d28675");
action.sessionId = "5357e54aa55a4846ac96972776d3100f";
QueryIpRangeAction.Result res = action.call();
```

Python SDK

```
QueryIpRangeAction action = QueryIpRangeAction()
action.conditions = ["uuid=2cda79a574344eeb9f38393253ddf858"]
action.sessionId = "c87d6755153549f6a4e7373b601ac99f"
QueryIpRangeAction.Result res = action.call()
```

4.2.16 更新IP地址范围(UpdateIpRange)

API请求

URLs

```
PUT zstack/v1/l3-networks/ip-ranges/{uuid}/actions
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "updateIpRange": {},
  "systemTags": [],
  "userTags": []
}
```

上述示例中systemTags、userTags字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X PUT -d '{"updateIpRange":{}}' \
http://localhost:8080/zstack/v1/l3-networks/ip-ranges/0ebb3eb1c1e134bea93c2f9ed2e2beef/
actions
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID ，唯一标示该资源		0.6
name (可选)	String	body(包含在 updateIpRange 结构中)	三层网络的名称		0.6
description (可选)	String	body(包含在 updateIpRange 结构中)	三层网络的详细描述		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

返回示例

```
{
  "inventory": {
    "l3NetworkUuid": "f7f0ee7b4978413fa36c811e47dd56a3",
    "name": "Test-IPRange",
    "networkCidr": "192.168.10.0/24"
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	0.6
inventory	IpRangeInventory	详情参考 inventory	0.6

#error

名	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6
I3NetworkUuid	String	三层网络UUID	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startIp	String		0.6
endIp	String		0.6
netmask	String		0.6
gateway	String		0.6
networkCidr	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

SDK示例

Java SDK

```
UpdateIpRangeAction action = new UpdateIpRangeAction();
action.uuid = "97054a54a1fc4b49ba3d7a5d44b9e385";
action.sessionId = "8644fd2755f148359edff9521b27868b";
```

```
UpdateIpRangeAction.Result res = action.call();
```

Python SDK

```
UpdateIpRangeAction action = UpdateIpRangeAction()
action.uuid = "2b4499297735445482dc994113574b41"
action.sessionId = "0a16c9ea3e6e4ad9b1ff71a826b67277"
UpdateIpRangeAction.Result res = action.call()
```

4.2.17 通过网络CIDR添加IP地址范围(AddIpRangeByNetworkCidr)

API请求

URLs

```
POST zstack/v1/l3-networks/{l3NetworkUuid}/ip-ranges/by-cidr
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "name": "Test-IPRange",
    "networkCidr": "192.168.10.0/24"
  },
  "systemTags": [],
  "userTags": []
}
```

上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"name":"Test-IPRange","networkCidr":"192.168.10.0/24"}}' \
http://localhost:8080/zstack/v1/l3-networks/c186887bc5e43da7ba42bd5d66eba0cd/ip-ranges/by-cidr
```

参数列表

名字	类型	位置	描述	可选值	起始版本
name	String	body(包含在params结构中)	三层网络的名称		0.6

名字	类型	位置	描述	可选值	起始版本
description (可选)	String	body(包含在params结构中)	三层网络的详细描述		0.6
l3NetworkUuid	String	url	三层网络UUID		0.6
networkCidr	String	body(包含在params结构中)	网络CIDR		0.6
resourceUuid (可选)	String	body(包含在params结构中)	资源UUID。若指定，三层网络会使用该字段值作为UUID		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

返回示例

```
{
  "inventory": {
    "l3NetworkUuid": "4e0926c4fa27455f88c7d94420bf3422",
    "name": "Test-IPRange",
    "networkCidr": "192.168.10.0/24"
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventory	IpRangeInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6
I3NetworkUuid	String	三层网络UUID	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
startIp	String		0.6
endIp	String		0.6
netmask	String		0.6
gateway	String		0.6
networkCidr	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

SDK示例

Java SDK

```
AddIpRangeByNetworkCidrAction action = new AddIpRangeByNetworkCidrAction();
action.name = "Test-IPRange";
action.I3NetworkUuid = "c029e15d877a450282d95de9ad8802b7";
action.networkCidr = "192.168.10.0/24";
```

```
action.sessionId = "556aee28a9ef4f62b8cd4f99e7f151c1";
AddIpRangeByNetworkCidrAction.Result res = action.call();
```

Python SDK

```
AddIpRangeByNetworkCidrAction action = AddIpRangeByNetworkCidrAction()
action.name = "Test-IPRange"
action.l3NetworkUuid = "a42e0f209ff347ffbd0966681b99a075"
action.networkCidr = "192.168.10.0/24"
action.sessionId = "d7048b15af6345529888372c374fd7ac"
AddIpRangeByNetworkCidrAction.Result res = action.call()
```

4.2.18 获取三层网络Mtu值(GetL3NetworkMtu)

API请求

URLs

```
GET zstack/v1/l3-networks/{l3NetworkUuid}/mtu
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth dc112a530fd2402cae6b4962f65eae4e" \
-X GET http://localhost:8080/zstack/v1/l3-networks/be54f79eb90f4f929da7fe64e1aea8e4/mtu
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l3NetworkUuid	String	url	三层网络UUID		2.1
systemTags (可选)	List	query			2.1
userTags (可选)	List	query			2.1

API返回

返回示例

```
{
  "mtu": 9216.0
}
```

}

名字	类型	描述	起始版本
mtu	Integer		2.1
success	boolean		2.1
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	2.1
error	ErrorCode	详情参考 error	2.1

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.1
description	String	错误的概要描述	2.1
details	String	错误的详细信息	2.1
elaboration	String	保留字段，默认为null	2.1
opaque	LinkedHashMap	保留字段，默认为null	2.1
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.1

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.1
description	String	错误的概要描述	2.1
details	String	错误的详细信息	2.1
elaboration	String	保留字段，默认为null	2.1
opaque	LinkedHashMap	保留字段，默认为null	2.1

名字	类型	描述	起始版本
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.1

SDK示例

Java SDK

```
GetL3NetworkMtuAction action = new GetL3NetworkMtuAction();
action.l3NetworkUuid = "fd8d572d777d4383b201ddfce629d50e";
action.sessionId = "0cc5202de3ce4881a305e5fa1d431aaa";
GetL3NetworkMtuAction.Result res = action.call();
```

Python SDK

```
GetL3NetworkMtuAction action = GetL3NetworkMtuAction()
action.l3NetworkUuid = "7615cfd154a048dcab07a1114ee6205d"
action.sessionId = "d0f3cf05a49345438a92108d2fd86cfe"
GetL3NetworkMtuAction.Result res = action.call()
```

4.2.19 设置三层网络Mtu值(SetL3NetworkMtu)

API请求

URLs

```
POST zstack/v1/l3-networks/{l3NetworkUuid}/mtu
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "mtu": 9216.0
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
```

```
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"mtu":9216.0}}' \
http://localhost:8080/zstack/v1/l3-networks/817672c1bb903264a9a41c34332c3145/mtu
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l3NetworkUuid	String	url	三层网络UUID		2.1
mtu	Integer	body(包含在 params 结构中)			2.1
systemTags (可选)	List	body			2.1
userTags (可选)	List	body			2.1

API返回

该API成功时返回一个空的JSON结构{}，出错时返回的JSON结构包含一个error字段，例如：

```
{
  "error": {
    "code": "SYS.1001",
    "description": "A message or a operation timeout",
    "details": "Create VM on KVM timeout after 300s"
  }
}
```

SDK示例

Java SDK

```
SetL3NetworkMtuAction action = new SetL3NetworkMtuAction();
action.l3NetworkUuid = "b19ce058b0f54c6d83c14078c0079c2c";
action.mtu = 9216.0;
action.sessionId = "b441c616886c4183a51ba044e35c7d66";
SetL3NetworkMtuAction.Result res = action.call();
```

Python SDK

```
SetL3NetworkMtuAction action = SetL3NetworkMtuAction()
action.l3NetworkUuid = "67dc01887dfc4142aabe4221675c1b98"
action.mtu = 9216.0
action.sessionId = "d5188e0801994e3a9dfb3d3ebc2f7a39"
```

```
SetL3NetworkMtuAction.Result res = action.call()
```

4.2.20 获取三层网络上路由器的接口地址 (GetL3NetworkRouterInterfaceIp)

API请求

URLs

```
GET zstack/v1/l3-networks/{l3NetworkUuid}/router-interface-ip
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X GET http://localhost:8080/zstack/v1/l3-networks/71aa55703e063d9db0f13fa0b285a82b/
router-interface-ip
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l3NetworkUuid	String	url	三层网络UUID		2.2
systemTags (可选)	List	query			2.2
userTags (可 选)	List	query			2.2

API返回

返回示例

```
{
  "routerInterfaceIp": "192.168.0.2"
}
```

名字	类型	描述	起始版本
routerInterfaceIp	String	三层网络上路由器的接口地址，仅当会在普通三层网络上创建云路由	2.2

名字	类型	描述	起始版本
		器或在VPC网络上加载VPC路由器时有效	
success	boolean		2.2
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	2.2

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

SDK示例

Java SDK

```
GetL3NetworkRouterInterfaceIpAction action = new GetL3NetworkRouterInterfaceIpAction();
action.l3NetworkUuid = "23453c09a0f534d5bb10a3267c09aac8";
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c";
GetL3NetworkRouterInterfaceIpAction.Result res = action.call();
```

Python SDK

```
GetL3NetworkRouterInterfaceIpAction action = GetL3NetworkRouterInterfaceIpAction()
action.l3NetworkUuid = "5cece4a676083f6fa3491a068e3e34e6"
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c"
```

```
GetL3NetworkRouterInterfaceIpAction.Result res = action.call()
```

4.2.21 设置三层网络路由器接口IP(SetL3NetworkRouterInterfaceIp)

设置三层网络上路由器接口地址，仅当会在普通三层网络上创建云路由器或在VPC网络上加载VPC路由器时有效。

API请求

URLs

```
POST zstack/v1/l3-networks/{l3NetworkUuid}/router-interface-ip
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "routerInterfaceIp": "192.168.10.2"
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"routerInterfaceIp":"192.168.10.2"}}' \
http://localhost:8080/zstack/v1/l3-networks/40124528890f368c8752825634144fd5/router-interface-ip
```

参数列表

名字	类型	位置	描述	可选值	起始版本
l3NetworkUuid	String	url	三层网络UUID		2.1
routerInterfaceIp	String	body(包含在params结构中)			2.1

名字	类型	位置	描述	可选值	起始版本
systemTags (可选)	List	body			2.1
userTags (可选)	List	body			2.1

API返回

该API成功时返回一个空的JSON结构`{}`，出错时返回的JSON结构包含一个error字段，例如：

```
{
  "error": {
    "code": "SYS.1001",
    "description": "A message or a operation timeout",
    "details": "Create VM on KVM timeout after 300s"
  }
}
```

SDK示例

Java SDK

```
SetL3NetworkRouterInterfacelpAction action = new SetL3NetworkRouterInterfacelpAction();
action.l3NetworkUuid = "00ab12382f544de0ac71b48afe7ecd9c";
action.routerInterfacelp = "192.168.10.2";
action.sessionId = "279a26540d254f42b7a2320652e6c64b";
SetL3NetworkRouterInterfacelpAction.Result res = action.call();
```

Python SDK

```
SetL3NetworkRouterInterfacelpAction action = SetL3NetworkRouterInterfacelpAction()
action.l3NetworkUuid = "3558f265c4204eae93104e3ffa974661"
action.routerInterfacelp = "192.168.10.2"
action.sessionId = "97c0fdecc551496d86b8861f045df21a"
SetL3NetworkRouterInterfacelpAction.Result res = action.call()
```

4.3 路由资源相关接口

4.3.1 创建云路由器(CreateVirtualRouterVm)

API请求

URLs

POST zstack/v1/vm-instances/appliances/virtual-routers

Headers

Authorization: OAuth the-session-uuid

Body

```
{
  "params": {
    "managementNetworkUuid": "8b38d116daad4ee3bc0303296a840be6",
    "publicNetworkUuid": "4488c6d9a79b44828fbc06225fc1ece",
    "name": "Test-Router",
    "instanceOfferingUuid": "ab74f5e92c3b4d3db41d0c4bf0313bec",
    "imageUuid": "5f7f57ea034f4c06aa7a5a1436837d93",
    "clusterUuid": "ef43a5e420e848d198895a185d0f076e",
    "description": "this is a virtual router vm",
    "strategy": "InstantStart"
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 9c9b968bf5a64cd088671926ad1fdbaa" \
-X POST -d '{"params":{"managementNetworkUuid":"7b3415ab979d440b8aacc150db1ea7d9",
"publicNetworkUuid":"e59452b590e84ccbb8d4a0e8444288cb","name":"Test-Router","instanceOfferingUuid":"d95d94d2291349dcbf5e3f6668237766","imageUuid":"2dd018163906422eaf3b6beaf19f43f3","clusterUuid":"a1dd0c1b28a046b79d200677f9fef9d4","description":"this is a virtual router vm","strategy":"InstantStart"}}' \
http://localhost:8080/zstack/v1/vm-instances/appliances/virtual-routers
```

参数列表

名字	类型	位置	描述	可选值	起始版本
managementNetworkUuid	String	body(包含在 params 结构中)	管理L3网络UUID		0.6

名字	类型	位置	描述	可选值	起始版本
publicNetworkUuid	String	body(包含在 params 结构中)	公有L3网络UUID		0.6
networkServicesProvided	Set	body(包含在 params 结构中)	网络服务模块		0.6
name	String	body(包含在 params 结构中)	资源名称		0.6
instanceOfferingUuid	String	body(包含在 params 结构中)	计算规格UUID		0.6
imageUuid	String	body(包含在 params 结构中)	镜像UUID		0.6
l3NetworkUuids	List	body(包含在 params 结构中)	三层网络UUID		0.6
type (可选)	String	body(包含在 params 结构中)	类型	- UserVm - ApplianceVm	0.6
rootDiskOfferingUuid (可选)	String	body(包含在 params 结构中)	根磁盘规格UUID		0.6
dataDiskOfferingUuids (可选)	List	body(包含在 params 结构中)	数据盘规格UUID		0.6
zoneUuid (可选)	String	body(包含在 params 结构中)	区域UUID		0.6
clusterUuid (可选)	String	body(包含在 params 结构中)	集群UUID		0.6

名字	类型	位置	描述	可选值	起始版本
hostUuid (可选)	String	body(包含在 params 结构中)	物理机UUID		0.6
primary Storage UuidFor RootVolume (可选)	String	body(包含在 params 结构中)	根磁盘主存储UUID		0.6
description (可选)	String	body(包含在 params 结构中)	资源的详细描述		0.6
default L3NetworkUuid (可选)	String	body(包含在 params 结构中)	默认三层网络UUID		0.6
strategy (可选)	String	body(包含在 params 结构中)	策略	- InstantStart - JustCreate	0.6
resourceUuid (可选)	String	body(包含在 params 结构中)	资源UUID		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

返回示例

```
{
  "inventory": {
    "uuid": "e71c57036074454faed35d94f3e8d0f9",
    "name": "Test-VM",
    "description": "web server VM",
    "zoneUuid": "e5179b32ff4d4462b78911777021b94c",
    "clusterUuid": "f2fc10c58e92444d842df7942e8a0079",
    "imageUuid": "4ca09642a7e7426db2c256e3ebe37e06",
    "hostUuid": "ebf1b3984c174efcae5c88b1a6fed383",
    "lastHostUuid": "ec76c626b26e4eeca0e8f83756679813",
    "instanceOfferingUuid": "7f8716c0a95245e89b1f59fef6217d68",
    "rootVolumeUuid": "90932aae734f4c5599fe1f9d6bf5c509",
    "platform": "Linux",
```

```

"defaultL3NetworkUuid": "ae7bfdd40d8d4591a7f0efd598e389f1",
"type": "UserVm",
"hypervisorType": "KVM",
"memorySize": 8.589934592E9,
"cpuNum": 1.0,
"allocatorStrategy": "LastHostPreferredAllocatorStrategy",
"createDate": "May 3, 2017 3:46:40 PM",
"lastOpDate": "May 3, 2017 3:46:40 PM",
"state": "Running",
"vmNics": [
  {
    "uuid": "cfc7610b4ed54533b741d71c6d7123ec",
    "vmInstanceUuid": "e71c57036074454faed35d94f3e8d0f9",
    "usedIpUuid": "f560ed8f9b054fe8a1e0f8119a75aa91",
    "l3NetworkUuid": "ae7bfdd40d8d4591a7f0efd598e389f1",
    "ip": "192.168.1.10",
    "mac": "00:0c:29:bd:99:fc",
    "netmask": "255.255.255.0",
    "gateway": "192.168.1.1",
    "deviceId": 0.0,
    "createDate": "May 3, 2017 3:46:40 PM",
    "lastOpDate": "May 3, 2017 3:46:40 PM"
  }
],
"allVolumes": [
  {
    "uuid": "90932aae734f4c5599fe1f9d6bf5c509",
    "name": "Root-Volume-For-VM-e71c57036074454faed35d94f3e8d0f9",
    "primaryStorageUuid": "a1380ec0da6543a38db798921223366a",
    "vmInstanceUuid": "e71c57036074454faed35d94f3e8d0f9",
    "diskOfferingUuid": "b8344fce2e3e4fa4ad5831cf0705d504",
    "rootImageUuid": "4ca09642a7e7426db2c256e3ebe37e06",
    "installPath": "/zstack_ps/rootVolumes/acct-36c27e8ff05c4780bf6d2fa65700f22e/vol-90932aae734f4c5599fe1f9d6bf5c509/90932aae734f4c5599fe1f9d6bf5c509.qcow2",
    "type": "Root",
    "format": "qcow2",
    "size": 1.073741824E11,
    "actualSize": 2.147483648E10,
    "deviceId": 0.0,
    "state": "Enabled",
    "status": "Ready",
    "createDate": "May 3, 2017 3:46:40 PM",
    "lastOpDate": "May 3, 2017 3:46:40 PM"
  }
]
}
}

```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventory	VmInstanceInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
clusterUuid	String	集群UUID	0.6
imageUuid	String	镜像UUID	0.6
hostUuid	String	物理机UUID	0.6
lastHostUuid	String		0.6
instanceOfferingUuid	String	计算规格UUID	0.6
rootVolumeUuid	String	根云盘UUID	0.6
platform	String		0.6
defaultL3NetworkUuid	String		0.6
type	String		0.6
hypervisorType	String		0.6
memorySize	Long		0.6
cpuNum	Integer		0.6

名字	类型	描述	起始版本
cpuSpeed	Long		0.6
allocatorStrategy	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
state	String		0.6
vmNics	List	详情参考 vmNics	0.6
allVolumes	List	详情参考 allVolumes	0.6

#vmNics

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
vmInstanceUuid	String	云主机UUID	0.6
l3NetworkUuid	String	三层网络UUID	0.6
ip	String		0.6
mac	String		0.6
netmask	String		0.6
gateway	String		0.6
metaData	String		0.6
deviceId	Integer		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

#allVolumes

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6

名字	类型	描述	起始版本
primaryStorageUuid	String	主存储UUID	0.6
vmInstanceUuid	String	云主机UUID	0.6
diskOfferingUuid	String	云盘规格UUID	0.6
rootImageUuid	String		0.6
installPath	String		0.6
type	String		0.6
format	String		0.6
size	Long		0.6
actualSize	Long		0.6
deviceId	Integer		0.6
state	String		0.6
status	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
isShareable	Boolean		0.6

SDK示例

Java SDK

```

CreateVirtualRouterVmAction action = new CreateVirtualRouterVmAction();
action.managementNetworkUuid = "8dc604cfaddc4e2395a09f32f60abee1";
action.publicNetworkUuid = "414cdd21b6b84a0db7290e0e4a2a49a2";
action.name = "Test-Router";
action.instanceOfferingUuid = "b74da68053254514be0d597c4b8eea7c";
action.imageUuid = "70a05633d2f54f979327b579bf66cbbc";
action.clusterUuid = "e510a065ac2145358e6efdb12ab4710e";
action.description = "this is a virtual router vm";
action.strategy = "InstantStart";
action.sessionId = "ff8061465d514d95971d2d079aad53ff";
CreateVirtualRouterVmAction.Result res = action.call();

```

Python SDK

```

CreateVirtualRouterVmAction action = CreateVirtualRouterVmAction()
action.managementNetworkUuid = "80020b31b41746ddb6be7f1a54016b19"
action.publicNetworkUuid = "3933ea59c2024821a184abdde140c691"
action.name = "Test-Router"
action.instanceOfferingUuid = "0374f232d2254157963d781dabab516f"
action.imageUuid = "3e101d35e9b645398e51f670b692e024"
action.clusterUuid = "d24db37a8b0347329d4bcd80aeb774c"

```

```

action.description = "this is a virtual router vm"
action.strategy = "InstantStart"
action.sessionId = "16a8ac9702774c7692a9d856f4e74f49"
CreateVirtualRouterVmAction.Result res = action.call()

```

4.3.2 重连云路由器(ReconnectVirtualRouter)

API请求

URLs

```
PUT zstack/v1/vm-instances/appliances/virtual-routers/{vmInstanceUuid}/actions
```

Headers

Authorization: OAuth the-session-uuid

Body

```

{
  "reconnectVirtualRouter": {},
  "systemTags": [],
  "userTags": []
}

```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```

curl -H "Content-Type: application/json" \
-H "Authorization: OAuth bd24a72b74a148e8b18ee2b2367534d6" \
-X PUT -d '{"reconnectVirtualRouter":{}}' \
http://localhost:8080/zstack/v1/vm-instances/appliances/virtual-routers/74f9bb00796c366a83a6201df9e26f43/actions

```

参数列表

名字	类型	位置	描述	可选值	起始版本
vmInstanceUuid	String	url	云主机UUID		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

返回示例

```
{
  "inventory": {
    "managementNetworkUuid": "9a154450da784a418db5406a463d2235",
    "name": "Test-Router",
    "description": "this is a virtual router vm",
    "clusterUuid": "9b359b8de95d4d97a19d53bf4193993a",
    "imageUuid": "c7bbef849e5d47478ee1db445d55530b",
    "instanceOfferingUuid": "5437f63807864c37b4d3042ee28f2680"
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventory	ApplianceVmInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
applianceVmType	String		0.6

名字	类型	描述	起始版本
managementNetworkUuid	String		0.6
defaultRouteL3NetworkUuid	String		0.6
status	String		0.6
agentPort	Integer		0.6
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
clusterUuid	String	集群UUID	0.6
imageUuid	String	镜像UUID	0.6
hostUuid	String	物理机UUID	0.6
lastHostUuid	String		0.6
instanceOfferingUuid	String	计算规格UUID	0.6
rootVolumeUuid	String	根云盘UUID	0.6
platform	String		0.6
defaultL3NetworkUuid	String		0.6
type	String		0.6
hypervisorType	String		0.6
memorySize	Long		0.6
cpuNum	Integer		0.6
cpuSpeed	Long		0.6
allocatorStrategy	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
state	String		0.6
vmNics	List	详情参考 vmNics	0.6

名字	类型	描述	起始版本
allVolumes	List	详情参考 allVolumes	0.6

#vmNics

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
vmInstanceUuid	String	云主机UUID	0.6
l3NetworkUuid	String	三层网络UUID	0.6
ip	String		0.6
mac	String		0.6
netmask	String		0.6
gateway	String		0.6
metaData	String		0.6
deviceId	Integer		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

#allVolumes

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
primaryStorageUuid	String	主存储UUID	0.6
vmInstanceUuid	String	云主机UUID	0.6
diskOfferingUuid	String	云盘规格UUID	0.6
rootImageUuid	String		0.6
installPath	String		0.6
type	String		0.6

名字	类型	描述	起始版本
format	String		0.6
size	Long		0.6
actualSize	Long		0.6
deviceId	Integer		0.6
state	String		0.6
status	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
isShareable	Boolean		0.6

SDK示例

Java SDK

```
ReconnectVirtualRouterAction action = new ReconnectVirtualRouterAction();
action.vmlInstanceUuid = "3b20e87b87b442518eff8744c2cfde8";
action.sessionId = "ce03736cdddf4c49bce35d8e55e189c7";
ReconnectVirtualRouterAction.Result res = action.call();
```

Python SDK

```
ReconnectVirtualRouterAction action = ReconnectVirtualRouterAction()
action.vmlInstanceUuid = "ae937aa909594b1bb2e36359cd7dbc5b"
action.sessionId = "4434d7282a7d4940bb0695c0d252bfae"
ReconnectVirtualRouterAction.Result res = action.call()
```

4.3.3 查询云路由器 (QueryVirtualRouterVm)

API请求

URLs

```
GET zstack/v1/vm-instances/appliances/virtual-routers
GET zstack/v1/vm-instances/appliances/virtual-routers/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 823cd67ecc7c4e6fa25efbc7f9b5130b" \
```

```
-X GET http://localhost:8080/zstack/v1/vm-instances/appliances/virtual-routers?q=name=Test-Router
```

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 5b8a868c89ef434086caeb6ade4613bf" \
-X GET http://localhost:8080/zstack/v1/vm-instances/appliances/virtual-routers/a1f8c5edc13743d79ec8c9d912d1c280
```

可查询字段

运行 `zstack-cli` 命令行工具，输入 `QueryVirtualRouterVm` 并按 Tab 键查看所有可查询字段以及可跨表查询的资源名。

API返回

返回示例

```
{
  "inventories": [
    {
      "uuid": "eb4c132875424e19aa200c5685f7d9ca",
      "name": "Test-VM",
      "description": "Appliance VM",
      "zoneUuid": "453595b58b1f407aa8b551c279738191",
      "clusterUuid": "bc24a488a7c740ef828839d6624b15a2",
      "imageUuid": "729f3c664fb14ffa8886cbf5d14bec47",
      "hostUuid": "bdd8a04c22b6482287a2ac071319f59d",
      "lastHostUuid": "bc6337a5a73142c8afbc4108f69595cf",
      "instanceOfferingUuid": "5beedb45ba9547a4a154c747c69c38bd",
      "rootVolumeUuid": "b07d68c5879b49a680f6fc375d06603f",
      "platform": "Linux",
      "defaultL3NetworkUuid": "dfd914d7f33a434a880fd0cd29db1ad1",
      "type": "UserVm",
      "hypervisorType": "KVM",
      "memorySize": 8.589934592E9,
      "cpuNum": 1.0,
      "allocatorStrategy": "LastHostPreferredAllocatorStrategy",
      "createDate": "Jun 7, 2017 9:21:17 PM",
      "lastOpDate": "Jun 7, 2017 9:21:17 PM",
      "state": "Running",
      "vmNics": [
        {
          "uuid": "ba1a965bee39412c962735987da7722a",
          "vmInstanceUuid": "eb4c132875424e19aa200c5685f7d9ca",
          "usedIpUuid": "7250d89102a44251aa49a5bcf093d484",
          "l3NetworkUuid": "dfd914d7f33a434a880fd0cd29db1ad1",
          "ip": "192.168.1.10",
          "mac": "00:0c:29:bd:99:fc",
          "netmask": "255.255.255.0",
          "gateway": "192.168.1.1",
          "deviceId": 0.0,
          "createDate": "Jun 7, 2017 9:21:17 PM",
          "lastOpDate": "Jun 7, 2017 9:21:17 PM"
        }
      ],
      "allVolumes": [
        {
          "uuid": "b07d68c5879b49a680f6fc375d06603f",
```

```

"name": "Root-Volume-For-VM-eb4c132875424e19aa200c5685f7d9ca",
"primaryStorageUuid": "f82bd3273e1a42a7bc53f1d098d07c63",
"vmInstanceUuid": "eb4c132875424e19aa200c5685f7d9ca",
"diskOfferingUuid": "71a6bbc236974c188ad258875a0cc837",
"rootImageUuid": "729f3c664fb1ffa8886cbf5d14bec47",
"installPath": "/zstack_ps/rootVolumes/acct-36c27e8ff05c4780bf6d2fa65700f22e/vol-b07d68c5879b49a680f6fc375d06603f/b07d68c5879b49a680f6fc375d06603f.qcow2",
"type": "Root",
"format": "qcow2",
"size": 1.073741824E11,
"actualSize": 2.147483648E10,
"deviceId": 0.0,
"state": "Enabled",
"status": "Ready",
"createDate": "Jun 7, 2017 9:21:17 PM",
"lastOpDate": "Jun 7, 2017 9:21:17 PM"
}
]
}
]
}

```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventories	List	详情参考 inventories	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventories

名字	类型	描述	起始版本
applianceVmType	String		0.6
managementNetworkUuid	String		0.6
defaultRouteL3NetworkUuid	String		0.6
status	String		0.6
agentPort	Integer		0.6
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
clusterUuid	String	集群UUID	0.6
imageUuid	String	镜像UUID	0.6
hostUuid	String	物理机UUID	0.6
lastHostUuid	String		0.6
instanceOfferingUuid	String	计算规格UUID	0.6
rootVolumeUuid	String	根云盘UUID	0.6
platform	String		0.6
defaultL3NetworkUuid	String		0.6
type	String		0.6
hypervisorType	String		0.6
memorySize	Long		0.6
cpuNum	Integer		0.6
cpuSpeed	Long		0.6
allocatorStrategy	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
state	String		0.6

名字	类型	描述	起始版本
vmNics	List	详情参考 vmNics	0.6
allVolumes	List	详情参考 allVolumes	0.6

#vmNics

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
vmInstanceUuid	String	云主机UUID	0.6
l3NetworkUuid	String	三层网络UUID	0.6
ip	String		0.6
mac	String		0.6
netmask	String		0.6
gateway	String		0.6
metaData	String		0.6
deviceId	Integer		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

#allVolumes

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
primaryStorageUuid	String	主存储UUID	0.6
vmInstanceUuid	String	云主机UUID	0.6
diskOfferingUuid	String	云盘规格UUID	0.6
rootImageUuid	String		0.6
installPath	String		0.6

名字	类型	描述	起始版本
type	String		0.6
format	String		0.6
size	Long		0.6
actualSize	Long		0.6
deviceId	Integer		0.6
state	String		0.6
status	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
isShareable	Boolean		0.6

SDK示例

Java SDK

```
QueryVirtualRouterVmAction action = new QueryVirtualRouterVmAction();
action.conditions = asList("name=Test-Router");
action.sessionId = "f25578538ecf4a55b98d51e5b8788adb";
QueryVirtualRouterVmAction.Result res = action.call();
```

Python SDK

```
QueryVirtualRouterVmAction action = QueryVirtualRouterVmAction()
action.conditions = ["name=Test-Router"]
action.sessionId = "d1cd40ad61f149689bb358c997377d4a"
QueryVirtualRouterVmAction.Result res = action.call()
```

4.3.4 查询服务云主机(QueryApplianceVm)

API请求

URLs

```
GET zstack/v1/vm-instances/appliances
```

```
GET zstack/v1/vm-instances/appliances/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth fee14fed21eb4535bcd2fd6a11fa9dcd" \
-X GET http://localhost:8080/zstack/v1/vm-instances/appliances?q=uuid=59b2f0ff7d3e49faab66fdb4bdfee249
```

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth f11e69b5788b458e8149425132902365" \
-X GET http://localhost:8080/zstack/v1/vm-instances/appliances/5eaf0998a54345e5bb25bfdad34b2b31
```

可查询字段

运行 **zstack-cli** 命令行工具，输入 **QueryApplianceVm** 并按 Tab 键查看所有可查询字段以及可跨表查询的资源名。

API返回

返回示例

```
{
  "inventories": [
    {
      "uuid": "f71a399a79794ab8a41e26a601ac7387",
      "name": "Test-VM",
      "description": "Appliance VM",
      "zoneUuid": "4cf173a14e67440c9349c1e236a56c37",
      "clusterUuid": "07412e72791c474fa240b75da1b210c5",
      "imageUuid": "7bc33ed9760f424a9b9ff21fc0bbb48e",
      "hostUuid": "0cbed62c52bc407e994c966f246363bd",
      "lastHostUuid": "08921382c4f44705bae7cba6414c811d",
      "instanceOfferingUuid": "e056c84995cb4d68ac78dbcc30af3f4f",
      "rootVolumeUuid": "2557a6f997ee457fad526e179a76e8c0",
      "platform": "Linux",
      "defaultL3NetworkUuid": "748fc87b7fcb456cbea47432e52b377d",
      "type": "UserVm",
      "hypervisorType": "KVM",
      "memorySize": 8.589934592E9,
      "cpuNum": 1.0,
      "allocatorStrategy": "LastHostPreferredAllocatorStrategy",
      "createDate": "Jun 7, 2017 9:20:39 PM",
      "lastOpDate": "Jun 7, 2017 9:20:39 PM",
      "state": "Running",
      "vmNics": [
        {
          "uuid": "d2dfc67ddf28431bb64cc90fe0e113f6",
          "vmInstanceUuid": "f71a399a79794ab8a41e26a601ac7387",
          "usedIpUuid": "f654374b78534f5fb9b75873084932a3",
          "l3NetworkUuid": "748fc87b7fcb456cbea47432e52b377d",

```

```

"ip": "192.168.1.10",
"mac": "00:0c:29:bd:99:fc",
"netmask": "255.255.255.0",
"gateway": "192.168.1.1",
"deviceId": 0.0,
"createDate": "Jun 7, 2017 9:20:39 PM",
"lastOpDate": "Jun 7, 2017 9:20:39 PM"
},
],
"allVolumes": [
{
"uuid": "2557a6f997ee457fad526e179a76e8c0",
"name": "Root-Volume-For-VM-f71a399a79794ab8a41e26a601ac7387",
"primaryStorageUuid": "d00057b39df9494bbaf00d047dd04825",
"vmInstanceUuid": "f71a399a79794ab8a41e26a601ac7387",
"diskOfferingUuid": "20346d1b6643492fbc4da8d6baf4b6d2",
"rootImageUuid": "7bc33ed9760f424a9b9ff21fc0bbb48e",
"installPath": "/zstack_ps/rootVolumes/acct-36c27e8ff05c4780bf6d2fa65700f22e/vol-2557a6f997ee457fad526e179a76e8c0/2557a6f997ee457fad526e179a76e8c0.qcow2",
"type": "Root",
"format": "qcow2",
"size": 1.073741824E11,
"actualSize": 2.147483648E10,
"deviceId": 0.0,
"state": "Enabled",
"status": "Ready",
"createDate": "Jun 7, 2017 9:20:39 PM",
"lastOpDate": "Jun 7, 2017 9:20:39 PM"
}
]
}
}
}

```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventories	List	详情参考 inventories	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6

名字	类型	描述	起始版本
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventories

名字	类型	描述	起始版本
applianceVmType	String		0.6
managementNetworkUuid	String		0.6
defaultRouteL3NetworkUuid	String		0.6
status	String		0.6
agentPort	Integer		0.6
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
zoneUuid	String	区域UUID	0.6
clusterUuid	String	集群UUID	0.6
imageUuid	String	镜像UUID	0.6
hostUuid	String	物理机UUID	0.6
lastHostUuid	String		0.6
instanceOfferingUuid	String	计算规格UUID	0.6
rootVolumeUuid	String	根云盘UUID	0.6
platform	String		0.6
defaultL3NetworkUuid	String		0.6
type	String		0.6
hypervisorType	String		0.6
memorySize	Long		0.6

名字	类型	描述	起始版本
cpuNum	Integer		0.6
cpuSpeed	Long		0.6
allocatorStrategy	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
state	String		0.6
vmNics	List	详情参考 vmNics	0.6
allVolumes	List	详情参考 allVolumes	0.6

#vmNics

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
vmInstanceUuid	String	云主机UUID	0.6
l3NetworkUuid	String	三层网络UUID	0.6
ip	String		0.6
mac	String		0.6
netmask	String		0.6
gateway	String		0.6
metaData	String		0.6
deviceId	Integer		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6

#allVolumes

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6

名字	类型	描述	起始版本
description	String	资源的详细描述	0.6
primaryStorageUuid	String	主存储UUID	0.6
vmInstanceUuid	String	云主机UUID	0.6
diskOfferingUuid	String	云盘规格UUID	0.6
rootImageUuid	String		0.6
installPath	String		0.6
type	String		0.6
format	String		0.6
size	Long		0.6
actualSize	Long		0.6
deviceId	Integer		0.6
state	String		0.6
status	String		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
isShareable	Boolean		0.6

SDK示例

Java SDK

```
QueryApplianceVmAction action = new QueryApplianceVmAction();
action.conditions = asList("uuid=691c9a2eb7964cdd8cfad52a4acece7b");
action.sessionId = "9fec03d462184d6b9a87d314c00597ed";
QueryApplianceVmAction.Result res = action.call();
```

Python SDK

```
QueryApplianceVmAction action = QueryApplianceVmAction()
action.conditions = ["uuid=0c17514849ac49809d6e6047fc4b4978"]
action.sessionId = "485840ed54bc40cc9e47feabd0475ae7"
```

```
QueryApplianceVmAction.Result res = action.call()
```

4.3.5 创建云路由规格(CreateVirtualRouterOffering)

API请求

URLs

```
POST zstack/v1/instance-offerings/virtual-routers
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "zoneUuid": "b2826f6c4a5b433b87916e2240132ffa",
    "managementNetworkUuid": "2f8e7e8636db4313b735edddc488a961",
    "imageUuid": "bc78c126c6a94ee3aec0c840b50730a7",
    "publicNetworkUuid": "95c8cc36b34e4ad89d0eb9b0b6ee02e6",
    "isDefault": true,
    "name": "VirtualRouter-Offering",
    "cpuNum": 2.0,
    "memorySize": 1024.0,
    "sortKey": 0.0,
    "type": "VirtualRouter"
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth bca83008f4154b488a8167f021144e4d" \
-X POST -d '{"params":{"zoneUuid":"a0f488a96504388cae48c6c489c5f368","managementNetworkUuid":"3f5412e1ad0b3021bbdb77503fc7de86","imageUuid":"23743ebd52213235a11e21245100bd57","publicNetworkUuid":"1a24d621f66432eb845b3835697538ed","isDefault":true,"name":"VirtualRouter-Offering","cpuNum":2.0,"memorySize":1024.0,"sortKey":0.0,"type":"VirtualRouter"}}' \
http://localhost:8080/zstack/v1/instance-offerings/virtual-routers
```

参数列表

名字	类型	位置	描述	可选值	起始版本
zoneUuid	String	body(包含在 params 结构中)	区域UUID		0.6
managementNetworkUuid	String	body(包含在 params 结构中)	管理L3网络UUID		0.6
imageUuid	String	body(包含在 params 结构中)	镜像UUID		0.6
publicNetworkUuid (可选)	String	body(包含在 params 结构中)	公有L3网络UUID		0.6
isDefault (可选)	Boolean	body(包含在 params 结构中)	默认		0.6
name	String	body(包含在 params 结构中)	资源名称		0.6
description (可选)	String	body(包含在 params 结构中)	资源的详细描述		0.6
cpuNum	int	body(包含在 params 结构中)	CPU数量		0.6
memorySize	long	body(包含在 params 结构中)	内存大小		0.6
allocatorStrategy (可选)	String	body(包含在 params 结构中)	分配策略		0.6
sortKey (可选)	int	body(包含在 params 结构中)	排序主键		0.6

名字	类型	位置	描述	可选值	起始版本
type (可选)	String	body(包含在 params 结构中)	类型		0.6
resourceUuid (可选)	String	body(包含在 params 结构中)	资源UUID		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可选)	List	body	用户标签		0.6

API返回

返回示例

```
{
  "inventory": {
    "uuid": "db5bb944a9ef48eda9ff68d3eaedefed",
    "name": "instanceOffering1",
    "cpuNum": 2.0,
    "cpuSpeed": 1.0,
    "type": "UserVm",
    "allocatorStrategy": "Mevoco",
    "createDate": "Jun 7, 2017 9:20:13 PM",
    "lastOpDate": "Jun 7, 2017 9:20:13 PM",
    "state": "Enabled"
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	0.6
inventory	InstanceOfferingInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
cpuNum	Integer	CPU数量	0.6
cpuSpeed	Integer	CPU速度	0.6
memorySize	Long	内存大小	0.6
type	String	类型	0.6
allocatorStrategy	String	分配策略	0.6
sortKey	Integer		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
state	String	状态（启用，禁用）	0.6

SDK示例

Java SDK

```
CreateVirtualRouterOfferingAction action = new CreateVirtualRouterOfferingAction();
action.zoneUuid = "389c6a48c332450ea280933b34e61ac4";
```

```

action.managementNetworkUuid = "381c2166c0454d95aed829715ae294b8";
action.imageUuid = "a4da1399797145bc87e36b424e5336af";
action.publicNetworkUuid = "5bf6caac74304a489eda76ae64624f6b";
action.isDefault = true;
action.name = "VirtualRouter-Offering";
action.cpuNum = 2.0;
action.memorySize = 1024.0;
action.sortKey = 0.0;
action.type = "VirtualRouter";
action.sessionId = "b811bf42d0b541ea993787b1fa80e724";
CreateVirtualRouterOfferingAction.Result res = action.call();

```

Python SDK

```

CreateVirtualRouterOfferingAction action = CreateVirtualRouterOfferingAction()
action.zoneUuid = "d115bed650ad45128aff5216c9f72d23"
action.managementNetworkUuid = "1b4f915b327d488db04c517b767e2731"
action.imageUuid = "737992cde79646ce915681b74e4c4de7"
action.publicNetworkUuid = "6af154a9cd274134ac44d2d889474158"
action.isDefault = true
action.name = "VirtualRouter-Offering"
action.cpuNum = 2.0
action.memorySize = 1024.0
action.sortKey = 0.0
action.type = "VirtualRouter"
action.sessionId = "71db629e6f484b3187c48ecb1d65dcea"
CreateVirtualRouterOfferingAction.Result res = action.call()

```

4.3.6 查询云路由规格(QueryVirtualRouterOffering)

API请求

URLs

```

GET zstack/v1/instance-offerings/virtual-routers
GET zstack/v1/instance-offerings/virtual-routers/{uuid}

```

Headers

```

Authorization: OAuth the-session-uuid

```

Curl示例

```

curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 94af029a0bed40d68dbabeff17f22add" \
-X GET http://localhost:8080/zstack/v1/instance-offerings/virtual-routers

```

```

curl -H "Content-Type: application/json" \
-H "Authorization: OAuth db0361e95b29493db9bc03cd35db6129" \
-X GET http://localhost:8080/zstack/v1/instance-offerings/virtual-routers/e422506307254250b817c4f7aa70f277

```

可查询字段

运行 `zstack-cli` 命令行工具，输入 `QueryVirtualRouterOffering` 并按 Tab 键查看所有可查询字段以及可跨表查询的资源名。

API 返回

返回示例

```
{
  "inventories": [
    {
      "managementNetworkUuid": "6d5e54344898416c9a20a104341c82d5",
      "publicNetworkUuid": "c6a23a0ed6ad4337a35d8a946694f8ea",
      "zoneUuid": "a12ffae2039847a8b9046478ad03fec",
      "isDefault": true,
      "imageUuid": "ca56a16b15bb47c0a26055e28d1e621f",
      "name": "VirtualRouter-Offering",
      "cpuNum": 2.0,
      "cpuSpeed": 1.0,
      "memorySize": 1024.0,
      "type": "VirtualRouter"
    }
  ]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	0.6
inventories	List	详情参考 inventories	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventories

名字	类型	描述	起始版本
managementNetworkUuid	String		0.6
publicNetworkUuid	String		0.6
zoneUuid	String	区域UUID	0.6
isDefault	Boolean		0.6
imageUuid	String	镜像UUID	0.6
uuid	String	资源的UUID，唯一标示该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
cpuNum	Integer		0.6
cpuSpeed	Integer		0.6
memorySize	Long		0.6
type	String		0.6
allocatorStrategy	String		0.6
sortKey	Integer		0.6
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
state	String		0.6

SDK示例

Java SDK

```
QueryVirtualRouterOfferingAction action = new QueryVirtualRouterOfferingAction();
action.conditions = asList();
action.sessionId = "8779bc03a81d463bbebee36eedcc8197";
QueryVirtualRouterOfferingAction.Result res = action.call();
```

Python SDK

```
QueryVirtualRouterOfferingAction action = QueryVirtualRouterOfferingAction()
action.conditions = []
action.sessionId = "ec9a74a6c42241fb95575a9ba79c6fa6"
```

```
QueryVirtualRouterOfferingAction.Result res = action.call()
```

4.3.7 更新云路由规格(UpdateVirtualRouterOffering)

API请求

URLs

```
PUT zstack/v1/instance-offerings/virtual-routers/{uuid}/actions
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "updateVirtualRouterOffering": {
    "isDefault": false
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X PUT -d '{"updateVirtualRouterOffering":{"isDefault":false}}' \
http://localhost:8080/zstack/v1/instance-offerings/virtual-routers/46a68145223e3c4fb25b5de2b12e43c2/actions
```

参数列表

名字	类型	位置	描述	可选值	起始版本
isDefault (可选)	Boolean	body(包含在 updateVirtualRouterOffering 结构中)	默认		0.6
imageUuid (可选)	String	body(包含在 updateVirtualRouterOffering 结构中)	镜像UUID		0.6

名字	类型	位置	描述	可选值	起始版本
		outerOffering 结构中)			
uuid	String	url	资源的UUID ，唯一标示该 资源		0.6
name (可选)	String	body(包含 在 updateV irtualR outerOffering 结构中)	资源名称		0.6
description (可 选)	String	body(包含 在 updateV irtualR outerOffering 结构中)	资源的详细描 述		0.6
systemTags (可选)	List	body	系统标签		0.6
userTags (可 选)	List	body	用户标签		0.6
allocatorStrat egy (可选)	String	body(包含 在 updateV irtualR outerOffering 结构中)			2.3

API返回

返回示例

```
{
  "inventory": {
    "uuid": "99395ef3a9083f7ba2d64201d6afc6b0",
    "name": "instanceOffering1",
    "cpuNum": 2.0,
    "cpuSpeed": 1.0,
    "type": "UserVm",
    "allocatorStrategy": "Mevoco",
    "createDate": "Nov 14, 2017 10:20:57 PM",
    "lastOpDate": "Nov 14, 2017 10:20:57 PM",
    "state": "Enabled"
  }
}
```

}

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	0.6
inventory	InstanceOfferingInventory	详情参考 inventory	0.6

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	0.6
description	String	错误的概要描述	0.6
details	String	错误的详细信息	0.6
elaboration	String	保留字段，默认为null	0.6
opaque	LinkedHashMap	保留字段，默认为null	0.6
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	0.6

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	0.6
name	String	资源名称	0.6
description	String	资源的详细描述	0.6
cpuNum	Integer	CPU数量	0.6
cpuSpeed	Integer	CPU速度	0.6
memorySize	Long	内存大小	0.6
type	String	类型	0.6

名字	类型	描述	起始版本
allocatorStrategy	String	分配策略	0.6
sortKey	Integer	0.6	
createDate	Timestamp	创建时间	0.6
lastOpDate	Timestamp	最后一次修改时间	0.6
state	String	状态（启用，禁用）	0.6

SDK示例

Java SDK

```
UpdateVirtualRouterOfferingAction action = new UpdateVirtualRouterOfferingAction();
action.isDefault = false;
action.uuid = "46a68145223e3c4fb25b5de2b12e43c2";
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c";
UpdateVirtualRouterOfferingAction.Result res = action.call();
```

Python SDK

```
UpdateVirtualRouterOfferingAction action = UpdateVirtualRouterOfferingAction()
action.isDefault = false
action.uuid = "46a68145223e3c4fb25b5de2b12e43c2"
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c"
UpdateVirtualRouterOfferingAction.Result res = action.call()
```

4.3.8 获取云路由器可加载外部网络(GetAttachablePublicL3ForVRouter)

获取云路由可加载的公有网络和系统网络，将自动排除地址冲突的网络。

API请求

URLs

```
GET zstack/v1/vm-instances/appliances/virtual-routers/{vmInstanceUuid}/attachable-public-l3s
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 15085fa6b94f46c1bd4bb4f6b69aa092" \
```

```
-X GET http://localhost:8080/zstack/v1/vm-instances/appliances/virtual-routers/ac547d8a596c45ca864cd53235499c18/attachable-public-l3s
```

参数列表

名字	类型	位置	描述	可选值	起始版本
vmInstanceUuid	String	url	云路由的UUID		2.2
systemTags (可选)	List	query			2.2
userTags (可选)	List	query			2.2

API返回

返回示例

```
{
  "inventories": [
    {
      "uuid": "03108cd5c1c9427ea954fe35e8ae6811",
      "name": "test-pub-l3",
      "description": "test pub l3",
      "zoneUuid": "a18dc0e9dc054349af6aded2ef17f0",
      "l2NetworkUuid": "e1791fb069554a69b05caf77d6d900a7",
      "system": true,
      "createDate": "Sep 29, 2017 2:41:24 PM",
      "lastOpDate": "Sep 29, 2017 2:41:24 PM",
      "ipRanges": [
        {
          "l3NetworkUuid": "3c26ee27f6e549b9974535ab1e04ad33",
          "name": "test ip range",
          "startIp": "100.64.0.10",
          "endIp": "100.64.0.100",
          "netmask": "255.255.255.0",
          "gateway": "100.64.0.1",
          "networkCidr": "100.64.0.0/24",
          "createDate": "Sep 29, 2017 2:41:24 PM",
          "lastOpDate": "Sep 29, 2017 2:41:24 PM"
        }
      ]
    }
  ]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	2.2

名字	类型	描述	起始版本
inventories	List	详情参考 inventories	2.2

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.2
description	String	错误的概要描述	2.2
details	String	错误的详细信息	2.2
elaboration	String	保留字段，默认为null	2.2
opaque	LinkedHashMap	保留字段，默认为null	2.2
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.2

#inventories

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	2.2
name	String	资源名称	2.2
description	String	资源的详细描述	2.2
type	String		2.2
zoneUuid	String	区域UUID	2.2
l2NetworkUuid	String	二层网络UUID	2.2
state	String		2.2
dnsDomain	String		2.2
system	Boolean		2.2
category	String		2.2
createDate	Timestamp	创建时间	2.2
lastOpDate	Timestamp	最后一次修改时间	2.2

名字	类型	描述	起始版本
dns	List		2.2
ipRanges	List	详情参考 ipRanges	2.2
networkServices	List	详情参考 networkServices	2.2

#ipRanges

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.2
l3NetworkUuid	String	三层网络UUID	2.2
name	String	资源名称	2.2
description	String	资源的详细描述	2.2
startIp	String		2.2
endIp	String		2.2
netmask	String		2.2
gateway	String		2.2
networkCidr	String		2.2
createDate	Timestamp	创建时间	2.2
lastOpDate	Timestamp	最后一次修改时间	2.2

#networkServices

名字	类型	描述	起始版本
l3NetworkUuid	String	三层网络UUID	2.2
networkServiceProviderUuid	String	网络服务提供模块UUID	2.2
networkServiceType	String		2.2

SDK示例

Java SDK

```
GetAttachablePublicL3ForVRouterAction action = new GetAttachablePublicL3ForVRouterAction();
```

```
action.vmlInstanceUuid = "851b21ed7425479daf3783b886f6ab35";
action.sessionId = "8f1a0a142f514a49849df351c40d6116";
GetAttachablePublicL3ForVRouterAction.Result res = action.call();
```

Python SDK

```
GetAttachablePublicL3ForVRouterAction action = GetAttachablePublicL3ForVRouterAction()
action.vmlInstanceUuid = "f78e670c912447b68dcae9a6af79617c"
action.sessionId = "99e6b06b41ff4d16bb2c2b11e4e3a7bb"
GetAttachablePublicL3ForVRouterAction.Result res = action.call()
```

4.3.9 云路由路由表相关接口

4.3.9.1 创建云路由路由表(CreateVRouterRouteTable)

API请求

URLs

```
POST zstack/v1/vrouter-route-tables
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "name": "Test-VRouterRouteTable",
    "description": "Test-VRouterRouteTable"
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中systemTags、userTags字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"name":"Test-VRouterRouteTable","description":"Test-VRouterRouteTable"},"systemTags":[]}' \
http://localhost:8080/zstack/v1/vrouter-route-tables
```

参数列表

名字	类型	位置	描述	可选值	起始版本
name	String	body(包含在params结构中)	资源名称		2.1
description (可选)	String	body(包含在params结构中)	资源的详细描述		2.1
resourceUuid (可选)	String	body(包含在params结构中)			2.1
systemTags (可选)	List	body			2.1
userTags (可选)	List	body			2.1

API返回

返回示例

```
{
  "inventory": {
    "name": "test-route-table",
    "description": "Test route table",
    "attachedRouterRefs": [],
    "routeEntries": []
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	2.1
inventory	VRouterRouteTableInventory	详情参考 inventory	2.1

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.1
description	String	错误的概要描述	2.1
details	String	错误的详细信息	2.1
elaboration	String	保留字段，默认为null	2.1
opaque	LinkedHashMap	保留字段，默认为null	2.1
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.1

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标识该资源	2.1
name	String	资源名称	2.1
description	String	资源的详细描述	2.1
createDate	Timestamp	创建时间	2.1
lastOpDate	Timestamp	最后一次修改时间	2.1
attachedRouterRefs	List	详情参考 attachedRouterRefs	2.1
routeEntries	List	详情参考 routeEntries	2.1

#attachedRouterRefs

名字	类型	描述	起始版本
virtualRouterVmUuid	String		2.1
routeTableUuid	String		2.1

#routeEntries

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.1
description	String	资源的详细描述	2.1
type	String	类型 允许用户添加"静态路由"、"黑洞路由"两种类型，系统会根据是否填下一跳自动判断类型	2.1
routeTableUuid	String	云路由路由表UUID	2.1
destination	String	目标网络地址	2.1
target	String	下一跳地址 为一个云路由设备目前可以直接到达的IP地址，如果不能直接到达，将会进行递归路由	2.1
distance	Integer	路由优先级，在最小匹配下如果有多条路由规则匹配，优先级数字小的规则将会被匹配	2.1
createDate	Timestamp	创建时间	2.1
lastOpDate	Timestamp	最后一次修改时间	2.1

SDK示例

Java SDK

```
CreateVRouterRouteTableAction action = new CreateVRouterRouteTableAction();
action.name = "Test-VRouterRouteTable";
action.description = "Test-VRouterRouteTable";
action.sessionId = "36ae8b4476774bbd819b694d466f5351";
CreateVRouterRouteTableAction.Result res = action.call();
```

Python SDK

```
CreateVRouterRouteTableAction action = CreateVRouterRouteTableAction()
action.name = "Test-VRouterRouteTable"
```

```
action.description = "Test-VRouterRouteTable"
action.sessionId = "c68d30a1ccc9487381ea5ec872cfcccd"
CreateVRouterRouteTableAction.Result res = action.call()
```

4.3.9.2 删除云路由路由表(DeleteVRouterRouteTable)

API请求

URLs

```
DELETE zstack/v1/vrouter-route-tables/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth c92c3c7cf7c64dac9e783ab3414e7901" \
-X DELETE http://localhost:8080/zstack/v1/vrouter-route-tables/865cd01ad01f4c46b0a63163eb13e20b?
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID ，唯一标示该资源		2.1
deleteMode (可选)	String	body			2.1
systemTags (可选)	List	body			2.1
userTags (可选)	List	body			2.1

API返回

该API成功时返回一个空的JSON结构{}，出错时返回的JSON结构包含一个error字段，例如：

```
{
  "error": {
    "code": "SYS.1001",
    "description": "A message or a operation timeout",
    "details": "Create VM on KVM timeout after 300s"
  }
}
```

```
}
```

SDK示例

Java SDK

```
DeleteVRouterRouteTableAction action = new DeleteVRouterRouteTableAction();
action.uuid = "fb0b9186685544bcb5af032b1426736a";
action.deleteMode = "Permissive";
action.sessionId = "aa1c19e2398949c99132c47e97f11569";
DeleteVRouterRouteTableAction.Result res = action.call();
```

Python SDK

```
DeleteVRouterRouteTableAction action = DeleteVRouterRouteTableAction()
action.uuid = "dcf18f99774046de99d5791bfba27b35"
action.deleteMode = "Permissive"
action.sessionId = "c2345a70fb6b4ffabc717ca555e60782"
DeleteVRouterRouteTableAction.Result res = action.call()
```

4.3.9.3 查询云路由路由表(QueryVRouterRouteTable)

API请求

URLs

```
GET zstack/v1/vrouter-route-tables
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 48eaf58e3b6e4c51a56c8d9caa991311" \
-X GET http://localhost:8080/zstack/v1/vrouter-route-tables
```

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 20f83215e3bf491782fe7646a42b5c8b" \
-X GET http://localhost:8080/zstack/v1/vrouter-route-tables/3b8991d7295a4cccb08d028911a9164b
```

可查询字段

运行zstack-cli命令行工具，输入QueryVRouterRouteTable并按Tab键查看所有可查询字段以及可跨表查询的资源名。

API返回

该API成功时返回一个空的JSON结构{}，出错时返回的JSON结构包含一个error字段，例如：

```
{
  "error": {
    "code": "SYS.1001",
    "description": "A message or a operation timeout",
    "details": "Create VM on KVM timeout after 300s"
  }
}
```

SDK示例

Java SDK

```
QueryVRouterRouteTableAction action = new QueryVRouterRouteTableAction();
action.conditions = asList();
action.sessionId = "1e7d262944584f57a8908c9acd1ec62f";
QueryVRouterRouteTableAction.Result res = action.call();
```

Python SDK

```
QueryVRouterRouteTableAction action = QueryVRouterRouteTableAction()
action.conditions = []
action.sessionId = "6ad6d8b304984d3ba5b64ddd2062efcb"
QueryVRouterRouteTableAction.Result res = action.call()
```

4.3.9.4 获取路由器实时路由表(GetVRouterRouteTable)

返回云路由设备实际实时路由表，包括非用户添加的全部路由。

API请求

URLs

```
GET zstack/v1/vrouter-route-tables/vrouter/{virtualRouterVmUuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth ef1c95285ef3435c949854955edc6c66" \
-X GET http://localhost:8080/zstack/v1/vrouter-route-tables/vrouter/27e810f019b74106877814ddb0c6e3bf
```

参数列表

名字	类型	位置	描述	可选值	起始版本
virtualRouterVmUuid	String	url	云路由设备UUID		2.1
systemTags (可选)	List	query			2.1
userTags (可选)	List	query			2.1

API返回

返回示例

```
{
  "inventories": [
    {
      "type": "ZStack"
      "destination": "0.0.0.0/0"
      "target": "100.64.201.1"
      "status": "selected"
    },
    {
      "uuid": "7b86eaba601c4ceda5987036c9ea7981"
      "type": "UserStatic"
      "destination": "100.64.0.0/24"
      "target": "100.64.201.10"
      "distance": "128"
      "status": "selected"
    },
    {
      "uuid": "d15ebfe03aec49e6a0453e74eede350d"
      "type": "UserStatic"
      "destination": "100.64.0.0/24"
      "target": "100.64.201.11"
      "distance": "128"
      "status": "active"
    },
    {
      "type": "ZStack"
      "destination": "10.0.57.94/32"
      "target": "eth0"
    },
    {
      "uuid": "50e6070313fa4a05b17907a32be0a63f"
      "type": "UserStatic"
      "destination": "10.0.57.94/32"
      "target": "100.64.201.12"
      "distance": "1"
      "status": "inactive"
    },
    {
      "type": "DirectConnect"
      "destination": "192.168.197.0/24"
      "target": "eth1"
      "status": "selected"
    }
  ],
}
```

```
{
  "uuid": "6b446d5cf776471c913582ab30bd6340"
  "type": "DirectConnect"
  "destination": "192.168.197.0/24"
  "target": "100.64.201.13"
  "distance": "1"
  "status": "inactive"
},
{
  "uuid": "0aad7115c1f94aa1bf5db1a4d6d3fd17"
  "type": "UserBlackHole"
  "destination": "192.168.198.0/24"
  "distance": "1"
  "status": "selected"
},
]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	2.1
inventories	List	详情参考 inventories	2.1

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.1
description	String	错误的概要描述	2.1
details	String	错误的详细信息	2.1
elaboration	String	保留字段，默认为null	2.1
opaque	LinkedHashMap	保留字段，默认为null	2.1
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.1

#inventories

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.1
description	String	资源的详细描述	2.1
type	String	除用户添加的 静态路由、黑洞路由 外， ZStack 类型表示由系统添加的路由， DirectConnect 表示直连路由	2.1
destination	String	目标网络地址	2.1
target	String	下一跳地址或网卡	2.1
distance	Integer	路由优先级，在最小匹配下如果有多条路由规则匹配，优先级数字小的规则将会被匹配	2.1
status	String	路由状态 active：活跃的路由 inactive：表示非活跃路由 selected：表示多路径路由中活跃的路由	2.1

SDK示例

Java SDK

```
GetVRouterRouteTableAction action = new GetVRouterRouteTableAction();
action.virtualRouterVmUuid = "dfa2698bd9f04cd5be56e5b51826359d";
action.sessionId = "8fbf06a3245a42fc99ef63f618364de7";
GetVRouterRouteTableAction.Result res = action.call();
```

Python SDK

```
GetVRouterRouteTableAction action = GetVRouterRouteTableAction()
action.virtualRouterVmUuid = "1646d8ca4e144f63896c26488491e016"
action.sessionId = "161364959641477981aa68cd63b19348"
```

```
GetVRouterRouteTableAction.Result res = action.call()
```

4.3.9.5 添加云路由路由条目(AddVRouterRouteEntry)

API请求

URLs

```
POST zstack/v1/vrouter-route-tables/{routeTableUuid}/route-entries
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "description": "Test route",
    "type": "UserStatic",
    "destination": "192.168.2.0/24",
    "target": "172.20.1.1"
  },
  "systemTags": [],
  "userTags": []
}
```

 **注：** 上述示例中systemTags、userTags字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"description":"Test route","type":"UserStatic","destination":"192.168.2.0/24","target":"172.20.1.1"}}' \
http://localhost:8080/zstack/v1/vrouter-route-tables/011e554588e93a849e890685d6adb6be/route-entries
```

参数列表

名字	类型	位置	描述	可选值	起始版本
description (可选)	String	body(包含在params结构中)	资源的详细描述		2.1
type (可选)	String	body(包含在params结构中)	类型 允许用户添加"静态路由"、"黑洞"	- UserStatic - UserBlackHole	2.1

名字	类型	位置	描述	可选值	起始版本
			路由"两种类型，系统会根据是否填下一跳自动判断类型		
routeTableUuid	String	url	云路由路由表UUID		2.1
destination	String	body(包含在params结构中)	目标网络地址，使用网络地址CIDR格式，如果用户填写的不是标准CIDR格式，系统会自动转换		2.1
target (可选)	String	body(包含在params结构中)	下一跳地址 为一个云路由设备目前可以直接到达的IP地址，如果不能直接到达，将会进行递归路由		2.1
distance (可选)	Integer	body(包含在params结构中)	路由优先级，在最小匹配下如果有多条路由规则匹配，优先级数字小的规则将会被匹配		2.1
resourceUuid (可选)	String	body(包含在params结构中)			2.1
systemTags (可选)	List	body			2.1

名字	类型	位置	描述	可选值	起始版本
userTags (可选)	List	body			2.1

API返回

返回示例

```
{
  "inventory": {
    "description": "Test route",
    "type": "UserStatic",
    "routeTableUuid": "5dad5ac78c3642088dc3ed6bc4d8c853",
    "destination": "192.168.2.0/24",
    "target": "172.20.1.1",
    "distance": 128.0
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	2.1
inventory	VRouterRouteEntryInventory	详情参考 inventory	2.1

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.1
description	String	错误的概要描述	2.1
details	String	错误的详细信息	2.1
elaboration	String	保留字段，默认为null	2.1
opaque	LinkedHashMap	保留字段，默认为null	2.1
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.1

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.1
description	String	资源的详细描述	2.1
type	String	类型，允许用户添加"静态路由"、"黑洞路由"两种类型，系统会根据是否填下一跳自动判断类型，在外在获取实时路由表API还会返回"ZStack"类型表示由系统添加的路由，"DirectConnect"直连路由	2.1
routeTableUuid	String	云路由路由表UUID	2.1
destination	String	目标网络地址	2.1
target	String	下一条地址	2.1
distance	Integer	路由优先级，在最小匹配下如果有多条路由规则匹配，优先级数字小的规则将会被匹配	2.1
createDate	Timestamp	创建时间	2.1
lastOpDate	Timestamp	最后一次修改时间	2.1

SDK示例

Java SDK

```
AddVRouterRouteEntryAction action = new AddVRouterRouteEntryAction();
action.description = "Test route";
action.type = "UserStatic";
action.routeTableUuid = "5c33ae47c7c94aaea08ead913ddfe0b4";
action.destination = "192.168.2.0/24";
action.target = "172.20.1.1";
action.sessionId = "f75a12a294fe4128b105e1d8f97257b3";
```

```
AddVRouterRouteEntryAction.Result res = action.call();
```

Python SDK

```
AddVRouterRouteEntryAction action = AddVRouterRouteEntryAction()
action.description = "Test route"
action.type = "UserStatic"
action.routeTableUuid = "4fab7c04998c44b4b6d2bc4da0b35ddb"
action.destination = "192.168.2.0/24"
action.target = "172.20.1.1"
action.sessionId = "1510b167ddf54461b69e41802ddcf5eb"
AddVRouterRouteEntryAction.Result res = action.call()
```

4.3.9.6 删除云路由路由条目(DeleteVRouterRouteEntry)

API请求

URLs

```
DELETE zstack/v1/vrouter-route-tables/{routeTableUuid}/route-entries/{uuid}
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 07ed3c114a7f4ca0a10861d448a8b973" \
-X DELETE http://localhost:8080/zstack/v1/vrouter-route-tables/a713099cca4c487e8ee46
ed2ad798903/route-entries/44f6816c4c7444b9b93b11c385f2bb19?
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID ，唯一标示该资源		2.1
routeTableUuid	String	url			2.1
deleteMode (可选)	String	body			2.1
systemTags (可选)	List	body			2.1
userTags (可选)	List	body			2.1

API返回

该API成功时返回一个空的JSON结构{}，出错时返回的JSON结构包含一个error字段，例如：

```
{
  "error": {
    "code": "SYS.1001",
    "description": "A message or a operation timeout",
    "details": "Create VM on KVM timeout after 300s"
  }
}
```

SDK示例

Java SDK

```
DeleteVRouterRouteEntryAction action = new DeleteVRouterRouteEntryAction();
action.uuid = "6cc2578afb134d03959feb8cd586de06";
action.routeTableUuid = "e9b7f623b59d4cb5832a0eef61fdc273";
action.deleteMode = "Permissive";
action.sessionId = "eeda6890b1a2428087ae1c3d5b8e1311";
DeleteVRouterRouteEntryAction.Result res = action.call();
```

Python SDK

```
DeleteVRouterRouteEntryAction action = DeleteVRouterRouteEntryAction()
action.uuid = "eb9a0a4ab242430896d50be1a75c1c0d"
action.routeTableUuid = "295cbd7c2b1746ba9907eabecdb6cdf0"
action.deleteMode = "Permissive"
action.sessionId = "dd36b5ccb2444679820945da4ba4ebf3"
DeleteVRouterRouteEntryAction.Result res = action.call()
```

4.3.9.7 查询云路由路由条目(QueryVRouterRouteEntry)

API请求

URLs

```
GET zstack/v1/vrouter-route-tables/route-entries
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth 0a9d413ce14245f89cde947aa57ae4c3" \
-X GET http://localhost:8080/zstack/v1/vrouter-route-tables/route-entries
```

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth eb2e3203532b405cbe9c2fde23deb53e" \
```

```
-X GET http://localhost:8080/zstack/v1/vrouter-route-tables/route-entries/6c19d2d2bad3489f8b095c98fea971a5
```

可查询字段

运行zstack-cli命令行工具，输入QueryVRouterRouteEntry并按Tab键查看所有可查询字段以及可跨表查询的资源名。

API返回

返回示例

```
{
  "inventories": [
    {
      "description": "Test route",
      "type": "UserStatic",
      "routeTableUuid": "4b9b27aa14794f3c98f23818d8a4b281",
      "destination": "192.168.2.0/24",
      "target": "172.20.1.1"
    }
  ]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	2.1
inventories	List	详情参考 inventories	2.1

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.1
description	String	错误的概要描述	2.1
details	String	错误的详细信息	2.1
elaboration	String	保留字段，默认为null	2.1
opaque	LinkedHashMap	保留字段，默认为null	2.1

名字	类型	描述	起始版本
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.1

#inventories

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.1
description	String	资源的详细描述	2.1
type	String	类型 允许用户添加"静态路由"、"黑洞路由"两种类型，系统会根据是否填下一跳自动判断类型	2.1
routeTableUuid	String	云路由路由表UUID	2.1
destination	String	目标网络地址	2.1
target	String	下一跳地址 为一个云路由设备目前可以直接到达的IP地址，如果不能直接到达，将会进行递归路由	2.1
distance	Integer	路由优先级，在最小匹配下如果有多条路由规则匹配，优先级数字小的规则将会被匹配	2.1
createDate	Timestamp	创建时间	2.1
lastOpDate	Timestamp	最后一次修改时间	2.1

SDK示例

Java SDK

```
QueryVRouterRouteEntryAction action = new QueryVRouterRouteEntryAction();
action.conditions = asList();
action.sessionId = "4388bb850faf4cf69e13ecb9bbdcd77e";
QueryVRouterRouteEntryAction.Result res = action.call();
```

Python SDK

```
QueryVRouterRouteEntryAction action = QueryVRouterRouteEntryAction()
action.conditions = []
action.sessionId = "116b877273e84748994435d8603423b4"
QueryVRouterRouteEntryAction.Result res = action.call()
```

4.3.9.8 绑定云路由路由表到云路由器(AttachVRouterRouteTableToVRouter)

API请求

URLs

```
POST zstack/v1/vrouter-route-tables/{routeTableUuid}/attach
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "virtualRouterVmUuid": "49e427183ffe4dd8b7c40a635f11f71b"
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中systemTags、userTags字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"virtualRouterVmUuid":"2b8bbccba54438e994d4d14d570a1807"}}' \
http://localhost:8080/zstack/v1/vrouter-route-tables/ee1fc77524993a60a8c0e4b7245470db/attach
```

参数列表

名字	类型	位置	描述	可选值	起始版本
routeTableUuid	String	url			2.1
virtualRouterVmUuid	String	body(包含在params结构中)	云路由设备UUID		2.1
systemTags (可选)	List	body			2.1
userTags (可选)	List	body			2.1

API返回

返回示例

```
{
  "inventory": {
    "name": "test-route-table",
    "description": "Test route table",
    "attachedRouterRefs": [],
    "routeEntries": []
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	2.1
inventory	VRouterRouteTableInventory	详情参考 inventory	2.1

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.1
description	String	错误的概要描述	2.1
details	String	错误的详细信息	2.1

名字	类型	描述	起始版本
elaboration	String	保留字段，默认为null	2.1
opaque	LinkedHashMap	保留字段，默认为null	2.1
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.1

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.1
name	String	资源名称	2.1
description	String	资源的详细描述	2.1
createDate	Timestamp	创建时间	2.1
lastOpDate	Timestamp	最后一次修改时间	2.1
attachedRouterRefs	List	详情参考 attachedRouterRefs	2.1
routeEntries	List	详情参考 routeEntries	2.1

#attachedRouterRefs

名字	类型	描述	起始版本
virtualRouterVmUuid	String		2.1
routeTableUuid	String		2.1

#routeEntries

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.1
description	String	资源的详细描述	2.1
type	String	类型 允许用户添加"静态路由"、"黑洞路	2.1

名字	类型	描述	起始版本
		由"两种类型，系统会根据是否填下一跳自动判断类型"	
routeTableUuid	String	云路由路由表UUID	2.1
destination	String	目标网络地址	2.1
target	String	下一跳地址 为一个云路由设备目前可以直接到达的IP地址，如果不能直接到达，将会进行递归路由	2.1
distance	Integer	路由优先级，在最小匹配下如果有多条路由规则匹配，优先级数字小的规则将会被匹配	2.1
createDate	Timestamp	创建时间	2.1
lastOpDate	Timestamp	最后一次修改时间	2.1

SDK示例

Java SDK

```
AttachVRouterRouteTableToVRouterAction action = new AttachVRouterRouteTableToVRouterAction();
action.routeTableUuid = "9d77dce987404000a00dd64594863e32";
action.virtualRouterVmUuid = "bf7963808fa4d64b94cbbc8eb593398";
action.sessionId = "48e036e5881d41fa9070d2b6aa75c7ee";
AttachVRouterRouteTableToVRouterAction.Result res = action.call();
```

Python SDK

```
AttachVRouterRouteTableToVRouterAction action = AttachVRouterRouteTableToVRouterAction()
action.routeTableUuid = "0afcb79f56f74b1a96cce814362f861f"
action.virtualRouterVmUuid = "611126cb62094da4b6023824f0f2d0c9"
action.sessionId = "56fd37c6dff343948028d9b826ea4b10"
```

```
AttachVRouterRouteTableToVRouterAction.Result res = action.call()
```

4.3.9.9 解绑云路由路由表(DetachVRouterRouteTableFromVRouter)

从云路由设备解绑云路由路由表。

API请求

URLs

```
POST zstack/v1/vrouter-route-tables/{routeTableUuid}/detach
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "virtualRouterVmUuid": "17b4381cc9664578b496256e08638681"
  },
  "systemTags": [],
  "userTags": []
}
```

 **注：** 上述示例中systemTags、userTags字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"virtualRouterVmUuid":"50391e085c803988a3fa27e210834a16"}}' \
http://localhost:8080/zstack/v1/vrouter-route-tables/2ef92879139f36e291314bed861a0cc7/detach
```

参数列表

名字	类型	位置	描述	可选值	起始版本
routeTableUuid	String	url	云路由路由表UUID		2.1
virtualRouterVmUuid	String	body(包含在params结构中)	云路由表设备UUID		2.1

名字	类型	位置	描述	可选值	起始版本
systemTags (可选)	List	body			2.1
userTags (可选)	List	body			2.1

API返回

返回示例

```
{
  "inventory": {
    "name": "test-route-table",
    "description": "Test route table",
    "attachedRouterRefs": [],
    "routeEntries": []
  }
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	2.1
inventory	VRouterRouteTableInventory	详情参考 inventory	2.1

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.1
description	String	错误的概要描述	2.1
details	String	错误的详细信息	2.1
elaboration	String	保留字段，默认为null	2.1
opaque	LinkedHashMap	保留字段，默认为null	2.1
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.1

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.1
name	String	资源名称	2.1
description	String	资源的详细描述	2.1
createDate	Timestamp	创建时间	2.1
lastOpDate	Timestamp	最后一次修改时间	2.1
attachedRouterRefs	List	详情参考 attachedRouterRefs	2.1
routeEntries	List	详情参考 routeEntries	2.1

#attachedRouterRefs

名字	类型	描述	起始版本
virtualRouterVmUuid	String		2.1
routeTableUuid	String		2.1

#routeEntries

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.1
description	String	资源的详细描述	2.1
type	String	类型 允许用户添加"静态路由"、"黑洞路由"两种类型，系统会根据是否填下一跳自动判断类型	2.1
routeTableUuid	String	云路由路由表UUID	2.1
destination	String	目标网络地址	2.1
target	String	下一跳地址	2.1

名字	类型	描述	起始版本
		为一个云路由设备目前可以直接到达的IP地址，如果不能直接到达，将会进行递归路由	
distance	Integer	路由优先级，在最小匹配下如果有多条路由规则匹配，优先级数字小的规则将会被匹配	2.1
createDate	Timestamp	创建时间	2.1
lastOpDate	Timestamp	最后一次修改时间	2.1

SDK示例

Java SDK

```
DetachVRouterRouteTableFromVRouterAction action = new DetachVRouterRouteTableFromVRouterAction();
action.routeTableUuid = "c157be1c2fac4806bd0e1ce04087a9b5";
action.virtualRouterVmUuid = "b844d0085c544aecb9a1c49e6c0fa4e9";
action.sessionId = "7299252a24404d92a5ee621ea5a2e6df";
DetachVRouterRouteTableFromVRouterAction.Result res = action.call();
```

Python SDK

```
DetachVRouterRouteTableFromVRouterAction action = DetachVRouterRouteTableFromVRouterAction()
action.routeTableUuid = "a7bece0ef6e04218af583d0d20078ec2"
action.virtualRouterVmUuid = "06a9a546ec6f4395b46fe5704f476ca2"
action.sessionId = "6923e611467343a3ad2f4c47163ad4bd"
DetachVRouterRouteTableFromVRouterAction.Result res = action.call()
```

4.3.9.10 查询绑定关系(QueryVirtualRouterVRouterRouteTableRef)

查询云路由设备与云路由路由表绑定关系。

API请求

URLs

```
GET zstack/v1/vrouter-route-tables/virtual-router-refs
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \  
-H "Authorization: OAuth 735e8b68e40442659dac22fc30d7774b" \  
-X GET http://localhost:8080/zstack/v1/vrouter-route-tables/virtual-router-refs
```

可查询字段

运行 `zstack-cli` 命令行工具，输入 `QueryVirtualRouterVRouterRouteTableRef` 并按 Tab 键查看所有可查询字段以及可跨表查询的资源名。

API返回

返回示例

```
{  
  "inventories": [  
    {  
      "virtualRouterVmUuid": "ebec6ece12504916a12f13441282bb67",  
      "routeTableUuid": "fbf34e1fc6924ff3ba4c338435dd8e61"  
    }  
  ]  
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	2.1
inventories	List	详情参考 inventories	2.1

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.1
description	String	错误的概要描述	2.1
details	String	错误的详细信息	2.1
elaboration	String	保留字段，默认为null	2.1
opaque	LinkedHashMap	保留字段，默认为null	2.1
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.1

#inventories

名字	类型	描述	起始版本
virtualRouterVmUuid	String		2.1
routeTableUuid	String		2.1

SDK示例

Java SDK

```
QueryVirtualRouterVRouterRouteTableRefAction action = new QueryVirtualRouterVRouterRouteTableRefAction();
action.conditions = asList();
action.sessionId = "ec7b3e5724604965a1dcf3c7eb58a7c1";
QueryVirtualRouterVRouterRouteTableRefAction.Result res = action.call();
```

Python SDK

```
QueryVirtualRouterVRouterRouteTableRefAction action = QueryVirtualRouterVRouterRouteTableRefAction()
action.conditions = []
action.sessionId = "b2ca8cc84dae431e95cff3e95ace26c1"
```

```
QueryVirtualRouterVRouterRouteTableRefAction.Result res = action.call()
```

4.4 VPC相关接口

4.4.1 创建VPC云路由(CreateVpcVRouter)

API请求

URLs

```
POST zstack/v1/vpc/virtual-routers
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "name": "TestVPC",
    "virtualRouterOfferingUuid": "d9a495f730564ce58328dbfdb1d05721",
    "description": "this is a vpc for test"
  },
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"name":"TestVPC","virtualRouterOfferingUuid":"81e8c1f14a6e352884e6941c4a2de95a","description":"this is a vpc for test"}}' \
http://localhost:8080/zstack/v1/vpc/virtual-routers
```

参数列表

名字	类型	位置	描述	可选值	起始版本
name	String	body(包含在params结构中)	资源名称		2.3
virtualRouterOfferingUuid	String	body(包含在params结构中)	云路由规格		2.3

名字	类型	位置	描述	可选值	起始版本
description (可选)	String	body(包含在params结构中)	资源的详细描述		2.3
resourceUuid (可选)	String	body(包含在params结构中)	资源的UUID		2.3
systemTags (可选)	List	body	系统标签		2.3
userTags (可选)	List	body	用户标签		2.3

API返回

返回示例

```
{
  "inventory": {
    "applianceVmType": "vpcvrouter",
    "managementNetworkUuid": "a01fad3eb1d14bd787b8830e1d287ee3",
    "status": "Connected",
    "agentPort": 7272.0,
    "uuid": "4de286dcb34343a0b5dedaea96307c06",
    "name": "TestVPC",
    "description": "this is a vpc for test",
    "zoneUuid": "88634bc51a1345b98058c1827999bd75",
    "clusterUuid": "5ebca05bc3cf4d5b970ddb32b10455c0",
    "imageUuid": "490043dfcb814b119f0f6c264ed6bb1e",
    "hostUuid": "a19572ae4fcc4b7d8a090d06c99067fa",
    "lastHostUuid": "d374797d2d01495cb4dedd9e452909d0",
    "instanceOfferingUuid": "8fcc3c2b5e1b41e586ed07ffece475e4",
    "rootVolumeUuid": "ed5b232109f64f0dbe3342ffa8924d14",
    "platform": "Linux",
    "defaultL3NetworkUuid": "2aedc4cd00f8495f8fc08c991dd7be4c",
    "type": "ApplianceVm",
    "hypervisorType": "KVM",
    "memorySize": 8.589934592E9,
    "cpuNum": 4.0,
    "allocatorStrategy": "LastHostPreferredAllocatorStrategy",
    "createDate": "Nov 9, 2017 4:20:39 PM",
    "lastOpDate": "Nov 9, 2017 4:20:39 PM",
    "state": "Running",
    "vmNics": [
      {
        "uuid": "6c8b46eec225440bb7179b6c797937ed",
        "vmInstanceUuid": "4de286dcb34343a0b5dedaea96307c06",
        "usedIpUuid": "033b876b861f4b9abaf594a893c1c94f",
        "l3NetworkUuid": "49e2d31895c94bf0808372360ebcae31",
        "ip": "192.168.1.10",
        "mac": "00:0c:29:bd:99:fc",
        "netmask": "255.255.255.0",
        "gateway": "192.168.1.1",

```

```

    "deviceId": 0.0,
    "createDate": "Nov 9, 2017 4:20:39 PM",
    "lastOpDate": "Nov 9, 2017 4:20:39 PM"
  },
  "allVolumes": [
    {
      "uuid": "ed5b232109f64f0dbe3342ffa8924d14",
      "name": "Root-Volume-For-VM-4de286dcb34343a0b5dedaea96307c06",
      "primaryStorageUuid": "c3f1cbaf9b014c1d8989ff940ed5593a",
      "vmInstanceUuid": "4de286dcb34343a0b5dedaea96307c06",
      "diskOfferingUuid": "93bfd802ebea486ab4ce071967235b76",
      "rootImageUuid": "490043dfeb814b119f0f6c264ed6bb1e",
      "installPath": "/zstack_ps/rootVolumes/acct-36c27e8ff05c4780bf6d2fa65700f22e/vol-
ed5b232109f64f0dbe3342ffa8924d14/ed5b232109f64f0dbe3342ffa8924d14.qcow2",
      "type": "Root",
      "format": "qcow2",
      "size": 1.073741824E11,
      "actualSize": 2.147483648E10,
      "deviceId": 0.0,
      "state": "Enabled",
      "status": "Ready",
      "createDate": "Nov 9, 2017 4:20:39 PM",
      "lastOpDate": "Nov 9, 2017 4:20:39 PM"
    }
  ]
}

```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。 详情参考 error	2.3
inventory	VmInstanceInventory	详情参考 inventory	2.3

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.3
description	String	错误的概要描述	2.3
details	String	错误的详细信息	2.3
elaboration	String	保留字段，默认为null	2.3
opaque	LinkedHashMap	保留字段，默认为null	2.3

名字	类型	描述	起始版本
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.3

#inventory

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.3
name	String	资源名称	2.3
description	String	资源的详细描述	2.3
zoneUuid	String	区域UUID	2.3
clusterUuid	String	集群UUID	2.3
imageUuid	String	镜像UUID	2.3
hostUuid	String	物理机UUID	2.3
lastHostUuid	String	上一次运行云路由设备的物理机UUID	2.3
instanceOfferingUuid	String	计算规格UUID	2.3
rootVolumeUuid	String	根云盘UUID	2.3
platform	String	云路由设备运行平台	2.3
defaultL3NetworkUuid	String	默认三层网络UUID	2.3
type	String	云路由设备类型	2.3
hypervisorType	String	云路由设备的hypervisor类型	2.3
memorySize	Long	内存大小	2.3
cpuNum	Integer	cpu数量	2.3
cpuSpeed	Long	cpu主频	2.3
allocatorStrategy	String	分配策略	2.3
createDate	Timestamp	创建时间	2.3
lastOpDate	Timestamp	最后一次修改时间	2.3

名字	类型	描述	起始版本
state	String	云路由设备的可用状态	2.3
vmNics	List	详情参考 vmNics	2.3
allVolumes	List	详情参考 allVolumes	2.3

#vmNics

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.3
vmInstanceUuid	String	云主机UUID	2.3
I3NetworkUuid	String	三层网络UUID	2.3
ip	String	ip地址	2.3
mac	String	mac地址	2.3
netmask	String	子网掩码	2.3
gateway	String	网关	2.3
metaData	String	内部使用的保留域	2.3
deviceId	Integer	设备ID - 标识网卡在客户操作系统（guest operating system）以太网设备中顺序的整型数字。 - 例如，0通常代表eth0，1通常代表eth1。	2.3
createDate	Timestamp	创建时间	2.3
lastOpDate	Timestamp	最后一次修改时间	2.3

#allVolumes

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.3

名字	类型	描述	起始版本
name	String	资源名称	2.3
description	String	资源的详细描述	2.3
primaryStorageUuid	String	主存储UUID	2.3
vmInstanceUuid	String	云主机UUID	2.3
diskOfferingUuid	String	云盘规格UUID	2.3
rootImageUuid	String	根云盘镜像UUID	2.3
installPath	String	安装路径	2.3
type	String	云盘类型	2.3
format	String	云盘格式	2.3
size	Long	云盘虚拟容量	2.3
actualSize	Long	云盘实际容量	2.3
deviceId	Integer	设备ID	2.3
state	String	云盘的可用状态	2.3
status	String	云盘的连接状态	2.3
createDate	Timestamp	创建时间	2.3
lastOpDate	Timestamp	最后一次修改时间	2.3
isShareable	Boolean	云盘是否共享	2.3

SDK示例

Java SDK

```
CreateVpcVRouterAction action = new CreateVpcVRouterAction();
action.name = "TestVPC";
action.virtualRouterOfferingUuid = "c9cfb57035294ff4abeb400922f50d09";
action.description = "this is a vpc for test";
action.sessionId = "41652c67906143948e1fecc1054b65bd";
CreateVpcVRouterAction.Result res = action.call();
```

Python SDK

```
CreateVpcVRouterAction action = CreateVpcVRouterAction()
action.name = "TestVPC"
action.virtualRouterOfferingUuid = "a7eaafd48945481fa2d9a35032b8770b"
action.description = "this is a vpc for test"
action.sessionId = "9d1cc409dd8b47d3aa4ead7bccf875f2"
```

```
CreateVpcVRouterAction.Result res = action.call()
```

4.4.2 获取VPC云路由可加载的三层网络(GetAttachableVpcL3Network)

API请求

URLs

```
POST zstack/v1/vpc/virtual-routers/{uuid}/attachable-vpc-l3s
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {},
  "systemTags": [],
  "userTags": []
}
```



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{}}' http://localhost:8080/zstack/v1/vpc/virtual-routers/417478f0b61732408176f16615ab6d32/attachable-vpc-l3s
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID ，唯一标示该资源		2.3
systemTags (可选)	List	body	系统标签		2.3
userTags (可选)	List	body	用户标签		2.3

API返回

返回示例

```
{
  "inventories": [
    {
      "uuid": "81c29ac529163ffa9d4f246d68518dd1",
      "name": "test-l3",
      "description": "test l3",
      "type": "L3VpcNetwork",
      "zoneUuid": "aea3c47aa8273399ac8827fbc78aae67",
      "l2NetworkUuid": "cbba13d2e8273d529cfcb89c66d54fb2",
      "system": false,
      "category": "Private",
      "createDate": "Nov 14, 2017 10:20:57 PM",
      "lastOpDate": "Nov 14, 2017 10:20:57 PM",
      "ipRanges": [
        {
          "l3NetworkUuid": "cc5980d5c4173dd5b57dc1b28cdb0683",
          "name": "test ip range",
          "startIp": "100.64.0.10",
          "endIp": "100.64.0.100",
          "netmask": "255.255.255.0",
          "gateway": "100.64.0.1",
          "networkCidr": "100.64.0.0/24",
          "createDate": "Nov 14, 2017 10:20:57 PM",
          "lastOpDate": "Nov 14, 2017 10:20:57 PM"
        }
      ]
    }
  ]
}
```

名字	类型	描述	起始版本
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	2.3
inventories	List	详情参考 inventories	2.3

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.3
description	String	错误的概要描述	2.3
details	String	错误的详细信息	2.3

名字	类型	描述	起始版本
elaboration	String	保留字段，默认为null	2.3
opaque	LinkedHashMap	保留字段，默认为null	2.3
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.3

#inventories

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.3
name	String	资源名称	2.3
description	String	资源的详细描述	2.3
type	String	三层网络类型	2.3
zoneUuid	String	区域UUID	2.3
l2NetworkUuid	String	二层网络UUID	2.3
state	String	三层网络的可用状态	2.3
dnsDomain	String	DNS域	2.3
system	Boolean	是否用于系统云主机	2.3
category	String	网络类型，需要与system标签搭配使用，system为true时可设置为Public、Private	2.3
createDate	Timestamp	创建时间	2.3
lastOpDate	Timestamp	最后一次修改时间	2.3
dns	List	域名解析服务器	2.3
ipRanges	List	详情参考 ipRanges	2.3
networkServices	List	详情参考 networkServices	2.3

#ipRanges

名字	类型	描述	起始版本
uuid	String	资源的UUID，唯一标示该资源	2.3
I3NetworkUuid	String	三层网络UUID	2.3
name	String	资源名称	2.3
description	String	资源的详细描述	2.3
startIp	String	起始ip	2.3
endIp	String	结束ip	2.3
netmask	String	子网掩码	2.3
gateway	String	网关	2.3
networkCidr	String	CIDR范围	2.3
createDate	Timestamp	创建时间	2.3
lastOpDate	Timestamp	最后一次修改时间	2.3

#networkServices

名字	类型	描述	起始版本
I3NetworkUuid	String	三层网络UUID	2.3
networkServiceProviderUuid	String	网络服务提供模块UUID	2.3
networkServiceType	String	网络服务类型	2.3

SDK示例

Java SDK

```
GetAttachableVpcL3NetworkAction action = new GetAttachableVpcL3NetworkAction();
action.uuid = "417478f0b61732408176f16615ab6d32";
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c";
GetAttachableVpcL3NetworkAction.Result res = action.call();
```

Python SDK

```
GetAttachableVpcL3NetworkAction action = GetAttachableVpcL3NetworkAction()
action.uuid = "417478f0b61732408176f16615ab6d32"
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c"
```

```
GetAttachableVpcL3NetworkAction.Result res = action.call()
```

4.4.3 获取实时流量状态(GetVpcVRouterDistributedRoutingConnections)

获取VPC云路由实时流量状态。

API请求

URLs

```
GET zstack/v1/vpc/virtual-routers/{uuid}/tracked-connections
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X GET http://localhost:8080/zstack/v1/vpc/virtual-routers/f37467649dca3ee8a0ddad0b856a8df6/tracked-connections
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID，唯一标示该资源。需要开启分布式路由，才能获取到VPC云路由实时流量状态。		2.3
systemTags (可选)	List	query	系统标签		2.3
userTags (可选)	List	query	用户标签		2.3

API返回

返回示例

```
{
```

```

"inventories": {
  "192.168.31.156,192.168.105.175": {
    "sourceL2NetworkType": "L2VlanNetwork",
    "destinationL2NetworkType": "L2VlanNetwork",
    "sourceMac": "fa:bf:6e:37:c3:00",
    "destinationMac": "fa:3a:b3:ae:f4:00",
    "sourceL2NetworkVni": 3101.0,
    "destinationL2NetworkVni": 3105.0,
    "lastOpDate": "2017-12-02 15:23:44.872099448 +0800 CST m\u003d+70467.712498312",
    "status": "ZSNP_DST_SUCC"
  }
}

```

名字	类型	描述	起始版本
inventories	Map	实时流量状态集合，结构为Map - Key是通信两端的地址； - 内容为一个Map，包含：源网络和目的网络的类型、Vni，通信两端的Mac，上一次操作的时间以及目前的优化状态； - 其中ZSNP_DST_SUCC表示优化成功	2.3
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考error	2.3

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.3
description	String	错误的概要描述	2.3
details	String	错误的详细信息	2.3

名字	类型	描述	起始版本
elaboration	String	保留字段，默认为null	2.3
opaque	LinkedHashMap	保留字段，默认为null	2.3
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.3

SDK示例

Java SDK

```
GetVpcVRouterDistributedRoutingConnectionsAction action = new GetVpcVRouterDistributedRoutingConnectionsAction();
action.uuid = "f37467649dca3ee8a0ddad0b856a8df6";
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c";
GetVpcVRouterDistributedRoutingConnectionsAction.Result res = action.call();
```

Python SDK

```
GetVpcVRouterDistributedRoutingConnectionsAction action = GetVpcVRouterDistributedRoutingConnectionsAction()
action.uuid = "f37467649dca3ee8a0ddad0b856a8df6"
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c"
GetVpcVRouterDistributedRoutingConnectionsAction.Result res = action.call()
```

4.4.4 获取分布式路由是否打开(GetVpcVRouterDistributedRoutingEnabled)

获取VPC云路由分布式路由是否打开。

API请求

URLs

```
GET zstack/v1/vpc/virtual-routers/{uuid}/distributed-routing
```

Headers

```
Authorization: OAuth the-session-uuid
```

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
```

```
-X GET http://localhost:8080/zstack/v1/vpc/virtual-routers/9fab45026c2239d884db36ce9190660f/distributed-routing
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID ，唯一标示该资源		2.3
systemTags (可选)	List	query	系统标签		2.3
userTags (可选)	List	query	用户标签		2.3

API返回

返回示例

```
{
  "enabled": true
}
```

名字	类型	描述	起始版本
enabled	boolean	获取VPC云路由分布式路由是否打开	2.3
error	ErrorCode	错误码，若不 为null，则表示操作失败，操作成功时该字段 为null。 详情参考 error	2.3

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全 局唯一标识，例如SYS .1000, HOST.1001	2.3
description	String	错误的概要描述	2.3
details	String	错误的详细信息	2.3
elaboration	String	保留字段，默认为null	2.3

名字	类型	描述	起始版本
opaque	LinkedHashMap	保留字段，默认为null	2.3
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.3

SDK示例

Java SDK

```
GetVpcVRouterDistributedRoutingEnabledAction action = new GetVpcVRouterDistributedRoutingEnabledAction();
action.uuid = "9fab45026c2239d884db36ce9190660f";
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c";
GetVpcVRouterDistributedRoutingEnabledAction.Result res = action.call();
```

Python SDK

```
GetVpcVRouterDistributedRoutingEnabledAction action = GetVpcVRouterDistributedRoutingEnabledAction()
action.uuid = "9fab45026c2239d884db36ce9190660f"
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c"
GetVpcVRouterDistributedRoutingEnabledAction.Result res = action.call()
```

4.4.5 设置分布式路由开关(SetVpcVRouterDistributedRoutingEnabled)

设置VPC云路由分布式路由开关。

API请求

URLs

```
POST zstack/v1/vpc/virtual-routers/{uuid}/distributed-routing
```

Headers

```
Authorization: OAuth the-session-uuid
```

Body

```
{
  "params": {
    "stateEvent": "enable"
  },
  "systemTags": [],
  "userTags": []
}
```

}



注：上述示例中**systemTags**、**userTags**字段可以省略。列出是为了表示body中可以包含这两个字段。

Curl示例

```
curl -H "Content-Type: application/json" \
-H "Authorization: OAuth b86c9016b4f24953a9edefb53ca0678c" \
-X POST -d '{"params":{"stateEvent":"enable"}}' \
http://localhost:8080/zstack/v1/vpc/virtual-routers/fed6eb5e41d83512ba1f836b5473b4fc/distributed-routing
```

参数列表

名字	类型	位置	描述	可选值	起始版本
uuid	String	url	资源的UUID ，唯一标示该资源		2.3
stateEvent	String	body(包含在params结构中)	资源的可用状态	- enable - disable	2.3
systemTags (可选)	List	body	系统标签		2.3
userTags (可选)	List	body	用户标签		2.3

API返回

返回示例

```
{
  "enabled": true
}
```

名字	类型	描述	起始版本
enabled	boolean	设置VPC云路由分布式路由开关是否打开	2.3
error	ErrorCode	错误码，若不为null，则表示操作失败，操作成功时该字段为null。详情参考 error	2.3

#error

名字	类型	描述	起始版本
code	String	错误码号，错误的全局唯一标识，例如SYS.1000, HOST.1001	2.3
description	String	错误的概要描述	2.3
details	String	错误的详细信息	2.3
elaboration	String	保留字段，默认为null	2.3
opaque	LinkedHashMap	保留字段，默认为null	2.3
cause	ErrorCode	根错误，引发当前错误的源错误，若无原错误，该字段为null	2.3

SDK示例

Java SDK

```
SetVpcVRouterDistributedRoutingEnabledAction action = new SetVpcVRouterDistributedRoutingEnabledAction();
action.uuid = "fed6eb5e41d83512ba1f836b5473b4fc";
action.stateEvent = "enable";
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c";
SetVpcVRouterDistributedRoutingEnabledAction.Result res = action.call();
```

Python SDK

```
SetVpcVRouterDistributedRoutingEnabledAction action = SetVpcVRouterDistributedRoutingEnabledAction()
action.uuid = "fed6eb5e41d83512ba1f836b5473b4fc"
action.stateEvent = "enable"
action.sessionId = "b86c9016b4f24953a9edefb53ca0678c"
SetVpcVRouterDistributedRoutingEnabledAction.Result res = action.call()
```

术语表

区域 (Zone)

ZStack中最大的一个资源定义，包括集群、二层网络、主存储等资源。

集群 (Cluster)

一个集群是类似物理主机 (Host) 组成的逻辑组。在同一个集群中的物理主机必须安装相同的操作系统 (虚拟机管理程序, Hypervisor)，拥有相同的二层网络连接，可以访问相同的主存储。在实际的数据中心，一个集群通常对应一个机架 (Rack)。

管理节点 (Management Node)

安装系统的物理主机，提供UI管理、云平台部署功能。

计算节点 (Compute Node)

也称之为物理主机 (或物理机)，为云主机实例提供计算、网络、存储等资源的物理主机。

主存储 (Primary Storage)

用于存储云主机磁盘文件的存储服务器。支持本地存储、NFS、Ceph、FusionStor、Shared Mount Point等类型。

镜像服务器 (Backup Storage)

也称之为备份存储服务器，主要用于保存镜像模板文件。建议单独部署镜像服务器。

镜像仓库 (Image Store)

镜像服务器的一种类型，可以为正在运行的云主机快速创建镜像，高效管理云主机镜像的版本变迁以及发布，实现快速上传、下载镜像，镜像快照，以及导出镜像的操作。

云主机 (VM Instance)

运行在物理机上的虚拟机实例，具有独立的IP地址，可以访问公共网络，运行应用服务。

镜像 (Image)

云主机或云盘使用的镜像模板文件，镜像模板包括系统云盘镜像和数据云盘镜像。

云盘 (Volume)

云主机的数据盘，给云主机提供额外的存储空间，共享云盘可挂载到一个或多个云主机共同使用。

计算规格 (Instance Offering)

启动云主机涉及到的CPU数量、内存、网络设置等规格定义。

云盘规格 (Disk Offering)

创建云盘容量大小的规格定义。

二层网络 (L2 Network)

二层网络对应于一个二层广播域，进行二层相关的隔离。一般用物理网络的设备名称标识。

三层网络 (L3 Network)

云主机使用的网络配置，包括IP地址范围、网关、DNS等。

公有网络 (Public Network)

由因特网信息中心分配的公有IP地址或者可以连接到外部互联网的IP地址。

私有网络 (Private Network)

云主机连接和使用的内部网络。

L2NoVlanNetwork

物理主机的网络连接不采用Vlan设置。

L2VlanNetwork

物理主机节点的网络连接采用Vlan设置，Vlan需要在交换机端提前进行设置。

VXLAN网络池 (VXLAN Network Pool)

VXLAN网络中的 Underlay 网络，一个 VXLAN 网络池可以创建多个 VXLAN Overlay 网络（即 VXLAN 网络），这些 Overlay 网络运行在同一组 Underlay 网络设施上。

VXLAN网络 (VXLAN)

使用 VXLAN 协议封装的二层网络，单个 VXLAN 网络需从属于一个大的 VXLAN 网络池，不同 VXLAN 网络间相互二层隔离。

云路由 (vRouter)

云路由通过定制的Linux云主机来实现的多种网络服务。

安全组 (Security Group)

针对云主机进行第三层网络的防火墙控制，对IP地址、网络包类型或网络包流向等可以设置不同的安全规则。

弹性IP (EIP)

公有网络接入到私有网络的IP地址。

快照 (Snapshot)

某一个时间点上某一个磁盘的数据备份。包括自动快照和手动快照两种类型。