Load Balancing Service Management API Specification

11/11/10

Table of Contents

Management API Operations 1
Listing, Creating, Updating, and Deleting Load Balancing Clusters 1
Description 1
Sample XML Response (/clusters) 1
Sample JSON Response (/clusters) 1
Sample SML POST Request (/clusters) 1
Sample JSON POST Request (/clusters) 2
Sample XML PUT Request (/clusters/clusterid) 2
Sample JSON PUT Request (/clusters/clusterid) 2
Listing, Creating, Updating, and Deleting Load Balancing Host Machines 2
Description 2
Sample XML Response (/hosts) (/clusters/clusterId/hosts) 3
Sample JSON Response (/hosts) (/clusters/clusterId/hosts) 3
Sample XML POST Request (/hosts) 3
Sample JSON POST Request (/hosts) 4
Sample XML PUT Request (/hosts/hostid) 4
Sample JSON PUT Request (/hosts/hostid) 4
Virtual IP Management 4
Description 4
Sample XML Response (/virtualips) (/clusters/clusterId/virtualips) 4
Sample JSON Response (/virtualips) (/clusters/clusterId/virtualips) 5
Sample XML POST Request (/clusters/clusterId/virtualips) 5
Sample XML POST Request (/clusters/clusterId/virtualipsblocks) 5
Sample JSON POST Request (/clusters/clusterId/virtualips) 5
Rate Limiting a Load Balancer 5
Description 5
Sample XML Response (/loadbalancers/loadBalancerId/ratelimit) 5
Sample JSON Response (/loadbalancers/loadBalancerId/ratelimit) 6
Sample XML POST Request (/loadbalancers/loadBalancerId/ratelimit) 6
Sample JSON POST Request (/loadbalancers/loadBalancerId/ratelimit) 6
Sample XML PUT Request (/loadbalancers/loadBalancerId/ratelimit) 6
Sample JSON PUT Request (/loadbalancers/loadBalancerId/ratelimit) 6
Moving Load Balancers Between Host Machines / Adjusting Stickiness of Load Balancer Hosts 6
Description 6
Sample XML PUT Request (/loadbalancers/loadBalancerId/hosts) 7
Sample JSON PUT Request (/loadbalancers/loadBalancerId/hosts) 7
Sample XML Response (/loadbalancers/reassignhosts) 7
Sample JSON Response (/loadbalancers/reassignhosts) 7
Host machine Configuration Backup & Restoration 7
Description 7
Sample XML Response (/hosts/backups) 8
Sample JSON Response (/hosts/backups) 8
Sample XML Response (hosts/hostId/backups) 8
Sample JSON Response (hosts/hostId/backups) 8
Sample XML POST Request (/hosts/hostId/backups) 8
Sample JSON POST Request (/hosts/hostId/backups) 8
Customer List by Host/Cluster 8
Description 8

Sample XML Request ByName (/hosts/customers) (/clusters/customers)..... 8
Sample XML Request ById (/hosts/customers) (/clusters/customers) 9
Sample XML Response (/hosts/customers) (/clusters/customers) 9
Sample JSON Response (/hosts/hostId/customers) (/clusters/clusterId/customers) 9
Capacity and Availability Reports 10
Load Balancing Host Capacity Planning Board 10
Description 10
Sample XML Response (/hosts/capacityreport) (/hosts/hostId/capacityreport) 10
Sample JSON Response (/hosts/capacityreport) (/hosts/hostId/capacityreport) 10
Load Balancing Cluster Virtual IP Availability Report 10
Description 10
Sample XML Response (/virtualips/availabilityreport) (/cluster/clusterId/availabilityreport) 10
Sample JSON Response (/virtualips/availabilityreport) (/cluster/clusterId/availabilityreport) 11
Extended Customer API Operations 12
Assigning New Virtual IPs to a Load Balancer 12
Description 12
Sample XML POST Request (/loadbalancers/loadBalancerId/virtualips) 12
Sample JSON POST Request (/loadbalancers/loadBalancerId/virtualips)..... 12
Synchronizing a Load Balancer 12
Description 12
Suspending a Load Balancer 12
Description 12
Sample XML Response (/loadbalancers/loadBalancerId/suspension) 13
Sample JSON Response (/loadbalancers/loadBalancerId/suspension) 13
Sample XML POST Request (/loadbalancers/loadBalancerId/suspension) 13
Sample JSON Response (/loadbalancers/loadBalancerId/suspension) 13
Viewing Extended Details of a Load Balancer 13
Description 13
Sample XML Response (/loadbalancers/loadBalancerId/extendedview) 14
Sample JSON Response (/loadbalancers/loadBalancerId/extendedview)..... 14
Listing Account Info 14
Description 14
Sample XML Response (/account/354934/loadbalancers) 15
Sample XML Response (/account/1106/billing/5) for fetch 5 days worth of info 15

Management API Operations

Listing, Creating, Updating, and Deleting Load Balancing Clusters

Normal Response Code(s): 200

Error Response Code(s): loadBalancerManagementFault (400, 500), serviceUnavailable (503), unauthorized (401), badRequest (400), overLimit (413)

Description

The cluster operations allow for listing and manipulating clusters. In order to remove a cluster, it may not have any host machines or virtual IP's associated with it. To create a new cluster, the caller must supply their username, password, cluster name, cluster description and datacenter attributes within the request.

The GET response contains numerous attributes (numberOfLoadBalancingConfigurations, numberOfUniqueCustomers, numberOfHostMachines, and utilization) that are generated and immutable. These attributes are calculated based on configurations associated with this cluster.

Sample XML Response (/clusters)

<clusters xmlns="http://docs.rackspacecloud.com/loadbalancers/api/mgmt/v1.0"> <cluster id="1" name="Cluster Alpha" description="The best cluster, ever." datacenter="DFW" numberOfLoadBalancingConfigurations="410" numberOfUniqueCustomers="348" numberOfHostMachines="8" utilization="59%" /> <cluster id="2" name="Cluster Beta" description="The second best cluster, ever." datacenter="ORD" numberOfLoadBalancingConfigurations="580" numberOfUniqueCustomers="490" numberOfHostMachines="8" utilization="76%" /> </clusters>

Sample JSON Response (/clusters)

{"clusters": { "cluster": [{ "id": "1", "name": "Cluster Alpha", "description": "The best cluster, ever.", "dataCenter": "DFW", "numberOfLoadBalancingConfigurations": "410", "numberOfUniqueCustomers": "348", "numberOfHostMachines": "8", "utilization": "59%" }, { "id": "1", "name": "Cluster Beta", "description": "The second best cluster, ever.", "dataCenter": "DFW", "numberOfLoadBalancingConfigurations": "580", "numberOfUniqueCustomers": "490", "numberOfHostMachines": "8", "utilization": "76%" }] } }

Sample SML POST Request (/clusters)

<cluster xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" name="Cluster Gamma" description="The third best cluster, ever." dataCenter="DFW" username="my username" password="my password"/>

Sample JSON POST Request (/clusters)

{"cluster": { "name": "Cluster Gamma", "description": "The third best cluster, ever.", "datacenter": "DFW", "username" : "my username", "password" : "my password" } }

Sample XML PUT Request (/clusters/clusterid)

<cluster xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" name="Cluster Delta" description="The fourth best cluster, ever." username="aaa"/>

Sample JSON PUT Request (/clusters/clusterid)

{"cluster": { "name": "Cluster Delta", "description": "The fourth best cluster, ever.", "username": "aaa" } }

Listing, Creating, Updating, and Deleting Load Balancing Host Machines

Description

The host operations allow for retrieval of host configuration data and statistics as well as the ability to add, manipulate or delete load balancing hosts.

Creating a New Host

When creating a new host, the caller must supply the following attributes:

- name
- clusterid
- coreDeviceId
- zone
- maxConcurrentConnections
- managementIpAddress
- managementSoapInterface

Additionally, the system will attempt to validate that it can properly access the host based on the managementSoapInterface, managementSoapUsername, and managementSoapPassword. The system should automatically apply a BURN_IN status to all newly added host machines.

For an example of how a new host is created, see the "Sample XML Post Request" below.

Mutable Attributes

The following list of attributes are mutable via the PUT HTTP operation:

- name
- coreDeviceId
- status (Active, Active Target, Maintenance, Failover
- maxConcurrentConnections
- managementIpAddress
- managementSoapInterface

The GET response contains numerous attributes (utilization, numberOfLoadBalancingConfigurations, and numberOfUniqueCustomers) that are generated based on state data and are immutable. These attributes are calculated based on configurations associated with this host.

Sample XML Response (/hosts) (/clusters/clusterId/hosts)

<hosts xmlns="http://docs.rackspacecloud.com/loadbalancers/api/mgmt/v1.0"> <host id="1" name="host1" clusterId="1" coreDeviceId="14410" zone="A" status="ACTIVE_TARGET" maxConcurrentConnections="150000" managementIpAddress="10.1.1.1" managementSoapInterface="http://10.1.1.1:9090/soap" utilization="60%" numberOfLoadBalancingConfigurations="414" numberOfUniqueCustomers="141" /> <host id="2" name="host2" clusterId="1" coreDeviceId="15510" zone="B" status="ACTIVE_TARGET" maxConcurrentConnections="150000" managementIpAddress="10.1.1.2" managementSoapInterface="http://10.1.1.2:9090/soap" utilization="59%" numberOfLoadBalancingConfigurations="520" numberOfUniqueCustomers="515" /> </hosts>

Sample JSON Response (/hosts) (/clusters/clusterId/hosts)

{"hosts": { "host": [{ "id": "1", "name": "host1", "clusterId": "1", "coreDeviceId": "14410", "zone": "A", "status": "ACTIVE_TARGET", "maxConcurrentConnections": "150000", "managementIpAddress": "10.1.1.1", "managementSoapInterface": "http://10.1.1.1:9090/soap", "utilization": "60%", "numberOfLoadBalancingConfiguraions": "414", "numberOfUniqueCustomers": "141" }, { "id": "2", "name": "1", "clusterId": "1", "coreDeviceId": "15510-44140", "zone": "B", "status": "ACTIVE_TARGET", "maxConcurrentConnections": "150000", "managementIpAddress": "10.1.1.2", "managementSoapInterface": "httpd://10.1.1.2:9090/soap", "utilization": "59%", "numberOfLoadBalancingConfigurations": "520", "numberOfUniqueCustomers": "515" }] } }

Sample XML POST Request (/hosts)

<host xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" name="my-new-host" clusterId="1" zone="A" coreDeviceId="144410-44000"

status="BURN_IN" maxConcurrentConnections="150000" managementIpAddress="10.1.1.1" managementSoapInterface="http://10.1.1.1:9090/soap" />

Sample JSON POST Request (/hosts)

{"host": { "name": "my-new-host", "clusterId": "1", "zone": "A", "coreDeviceId": "144410-44000", "status": "BURN_IN", "maxConcurrentConnections": "150000", "managementIpAddress": "10.1.1.1", "managementSoapInterface": "http://10.1.1.1:9090/soap" } }

Sample XML PUT Request (/hosts/hostid)

<host xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" name="my-next-host" coreDeviceId="144410-44001" status="BURN_IN" maxConcurrentConnections="150000" managementIpAddress="10.1.1.2" managementSoapInterfac="http://10.1.1.2:9090/soap" />

Sample JSON PUT Request (/hosts/hostid)

{"host": { "name": "my-next-host", "coreDeviceId": "144410-44001", "status": "BURN_IN", "maxConcurrentConnections": "150000", "managementIpAddress": "10.1.1.2", "managementSoapInterface": "http://10.1.1.2:9090/soap" } }

Virtual IP Management

Description

The virtual IP operations allow the caller to view, create, and remove virtual IPs from an environment. Virtual IPs are automatically assigned to every newly created load balancer and can be added on-demand by a support or service administrator with proper justification. Management of the service requires blocks of IP addresses to be allocated from time-to-time to ensure availability for customers.

In order to assign a virtual IP to the environment via the POST operation, the caller must supply the address, and type attributes as part of the virtualip element. A sample POST request can be found below.

Note: In the event a virtual IP must be removed from the cluster, the DELETE operation can be used; however, to delete a virtual IP it must not have a load balancer associated to it.

Sample XML Response (/virtualips) (/clusters/clusterId/virtualips)

<virtualips> <virtualip id="411" loadBalancerId="1" clusterId="1" address="98.128.33.1" type="PUBLIC" /> <virtualip id="501" clusterId="1" address="10.41.133.4" type="SERVICENET" /> </virtualips>

Sample JSON Response (/virtualips) (/clusters/clusterId/virtualips)

{"virtualips": { "virtualip": [{ "id": "411", "loadBalancerId": "1", "clusterId": "1", "address": "98.128.33.1", "type": "PUBLIC" }, { "id": "501", "loadBalancerId": "1", "clusterId": "1", "address": "10.41.133.4", "type": "SERVICENET" }] } }

Sample XML POST Request (/clusters/clusterId/virtualips)

<virtualips> <virtualip address="98.128.33.4" type="PUBLIC" /> <virtualip address="98.128.33.5" type="PUBLIC" /> </virtualips>

Sample XML POST Request (/clusters/clusterId/virtualipsblocks)

<virtualIpBlocks type="PUBLIC" xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" xmlns:ns2="http://docs.rackspacecloud.com/loadbalancers/api/v1.0"> <virtualIpBlock value="192.168.0.0/24"/> <virtualIpBlock value="127.0.0.1/24"/> </virtualIpBlocks>

Sample JSON POST Request (/clusters/clusterId/virtualips)

{"virtualips": { "virtualip": [{ "address": "98.128.33.4", "type": "PUBLIC" }, { "address": "98.128.33.5", "type": "PUBLIC" }] } }

Rate Limiting a Load Balancer

Description

Rate limiting allows a service administrator to artificially limit the number of requests that are permitted to transit a given load balancer. This action can be taken when a particular load balancer is servicing questionable traffic, the victim of a DDoS, etc. To define a rate limit, the caller must supply a ticket id, maximum number of connections per second, and an expiration time.

Additionally, rate limit information is returned as part of the "extended details of a load balancer", which is documented below.

Sample XML Response (/loadbalancers/loadBalancerId/ratelimit)

<rateLimit ticketId="44410" expirationTime="2010-10-17T00:00:00-05:00" maxRequestsPerSecond="150" />

Sample JSON Response (/loadbalancers/loadBalancerId/ratelimit)

{"rateLimit": { "ticketId": "44410", "expirationTime": "2010-10-17 00:00:00", "maxRequestPerSecond": "150" } }

Sample XML POST Request (/loadbalancers/loadBalancerId/ratelimit)

<rateLimit xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" ticketId="44410" expirationTime="2010-10-17T00:00:00" maxRequestsPerSecond="150" />

Sample JSON POST Request (/loadbalancers/loadBalancerId/ratelimit)

{"rateLimit": { "ticketId": "44411", "expirationTime": "2010-10-17 00:00:00", "maxRequestsPerSecond": "150" } }

Sample XML PUT Request (/loadbalancers/loadBalancerId/ratelimit)

<rateLimit xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" ticketId="44411" expirationTime="2010-10-18 00:00:00" maxRequestsPerSecond="150" />

Sample JSON PUT Request (/loadbalancers/loadBalancerId/ratelimit)

{"rateLimit": { "ticketId": "44411", "expirationTime": "2010-10-17 00:00:00", "maxRequestsPerSecond": "150" } }

Moving Load Balancers Between Host Machines / Adjusting Stickiness of Load Balancer Hosts

Description

Service administrators may re-assign a load balancer to a different host machine within the same cluster. This action can be taken if capacity warrants it or if a particular configuration needs to be isolated from others within the environment. Both active and failover hosts may be changed.

Additionally, this operation allows for a service administrator to define a load balancer's host configurations as being "sticky", which will prohibit the system from automatically moving this configuration between hosts to balance the host performance.

A load balancer that is defined as being ACTIVE on multiple hosts will allow the load balancer host machines to service traffic for a single VIP across multiple systems. This is an advanced feature that should be used cautiously as it can potentially amplify DDoS and other types of malicious traffic.

Sample XML PUT Request (/loadbalancers/loadBalancerId/hosts)

<host xmlns=http://docs.rackspacecloud.com/host/api/management/v1.0"> <id="1"/> </host>

Sample JSON PUT Request (/loadbalancers/loadBalancerId/hosts)

{"host": { "host": { "id": "1", "type": "ACTIVE" } } }

Sample XML Response (/loadbalancers/reassignhosts)

<loadBalancers xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0"> <loadBalancer id="1" /> </loadBalancers>

Sample JSON Response (/loadbalancers/reassignhosts)

{"loadbalancers": { "loadbalancer": [{ "id": "1" }, { "id": "2" }] } }

Host machine Configuration Backup & Restoration

Description

The host backup and restoration tools allow for service administrators to take periodic backups of the current state of the configuration on any given host machine. These functions allow the caller to view, create new backups, purge backups, and restore any available configuration.

Non mutable Parameters:

- backupTime
- hostId
- id

Callers are not required to supply request bodies for the PUT (Restore Backup) operations.

Creating a New Backup

When creating a new backup, the caller must supply the following attributes:

- name

Sample XML Response (/hosts/backups)

<backups> <backup id="1" name="NightlyBackUp" backupTime="2010-10-17 00:00:00" hostId="1234" /> </backups>

Sample JSON Response (/hosts/backups)

{"backups": { "backup": { "id": "1", "name": "NightlyBackUp", "backupTime": "2010-10-17 00:00:00" "hostId": "1234" } } }

Sample XML Response (hosts/hostId/backups)

<backups> <backup id="1" name="great backup" backupTime="2010-10-17 00:00:00" /> </backups>

Sample JSON Response (hosts/hostId/backups)

{"backups": { "backup": { "id": "1", "name": "NightlyBackUp", "backupTime": "2010-10-17 00:00:00" } } }

Sample XML POST Request (/hosts/hostId/backups)

<backup xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" name="NightlyBackUp" />

Sample JSON POST Request (/hosts/hostId/backups)

{"backup": { "name": "NightlyBackup" } }

Customer List by Host/Cluster

Description

The generated customer list allows external services, which contain customer contact information, to query the load balancing service to determine the breakdown of customers at either the cluster or host level.

Sample XML Request ByName (/hosts/customers) (/clusters/customers)

<byIdOrName xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" xmlns:ns2="http://docs.rackspacecloud.com/loadbalancers/api/v1.0" name="newhost" />

Sample XML Request ById (/hosts/customers) (/clusters/customers)

```
<byIdOrName xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" xmlns:ns2="http://docs.rackspacecloud.com/loadbalancers/api/v1.0" id="2" />
```

Sample XML Response (/hosts/customers) (/clusters/customers)

```
<?xml version="1.0" ?> <customerList xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" xmlns:ns2="http://docs.rackspacecloud.com/loadbalancers/api/v1.0"> <customer accountId="666"> <loadBalancer id="52" name="LB.4707"/> <loadBalancer id="53" name="LB.3133"/> <loadBalancer id="54" name="LB.6882"/> <loadBalancer id="55" name="LB.5095"/> <loadBalancer id="56" name="LB.3135"/> <loadBalancer id="57" name="LB.3399"/> <loadBalancer id="58" name="LB.1341"/> <loadBalancer id="59" name="LB.2750"/> <loadBalancer id="60" name="LB.1980"/> <loadBalancer id="61" name="LB.1237"/> </customer> <customer accountId="999"> <loadBalancer id="47" name="LB.7090"/> <loadBalancer id="48" name="LB.3226"/> <loadBalancer id="49" name="LB.912"/> <loadBalancer id="50" name="LB.6697"/> <loadBalancer id="51" name="LB.6897"/> </customer> <customer accountId="354934"> <loadBalancer id="45" name="LB.5001"> <ns2:nodes> <ns2:node id="202" ip="21.248.193.163"/> <ns2:node id="204" ip="142.120.171.164"/> <ns2:node id="203" ip="231.162.141.253"/> <ns2:node id="205" ip="15.162.23.37"/> <ns2:node id="201" ip="182.254.203.171"/> </ns2:nodes> </loadBalancer> <loadBalancer id="46" name="LB.9540"> <ns2:nodes> <ns2:node id="209" ip="200.47.43.240"/> <ns2:node id="207" ip="168.80.57.76"/> <ns2:node id="206" ip="229.220.95.81"/> <ns2:node id="208" ip="114.48.233.151"/> <ns2:node id="210" ip="228.221.170.235"/> </ns2:nodes> </loadBalancer> </customer> </customerList>
```

Sample JSON Response (/hosts/hostId/customers) (/clusters/clusterId/customers)

```
{"customerlist": { "customer": { "id": "327957", "loadbalancer": [ { "id": "1", "name": "lb-1" }, { "id": "2", "name": "lb-2" } ] } } }
```


Capacity and Availability Reports

Load Balancing Host Capacity Planning Board

Description

The load balancing host capacity reports provide insight into the available capacity of a given host machine. It also supplies an estimated runway of available capacity based on historical provisioning activity (activity taking place over a period of one week).

Sample XML Response (/hosts/capacityreport) (/hosts/hostId/capacityreport)

<hostcapacityreports> <hostcapacityreport hostId="1" hostName="The Best Host Ever" totalConcurrentConnectionCapacity="150000" availableConcurrentConnections="100000" allocatedConcurrentConnections="50000" allocatedConcurrentConnectionsToday="14000" allocatedConcurrentConnectionsInPastSevenDays="25000" remainingDaysOfCapacity="9.55" /> </hostcapacityreports>

Sample JSON Response (/hosts/capacityreport) (/hosts/hostId/capacityreport)

{"hostcapacityreports": { "hostcapacityreport": { "hostId": "1", "hostName": "The Best Host Ever", "totalConcurrentConnectionCapacity": "150000", "availableConcurrentConnections": "100000", "allocatedConcurrentConnections": "50000", "allocatedConcurrentConnectionsToday": "14000", "remainingDaysOfCapacity": "9.55" } } }

Load Balancing Cluster Virtual IP Availability Report

Description

The virtual IP availability report provides insight into the availability of both public and ServiceNet IP addresses that are provisioned to a cluster. It also supplies an estimated runway of available capacity based on historical provisioning activity (activity taking place over a period of one week).

Sample XML Response (/virtualips/availabilityreport) (/cluster/clusterId/availabilityreport)

<virtualipavailabilityreports> <virtualipavailabilityreport clusterId="1" clusterName="My Little Cluster" totalPublicIpAddresses="254" totalServiceNetAddresses="254" freeAndClearPublicIpAddresses="128" freeAndClearServiceNetIpAddresses="128"


```
publicIpAddressesInHolding="14" serviceNetIpAddressesInHolding="21"
publicIpAddressesAllocatedToday="15" serviceNetIpAddressesAllocatedToday="4"
allocatedPublicIpAddressesInLastSevenDays="45"
allocatedServiceNetIpAddressesInLastSevenDays="15"
remainingDaysOfPublicIpAddresses="50.11"
remainingDaysOfServiceNetIpAddresses="14.41" /> </virtualipavailabilityreports>
```

Sample JSON Response (/virtualips/availabilityreport) (/cluster/clusterId/availabilityreport)

```
{"virtualipavailabilityreport": { "clusterId": "1", "clusterName": "My Little
Cluster", "totalPublicIpAddresses": "254", "totalServiceNetAddresses": "254",
"freeAndClearPublicIpAddresses": "128", "freeAndClearServiceNetIpAddresses": "128",
"publicIpAddressesInHolding": "14", "serviceNetIpAddressesInHolding": "21",
"publicIpAddressesAllocatedToday": "15", "serviceNetIpAddressesAllocatedToday": "4",
"allocatedPublicIpAddressesInLastSevenDays": "45",
"allocatedServiceNetIpAddressesInLastSevenDays": "15",
"remainingDaysOfPublicIpAddresses": "50.11",
"remainingDaysOfServiceNetIpAddresses": "14.41" } }
```


Extended Customer API Operations

Assigning New Virtual IPs to a Load Balancer

Description

This feature allows a user to provision a new PUBLIC or SERVICENET address to the specified load balancer. This feature is restricted because a user must justify the need for additional IP addresses due to IANA requirements. An example of a reasonable justification would be the need to have a dedicated IP address for SSL termination.

To retrieve the assigned virtual IP, the caller must perform a subsequent GET on the /loadbalancers/loadBalancerId/virtualips URI.

Sample XML POST Request (/loadbalancers/loadBalancerId/virtualips)

<virtualIp xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0 type="PUBLIC" />

Sample JSON POST Request (/loadbalancers/loadBalancerId/virtualips)

]{"virtualip": { "type": "PUBLIC" } }

Synchronizing a Load Balancer

Description

This feature allows for a caller to synchronize the specified load balancer with a particular authoritative resource. At the time of this writing the authoritative source is the database in which all load balancer configurations are stored. We may plan on adding Zeus as an authoritative source later. If a caller chooses to synchronize from the database then Zeus will be updated to match the load balancer's configuration in the database. The caller only needs to send a POST request without a body to the specified URI. This feature should only be used in the event that Zeus contains mismatching configuration information (which should hopefully never even happen).

Suspending a Load Balancer

Description

This feature allows for a caller to suspend and or unsuspend the specified load balancer. In order to suspend a load balancer, the caller must supply a reason, the ticket issuer(user)

and a ticket identifier, which can be viewed by other users with support and service admin access levels.

If a caller requests suspension details for an unsuspended load balancer, an empty suspension element (<suspension />) will be returned.

Note
While customers are not permitted to delete suspended load balancers, a user with elevated permissions may do so by issuing a DELETE request against the /loadbalancers/loadBalancerId URI.

Sample XML Response (/loadbalancers/loadBalancerId/suspension)

<suspension xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.00" reason="User suspected of fraud" ticketId="1054" user="jdoe" />

Sample JSON Response (/loadbalancers/loadBalancerId/suspension)

{"suspension": { "reason": "User suspected of fraud", "ticketId": "1054", "user": "jdoe" } }

Sample XML POST Request (/loadbalancers/loadBalancerId/suspension)

<suspension xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" reason="Non-payment" ticketId="4141" user="bob"/>

Sample JSON Response (/loadbalancers/loadBalancerId/suspension)

{"suspension": { "reason": "Non-payment", "ticketId": "4141", "user": "bob" } }

Viewing Extended Details of a Load Balancer

Description

This operation provides the detailed output for a specific load balancer configured and associated with the designated account. It differs from the customer view because it provides the total active connections, host machine information, and a rate limit (if applicable) as part of the loadbalancer element.

Note
An extended detail view is not available for a list of load balancers.

Sample XML Response (/loadbalancers/loadBalancerId/extendedview)

<loadBalancer xmlns="http://docs.rackspacecloud.com/loadbalancers/api/v1.0" id="2000" name="sample-loadbalancer" protocol="HTTP" port="80" algorithm="RANDOM" status="ACTIVE" totalActiveConnections="340"> <host id="1" type="ACTIVE" /> <currentUsage incomingTransfer="1.40" outgoingTransfer="20.14" /> <virtualIps> <virtualIp id="1000" address="206.10.10.210" type="PUBLIC" /> </virtualIps> <nodes> <node nodeId="1041" ip="10.1.1.1" port="80" condition="ENABLED" status="ONLINE" /> <node nodeId="1411" ip="10.1.1.2" port="80" condition="ENABLED" status="ONLINE" /> </nodes> <sessionPersistence persistenceType="HTTP_COOKIE"/> <connectionLimits minConnections="10" maxConnectionsFromIp="100" maxConnectionRateFromIp="50" maxConnectionRateTimer="60" /> <connectionLoggging /> <cluster name="c1.dfw1" /> <ratelimit ticketId="4100" expirationTime="2010-10-17 00:00:00" maxRequestsPerSecond="100" /> <created time="2010-06-01T12:00:00Z" /> <updated time="2010-06-01T12:00:00Z" /> </loadBalancer>

Sample JSON Response (/loadbalancers/loadBalancerId/extendedview)

{"loadbalancer": { "id": "2000", "name": "sample-loadbalancer", "protocol": "HTTP", "port": "80", "algorithm": "RANDOM", "status": "ACTIVE", "totalActiveConnections": "340", "currentUsage": { "incomingTransfer": "1.40", "outgoingTransfer": "20.14" }, "host": { "id": "1", "type": "ACTIVE" }, "virtualIps": { "virtualIp": { "id": "1000", "address": "206.10.10.2010", "type": "PUBLIC" } }, "nodes": { "node": [{ "nodeId": "1041", "ip": "10.1.1.1", "port": "9090", "condition": "ENABLED", "status": "ONLINE" }, { "nodeId": "1042", "ip": "10.1.1.2", "port": "80", "condition": "ENABLED", "status": "ONLINE" }] }, "connectionLogging":"true", "rateLimit": { "ticketId": "1123", "expirationTime": "1283277190574", "maxRequestsPerSecond": "37" }, "sessionPersistence": { "persistenceType": "HTTP_COOKIE" }, "connectionLimits": { "minConnections": "10", "maxConnectionsFromIp": "100", "maxConnectionsRateFromIp": "50", "maxConnectionsRateTimer": "60" }, "cluster": { "name": "c1.dfw1", "description": "Cluster Description" }, "created": { "time": "2010-06-01 00: 00: 00" }, "updated": { "time": "2010-06-01 00: 00: 00" } } }

Listing Account Info

Normal Response Code(s): 200

Error Response Code(s): loadBalancerManagementFault (400, 500), serviceUnavailable (503), unauthorized (401), badRequest (400), overLimit (413)

Description

Gets info on accounts

Sample XML Response (/account/354934/loadbalancers)

<?xml version="1.0" ?> <accountLoadBalancers xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" xmlns:ns2="http://docs.rackspacecloud.com/loadbalancers/api/v1.0"> <accountLoadBalancer clusterId="1" clusterName="My Cluster" loadBalancerId="3" loadBalancerName="DieAlready" protocol="HTTP" Concurrent Connections"/> <accountLoadBalancer clusterId="1" clusterName="My Cluster" loadBalancerId="5" loadBalancerName="DontDeleteMyStuff" protocol="HTTP" Concurrent Connections"/> <accountLoadBalancer clusterId="1" clusterName="My Cluster" loadBalancerId="7" loadBalancerName="WhatTheHell" protocol="HTTP" Concurrent Connections"/> <accountLoadBalancer clusterId="1" clusterName="My Cluster" loadBalancerId="8" loadBalancerName="Rats" protocol="HTTP" Concurrent Connections"/> <accountLoadBalancer clusterId="1" clusterName="My Cluster" loadBalancerId="11" loadBalancerName="WorkAlready" protocol="HTTP" Concurrent Connections"/> <accountLoadBalancer clusterId="1" clusterName="My Cluster" loadBalancerId="25" loadBalancerName="Testing123" protocol="HTTP" Concurrent Connections"/> <accountLoadBalancer clusterId="14" clusterName="YYYY Delta" loadBalancerId="45" loadBalancerName="LB.5001" protocol="LDAP" Concurrent Connections"/> <accountLoadBalancer clusterId="14" clusterName="YYYY Delta" loadBalancerId="46" loadBalancerName="LB.9540" protocol="IMAPS" Concurrent Connections"/> <accountLoadBalancer clusterId="1" clusterName="My Cluster" loadBalancerId="62" loadBalancerName="TestForFailure" protocol="HTTP" Concurrent Connections"/> <accountLoadBalancer clusterId="1" clusterName="My Cluster" loadBalancerId="63" loadBalancerName="TestForPass" protocol="HTTP" Concurrent Connections"/> </accountLoadBalancers>

Sample XML Response (/account/1106/billing/5) for fetch 5 days worth of info

<?xml version="1.0" ?> <accountBilling accountId="1106" xmlns="http://docs.rackspacecloud.com/loadbalancers/api/management/v1.0" xmlns:ns2="http://docs.rackspacecloud.com/loadbalancers/api/v1.0"> <loadBalancerBilling loadBalancerId="2" loadBalancerName="JORGE-1"> <historicalUsage endTime="2010-10-14T17:36:22-05:00" incomingTransfer="67792981391" outgoingTransfer="39363803171" startTime="2010-10-15T17:36:22-05:00"/> <historicalUsage endTime="2010-10-13T17:36:22-05:00" incomingTransfer="23429715567" outgoingTransfer="27522102645" startTime="2010-10-14T17:36:22-05:00"/> <historicalUsage endTime="2010-10-12T17:36:22-05:00" incomingTransfer="15508685570" outgoingTransfer="29388548611" startTime="2010-10-13T17:36:22-05:00"/> <historicalUsage endTime="2010-10-11T17:36:22-05:00" incomingTransfer="49975945079" outgoingTransfer="55074868570" startTime="2010-10-12T17:36:22-05:00"/> <historicalUsage endTime="2010-10-10T17:36:22-05:00" incomingTransfer="45605244326" outgoingTransfer="43370190617" startTime="2010-10-11T17:36:22-05:00"/> <historicalUsage endTime="2010-10-09T17:36:22-05:00" incomingTransfer="39903649162" outgoingTransfer="23770260692" startTime="2010-10-10T17:36:22-05:00"/> </loadBalancerBilling> <loadBalancerBilling loadBalancerId="9" loadBalancerName="JORGE-2"> <historicalUsage endTime="2010-10-14T17:36:22-05:00" incomingTransfer="11079393993"


```
outgoingTransfer="59238355558"                startTime="2010-10-15T17:36:22-05:00"/>
<historicalUsage                                endTime="2010-10-13T17:36:22-05:00"
incomingTransfer="46272183689"                        outgoingTransfer="65643291367"
startTime="2010-10-14T17:36:22-05:00"/>                                <historicalUsage
endTime="2010-10-12T17:36:22-05:00"                 incomingTransfer="58595481935"
outgoingTransfer="33592995296"                startTime="2010-10-13T17:36:22-05:00"/>
<historicalUsage                                endTime="2010-10-11T17:36:22-05:00"
incomingTransfer="45913967625"                        outgoingTransfer="25494936095"
startTime="2010-10-12T17:36:22-05:00"/>                                <historicalUsage
endTime="2010-10-10T17:36:22-05:00"                 incomingTransfer="64363283540"
outgoingTransfer="43166334902"                startTime="2010-10-11T17:36:22-05:00"/>
<historicalUsage                                endTime="2010-10-09T17:36:22-05:00"
incomingTransfer="23504069748"                        outgoingTransfer="23423742286"
startTime="2010-10-10T17:36:22-05:00"/> </loadBalancerBilling> </accountBilling>
```

