**[DCFabricSDN控制器]**

**北向接口文档**

**文档编号：DCFabric**SDN-Controller -V1.4

**文档名称：**开源DCFabricSDN控制器北向接口文档

**版本信息：**V1.4

**建立日期：**2014年04月13日

**文档修订记录**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 版本编号或者更改记录编号 | 修订原因 | **简要说明（变更内容和变更范围）** | 日期 | 变更人 | 批准日期 | 批准人 |
| V1.0 | 创建 | 首次创建 | 2014-04-13 | 薛秋宝 |  |  |
| V1.1 | 增加 | 增加角色配置、meter表、group表等内容 | 2014-07-20 | 薛秋宝 |  |  |
| V1.2 | 增加 | 增加与openstack对接的接口 | 2014-09-17 | 薛秋宝 |  |  |
| V1.3 | 增加 | 增加静态流表下发接口  Group表下发增加多个bucket,支持PUT消息进行修改 | 2014-11-12 | 薛秋宝 |  |  |
| V1.4 | 增加 | 增加SFabric相关接口 | 2015-05-12 | 许延伟 |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

修订原因：创建；增加；修改；删除

文档审批信息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 审批人 | 角色 | 审批日期 | 签字 | 备注 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**目录**

[1、交换机状态查询 5](#_Toc403564923)

[1.1 厂商版本信息 5](#_Toc403564924)

[1.2 IP端口信息 5](#_Toc403564925)

[2、拓扑信息查询 6](#_Toc403564926)

[2.1 网络拓扑 6](#_Toc403564927)

[2.2 主机拓扑 6](#_Toc403564928)

[3、流表配置管理 7](#_Toc403564929)

[3.1 静态流表的下发 7](#_Toc403564930)

[3.2 静态流表的查询 8](#_Toc403564931)

[3.3 动态流表的查询 8](#_Toc403564932)

[3.4 清空所有流表 8](#_Toc403564933)

[4、防火墙配置管理 8](#_Toc403564934)

[4.1 防火墙状态查询 8](#_Toc403564935)

[4.2 开启防火墙 9](#_Toc403564936)

[4.3 关闭防火墙 9](#_Toc403564937)

[4.4 添加防火墙规则 9](#_Toc403564938)

[4.5查询防火墙规则 9](#_Toc403564939)

[4.6 删除防火墙规则 9](#_Toc403564940)

[5、L3网关配置 10](#_Toc403564941)

[5.1 添加网关 10](#_Toc403564942)

[5.2 删除网关 10](#_Toc403564943)

[5.3 查询网关 10](#_Toc403564944)

[6、角色配置(只针对openflow1.3) 10](#_Toc403564945)

[6.1 角色设置 10](#_Toc403564946)

[7、meter表配置 11](#_Toc403564947)

[7.1 meter表下发 11](#_Toc403564948)

[8、group表配置 11](#_Toc403564949)

[8.1 group表下发 11](#_Toc403564950)

[9、neutron相关接口 12](#_Toc403564951)

[9.1 查询网络、子网、端口 12](#_Toc403564952)

[9.2 添加网络、子网、端口 12](#_Toc403564953)

[9.3 更新网络、子网、端口 12](#_Toc403564954)

[9.4 删除网络、子网、端口 13](#_Toc403564955)

北向接口说明：

接口方式： Restful接口

数据格式：json格式（暂不支持xml格式）

通信协议： http

请求方法： GET、POST、PUT、DELETE

通信端口: 8080（默认为8080，控制器支持可配置）

例如：<http://10.8.1.44:8080/gn/topology/links/json> GET

# 1、交换机状态查询

## 1.1 厂商版本信息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 应答消息示例 |
| 1 | /gn/core/switch/all/desc/json | GET | 获取所有交换机的的厂商、软硬件版本、openflow协议版本等信息 | 无 | 例：  "00:00:00:0c:29:3f:ae:e6": [  {  "length": 1056,  "manufacturerDescription": "Nicira, Inc.",  "hardwareDescription": "Open vSwitch",  "softwareDescription": "2.1.0",  "serialNumber": "None",  "datapathDescription": "None"  "openflow": "of1.0/of1.3.1"  }  ] |

## 1.2 IP端口信息

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 应答消息示例 |
| 1 | /gn/core/controller/switches/json | GET | 获取所有交换机的dpid、ip、端口等信息 | 无 | 例：  [  {  "dpid": " 0: 0: 0: c:29:3f:ae:d2",  "inetAddress": "/10.8.1.213:33793",  "ports": [  {  "name": "eth1",  "state": "0",  "hardwareAddress": " 0: c:29:3f:ae:d2",  "portNumber": "1",  "config": "0",  "currentFeatures": "672",  "advertisedFeatures": "687",  "supportedFeatures": "687",  "peerFeatures": "0"  }  ],  "buffers": "256",  "connectedSince": "1397552437"  }  ] |

# 2、拓扑信息查询

## 2.1 网络拓扑

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 应答消息示例 |
| 1 | /gn/topology/links/json | GET | 获取交换机的网络拓扑 | 无 | 例：  {  " 0: 0: 0: c:29:3f:ae:e6": [  {  "src-port": 1,  "dst-switch": " 0: 0: 0: c:29:3f:ae:d2",  "dst-port": 2  },  {  "src-port": 2,  "dst-switch": " 0: 0: 0: c:29:3f:ae: 4",  "dst-port": 1  }  ]  } |

## 2.2 主机拓扑

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 应答消息示例 |
| 1 | /gn/topology/hosts/json | GET | 获取交换机的主机拓扑 | total: 表示某个交换机某个端口下主机的总个数 | " 0: 0: 0: 0: 0: 0: 0: 1": [  {  "src-port": 1,  "hosts": [  {  "mac": " 0: 0: 0: 0: 0: 1",  "ip": "192.168.0.40"  },  {  "mac": " 0: 1: 0: 0: 0: 1",  "ip": "192.168.0.41"  }  ],  "total": "2"  } |

# 3、流表配置管理

## 3.1 静态流表的下发

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 请求消息示例 |
| 1 | /gn/staticflowentrypusher/json | POST | 给某个交换机添加一条静态流表项 | "**switchid**":"<xx:xx:xx:xx:xx:xx:xx:xx>",**必选项**  "**table-id**":"< uint8>",  "idle\_timeout ":"< short>",  "hard\_timeout":"< short>",  "**src-inport**":"<short>",  "**vlan-id**":"<short>"  "**src-mac**":"<xx:xx:xx:xx:xx:xx>",  "**dst-mac**": "<xx:xx:xx:xx:xx:xx>",   "**dl-type**": "<ARP *or* IPv4 or IPv6 or MPLS or VLAN>",  "**src-ip**": "<A.B.C.D/M>",  "**dst-ip**": "<A.B.C.D/M>",  "**dst-ipv6**": "<A:B:C::D>",  "**nw-proto**": "<TCP *or* UDP *or* ICMP>",  "**icmpv4\_type**":"uint8",  "**tp-src**": "<short>",  "**tp-dst**": "<short>",  “**vlan-priority**”:<uint8>  “**nw-tos**”:<uint8>  “**priority**”:<short>  "**instruction**":"<key>=<value> " //"gototable=2"  "**action**":"<key>=<value> "//"output=1"  "**action\_push**":"<key>=<value> "//["push-vlan=1", "push-mpls=1"]  "**action\_pop**":"<key>=<value> "//["pop-vlan=1", "pop-mpls=1"]  "**meter-id**":"<uint32>" //流表关联meter表  "**group-id**":”<uint32>" //流表关联group表  "mpls\_lable":":<uint32>" //mpls标签值匹配  "arp\_spa":"<uint32>" //ARP源IPv4地址  "arp\_tpa":"<uint32>"//ARP目的IPv4地址  "arp-sha":"<xx:xx:xx:xx:xx:xx>", //ARP sender MAC  "arp-tha":"<xx:xx:xx:xx:xx:xx>", //ARP target MAC | {"switchid":"00:00:00:00:00:00:00:01","dl-type":"IPv4","nw-proto":"ICMP","action":"output=1"} |

## 3.2静态流表的查询

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 应答消息示例 |
| 1 | /gn/staticflowentrypusher/list/all/json | GET | 查询所有交换机所有静态下发的流表 | 无 | {"switchid":"00:00:00:00:00:00:00:01","dl-type":"IPv4","nw-proto":"ICMP","action":"output=1"} |

## 3.3动态流表的查询

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 应答消息示例 |
| 1 | /gn/dynamicflowentrypusher/list/all/json | GET | 列出所有交换机所有动态下发的流表 | 无 | {  " 0: 0: 0: 0: 0: 0: 0: 2": [  {  "table": 0,  "idle\_timeout": 10,  "hard\_timeout": 20,  "priority": 1,  "dl\_dst": "d2: c:e4:52:b8:5d",  "actions=output": 2  }  ]  } |

## 3.4清空所有流表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 请求消息示例 |
| 1 | /gn/delallflow/json | DELETE | 清空某个交换机上所有的流表 | "**switchid**":"<xx:xx:xx:xx:xx:xx:xx:xx>"  switchid表示交换机的dpid, | {"switchid":"00:00:00:00:00:00:00:01"} |

# 4、防火墙配置管理

## 4.1 防火墙状态查询

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 应答消息示例 |
| 1 | /gn/firewall/status/json | GET | 查看控制器防火墙模块的状态 | 无 | {"result": "firewall enabled"} |

## 4.2 开启防火墙

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 应答消息示例 |
| 1 | /gn/firewall/enable/json | GET | 开启防火墙 | 无 | {"status":"success","details":"firewall running"} |

## 4.3 关闭防火墙

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 应答消息示例 |
| 1 | /gn/firewall/disable/json | GET | 关闭防火墙 | 无 | {"status":"success","details":"firewallstopped"} |

## 4.4 添加防火墙规则

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 请求消息示例 |
| 1 | /gn/firewall/rules/json | POST | 增加一条防火墙规则 | "**switchid**":"<xx:xx:xx:xx:xx:xx:xx:xx>",  "**src-inport**":"<short>",  “**vlan-id**”:<short> "**src-mac**": "<xx:xx:xx:xx:xx:xx>", "**dst-mac**": "<xx:xx:xx:xx:xx:xx>",   "**dl-type**": "<ARP *or* IPv4>",  "**src-ip**": "<A.B.C.D/M>",  "**dst-ip**": "<A.B.C.D/M>",   "**nw-proto**": "<TCP *or* UDP *or* ICMP>",   "**tp-src**": "<short>",   "**tp-dst**": "<short>",  “**vlan-priority**”:<uint8>  “**nw-tos**”:<uint8>  “**priority**”:<short>  "**action**": "<ALLOW *or* DENY>" | {"dl-type":"IPv4","nw-proto":"ICMP","action":"DENY"} |

## 4.5查询防火墙规则

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 请求消息示例 |
| 1 | /gn/firewall/rules/json | GET | 列出所有的防火墙规则 | 无 | {"ruleid":"1","dl-type":"IPv4","nw-proto":"ICMP","action":"DENY"} |

## 4.6 删除防火墙规则

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 应答消息示例 |
| 1 | /gn/firewall/rules/json | DELETE | 删除一条防火墙规则 | {"**ruleid**": "<int>"}  int表述rule id，是一个防火墙规则序号,查询防火墙规则时可以获取到 | {"status":"Rule deleted"} |

# 5、L3网关配置

## 5.1 添加网关

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 请求消息示例 |
| 1 | /gn/subnet/json | POST | 添加L3网关配置信息 | {"name":"value1","subnet":"N/M"}  name: 网关名  subnet: 网段 | {"name":"gw1","subnet":"192.168.1.1/24"} |

## 5.2 删除网关

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 请求消息示例 |
| 1 | /gn/subnet/json | DELETE | 删除指定网关的配置信息 | {"name":"value1"}  name: 网关名 | {"name":"gw1"} |

## 5.3 查询网关

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 响应消息示例 |
| **1** | /gn/subnet/list/all/json | GET | 列出控制器所有网关的配置信息 | 无 | {  "subnetConfig": [  {  "name": "gw1",  "subnet": "192.168.1.1/24"  },  {  "name": "gw2",  "subnet": "192.168.2.1/24"  }  ]  } |

# 6、角色配置(只针对openflow1.3)

## 6.1 角色设置

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 请求消息示例 |
| 1 | /gn/role/json | POST | 设置控制器的角色  （测试用接口） | "switchid":"<xx:xx:xx:xx:xx:xx:xx:xx>",  "role":"master/slave/equal",  "generation\_id":"<uint64>"  switchid: 交换机的dpid,  role: 角色类型  generation\_id: 会话id | {"switchid":"00:00:00:00:00:00:00:01", "role":"master",  "generation\_id":"10"  } |

# 7、meter表配置

## 7.1 meter表下发

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 请求消息示例 |
| 1 | /gn/meter/json | POST | 给某个交换机下发meter表配置 | "switchid":"<xx:xx:xx:xx:xx:xx:xx:xx>",  "meter-id":"<uint32>",  "flags":"kbps/pktps",  "rate":"<uint32>"  switchid交换机的dpid,  meter-id: meter表的id,  flags: 限速的方式，  rate: 速率大小 | {"switchid":"00:00:00:00:00:00:00:01",  "meter-id":"1",  "flags":"kbps/pktps",  "rate":"500"  } |

# 8、group表配置

## 8.1 group表下发

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 请求消息示例 |
| 1 | /gn/group/json | POST | 给某个交换机下发group表配置 | "switchid":"<xx:xx:xx:xx:xx:xx:xx:xx>",  "group-id":"<uint32>",  "type":"all/select/indirect/ff",  "weight":"<uint16>",  "watch\_port":"<uint32>",  "watch\_group":"<uint32>",  "outport":"<uint32>"  switchid: 交换机的dpid,  group-id: group表id  type: group表类型  outport: action转发口 | {  "switchid": "00:00:00:00:00:00:00:01",  "group-id": "1",  "type": "select",  "bucket": [  {  "weight": "0",  "watch\_port": "0",  "watch\_group": "0",  "outport": "2"  },  {  "weight": "0",  "watch\_port": "0",  "watch\_group": "0",  "outport": "3"  }  ]  } |
| 2 | /gn/group/json | PUT | 修改之前下发的group表 | 同上 | 同上 |

# 9、neutron相关接口

## 9.1 查询网络、子网、端口

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 |
| 1 | /controller/nb/v2/neutron/networks  /controller/nb/v2/neutron/networks/{netUUID} | GET | 查询neutron网络信息 |
| 2 | /controller/nb/v2/neutron/subnets  /controller/nb/v2/neutron/subnets/{subnetUUID} | GET | 查询neutron子网信息 |
| 3 | /controller/nb/v2/neutron/ports  /controller/nb/v2/neutron/ports//{portUUID} | GET | 查询neutron端口信息 |

## 9.2添加网络、子网、端口

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 |
| 1 | /controller/nb/v2/neutron/networks | POST | 增加neutron网络信息 |
| 2 | /controller/nb/v2/neutron/subnets | POST | 增加neutron子网信息 |
| 3 | /controller/nb/v2/neutron/ports | POST | 增加neutron端口信息 |

## 9.3更新网络、子网、端口

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 |
| 1 | /controller/nb/v2/neutron/networks/{netUUID} | PUT | 更新neutron网络信息 |
| 2 | /controller/nb/v2/neutron/subnets/{subnetUUID} | PUT | 更新neutron子网信息 |
| 3 | /controller/nb/v2/neutron/ports//{portUUID} | PUT | 更新neutron端口信息 |

## 9.4删除网络、子网、端口

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 |
| 1 | /controller/nb/v2/neutron/networks/{netUUID} | DELETE | 删除neutron网络信息 |
| 2 | /controller/nb/v2/neutron/subnets/{subnetUUID} | DELETE | 删除neutron子网信息 |
| 3 | /controller/nb/v2/neutron/ports//{portUUID} | DELETE | 删除neutron端口信息 |

# 9、SFabric相关接口

## 9.1 启动SFabric

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 响应消息示例 |
| 1 | /gn/fabric/setup/json | POST | 启动SFabric功能模块以及相关线程，初始化SFabric。启动Topo自适应，根据Topo变化自动更新SFabric。同时，L2功能失效。 | 无 | {  "retCode": "返回码",  "retMsg": "返回消息"  } |
| 2 | /gn/fabric/setupparts/json | POST | 根据指定的交换机的DPID，来启动SFabric功能模块以及相关线程，初始化SFabric。启动Topo自适应，根据Topo变化自动更新SFabric。DPID不在列表中的交换机，不会分配SFabric标记，只能做为中间交换机使用。同时，L2功能失效。 | 请求报文：  {  "dpidList":"3,4,6,7"  }  dpidList：分配SFabric的标记的交换机的DPID列表，用“,”来分割DPID | {  "retCode": "返回码",  "retMsg": "返回消息"  } |

## 9.2 关闭SFabric

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 响应消息示例 |
| 1 | /gn/fabric/delete/json | DELETE | 关闭SFabric功能模块以及相关线程，清理所有与SFabric功能相关的数据 | 无 | {  "retCode": "返回码",  "retMsg": "返回消息"  } |

## 9.3获取SFabric建立后的路径

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | URI | 请求方法 | 功能描述 | 参数解析 | 响应消息示例 |
| 1 | /gn/fabric/getpath /json | POST | 获取源和目的两个交换机之间的SFabric路径 | {  "srcDPID":"3",  "dstDPID":"7"  }  srcDPID：源交换机节点的DPID  dstDPID：目的交换机节点的DPID | 响应报文：  {  "path":  [  {  "DPID": "00:00:00:00:00:00:00:03",  "Port": 3  },  {  "DPID": "00:00:00:00:00:00:00:02",  "Port": 3  },  {  "DPID": "00:00:00:00:00:00:00:01",  "Port": 2  },  {  "DPID": "00:00:00:00:00:00:00:05",  "Port": 2  },  {  "DPID": "00:00:00:00:00:00:00:07",  "Port": 0  }  ],  "retCode": 0,  "retMsg": "操作成功"  } |