CloudPlatform

(powered by Apache CloudStack)

Developer’s Guide

For Version 3.0

Revised October 24, 2012 4:46 PM Pacific

Contents

Introduction

The CloudPlatform™ Web Services Query HTTP API is loosely based on the REST architecture and allows developers to create new management solutions or integrate existing systems with CloudPlatform. It supports POST/GET requests and returns both XML and JSON response formats.

Roles

The CloudPlatform API supports three access roles.

- **Root Admin**. Access to all features of the cloud, including both virtual and physical resource management.
- **Domain Admin**. Access to only the virtual resources of the clouds that belong to the administrator's domain.
- **User**. Access to only the features that allow management of the user's virtual instances, storage, and network.

API Reference Documentation

Find the CloudPlatform API Reference for your particular 3.0.x release at the following links. Choose the API documentation link that applies to the user role that will be issuing the API calls: root admin, domain admin, or user.

3.0.5

- http://download.cloud.com/releases/3.0.5/api_3.0.5/TOC_Root_Admin.html
- http://download.cloud.com/releases/3.0.5/api_3.0.5/TOC_Domain_Admin.html
- http://download.cloud.com/releases/3.0.5/api_3.0.5/TOC_User.html

3.0.3 - 3.0.4

- http://download.cloud.com/releases/3.0.3/api_3.0.3/TOC_Root_Admin.html
- http://download.cloud.com/releases/3.0.3/api_3.0.3/TOC_Domain_Admin.html
- http://download.cloud.com/releases/3.0.3/api_3.0.3/TOC_User.html

3.0.0 - 3.0.2

- http://download.cloud.com/releases/3.0.0/api_3.0.0/TOC_Root_Admin.html
- http://download.cloud.com/releases/3.0.0/api_3.0.0/TOC_Domain_Admin.html
- http://download.cloud.com/releases/3.0.0/api_3.0.0/TOC_User.html

Getting Started

To get started using the CloudPlatform API, you should have the following:

- URL of the CloudPlatform server you wish to integrate with.
- Both the API Key and Secret Key for an account. This should have been generated by the administrator of the cloud instance and given to you.
- Familiarity with HTTP GET/POST and query strings.
- Knowledge of either XML or JSON.
- Knowledge of a programming language that can generate HTTP requests; for example, Java or PHP.

Making API Requests

All CloudPlatform API requests are submitted in the form of a HTTP GET/POST with an associated command and any parameters. A request is composed of the following whether in HTTP or HTTPS:

- CloudPlatform API URL: This is the web services API entry point (for example, http://www.cloud.com:8080/client/api)
- Command: The web services command you wish to execute, such as start a virtual machine or create a disk volume
- Parameters: Any additional required or optional parameters for the command

A sample API GET request looks like the following:

```
http://localhost:8080/client/api?command=deployVirtualMachine&serviceOfferingId=1&disk
OfferingId=1&templateId=2&zoneId=4&apiKey=miVr6X7u6bN_sdahOBpjNejPgEsT35eXq-
jB8CG20YI3yaxXcgpyuaIRmFI_EJTVwZ0nUkkJbPmY3y2bciKwFQ&signature
=Lxx1DM40AjcXU%2FcaiK8RAP0O1hU%3D
```

Or in a more readable format:

```
1.  http://localhost:8080/client/api
2.  ?command=deployVirtualMachine
3.  &serviceOfferingId=1
4.  &diskOfferingId=1
5.  &templateId=2
6.  &zoneId=4
7.  &apiKey=miVr6X7u6bN_sdahOBpjNejPgEsT35eXqjB8CG20YI3yaxXcgpyuaIRmFI_EJTVwZ0nUkkJbPmY
    3y2bciKwFQ
8.  &signature=Lxx1DM40AjcXU%2FcaiK8RAP0O1hU%3D
```

The first line is the CloudPlatform API URL. This is the Cloud instance you wish to interact with.

The second line refers to the command you wish to execute. In our example, we are attempting to deploy a fresh new virtual machine. It is preceded by a (?) to separate itself from the CloudPlatform API URL.

Lines 3-6 are the parameters for this given command. To see the command and its request parameters, please refer to the appropriate section in the CloudPlatform API documentation. Each parameter field-value pair (field=value) is preceded by an ampersand character (&).

Line 7 is the user API Key that uniquely identifies the account. See Signing API Requests on page 8.

Line 8 is the signature hash created to authenticate the user account executing the API command. See Signing API Requests on page 8.

Enabling API Call Expiration

You can set an expiry timestamp on API calls to prevent replay attacks over non-secure channels, such as HTTP. The server tracks the expiry timestamp you have specified and rejects all the subsequent API requests that come in after this validity period.

To enable this feature, add the following parameters to the API request:

- **signatureVersion=3**: If the signatureVersion parameter is missing or is not equal to 3, the expires parameter is ignored in the API request.
- **expires=YYYY-MM-DDThh:mm:ssZ**: Specifies the date and time at which the signature included in the request is expired. The timestamp is expressed in the YYYY-MM-DDThh:mm:ssZ format, as specified in the ISO 8601 standard.

For example:

```
expires=2011-10-10T12:00:00+0530
```

A sample API request with expiration is given below:

```
http://<IP Address>:8080/client/api?command=listZones&signatureVersion=3&expires=2011-
10-10T12:00:00+0530&apiKey=miVr6X7u6bN_sdahOBpjNejPgEsT35eXq-
jB8CG20YI3yaxXcgpyuaIRmFI_EJTVwZ0nUkkJbPmY3y2bciKwFQ&signature=Lxx1DM40AjcXU%2FcaiK8RA
P0O1hU%3D
```

Signing API Requests

Whether you access the CloudPlatform API with HTTP or HTTPS, it must still be signed so that CloudPlatform can verify the caller has been authenticated and authorized to execute the command. Make sure that you have both the API Key and Secret Key provided by the CloudPlatform administrator for your account before proceeding with the signing process.

To show how to sign a request, we will re-use the previous example.

```
http://localhost:8080/client/api?command=deployVirtualMachine&serviceOfferingId=1&disk
OfferingId=1&templateId=2&zoneId=4&apiKey=miVr6X7u6bN_sdahOBpjNejPgEsT35eXq-
jB8CG20YI3yaxXcgpyuaIRmFI_EJTVwZ0nUkkJbPmY3y2bciKwFQ&signature
=Lxx1DM40AjcXU%2FcaiK8RAP0O1hU%3D
```

Breaking this down, we have several distinct parts to this URL.

- **Base URL:** This is the base URL to the CloudPlatform Management Server.

```
http://localhost:8080
```

- **API Path:** This is the path to the API Servlet that processes the incoming requests.

```
/client/api?
```

- **Command String:** This part of the query string comprises of the command, its parameters, and the API Key that identifies the account.

NOTE: As with all query string parameters of field-value pairs, the "field" component is case insensitive while all "value" values are case sensitive.

```
command=deployVirtualMachine&serviceOfferingId=1&diskOfferingId=1&templateId=2&zoneId=
4&apiKey=miVr6X7u6bN_sdahOBpjNejPgEsT35eXq-
jB8CG20YI3yaxXcgpyuaIRmFI_EJTVwZ0nUkkJbPmY3y2bciKwFQ
```

- **Signature:** This is the hashed signature of the Base URL that is generated using a combination of the user's Secret Key and the HMAC SHA-1 hashing algorithm.

```
&signature=Lxx1DM40AjcXU%2FcaiK8RAP0O1hU%3D
```

Every API request has the format Base URL+API Path+Command String+Signature.

To generate the signature.

1. For each field-value pair (as separated by a '&') in the Command String, URL encode each value so that it can be safely sent via HTTP GET.

   NOTE: Make sure all spaces are encoded as "%20" rather than "+".

2. Lower case the entire Command String and sort it alphabetically via the field for each field-value pair. The result of this step would look like the following.

```
apikey=mivr6x7u6bn_sdahobpjnejpgest35exq-
jb8cg20yi3yaxxcgpyuairmfi_ejtvwz0nukkjbpmy3y2bcikwfq&command=deployvirtualmachine&disk
offeringid=1&serviceofferingid=1&templateid=2&zoneid=4
```

3. Take the sorted Command String and run it through the HMAC SHA-1 hashing algorithm (most programming languages offer a utility method to do this) with the user's Secret Key. Base64 encode the resulting byte array in UTF-8 so that it can be safely transmitted via HTTP. The final string produced after Base64 encoding should be "Lxx1DM40AjcXU%2FcaiK8RAP0O1hU%3D".

By reconstructing the final URL in the format Base URL+API Path+Command String+Signature, the final URL should look like:

```
http://localhost:8080/client/api?command=deployVirtualMachine&serviceOfferingId=1&disk
OfferingId=1&templateId=2&zoneId=4&apiKey=miVr6X7u6bN_sdahOBpjNejPgEsT35eXq-
jB8CG20YI3yaxXcgpyuaIRmFI_EJTVwZ0nUkkJbPmY3y2bciKwFQ&signature
=Lxx1DM40AjcXU%2FcaiK8RAP0O1hU%3D
```

Responses

This section describes what to expect with the responses to your API requests.

Response Formats: XML and JSON

CloudPlatform supports two formats as the response to an API call. The default response is XML. If you would like the response to be in JSON, add `&response=json` to the Command String.

Sample XML Response:

```
<listipaddressesresponse>
  <allocatedipaddress>
    <ipaddress>192.168.10.141</ipaddress>
    <allocated>2009-09-18T13:16:10-0700</allocated>
    <zoneid>4</zoneid>
    <zonename>WC</zonename>
    <issourcenat>true</issourcenat>
  </allocatedipaddress>
</listipaddressesresponse>
```

Sample JSON Response:

```
{ "listipaddressesresponse" :
  { "allocatedipaddress" :
    [
      {
        "ipaddress" : "192.168.10.141",
        "allocated" : "2009-09-18T13:16:10-0700",
        "zoneid" : "4",
        "zonename" : "WC",
        "issourcenat" : "true"
      }
    ]
  }
}
```

Maximum Result Pages Returned

For each cloud, there is a default upper limit on the number of results that any API command will return in a single page. This is to help prevent overloading the cloud servers and prevent DOS attacks. For example, if the page size limit is 500 and a command returns 10,000 results, the command will return 20 pages.

The default page size limit can be different for each cloud. It is set in the global configuration parameter default.page.size. If your cloud has many users with lots of VMs, you might need to increase the value of this parameter. At the same time, be careful not to set it so high that your site can be taken down by an enormous return from an API call. For more information about how to set global configuration parameters, see "Describe Your Deployment" in the Installation Guide.

To decrease the page size limit for an individual API command, override the global setting with the page and pagesize parameters, which are available in any list* command (listCapabilities, listDiskOfferings, etc.).

- Both parameters must be specified together.
- The value of the pagesize parameter must be smaller than the value of default.page.size. That is, you can not increase the number of possible items in a result page, only decrease it.

For syntax information on the list* commands, see the API Reference.

Error Handling

If an error occurs while processing an API request, the appropriate response in the format specified is returned. Each error response consists of an error code and an error text describing what possibly can go wrong. For an example error response, see page 13.

An HTTP error code of 401 is always returned if API request was rejected due to bad signatures, missing API Keys, or the user simply did not have the permissions to execute the command.

Asynchronous Commands

Starting in CloudStack 2.x, the notion of asynchronous commands has been introduced. Commands are designated as asynchronous when they can potentially take a long period of time to complete such as creating a snapshot or disk volume. They differ from synchronous commands by the following:

- They are identified in the API Reference by an (A).
- They will immediately return a job ID to refer to the job that will be responsible in processing the command.
- If executed as a "create" resource command, it will return the resource ID as well as the job ID.
- You can periodically check the status of the job by making a simple API call to the command, queryAsyncJobResult and passing in the job ID.

Job Status

The key to using an asynchronous command is the job ID that is returned immediately once the command has been executed. With the job ID, you can periodically check the job status by making calls to queryAsyncJobResult command. The command will return three possible job status integer values:

- 0 - Job is still in progress. Continue to periodically poll for any status changes.
- 1 - Job has successfully completed. The job will return any successful response values associated with command that was originally executed.
- 2 - Job has failed to complete. Please check the <jobresultcode> tag for failure reason code and <jobresult> for the failure reason.

Example

The following shows an example of using an asynchronous command. Assume the API command:

```
command=deployVirtualMachine&zoneId=1&serviceOfferingId=1&diskOfferingId=1&templateId=
1
```

CloudPlatform will immediately return a job ID and any other additional data.

```
<deployvirtualmachineresponse>
  <jobid>1</jobid>
  <id>100</id>
</deployvirtualmachineresponse>
```

Using the job ID, you can periodically poll for the results by using the queryAsyncJobResult command.

```
command=queryAsyncJobResult&jobId=1
```

Three possible results could come from this query.

Job is still pending:

```
<queryasyncjobresult>
  <jobid>1</jobid>
  <jobstatus>0</jobstatus>
  <jobprocstatus>1</jobprocstatus>
<queryasyncjobresult>
```

Job has succeeded:

```
<queryasyncjobresultresponse cloud-stack-version="3.0.1.6">
  <jobid>1</jobid>
  <jobstatus>1</jobstatus>
  <jobprocstatus>0</jobprocstatus>
  <jobresultcode>0</jobresultcode>
  <jobresulttype>object</jobresulttype>
  <jobresult>
    <virtualmachine>
    <id>450</id>
    <name>i-2-450-VM</name>
    <displayname>i-2-450-VM</displayname>
    <account>admin</account>
    <domainid>1</domainid>
    <domain>ROOT</domain>
    <created>2011-03-10T18:20:25-0800</created>
    <state>Running</state>
    <haenable>false</haenable>
    <zoneid>1</zoneid>
    <zonename>San Jose 1</zonename>
    <hostid>2</hostid>
    <hostname>905-13.sjc.lab.vmops.com</hostname>
    <templateid>1</templateid>
    <templatename>CentOS 5.3 64bit LAMP</templatename>
    <templatedisplaytext>CentOS 5.3 64bit LAMP</templatedisplaytext>
    <passwordenabled>false</passwordenabled>
    <serviceofferingid>1</serviceofferingid>
    <serviceofferingname>Small Instance</serviceofferingname>
    <cpunumber>1</cpunumber>
    <cpuspeed>500</cpuspeed>
    <memory>512</memory>
    <guestosid>12</guestosid>
    <rootdeviceid>0</rootdeviceid>
    <rootdevicetype>NetworkFilesystem</rootdevicetype>
    <nic>
      <id>561</id>
      <networkid>205</networkid>
      <netmask>255.255.255.0</netmask>
      <gateway>10.1.1.1</gateway>
      <ipaddress>10.1.1.225</ipaddress>
      <isolationuri>vlan://295</isolationuri>
      <broadcasturi>vlan://295</broadcasturi>
      <traffictype>Guest</traffictype>
      <type>Virtual</type>
      <isdefault>true</isdefault>
    </nic>
    <hypervisor>XenServer</hypervisor>
    </virtualmachine>
  </jobresult>
</queryasyncjobresultresponse>
```

Job has failed:

```
<queryasyncjobresult>
  <jobid>1</jobid>
  <jobstatus>2</jobstatus>
  <jobprocstatus>0</jobprocstatus>
  <jobresultcode>551</jobresultcode>
  <jobresulttype>text</jobresulttype>
  <jobresult>
    Unable to deploy virtual machine id = 100 due to not enough capacity
  </jobresult>
<queryasyncjobresult>
```

Working With Usage Data

The Usage Server provides aggregated usage records which you can use to create billing integration for the CloudPlatform platform. The Usage Server works by taking data from the events log and creating summary usage records that you can access using the listUsageRecords API call.

The usage records show the amount of resources, such as VM run time or template storage space, consumed by guest instances. In the special case of bare metal instances, no template storage resources are consumed, but records showing zero usage are still included in the Usage Server's output.

The Usage Server runs at least once per day. It can be configured to run multiple times per day. Its behavior is controlled by configuration settings as described in the CloudPlatform Administration Guide.

Usage Record Format

Virtual Machine Usage Record Format

For running and allocated virtual machine usage, the following fields exist in a usage record.

- **account** – name of the account
- **accountid** – ID of the account
- **domainid** – ID of the domain in which this account resides
- **zoneid** – Zone where the usage occurred
- **description** – A string describing what the usage record is tracking
- **usage** – String representation of the usage, including the units of usage (e.g. 'Hrs' for VM running time)
- **usagetype** – A number representing the usage type (see Usage Types on page 20)
- **rawusage** – A number representing the actual usage in hours
- **virtualMachineId** – The ID of the virtual machine
- **name** – The name of the virtual machine
- **offeringid** – The ID of the service offering
- **templateid** – The ID of the template or the ID of the parent template. The parent template value is present when the current template was created from a volume.
- **usageid** – Virtual machine ID
- **type** – Hypervisor

- **startdate, enddate** – The range of time for which the usage is aggregated; see Dates in the Usage Record on page 21

Network Usage Record Format

For network usage (bytes sent/received), the following fields exist in a usage record.

- **account** – name of the account
- **accountid** – ID of the account
- **domainid** – ID of the domain in which this account resides
- **networkid** – ID of the network for which the traffic was generated
- **zoneid** – Zone where the usage occurred
- **description** – A string describing what the usage record is tracking
- **usagetype** – A number representing the usage type (see Usage Types on page 20)
- **rawusage** – A number representing the actual usage in bytes
- **usageid** – Device ID (virtual router ID or external device ID)
- **type** – Device type (domain router, external load balancer, etc.)
- **startdate, enddate** – The range of time for which the usage is aggregated; see Dates in the Usage Record on page 21

IP Address Usage Record Format

For IP address usage the following fields exist in a usage record.

- **account** – name of the account
- **accountid** – ID of the account
- **domainid** – the id of the domain in which this account resides
- **zoneid** – the zone where the usage occurred
- **description** – a string describing what the usage record is tracking
- **usage** – String representation of the usage, including the units of usage
- **usagetype** – a number representing the usage type (see Usage Types on page 20)
- **rawusage** – A number representing the actual usage in hours

- **usageid** – IP address ID
- **startdate, enddate** – The range of time for which the usage is aggregated; see Dates in the Usage Record on page 21
- **issourcenat** – Whether source NAT is enabled for the IP address
- **iselastic** - True if the IP address is elastic.

Disk Volume Usage Record Format

For disk volumes, the following fields exist in a usage record.

- **account** – name of the account
- **accountid** – ID of the account
- **domainid** – ID of the domain in which this account resides
- **zoneid** – Zone where the usage occurred
- **description** – A string describing what the usage record is tracking
- **usage** – String representation of the usage, including the units of usage ('Hrs' for hours)
- **usagetype** – A number representing the usage type (see Usage Types on page 20)
- **rawusage** – A number representing the actual usage in hours
- **offeringid** – The ID of the disk offering
- **usageid** – The volume ID
- **templateid** – ROOT template ID
- **size** – The amount of storage allocated
- **startdate, enddate** – The range of time for which the usage is aggregated; see Dates in the Usage Record on page 21

Template, ISO, and Snapshot Usage Record Format

- **account** – name of the account
- **accountid** – ID of the account
- **domainid** – ID of the domain in which this account resides
- **zoneid** – Zone where the usage occurred

- **description** – A string describing what the usage record is tracking
- **usage** – String representation of the usage, including the units of usage ('Hrs' for hours)
- **usagetype** – A number representing the usage type (see Usage Types on page 20)
- **rawusage** – A number representing the actual usage in hours
- **templateid** – Included only for templates (usage type 7). Source template ID.
- **usageid** – ID of the template, ISO, or snapshot
- **size** – Size of the template, ISO, or snapshot
- **startdate, enddate** – The range of time for which the usage is aggregated; see Dates in the Usage Record on page 21

Load Balancer Policy or Port Forwarding Rule Usage Record Format

- **account** – name of the account
- **accountid** – ID of the account
- **domainid** – ID of the domain in which this account resides
- **zoneid** – Zone where the usage occurred
- **description** – A string describing what the usage record is tracking
- **usage** – String representation of the usage, including the units of usage ('Hrs' for hours)
- **usageid** – ID of the load balancer policy or port forwarding rule
- **usagetype** – A number representing the usage type (see Usage Types on page 20)
- **rawusage** – A number representing the actual usage in hours
- **startdate, enddate** – The range of time for which the usage is aggregated; see Dates in the Usage Record on page 21

Network Offering Usage Record Format

- **account** – name of the account
- **accountid** – ID of the account
- **domainid** – ID of the domain in which this account resides
- **zoneid** – Zone where the usage occurred

- **description** – A string describing what the usage record is tracking
- **usage** – String representation of the usage, including the units of usage ('Hrs' for hours)
- **usagetype** – A number representing the usage type (see Usage Types on page 20)
- **rawusage** – A number representing the actual usage in hours
- **offeringid** – Network offering ID
- **virtualMachineId** – The ID of the virtual machine
- **isdefault** – True if the default NIC uses this network offering
- **startdate, enddate** – The range of time for which the usage is aggregated; see Dates in the Usage Record on page 21

VPN User Usage Record Format

- **account** – name of the account
- **accountid** – ID of the account
- **domainid** – ID of the domain in which this account resides
- **zoneid** – Zone where the usage occurred
- **description** – A string describing what the usage record is tracking
- **usage** – String representation of the usage, including the units of usage ('Hrs' for hours)
- **usageid** – VPN user ID
- **usagetype** – A number representing the usage type (see Usage Types on page 20)
- **rawusage** – A number representing the actual usage in hours
- **startdate, enddate** – The range of time for which the usage is aggregated; see Dates in the Usage Record on page 21

Usage Types

The following table shows all usage types in CloudPlatform.

Type ID	Type Name	Description
1	RUNNING_VM	Tracks the total running time of a VM per usage record period. If the VM is upgraded during the usage period, you will get a separate Usage Record for the new upgraded VM.
2	ALLOCATED_VM	Tracks the total time the VM has been created to the time when it has been destroyed. This usage type is also useful in determining usage for specific templates such as Windows-based templates.
3	IP_ADDRESS	Tracks the public IP address owned by the account.
4	NETWORK_BYTES_SENT	Tracks the total number of bytes sent by all the VMs for an account. Cloud.com does not currently track network traffic per VM.
5	NETWORK_BYTES_RECEIVED	Tracks the total number of bytes received by all the VMs for an account. Cloud.com does not currently track network traffic per VM.
6	VOLUME	Tracks the total time a disk volume has been created to the time when it has been destroyed.
7	TEMPLATE	Tracks the total time a template (either created from a snapshot or uploaded to the cloud) has been created to the time it has been destroyed. The size of the template is also returned.
8	ISO	Tracks the total time an ISO has been uploaded to the time it has been removed from the cloud. The size of the ISO is also returned.
9	SNAPSHOT	Tracks the total time from when a snapshot has been created to the time it has been destroyed.
11	LOAD_BALANCER_POLICY	Tracks the total time a load balancer policy has been created to the time it has been removed. Cloud.com does not track whether a VM has been assigned to a policy.

12	PORT_FORWARDING_RULE	Tracks the time from when a port forwarding rule was created until the time it was removed.
13	NETWORK_OFFERING	The time from when a network offering was assigned to a VM until it is removed.
14	VPN_USERS	The time from when a VPN user is created until it is removed.

Example response from listUsageRecords

```
<listusagerecordsresponse>
  <count>1816</count>
  <usagerecord>
    <account>user5</account>
    <accountid>10004</accountid>
    <domainid>1</domainid>
    <zoneid>1</zoneid>
    <description>
      i-3-4-WC running time (ServiceOffering: 1) (Template: 3)
    </description>
    <usage>2.95288 Hrs</usage>
    <usagetype>1</usagetype>
    <rawusage>2.95288</rawusage>
    <virtualmachineid>4</virtualmachineid>
    <name>i-3-4-WC</name>
    <offeringid>1</offeringid>
    <templateid>3</templateid>
    <usageid>245554</usageid>
    <type>XenServer</type>
    <startdate>2009-09-15T00:00:00-0700</startdate>
    <enddate>2009-09-18T16:14:26-0700</enddate>
  </usagerecord>

   … (1,815 more usage records)
</listusagerecordsresponse>
```

Dates in the Usage Record

Usage records include a start date and an end date. These dates define the period of time for which the raw usage number was calculated. If daily aggregation is used, the start date is midnight on the day in question and the end date is 23:59:59 on the day in question (with one exception; see below). A virtual machine could have been deployed at noon on that day, stopped at 6pm on that day, then started up again at 11pm. When usage is calculated on that day, there will be 7 hours of running VM usage (usage type 1) and 12 hours of allocated VM usage (usage type 2). If the same virtual machine runs for the entire next day, there will 24 hours of both running VM usage (type 1) and allocated VM usage (type 2).

Note: The start date is not the time a virtual machine was started, and the end date is not the time when a virtual machine was stopped. The start and end dates give the time range within which usage was calculated.

For network usage, the start date and end date again define the range in which the number of bytes transferred was calculated. If a user downloads 10 MB and uploads 1 MB in one day, there will be two records, one showing the 10 megabytes received and one showing the 1 megabyte sent.

There is one case where the start date and end date do not correspond to midnight and 11:59:59pm when daily aggregation is used. This occurs only for network usage records. When the usage server has more than one day's worth of unprocessed data, the old data will be included in the aggregation period. The start date in the usage record will show the date and time of the earliest event. For other types of usage, such as IP addresses and VMs, the old unprocessed data is not included in daily aggregation.

Event Types

The following is the full list of event types. The current events can be found by calling listEvents.

VM.CREATE
VM.DESTROY
VM.START
VM.STOP
VM.REBOOT
VM.UPGRADE
VM.RESETPASSWORD
ROUTER.CREATE
ROUTER.DESTROY
ROUTER.START
ROUTER.STOP
ROUTER.REBOOT
ROUTER.HA
PROXY.CREATE
PROXY.DESTROY
PROXY.START
PROXY.STOP
PROXY.REBOOT
PROXY.HA
VNC.CONNECT
VNC.DISCONNECT
NET.IPASSIGN
NET.IPRELEASE
NET.RULEADD
NET.RULEDELETE
NET.RULEMODIFY
NETWORK.CREATE
NETWORK.DELETE
LB.ASSIGN.TO.RULE
LB.REMOVE.FROM.RULE
LB.CREATE
LB.DELETE
LB.UPDATE
USER.LOGIN
USER.LOGOUT
USER.CREATE
USER.DELETE
USER.UPDATE
USER.DISABLE
TEMPLATE.CREATE
TEMPLATE.DELETE
TEMPLATE.UPDATE
TEMPLATE.COPY
TEMPLATE.DOWNLOAD.START
TEMPLATE.DOWNLOAD.SUCCESS
TEMPLATE.DOWNLOAD.FAILED
TEMPLATE.EXTRACT
TEMPLATE.UPLOAD
TEMPLATE.CLEANUP
VOLUME.CREATE
VOLUME.DELETE
VOLUME.ATTACH
VOLUME.DETACH
VOLUME.EXTRACT
VOLUME.UPLOAD
SERVICEOFFERING.CREATE
SERVICEOFFERING.UPDATE
SERVICEOFFERING.DELETE
DOMAIN.CREATE
DOMAIN.DELETE
DOMAIN.UPDATE
SNAPSHOT.CREATE
SNAPSHOT.DELETE
SNAPSHOTPOLICY.CREATE
SNAPSHOTPOLICY.UPDATE
SNAPSHOTPOLICY.DELETE
ISO.CREATE
ISO.DELETE
ISO.COPY
ISO.ATTACH
ISO.DETACH
ISO.EXTRACT
ISO.UPLOAD
SERVICE.OFFERING.CREATE
SERVICE.OFFERING.EDIT
SERVICE.OFFERING.DELETE
DISK.OFFERING.CREATE
DISK.OFFERING.EDIT
DISK.OFFERING.DELETE
NETWORK.OFFERING.CREATE
NETWORK.OFFERING.EDIT
NETWORK.OFFERING.DELETE
POD.CREATE
POD.EDIT
POD.DELETE
ZONE.CREATE
ZONE.EDIT
ZONE.DELETE
VLAN.IP.RANGE.CREATE
VLAN.IP.RANGE.DELETE
CONFIGURATION.VALUE.EDIT
SG.AUTH.INGRESS
SG.REVOKE.INGRESS
HOST.RECONNECT
MAINT.CANCEL
MAINT.CANCEL.PS
MAINT.PREPARE
MAINT.PREPARE.PS
VPN.REMOTE.ACCESS.CREATE
VPN.REMOTE.ACCESS.DESTROY
VPN.USER.ADD
VPN.USER.REMOVE
NETWORK.RESTART
UPLOAD.CUSTOM.CERTIFICATE
STATICNAT.ENABLE
STATICNAT.DISABLE
SSVM.CREATE
SSVM.DESTROY
SSVM.START
SSVM.STOP
SSVM.REBOOT
SSVM.H

Alerts

The following is the list of alert type numbers. The current alerts can be found by calling listAlerts.

```
MEMORY = 0
CPU = 1
STORAGE =2
STORAGE_ALLOCATED = 3
PUBLIC_IP = 4
PRIVATE_IP = 5
HOST = 6
USERVM = 7
DOMAIN_ROUTER = 8
CONSOLE_PROXY = 9
ROUTING = 10// lost connection to default route (to the gateway)
STORAGE_MISC = 11 // lost connection to default route (to the gateway)
USAGE_SERVER = 12 // lost connection to default route (to the gateway)
MANAGMENT_NODE = 13 // lost connection to default route (to the gateway)
DOMAIN_ROUTER_MIGRATE = 14
CONSOLE_PROXY_MIGRATE = 15
USERVM_MIGRATE = 16
VLAN = 17
SSVM = 18
USAGE_SERVER_RESULT = 19
STORAGE_DELETE = 20;
UPDATE_RESOURCE_COUNT = 21; //Generated when we fail to update the resource count
USAGE_SANITY_RESULT = 22;
DIRECT_ATTACHED_PUBLIC_IP = 23;
LOCAL_STORAGE = 24;
RESOURCE_LIMIT_EXCEEDED = 25; //Generated when the resource limit exceeds the limit.
Currently used for recurring snapshots only
```

Time Zones

The following time zone identifiers are accepted by the CloudPlatform API. There are several places that have a time zone as a required or optional parameter. These include scheduling recurring snapshots, creating a user, and specifying the usage time zone in the Configuration table.

Etc/GMT+12
Etc/GMT+11
Pacific/Samoa
Pacific/Honolulu
US/Alaska
America/Los_Angeles
Mexico/BajaNorte
US/Arizona
US/Mountain
America/Chihuahua
America/Chicago
America/Costa_Rica
America/Mexico_City
Canada/Saskatchewan
America/Bogota
America/New_York
America/Caracas
America/Asuncion
America/Cuiaba
America/Halifax
America/La_Paz
America/Santiago
America/St_Johns
America/Araguaina
America/Argentina/Buenos_Aires
America/Cayenne
America/Godthab
America/Montevideo
Etc/GMT+2
Atlantic/Azores
Atlantic/Cape_Verde
Africa/Casablanca
Etc/UTC
Atlantic/Reykjavik
Europe/London
CET
Europe/Bucharest
Africa/Johannesburg
Asia/Beirut
Africa/Cairo
Asia/Jerusalem
Europe/Minsk
Europe/Moscow
Africa/Nairobi
Asia/Karachi
Asia/Kolkata
Asia/Bangkok
Asia/Shanghai
Asia/Kuala_Lumpur
Australia/Perth
Asia/Taipei
Asia/Tokyo
Asia/Seoul
Australia/Adelaide
Australia/Darwin
Australia/Brisbane
Australia/Canberra
Pacific/Guam
Pacific/Auckland

What's New?

The following describes any new major features of each CloudPlatform version as it applies to API usage.

What's New in 3.0.x?

Enabling Port 8096

Port 8096, which allows API calls without authentication, is closed and disabled by default on any fresh 3.0.1 installations. You can enable 8096 (or another port) for this purpose as follows:

1. Ensure that the first Management Server is installed and running.
2. Set the global configuration parameter integration.api.port to the desired port.
3. Restart the Management Server.
4. On the Management Server host machine, create an iptables rule allowing access to that port.

Stopped VM

CloudPlatform now supports creating a VM without starting it. You can determine whether the VM needs to be started as part of the VM deployment. A VM can now be deployed in two ways: create and start a VM (the default method); or create a VM and leave it in the stopped state.

A new request parameter, startVM, is introduced in the deployVm API to support the stopped VM feature.

The possible values are:

- true - The VM starts as a part of the VM deployment.
- false - The VM is left in the stopped state at the end of the VM deployment.

The default value is true.

Change to Behavior of List Commands

There was a major change in how our List* API commands work in CloudStack 3.0 compared to 2.2.x. The rules below apply only for managed resources – those that belong to an account, domain, or project. They are irrelevant for the List* commands displaying unmanaged (system) resources, such as hosts, clusters, and external network resources.

When no parameters are passed in to the call, the caller sees only resources owned by the caller (even when the caller is the administrator). Previously, the administrator saw everyone else's resources by default.

When accountName and domainId are passed in:

- The caller sees the resources dedicated to the account specified.
- If the call is executed by a regular user, the user is authorized to specify only the user's own account and domainId.

- If the caller is a domain administrator, CloudStack performs an authorization check to see whether the caller is permitted to view resources for the given account and domainId.

When projectId is passed in, only resources belonging to that project are listed.

When domainId is passed in, the call returns only resources belonging to the domain specified. To see the resources of subdomains, use the parameter isRecursive=true. Again, the regular user can see only resources owned by that user, the root administrator can list anything, and a domain administrator is authorized to see only resources of the administrator's own domain and subdomains.

To see all resources the caller is authorized to see, except for Project resources, use the parameter listAll=true.

To see all Project resources the caller is authorized to see, use the parameter projectId=-1.

There is one API command that doesn't fall under the rules above completely: the listTemplates command. This command has its own flags defining the list rules:

listTemplates Flag	Description
featured	Returns templates that have been marked as featured and public
self	Returns templates that have been registered or created by the calling user
selfexecutable	Same as self, but only returns templates that are ready to be deployed with
sharedexecutable	Ready templates that have been granted to the calling user by another user
executable	Templates that are owned by the calling user, or public templates, that can be used to deploy a new VM
community	Returns templates that have been marked as public but not featured
all	Returns all templates (only usable by admins)

The CloudPlatform UI on a general view will display all resources that the logged-in user is authorized to see, except for project resources. To see the project resources, select the project view.

Removed API commands

- createConfiguration (Adds configuration value)
- configureSimulator (Configures simulator)

Added API commands

Added in 3.0.3:

- enableCiscoNexusVSM (Enables Nexus 1000v dvSwitch in CloudPlatform.)
- disableCiscoNexusVSM (Disables Nexus 1000v dvSwitch in CloudPlatform.)
- deleteCiscoNexusVSM (Deletes Nexus 1000v dvSwitch in CloudPlatform.)
- listCiscoNexusVSMs (Lists the control VLAN ID, packet VLAN ID, and data VLAN ID, as well as the IP address of the Nexus 1000v dvSwitch.)
- markDefaultZoneForAccount (Marks a default zone for the current account.)
- uploadVolume (Uploads a data disk to a VM)

Added in 3.0.2:

- changeServiceForSystemVm

  Changes the service offering for a system VM (console proxy or secondary storage). The system VM must be in a "Stopped" state for this command to take effect.

Added in 3.0.1:

- ldapRemove (Remove the LDAP context for this site.)

Added in 3.0.0:

- assignVirtualMachine (Move a user VM to another user under same domain.)
- restoreVirtualMachine (Restore a VM to original template or specific snapshot)
- createLBStickinessPolicy (Creates a Load Balancer stickiness policy)
- deleteLBStickinessPolicy (Deletes a LB stickiness policy.)
- listLBStickinessPolicies (Lists LBStickiness policies.)
- ldapConfig (Configure the LDAP context for this site.)
- addSwift (Adds Swift.)
- listSwifts (List Swift.)
- migrateVolume (Migrate volume)
- updateStoragePool (Updates a storage pool.)
- authorizeSecurityGroupEgress (Authorizes a particular egress rule for this security group)
- revokeSecurityGroupEgress (Deletes a particular egress rule from this security group)

- createNetworkOffering (Creates a network offering.)
- deleteNetworkOffering (Deletes a network offering.)
- createProject (Creates a project)
- deleteProject (Deletes a project)
- updateProject (Updates a project)
- activateProject (Activates a project)
- suspendProject (Suspends a project)
- listProjects (Lists projects and provides detailed information for listed projects)
- addAccountToProject (Adds acoount to a project)
- deleteAccountFromProject (Deletes account from the project)
- listProjectAccounts (Lists project's accounts)
- listProjectInvitations (Lists an account's invitations to join projects)
- updateProjectInvitation (Accepts or declines project invitation)
- deleteProjectInvitation (Deletes a project invitation)
- updateHypervisorCapabilities (Updates a hypervisor capabilities.)
- listHypervisorCapabilities (Lists all hypervisor capabilities.)
- createPhysicalNetwork (Creates a physical network)
- deletePhysicalNetwork (Deletes a Physical Network.)
- listPhysicalNetworks (Lists physical networks)
- updatePhysicalNetwork (Updates a physical network)
- listSupportedNetworkServices (Lists all network services provided by CloudPlatform or for the given Provider.)
- addNetworkServiceProvider (Adds a network serviceProvider to a physical network)
- deleteNetworkServiceProvider (Deletes a Network Service Provider.)
- listNetworkServiceProviders (Lists network serviceproviders for a given physical network.)
- updateNetworkServiceProvider (Updates a network serviceProvider of a physical network)
- addTrafficType (Adds traffic type to a physical network)
- deleteTrafficType (Deletes traffic type of a physical network)

- listTrafficTypes (Lists traffic types of a given physical network.)
- updateTrafficType (Updates traffic type of a physical network)
- listTrafficTypeImplementors (Lists implementors of implementor of a network traffic type or implementors of all network traffic types)
- createStorageNetworkIpRange (Creates a Storage network IP range.)
- deleteStorageNetworkIpRange (Deletes a storage network IP Range.)
- listStorageNetworkIpRange (List a storage network IP range.)
- updateStorageNetworkIpRange (Update a Storage network IP range, only allowed when no IPs in this range have been allocated.)
- listUsageTypes (List Usage Types)
- addF5LoadBalancer (Adds a F5 BigIP load balancer device)
- configureF5LoadBalancer (configures a F5 load balancer device)
- deleteF5LoadBalancer (delete a F5 load balancer device)
- listF5LoadBalancers (lists F5 load balancer devices)
- listF5LoadBalancerNetworks (lists network that are using a F5 load balancer device)
- addSrxFirewall (Adds a SRX firewall device)
- deleteSrxFirewall (delete a SRX firewall device)
- configureSrxFirewall (Configures a SRX firewall device)
- listSrxFirewalls (lists SRX firewall devices in a physical network)
- listSrxFirewallNetworks (lists network that are using SRX firewall device)
- addNetscalerLoadBalancer (Adds a netscaler load balancer device)
- deleteNetscalerLoadBalancer (delete a netscaler load balancer device)
- configureNetscalerLoadBalancer (configures a netscaler load balancer device)
- listNetscalerLoadBalancers (lists netscaler load balancer devices)
- listNetscalerLoadBalancerNetworks (lists network that are using a netscaler load balancer device)
- createVirtualRouterElement (Create a virtual router element.)
- configureVirtualRouterElement (Configures a virtual router element.)
- listVirtualRouterElements (Lists all available virtual router elements.)

Added CloudPlatform Error Codes

You can now find the CloudPlatform-specific error code in the exception response for each type of exception. The following list of error codes is added to the new class named CSExceptionErrorCode. These codes are applicable in CloudPlatform 3.0.3 and later versions.

- 4250 : "com.cloud.utils.exception.CloudRuntimeException"
- 4255 : "com.cloud.utils.exception.ExceptionUtil"
- 4260 : "com.cloud.utils.exception.ExecutionException"
- 4265 : "com.cloud.utils.exception.HypervisorVersionChangedException"
- 4270 : "com.cloud.utils.exception.RuntimeCloudException"
- 4275 : "com.cloud.exception.CloudException"
- 4280 : "com.cloud.exception.AccountLimitException"
- 4285 : "com.cloud.exception.AgentUnavailableException"
- 4290 : "com.cloud.exception.CloudAuthenticationException"
- 4295 : "com.cloud.exception.CloudExecutionException"
- 4300 : "com.cloud.exception.ConcurrentOperationException"
- 4305 : "com.cloud.exception.ConflictingNetworkSettingsException"
- 4310 : "com.cloud.exception.DiscoveredWithErrorException"
- 4315 : "com.cloud.exception.HAStateException"
- 4320 : "com.cloud.exception.InsufficientAddressCapacityException"
- 4325 : "com.cloud.exception.InsufficientCapacityException"
- 4330 : "com.cloud.exception.InsufficientNetworkCapacityException"
- 4335 : "com.cloud.exception.InsufficientServerCapacityException"
- 4340 : "com.cloud.exception.InsufficientStorageCapacityException"
- 4345 : "com.cloud.exception.InternalErrorException"
- 4350 : "com.cloud.exception.InvalidParameterValueException"
- 4355 : "com.cloud.exception.ManagementServerException"
- 4360 : "com.cloud.exception.NetworkRuleConflictException"
- 4365 : "com.cloud.exception.PermissionDeniedException"

- 4370 : "com.cloud.exception.ResourceAllocationException"
- 4375 : "com.cloud.exception.ResourceInUseException"
- 4380 : "com.cloud.exception.ResourceUnavailableException"
- 4385 : "com.cloud.exception.StorageUnavailableException"
- 4390 : "com.cloud.exception.UnsupportedServiceException"
- 4395 : "com.cloud.exception.VirtualMachineMigrationException"
- 4400 : "com.cloud.exception.AccountLimitException"
- 4405 : "com.cloud.exception.AgentUnavailableException"
- 4410 : "com.cloud.exception.CloudAuthenticationException"
- 4415 : "com.cloud.exception.CloudException"
- 4420 : "com.cloud.exception.CloudExecutionException"
- 4425 : "com.cloud.exception.ConcurrentOperationException"
- 4430 : "com.cloud.exception.ConflictingNetworkSettingsException"
- 4435 : "com.cloud.exception.ConnectionException"
- 4440 : "com.cloud.exception.DiscoveredWithErrorException"
- 4445 : "com.cloud.exception.DiscoveryException"
- 4450 : "com.cloud.exception.HAStateException"
- 4455 : "com.cloud.exception.InsufficientAddressCapacityException"
- 4460 : "com.cloud.exception.InsufficientCapacityException"
- 4465 : "com.cloud.exception.InsufficientNetworkCapacityException"
- 4470 : "com.cloud.exception.InsufficientServerCapacityException"
- 4475 : "com.cloud.exception.InsufficientStorageCapacityException", 4475
- 4480 : "com.cloud.exception.InsufficientVirtualNetworkCapcityException"
- 4485 : "com.cloud.exception.InternalErrorException"
- 4490 : "com.cloud.exception.InvalidParameterValueException"
- 4495 : "com.cloud.exception.ManagementServerException"
- 4500 : "com.cloud.exception.NetworkRuleConflictException"

- 4505 : "com.cloud.exception.PermissionDeniedException"
- 4510 : "com.cloud.exception.ResourceAllocationException"
- 4515 : "com.cloud.exception.ResourceInUseException"
- 4520 : "com.cloud.exception.ResourceUnavailableException"
- 4525 : "com.cloud.exception.StorageUnavailableException"
- 4530 : "com.cloud.exception.UnsupportedServiceException"
- 4535 : "com.cloud.exception.VirtualMachineMigrationException"
- 9999 : "com.cloud.api.ServerApiException"

Contacting Support

The support team is available to help customers plan and execute their installations. To contact the support team, log in to the support portal at support.citrix.com/cloudsupport using the account credentials you received when you purchased your support contract.