IBM WebSphere CloudBurst Appliance V2.0

REST APIs

This presentation will discuss the REST APIs provided with IBM WebSphere® CloudBurst™ version 2.0.

Agenda

- Communicating with WebSphere CloudBurst
- WebSphere CloudBurst REST interface
- Examples
- When to use the REST API
- Summary

This presentation will begin with a general discussion of the WebSphere CloudBurst REST interface, followed by some examples of using the REST APIs.

Communicating with WebSphere CloudBurst

This section will discuss the various ways you can communicate with IBM WebSphere CloudBurst.

Communicating with WebSphere CloudBurst

- Three options for communication
  - WebSphere CloudBurst administrative console
  - Command-line interface
  - REST API
- REST API using HTTP with no need for Java™ runtime environment

REST APIs

WebSphere CloudBurst has several mechanisms administrators can use to communicate with it. The administrative console provides a Web browser-based UI that can be accessed to modify WebSphere CloudBurst settings, manage users, cloud resources, patterns, and deploy and monitor virtual systems. Also available to users is the command-line interface which allows users to manage their WebSphere CloudBurst appliance settings and cloud resources in a robust command-line tool. A third option that is available to interface with the WebSphere CloudBurst appliance is using the REST API. The REST API is an HTTP-based service that accepts HTTP requests and performs WebSphere CloudBurst actions depending on the type of request that is sent. The REST API provides a mechanism for users to work with CloudBurst without having the need for a Java runtime environment.

Section

IBM WebSphere CloudBurst REST interface

REST APIs

This section will discuss the IBM WebSphere CloudBurst REST interface.

What is REST?

- REST stands for Representational State Transfer
- REST APIs implemented as HTTP patterns
- Servlet generally used to accept the REST requests
- REST request have patterns
  - Example: GET HTTP request with pattern /resources/users

REST APIs

REST stands for representational state transfer. It is typically implemented as a servlet that responds to HTTP requests. The REST service will accept HTTP requests and parse them in order to determine what action to perform. Depending on the request sent, different data is sent to and returned from the servlet hosting the REST service. The WebSphere CloudBurst REST API uses JSON format to pass data back and forth. The REST API documents the patterns that are available for REST calls, and the actions that are available to be performed by the REST service. The API also documents the data that must be sent with a REST request, and the data that is returned in a response from the REST service.

WebSphere CloudBurst REST API

- WebSphere CloudBurst Appliance exposes subset of function using a REST API
- REST enabled by default
- REST API supported over the HTTPS protocol on port 443

REST APIs

The features that you can call on the IBM WebSphere CloudBurst appliance using the REST API are a subset of all of the WebSphere CloudBurst features available. The IBM WebSphere CloudBurst administrative console and the command line interface provide a full set of WebSphere CloudBurst functionality. One of those two options should be used if the REST API does not provide the features that you require. The REST API is enabled by default and there are no extra settings that you have to configure in order to enable it. There is no way to disable the REST interface. The IBM WebSphere CloudBurst REST API is available to be called by way of HTTPS on port 443.

WebSphere CloudBurst REST API

- WebSphere CloudBurst appliance uses self-signed certificate for SSL sessions
- Certificate used for GUI, CLI and REST API
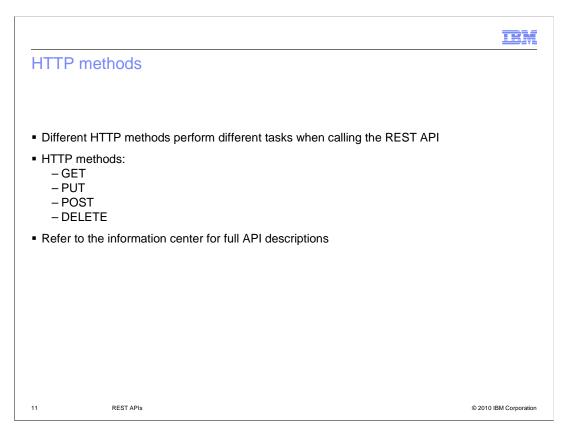- HTTPS clients must accept SSL certificate during the SSL handshake to use the REST API

REST APIs

The IBM WebSphere CloudBurst appliance uses a secure connection when accessing the REST API. The HTTP client that you use to access the WebSphere CloudBurst REST API must support SSL certificates in order to work. This functionality is supported with most Web browsers and a Web browser is a convenient way to try out some of the REST API functions. Using a Web browser as the client to interact with the REST API, you can point to the URL of the WebSphere CloudBurst appliance REST service using HTTPS on port 443. Although you can use a Web browser to access the REST API on the WebSphere CloudBurst Appliance, it is not recommended for production use as you cannot send the appropriate headers to access the public REST API as part of your REST request. Without including the proper headers during REST calls to access the public API, you are not guaranteed to have a properly formatted REST response, or a response that includes all of the data you may be expecting.

WebSphere CloudBurst REST actions

- Certificates
- Cloud groups
- Diagnostics.zip file
- Trace.zip
- Hypervisors
- IP addresses
- IP groups

REST APIs

The functionality provided by the IBM WebSphere CloudBurst REST API is a subset of the full functionality provided by the appliance. Some of the actions you can perform with REST calls to the WebSphere CloudBurst appliance include working with certificates, cloud groups, and gathering diagnostics and trace log files. You can also work with hypervisors, IP addresses, and IP groups.

WebSphere CloudBurst REST actions

- Log Viewer Manager
- Networks
- Patterns
- Storage
- Users and user groups
- Virtual machines
- Virtual systems

REST APIs

Other available WebSphere CloudBurst REST actions are available for working with the log file viewer manager, the network settings, and viewing lists of available patterns. You can use the REST API to view storage information, work with users and user groups, and manage virtual systems and virtual machines. The WebSphere CloudBurst information center contains a comprehensive list of available actions for each resource that can be managed using the REST API.

HTTP methods

- Different HTTP methods perform different tasks when calling the REST API
- HTTP methods:
  - GET
  - PUT
  - POST
  - DELETE
- Refer to the information center for full API descriptions

REST APIs

The REST API on the WebSphere CloudBurst appliance requires that different types of HTTP methods be sent depending on what functionality you want to invoke. For example, if you want to list information regarding the users that are registered to use the appliance you can do so using an HTTP GET method. If you want to perform a more advanced invocation such as adding a new IP Group to the WebSphere CloudBurst appliance, you will have to use an HTTP POST method. You will also have to pass some information in the body of the HTTPS request to the device. The data that is passed in the body of the device must be in JSON format and provide the data expected for the particular REST API call being made.

HTTP response status codes

- Success codes returned on successful calls to the REST API
- Error codes returned on unsuccessful calls to the REST API
- Example success codes: HTTP 201, 202, 203 code
- Example unsuccessful codes: HTTP 400, 403, 404, 500
- See Information Center for which exact codes are returned for REST API calls

REST APIs

Since REST APIs to the IBM WebSphere CloudBurst appliance are HTTP calls, HTTP response status codes are returned with each REST API invocation. HTTP response status codes are classified by the number they start with. HTTP response status codes that are in the 100 range are informational codes. HTTP response status codes that are in the 200 range are success codes. Response status codes in the 300 range are redirection status codes, and those in the 400 or 500 ranges are client error and server error codes. Successful calls to the WebSphere CloudBurst REST API will yield a 200 level response code, indicating a successful call to the REST API. Unsuccessful calls will have a response code in the 400 or 500 level range. Depending on the call you make to the WebSphere CloudBurst REST API, different success codes and unsuccessful HTTP response codes apply. It is best to check the IBM WebSphere CloudBurst information center to see what response codes apply to which REST API calls.

Data format and headers for REST calls

- Data passed to and from the WebSphere CloudBurst in JSON format
- Header should be included in your request to indicate the ability of your client to handle JSON responses
  - *"Accept: application/json"*
- REST API supports HTTP basic authentication
  - *Cookies returned to client after initial REST call*
  - *zsessionid* and *SimpleToken* that should be included with subsequent HTTP requests that are part of the same session

REST APIs

There are some HTTP headers that are expected in IBM WebSphere CloudBurst REST API calls. One particular header is the accept application JSON header which should be passed with your request to indicate that your REST client can accept JSON data from WebSphere CloudBurst. The WebSphere CloudBurst REST API supports HTTP authentication. When your client authenticates with the WebSphere CloudBurst appliance, two cookies are provided with the response. These cookies, zsessionid and SimpleToken, should be included in subsequent HTTP requests to WebSphere CloudBurst in the same session.
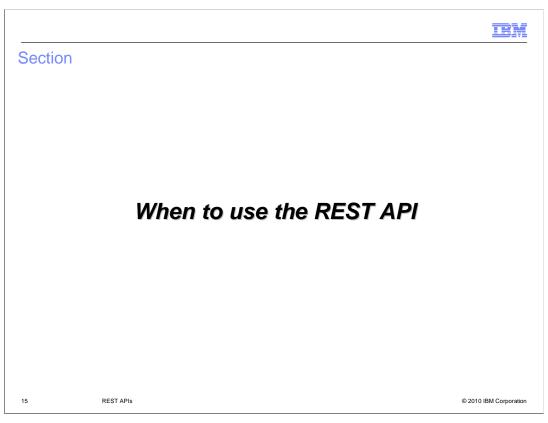
Data format and headers for REST calls

- Header expected for version 2.0 API calls
  - *"X-CloudBurst-API-Version: 2.0"*
- For requests that contain data an extra header must be specified
  - *"Content-Type: application/json"*
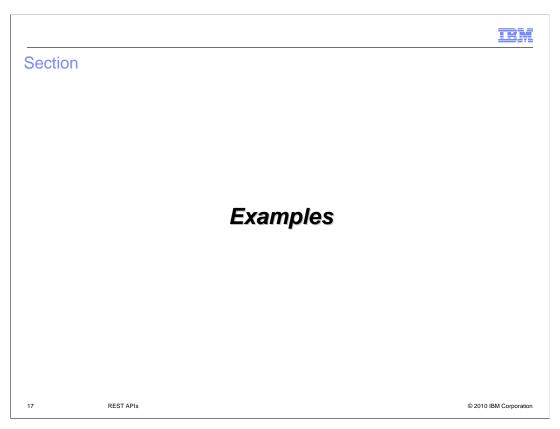- See http://json.org/ for information on JSON

REST APIs

The "X CloudBurst API Version 2.0" header should be included in HTTP calls to the WebSphere CloudBurst appliance. This serves two purposes. First, it ensures that you are calling the WebSphere CloudBurst public REST API. Calling the public API ensures that the data you receive from your requests is sent in a standardized format. Secondly, adding that header to the REST request will make sure that the response you receive from the appliance will be consistently formatted the same way, even if the appliance firmware is upgraded to a newer version. For requests made using the REST API that require data to be passed in the body of the HTTP message, the content type header must specify that the data is in application JSON format. For more information on what JSON format is, see json.org. Examples of JSON data passed during WebSphere CloudBurst REST API calls are included later in this presentation. Reference the IBM WebSphere CloudBurst information center for more examples.

Section

When to use the REST API

REST APIs

This section will discuss usage scenarios of the REST API.

Using the WebSphere CloudBurst REST API

- REST API helps integrate with applications and other software products
- IBM Tivoli® Service Automation Manager is using the REST API to interface with the WebSphere CloudBurst appliance

　　REST APIs　　

The WebSphere CloudBurst REST API is intended for use to integrate enterprise applications and software products with the WebSphere CloudBurst appliance. An example of an application that is interfacing with WebSphere CloudBurst REST API is the IBM Tivoli Service Automation Manager.

Section

Examples

REST APIs

This section will show some examples of making REST API calls to the IBM WebSphere CloudBurst appliance.

Log files

- Log file retrieval
- HTTP method: GET
- Pattern: /resources/diagnostics.zip
  - https://<APPLIANCE_IP>:443/resources/diagnostics.zip
  - https://<APPLIANCE_IP>:443/resources/diagnostics.zip?latest
- Pattern: /resources/trace.zip
  - https://<APPLIANCE_IP>:443/resources/trace.zip
  - https://<APPLIANCE_IP>:443/resources/trace.zip?latest

18   REST APIs   © 2010 IBM Corporation

The first example you will see here is a helpful REST call that can be made to the IBM WebSphere CloudBurst appliance using your Web browser. This example shows how to retrieve the diagnostics and trace log files. These files contain diagnostic logging information that can help to troubleshoot problems you experience with the WebSphere CloudBurst appliance. The HTTP method that you use to place this REST API call is a GET method, which is the standard method used by Web browsers when accessing Web pages. A browser is a convenient client to use to call the REST API. The patterns that you will use for these REST API calls are /resources/diagnostics.zip and /resources/trace.zip. These calls to the WebSphere CloudBurst appliance with your Web browser will yield a prompt to either open or save the archive files. A variation of this call is to append question mark and the word latest to the end of the URI. Appending this extra parameter to the end of the URI will place the REST API call that will return the diagnostics and trace files that contain only the latest log information.

Get IP groups

- Gets JSON list of available IP groups

- HTTP method: GET

- Pattern: /resources/ipGroups
  – https://<APPLIANCE_IP>:443/resources/ipGroups

- Sample response:

- [{ "created": 1242243975898, "gateway": "10.0.0.1", "id": 2, "name": "ten", "netmask": "255.0.0.0", "networks": ["/resources/networks/1", "/resources/networks/2"], "primarydns": "10.0.0.2", "secondarydns": "10.0.0.3", "subnetaddress": "10.0.0.0", "updated": 1242738679669 }]

REST APIs

This example shows how to get a list of available IP groups on the WebSphere CloudBurst appliance. This method is also an HTTP GET method. The pattern to get this information is /resources/ipGroups. When this call is placed to your WebSphere CloudBurst device using your Web browser, a file of type JSON is returned and you are prompted to open or save the file. Examining the contents of the file will reveal data in JSON format showing the IP group information for the WebSphere CloudBurst device.

Get the WebSphere CloudBurst users

- Get a list of WebSphere CloudBurst users
- HTTP method: Get
- Pattern: /resources/users
  - https://<APPLIANCE_IP>:443/resources/users

REST APIs

This example shows how to get a list of users registered to the WebSphere CloudBurst device. This is also an HTTP GET request that can be placed using your Web browser. The pattern to place the REST API call is slash resources slash users. Placing this call to your WebSphere CloudBurst REST interface will return a list of users registered to the appliance in JSON format.

Get users sample JSON response

```
[ {
  "username": "cbadmin",
  "currentstatus": "RM01062",
  "name": "Administrator",
  "id": 1,
  "email": "",
  "currentmessage": "RM02012"
},
{
  "username": "Deployer",
  "currentstatus": "RM01062",
  "name": "Ms. Pattern Deployer",
  "id": 2,
  "email": "patterns@ibm.com",
  "currentmessage": "RM02013"
} ]
```

Here you can see the information in JSON format that is returned when a successful call to the users REST API is placed. In this example you can see that there are two users registered to the WebSphere CloudBurst appliance, cbadmin, and Deployer. The users' names, the user ID, e-mail address, and current status codes are all returned as part of the JSON message. Note that the square and curly braces in the message are clear indications that this is JSON formatted data.

Add a hypervisor

- HTTP method: POST

- Pattern: /resources/hypervisors

- JSON data to pass to the WebSphere CloudBurst appliance
  - [{ "address": "anotherhypervisor.mycompany.com", "name": "another hypervisor", "password": "thepassword", "type": "ESX", "userid": "root" }]

- Note: cannot do this using Web browser as client

REST APIs

So far you have seen how to place calls to the WebSphere CloudBurst REST interface using HTTP Get methods and a Web browser. Here is another type of REST call that creates a new hypervisor. You cannot perform this operation by inputting the URI into a standard Web browser as you saw in the previous examples. This REST call requires that an HTTP POST call be made. Extra information is passed to the REST API in JSON format in order to provide the WebSphere CloudBurst appliance with the information that it needs to register the new hypervisor. The HTTP POST call includes a body containing JSON data. The JSON data includes the address of the hypervisor and the name for the hypervisor. It also must include the hypervisor type, and the user ID and password to access the hypervisor. In this example, the type of hypervisor is ESX.

JSON response to add a hypervisor call

```
[{ "address":  https://anotherhypervisor.mycompany.com/sdk",
"certified": "I",
"cloud": null,
"created": 1245249680477,
"currentmessage": "RM03106",
"currentmessage_text": "Maintenance mode (must add to a cloud group to start)",
"currentstatus": "RM01025", "currentstatus_text": "Maintenance mode",
"desiredstatus": "RM01025", "desiredstatus_text": "Maintenance mode",
"id": 5,
"name": "another hypervisor",
"networks": [ ],
"password": "124524968047629",
"storage": [ ],
"type": "ESX",
"updated": 1245249680477,
"userid": "root" } ]
```

When placing the POST call to the WebSphere CloudBurst device and including the JSON data, the WebSphere CloudBurst appliance will return a response call in JSON format when successful. The information in the response includes a summary of the hypervisor configuration data. The hypervisor address is included, in addition to the current status code, name, hypervisor type, user ID, and the hypervisor ID.

Summary

REST APIs

This section will summarize the IBM WebSphere CloudBurst REST API presentation.

Summary

- REST APIs perform operations on the WebSphere CloudBurst appliance over HTTP
- The REST APIs support subset of administrative console and command-line interface functionality
- REST patterns used to perform actions
- JSON data is passed to and from the WebSphere CloudBurst appliance

The IBM WebSphere CloudBurst REST API provides another option when communicating with the IBM WebSphere CloudBurst appliance. Along with using the WebSphere CloudBurst administrative console and command-line interface, the REST API provides an option for working with the appliance using HTTP calls. The activities that can be performed with the REST API are a subset of the actions that are available using the administrative console and the command-line interface. Depending on what actions you want to perform using the REST API, you will use different HTTP methods, such as GET, PUT, POST, and DELETE. For WebSphere CloudBurst activities called using the REST API that require data, information is passed in the body of the HTTP message in JSON format. HTTP response status codes are returned to the calling application from the WebSphere CloudBurst device when working with the REST API. The 200 level status codes indicate a successful REST API call, while a 400 and 500 level status response code indicates a client or server failure. Refer to the IBM WebSphere CloudBurst information center for full documentation regarding the REST APIs and the data necessary for each call.

Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?

- Did it help you solve a problem or answer a question?

- Do you have suggestions for improvements?

Click to send email feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_CB20_RESTAPIs.ppt

This module is also available in PDF format at: ../CB20_RESTAPIs.pdf

26 REST APIs © 2010 IBM Corporation

You can help improve the quality of IBM Education Assistant content by providing feedback.