

WebFOCUS

WebFOCUS Embedded
Business Intelligence User's Guide
Release 8.2 Version 03

May 10, 2018

Active Technologies, EDA, EDA/SQL, FIDEL, FOCUS, Information Builders, the Information Builders logo, iWay, iWay Software, Parlay, PC/FOCUS, RStat, Table Talk, Web390, WebFOCUS, WebFOCUS Active Technologies, and WebFOCUS Magnify are registered trademarks, and DataMigrator and Hyperstage are trademarks of Information Builders, Inc.

Adobe, the Adobe logo, Acrobat, Adobe Reader, Flash, Adobe Flash Builder, Flex, and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Due to the nature of this material, this document refers to numerous hardware and software products by their trademarks. In most, if not all cases, these designations are claimed as trademarks or registered trademarks by their respective companies. It is not this publisher's intent to use any of these names generically. The reader is therefore cautioned to investigate all claimed trademark rights before using any of these names other than to refer to the product described.

Copyright © 2018, by Information Builders, Inc. and iWay Software. All rights reserved. Patent Pending. This manual, or parts thereof, may not be reproduced in any form without the written permission of Information Builders, Inc.

Contents

Introducing WebFOCUS Embedded Business Intelligence	11
1. Introducing WebFOCUS Embedded Business Intelligence	13
Understanding Techniques and Considerations For Embedding Business Intelligence.....	13
Implementing Single Sign-On.....	13
Implementing Embedded Business Intelligence in Same Origin and Cross-Origin Scenarios.....	15
Best Practices for Embedding BI.....	16
Flow Control Options.....	16
User Context Options.....	16
WebFOCUS RESTful Web Services	17
2. Introducing WebFOCUS RESTful Web Services	19
What Is REST?.....	19
What are RESTful Web Services?.....	20
Considerations When Using HTTP Methods: GET and POST.....	20
3. WebFOCUS Repository RESTful Web Service Requests	23
Authenticating WebFOCUS Sign-On Requests.....	23
Cross-Site Request Forgery (CSRF).....	26
Obtaining a CSRF Token.....	28
Passing a CSRF Token.....	29
Configuring Single Sign On.....	30
Example 1: Adapt the Initial Sign In Request for Single Sign On Environments.....	32
Example 2: SiteMinder (Initial Request).....	34
Example 3: SiteMinder (Subsequent Requests).....	36
Signing Out of WebFOCUS.....	36
WebFOCUS Repository.....	37
Creating and Updating a Folder.....	37
Deleting a Folder.....	41
Deleting a WebFOCUS Repository Report.....	43
Listing Folders and Subfolders.....	44
Listing Reports, Schedules, and Library Content Within the WebFOCUS Repository..	50
Listing the Versions for a Report Library Report.....	54
Listing the Parameters for a Repository Report.....	58

Running a Report From the WebFOCUS Repository	61
Change Management Export.	63
Change Management Import.	65
Publishing an Item.	67
Unpublishing an Item.	68
Copying an Item.	70
Moving an Item.	72
Renaming an Item.	75
Uploading a WebFOCUS Report.	77
Creating a URL Link.	80
Retrieving Content for a WebFOCUS Report and URL.	83
4. WebFOCUS Reporting Server RESTful Web Service Requests	85
Listing WebFOCUS Reporting Server Nodes.	85
Creating an Application.	86
Listing Applications.	88
Listing Files Within an Application.	89
Listing the Parameters for a Report Within an Application.	91
Running a Report Within an Application.	94
Deleting an Application.	96
Change Management Export.	97
Change Management Import.	98
Deleting a Role.	98
Adding a Rule.	99
5. WebFOCUS Security Administration RESTful Web Service Requests	103
Listing Users.	103
Listing Groups.	106
Listing Privileges.	108
Listing Roles.	109
Listing Users Within a Group.	111
Adding and Updating a User.	114
Deleting a User.	116
Adding and Updating a Group.	117
Deleting a Group.	118

Adding a User to a Group.	119
Removing a User From a Group.	121
Adding a Role.	123
Deleting a Role.	128
Adding a Rule.	129
Deleting a Rule.	132
Listing Rules for a Subject.	134
Listing Rules for a Resource.	136
Listing Rules for a Role.	137
Expanding a Policy String.	138
Creating a Policy String.	142
Running a Resource Template.	146
Changing a Password for a User.	148
6. ReportCaster RESTful Web Service Requests	151
Retrieving Reports From the ReportCaster Library.	151
Deleting a Version of a Report From the ReportCaster Library.	153
Creating and Updating an Address Book.	154
Creating and Updating a Library Access List.	163
Deleting a Library Access List.	170
Creating and Updating a Schedule.	172
Schedule rootObject.	174
Schedule Properties.	174
Notification.	175
Distribution.	176
Report Library.	176
Email.	180
FTP.	184
Printer.	188
WebFOCUS Repository.	189
Recurrence.	190
Run Once.	190
Minutes.	191
Hourly.	192

Daily.....	194
Weekly.....	196
Monthly.....	199
Yearly.....	202
Custom.....	204
Task.....	207
WebFOCUS Report.....	207
WebFOCUS Server Procedure.....	210
File.....	213
FTP.....	214
URL.....	215
Closing Tag.....	216
Example 1: Creating a Schedule.....	217
Example 2: Updating a Schedule.....	222
Running a Schedule.....	224
Retrieving a Schedule.....	225
Deleting a Schedule.....	228
Deleting an Address Book.....	230
Log Functionality.....	232
Deleting a Specific Log.....	232
Deleting Logs for a Specific Time Period.....	233
Deleting Logs for an Owner.....	234
Deleting Logs for a Schedule ID.....	235
Deleting Logs for a Schedule ID Within a Time Period.....	236
Retrieving Last Log for a Schedule ID.....	237
Retrieving the Log for a Job ID.....	246
Retrieving the Log List for an Owner.....	254
Retrieving the Log List for an Owner Within a Time Period.....	258
Retrieving the Log List for a Schedule.....	262
Retrieving a List of Schedule Owners.....	267
Console Functionality.....	268
Changing Job Priority.....	268
Retrieving Job Status.....	269

Listing Jobs in the Queue.....	270
Listing Jobs in the Queue for an Owner.....	276
Listing Running Jobs.....	281
Listing Running Jobs for an Owner.....	294
Removing a Job From the Job Queue.....	306
7. Using the RESTful Web Services Test Page	309
Accessing the Test Page.....	309
Using the Test Page.....	310
8. Alternative Method of Calling WebFOCUS RESTful Web Service Requests	313
Calling WebFOCUS RESTful Web Service Requests.....	313
9. Visual Basic .NET, Java, HTML and jQuery Code Examples	315
Signing In to WebFOCUS.....	315
Visual Basic .NET Example.....	315
Java Example.....	316
HTML and jQuery Example.....	317
Listing Folders From WebFOCUS.....	318
Visual Basic .NET Example.....	318
Java Example.....	319
HTML and jQuery Example.....	320
Running a WebFOCUS Report.....	321
Visual Basic .NET Example.....	322
Java Example.....	323
HTML and jQuery Example.....	324
Handling Drill-downs, Active Cache, and On-Demand Paging Reports.....	326
Visual Basic .NET Example (signOn.aspx and WebForm2.aspx).....	326
Java Example (signOn.jsp and WebForm2.jsp).....	329
HTML and jQuery Example (drillOne.html and drillTwo.html).....	333
Parsing the XML Response of a SignOn Request to Obtain the CSRF Name and Value...	337
Java Example.....	338
XML Parser Class.....	339
Visual Basic .NET Example.....	341
XML Parser Function.....	342
10. Accessing InfoAssist Directly Through URL Calls	345

Starting InfoAssist.	345
WebFOCUS Open Portal Services	349
11. Introducing WebFOCUS Open Portal Services	351
WebFOCUS Open Portal Services.	351
Benefits of Using WebFOCUS Open Portal Services.	352
Java Portlet Specification 2.0 (JSR 286) Support.	352
12. Using WebFOCUS Portal Components	355
WebFOCUS Open Portal Services Components Overview.	355
WebFOCUS Report Component.	356
WebFOCUS Deferred Status Component.	356
WebFOCUS Resource Tree Component.	357
WebFOCUS Portal Component.	358
WebFOCUS Portal Tree Component.	358
Using WebFOCUS Open Portal Services Components.	358
Setting the Source URL Parameter.	359
Using the WebFOCUS Report Component.	359
Launch Mode.	359
Folder Mode.	361
List Mode.	363
WebFOCUS Report Component Parameters.	365
WebFOCUS Report Component Configurations.	369
Using the WebFOCUS Deferred Status Component.	370
Using the WebFOCUS Resource Tree Component.	370
Content Node.	371
Favorites Node.	372
Mobile Favorites Node.	373
Recent Items Node.	374
WebFOCUS Resource Tree Component Parameters.	374
Usage Considerations.	377
Right-Click Context Menu Persists When Working in Another Portlet.	377
Portlet Menu Options to Avoid.	378
13. Installing WebFOCUS App Parts for Microsoft SharePoint 2016	379

- On-Premise SharePoint Server..... 379
- Using Apps (Add-ins) With a Developer Site and Other Site Types Through the App Catalog..... 381
- Using SharePoint on Microsoft Office 365 and Azure (In the Cloud)..... 383
- 14. Installing WebFOCUS Web Parts for Microsoft SharePoint 2013 385
 - Microsoft SharePoint Portal Server 2013.....385
- 15. Embedding WebFOCUS Business Intelligence Content Into Salesforce.com 391
 - Embedding a URL to Run a WebFOCUS Report.....391
 - Configuring SAML Authentication.....395
 - Enabling the Identity Provider.....396
 - Configuring WebFOCUS and Generating the wfspMetadata.xml File..... 399
 - Configuring WebFOCUS as a Service Provider for Salesforce.com.....404
 - Programming Solutions.....410
- 16. Accessing WebFOCUS Components Directly Through URL Calls423
 - Report Component.....423
 - Deferred Status Component.....425
 - Resource Tree Component.....426
 - Portal Component.....426
 - Portal Tree Component.....427
- WebFOCUS Embedded Business Intelligence Demonstration Application 429**
- 17. WebFOCUS Embedded Business Intelligence Demonstration Application 431
 - Installing the Embedded Business Intelligence Demonstration Application..... 431
 - Installing the Sample Embedded Content..... 432
 - Importing the Sample User (ffadv)..... 436
 - Installing the Embedded BI Demo Application (Fintoso Financial)..... 441
 - Required HTML 5 Chart Extensions..... 443
 - Configuring the Embedded Business Intelligence Demonstration Application..... 443
 - Configuring a Back Channel Ticket Request..... 444
 - Configuring WebFOCUS.....445
 - Using the Trusted Ticket Test Pages452
 - Using the Embedded Business Intelligence Demonstration Application (Fintoso Financial)..... 454

Accessing and Running the Embedded BI Demo Application.	454
Reviewing the Internal (Back-End) Functionality of the Embedded BI Demo Application.	467
Additional Considerations for Embedded Business Intelligence.	469
Hiding BI Portal Features.	469
Branding and Rebranding.	469
Responsive Web Design.	470
Alternate Security Zone.	471
Customizing the Embedded Business Intelligence Demonstration Application (Fintoso Financial).	472
Registering User Names.	472
Using Different BI Portal Content.	472
Troubleshooting.	473
Pop-up Message: Failed to Obtain a Trusted Ticket From WebFOCUS.	473
Ticket Value is: null.	474
Ticket Value is: -1.	474
BI Portal Tabs Display an Error or are Blank.	476
Appendix: Detailed Request/Response Flow for the Embedded Business Intelligence Demonstration Application.	478

Introducing WebFOCUS Embedded Business Intelligence

The *WebFOCUS Embedded Business Intelligence User's Guide* introduces WebFOCUS embedded business intelligence (BI) and includes comprehensive content on WebFOCUS RESTful Web Services for developers and WebFOCUS Open Portal Services, which are key components in embedded BI solutions. For more information, see [WebFOCUS RESTful Web Services](#) on page 17 and [WebFOCUS Open Portal Services](#) on page 349.

In addition, the embedded BI demonstration (“demo”) application that is packaged with WebFOCUS Release 8.2 Version 03 is documented in this user's guide. For more information, see [WebFOCUS Embedded Business Intelligence Demonstration Application](#) on page 429.

Introducing WebFOCUS Embedded Business Intelligence

WebFOCUS embedded Business Intelligence (BI) provides the capability of incorporating WebFOCUS content, analytics, and functionality (features) into an external application.

This section provides an introduction to WebFOCUS embedded BI, which also highlights key features and considerations.

In this chapter:

- [Understanding Techniques and Considerations For Embedding Business Intelligence](#)
 - [Best Practices for Embedding BI](#)
-

Understanding Techniques and Considerations For Embedding Business Intelligence

There are several approaches and techniques that can be used to embed business intelligence (BI) into an external application, including:

1. Embedding WebFOCUS BI Portal pages or other WebFOCUS content into an HTML iframe.
2. Using WebFOCUS web services to embed content into the application.
3. Launching WebFOCUS tools such as InfoAssist from the application.
4. Using WebFOCUS Open Portal Services to embed content (specifically, WebFOCUS portlets) in a JSR 286-compliant portal environment (for example, Microsoft SharePoint).

The embedded BI demonstration ("demo") application that is packaged with WebFOCUS enables you to explore the iframe and web services embedding options.

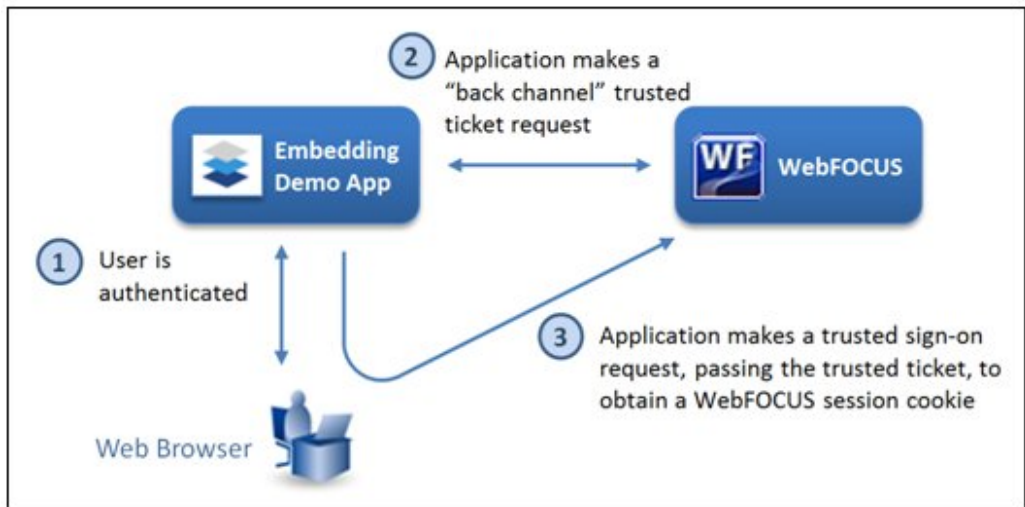
Implementing Single Sign-On

An important developer consideration for an embedded BI application is how to implement single sign-on (SSO) between the embedded application and WebFOCUS so that the user is not required to provide credentials on multiple occasions. There are several options to consider, including:

- Windows Authentication
- SAML 2.0
- Web SSO Products
- Custom Solutions

In this content and accompanying embedded BI demo application, you will learn how the Trusted Ticket Authentication feature in WebFOCUS can be used to implement SSO.

As shown in the following diagram, the embedded BI demo application authenticates the user (1) with a simple method, as described in *Registering User Names* and then makes a trusted ticket request (2) to WebFOCUS. This is typically referred to as a *back channel* request because the connection is established directly between the server hosting the embedded application and the server hosting WebFOCUS, and is therefore not seen by the network where the web browser of the user is running.



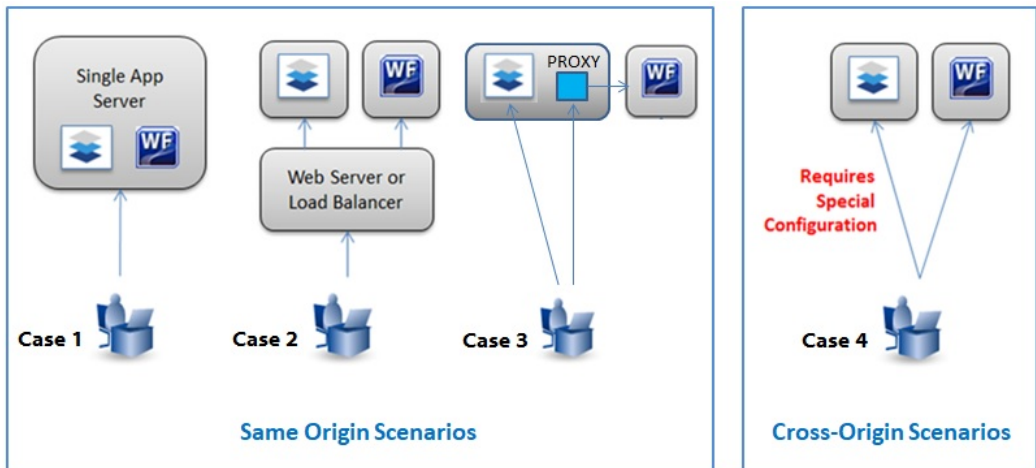
WebFOCUS verifies that the trusted ticket request originates from a trusted host and returns a trusted authentication ticket. The application then presents this ticket in a trusted sign-on request (3), originating from the web browser of the user, in order to obtain a WebFOCUS session cookie. Moving forward from this point, the embedded BI demo application can request content from WebFOCUS using URL requests or the WebFOCUS RESTful Web Services API.

If the embedded application will be making POST requests that create or update WebFOCUS resources, then a Cross-Site Request Forgery (CSRF) token must be obtained from WebFOCUS and submitted with these requests. The trusted sign-on request can include an option to obtain a CSRF token. The embedded BI demo application receives a CSRF token, but does not use it.

Implementing Embedded Business Intelligence in Same Origin and Cross-Origin Scenarios

Another important consideration is the deployment topology, which relates to where the embedded application and WebFOCUS reside. Often times both are deployed on or behind a single web host. This includes when both are deployed on the same application server (as depicted by **Case 1** in the following diagram) and when they are deployed on different servers, but accessed from a single web server or load balancer (**Case 2**). In addition, a proxy method can also be used (as depicted by **Case 3** in the following diagram), where the browser communicates to a proxy, which acts as an intermediary to WebFOCUS.

These are referred to as *same origin* scenarios.



At times it may be necessary for the user to access the embedded application and WebFOCUS from different web hosts as depicted by **Case 4** in this diagram. This is referred to as a *cross-origin* scenario. Due to security controls built into modern web browsers, you will need to configure the WebFOCUS iframe embedding and Cross-Origin Request Sharing (CORS) features before the embedded BI demo application can be used in a cross-origin scenario. For more information on configuring and using cross-origin settings in WebFOCUS, see the *WebFOCUS Security and Administration* content.

Other considerations for embedded BI include branding/rebranding and responsive web design. The embedded BI demo application shows how you can restyle WebFOCUS content and BI Portal pages so that they blend in visually with the host application in order to provide an improved user experience. The sample BI Portals that are included with the demo have responsive page layouts, which require the hosting iframe height to be dynamically adjusted as required by the portal page. This can be accomplished by setting the *Broadcast height for embedding* option in the BI Portals and adding an event listener to the embedded BI demo. For more information on these settings and options, see *Additional Considerations for Embedded Business Intelligence*.

As you can see, WebFOCUS provides robust support for embedded BI. This enables customers to extend commercial and custom web applications with the extensive business intelligence and analytics capabilities available in WebFOCUS with minimal effort and with an exceptional user experience.

Best Practices for Embedding BI

This section outlines several best practices for embedding BI.

Flow Control Options

- Front Channel.** Used with iframe and/or AJAX (Asynchronous JavaScript and XML) approaches where the browser connects to WebFOCUS on the same or a different origin host.
- Proxy Method.** A browser communicates with a proxy that acts as an intermediary to WebFOCUS.

User Context Options

- Service Account.** Typically used with a proxy but does not require it. This would require a user ID and password for the service account. If user-specific content will be returned, then a user ID parameter must be injected by the proxy code so WebFOCUS can use it to filter the data (since in this case WebFOCUS runs all requests as the service account user).
- Trusted Ticket.** Required for iframe Portal, hyperlink InfoAssist, hyperlink Insight, and PGX page embedding. It is recommended when embedding charts that have interactive features, such as autolink drill-downs. It can be used with proxy or front channel configurations.

WebFOCUS RESTful Web Services

This content describes how to develop and use WebFOCUS RESTful Web Services. It is intended for experienced developers who will use this capability to expose WebFOCUS content and functionality as callable services from a Microsoft Visual Studio .NET or J2EE development platform. Developers should have knowledge of RESTful web service technology and object oriented programming.

This section provides an introduction to REST and RESTful web services in the context of WebFOCUS.

In this chapter:

- [What Is REST?](#)
 - [What are RESTful Web Services?](#)
 - [Considerations When Using HTTP Methods: GET and POST](#)
-

What Is REST?

The REST architectural style was developed in parallel with HTTP Version 1.1, based on the existing design of HTTP Version 1.0. The largest implementation of a system conforming to the REST architectural style is the World Wide Web. REST exemplifies how the architecture of the web emerged by characterizing and constraining the macro-interactions of the four components of the web, namely origin servers, gateways, proxies and clients, without imposing limitations on the individual participants. As such, REST essentially governs the proper behavior of participants.

REST-style architectures consist of clients and servers. Clients initiate requests to servers, servers process requests and return appropriate responses. Requests and responses are built around the transfer of representations of resources. A resource can be essentially any coherent and meaningful concept that may be addressed. A representation of a resource is typically a document that captures the current or intended state of a resource.

The client begins sending requests when it is ready to make the transition to a new state. While one or more requests are outstanding, the client is considered to be in transition. The representation of each application state contains links that may be used the next time the client chooses to initiate a new state transition.

REST facilitates the transaction between web servers by allowing loose coupling between different services. REST is less strongly typed than its counterpart, SOAP. The REST language is based on the use of nouns and verbs, and has an emphasis on readability. Unlike SOAP, REST does not require XML parsing and does not require a message header to and from a service provider. This ultimately uses less bandwidth. REST error handling is also different from that used by SOAP.

What are RESTful Web Services?

A RESTful web service (also called a RESTful web API) is a web service that is implemented using HTTP and the principles of REST. It is a collection of resources with four defined aspects:

- ❑ Base URL for the web service, such as:

<http://example.com/resources>

- ❑ Internet media type of the data supported by the web service. This is usually XML, but can be any other valid Internet media type providing that it is a valid hypertext standard.
- ❑ Set of operations supported by the web service using HTTP methods (for example, GET, PUT, POST, or DELETE).
- ❑ The API must be hypertext driven.

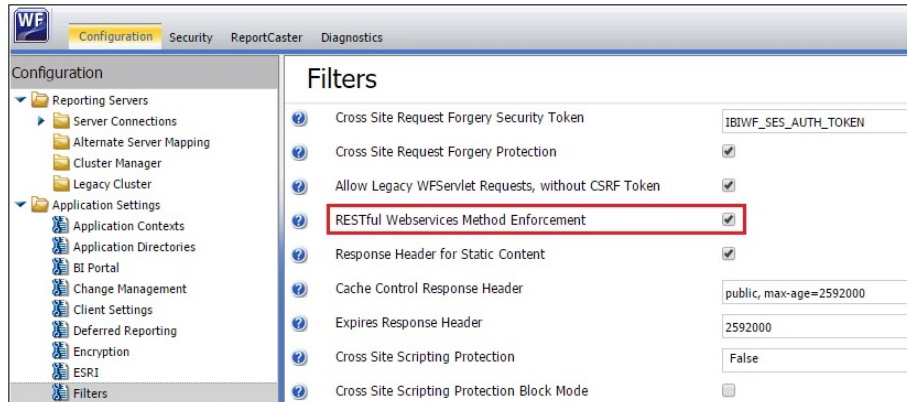
Considerations When Using HTTP Methods: GET and POST

Throughout this content, each WebFOCUS RESTful Web Service request that is documented indicates a specific HTTP method to use. This section describes several considerations that developers need to be aware of when using WebFOCUS RESTful Web Service requests with GET and POST HTTP methods.

- ❑ If the method indicated is GET, then a GET or a POST may be used. If the method indicated is POST, then only a POST may be used.
- ❑ If the WebFOCUS RESTful Web Service request uses a GET method, but the action is not allowed to be a GET, then the following error is generated:

[ERROR_INVALID_HTTP_REQUEST_TYPE](#)

Note: The RESTful Webservices Method Enforcement parameter, which can be active or inactive, controls this behavior. This parameter is located in the WebFOCUS Administration Console (under Application Settings, Filters), as shown in the following image.



- ❑ Before you use the POST method, you need to determine if the environment you are working in is accessing WebFOCUS from different web hosts, which is referred to as a cross-origin scenario. If so, then you must configure Cross-Origin Request Sharing (CORS) features and Cross-Site Request Forgery (CSRF) functionality in WebFOCUS.
- ❑ For more information on configuring and using cross-origin settings in WebFOCUS, see the *WebFOCUS Security and Administration* content.
- ❑ For more information on configuring CSRF functionality, see [Cross-Site Request Forgery \(CSRF\)](#) on page 26.
- ❑ The GET method has a limitation related to the amount of data that can be sent in a query. If you need to run a report containing a large amount of parameter data (for example, approximately 2000 bytes or 4000 in some cases), then you may need to use a POST method as an alternative.



Chapter 3

WebFOCUS Repository RESTful Web Service Requests

This section describes the format and structure of WebFOCUS authentication and Repository RESTful web service requests.

In this chapter:

- [Authenticating WebFOCUS Sign-On Requests](#)
 - [Cross-Site Request Forgery \(CSRF\)](#)
 - [Configuring Single Sign On](#)
 - [Signing Out of WebFOCUS](#)
 - [WebFOCUS Repository](#)
-

Authenticating WebFOCUS Sign-On Requests

This RESTful web service request can be used to authenticate WebFOCUS sign-on requests. The XML response that is returned indicates whether the authentication was successful or unsuccessful. The web service response also includes a *jsessionid*, which exists within the HTTP header. All subsequent WebFOCUS RESTful web services requests must have the *jsessionid* in the HTTP header. If an application is required to interact with specific WebFOCUS components (for example, WebFOCUS InfoAssist), then the *jsessionid* is also used when sending the HTTP request to open the component. This eliminates the need to reauthenticate to WebFOCUS. In addition, if you are already signed on to the WebFOCUS BI Portal, you are not required to run this sign-on request. The *jsessionid* is returned in the HTTP header after a successful sign on.

Note: By default, when using RESTful web services with Central Authentication Service (CAS) or Security Assertion Markup Language (SAML), pre-authentication attempts to access protected resources from a user who has not yet signed in to CAS or SAML will redirect the request to the CAS or SAML sign-in pages, which is an undesirable response. To change this response to an HTTP 401 (Unauthorized) status code and allow the application to initiate the authentication, you must configure a setting within the `securitysettings.xml` file to disable anonymous access, and create an HTTP request header within the RESTful application to indicate an HTTP 401 response instead of a redirect.

- ❑ Within the `securitysettings.xml` file, which is located in the `config` directory of the WebFOCUS Client installation, set:

```
anonymousAuthEnabled=false
```

- ❑ Within the RESTful application, create the following HTTP Request Header:

```
disallowSignInRedirect=true
```

HTTP Method: POST

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Body Format:

```
IBIRS_action=signOn&IBIRS_userName=Userid&IBIRS_password=Password
```

where:

Userid

Is the user ID that is required to authenticate to the WebFOCUS Repository.

Password

Is the password that is required to authenticate to the WebFOCUS Repository.

Example:

In the following example, a sign-on attempt is made to the WebFOCUS Repository with a user ID value of *admin* and a password value of *admin*.

Post Request URL:

```
http://localhost:8080/ibi_apps/rs/ibfs
```

Body:

```
IBIRS_action=signOn&IBIRS_userName=admin&IBIRS_password=admin
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
  returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSUserObject" description="" dummy="false" email=""
    fullPath="IBFS:/SSYS/USERS/admin" name="admin" password="" type="User">
    <status _jt="IBSSUserStatus" name="UNDEFINED"/>
    <groups _jt="ArrayList" size="0"/>
  </rootObject>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the sign-on attempt to the WebFOCUS Repository was successful.

The following is a sample- response trace from an authentication request:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
X-XSS-protection: 0
Set-Cookie: JSESSIONID=BD61C838569C30474977ACDE3DAD8F54; Path=/ibi_apps/; HttpOnly
Expires: Mon, 24 Sep 2012 09:12:48 GMT
Cache-Control: private
Set-Cookie: WF_SESSIONID=1932062683094412614; Path=/
IBI_Messages: 2
IBI_Message1: (IBFS10000) SUCCESS
IBI_Message2: <IBIWF_SES_AUTH_TOKEN>=<null>
Content-Type: text/xml;charset=iso-8859-1
Transfer-Encoding: chunked
Date: Mon, 24 Sep 2012 09:07:48 GMT
205
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc_jt="IBFSResponseObject" language="EN" name="signOn" returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSUserObject" description="" dummy="false" email=""
    fullPath="IBFS:/SSYS/USERS/admin" name="admin" password=""
    rsPath="/ibi_apps/rs/ibfs/SSYS/USERS/admin" type="User">
    <status _jt="IBSSUserStatus" name="UNDEFINED"/>
    <groups_jt="ArrayList" size="0"/>
  </rootObject>
</ibfsrpc>
```

The Set-Cookie parameter that appears in line four of this sample establishes the Session ID for users and must be included in all subsequent request messages during the session.

The following is a sample trace of a subsequent request:

```
GET http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository?IBIRS_action=get
HTTP/1.1
Host: localhost:8080
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:15.0) Gecko/20100101
Firefox/15.0.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Cookie: JSESSIONID=BD61C838569C30474977ACDE3DAD8F54;
wcNewPreference=1963156A6FD0D3C6EE81F2C992ED527D;
WF_SESSIONID=1932062683094412614
```

Cross-Site Request Forgery (CSRF)

A Cross-Site Request Forgery (CSRF), also known as a one-click attack or session riding, is a type of malicious exploit of a website whereby unauthorized commands are transmitted from a user that the website trusts.

To prevent these types of attacks, WebFOCUS must be configured to use CSRF token protection. Under this configuration, a CSRF token is generated every time the WebFOCUS RESTful Web Services authentication request (*IBIRS_action=signOn*) is run, and the CSRF token, containing a token name and a token value, is returned within the response.

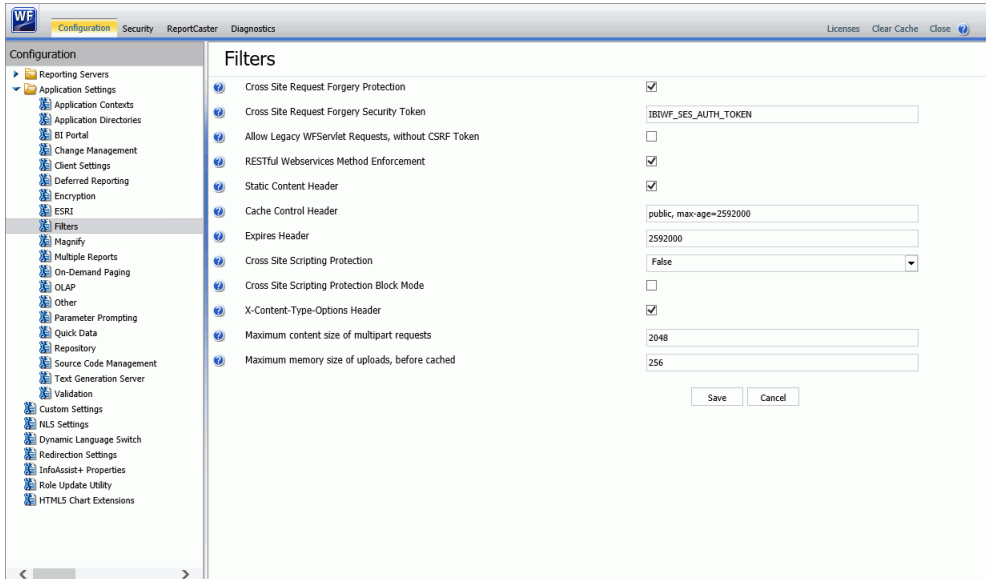
The CSRF token must be sent as a parameter for any HTTP POST request. Otherwise, a 403 HTTP return code will occur and a CSRF error message will be logged in the *websecurity.yyyy-mm-dd.log* file.

The CSRF token prevents attacks by providing a value that was generated randomly and then stored in the web session of the actual authenticated user. The client can then check for this value when processing all remaining requests and responses during the session, and can confirm that the request or response is legitimate because it contains the value assigned to that session. Requests or responses that do not contain a CSRF token, or that contain a CSRF token with a name or value that does not match the name or value assigned by the server, are rejected as invalid.

CSRF token protection is supported in Kerberos and SSO environments, but to obtain the delivery of a CSRF token for pre-authenticated users, an explicit *IBIRS_action=signOn* request that contains the ID of the pre-authenticated user exactly as it appears in the database of the authentication application and no password must be added to the initial sign in transaction. For more information, see [Configuring Single Sign On](#) on page 30.

CSRF protection is enabled by default in the WebFOCUS Administration Console. To confirm and also view the related settings, on the Configuration tab, under the Application Settings folder, click *Filters*.

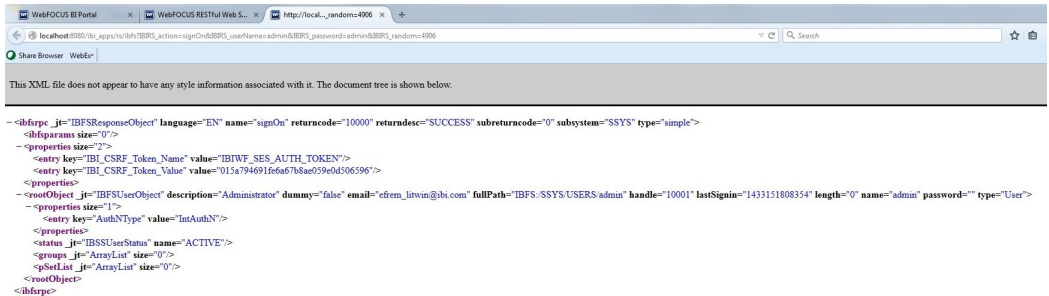
The Filters pane opens, as shown in the following image.



Note that the *Cross Site Request Forgery Protection* check box is selected.

Obtaining a CSRF Token

The CSRF token is returned in the response of the WebFOCUS RESTful Web Services authentication action (*IBIRS_action=signOn*), as shown in the following image.



In this example, the name of the CSRF token is *IBIWF_SES_AUTH_TOKEN* and the value for the CSRF token is *015a794691fe6a67b8ae059e0d506596*.

The name of the CSRF token is taken from the value assigned to the Cross Site Request Forgery Security Token (IBI_CSRF_TOKEN_NAME) setting located on the Filters pane of the Administration Console Configuration tab. By default, the value assigned to this setting is IBIWF_SES_AUTH_TOKEN. However, if your configuration assigns a different value to this setting, WebFOCUS returns a different CSRF token name in the response to the sign-in request message.

Passing a CSRF Token

If WebFOCUS is configured to use CSRF token protection, then the CSRF token is passed as a parameter within the body of the POST request for all actions that require a CSRF token.

Example:

The following example shows the WebFOCUS RESTful Web Service request to add a user. This request includes a CSRF token, as shown in the body of the request in the following example.

Post Request URL:

```
http://localhost:8080/ibi_apps/rs
```

Body:

```
IBIRS_path=/SSYS/USERS/testuser&IBIRS_action=put
&IBIRS_object=<object _jt="IBFSUserObject" description="Test Userid"
email="restid@informationbuilders.com" password="rest" type="User">
<status _jt="IBSSUserStatus" name="ACTIVE"/>
</object>
&IBIRS_service=ibfs&IBIWF_SES_AUTH_TOKEN=015a794691fe6a67b8ae059e0d506596
```

Typically, the response returns XML code identifying the entry for the new user as shown in the following example.

```
<ibfsrpc _jt="IBFSResponseObject" language="en_US" name="put"
returncode="10000" returndesc="SUCCESS" subreturncode="0" type="simple">
  <ibfsparams size="5">
    <entry key="IBIRS_object" value="****"/>
    <entry key="IBIRS_private" value="__null"/>
    <entry key="IBIRS_replace" value="true"/>
    <entry key="IBIRS_path" value="/SSYS/USERS/testuser"/>
    <entry key="IBIRS_args" value="__null"/>
  </ibfsparams>
  <rootobject _jt="IBFSUserObject" description="Test Userid" dummy="false"
email="restid@informationbuilders.com" fullPath="IBFS:/SSYS/USERS/testuser"
handle="1784804352" length="0" name="testuser" nameSpace="DB" policy="/+//
+8P///30f/e///+/////////v+AAAA" rsPath="/ibi_apps/rs/ibfs/SSYS/USERS/
testuser" thumbPath="/ibi_apps/ibi_html/ibi_images/file_type/unknown.svg"
type="User" userStatusDisplay="Active">
  <status _jt="IBSSUserStatus" name="ACTIVE"/>
  <groups _jt="ArrayList" size="0"/>
  <pSetList _jt="ArrayList" size="0"/>
</rootObject>
</ibfsrpc>
```

However, if the CSRF token is not sent, or if an invalid CSRF token is sent in requests that require a CSRF token, then the following error message will be returned in the response:

```
<!DOCTYPE HTML>
<HTML>
<HEAD>
<title>403 - Access Denied</title>
</HEAD>
<body style="background-color:#d9e1f2; margin:0;">
  <div align="center" style="position:relative;font-family:Arial;top:
172px;font-size:25pt;">403 - Access Denied</div>
  <div align="center" style="position:relative;top:178px;font-size:9pt;font-
family:Tahoma;color: #485059;">You are not authorized to view this page</
div>
</body>
</HTML>
```

Configuring Single Sign On

WebFOCUS security can be configured to integrate with software service vendors, such as IBM Tivoli® Access Manager and Computer Associates (CA) SiteMinder®. In addition, authentication methodologies, such as Basic Authentication, Integrated Windows Authentication (IWA), and Kerberos, can also be configured with WebFOCUS security. When WebFOCUS security is configured in this manner, the RESTful web service request to authenticate WebFOCUS is not required. For more information, see [Authenticating WebFOCUS Sign-On Requests](#) on page 23.

However, when working in environments that support Single Sign-On authentication, an `IBIRS_action=signOn` request must be included in the initial sign-in transaction to enable WebFOCUS to support the use of CSRF tokens, as shown in the following example.

```
var IBIRS_action = "signOn";  
var IBIRS_userName = "user_id";  
var IBIRS_password = " ";
```

where:

user_id

Is the ID of the user as recorded in the single sign-on provider.

The `signOn` request must include the ID of the User sending the request and a blank password, even though these values are not needed for authentication.

When WebFOCUS returns a CSRF token to a user, that token needs to be added to all HTTP POST requests originating from that user during the remainder of the session.

A session identifier named `WF-JSESSION ID`, by default, is also returned to the user within a cookie that is included in the header of the response message to the `signOn` request, as shown in the following example.

```
Set-Cookie: WF-JSESSIONID=0000v6lbcwkcbjsF-XoA1s3IAHe:-1;
```

This cookie identifies the user to the server, and to prevent errors, it must be included in the HTTP header of all subsequent RESTful web service request messages delivered from that user during the session.

After receiving a response to the first RESTful web service request, the client application must parse the response header to retrieve the cookies and send them to subsequent RESTful web service requests. The reason for this is to reuse the session in the application server.

Example 1: Adapt the Initial Sign In Request for Single Sign On Environments

The following example shows how to create a signOn request in a single sign on environment.

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
  <meta charset="utf-8" />
  <script type="text/javascript" src="http://code.
    jquery.com/jquery-3.1.0.js"> </script>
  <script type='text/javascript' src="http://cdnjs.cloudflare.com/ajax/libs/
    jquery-ajaxtransport-xdomainrequest/1.0.1/jquery.xdomainrequest.min.js"></
    script>

  <script type="text/javascript">
    var csrf_name;
    var csrf_value;
    var frameToBeWorkedOn = "#AjaxPlaceholder";
    var contentType = "application/x-www-form-urlencoded; charset=utf-8";

    //To run with preauthentication, use the "Modify Headers" add-in in the
    //browser to set a request header of SM_USER with a value of "rest"
    //which is the userid in Security Center. Then in WebFOCUS Admin Console -
    >
    //Security tab, turn off all authentication schemes except for
    //Preauthentication
    //Use SM_USER and keep all the defaults **** Make sure to START the Modify
    //Headers add-in or the header variable is not sent

    // $(document).ready(function (IBIRS_action, IBIRS_userName,
    IBIRS_password) {
      $(document).ready(function (IBIRS_action, IBIRS_userName) {

        if (window.XDomainRequest)
          contentType = "text/plain";

        var webMethod = "http://as8200.ibi.com:8080/ibi_apps/rs";
        var IBIRS_action = "signOn";
        var IBIRS_userName = "rest";
        var IBIRS_password = "";
        var parameters = 'IBIRS_action=' + IBIRS_action +
          '&IBIRS_userName=' +
          IBIRS_userName + '&IBIRS_password=' + IBIRS_password;
        var parameters = 'IBIRS_action=' + IBIRS_action +
          '&IBIRS_userName='
          + IBIRS_userName;
```



```

$.ajax({
  type: "POST",
  url: webMethod,
  data: parameters,
  dataType: "xml",
  xhrFields: {
    withCredentials: true
  },
  crossDomain: true,
  contentType: contentType,
  success: xmlParser,
  error: function(jqXHR, textStatus, errorThrown)
  {
    alert("You can not send Cross Domain AJAX requests: " +
      errorThrown);
  }
});
});

function xmlParser(xml) {

  $(xml).find("entry").each(function () {
    if ($(this).attr("key") == "IBI_CSRF_Token_Name") {
      csrf_name = $(this).attr("value");
    }
    if ($(this).attr("key") == "IBI_CSRF_Token_Value") {
      csrf_value = $(this).attr("value");
    }
  });

  runReport();
}

```

Example 2: SiteMinder (Initial Request)

When working with SiteMinder, the SMSESSION cookie must be passed in the RESTful web service request header in addition to the cookie containing the WF-JSESSION ID.

Request:

```
GET http://host:port/ibi_apps/rs?IBIRS_action=TEST HTTP/1.1
Host: host:port
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:20.0) Gecko/20100101 Firefox/20.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Cookie: SMSESSION=9XYcYZnboGIIfmHEeeZJ8qSQY8Q86jN/WYZ/tco/xYuXM0hNVSi4VI0kDKLq/C0
RHARUYd/J6og1b5w1M+I2alSoUJz8m28cUj13Pt221ubduHvaAmEAWHh86lQhUmLc/yaе552m
YoURSzhz2LexeE+7KgeK8fFvTbjX12DXHPbv8vpkas80NeYnaqJbs4Td4jbt0A0Lf92k2K5H
87CDNgr+lT6iWAVEWo972+eSd7t+/iD3MDaadal7CnTlnUk1BYBTQxHnk8tg3eHUxy61Lqc7M
K/xmcf+f27S4acueluk2UAeGLG9b+qkmQ8qZ9fZ/equ5tpUL3LZlRwsq9Zf/XXgYM/zUq6f29
mJ0llsi9XU/KIO3TyPMiBT+g j3bGsK3H5Zw8KuqCJafSuqG9IzohJFtNuOokCp6Qrm2DtGXhn
fiuYKmwMd006acFh6kVNhMsNEeiTZ6Uo2spccoHJ8I1MA9F7WkF1/yvdghftdYcD6dKIGYFO7
biKfPhAy/rjtjd23HP138V5jmMTz3A0LeLvjnlsGbxNoTKg/PVf3NPM1o5lsltTwvKYLZbx87
WotlpOVhiAslwre/2UW7kHHIpeX1N3VP4E3ZmYDCXuxX+aJDwGEUzzAbi9uxu/aVDRMRSJY5R
LggW8dyugcfBagJ94+n8WvC8tsG7nnlVDEewQNbay7w3lrWp0SYVd227KjfdSt1N9eTs08vKD
sneKjseSchZV0hCL62lzh1JwAaJg3FJNMpnIGG6MmrJ66RC4AhMaKWJgY1p0Li414V3ne1J29
YfnKE7PAVvy9jfn7iZO8vWT5EunMYPrNgsMH+dZ6atK5xx5lSCO76uYtEislwScoCQvgV6kZi
RLyLwPv03kWeINwAkyM3QdmqAWEutR4L7NyTL4bThU5nXuScRCrQ1+EiqOxPKCBh
Connection: keep-alive
```

Response:

```

HTTP/1.1 200 OK
Date: Mon, 06 May 2013 13:38:07 GMT
Server: Apache/2.0.58 (Unix) mod_ssl/2.0.58 OpenSSL/0.9.8e-fips-rhel5
Set-Cookie: SMSESSION=jNji3BS1Zavfl0YRdpNd50mdUsBGBaoaD8DcoIqG/EnvCE2/VqlM3wAcPfr25I0
JZHmLoewUFMrz60pSwkycBk1MQLDWv2LkQVa/lESzr9PqzONyiSwXDPHwa5MXdgpmsH58b2aA
f3x1lpKZ/EX3D6VDPaIrrmnZE4LY7GK5YD5+wr/hVDBVWKmVlphbefCjDvlfanfUCZmau8gd1N
6Csxv52ULat8QBoRmXYh+iDxDpCPqDM4Nc8z3TiVeHhsRyE+7xsAoY+22+E2Vkj8EDv/hCdL
ar9VS+nBtPALuN/Otze1C/ZRDi9X90yL3++ecsrpLW+ioqRznh7c043URUNqoPz9M3Ea8uDJO
RSdeQ9QeoAZ8x+4y9jPEMDVdBSJqE7EZlm6d6BMA DPDAUPPP+BYMwx/EHSzM6rbpH+NJT6GOG
M9gkvLhH31BjibJZf2VvDPsgzHzIONT1xDJgGcyLTiXAt8m17ufvphnJZbpFtMi0WKfHm16Rz
TwZ+9KvPW2Toem35zhFXU2gFXE/3lgj9sq7MKmihdXe1D022Rd0j7ti99PZg8Q08wsVaHh4P8
8/ITTy/DrTFqMhdu97YUEW7bAHLKK6OPZtpDWCqix3T9/+ZA6MICdSuWRzX1bD2sXQs/zIsga
e/K2RHkNTSMA0bKzR+cFUsDzooM5yWApAXvYe/WsB59jOQYrEidG4//f1Q7MT7F8DnTnVDjWs
j9JlgLvewdiJWVgP+knPnaiR9oZlGseqCjAuCbbxFcpVhKprrrx/urqNzwm9Yz0xKcd8jvXA8
lrT0yiN+jarm/nHfyjJLYt1fBOuhXploQn7TR7ZixA4n57R897LzbmZK6CsyreFJ11UbiyqSb
X40M0qx+HHJ3ev7D8t+Rbdn/5UdHzGFCi1S2ZHPkbe+gO9H1OwxNSmnwIDEUGjQUra7vmvZaU
5cUeAXFHvCUTKVC811vtdSd+eAaLau5THQ11PylRSTQ0f/DwxU1Mon6EZTkRLLxR+2mvnpN6P
wj; path=/; domain=.ibi.com
X-XSS-protection: 0
Expires: Mon, 06 May 2013 13:43:07 GMT
Cache-Control: private
Set-Cookie: WF-JSESSIONID=0000v6lbcwkcbsf-XoA1s3IAHe:-1; Path=/ibi_apps

Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Transfer-Encoding: chunked
Content-Type: text/html; charset=utf-8
Content-Language: en-US

```

The SiteMinder SMSESSION Cookie identifies the authenticated session. For more information, see:

<https://docops.ca.com/ca-single-sign-on/12-52-sp1/en/configuring/web-agent-configuration/session-protection/session-cookie-management>

Example 3: SiteMinder (Subsequent Requests)

For all subsequent requests, the SMSESSION cookie, as well as the session cookie retrieved in the initial RESTful web service request, must be passed in the RESTful web service request header, as shown in the following example.

```
GET http://host:port/ibi_apps/rs/ibfs/WFC/Repository?IBIRS_path=%2FWFC
%2FRepository&IBIRS_action=get&IBIRS_args=__null HTTP/1.1
Host: host:port
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:20.0) Gecko/20100101 Firefox/20.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://host:port/ibi_apps/rs/ibfs?IBIRS_action=TEST
Cookie: SMSESSION=jNji3BSlZavf10YRdpNd50mdUsBGBaoaD8DCoIqG/EnvCE2/VqLM3wAcPFr25IO
JZHmLoewUFMrz60pSwkycBklMQLDWv2LkQVa/LESzr9PqzONYiSwXDPHwa5MXdgpmsH58b2aA
f3x1lpKZ/EX3D6VDPaIrRmnZE4LY7GK5YD5+wr/hVDBVWKmVlphbefcJdVlanfUCZmau8gdLN
6Csxv52ULat8QBoRmXYh+iDxDpCpQDM4Nc8z3TiVeHhsRyE+7xsAoY+22+E2VkJ8EDv/hCdL
ar9VS+nBtPALuN/Otze1C/ZRDi9X90yL3++ecsrpLW+ioqRznh7c043URUNqoPz9M3Ea8uDJO
RSdeQ9QeoAZ8x+4y9jPEMDVdBSJqE7EZlm6d6BMA DPDAUPPP+BYMwx/EHSzM6rbpH+NJT6GOG
M9gkvLhH31BjibJZf2VvDPsgzHzIONT1xDJgGcyLTiXAt8m17ufvphnJZbpFtMi0WKfHML6Rz
TwZ+9KvPW2Toem35zhFXU2gFxE/3lgj9sq7MKmihdXe1D022Rd0j7ti99PZg8Q08wsVaHh4P8
8/ITTy/DrTFqMhdu97YUEW7bAHLKK6OPZtpDWCqix3T9/+ZA6MICdSuWRzX1bD2sXQs/zIsga
e/K2RHkNTSMA0bKzR+cFUsDzooM5yWApAXvYe/WsB59jOQYrEIdG4//f1Q7MT7F8DnTnVDjWs
j9JlgLvewdiJWVgP+knPnaiR9oZlGseqCjAuCbbxFcpVhKprrx/urqNzwm9Yz0xKcd8jvXA8
lrT0yiN+jarm/nHfyjJLYt1fBOuhXploQn7TR7ZixA4n57R897LzBmZK6CsyreFJ11UbiyqSb
X40M0qx+HHJ3eV7D8t+Rbdn/5UdHzGFCi1S2ZHPkbe+gO9H1OwxNSmnwIDEUGjQUra7vmvZaU
5cUeAXFHvCUTKVC81lvtDsd+eAaLau5THQ1lPylRSTQ0f/DwxU1Mon6EZTkRLLxR+2mvpN6P
wj; WF-JSESSIONID=0000v6lbcwkcbjsF-XoA1s3IAHe:-1; Connection: keep-alive
```

Signing Out of WebFOCUS

This RESTful web service request can be used to sign out of WebFOCUS. Therefore, all subsequent WebFOCUS RESTful web services requests will not run successfully once signed out. If a WebFOCUS session exists within the same browser session, this session will also be signed out.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Body Format:

```
IBIRS_action=signOff
```

Example:

In the following example, a sign-out request is made to WebFOCUS.

Post Request URL:

```
http://localhost:8080/ibi_apps/rs/ibfs
```

Body:

```
IBIRS_action=signOff
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="signOff"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="string">OK</rootObject>
</ibfsrpc>
```

WebFOCUS Repository

This section describes the format and structure of RESTful web service requests that are used for a variety of WebFOCUS Repository tasks.

Creating and Updating a Folder

This RESTful web service request can be used to create and update a folder within the WebFOCUS Repository.

HTTP Method: POST

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder to be created. If the folder being created is a subfolder of an existing folder, then the existing folder name is also included in the REST URL. This shows the path to the folder being created. For example, ExistingFolder/FolderName.

Body Format:

IBIRS_action=put&IBIRS_object=Object&IBIRS_private=MakeFolderPrivate&IBIRS_replace=ReplaceFolderProperties

where:

Object

Is the XML object defining the attributes for the folder using the following format:

```
<object _jt="IBFSMRObject"
container="true" description="FolderDescription" summary="Summary"
appName="AppList">
<properties size="numberOfProperties">
<entry key="propertyN" />
</properties>
</object>
```

where:

FolderDescription

Is a description of the folder being created.

Summary

Is a brief description describing the contents of the folder.

AppList (Optional)

List of applications used in the search path. For example:

```
appName="ibisamp ibidemo"
```

properties (Optional)

numberOfProperties

Is the number of properties that are to be applied to the folder.

propertyN

The property that is applied to the folder. Each property exists with an opening and closing *entry* tag. For example:

```
<entry key="autogenmyreports" />
```

Valid properties:

- autogenmyreports.** Automatically creates My Content folders.
- hidden.** Do not show in the list of folders.

MakeFolderPrivate

Determines whether to make a folder private. Specify *true* or *false*. By default, this attribute is set to *true*.

ReplaceFolderProperties

Determines whether the properties of the folder (for example, FolderDescription and Summary) can be updated.

Specify one of the following:

- true.** Update the properties of the folder. To update the properties of the folder, the existing properties must be retrieved. The retrieved XML object would then be modified and used as input. The following REST URL retrieves the existing properties for a folder:

```
http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName?
IBIRS_action=get
```

- false.** Do not update the properties of the folder.

Example 1:

In the following example, a folder called Financial_Reports is created, which has SEC Filings as the description.

POST Request URL:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports
```

Body:

```
IBIRS_action=put&IBIRS_object=<object _jt="IBFSMRObject" container="true"
  description="SEC Filings" summary="Quarterly and Yearly Financial Reports reported
to the Securities and Exchange Commission">
  </object>&IBIRS_replace=false
```

Response:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSMRObject" binary="false" container="true"
createdBy="admin"
  createdOn="1345146734216" defaultLng="en_US" description="SEC Filings"
dummy="false"
  effectiveRSName="EDASERVE" fullPath="IBFS:/WFC/Repository/
Financial_Reports"
  handle="75d099c0_163a_46d8_ba25_ec0be965b15d"
lastModified="1345146734216"
  lastaccessBy="admin" lastaccessOn="1345146734216" lastmodBy="admin"
length="0"
  name="Financial_Reports" ownerId="10001" ownerName="admin"
ownerType="U"
  policy="//v+f////////9//////////+AAAAA" returnedLng="en_US"
summary="Quarterly and Yearly Financial Reports reported to the
Securities and Exchange Commission"
type="MRFolder">
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="SEC Filings"/>
        <item _jt="string" index="1" value="Quarterly and Yearly Financial
Reports reported to the Securities and Exchange Commission"/>
      </value>
    </entry>
  </nlsValues>
  <properties size="0"/>
</rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the folder was successfully created.

Example 2:

In the following example, a folder called `Financial_Reports` is updated with `Financial Quarterly-Yearly Reports` set as the new description.

The following REST URL retrieves the existing properties for the `Financial_Reports` folder:

```

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports?
IBIRS_action=get

```

POST Request URL:

```

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports

```


Body:

```

IBIRS_action=put&IBIRS_object=
<rootObject _jt="IBFSMRObject" binary="false" container="true"
  createdBy="admin" createdOn="1349964405620" defaultLng="en_US"
  description="Financial Quarterly-Yearly Reports" dummy="false"
effectiveRSName="EDASERVE"
  fullPath="IBFS:/WFC/Repository/Financial_Reports"
handle="5d81bab8_7db7_40c9_96b9_df2b00ce3278"
  lastModified="1349964405620" lastaccessBy="admin"
lastaccessOn="1349969821584" lastmodBy="admin"
  length="0" name="Financial_Reports" ownerId="10001" ownerName="admin"
ownerType="U"
  policy="//3/D///9+f/////f/////////8AAAA=" returnedLng="en_US"
  rsPath="/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports"
  summary="Quarterly and Yearly Financial Reports reported to the
Securities and Exchange Commission"
  type="MRFolder">
  <children _jt="ArrayList" size="0"/>
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="Financial Quarterly-Yearly
Reports"/>
        <item _jt="string" index="1"
          value="Quarterly and Yearly Financial Reports reported to the
Securities and Exchange Commission"/>
      </value>
    </entry>
  </nlsValues>
  <properties size="0"/>
</rootObject>&IBIRS_replace=true

```

Response:

If the value for the *returncode* attribute in the XML response is 10000, then the folder was successfully updated.

Deleting a Folder

This RESTful web service request can be used to delete a folder or subfolder within the WebFOCUS Repository.

HTTP Method: DELETE

REST URL Format:

```

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName?
IBIRS_action=delete

```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder to be deleted. If the folder being deleted is a subfolder, then the folder above the subfolder is also included in the REST URL. This shows the path to the folder being deleted. For example, ParentFolderName/FolderName.

Example:

In the following example, the Manufacturing_Reports folder is deleted from the Car_Reports folder, which is within the RESTful_Web_Services folder.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/Manufacturing_Reports?IBIRS_action=delete
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
  returncode="10000" returndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfparams size="0"/>
  <rootObject _jt="IBFSMRObject" binary="false" container="true"
    createdBy="admin" createdOn="1345149829421"
    defaultLng="en_US" description="Manufacturing Reports" dummy="false"
    effectiveRSName="EDASERVE"
    fullPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/
      Manufacturing_Reports"
    handle="bb7ea628_2068_4d1c_b3cb_80555a30d53f"
    lastModified="1345149829421" lastaccessBy="admin"
    lastaccessOn="1345152035853" lastmodBy="admin" length="0"
    name="Manufacturing_Reports"
    returnedLng="en_US" type="MRFolder">
    <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
```

```

    <key _jt="string" value="en_US"/>
    <value _jt="ArrayList" size="2">
      <item _jt="string" index="0" value="Manufacturing Reports"/>
    </value>
  </entry>
</nlsValues>
<properties size="0"/>
</rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the folder was successfully deleted.

Deleting a WebFOCUS Repository Report

This RESTful web service request can be used to delete a report from the WebFOCUS Repository.

HTTP Method: DELETE

REST URL Format:

```

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ReportName?
IBIRS_action=delete

```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder used for the stored WebFOCUS report. If the folder used for the stored WebFOCUS report exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

ReportName

Is the name of the WebFOCUS report to delete, which must have a .fex extension.

Example:

In the following example, the Income_Statement_March_2010 report is deleted from the Quarterly folder, which is within the Financial_Reports folder.

Request:

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports/Quarterly/Income_Statement_March_2010.fex?IBIRS_action=delete

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSMRObject" appName="ibisamp" binary="false"
createdBy="admin"
  createdOn="1345218342649" defaultLng="en_US" description="Income
Statement - March 2010"
  dummy="false" extension="fex"
fullPath="IBFS:/WFC/Repository/Financial_Reports/Quarterly/
Income_Statement_March_2010.fex"
  handle="7fefd079_cc95_4b8e_a99e_6d2f71e23020" inheritedPrivacy="true"
lastModified="1345218342649"
  lastaccessBy="admin" lastaccessOn="1345219257305" lastmodBy="admin"
length="5231"
  name="Income_Statement_March_2010.fex" ownerId="10001"
ownerName="admin" ownerType="U"
  policy="//v+f////////f9////////9////////+AAAAA" returnedLng="en_US"
type="FexFile">
  <content _jt="IBFSByteContent"
char_set="Cp1252">LSpEbyBub3QgZGVsZXRlIG9yIG1vZGlmeSB0aGUgY2
9tbWVudHMgYmV...1NVTU1BUlkuUVVPVEVEU1RSSU5HLCakCkVORFNWUx
FCkVORAoKLVJVTgo=
  </content>
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="Income Statement - March 2010"/>
      </value>
    </entry>
  </nlsValues>
  <properties size="1">
    <entry key="tool" value="infoAssist,report,IAFull"/>
  </properties>
</rootObject>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the report was successfully deleted.

Listing Folders and Subfolders

This RESTful web service request can be used to retrieve a list of folders and subfolders within the WebFOCUS Repository.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName?IBIRS_action=get`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder to be used in retrieving a list of its subfolders. To obtain a list of folders, *FolderName* should not be included in the REST URL. To obtain additional levels of subfolders for a particular subfolder, the path to the subfolder name must be included in the REST URL. For example, ParentFolderName/*FolderName*.

Example 1:

In the following example, a list of folders is retrieved.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository?IBIRS_action=get`

Response:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSMRObject" binary="false" container="true"
createdBy="WebFOCUS"
  createdOn="1344536982043" defaultLng="en_US" description="Content"
dummy="false"
  effectiveRSName="EDASERVE" expireDate="1344536982047" externalId=""
  fullPath="IBFS:/WFC/Repository" handle="000000000001"
lastModified="1344536982047"
  lastaccessBy="admin" lastaccessOn="1345146849357" lastmodBy="WebFOCUS"
length="0"
  name="Repository" policy="///+f////////9//////////+AAAAA"
returnedLng="en_US"
  summary="Content Root" type="MRRepository">
  <children _jt="ArrayList" size="3">
    <item _jt="IBFSMRObject" binary="false" container="true"
createdBy="WebFOCUS"
  createdOn="1344536982083" defaultLng="en_US" description="Public"
dummy="false"
  effectiveRSName="EDASERVE" expireDate="1344536982083" externalId=""
  fullPath="IBFS:/WFC/Repository/Public" handle="000000000004" index="0"
lastModified="1344536982083" lastaccessBy="admin"
lastaccessOn="1344957209010"
  lastmodBy="WebFOCUS" length="0" name="Public" parent="Repository"
  policy="///+f////////9//////////+AAAAA" returnedLng="en_US"
  summary="Public Folder" type="MRFolder">
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
  <entry>
  <key _jt="string" value="en_US"/>
  <value _jt="ArrayList" size="2">
    <item _jt="string" index="0" value="Public"/>
    <item _jt="string" index="1" value="Public Folder"/>
  </value>

```

```

    </entry>
  </nlsValues>
  <properties size="0"/>
</item>
<item _jt="IBFSMRObject" binary="false" container="true"
createdBy="admin"
  createdOn="1345146734216" defaultLng="en_US" description="SEC
Filings" dummy="false"
  effectiveRSName="EDASERVE" fullPath="IBFS:/WFC/Repository/
Financial_Reports"
  handle="75d099c0_163a_46d8_ba25_ec0be965b15d" index="1"
lastModified="1345146734216"
  lastaccessBy="admin" lastaccessOn="1345146755132" lastmodBy="admin"
length="0"
  name="Financial_Reports" ownerId="10001" ownerName="admin"
ownerType="U"
  parent="Repository" policy="//v+f////////9//////////+AAAAA"
  returnedLng="en_US"
  summary="Quarterly and Yearly Financial Reports reported to the
Securities and Exchange Commission"
  type="MRFolder">
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="SEC Filings"/>
        <item _jt="string" index="1"
          value="Quarterly and Yearly Financial Reports reported to the
Securities and Exchange Commission"/>
      </value>
    </entry>
  </nlsValues>
  <properties size="0"/>
</item>
<item _jt="IBFSMRObject" binary="false" container="true"
createdBy="admin" createdOn="1344607303673"
  defaultLng="en_US" description="RESTful Web Services" dummy="false"
effectiveRSName="EDASERVE"
  fullPath="IBFS:/WFC/Repository/RESTful_Web_Services"
handle="ac08f200_d2f2_4ab6_9b60_b62d8f2ad345"
  index="2" lastModified="1344957300737" lastaccessBy="admin"
lastaccessOn="1345146071751"
  lastmodBy="admin" length="0" name="RESTful_Web_Services"
ownerId="10001" ownerName="admin"
  ownerType="U" parent="Repository" policy="//v+f////////
9//////////+AAAAA" returnedLng="en_US"
  summary="For documenting RESTful Web Services" type="MRFolder">

```

```

<nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
  <entry>
    <key _jt="string" value="en_US"/>
    <value _jt="ArrayList" size="2">
      <item _jt="string" index="0" value="RESTful Web Services"/>
      <item _jt="string" index="1" value="For documenting RESTful Web
Services"/>
    </value>
  </entry>
</nlsValues>
<properties size="0"/>
</item>
</children>
<nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
  <entry>
    <key _jt="string" value="en_US"/>
    <value _jt="ArrayList" size="2">
      <item _jt="string" index="0" value="Content"/>
      <item _jt="string" index="1" value="Content Root"/>
    </value>
  </entry>
</nlsValues>
<properties size="0"/>
</rootObject>
</ibfsrpc>

```

Each folder definition is defined within the opening and closing *item* tag. The *type* attribute for a folder is *MRFolder*. The *name* attribute defines the name of the folder. The *description* attribute defines the title for the folder. The *summary* attribute defines a brief description for the contents of the folder.

In this example, there are three folders, as listed in the following table.

Folder Name	Title	Summary
Public	Public	Public Folder.
RESTful_Web_Services	RESTful Web Services	For documenting RESTful Web Services.
Financial_Reports	SEC Filings	Quarterly and Yearly Financial Reports reported to the Securities and Exchange Commission.

Example 2:

In the following example, a list of subfolders for the SEC Filings (Financial_Reports) folder is retrieved.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports?
IBIRS_action=get
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSMRObject" binary="false" container="true"
createdBy="admin"
  createdOn="1345146734216" defaultLng="en_US" description="SEC Filings"
dummy="false"
  effectiveRSName="EDASERVE" fullPath="IBFS:/WFC/Repository/
Financial_Reports"
  handle="75d099c0_163a_46d8_ba25_ec0be965b15d"
lastModified="1345146734216"
  lastaccessBy="admin" lastaccessOn="1345147040831" lastmodBy="admin"
length="0"
  name="Financial_Reports" ownerId="10001" ownerName="admin"
ownerType="U"
  policy="//v+f////////9//////////+AAAAA" returnedLng="en_US"
  summary="Quarterly and Yearly Financial Reports reported to the
Securities and Exchange Commission"
  type="MRFolder">
  <children _jt="ArrayList" size="1">
  <item _jt="IBFSMRObject" binary="false" container="true"
createdBy="admin" createdOn="1345147005204"
  defaultLng="en_US" description="Quarterly" dummy="false"
effectiveRSName="EDASERVE"
  fullPath="IBFS:/WFC/Repository/Financial_Reports/Quarterly"
handle="a0cfcde1_fb34_4b07_b20d_4144094ec5c2"
  index="0" inheritedPrivacy="true" lastModified="1345147005204"
lastaccessBy="admin"
  lastaccessOn="1345147013034" lastmodBy="admin" length="0"
name="Quarterly" ownerId="10001"
  ownerName="admin" ownerType="U" parent="Financial_Reports" policy="//v
+f////////f9////////9//////////+AAAAA"
  returnedLng="en_US"
  summary="Quarterly Financial Reports reported to the Securities and
```

```

Exchange Commission" type="MRFolder">
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="Quarterly"/>
        <item _jt="string" index="1"
          value="Quarterly Financial Reports reported to the Securities
and Exchange Commission"/>
      </value>
    </entry>
  </nlsValues>
  <properties size="0"/>
</item>
</children>
<nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
  <entry>
    <key _jt="string" value="en_US"/>
    <value _jt="ArrayList" size="2">
      <item _jt="string" index="0" value="SEC Filings"/>
      <item _jt="string" index="1"
        value="Quarterly and Yearly Financial Reports reported to the
Securities and Exchange Commission"/>
    </value>
  </entry>
</nlsValues>
  <properties size="0"/>
</rootObject>
</ibfsrpc>

```

Each folder definition is defined within the opening and closing *item* tag. The *name* attribute defines the name of the folder. The *description* attribute defines the title for the folder. The *summary* attribute defines a brief description for the contents of the folder.

In this example, there is one subfolder, as listed in the following table.

Subfolder Name	Title	Summary
Quarterly	Quarterly	Quarterly Financial Reports reported to the Securities and Exchange Commission.

Listing Reports, Schedules, and Library Content Within the WebFOCUS Repository

This RESTful web service request can be used to retrieve the content list within a folder. The content can be additional subfolders, WebFOCUS reports, ReportCaster Schedules, and Library Content.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName?
IBIRS_action=get
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder to be used in retrieving the content list. If the content exists in a subfolder, then the path to the subfolder name must be included in the REST URL. For example, ParentFolderName/FolderName.

Example:

In the following example, a content list for the Car_Reports folder is retrieved. The Car_Reports folder is a subfolder of the RESTful_Web_Services folder.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_
Web_Services/Car_Reports?IBIRS_action=get
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
  type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSMRObject" binary="false" container="true"
createdBy="admin"
  createdOn="1344607319557" defaultLng="en_US" description="Car Reports"
dummy="false"
  effectiveRSName="EDASERVE" fullPath="IBFS:/WFC/Repository/
RESTful_Web_Services/Car_Reports"
```

```

    handle="c60b1f9a_05ef_4e72_a737_e869917607db" inheritedPrivacy="true"
lastModified="1344607319557"
    lastaccessBy="admin" lastaccessOn="1345149848357" lastmodBy="admin"
length="0" name="Car_Reports"
    ownerId="10001" ownerName="admin" ownerType="U" policy="//v+f////////
f9////////9////////+AAAAA"
    returnedLng="en_US" type="MRFolder">
    <children _jt="ArrayList" size="6">
    <item _jt="IBFSMRObject" binary="false" container="true"
createdBy="admin" createdOn="1345149829421"
    defaultLng="en_US" description="Manufacturing Reports" dummy="false"
effectiveRSName="EDASERVE"
fullPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/
Manufacturing_Reports"
    handle="bb7ea628_2068_4d1c_b3cb_80555a30d53f" index="0"
inheritedPrivacy="true"
    lastModified="1345149829421" lastaccessBy="admin"
lastaccessOn="1345149829421" lastmodBy="admin"
    length="0" name="Manufacturing_Reports" ownerId="10001"
ownerName="admin" ownerType="U"
    parent="Car_Reports" policy="//v+f////////f9////////9////////+AAAAA"
returnedLng="en_US" type="MRFolder">
    <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
    <key _jt="string" value="en_US"/>
    <value _jt="ArrayList" size="2">
    <item _jt="string" index="0" value="Manufacturing Reports"/>
    </value>
    </entry>
    </nlsValues>
    <properties size="0"/>
    </item>
    <item _jt="IBFSMRObject" appName="ibisamp" binary="false"
createdBy="admin" createdOn="1345044807527"
    defaultLng="en_US" description="Sales Chart By Country" dummy="false"
effectiveAppName="ibisamp"
effectiveRSName="EDASERVE"
    extension="fex" fullPath="IBFS:/WFC/Repository/RESTful_Web_Services/
Car_Reports/
Sales_Chart_By_Country.fex"
    handle="5f4447c8_406e_41f6_8eca_7e056a4c1f27" index="1"
inheritedPrivacy="true"
    lastModified="1345044807527" lastaccessBy="admin"
lastaccessOn="1345047740027" lastmodBy="admin"
    length="5623" name="Sales_Chart_By_Country.fex" ownerId="10001"
ownerName="admin" ownerType="U"
    parent="Car_Reports" policy="//v+f////////f9////////9////////+AAAAA"

```

```

returnedLng="en_US" type="FexFile">
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="Sales Chart By Country"/>
      </value>
    </entry>
  </nlsValues>
  <properties size="1">
    <entry key="tool" value="infoAssist,chart,IAFull"/>
  </properties>
</item>
.
.
.
  <item _jt="CasterLibVersion" compressFormat="O$$" compressSize="0"
format="HTML"
    id="L8c1297c11613114a1flaccel6fc9173c28b9" index="1" size="1284"
versionNumber="3">
    <createDate _jt="calendar" time="1344779970997" timeZone="America/
New_York"/>
    <expireDate _jt="calendar" time="32474876370997" timeZone="America/
New_York"/>
  </item>
  <item _jt="CasterLibVersion" compressFormat="O$$" compressSize="0"
format="HTML"
    id="L84a1a1bc1e50014fd419eb21d05515d9f90c" index="2" size="1284"
versionNumber="4">
    <createDate _jt="calendar" time="1344978446241" timeZone="America/
New_York"/>
    <expireDate _jt="calendar" time="32474902046242" timeZone="America/
New_York"/>
  </item>
  <item _jt="CasterLibVersion" compressFormat="O$$" compressSize="0"
format="HTML"
    id="L95dd1bb0142d0145a919b0flb4ee0ce8390f" index="3" size="1284"
versionNumber="5">
    <createDate _jt="calendar" time="1344978694335" timeZone="America/
New_York"/>
    <expireDate _jt="calendar" time="32474902294335" timeZone="America/
New_York"/>
  </item>
</versionList>
<category id="RESTful_Web_Services/"
  isCategory="true" isMre="false" name="Weekly Reports"/>
</casterObject>
</item>

```

```
</children>
<nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
  <entry>
    <key _jt="string" value="en_US"/>
    <value _jt="ArrayList" size="2">
      <item _jt="string" index="0" value="Car Reports"/>
    </value>
  </entry>
</nlsValues>
<properties size="0"/>
</rootObject>
</ibfsrpc>
```

Each content definition is defined within the opening and closing *item* tag.

The *type* attribute defines the content type for one of the following content items:

- MRFolder.** Subfolder.
- FexFile.** WebFOCUS Report.
- CasterSchedule.** ReportCaster Schedule.
- CasterLibrary.** Library Content.
- CasterAccessList.** Library Access List.

The *name* attribute defines the name for the content item. The *description* attribute defines the title for the item. The *summary* attribute defines a brief description for the content item.

Listing the Versions for a Report Library Report

This RESTful web service request can be used to retrieve the versions list for a WebFOCUS report within the Report Library.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ContentName?
IBIRS_action=get
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder used for the stored WebFOCUS report. If the folder used for the stored WebFOCUS report exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

ContentName

Is the name of the stored WebFOCUS report as defined in the name attribute when listing the content of a folder. For more information, see [Listing Reports, Schedules, and Library Content Within the WebFOCUS Repository](#) on page 50.

Example:

In the following example, a versions list for the stored library report identified by L1748ltvgq02.lib within the Car_Reports folder is retrieved. The Car_Reports folder is a subfolder of the RESTful_Web_Services folder. L1748ltvgq02.lib is defined in the *name* attribute when listing the content of a folder. For more information, see [Listing Reports, Schedules, and Library Content Within the WebFOCUS Repository](#) on page 50.

The *description* attribute in the content list defines the title for stored report content. The title for L1748ltvgq02.lib, as per the Listing Reports, Schedules, and Library Content example, is defined as *Sales for a Specific Country*.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/L1748ltvgq02.lib?IBIRS_action=get
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
```

```

<ibfsparams size="0"/>
<rootObject _jt="IBFSCasterObject" binary="false" createdBy="admin"
createdOn="1344616201760"
  defaultLng="en_US" description="Sales for a Specific Country"
dummy="false"
  effectiveRSName="EDASERVE" extension="lib"
externalId="Le218a4d048cd45e4f9174bf1edc5e5a6"
fullPath="IBFS:/WFC/Repository/RESTful_Web_Services/
Car_Reports/L1748ltvgq02.lib"
  handle="51254a92I811dI4cd1Ib9f9If456ca5f00b9" inheritedPrivacy="true"
lastModified="1344616201760"
  lastaccessBy="admin" lastaccessOn="1345147221049" lastmodBy="admin"
length="0"
  name="L1748ltvgq02.lib" ownerId="10001" ownerName="admin"
ownerType="U"
  policy="//v+f////////f9////////9////////+AAAAA" returnedLng="en_US"
summary="Sales for a Specific Country"
  type="CasterLibrary">
<nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
<entry>
  <key _jt="string" value="en_US"/>
  <value _jt="ArrayList" size="2">
<item _jt="string" index="0" value="Sales for a Specific Country"/>
<item _jt="string" index="1" value="Sales for a Specific Country"/>
  </value>
</entry>
</nlsValues>
<properties size="2">
  <entry key="id" value="Le218a4d048cd45e4f9174bf1edc5e5a6"/>
  <entry key="tool" value="reportlibrary"/>
</properties>
<casterObject _jt="CasterContent" accessList="" accessType="OWNER"
category="Weekly Reports"
  description="Sales for a Specific Country" expireInterval="1"
expireMode="N"
  ibfsId="51254a92I811dI4cd1Ib9f9If456ca5f00b9"
ibfsPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports"
id="Le218a4d048cd45e4f9174bf1edc5e5a6"
  isWatch="false" lastExecution="1344978694335" lastVersion="5"
name="L1748ltvgq02.lib" owner="admin"
policy="subscribe,|,open,delete,rename,|,security;makeRules;viewRules"

```



```

reportgid="51254a92I81ldI4cdlIb9f9If456ca5f00b9"
scheduleId="Sa48balf3sa760s4e57sb349s4abda6168a17"
summary="Sales for a Specific Country"
taskId="T66cala2btd636t4e1dtad30t6930ae58ea09">
  <lastExecTime _jt="calendar" time="1344978694335" timeZone="America/
New_York"/>
  <versionList _jt="array" itemsClass="CasterLibVersion" size="4">
    <item _jt="CasterLibVersion" compressFormat="O$$" compressSize="0"
format="HTML"
      id="Lf7badbf2lcalcl4dcfla05dl5a8ddb387705" index="0" size="1284"
versionNumber="2">
      <createDate _jt="calendar" time="1344616201629"
timeZone="America/New_York"/>
      <expireDate _jt="calendar" time="32474885401652"
timeZone="America/New_York"/>
    </item>
    <item _jt="CasterLibVersion" compressFormat="O$$" compressSize="0"
format="HTML"
      id="L8c1297c1l6131l4a1flaccel6fc9173c28b9" index="1" size="1284"
versionNumber="3">
      <createDate _jt="calendar" time="1344779970997"
timeZone="America/New_York"/>
timeZone="America/New_York"/>
      <expireDate _jt="calendar" time="32474902046242" timeZone="America/
New_York"/>
    </item>
    <item _jt="CasterLibVersion" compressFormat="O$$" compressSize="0"
format="HTML"
      id="L95dd1bb0l42d0l45a9l9b0flb4ee0ce8390f" index="3" size="1284"
versionNumber="5">
      <createDate _jt="calendar" time="1344978694335" timeZone="America/
New_York"/>
      <expireDate _jt="calendar" time="32474902294335" timeZone="America/
New_York"/>
    </item>
  </versionList>
  <category id="RESTful_Web_Services/" isCategory="true"
isMre="false" name="Weekly Reports"/>
</casterObject>
</rootObject>
</ibfsrpc>

```

Each version definition is defined within the opening and closing *item* tag.

The *version* attributes are defined in the following list:

- versionNumber.** The version number for the stored WebFOCUS report.
- format.** The format of the WebFOCUS Report (for example, HTML).
- createDate.** The date the version was created.

- ❑ **expireDate.** The date the version will expire from the Report Library.

Listing the Parameters for a Repository Report

This RESTful web service request can be used to retrieve the current parameters for a WebFOCUS report in the WebFOCUS Repository.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/FexName?  
IBIRS_action=describeFex
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder used for the stored WebFOCUS report. If the folder used for the stored WebFOCUS report exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

FexName

Is the name of the WebFOCUS report as defined in the *name* attribute when listing the content of a folder. For more information, see [Listing Reports, Schedules, and Library Content Within the WebFOCUS Repository](#) on page 50.

Example:

In the following example, the current parameters for the Sales_for_a_Specific_Country.fex, which exists in the Car_Reports folder, is retrieved. The Car_Reports folder is a subfolder of the RESTful_Web_Services folder. Sales_for_a_Specific_Country.fex is defined in the *name* attribute when listing the content of a folder. For more information, see [Listing Reports, Schedules, and Library Content Within the WebFOCUS Repository](#) on page 50.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_  
Services/Car_Reports/Sales_for_a_Specific_Country.fex?  
IBIRS_action=describeFex
```

Response:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfparams size="0"/>
  <rootObject isSavedParam="false">
    <bindingInfo _jt="HashMap" loadFactor="0.75" threshold="24">
      <entry>
        <key _jt="string" value="SUBSYSTEM"/>
        <value isReqParm="false" value="Self Service"/>
      </entry>
      <entry>
        <key _jt="string" value="IBI_WF_charset"/>
        <value isReqParm="false" value="windows-1252"/>
      </entry>
      <entry>
        <key _jt="string" value="IBI_Webapp_Context_Default"/>
        <value isReqParm="false" value="/ibi_apps"/>
      </entry>
      <entry>
        <key _jt="string" value="SCRIPT_NAME"/>
        <value isReqParm="false" value="/ibi_apps/WFServlet"/>
      </entry>
      <entry>
        <key _jt="string" value="IBFS1_action"/>
        <value isReqParm="true" value="runItem"/>
      </entry>
      .
      .
      .
      <entry>
        <key _jt="string" value="SAVE_PARMRPT"/>
        <value isReqParm="false"
value="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/
Sales_for_a_Specific_Country.fex"/>
      </entry>
    </bindingInfo>
    <amperMap accessOrder="false" loadFactor="0.75" threshold="12">
      <entry>
        <key _jt="string" value="FOCFOCEXEC"/>
        <value format="" max="0.0" min="0.0" name="FOCFOCEXEC" strDef="">
          <type name="system"/>
        </value>
        <displayType name="prompt"/>
      </entry>
    </amperMap>
  </rootObject>
</ibfsrpc>

```

```

    <values accessOrder="false"
loadFactor="0.75" threshold="12"/>
    </value>
  </entry>
  <entry>
    <key _jt="string" value="FOCEXURL"/>
    <value format="" max="0.0" min="0.0" name="FOCEXURL" strDef="">
    <type name="set"/>
  <displayType name="prompt"/>
  <values accessOrder="false" loadFactor="0.75" threshold="12"/>
  </value>
  </entry>
  <entry>
    <key _jt="string" value="FOCHTMLURL"/>
    <value format="" max="0.0" min="0.0" name="FOCHTMLURL" strDef="">
    <type name="set"/>
  <displayType name="prompt"/>
  <values accessOrder="false" loadFactor="0.75" threshold="12"/>
  </value>
  </entry>
  <entry>
    <key _jt="string" value="GOOGLEMAPSAPIKEY"/>
    <value format="" max="0.0" min="0.0" name="GOOGLEMAPSAPIKEY" strDef="">
    <type name="set"/>
  <displayType name="prompt"/>
  <values accessOrder="false" loadFactor="0.75" threshold="12"/>
  </value>
  </entry>
  <entry>
    <key _jt="string" value="FOCREL"/>
    <value format="" max="0.0" min="0.0" name="FOCREL" strDef="">
    <type name="system"/>
  <displayType name="prompt"/>
  <values accessOrder="false" loadFactor="0.75" threshold="12"/>
  </value>
  </entry>
  <entry>
    <key _jt="string" value="EXCELSERVURL"/>
    <value format="" max="0.0" min="0.0" name="EXCELSERVURL" strDef="">
    <type name="set"/>
  <displayType name="prompt"/>
  <values accessOrder="false" loadFactor="0.75" threshold="12"/>
  </value>
  </entry>
  <entry>
    <key _jt="string" value="COUNTRY"/>
    <value description="Select Country:" format="" max="0.0" min="0.0"
name="COUNTRY" operation="" strDef="">
    <type name="unresolved"/>
    <displayType name="staticType"/>
    <values accessOrder="false" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="ENGLAND"/>
      <value _jt="string" value="ENGLAND"/>
    </entry>
    <entry>
      <key _jt="string" value="JAPAN"/>
      <value _jt="string" value="JAPAN"/>
    </entry>
    <entry>
      <key _jt="string" value="FRANCE"/>
      <value _jt="string" value="FRANCE"/>
    </entry>
    </values>
  </entry>

```

Each parameter definition is defined within the opening and closing *entry* tag.

The XML response that is returned includes many system parameters along with the parameters defined in the WebFOCUS report. Entries that have a *name* attribute for the *type* element of either *unresolved* or *defaultType* are the WebFOCUS report parameters, as shown in the following example:

```
<type name="unresolved" />
```

The *name* attribute within the *value* element defines the parameter that is being used in the selection, as shown in the following example:

```
<value description="Select Country:" format=""  
  max="0.0" min="0.0" name="COUNTRY" operation="" strDef="">
```

The *description* attribute within the *value* element defines the prompt title for the parameter.

If a parameter definition within a WebFOCUS report has a list of valid values for the selection, additional *entry* elements will exist in the XML within the parameter definition. The *value* attribute within the *key* element would contain each valid value.

```
<entry><key _jt="string" value="ENGLAND"/><value _jt="string"  
value="ENGLAND"/></entry>  
<entry><key _jt="string" value="JAPAN"/><value _jt="string"  
value="JAPAN"/></entry>  
<entry><key _jt="string" value="FRANCE"/><value _jt="string"  
value="FRANCE"/></entry>
```

In this example, ENGLAND, JAPAN, and FRANCE are the valid values that can be passed to this parameter.

Running a Report From the WebFOCUS Repository

This RESTful web service request can be used to run a report stored in the WebFOCUS Repository.

HTTP Method: POST

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ReportName
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder used for the stored WebFOCUS report. If the folder used for the stored WebFOCUS report exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

ReportName

Is the name of the WebFOCUS report to run. It must include a .fex extension.

Body Format:

```
IBIRS_action=run&IBIRS_proxyURL=clientPath&IBIRS_userName=Userid&  
IBIRS_password=Password&parmNameN=parmValueN&IBIRS_args=Object
```

where:

clientPath

Is the path to the client application making the RESTful web service calls to WebFOCUS. For example:

```
http://myapplication.maj.com/Sales/Monthly.aspx
```

The parameter is used when the initial WebFOCUS report contains drill-down links, links to images, On-Demand Paging reports, or Active Cache reports.

When you click on a drill-down link or pages in an On-Demand Paging report, the request will be routed to the client application, as defined by the *clientPath* value, instead of WebFOCUS. All of the parameter names and values are sent with the request. The client application will then have to redirect the request to the following URL, which is the WebFOCUS environment:

```
http://host:port/ibi_apps/rs/ibfs
```

Userid

Is the Reporting Server user ID. If the Reporting Server is running with Security Off or the Reporting Server sign-in credentials are configured in the WebFOCUS Reporting Server Client settings, then this parameter does not have to be sent in the REST request.

Password

Is the Reporting Server password. If the Reporting Server is running with Security Off or the Reporting Server sign-in credentials are configured in the WebFOCUS Reporting Server Client settings, then this parameter does not have to be sent in the REST request.

parmNameN

Is the name of the defined parameter that will be passed to the Reporting Server.

Note: The number of defined parameters can vary and depend on the number of parameters within the WebFOCUS report. For example, a WebFOCUS report that requires two parameters will also require these parameters and corresponding values to be set in the body of this RESTful web service (&parmName1=parmValue1&parmName2=parmValue2). In a different WebFOCUS report, there could be as many parameters as required (three, four, five, and so on).

parmValueN

Is the value of the defined parameter that will be passed to the Reporting Server.

Object (Optional)

Is the XML object that is used to turn off redirection when retrieving report output for MIME types like EXCEL and PDF using the following format:

```
<rootObject _jt="HashMap">
  <entry>
    <key _jt="string" value="IBFS_contextVars"/>
    <value _jt="HashMap">
      <entry>
        <key _jt="string" value="IBIWF_redirect"/>
        <value _jt="string" value="NEVER"/>
      </entry>
    </value>
  </entry>
</rootObject>
```

Example:

In the following example, the Sales_for_a_Specific_Country report is being executed only for Japan.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/
Car_Reports/Sales_for_a_Specific_Country.fex
```

Body:

```
IBIRS_action=run&COUNTRY=JAPAN
```

Response:

The response is a report in either HTML, Excel, PDF, active report, or a graph.

Change Management Export

This RESTful web service request can be used to export directories, files, and groups to be used by Change Management Import.

Note: This RESTful web service is common to functionality in the WebFOCUS Repository and the Reporting Server.

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/impex`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Body Format:

`IBIRS_action=cmExport&IBIRS_fileName=fileName`

where:

fileName

Is the name of the scenario for the Change Management Export. The scenario must exist in the /WebFOCUSxx/cm/export folder.

Example:

In the following example, the Change Management scenario called RESTWS is exported.

Request:

`http://localhost:8080/ibi_apps/rs/impex`

Body:

`IBIRS_action=cmExport&IBIRS_fileName=RESTWS`

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="cmExport"
returncode="1000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="1">
    <entry key="IBIRS_fileName" value="RESTWS"/>
  </ibfsparams>
  <rootObject _jt="string"/>
</ibfsrpc>
```


If the value for the *returncode* attribute in the XML response is 10000, then the scenario was exported successfully.

Change Management Import

This RESTful web service request can be used to import directories, files, and groups that were created using Change Management Export.

Note: This RESTful web service is common to functionality in the WebFOCUS Repository and the Reporting Server.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/impex

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Body Format:

*IBIRS_action=cmImport&IBIRS_fileName=fileName&IBIRS_resOverwrite=Overwrite
&IBIRS_importUsers=UserOpt&IBIRS_importRoles=RoleOpt
&IBIRS_importRules=RuleOpt&IBIRS_importGroups=GroupOpt*

where:

fileName

Is the name of the scenario in the Change Management Export. The scenario must exist in the /WebFOCUSxx/cm/import folder.

Overwrite

Can be set as follows:

- true.** Overwrites existing files and groups.
- false.** Does not overwrite existing files and groups.

UserOpt

Can be set as follows:

- 0.** Do not import users.
- 1.** Import users but do not overwrite.
- 2.** Import users and overwrite.

RoleOpt

Can be set as follows:

- 0.** Do not import roles.
- 1.** Import roles but do not overwrite.
- 2.** Import roles and overwrite.

RuleOpt

Can be set as follows:

- true.** Import rules.
- false.** Do not import rules.

GroupOpt

Can be set as follows:

- 0.** Do not import groups.
- 1.** Import groups but do not overwrite.
- 2.** Import groups and overwrite.

Example:

In the following example, the Change Management scenario called ImportMR is imported. Existing files will not be overwritten. Users, groups, roles, and rules will not be imported.

Request:

`http://localhost:8080/ibi_apps/rs/impex`

Body:

```
IBIRS_action=cmImport&IBIRS_fileName=ImportMR&IBIRS_resOverwrite=false
&IBIRS_importUsers=0&IBIRS_importRoles=0&IBIRS_importRules=false&IBIRS_imp
tGroups=0
```

Response:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="cmImport"
returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="7">
    <entry key="IBIRS_resOverwrite" value="false"/>
    <entry key="IBIRS_fileName" value="ImportMR"/>
    <entry key="IBIRS_runOptions" value="0"/>
    <entry key="IBIRS_importUsers" value="0"/>
    <entry key="IBIRS_importRoles" value="0"/>
    <entry key="IBIRS_importRules" value="false"/>
    <entry key="IBIRS_importGroups" value="0"/>
  </ibfsparams>
  <rootObject _jt="string"/>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the scenario was imported successfully.

Publishing an Item

This RESTful web service request can be used to publish an item.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ItemName

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder that will either contain the item (*ItemName*) to publish or be the folder that is published when *ItemName* is omitted. If the folder is a subfolder, then the path to the subfolder must be included in the REST URL. For example, TopFolderName/SubFolderName.

ItemName

Is the name of the item to publish, which can include WebFOCUS reports, schedules, library access lists, and library content.

Body Format:

IBIRS_action=publish

Example:

In the following example, a folder called Financial_Reports is published.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports`

Body:

IBIRS_action=publish

Response:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="publish"
returncode="10000"
    returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="1">
    <entry key="IBIRS_" value="/WFC/Repository/Financial_Reports"/>
  </ibfsparams>
  <rootObject _jt="string"/>
</ibfsrpc>
```

Unpublishing an Item

This RESTful web service request can be used to unpublish an item.

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ItemName`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder that will either contain the item (*ItemName*) to unpublish or be the folder that is unpublished when *ItemName* is omitted. If the folder is a subfolder, then the path to the subfolder must be included in the REST URL. For example, TopFolderName/SubFolderName.

ItemName

Is the name of the item to unpublish, which can include WebFOCUS reports, schedules, library access lists, and library content.

Body Format:

```
IBIRS_action=unpublish&IBIRS_ownerPath=OwnerPath&IBIRS_clearShares=OwnerPath
```

where:

OwnerPath

If the item is private, then the full path to the owner of the item. For example, /SSYS/USERS/admin.

OwnerPath

If the item is private, specify one of the following:

- true.** Unshares the item.
- false.** Does not unshare the item.

Example:

In the following example, a folder called Financial_Reports is unpublished.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports
```

Body:

```
IBIRS_action=unpublish&IBIRS_ownerPath=&IBIRS_clearShares=false
```

Response:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="unpublish"
returncode="10000"
    returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="3">
    <entry key="IBIRS_clearShares" value="false"/>
    <entry key="IBIRS_ownerPath"/>
    <entry key="IBIRS_" value="/WFC/Repository/Financial_Reports"/>
  </ibfsparams>
  <rootObject _jt="IBFSUserObject" description="Administrator"
dummy="false" email="restadmin@informationbuilders.com"
    fullPath="IBFS:/SSYS/USERS/admin" handle="10001" name="admin"
password="$faa2flda92f72a7d$0901495fld42962aa242af8aad5c7958a9f86013
a190482974970e81ee0259ba82cbd3856f01c6f29a
14abaf602143b5e79b3f18a4244b9018d9115892d363f4" rsPath="/
ibi_apps/rs/ibfs/SSYS/USERS/admin" type="User">
    <status _jt="IBSSUserStatus" name="ACTIVE"/>
    <groups _jt="ArrayList" size="0"/>
    <pSetList _jt="ArrayList" size="0"/>
  </rootObject>
</ibfsrpc>

```

Copying an Item

This RESTful web service request can be used to copy an item from one folder to another.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ItemName

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder that will either contain the item (*ItemName*) to copy or be the folder that is copied when *ItemName* is omitted. If the folder is a subfolder, then the path to the subfolder must be included in the REST URL. For example, TopFolderName/SubFolderName.

ItemName

Is the name of the item to copy, which can include WebFOCUS reports, schedules, library access lists, and library content.

Body Format:

```
IBIRS_action=copy&IBIRS_destination=destLocation&IBIRS_replace=destLocation
```

where:

destLocation

Is the destination location (specified as FolderName/ItemName) of the copied item.

destLocation

Specify one of the following:

- true.** Replaces the contents of the item.
- false.** Does not replace the contents of the item.

Example:

In the following example, the Drilldown_Report.fex WebFOCUS report is copied from the Car_Reports folder within the RESTful_Web_Services folder to the Financial_Reports folder. The contents are replaced.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/  
Car_Reports/Drilldown_Report.fex
```

Body:

```
IBIRS_action=copy&IBIRS_destination=/WFC/Repository/Financial_Reports/  
Drilldown_Report.fex&IBIRS_replace=true
```

Response:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="copy" returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="4">
    <entry key="IBIRS_destination" value="/WFC/Repository/Financial_Reports/
Drilldown_Report.fex"/>
    <entry key="IBIRS_replace" value="true"/>
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/WFC/Repository/RESTful_Web_Services/Car_Reports/
Drilldown_Report.fex"/>
  </ibfsparams>
  <rootObject _jt="IBFSMRObject" binary="false" createdBy="admin"
createdOn="1350346978647" defaultLng="en_US"
  description="Drilldown Report" dummy="false" extension="fex"
fullPath="IBFS:/WFC/Repository/Financial_Reports/Drilldown_Report.fex"
  handle="afba56f3_3e71_4ecf_9682_c88bb913634a" inheritedPrivacy="true"
lastModified="1350348325118" lastaccessBy="admin"
  lastaccessOn="1350348325118" lastmodBy="admin" length="5302"
name="Drilldown_Report.fex" ownerId="10001"
  ownerName="admin" ownerType="U" policy="//3/D///9+f7///
f7/////////8AAAA=" returnedLng="en_US"
rsPath="/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports/Drilldown_Report.fex"
type="FexFile">
  <content _jt="IBFSByteContent"
char_set="Cp1252">LSpEbyBub3QgZGVsZXRLIG9yIGlvZGlmeSB0aGUgY29tbWVudHMgYmV
sb3cKLsogVXNlZCB0byBUZXXN0IFJFUlQgQ29weSBmdW5jdGlvbmFsaXR5CiotSU5URVJQQUxf
Q09NTUVOVCBMSU5FizAkUEQ5NGJXd2dkbVZ5YzJsdmJqMGlNUzR3SWlCbGJtTnZar2x1Wnowa
.
.
.
UQUJMRSBTRVQgSFRNTEVOQ09ERSBPTGpPTiBUQUJMRSBTRVQgU1RZTEUgKgp
JTknMVURFPUlCRlM6L0ZJTEUvSUJJX0hUTUxfRElSL2phdmFhc3Npc3QvaW
50bc9FTi9FTklBRGVmYXVsdF9jb2liaW5lLnN0eSwkClRZUEU9UkVQTlJULC
BUSVRMRVRFWFQ9JldGX1RjVExFLlFVT1RFRFNuUklORywgU1VNTUFSWT0mV
0ZfU1VNTUFSWS5RVU9URURTVFJJTkcsICQKRU5EU1RZTEUKRU5ECgotU1VOCg==
  </content>
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="Drilldown Report"/>
      </value>
    </entry>
  </nlsValues>
  <properties size="1">
    <entry key="tool" value="infoAssist,report,IAFull"/>
  </properties>
</rootObject>
</ibfsrpc>

```

Moving an Item

This RESTful web service request can be used to move an item from one folder to another.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ItemName

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder that will either contain the item (*ItemName*) to move or be the folder that is moved when *ItemName* is omitted. If the folder is a subfolder, then the path to the subfolder must be included in the REST URL. For example, TopFolderName/SubFolderName.

ItemName

Is the name of the item to move, which can include WebFOCUS reports, schedules, library access lists, and library content.

Body Format:

IBIRS_action=move&IBIRS_destination=destLocation&IBIRS_replace=ReplaceFlag

where:

destLocation

Is the destination location (specified as FolderName/ItemName) of the moved item.

ReplaceFlag

Specify one of the following:

- true.** Replaces the contents of the item.
- false.** Does not replace the contents of the item.

Example:

In the following example, the Drilldown_Report.fex WebFOCUS report is moved from the Car_Reports folder within the RESTful_Web_Services folder to the Financial_Reports folder. The contents are not replaced.

Request:

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/
Car_Reports/Drilldown_Report.fex

Body:

IBIRS_action=move&IBIRS_destination=/WFC/Repository/Financial_Reports/
Drilldown_Report.fex&IBIRS_replace=false

Response:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="move"
returncode="10000"
    returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="4">
    <entry key="IBIRS_destination" value="/WFC/Repository/
Financial_Reports/Drilldown_Report.fex"/>
    <entry key="IBIRS_replace" value="false"/>
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/WFC/Repository/RESTful_Web_Services/
Car_Reports/Drilldown_Report.fex"/>
  </ibfsparams>
  <rootObject _jt="IBFSMRObject" binary="false" createdBy="admin"
createdOn="1348824882927" defaultLng="en_US"
    description="Drilldown Report" dummy="false" extension="fex"
fullPath="IBFS:/WFC/Repository/Financial_Reports/Drilldown_Report.fex"
```

```

        handle="286ace9f_4cd0_4a78_a26d_69dff1b72e0f"
inheritedPrivacy="true" lastModified="1350349735829"
        lastaccessBy="admin" lastaccessOn="1350349735811"
lastmodBy="admin" length="5302" name="Drilldown_Report.fex"
        ownerId="10001" ownerName="admin" ownerType="U"
policy="//3/D///9+f7/////f7////////8AAAA=" returnedLng="en_US"
rsPath="/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports/
Drilldown_Report.fex" type="FexFile">
    <content _jt="IBFSByteContent"
char_set="Cp1252">LSpEbyBub3QgZGVsZXRLIG9yIGlvZGlmeSB0aGUgY
29tbWVudHMgYmVsb3cKLSoGVXNlZCB0byBUZXXN0IFJFU1QgQ29weSBmdW5j
dGlvbmFsaXR5CiotSU5URVJQUxQ09NTUVVOCBMSU5FIzAkUEQ5NGJXd2d
kbVZ5YzJsdmJmQmGLlNUzR3SW1CbGJtTnZaR2x1Wnowa
        .
        .
        .
RlM6L0ZJTEUvSUJXJ0hUTUxfRElSL2phdmFhc3Npc3QvaW50bC9FTi9FTk1
BRGVmYXVsdF9jb2liaW51LnN0eSwkClRZUEU9UkVQTlJULCBUSVRMRVRWF
Q9JldGXL1RJVExFLlFVT1RFRFNuUklORywgU1VNTUFSWT0mV0ZfU1VNTUFSW
S5RVU9URURTVFJJTksICQKRu5EU1RZTEUKRU5ECgotU1VOCg==
    </content>
    <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
        <entry>
            <key _jt="string" value="en_US"/>
            <value _jt="ArrayList" size="2">
                <item _jt="string" index="0" value="Drilldown Report"/>
            </value>
        </entry>
    </nlsValues>
    <properties size="1">
        <entry key="tool" value="infoAssist,report,IAFull"/>
    </properties>
</rootObject>
</ibfsrpc>

```

Renaming an Item

This RESTful web service request can be used to rename an item.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ItemName

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder that will either contain the item (*ItemName*) to rename or be the folder that is renamed when *ItemName* is omitted. If the folder is a subfolder, then the path to the subfolder must be included in the REST URL. For example, TopFolderName/SubFolderName.

ItemName

Is the name of the item to rename, which can include WebFOCUS reports, schedules, library access lists, and library content.

Body Format:

`IBIRS_action=rename&IBIRS_newName=renamedItem`

where:

renamedItem

Is the name of the renamed item.

Example:

In the following example, the Financial_Reports folder is renamed to Financial_Reports_Renamed.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports`

Body:

`IBIRS_action=rename&IBIRS_newName=Financial_Reports_Renamed`

Response:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="rename" returncode="10000"
returndesc="SUCCESS"
  subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="3">
    <entry key="IBIRS_newName" value="Financial_Reports_Renamed"/>
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/WFC/Repository/Financial_Reports"/>
  </ibfsparams>
  <rootObject _jt="IBFSMRObject" binary="false" container="true" createdBy="admin"
  createdOn="1349964405620" defaultLng="en_US" description="Financial
Quarterly-Yearly Reports"
  dummy="false" fullPath="IBFS:/WFC/Repository/Financial_Reports_Renamed"
  handle="5d81bab8_7db7_40c9_96b9_df2b00ce3278"
lastModified="1350351652269" lastaccessBy="admin"
  lastaccessOn="1350351652269" lastmodBy="admin" length="0"
name="Financial_Reports_Renamed"
  ownerId="10001" ownerName="admin" ownerType="U" policy="//3/D///9+f/////
f/////////8AAAA="
  returnedLng="en_US" rsPath="/ibi_apps/rs/ibfs/WFC/Repository/
Financial_Reports_Renamed"
  summary="Quarterly and Yearly Financial Reports reported to the
Securities and Exchange Commission"
  type="MRFolder">
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="Financial Quarterly-Yearly
Reports"/>
        <item _jt="string" index="1" value="Quarterly and Yearly Financial
Reports reported to the Securities and Exchange Commission"/>
      </value>
    </entry>
  </nlsValues>
  <properties size="0"/>
</rootObject>
</ibfsrpc>

```

Uploading a WebFOCUS Report

This RESTful web service request can be used to upload a WebFOCUS report to the WebFOCUS environment.

HTTP Method: POST

REST URL Format:

*http://host:port/ibi_apps/rs/ibfs/WFC/Repository/*FolderName*/*FexName**

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder that will contain the WebFOCUS report to be uploaded. If the folder is a subfolder, then the path to the subfolder must be included in the REST URL. For example, TopFolderName/SubFolderName.

FexName

Is the name of the WebFOCUS report to be uploaded, which must include a .fex extension.

Body Format:

`IBIRS_action=put&IBIRS_object=Object`

where:

Object

Is the XML object defining the WebFOCUS report, which uses the following format:

```
<rootObject _jt="IBFSMRObject" description="ReportTitle" type="FexFile">
  <content _jt="IBFSByteContent" char_set="Cp1252">ContentBase64 </
content>
</rootObject>
```

where:

ContentBase64

Is the base64 encoded text of the WebFOCUS report to be uploaded.

ReportTitle

Is the title of the WebFOCUS report to be uploaded.

Example:

In the following example, a WebFOCUS report called Drilldown_Report.fex is created in the Financial_Reports folder.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports/
Drilldown_Report.fex
```

Body:

```

IBIRS_action=put&IBIRS_object=<rootObject _jt="IBFSMRObject" description="Drilldown
Report" type="FexFile">
  <content _jt="IBFSByteContent"
char_set="Cp1252">LSpEbyBub3QgZGVsZXRlIG9yIGlvZGlmeSB0aGUgY29tbWVudHMgYmV
sb3cKLSogVXNlZCB0byBUZXN0IFJFU1QgTW92ZSBmdW5jdGlvbWVsaXR
5CiotSU5URVJQQUxfQ09NTUVOVCBMSU5FIzAkUEQ5NGJXd2dkbVZ5YzJ
sdmJmMGlNUzR3SWlCbGJtTnZaR2x1WnowaVZWUkdMVGdpSuhOMFlXNwt
ZV3h2Ym1VOU1tNXZJa jgrRFFvOE1TMHRNUzR3TFMw
.
.
.
TgpPTiBUQUJMRSBTRVQgSFRNTEVOQ09ERSBPTgpPTiBUQUJMRSBTRVQgU1R
ZTEUgKgpJTkNMVURFPU1CRlM6L0ZJTEUvSUJjX0hUTUxfRElSL2phdmFhc3
Npc3QvaW50bc9FTi9FTklBRGVMYXVsdF9jb21iaW51LnN0eSwkClRZUEU9U
kVQTlJULCBUSVRMRVRFWFQ9JldGXlRjVExFLlFVT1RFRFNuUklORywgU1Vn
TUF5SWT0mV0ZfU1VNTUFSWS5RVU9URURTVFJjTkc5ICQKRu5EU1R2TEUK
RU5ECgotU1VOCg==
  </content>
</rootObject>

```

Response:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="put"
returncode="1000" returndesc="SUCCESS"
  subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="5">
    <entry key="IBIRS_replace" value="true"/>
    <entry key="IBIRS_private" value="__null"/>
    <entry key="IBIRS_object" value="&lt;rootObject
_jt=&quot;IBFSMRObject&quot;
  description=&quot;Drilldown Report&quot;
type=&quot;FexFile&quot;&gt;
      &lt;content _jt=&quot;IBFSByteContent&quot;
char_set=&quot;Cp1252&quot;&gt;LSpEbyBub3QgZGVsZXRlIG9yIGlvZ
GlmeSB0aGUgY29tbWVudHMgYmVsb3cKLSogVXNlZCB0byBUZXN0IFJFU1QgT
W92ZSBmdW5jdGlvbWVsaXR5Cio
      .
      .
      .
lFVT1RFRFNuUklORywgU1VnTUF5SWT0mV0ZfU1VNTUFSWS5RVU9URURTVFJjTkc5IC
QKRu5EU1R2TEUKRU5ECgotU1VOCg==&lt;/content&gt;&lt;/rootObject&gt;"/>

```

```

        <entry key="IBIRS_args" value="__null"/>
        <entry key="IBIRS_" value="/WFC/Repository/Financial_Reports/
Drilldown_Report.fex"/>
    </ibfsparams>
    <rootObject _jt="IBFSMRObject" binary="false" createdBy="admin"
createdOn="1350352555666" defaultLng="en_US"
        description="Drilldown Report" dummy="false" extension="fex"
fullPath="IBFS:/WFC/Repository/Financial_Reports/Drilldown_Report.fex"
        handle="ebd5f9e9_8607_439d_ac77_3089efb6184a"
inheritedPrivacy="true" lastModified="1350352555666"
        lastaccessBy="admin" lastaccessOn="1350352555666"
lastmodBy="admin" length="5302"
        name="Drilldown_Report.fex" ownerId="10001"
ownerName="admin" ownerType="U"
        policy="//3/D///9+f7////f7/////////8AAAA=" returnedLng="en_US"
rsPath="/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports/
Drilldown_Report.fex" type="FexFile">
        <content _jt="IBFSByteContent"
char_set="Cp1252">LSpEbyBub3QgZGVsZXRlIG9yIGlvZGlmeSB0aGUG
Y29tbWVudHMgYmVsb3cKLSogVXNlZCB0byBUZXRlIFJFU1QgTW92ZSBmdW
5jdGlvbmFsaXR5CiotSU5URVJJOQUxf
        .
        .
        .
lORywgU1VNTUFSWT0mV0ZfU1VNTUFSWS5RVU9URURTVFJTTksICQKRU5E
U1RZTEUKRU5ECgotU1VOCg==
    </content>
    <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
        <entry>
            <key _jt="string" value="en_US"/>
            <value _jt="ArrayList" size="2">
                <item _jt="string" index="0" value="Drilldown Report"/>
            </value>
        </entry>
    </nlsValues>
    <properties size="0"/>
</rootObject>
</ibfsrpc>

```

Creating a URL Link

This RESTful web service request can be used to create a URL link within the WebFOCUS environment.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/UrlName

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder that will contain the URL link. If the folder is a subfolder, then the path to the subfolder must be included in the REST URL. For example, TopFolderName/SubFolderName.

UrlName

Is the name of the URL link to be created, which must include a .url extension.

Body Format:

`IBIRS_action=put&IBIRS_object=Object`

where:

Object

Is the XML object defining the URL link, which uses the following format:

```
<rootObject _jt="IBFSMRObject" description="UrlLinkTitle"
type="URLFile"> <content _jt="IBFSByteContent"
char_set="Cp1252">UrlBase64</content>
<properties size="1">
<entry key="tool" value="url"/>
</properties>
</rootObject>
```

where:

UrlLinkTitle

Is the title of the URL link.

UrlBase64

Is the base64 encoded text of the URL.

Example:

In the following example, a URL called Yahoo.url is created in the Car_Reports folder. The URL of http://www.yahoo.com is base64 encoded.

Request:

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/Yahoo.url

Body:

```
IBIRS_action=put&IBIRS_object=<rootObject _jt="IBFSMRObject" description="Yahoo"
type="URLFile">
<content _jt="IBFSByteContent" char_set="Cp1252">aHR0cDovL3d3dy55YWhvby5jb20=</content>
<properties size="1">
<entry key="tool" value="url"/>
</properties>
</rootObject>
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="put" returncode="10000"
returnndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="5">
    <entry key="IBIRS_replace" value="true"/>
    <entry key="IBIRS_private" value="__null"/>
    <entry key="IBIRS_object" value="****"/>
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/WFC/Repository/RESTful_Web_Services/Car_Reports/
Yahoo.url"/>
  </ibfsparams>
  <rootObject _jt="IBFSMRObject" binary="false" createdBy="admin"
createdOn="1356625917312" defaultLng="en_US"
  description="Yahoo" dummy="false" extension="url" fullPath="IBFS:/WFC/
Repository/RESTful_Web_Services/Car_Reports/Yahoo.url"
  handle="1711f8b4_abbc_41c3_9c4c_7fd3288d4c62" lastModified="1356625917312"
lastaccessBy="admin"
  lastaccessOn="1356625917312" lastmodBy="admin" length="20" name="Yahoo.url"
  policy="////D//9+P////v////////+AAAA=" returnedLng="en_US"
rsPath="/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/Yahoo.url"
type="URLFile">
    <content _jt="IBFSByteContent" char_set="Cp1252">aHR0cDovL3d3dy55YWhvby5jb20=</
content>
    <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
      <entry>
        <key _jt="string" value="en_US"/>
        <value _jt="ArrayList" size="2">
          <item _jt="string" index="0" value="Yahoo"/>
        </value>
      </entry>
    </nlsValues>
    <properties size="0"/>
  </rootObject>
</ibfsrpc>
```

Retrieving Content for a WebFOCUS Report and URL

This RESTful web service request can be used to retrieve the textual content within a WebFOCUS report or URL link.

HTTP Method: GET

REST URL Format:

```
http://host[:port]/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ContentName?
IBIRS_action=getContent
```

where:

host

Is the name of the system where WebFOCUS is installed.

port (optional)

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder where the content exists. If the content exists in a subfolder, then the path to the subfolder name must be included in the REST URL. For example, ParentFolderName/FolderName.

ContentName

Is the name of the content, which must have a .fex extension for WebFOCUS reports and a .url extension for URL links.

Example 1:

In the following example, the content for the WebFOCUS report called Drilldown_Report.fex, from the Financial_Reports folder, is retrieved. The response Content-Type is text/plain.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports/
Drilldown_Report.fex?IBIRS_action=getContent
```

Response:

```
TABLE FILE CAR
HEADING
"Sales by models for &CAR"
SUM SALES BY MODEL
WHERE CAR EQ '&CAR';
"Last updated: &TOD &DATE"
ON TABLE PCHOLD FORMAT HTML
ON TABLE SET PAGE-NUM OFF
ON TABLE SET SQUEEZE ON
ON TABLE SET STYLE *
GRID=OFF, $
TYPE=HEADING, STYLE=BOLD, SIZE=18, $
TYPE=FOOTING, STYLE=ITALIC, $
TYPE=TITLE, STYLE=BOLD, $
ENDSTYLE
END
```

Example 2:

In the following example, the content for a URL called Yahoo.url, from the Car_Reports folder, is retrieved. The response Content-Type is text/plain.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/
Car_Reports/Yahoo.url?IBIRS_action=getContent
```

Response:

```
https://search.yahoo.com/search?p=Information+Builders
```

This section describes the format and structure of WebFOCUS Reporting Server RESTful web service requests.

In this chapter:

- [Listing WebFOCUS Reporting Server Nodes](#)
 - [Creating an Application](#)
 - [Listing Applications](#)
 - [Listing Files Within an Application](#)
 - [Listing the Parameters for a Report Within an Application](#)
 - [Running a Report Within an Application](#)
 - [Deleting an Application](#)
 - [Change Management Export](#)
 - [Change Management Import](#)
 - [Deleting a Role](#)
 - [Adding a Rule](#)
-

Listing WebFOCUS Reporting Server Nodes

This RESTful web service request can be used to list the Reporting Server nodes that are available to WebFOCUS.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs?IBIRS_action=get`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Example:

In the following example, Reporting Server nodes are listed.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/EDA?IBIRS_action=get
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action" returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSObject" container="true" description="EDA" dummy="false"
    fullPath="IBFS:/EDA" name="EDA" policy="///+f////////9//////////+AAAAA"
type="WebFOCUSComponent">
  <children _jt="ArrayList" size="1">
    <item _jt="IBFSEDANodeObject" container="true" defaultNode="true" description=""
dummy="false"
  fullPath="IBFS:/EDA/EDASERVE" host="MyComputer" index="0" name="EDASERVE"
nodeClass="CLIENT"
  parent="EDA" policy="///+f////////9//////////+AAAAA" port="8120"
  type="EDANode"/>
  </children>
</rootObject>
</ibfsrpc>
```

Each Reporting Server node definition is defined within the opening and closing *item* tag. The *name* attribute defines the name of the Reporting Server node. The *port* attribute defines the TCP/IP port used to communicate with the Reporting Server.

Creating an Application

This RESTful web service request can be used to create an application.

HTTP Method: POST

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/EDA/NodeName/AppName
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

NodeName

Is the name of the WebFOCUS Reporting Server node. For more information, see [Listing WebFOCUS Reporting Server Nodes](#) on page 85.

AppName

Is the name of the application to be created. If the application being created is a nested application of an existing application, then the existing application name is also included in the REST URL. This shows the path to the application being created (for example, ExistingApplication/ApplicationName).

Body Format:

```
IBIRS_action=put&IBIRS_object=Object
```

where:

Object

Is the XML object defining the attributes for the application using the following format:

```
<object _jt="IBFSFolder" container="true" type="IBFSFolder"></object>
```

Example:

In the following example, an application called Financial_Reports is created.

POST Request URL:

```
http://localhost:8080/ibi_apps/rs/ibfs/EDA/EDASERVE/Financial_Reports
```

Body:

```
IBIRS_action=put&IBIRS_object=<object _jt="IBFSFolder" container="true" type="IBFSFolder"></object>
```

Response:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSresponseObject" language="EN" name="put" returncode="10000"
returnndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="5">
    <entry key="IBIRS_replace" value="true"/>
    <entry key="IBIRS_path" value="/EDA/EDASERVE/Financial_Reports"/>
    <entry key="IBIRS_private" value="__null"/>
    <entry key="IBIRS_object" value="****"/>
    <entry key="IBIRS_args" value="__null"/>
  </ibfsparams>
  <rootObject _jt="IBFSFolder" container="true" description="Financial_Reports"
dummy="false" fullPath="IBFS:/EDA/EDASERVE/Financial_Reports" name="Financial_Reports"
policy="////D////fx////+/////////4AAAA" rsPath="/ibi_apps/rs/ibfs/EDA/EDASERVE/
Financial_Reports" type="IBFSFolder"/>
</ibfsrpc>

```

Listing Applications

This RESTful web service can be used to list the applications for a particular Reporting Server node.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/EDA/NodeName?IBIRS_action=get`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

NodeName

The name of the Reporting Server Node. For more information, see [Listing WebFOCUS Reporting Server Nodes](#) on page 85.

Example:

In the following example, the applications within the WebFOCUS Reporting Server called EDASERVE are listed.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/EDA/EDASERVE?IBIRS_action=get`

Response:


```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action" returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSEDANodeObject" container="true" defaultNode="true" description=""
    dummy="false" fullPath="IBFS:/EDA/EDASERVE" host="REST-COMPUTER" name="EDASERVE"
    nodeClass="CLIENT" policy="///+f////////9//////////+AAAAA" port="8120"
  type="EDANode">
    <children _jt="ArrayList" size="14">
      <item _jt="IBFSFolder" container="true" description="foccache" dummy="false"
        fullPath="IBFS:/EDA/EDASERVE/foccache" index="0" lastModified="1345560136000"
        name="foccache" parent="EDASERVE" policy="///+f////////9//////////+AAAAA"
      type="IBFSFolder"/>
      .
      .
      .
      <item _jt="IBFSFolder" container="true" description="maintain" dummy="false"
        fullPath="IBFS:/EDA/EDASERVE/maintain" index="12" lastModified="1344546157000"
        name="maintain"
        parent="EDASERVE" policy="///+f////////9//////////+AAAAA" type="IBFSFolder"/>
      <item _jt="IBFSFolder" container="true" description="session" dummy="false"
        fullPath="IBFS:/EDA/EDASERVE/session" index="13" lastModified="1344546157000"
        name="session"
        parent="EDASERVE" policy="///+f////////9//////////+AAAAA" type="IBFSFolder"/>
    </children>
  </rootObject>
</ibfsrpc>

```

Each application definition is defined within the opening and closing *item* tag. The *name* attribute defines the name of the application.

Listing Files Within an Application

This RESTful web service can be used to list all files within a particular application.

HTTP Method: GET

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/EDA/NodeName/AppName?IBIRS_action=get

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

NodeName

Is the name of the Reporting Server Node. For more information, see [Listing WebFOCUS Reporting Server Nodes](#) on page 85.

AppName

Is the name of the application containing the files to be listed. For more information, see [Listing Applications](#) on page 88.

Example:

In the following example, all files within the *ibisamp* application are listed.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/EDA/EDASERVE/ibisamp?IBIRS_action=get
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action" returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSFolder" container="true" description="ibisamp" dummy="false"
    fullPath="IBFS:/EDA/EDASERVE/ibisamp" lastModified="1345554266" name="ibisamp"
    policy="///+f////////9//////////+AAAAA" type="IBFSFolder">
    <children _jt="ArrayList" size="182">
      <item _jt="IBFSFile" description="cargraph.fex" dummy="false"
        fullPath="IBFS:/EDA/EDASERVE/ibisamp/cargraph.fex" index="0"
        lastModified="1328583952000"
        length="1471" name="cargraph.fex" parent="ibisamp" policy="///+f////////
        9//////////+AAAAA"
        type="IBFSFile"/>
      <item _jt="IBFSFile" description="carinst.fex" dummy="false"
        fullPath="IBFS:/EDA/EDASERVE/ibisamp/carinst.fex" index="1"
        lastModified="1328583952000"
        length="2624" name="carinst.fex" parent="ibisamp"
        policy="///+f////////9//////////+AAAAA"
        type="IBFSFile"/>
      .
      .
      .
      <item _jt="IBFSFile" description="wfmstart.html" dummy="false"
        fullPath="IBFS:/EDA/EDASERVE/ibisamp/wfmstart.html" index="181"
        lastModified="1328619018000"
        length="6364" name="wfmstart.html" parent="ibisamp" policy="///+f////////
        9//////////+AAAAA"
        type="IBFSFile"/>
    </children>
  </rootObject>
</ibfsrpc>
```

Each file definition is defined within the opening and closing *item* tag. The *name* attribute defines the name of the file.

The following list shows the WebFOCUS-specific file name extensions:

- fex.** WebFOCUS report.
- mas.** Master File Description.
- acx.** Access File.
- foc.** FOCUS database.
- etg.** Data Migrator flow.
- mnt.** Maintain procedure.
- wfm.** Maintain forms.
- fcf.** Maintain compiled.
- ftm.** Flat file usually used as a temporary file.

The *description* attribute defines the description that was used as input for the file.

Listing the Parameters for a Report Within an Application

This RESTful web service can be used to retrieve the current parameters for a WebFOCUS report stored within an application.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/EDA/NodeName/Appname/FexName?
IBIRS_action=describeFex
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

NodeName

Is the name of the Reporting Server Node. For more information, see [Listing WebFOCUS Reporting Server Nodes](#) on page 85.

Appname

Is the name of the application containing the files to be listed. For more information, see [Listing Applications](#) on page 88.

FexName

Is the name of the WebFOCUS report as defined in the *name* attribute when listing files within an application. For more information, see [Listing Files Within an Application](#) on page 89.

Example:

In this example, the current parameters for the carinst.fex report, which exists in the *ibisamp* application, are retrieved. The carinst.fex report is defined in the *name* attribute when listing the files within an application. For more information, see [Listing Files Within an Application](#) on page 89.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/EDA/EDASERVE/ibisamp/carinst.fex?  
IBIRS_action=describeFex
```

Response:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action" returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject isSavedParam="false">
    <bindingInfo _jt="HashMap" loadFactor="0.75" threshold="24">
      <entry>
        <key _jt="string" value="SUBSYSTEM"/>
        <value isReqParm="false" value="Self Service"/>
      </entry>
      <entry>
        <key _jt="string" value="IBI_WF_charset"/>
        <value isReqParm="false" value="windows-1252"/>
      </entry>
      <entry>
        .
        .
        .
      </entry>
      <key _jt="string" value="WF_TITLE"/>
      <value format="" max="0.0" min="0.0" name="WF_TITLE" strDef="">
        <type name="set"/>
        <displayType name="prompt"/>
        <values accessOrder="false" loadFactor="0.75" threshold="12"/>
      </value>
      </entry>
      <entry>
        <key _jt="string" value="FOCREL"/>
        <value format="" max="0.0" min="0.0" name="FOCREL" strDef="">
          <type name="system"/>
          <displayType name="prompt"/>
          <values accessOrder="false" loadFactor="0.75" threshold="12"/>
        </value>
      </entry>
      <entry>
        <key _jt="string" value="EXCELSERVURL"/>
        <value format="" max="0.0" min="0.0" name="EXCELSERVURL" strDef="">
          <type name="set"/>
          <displayType name="prompt"/>
          <values accessOrder="false" loadFactor="0.75" threshold="12"/>
        </value>
      </entry>
      <entry>
        <key _jt="string" value="COUNTRY"/>
        <value format="" max="0.0" min="0.0" name="COUNTRY" strDef="$*">
          <type name="defaultType"/>
          <displayType name="prompt"/>
          <values accessOrder="false" loadFactor="0.75" threshold="12"/>
        </value>
      </entry>
    </amperMap>
  </rootObject>
</ibfsrpc>

```

Each parameter definition is defined within the opening and closing *entry* tag.

The XML returned includes many system parameters along with the parameters defined in the WebFOCUS report. Entries that have a *name* attribute for the *type* element of either *unresolved* or *defaultType* are the WebFOCUS report parameters.

```
<type name="unresolved"/>
```

The *name* attribute within the *value* element defines the parameter that is being used in the selection.

```
<value name="COUNTRY" strDef="$*" min="0.0" max="0.0" format="">  
<type name="defaultType"/>
```

If a parameter definition within a WebFOCUS report has a default value, the *strDef* attribute within the *value* element will contain that value.

If a parameter definition within a WebFOCUS report has a Prompt title, the *description* attribute within the *value* element will contain the title.

If a parameter definition within a WebFOCUS report has a list of valid values for the selection, additional *entry* elements will exist in the XML within the parameter definition. The *value* attribute within the *key* element would contain each valid value.

Running a Report Within an Application

This RESTful web service can be used to run a report stored in an application.

HTTP Method: POST

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/EDA/NodeName/Appname/FexName
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

NodeName

Is the name of the Reporting Server Node. For more information, see [Listing WebFOCUS Reporting Server Nodes](#) on page 85.

Appname

Is the name of the application containing the files to be listed. For more information, see [Listing Applications](#) on page 88.

FexName

Is the name of the WebFOCUS report as defined in the *name* attribute when listing files within an application. For more information, see [Listing Files Within an Application](#) on page 89.

Body Format:

```
IBIRS_action=run&IBIRS_proxyURL=clientPath&IBIRS_userName=Userid&
IBIRS_password=Password&parmNameN=parmValueN&IBIRS_args=Object
```

where:

clientPath

Is the path to the client application making the RESTful web service calls to WebFOCUS. For example:

```
http://myapplication.maj.com/Sales/Monthly.aspx
```

The parameter is used when the initial WebFOCUS report contains drill-down links, links to images, On-Demand Paging reports, or Active Cache reports.

When you click on a drill-down link or pages in an On-Demand Paging report, the request will be routed to the client application, as defined by the *clientPath* value, instead of WebFOCUS. All of the parameter names and values are sent with the request. The client application will then have to redirect the request to the following URL, which is the WebFOCUS environment:

```
http://host:port/ibi_apps/rs/ibfs
```

Userid

Is the Reporting Server user ID. If the Reporting Server is running with Security Off or the Reporting Server sign-in credentials are configured in the WebFOCUS Reporting Server Client settings, then this parameter does not have to be sent in the REST request.

Password

Is the Reporting Server password. If the Reporting Server is running with Security Off or the Reporting Server sign-in credentials are configured in the WebFOCUS Reporting Server Client settings, then this parameter does not have to be sent in the REST request.

parmNameN

Is the name of the defined parameter that will be passed to the Reporting Server.

Note: The number of defined parameters can vary and depend on the number of parameters within the WebFOCUS report. For example, a WebFOCUS report that requires two parameters will also require these parameters and corresponding values to be set in the body of this RESTful web service (&parmName1=parmValue1&parmName2=parmValue2). In a different WebFOCUS report, there could be as many parameters as required (three, four, five, and so on).

parmValueN

Is the value of the defined parameter that will be passed to the Reporting Server.

Object (Optional)

Is the XML object that is used to turn off redirection when retrieving report output for MIME types like EXCEL and PDF using the following format:

```
<rootObject _jt="HashMap">
  <entry>
    <key _jt="string" value="IBFS_contextVars"/>
    <value _jt="HashMap">
      <entry>
        <key _jt="string" value="IBIWF_redirect"/>
        <value _jt="string" value="NEVER"/>
      </entry>
    </value>
  </entry>
</rootObject>
```

Example:

In the following example, the Sales_for_a_Specific_Country report is being executed only for Japan.

Request:

http://localhost:8080/ibi_apps/rs/ibfs/EDA/EDASERVE/ibisamp/carinst.fex

Body:

IBIRS_action=run&COUNTRY=JAPAN

Response:

The response is a report in either HTML, Excel, PDF, active report, or a graph.

Deleting an Application

This RESTful web service request can be used to delete an application.

HTTP Method: DELETE

REST URL Format:


```
http://host:port/ibi_apps/rs/ibfs/EDA/NodeName/Appname?IBIRS_action=delete
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

NodeName

Is the name of the Reporting Server Node. For more information, see [Listing WebFOCUS Reporting Server Nodes](#) on page 85.

AppName

Is the name of the application containing the files to be listed. For more information, see [Listing Applications](#) on page 88.

Example:

In the following example, the *wfretail* application is deleted.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/EDA/EDASERVE/wfretail?IBIRS_action=delete
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action" returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSFolder" container="true" description="wfretail" dummy="false"
    fullPath="IBFS:/EDA/EDASERVE/wfretail" name="wfretail" type="IBFSFolder"/>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the application was deleted successfully.

Change Management Export

This RESTful web service request can be used to export directories, files, and groups that were created using Change Management Import.

For more information, see [Change Management Export](#) on page 63.

Note: This RESTful web service is common to functionality in the WebFOCUS Repository and the Reporting Server.

Change Management Import

This RESTful web service request can be used to import directories, files, and groups that were created using Change Management Export.

For more information, see [Change Management Import](#) on page 65.

Note: This RESTful web service is common to functionality in the WebFOCUS Repository and the Reporting Server.

Deleting a Role

This RESTful web service request can be used to delete a role.

HTTP Method: DELETE

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/SSYS/ROLES/Role?IBIRS_action=delete
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Role

Is the name of the role to be deleted.

Example:

In the following example, a role called LibraryCustom is deleted.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/SSYS/ROLES/LibraryCustom?IBIRS_action=delete
```

Response:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="delete"
returncode="10000" returndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfsparams size="2">
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/SSYS/ROLES/LibraryCustom"/>
  </ibfsparams>
  <rootObject _jt="IBFSPermissionSetObject" description="Library Privilege
- Custom" dummy="false"
    fullPath="IBFS:/SSYS/ROLES/LibraryCustom" handle="381089792"
name="LibraryCustom"
    policy="////D///9+f/////f/////////8AAAA=" rsPath="/
ibi_apps/rs/ibfs/SSYS/ROLES/LibraryCustom"
    showPermissions="false" subsysNameList="WFC"
type="PermissionSet">
    <pSet _jt="IBSSPermissionSet" compLvl="100" description="Library
Privilege - Custom" id="381089792"
      name="LibraryCustom" shipped="false">
      <policy _jt="IBSSPolicy" derivedDate="1349173489158">
        <policy _jt="EnumMap" _keyJT="IBSSOperation" size="3">
          <entry>
            <key _jt="IBSSOperation" name="opLibrary"/>
            <value _jt="IBSSVerb" name="PERMIT"/>
          </entry>
          <entry>
            <key _jt="IBSSOperation" name="opList"/>
            <value _jt="IBSSVerb" name="PERMIT"/>
          </entry>
          <entry>
            <key _jt="IBSSOperation" name="opRCEexplorer"/>
            <value _jt="IBSSVerb" name="PERMIT"/>
          </entry>
        </policy>
      </pSet>
    <subsysList _jt="ArrayList" size="1">
      <item _jt="IBFSSubsystem" index="0" name="WFC"/>
    </subsysList>
  </rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the role was successfully deleted.

Adding a Rule

This RESTful web service request can be used to apply a rule against a specific item.

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/ItemToBeRestricted`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

ItemToBeRestricted

Is the path to the item that is being restricted. For example:

`/WFC/Repository/ParentFolder/FolderName`

Body Format:

`IBIRS_action=addRule&IBIRS_path=ItemToBeRestricted&IBIRS_subjectPath=GroupUser &IBIRS_verb=RestrictType&IBIRS_role=Role&IBIRS_applyTo=FolderChildren`

where:

ItemToBeRestricted

Is the path to the item that is being restricted. For example:

`/WFC/Repository/ParentFolder/FolderName`

GroupUser

Are the paths to groups or user IDs to which a specific role will be applied. For example:

`/SSYS/GROUPS/group1;/SSYS/GROUPS/group2`

RestrictType

Is one of the following types of restrictions that can be applied to a specific role:

- NOT_SET
- PERMIT
- DENY
- UNPERMIT
- UNDENY
- OVERPERMIT

CLEARINHERITANCE*Role*

Is the specific role that is applied to *GroupUser*. For example, List, Run, and ListAndRun.

FolderChildren

Determines whether the rule will be applied to only *ItemToBeRestricted*, *ItemToBeRestricted* and its children, or just the children. Valid values include:

FOLDER_AND_CHILDREN

FOLDER_ONLY

CHILDREN_ONLY

For example, FOLDER_AND_CHILDREN could be used to apply a rule for a specific folder and its subfolders.

Example:

In the following example, a rule is added to permit the user ID (restid) to list and run items from the Quarterly folder within Financial_Reports, including its subfolders.

POST Request:

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports/Quarterly

Body:

```
IBIRS_action=addRule&IBIRS_path=/WFC/Repository/Financial_Reports/Quarterly&IBIRS_subjectPath=/SSYS/USERS/restid&IBIRS_verb=PERMIT&IBIRS_role=ListAndRun&IBIRS_applyTo=FOLDER_AND_CHILDREN
```

Response:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="addRule"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="5">
    <entry key="IBIRS_verb" value="PERMIT"/>
    <entry key="IBIRS_role" value="ListAndRun"/>
    <entry key="IBIRS_applyTo" value="FOLDER_AND_CHILDREN"/>
    <entry key="IBIRS_subjectPath" value="/SSYS/USERS/restid"/>
    <entry key="IBIRS_" value="/WFC/Repository/Financial_Reports/Quarterly"/>
  </ibfsparams>
  <rootObject _jt="IBFSPermissionSetObject" description="List and run
content" dummy="false" fullPath="IBFS:/SSYS/ROLES/ListAndRun"
handle="10330"
  name="ListAndRun" policy="///D///9+f/////f/////////8AAAA=" rsPath="/
ibi_apps/rs/ibfs/SSYS/ROLES/ListAndRun" showPermissions="false"
  subsysNameList="WFC" type="PermissionSet">
    <pSet _jt="IBSSPermissionSet" compLvl="0" description="List and run
content" id="10330" name="ListAndRun" shipped="true">
      <policy _jt="IBSSPolicy" derivedDate="1348174711335">
        <policy _jt="EnumMap" _keyJT="IBSSOperation" size="2">
          <entry>
            <key _jt="IBSSOperation" name="opList"/>
            <value _jt="IBSSVerb" name="PERMIT"/>
          </entry>
          <entry>
            <key _jt="IBSSOperation" name="opRun"/>
            <value _jt="IBSSVerb" name="PERMIT"/>
          </entry>
        </policy>
      </policy>
      <subsysList _jt="ArrayList" size="1">
        <item _jt="IBFSSubsystem" index="0" name="WFC"/>
      </subsysList>
    </pSet>
  </rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the rule was successfully created.

This section describes the format and structure of WebFOCUS security administration web service requests.

In this chapter:

- Listing Users
- Listing Groups
- Listing Privileges
- Listing Roles
- Listing Users Within a Group
- Adding and Updating a User
- Deleting a User
- Adding and Updating a Group
- Deleting a Group
- Adding a User to a Group
- Removing a User From a Group
- Adding a Role
- Deleting a Role
- Adding a Rule
- Deleting a Rule
- Listing Rules for a Subject
- Listing Rules for a Resource
- Listing Rules for a Role
- Expanding a Policy String
- Creating a Policy String
- Running a Resource Template
- Changing a Password for a User

Listing Users

This RESTful web service request can be used to retrieve a list of existing WebFOCUS users.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/SSYS/USERS?IBIRS_action=get`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Example:

In the following example, a list of WebFOCUS users is retrieved.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/SSYS/USERS?IBIRS_action=get`

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action" returncode="10000"
  returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSObject" container="true" description="USERS" dummy="false"
    fullPath="IBFS:/SSYS/USERS" name="USERS" policy="///+f////////9//////////+AAAAA"
    type="WebFOCUSComponent">
    <children _jt="ArrayList" size="7">
      <item _jt="IBFSUserObject" description="Administrator" dummy="false"
        email="restadmin@informationbuilders.com"
        fullPath="IBFS:/SSYS/USERS/admin" handle="10001" index="0" name="admin"
        parent="USERS"
        password="$faa2f1da92f72a7d$0901495f1d42962aa242af8aad5c7958a9f86013a1904
        password="$94b192f81526ff9d$e71362964a5c2ef8e7814824dc247c8ee012ea118c1f6
        0402e2467f8ba0e5bcc508c3a8d973ecce0a8738d7445e25dcfb9a96411f6c7af6e6a5fe1
        051ccb669a" policy="///+f////////9//////////+AAAAA" type="User">
      <status _jt="IBSSUserStatus" name="ACTIVE"/>
    </children _jt="ArrayList" size="0"/>
  </rootObject _jt="IBFSObject" container="true" description="USERS" dummy="false"
    fullPath="IBFS:/SSYS/USERS" name="USERS" policy="///+f////////9//////////+AAAAA"
    type="WebFOCUSComponent">
  </ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action" returncode="10000"
    returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
```



```

    <pSetList _jt="ArrayList" size="0"/>
  </item>
  <item _jt="IBFSUserObject" description="MR admin 1" dummy="false" email=""
    fullPath="IBFS:/SSYS/USERS/mradmin1" handle="10004" index="2" name="mradmin1"
    parent="USERS"
    password="$ed23192360fcc75$6a8e50345185367b57f98b863e55b7e44fc94d10d3a1b
    0b6796774b694321bb57d6af841a30a4bd7f698c1e353db3cefe332e504bc854fe7878d12
    f664cc6cde" policy="///+f////////9//////////+AAAAA" type="User">
    <status _jt="IBSSUserStatus" name="ACTIVE"/>
    <groups _jt="ArrayList" size="0"/>
    <pSetList _jt="ArrayList" size="0"/>
  </item>
  <item _jt="IBFSUserObject" description="MR developer 1" dummy="false" email=""
    82974970e81ee0259ba82cbd3856f01c6f29a14abaf602143b5e79b3f18a4244b9018d911
    5892d363f4" policy="///+f////////9//////////+AAAAA" type="User">
    <status _jt="IBSSUserStatus" name="ACTIVE"/>
    <groups _jt="ArrayList" size="0"/>
    <pSetList _jt="ArrayList" size="0"/>
  </item>
  <item _jt="IBFSUserObject" description="Basic user" dummy="false" email=""
    fullPath="IBFS:/SSYS/USERS/auser" handle="10002" index="1" name="auser"
    parent="USERS"
    fullPath="IBFS:/SSYS/USERS/mrdev1" handle="10005" index="3" name="mrdev1"
    parent="USERS"
    password="$01265ddledf5431e$229e70a1c6068b977b241a63d0357818ac790448cb466
    d9c38e113380c29849f5a803025da486b9d7708025a4dd239d9ca123f458bfc7ff18ea5ae
    732c30a67e" policy="///+f////////9//////////+AAAAA" type="User">
    <status _jt="IBSSUserStatus" name="ACTIVE"/>
    <groups _jt="ArrayList" size="0"/>
    <pSetList _jt="ArrayList" size="0"/>
  </item>
  <item _jt="IBFSUserObject" description="WebFOCUS Public User" dummy="false" email=""
    fullPath="IBFS:/SSYS/USERS/public" handle="10007" index="4" name="public"
    parent="USERS"
    password="$4a4d50e70fc99c07$2306ff856f98e3a01bf3742f29e77a48078fb7447e1e9
    812a940e8f5b1cccb0132beb752de8d2af70ee45531934da6b0f2d1c81bd108af56d12a10
    6cdf8492f" policy="///+f////////9//////////+AAAAA" type="User">
    <status _jt="IBSSUserStatus" name="ACTIVE"/>
    <groups _jt="ArrayList" size="0"/>
    <pSetList _jt="ArrayList" size="1">
    <item _jt="string" index="0" value="WF_Role_Public"/>
  </pSetList>
</item>

```

```

    <item _jt="IBFSUserObject" description="User and Group administrator"
dummy="false" email=""
    fullPath="/SSYS/USERS/useradmin" handle="10006" index="5"
name="useradmin" parent="USERS"
password="$e7ac6cd796e2c928$569217829f425b3be8686288a648e1102bf5323140f79
b2051c920d59fc3f4cce410da7e49448a7f0efc849af04dfe92cd4ec2b78cdd59551981d9
a1799cd0a3" policy="///+f////////9//////////+AAAAA" type="User">
    <status _jt="IBSSUserStatus" name="ACTIVE"/>
    <groups _jt="ArrayList" size="0"/>
    <pSetList _jt="ArrayList" size="0"/>
</item>
    <item _jt="IBFSUserObject" description="Desktop guest account"
dummy="false" email=""
    fullPath="/SSYS/USERS/wfdesktop" handle="10008" index="6"
name="wfdesktop" parent="USERS"
password="$2d38eaabdf1c9719$d0b1b037587903b60623a1983a8064b626688b0182388
6215b445950752ab90de846c0caa4d0787f66835fbf642c190d040c518e816d99ce06ba50
e5f2485b8a" policy="///+f////////9//////////+AAAAA" type="User">
    <status _jt="IBSSUserStatus" name="ACTIVE"/>
    <groups _jt="ArrayList" size="0"/>
    <pSetList _jt="ArrayList" size="0"/>
</item>
</children>
</rootObject>
</ibfsrpc>

```

Each user definition is defined within the opening and closing *item* tag. The *name* attribute defines the name of the user ID. The *description* attribute defines the title for the user. The *email* attribute defines the email address for the user. The *name* attribute within the *status* element defines whether the user ID is active. For example:

```
<status name="ACTIVE" _jt="IBSSUserStatus"/>
```

Note: The *password* attribute is not a value.

Listing Groups

This RESTful web service request can be used to retrieve a list of existing WebFOCUS groups.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/SSYS/GROUPS?IBIRS_action=get
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Example:

In the following example, a list of WebFOCUS groups is retrieved.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/SSYS/GROUPS?IBIRS_action=get
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000" returndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSObject" container="true" description="GROUPS"
dummy="false" fullPath="IBFS:/SSYS/GROUPS" name="GROUPS"
  policy="///+f////////9//////////+AAAAA"
type="WebFOCUSComponent">
  <children _jt="ArrayList" size="6">
    <item _jt="IBFSGroupObject" container="true" description="All
defined users" dummy="false"
      fullPath="IBFS:/SSYS/GROUPS/EVERYONE" handle="10100" index="0"
name="EVERYONE" parent="GROUPS"
      policy="///+f/////9/9///4f+//P///+AAAAA" type="Group">
      <users _jt="ArrayList" size="0"/>
    </item>
    <item _jt="IBFSGroupObject" container="true"
description="Administrators" dummy="false"
      fullPath="IBFS:/SSYS/GROUPS/Administrators" handle="10101"
index="1" name="Administrators" parent="GROUPS"
      policy="///+f////////9//////////+AAAAA" type="Group">
      <users _jt="ArrayList" size="0"/>
    </item>
    <item _jt="IBFSGroupObject" container="true" description="Anonymous
Users" dummy="false"
      fullPath="IBFS:/SSYS/GROUPS/Anonymous" handle="10104"
index="2" name="Anonymous" parent="GROUPS"
      policy="///+f////////9//////////+AAAAA" type="Group">
      <users _jt="ArrayList" size="0"/>
    </item>
    <item _jt="IBFSGroupObject" container="true"
```

```

description="Userid/Group Administrators" dummy="false"
  fullPath="IBFS:/SSYS/GROUPS/UserAdmins" handle="10106"
index="3" name="UserAdmins" parent="GROUPS"
  policy="///+f////////9//////////+AAAAA" type="Group">
  <users _jt="ArrayList" size="0"/>
</item>
  <item _jt="IBFSGroupObject" container="true" description="WebFOCUS
Global Roles" dummy="false"
  fullPath="IBFS:/SSYS/GROUPS/WF_Global_Roles" handle="10111"
index="4" name="WF_Global_Roles" parent="GROUPS"
  policy="///+f////////9//////////+AAAAA" type="Group">
  <users _jt="ArrayList" size="0"/>
</item>
  <item _jt="IBFSGroupObject" container="true" description="WebFOCUS
Global Permissions" dummy="false"
  fullPath="IBFS:/SSYS/GROUPS/WF_Global_Permissions"
handle="10116" index="5" name="WF_Global_Permissions"
parent="GROUPS" policy="///+f////////9//////////+AAAAA" type="Group">
  <users _jt="ArrayList" size="0"/>
</item>
</children>
</rootObject>
</ibfsrpc>

```

Each group definition is defined within the opening and closing *item* tag. The *name* attribute defines the name of the group. The *description* attribute defines the title for the group.

Listing Privileges

This RESTful web service request can be used to retrieve a list of valid WebFOCUS privileges.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs?IBIRS_action=privileges`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Response:

A list of privileges is returned in an XML response document. Each privilege is defined within an `<item>` element tag:

```
<item type="Privilege" name="PrivilegeName" dummy="false"
description="PrivilegeDesc" subsysNameList="Subsystem"
parent="PRIVILEGES" ordinal="159" index="159"/>
```

where:

PrivilegeName

Is the name of the privilege.

PrivilegeDesc

Is the description of the privilege.

Subsystem

Is the subsystem that the privilege pertains to.

Example:

In the following example, a list of WebFOCUS privileges is retrieved.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs?IBIRS_action=privileges
```

Response:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="privileges"
returncode="10000" returndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSObject" container="true" description="PRIVILEGES"
dummy="true" name="PRIVILEGES" type="unknownType">
    <children _jt="ArrayList" size="180">
      <item description="InfoAssist Personal" dummy="false" index="159"
name="opInfoAssistPersonal" ordinal="159" parent="PRIVILEGES"
subsysNameList="Session" type="Privilege"/>
    </children>
  </rootObject>
</ibfsrpc>
```

In this sample response document, the name of the privilege is *opInfoAssistPersonal* and has a description of *InfoAssist Personal*. This privilege applies to the *Session* subsystem.

Listing Roles

This RESTful web service request can be used to retrieve a list of valid WebFOCUS roles.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/SSYS/ROLES?IBIRS_action=get`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Example:

In the following example, a list of WebFOCUS roles is retrieved.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/SSYS/ROLES?IBIRS_action=get`

Response:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="get"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="2">
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/SSYS/ROLES"/>
  </ibfsparams>
  <rootObject _jt="IBFSObject" container="true" description="ROLES"
dummy="false" fullPath="IBFS:/SSYS/ROLES" name="ROLES"
  policy="////D///9+f/////f/////////8AAAA=" rsPath="/
ibi_apps/rs/ibfs/SSYS/ROLES" type="WebFOCUSComponent">
    <children _jt="ArrayList" size="76">
      <item _jt="IBFSPermissionSetObject" description="Full control or
all privileges" dummy="false"
fullPath="IBFS:/SSYS/ROLES/SystemFullControl" handle="10301"
index="0" name="SystemFullControl" parent="ROLES"
policy="////D///9+f/////f/////////7+/8AAAA=" rsPath="/
ibi_apps/rs/ibfs/SSYS/ROLES/SystemFullControl" showPermissions="false"
subsysNameList="*" type="PermissionSet">
```

```

    <pSet _jt="IBSSPermissionSet" compLvl="1" description="Full
control or all privileges" id="10301" name="SystemFullControl"
shipped="true">
    <policy _jt="IBSSPolicy" derivedDate="1349171464497">
    <policy _jt="EnumMap" _keyJT="IBSSOperation" size="152">
    <entry>
    <key _jt="IBSSOperation" name="opViewPortal"/>
    <value _jt="IBSSVerb" name="OVERPERMIT"/>
    </entry>
    <entry>
    <key _jt="IBSSOperation" name="opList"/>
    <value _jt="IBSSVerb" name="OVERPERMIT"/>
    </entry>
    <entry>
    <key _jt="IBSSOperation" name="opViewProps"/>
    <value _jt="IBSSVerb" name="OVERPERMIT"/>
    </entry>
    <entry>
    <key _jt="IBSSOperation"
name="opDisplayVersionInfo"/>
    <value _jt="IBSSVerb" name="OVERPERMIT"/>
    </entry>
    .
    .
    <entry>
    <key _jt="IBSSOperation"
name="opInfoAssistPersonal"/>
    <value _jt="IBSSVerb" name="OVERPERMIT"/>
    </entry>
    </policy>
    </policy>
    <subsysList _jt="ArrayList" size="3">
    <item _jt="IBFSSubsystem" index="0" name="ROOT"/>
    <item _jt="IBFSSubsystem" index="1" name="WFC"/>
    <item _jt="IBFSSubsystem" index="2" name="BIP"/>
    </subsysList>
    </pSet>
  </item>
</children>
</rootObject>
</ibfsrpc>

```

Listing Users Within a Group

This RESTful web service request can be used to retrieve a list of existing WebFOCUS users within a particular group.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/SSYS/GROUPS/Group?IBIRS_action=get&IBIRS_args=
<object _jt="HashMap">
  <entry>
    <key _jt="string" value="TYPE"/>
    <value _jt="string" value="USERS"/>
  </entry>
</object>
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Group

Is the name of the group.

Example:

In the following example, a list of WebFOCUS users within the Administrators group is retrieved.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/SSYS/GROUPS/Administrators?
IBIRS_action=get&IBIRS_args=
<object _jt="HashMap">
  <entry>
    <key _jt="string" value="TYPE"/>
    <value _jt="string" value="USERS"/>
  </entry>
</object>
```

Response:


```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSresponseObject" language="EN" name="get" returncode="10000"
returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
  type="simple">
  <ibfsparams size="2">
    <entry key="IBIRS_args" value="&lt;object
_jt=&quot;HashMap&quot;&gt;&lt;entry&gt;&lt;key _jt=&quot;string&quot;
  value=&quot;TYPE&quot;/&gt;&lt;value _jt=&quot;string&quot;
value=&quot;USERS&quot;/&gt;&lt;/entry&gt;&lt;/object&gt;"/>
    <entry key="IBIRS_" value="/SSYS/GROUPS/Administrators"/>
  </ibfsparams>
  <rootObject _jt="IBFSGroupObject" container="true" description="Administrators"
dummy="false"
    fullPath="/IBFS:/SSYS/GROUPS/Administrators" name="Administrators"
policy="////D//9+P////v////////+AAAA="
    rsPath="/ibi_apps/rs/ibfs/SSYS/GROUPS/Administrators" type="Group">
    <children _jt="ArrayList" size="4">
      <item _jt="IBFSUserObject" description="Administrator" dummy="false"
email="restadmin@informationbuilders.com"
        fullPath="IBFS:/SSYS/USERS/admin" handle="10001" index="0" name="admin"
parent="Administrators"
        policy="////D//9+P////v////////+AAAA=" rsPath="/ibi_apps/rs/ibfs/SSYS/
USERS/admin" type="User">
        <status _jt="IBSSUserStatus" name="ACTIVE"/>
        <groups _jt="ArrayList" size="0"/>
        <pSetList _jt="ArrayList" size="0"/>
      </item>
      <item _jt="IBFSUserObject" description="MR admin 1" dummy="false" email=""
fullPath="IBFS:/SSYS/USERS/mradmin1"
        handle="10004" index="1" name="mradmin1" parent="Administrators"
policy="////D//9+P////v////////+AAAA="
        rsPath="/ibi_apps/rs/ibfs/SSYS/USERS/mradmin1" type="User">
        <status _jt="IBSSUserStatus" name="ACTIVE"/>
        <groups _jt="ArrayList" size="0"/>
        <pSetList _jt="ArrayList" size="0"/>
      </item>
      <item _jt="IBFSUserObject" description="Rest Userid" dummy="false"
email="restid@informationbuilders.com"
        fullPath="IBFS:/SSYS/USERS/restid77" handle="222102528" index="2"
name="restid" nameSpace="DB" parent="Administrators"
        policy="////D//9+P////v////////+AAAA="
        rsPath="/ibi_apps/rs/ibfs/SSYS/USERS/restid77" type="User">
        <status _jt="IBSSUserStatus" name="ACTIVE"/>
        <groups _jt="ArrayList" size="0"/>
        <pSetList _jt="ArrayList" size="0"/>
      </item>
    </children>
    <users _jt="ArrayList" size="0"/>
  </rootObject>
</ibfsrpc>

```

Each user is defined within the opening and closing *item* tag. The *name* attribute defines the name of the user. The *description* attribute defines the title for the user.

Adding and Updating a User

This RESTful web service request can be used to add or update a user to WebFOCUS.

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/SSYS/USERS/Userid`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Userid

Is the name of the user ID to be added.

Body Format:

`IBIRS_action=put&IBIRS_object=Object&IBIRS_replace=ReplaceUseridProperties`

where:

Object

Is the XML object defining the attributes for the user, using the following format:

```
<object _jt="IBFSUserObject" description="UseridTitle"
email="EmailAddress"
  password="Password" type="User" primaryGroupPath="IBFS:/SSYS/GROUPS/
groupName"><status _jt="IBSSUserStatus" name="Status"/></object>
```

where:

UseridTitle

Is the title for the user. If the title contains an ampersand character (&), this character should be encoded as &.

EmailAddress

Is the email address for the user.

Password

Is the password assigned to the user ID.

groupName

Is the primary group to which this user belongs.

Status

Is the status as to whether the user ID is to be added as an active or inactive user. The user ID can also be added or updated so that the password must be changed when signing on. Once the user signs on, the status is changed to Active. Choose from one of the following valid values:

- ACTIVE
- INACTIVE
- MUSTCHANGE

Replaceuseridproperties

Is an optional property that allows you to decide whether or not the properties for user ID can be updated. The following are examples of properties:

- Email Address
- Password
- User ID Title

You can choose *true* (default) or *false*.

Example:

In the following example, a user ID called *restid* is added. The title for the user is *Rest Userid*. The email address for the user is *restid@informationbuilders.com*. The password for the user is *rest*. The status for the user is *ACTIVE*.

Request:

http://localhost:8080/ibi_apps/rs/ibfs/SSYS/USERS/restid

Body:

```
IBIRS_action=put&IBIRS_object=<object _jt="IBFSUserObject "
description="Rest Userid" email="restid@informationbuilders.com"
  password="rest" type="User"><status _jt="IBSSUserStatus" name="ACTIVE"/></
object>
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000" returndesc="SUCCESS" subreturncode="0"
      subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSUserObject" description="Rest Userid" dummy="false"
email="restid@informationbuilders.com"
      fullPath="/SSYS/USERS/restid" name="restid" password="rest"
policy="///+f////////9//////////+AAAAA" type="User">
    <status _jt="IBSSUserStatus" name="ACTIVE"/>
    <groups _jt="ArrayList" size="0"/>
  </rootObject>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the user was added successfully.

Deleting a User

This RESTful web service request can be used to delete a user ID.

HTTP Method: DELETE

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/SSYS/USERS/Userid?IBIRS_action=delete`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Userid

Is the name of the user ID to be deleted.

Example:

In the following example, the user ID, *restid*, is deleted.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/SSYS/USERS/restid?IBIRS_action=delete`

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action" returncode="10000"
returnndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfparams size="0"/>
  <rootObject _jt="IBFSUserObject" description="Rest Userid" dummy="false"
email="restid@informationbuilders.com"
  fullPath="IBFS:/SSYS/USERS/restid" name="restid" nameSpace="DB"
password="$c35587264cbbbe38$ce25f3b448103e2031ee0b943bf8fd031b7bac26e1e05
91da4bb7105d2672f206de9eb7b39d4fb83eb6a01a0faea2ff1ec2ccaa70103f7723c89d0d426098c32"
  policy="///+f////////9//////////+AAAAA" type="User">
  <status _jt="IBSSUserStatus" name="ACTIVE"/>
  <groups _jt="ArrayList" size="0"/>
  <pSetList _jt="ArrayList" size="0"/>
</rootObject>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the user ID was deleted successfully.

Adding and Updating a Group

This RESTful web service request can be used to add or update a group to WebFOCUS.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/SSYS/GROUPS/Group

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Group

Is the name of the group to be added.

Body Format:

IBIRS_action=put&IBIRS_object=Object&IBIRS_replace=ReplaceGroupProperties

where:

Object

Is the XML object defining the attributes for the group, using the following format:

```
<object _jt="IBFSGroupObject" container="true" description="GroupTitle"
type="Group"></object>
```

where:

GroupTitle

Is the title for the group.

ReplaceGroupProperties

Is an optional property that allows you to decide whether or not the properties for a group can be updated.

You can choose *true* (default) or *false*.

Example:

In the following example, a group called RestUsers is added. The title for the group is RESTful Web Services Users.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/SSYS/GROUPS/RestUsers
```

Body:

```
IBIRS_action=put&IBIRS_object=
<object _jt="IBFSGroupObject" container="true" description="RESTful Web
Services Users" type="Group"></object>
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000" returndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSGroupObject" container="true" description="RESTful
Web Services Users" dummy="false"
    fullPath="/SSYS/GROUPS/RestUsers" name="RestUsers" policy="///
+f////////9//////////+AAAAA" type="Group">
    <users _jt="ArrayList" size="0"/>
  </rootObject>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the group was added successfully.

Deleting a Group

This RESTful web service request can be used to delete a group.

HTTP Method: DELETE

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/SSYS/GROUPS/Group?IBIRS_action=delete`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Group

Is the name of the group to be deleted.

Example:

In the following example, the group, *RestUsers*, is deleted.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/SSYS/GROUPS/RestUsers?
IBIRS_action=delete`

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000" returndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSGroupObject" container="true" description="RESTful
Web Services Users" dummy="false"
    fullPath="IBFS:/SSYS/GROUPS/RestUsers" name="RestUsers"
type="Group">
    <users _jt="ArrayList" size="0"/>
  </rootObject>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the group was deleted successfully.

Adding a User to a Group

This RESTful web service request can be used to add a user to a group.

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/SSYS/USERS/Userid`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Userid

Is the name of the user ID to be added to a group.

Body Format:

`IBIRS_action=addUserToGroup&IBIRS_groupPath=GroupPaths`

where:

GroupPaths

Are the paths to groups that the user ID is to be added. For example:

`/SSYS/GROUPS/group1;/SSYS/GROUPS/group2`

Example:

In the following example, the user ID *restid* is added to the *RestUsers* group.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/SSYS/USERS/restid`

Body:

`IBIRS_action=addUserToGroup&IBIRS_groupPath=/SSYS/GROUPS/RestUsers`

Response:


```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action"
returncode="10000" returndesc="SUCCESS" subreturncode="0"
    subsystem="SSYS" type="simple">
    <ibfsparams size="0"/>
    <rootObject _jt="IBFSUserObject" description="Rest Userid" dummy="false"
email="restid@informationbuilders.com"
        fullPath="IBFS:/SSYS/USERS/restid" handle="739804165"
name="restid" nameSpace="DB"
password="$c35587264cbbbe38$ce25f3b448103e2031ee0b943bf8fd031b7b
ac26e1e0591da4bb7105d2672f206de9eb7b39d4
        fb83eb6a01a0faea2ff1ec2ccaa70103f7723c89d0d426098c32"
policy="///+f////////9//////////+AAAAA" type="User">
    <status _jt="IBSSUserStatus" name="ACTIVE"/>
    <groups _jt="ArrayList" size="2">
        <item _jt="IBFSGroupObject" container="true" description="RESTful
Web Services Users" dummy="false"
            fullPath="IBFS:/SSYS/GROUPS/RestUsers" handle="1113254912"
index="0" name="RestUsers" type="Group">
            <users _jt="ArrayList" size="0"/>
        </item>
        <item _jt="IBFSGroupObject" container="true" description="All
defined users" dummy="false"
            fullPath="IBFS:/SSYS/GROUPS/EVERYONE" handle="10100"
index="1" name="EVERYONE" type="Group">
            <users _jt="ArrayList" size="0"/>
        </item>
    </groups>
    <pSetList _jt="ArrayList" size="0"/>
</rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the user ID was added successfully to the group.

Removing a User From a Group

This RESTful web service request can be used to remove a user from a group.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/SSYS/USERS/Userid

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Userid

Is the name of the user ID to be removed from a group.

Body Format:

IBIRS_action=removeUserFromGroup&IBIRS_groupPath=GroupPaths

where:

GroupPaths

Are the paths to groups that the user ID is to be removed. For example:

/SSYS/GROUPS/group1;/SSYS/GROUPS/group2

Example:

In the following example, the user ID *restid* is removed from the *RestUsers* group

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/SSYS/USERS/restid`

Body:

IBIRS_action=removeUserFromGroup&IBIRS_groupPath=/SSYS/GROUPS/RestUsers

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="IBIRS_action" returncode="10000"
returndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfsparams size="0"/>
  <rootObject _jt="IBFSUserObject" description="Rest Userid" dummy="false"
email="restid@informationbuilders.com"
  fullPath="IBFS:/SSYS/USERS/restid" handle="739804165" name="restid"
nameSpace="DB"
password="$c35587264cbbbe38$ce25f3b448103e2031ee0b943bf8fd031b7bac26e1e05
91da4bb7105d2672f206de9eb7b39d4fb83eb6a01a0faea2ff1ec2ccaa70103f7723c89d0d426098c32"
type="User">
  <status _jt="IBSSUserStatus" name="ACTIVE"/>
  <groups _jt="ArrayList" size="1">
    <item _jt="IBFSGroupObject" container="true" description="All defined users"
dummy="false"
      fullPath="IBFS:/SSYS/GROUPS/EVERYONE" handle="10100" index="0"
name="EVERYONE" type="Group">
      <users _jt="ArrayList" size="0"/>
    </item>
  </groups>
  <pSetList _jt="ArrayList" size="0"/>
</rootObject>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the user ID was removed successfully from the group.

Adding a Role

This RESTful web service request can be used to add a role and define the privileges that are associated with the role.

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/SSYS/ROLES/Role`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Role

Is the name of the role to be added.

Body Format:

`IBIRS_action=put&IBIRS_object=Object`

where:

Object

Is the XML object that defines the privileges associated with the role.

This XML object must have the following structure:

```
<object _jt="IBFSPermissionSetObject" description="RoleDescription"
showPermissions="true"
  subsysNameList="Subsystem" type="PermissionSet">
  <pSet _jt="IBSSPermissionSet" compLvl="100" shipped="true">
    <policy _jt="IBSSPolicy">
      <policy _jt="EnumMap" _keyJT="IBSSOperation">
        <entry>
          <key _jt="IBSSOperation" name="Privilege1"/>
          <value _jt="IBSSVerb" name="PERMIT"/>
        </entry>
        <entry>
          <key _jt="IBSSOperation" name="Privilege2"/>
          <value _jt="IBSSVerb" name="PERMIT"/>
        </entry>
      </policy>
    </policy>
    <subsysList _jt="ArrayList">
      <item _jt="IBFSSubsystem" index="0" name="Subsystem"/>
    </subsysList>
  </pSet>
</object>
```

where:

RoleDescription

Is the description of the role.

Subsystem

Is the subsystem associated with the role. Valid values include:

- Session
- WFC
- BIP
- EDA
- USERS
- GROUPS
- ROLES
- FILE
- WEB

PrivilegeN

Is the privilege name that is associated with the role. Each privilege is assigned within the opening and closing Entry tags. For a list of valid privileges, see [Listing Privileges](#) on page 108.

Example:

In the following example, a role called *LibraryCustom* is added. The description for the role is *Library Privilege - Custom*. The subsystem that the role is associated with is *WFC*. The privileges assigned to this role are *opLibrary*, *opList*, *opDisplayVersionInfo*, *opRCEXplorer*, *opPortalAccess*, and *opBidRunTime*.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/SSYS/ROLES/LibraryCustom
```

Body:

```
IBIRS_action=put&object=<object _jt="IBFSPermissionSetObject" description="Library
Privilege - Custom" showPermissions="true" subsysNameList="WFC" type="PermissionSet">
<pSet _jt="IBSSPermissionSet" compLvl="100" shipped="true">
  <policy _jt="IBSSPolicy">
    <policy _jt="EnumMap" _keyJT="IBSSOperation">
      <entry>
        <key _jt="IBSSOperation" name="opLibrary"/>
        <value _jt="IBSSVerb" name="PERMIT"/>
      </entry>
    <entry>
      <key _jt="IBSSOperation" name="opList"/>
      <value _jt="IBSSVerb" name="PERMIT"/>
    </entry>
  </policy>
</pSet>
</object>
```

```

</entry>
<entry>
  <key _jt="IBSSOperation" name="opDisplayVersionInfo"/>
  <value _jt="IBSSVerb" name="PERMIT"/>
</entry>
<entry>
  <key _jt="IBSSOperation" name="opRCEExplorer"/>
  <value _jt="IBSSVerb" name="PERMIT"/>
</entry>
<entry>
  <key _jt="IBSSOperation" name="opPortalAccess"/>
  <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
  <key _jt="IBSSOperation" name="opBidRunTime"/>
  <value _jt="IBSSVerb" name="PERMIT"/>
</entry>
</policy>
</pSet>
</object>
<subsysList _jt="ArrayList"><item _jt="IBFSSubsystem" index="0" name="WFC"/></
subsysList>
</pSet>
</object>

```

Response:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="put"
returncode="10000" returndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfsparams size="5">
    <entry key="IBIRS_replace" value="true"/>
    <entry key="IBIRS_private" value="__null"/>
    <entry key="IBIRS_object" value="&lt;object
_jt=&quot;IBFSPermissionSetObject&quot;; description=&quot;Library Privilege
- Custom&quot;;
      showPermissions=&quot;true&quot;;
subsysNameList=&quot;WFC&quot;; type=&quot;PermissionSet&quot;;&gt;
      &lt;pSet _jt=&quot;IBSSPermissionSet&quot;; compLvl=&quot;
100&quot;; shipped=&quot;true&quot;;&gt;
      &lt;policy _jt=&quot;IBSSPolicy&quot;;&gt; &lt;policy
_jt=&quot;EnumMap&quot;; _keyJT=&quot;IBSSOperation&quot;;&gt;
      &lt;entry&gt;&lt;key _jt=&quot;IBSSOperation&quot;;
name=&quot;opLibrary&quot;/&gt; &lt;value _jt=&quot;IBSSVerb&quot;;
name=&quot;PERMIT&quot;/&gt; &lt;/entry&gt;
&lt;entry&gt;&lt;key _jt=&quot;IBSSOperation&quot;;

```

```

        name="opList"/> <value
    _jt="IBSSVerb" name="PERMIT"/> </entry>
    </entry><key _jt="IBSSOperation"
name="opDisplayVersionInfo"/>
    <value _jt="IBSSVerb" name="PERMIT"/>
    </entry>
    <key _jt="IBSSOperation"
name="opRCEExplorer"/> <value _jt="IBSSVerb"
name="PERMIT"/> </entry>
    <key _jt="IBSSOperation"
name="opPortalAccess"/> <value
    _jt="IBSSVerb" name="DENY"/> </entry>
    <key _jt="IBSSOperation"
name="opBidRunTime"/> <value _jt="IBSSVerb"
name="PERMIT"/> </entry> </
policy> </policy>
    <subsysList _jt="ArrayList"><item
    _jt="IBFSSubsystem" index="0"
name="WFC"/></subsysList> </pSet>
</object> "/>
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/SSYS/ROLES/LibraryCustom"/>
</ibfparams>
    <rootObject _jt="IBFSPermissionSetObject" description="Library Privilege
- Custom" dummy="false" fullPath="IBFS:/SSYS/ROLES/LibraryCustom"
name="LibraryCustom" policy="///D///9+f/////f/////////
8AAAA=" rsPath="/ibi_apps/rs/ibfs/SSYS/ROLES/LibraryCustom"
showPermissions="false" subsysNameList="WFC"
type="PermissionSet">
    <pSet _jt="IBSSPermissionSet" compLvl="100" description="Library
Privilege - Custom" name="LibraryCustom" shipped="false">
    <policy _jt="IBSSPolicy" derivedDate="1349168261272">
    <policy _jt="EnumMap" _keyJT="IBSSOperation" size="3">
    <entry>
    <key _jt="IBSSOperation" name="opLibrary"/>
    <value _jt="IBSSVerb" name="PERMIT"/>
    </entry>
    <entry>
    <key _jt="IBSSOperation" name="opList"/>
    <value _jt="IBSSVerb" name="PERMIT"/>
    </entry>
    <entry>
    <key _jt="IBSSOperation" name="opRCEExplorer"/>
    <value _jt="IBSSVerb" name="PERMIT"/>
    </entry>
    </policy>
    </policy>
    <subsysList _jt="ArrayList" size="1">
    <item _jt="IBFSSubsystem" index="0" name="WFC"/>
    </subsysList>
    </pSet>
</rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the role was successfully added.

Deleting a Role

This RESTful web service request can be used to delete a role.

HTTP Method: DELETE

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/SSYS/ROLES/Role?IBIRS_action=delete`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Role

Is the name of the role to be deleted.

Example:

In the following example, a role called LibraryCustom is deleted.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/SSYS/ROLES/LibraryCustom?IBIRS_action=delete`

Response:


```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="delete"
returncode="10000" returndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfsparams size="2">
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/SSYS/ROLES/LibraryCustom"/>
  </ibfsparams>
  <rootObject _jt="IBFSPermissionSetObject" description="Library Privilege
- Custom" dummy="false"
    fullPath="IBFS:/SSYS/ROLES/LibraryCustom" handle="381089792"
name="LibraryCustom"
    policy="////D///9+f/////f/////////8AAAA=" rsPath="/
ibi_apps/rs/ibfs/SSYS/ROLES/LibraryCustom"
    showPermissions="false" subsystemNameList="WFC"
type="PermissionSet">
    <pSet _jt="IBSSPermissionSet" compLvl="100" description="Library
Privilege - Custom" id="381089792"
      name="LibraryCustom" shipped="false">
      <policy _jt="IBSSPolicy" derivedDate="1349173489158">
        <policy _jt="EnumMap" _keyJT="IBSSOperation" size="3">
          <entry>
            <key _jt="IBSSOperation" name="opLibrary"/>
            <value _jt="IBSSVerb" name="PERMIT"/>
          </entry>
          <entry>
            <key _jt="IBSSOperation" name="opList"/>
            <value _jt="IBSSVerb" name="PERMIT"/>
          </entry>
          <entry>
            <key _jt="IBSSOperation" name="opRCEXplorer"/>
            <value _jt="IBSSVerb" name="PERMIT"/>
          </entry>
        </policy>
      </pSet>
    <subsysList _jt="ArrayList" size="1">
      <item _jt="IBFSSubsystem" index="0" name="WFC"/>
    </subsysList>
  </rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the role was successfully deleted.

Adding a Rule

This RESTful web service request can be used to apply a rule against a specific item.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/ItemToBeRestricted

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

ItemToBeRestricted

Is the path to the item that is being restricted. For example:

/WFC/Repository/ParentFolder/FolderName

Body Format:

IBIRS_action=addRule&IBIRS_path=ItemToBeRestricted&IBIRS_subjectPath=GroupUser &IBIRS_verb=RestrictType&IBIRS_role=Role&IBIRS_applyTo=FolderChildren

where:

ItemToBeRestricted

Is the path to the item that is being restricted. For example:

/WFC/Repository/ParentFolder/FolderName

GroupUser

Are the paths to groups or user IDs to which a specific role will be applied. For example:

/SSYS/GROUPS/group1;/SSYS/GROUPS/group2

RestrictType

Is one of the following types of restrictions that can be applied to a specific role:

- NOT_SET
- PERMIT
- DENY
- UNPERMIT
- UNDENY
- OVERPERMIT

CLEARINHERITANCE*Role*

Is the specific role that is applied to *GroupUser*. For example, List, Run, and ListAndRun.

FolderChildren

Determines whether the rule will be applied to only *ItemToBeRestricted*, *ItemToBeRestricted* and its children, or just the children. Valid values include:

FOLDER_AND_CHILDREN

FOLDER_ONLY

CHILDREN_ONLY

For example, FOLDER_AND_CHILDREN could be used to apply a rule for a specific folder and its subfolders.

Example:

In the following example, a rule is added to permit the user ID (restid) to list and run items from the Quarterly folder within Financial_Reports, including its subfolders.

POST Request:

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports/Quarterly

Body:

```
IBIRS_action=addRule&IBIRS_path=/WFC/Repository/Financial_Reports/
Quarterly&IBIRS_subjectPath=/SSYS/USERS/restid&IBIRS_verb=PERMIT&
IBIRS_role=ListAndRun&IBIRS_applyTo=FOLDER_AND_CHILDREN
```

Response:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="addRule"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="5">
    <entry key="IBIRS_verb" value="PERMIT"/>
    <entry key="IBIRS_role" value="ListAndRun"/>
    <entry key="IBIRS_applyTo" value="FOLDER_AND_CHILDREN"/>
    <entry key="IBIRS_subjectPath" value="/SSYS/USERS/restid"/>
    <entry key="IBIRS_" value="/WFC/Repository/Financial_Reports/Quarterly"/>
  </ibfsparams>
  <rootObject _jt="IBFSPermissionSetObject" description="List and run
content" dummy="false" fullPath="IBFS://SSYS/ROLES/ListAndRun"
handle="10330"
  name="ListAndRun" policy="////D///9+f/////f/////////8AAAA=" rsPath="/
ibi_apps/rs/ibfs/SSYS/ROLES/ListAndRun" showPermissions="false"
  subsystemNameList="WFC" type="PermissionSet">
    <pSet _jt="IBSSPermissionSet" compLvl="0" description="List and run
content" id="10330" name="ListAndRun" shipped="true">
      <policy _jt="IBSSPolicy" derivedDate="1348174711335">
        <policy _jt="EnumMap" _keyJT="IBSSOperation" size="2">
          <entry>
            <key _jt="IBSSOperation" name="opList"/>
            <value _jt="IBSSVerb" name="PERMIT"/>
          </entry>
          <entry>
            <key _jt="IBSSOperation" name="opRun"/>
            <value _jt="IBSSVerb" name="PERMIT"/>
          </entry>
        </policy>
      </policy>
      <subsysList _jt="ArrayList" size="1">
        <item _jt="IBFSSubsystem" index="0" name="WFC"/>
      </subsysList>
    </pSet>
  </rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the rule was successfully created.

Deleting a Rule

This RESTful web service request can be used to remove a rule.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/ItemRestricted

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

ItemRestricted

Is the path to the item that is being restricted. For example:

/WFC/Repository/ParentFolder/FolderName

Body Format:

```
IBIRS_action=removeRule&IBIRS_path=ItemRestricted&IBIRS_subjectPath=GroupUser
&
IBIRS_role=Role
```

where:

ItemRestricted

Is the path to the item that is being restricted. For example:

/WFC/Repository/ParentFolder/FolderName

GroupUser

Is the path to a particular group or user ID to which a specific role was applied. For example:

/SSYS/USERS/userid

Role

Is the specific role that was applied to GroupUser. For example, List, Run, and ListAndRun.

Example:

In the following example, a rule is removed for user ID (restid) to list and run items from the Quarterly folder within Financial_Reports.

POST Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports/Quarterly
```

Body:

```
IBIRS_action=removeRule&IBIRS_path=/WFC/Repository/Financial_Reports/Quarterly
&IBIRS_subjectPath=/SSYS/USERS/restid&IBIRS_role=ListAndRun
```

Response:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="removeRule"
returncode="10000" returndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfsparams size="3">
    <entry key="IBIRS_role" value="ListAndRun"/>
    <entry key="IBIRS_subjectPath" value="/SSYS/USERS/restid"/>
    <entry key="IBIRS_" value="/WFC/Repository/Financial_Reports/
Quarterly"/>
  </ibfsparams>
  <rootObject _jt="IBFSPermissionSetObject" description="List and run
content" dummy="false" fullPath="IBFS:/SSYS/ROLES/ListAndRun"
  handle="10330" name="ListAndRun" policy="////D///9+f/////
f/////////8AAAA="
    rsPath="/ibi_apps/rs/ibfs/SSYS/ROLES/ListAndRun"
showPermissions="false" subsystemNameList="WFC" type="PermissionSet">
  <pSet _jt="IBSSPermissionSet" compLvl="0" description="List and run
content" id="10330" name="ListAndRun" shipped="true">
    <policy _jt="IBSSPolicy" derivedDate="1349182611014">
      <policy _jt="EnumMap" _keyJT="IBSSOperation" size="2">
        <entry>
          <key _jt="IBSSOperation" name="opList"/>
          <value _jt="IBSSVerb" name="PERMIT"/>
        </entry>
        <entry>
          <key _jt="IBSSOperation" name="opRun"/>
          <value _jt="IBSSVerb" name="PERMIT"/>
        </entry>
      </policy>
    </policy>
  </pSet>
  <subsysList _jt="ArrayList" size="1">
    <item _jt="IBFSSubsystem" index="0" name="WFC"/>
  </subsysList>
</rootObject>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the rule was successfully removed.

Listing Rules for a Subject

This RESTful web service request can be used to retrieve a list of rules for a specific subject.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/GroupUser

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

GroupUser

Is the path to a particular group or user ID. For example:

`/SSYS/USERS/userid`

Body Format:

`IBIRS_action=listRulesForSubject`

Example:

In the following example, a list rules is returned for user ID called *restid*.

POST Request:

`http://localhost:8080/ibi_apps/rs/ibfs/restid`

Body:

`IBIRS_action=listRulesForSubject`

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="listRulesForSubject"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="1">
    <entry key="IBIRS_path" value="/SSYS/USERS/restid"/>
  </ibfsparams>
  <rootObject _jt="IBFSObject" container="true" description="Rules for
User:restid" dummy="false" fullPath="NO PATH/RulesList" name="RulesList"
type="IBFSFolder">
    <children _jt="ArrayList" size="1">
      <item _jt="IBFSRuleObject" compLvl="0" dummy="false" index="0"
pSetName="ListAndRun" parent="RulesList" resPathName="IBFS:/WFC/Repository/
Financial_Reports/Quarterly" subject="restid" subjectType="U" type="Rule">
        <verb _jt="IBSSVerb" name="PERMIT"/>
        <applyTo name="FOLDER_AND_CHILDREN"/>
      </item>
    </children>
  </rootObject>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the rules were successfully returned.

Listing Rules for a Resource

This RESTful web service request can be used to retrieve a list of rules for a specific resource.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/Resource

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Resource

Is the path to a particular resource. For example:

/WFC/Repository/ParentFolder/FolderName

Body Format:

IBIRS_action=listRulesForResource

Example:

In the following example, a list rules is returned for the *Quarterly* folder within the *Financial_Reports* folder.

POST Request:

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Financial_Reports/Quarterly

Body:

IBIRS_action=listRulesForResource

Response:


```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="listRulesForResource"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="1">
    <entry key="IBIRS_path" value="/WFC/Repository/Financial_Reports/
Quarterly"/>
  </ibfsparams>
  <rootObject _jt="IBFSObject" container="true" description="RulesList"
dummy="false" fullPath="NO PATH/RulesList" name="RulesList"
type="IBFSFolder">
    <children _jt="ArrayList" size="1">
      <item _jt="IBFSRuleObject" compLvl="0" dummy="false" index="0"
pSetName="ListAndRun" parent="RulesList" resPathName="IBFS:/WFC/Repository/
Financial_Reports/Quarterly" subject="restid" subjectType="U" type="Rule">
        <verb _jt="IBSSVerb" name="PERMIT"/>
        <applyTo name="FOLDER_AND_CHILDREN"/>
      </item>
    </children>
  </rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the rules were successfully returned.

Listing Rules for a Role

This RESTful web service request can be used to retrieve a list of rules for a specific role.

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/Role`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Role

Is the specific role (for example, List, Run, and ListAndRun).

Body Format:

`IBIRS_action=listRulesForRole`

Example:

In the following example, a list rules is returned for the *ListAndRun* role.

POST Request:

`http://localhost:8080/ibi_apps/rs/ibfs/ListAndRun`

Body:

`IBIRS_action=listRulesForRole`

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="listRulesForRole"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="1">
    <entry key="IBIRS_path" value="/ListAndRun"/>
  </ibfsparams>
  <rootObject _jt="IBFSObject" container="true" description="Rules with
PSET:ListAndRun" dummy="false" fullPath="NO PATH/RulesList"
name="RulesList" type="IBFSFolder">
    <children _jt="ArrayList" size="2">
      <item _jt="IBFSRuleObject" compLvl="0" dummy="false" index="0"
pSetName="ListAndRun" parent="RulesList" resPathName="IBFS:/WFC/Repository/
Public" subject="EVERYONE" subjectType="G" type="Rule">
        <verb _jt="IBSSVerb" name="PERMIT"/>
        <applyTo name="FOLDER_AND_CHILDREN"/>
      </item>
      <item _jt="IBFSRuleObject" compLvl="0" dummy="false" index="1"
pSetName="ListAndRun" parent="RulesList" resPathName="IBFS:/WFC/Repository/
Financial_Reports/Quarterly" subject="restid" subjectType="U" type="Rule">
        <verb _jt="IBSSVerb" name="PERMIT"/>
        <applyTo name="FOLDER_AND_CHILDREN"/>
      </item>
    </children>
  </rootObject>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the rules were successfully returned.

Expanding a Policy String

This RESTful web service request can be used to expand a Base64- encoded policy string representing the Effective Policy to an XML document, which lists the privileges that are permitted or denied.

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/utlils`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Body Format:

`IBIRS_action=expandPolicy&IBIRS_base64Policy=PolicyString`

where:

PolicyString

Is the Base64-encoded policy string representing the Effective Policy. The string can be obtained by running RESTful Web Service requests that list various items (for example, Folders, Users, and Groups).

Example:

In this example, the Base64-encoded policy string containing the following value is expanded:

`////D////fx////+//////////4AAAA`

POST Request:

`http://localhost:8080/ibi_apps/rs/utlils`

Body:

`IBIRS_action=expandPolicy&IBIRS_base64Policy=////D////fx////+//////////4AAAA`

Response:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="expandPolicy"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="1">
    <entry key="IBIRS_base64Policy" value="////D////fx/////+//////////
4AAAA"/>
  </ibfsparams>
  <rootObject _jt="IBSSPolicy" derivedTime="1368100027309">
    <policy _jt="EnumMap" _keyJT="IBSSOperation" size="185">
      <entry>
        <key _jt="IBSSOperation" name="opLibrary"/>
        <value _jt="IBSSVerb" name="PERMIT"/>
      </entry>
      <entry>
        <key _jt="IBSSOperation" name="opViewPortal"/>
        <value _jt="IBSSVerb" name="PERMIT"/>
      </entry>
      <entry>
        <key _jt="IBSSOperation" name="opList"/>
        <value _jt="IBSSVerb" name="PERMIT"/>
      </entry>
      <entry>
        <key _jt="IBSSOperation" name="opViewProps"/>
        <value _jt="IBSSVerb" name="PERMIT"/>
      </entry>
      <entry>
        <key _jt="IBSSOperation" name="opDisplayVersionInfo"/>
        <value _jt="IBSSVerb" name="PERMIT"/>
      </entry>
      <entry>
        <key _jt="IBSSOperation" name="opFavorites"/>
        <value _jt="IBSSVerb" name="PERMIT"/>
      </entry>
      <entry>
        <key _jt="IBSSOperation" name="opMagnify"/>
        <value _jt="IBSSVerb" name="PERMIT"/>
      </entry>
    <entry>
      <key _jt="IBSSOperation" name="opMobileFavorites"/>
      <value _jt="IBSSVerb" name="PERMIT"/>
    </entry>
  </policy>
  .
  .
  .

```

```

        <entry>
          <key _jt="IBSSOperation" name="opCustom01"/>
          <value _jt="IBSSVerb" name="DENY"/>
        </entry>
      <entry>
        <key _jt="IBSSOperation" name="opCustom02"/>
        <value _jt="IBSSVerb" name="DENY"/>
      </entry>
    <entry>
      <key _jt="IBSSOperation" name="opCustom03"/>
      <value _jt="IBSSVerb" name="DENY"/>
    </entry>
  <entry>
    <key _jt="IBSSOperation" name="opCustom04"/>
    <value _jt="IBSSVerb" name="DENY"/>
  </entry>
<entry>
  <key _jt="IBSSOperation" name="opCustom05"/>
  <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
  <key _jt="IBSSOperation" name="opCustom06"/>
  <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
  <key _jt="IBSSOperation" name="opCustom07"/>
  <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
  <key _jt="IBSSOperation" name="opCustom08"/>
  <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
  <key _jt="IBSSOperation" name="opCustom09"/>
  <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
  <key _jt="IBSSOperation" name="opCustom10"/>
  <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
  <key _jt="IBSSOperation" name="opCustom11"/>
  <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
  <key _jt="IBSSOperation" name="opCustom12"/>
  <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
  <key _jt="IBSSOperation" name="opCustom13"/>
  <value _jt="IBSSVerb" name="DENY"/>
</entry>

```

```

<entry>
    <key _jt="IBSSOperation" name="opCustom14"/>
    <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
    <key _jt="IBSSOperation" name="opCustom15"/>
    <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
    <key _jt="IBSSOperation" name="opCustom16"/>
    <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
    <key _jt="IBSSOperation" name="opCustom17"/>
    <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
    <key _jt="IBSSOperation" name="opCustom18"/>
    <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
    <key _jt="IBSSOperation" name="opCustom19"/>
    <value _jt="IBSSVerb" name="DENY"/>
</entry>
<entry>
    <key _jt="IBSSOperation" name="opCustom20"/>
    <value _jt="IBSSVerb" name="DENY"/>
</entry>
</policy>
</rootObject>
</ibfsrpc>

```

Creating a Policy String

This RESTful web service request can be used to return a Base64- encoded policy string representing the Effective Policy based on an XML document, which lists the privileges that are permitted or denied.

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/utills`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Body Format:

`IBIRS_action=compactPolicy&IBIRS_policy=Policy`

where:

Policy

Is the XML object defining the Effective Policy.

Example:

In the following example, a Base64-encoded policy string will be created based on the XML document containing the Effective Policy.

POST Request:

`http://localhost:8080/ibi_apps/rs/utills`

Body:

```

IBIRS_action=compactPolicy&IBIRS_policy=<rootObject _jt="IBSSPolicy"
derivedTime="1368095042526"><policy _jt="EnumMap" _keyJT="IBSSOperation"
size="185"><entry><key _jt="IBSSOperation" name="opLibrary"/><value _jt="IBSSVerb"
name="PERMIT"/></entry><entry><key _jt="IBSSOperation" name="opViewPortal"/><value
_jt="IBSSVerb" name="PERMIT"/></entry><entry><key _jt="IBSSOperation" name="opList"/
><value _jt="IBSSVerb" name="PERMIT"/></entry><entry><key _jt="IBSSOperation"
name="opViewProps"/><value _jt="IBSSVerb" name="PERMIT"/></entry><entry><key
_jt="IBSSOperation" name="opDisplayVersionInfo"/><value _jt="IBSSVerb"
name="PERMIT"/></entry><entry><key _jt="IBSSOperation" name="opFavorites"/><value
_jt="IBSSVerb" name="PERMIT"/></entry><entry><key _jt="IBSSOperation" name="opMagnify"/
><value _jt="IBSSVerb" name="PERMIT"/></entry><entry><key _jt="IBSSOperation"
name="opMobileFavorites"/><value _jt="IBSSVerb"
.
.
name="opCustom01"/><value _jt="IBSSVerb" name="DENY"/></entry><entry><key
_jt="IBSSOperation" name="opCustom02"/><value _jt="IBSSVerb" name="DENY"/></
entry><entry><key _jt="IBSSOperation" name="opCustom03"/><value _jt="IBSSVerb"
name="DENY"/></entry><entry><key _jt="IBSSOperation" name="opCustom04"/><value
_jt="IBSSVerb" name="DENY"/></entry><entry><key _jt="IBSSOperation" name="opCustom05"/
><value _jt="IBSSVerb" name="DENY"/></entry><entry><key _jt="IBSSOperation"
name="opCustom06"/><value _jt="IBSSVerb" name="DENY"/></entry><entry><key
_jt="IBSSOperation" name="opCustom07"/><value _jt="IBSSVerb" name="DENY"/></
entry><entry><key _jt="IBSSOperation" name="opCustom08"/><value _jt="IBSSVerb"
name="DENY"/></entry><entry><key _jt="IBSSOperation" name="opCustom09"/><value
_jt="IBSSVerb" name="DENY"/></entry><entry><key _jt="IBSSOperation" name="opCustom10"/
><value _jt="IBSSVerb" name="DENY"/></entry><entry><key _jt="IBSSOperation"
name="opCustom11"/><value _jt="IBSSVerb" name="DENY"/></entry><entry><key
_jt="IBSSOperation" name="opCustom12"/><value _jt="IBSSVerb" name="DENY"/></
entry><entry><key _jt="IBSSOperation" name="opCustom13"/><value _jt="IBSSVerb"
name="DENY"/></entry><entry><key _jt="IBSSOperation" name="opCustom14"/><value
_jt="IBSSVerb" name="DENY"/></entry><entry><key _jt="IBSSOperation" name="opCustom15"/
><value _jt="IBSSVerb" name="DENY"/></entry><entry><key _jt="IBSSOperation"
name="opCustom16"/><value _jt="IBSSVerb" name="DENY"/></entry><entry><key
_jt="IBSSOperation" name="opCustom17"/><value _jt="IBSSVerb" name="DENY"/></
entry><entry><key _jt="IBSSOperation" name="opCustom18"/><value _jt="IBSSVerb"
name="DENY"/></entry><entry><key _jt="IBSSOperation" name="opCustom19"/><value
_jt="IBSSVerb" name="DENY"/></entry><entry><key _jt="IBSSOperation" name="opCustom20"/
><value _jt="IBSSVerb" name="DENY"/></entry></policy></rootObject>

```

Response:


```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="compactPolicy"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="1">
    <entry key="IBIRS_policy" value="&lt;rootObject
    _jt=&quot;IBSSPolicy&quot; derivedTime=&quot;
1368095042526&quot;&gt;&lt;policy _jt=&quot;EnumMap&quot;
_keyJT=&quot;IBSSOperation&quot; size=&quot;
185&quot;&gt;&lt;entry&gt;&lt;key _jt=&quot;IBSSOperation&quot;
name=&quot;opLibrary&quot;/&gt;&lt;value _jt=&quot;IBSSVerb&quot;
name=&quot;PERMIT&quot;/&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
_jt=&quot;IBSSOperation&quot; name=&quot;opViewPortal&quot;/&gt;&lt;value
_jt=&quot;IBSSVerb&quot; name=&quot;PERMIT&quot;/&gt;&lt;/
entry&gt;&lt;entry&gt;&lt;key _jt=&quot;IBSSOperation&quot;
name=&quot;opList&quot;/&gt;&lt;value _jt=&quot;IBSSVerb&quot;
name=&quot;PERMIT&quot;/&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
_jt=&quot;IBSSOperation&quot; name=&quot;opViewProps&quot;/&gt;&lt;value
_jt=&quot;IBSSVerb&quot; name=&quot;PERMIT&quot;/&gt;&lt;/
entry&gt;&lt;entry&gt;&lt;key _jt=&quot;IBSSOperation&quot;
name=&quot;opDisplayVersionInfo&quot;/&gt;&lt;value
_jt=&quot;IBSSVerb&quot; name=&quot;PERMIT&quot;/&gt;&lt;/
entry&gt;&lt;entry&gt;&lt;key _jt=&quot;IBSSOperation&quot;
name=&quot;opFavorites&quot;/&gt;&lt;value _jt=&quot;IBSSVerb&quot;
name=&quot;PERMIT&quot;/&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
_jt=&quot;IBSSOperation&quot; name=&quot;opMagnify&quot;/&gt;&lt;value
_jt=&quot;IBSSVerb&quot; name=&quot;PERMIT&quot;/&gt;&lt;/
entry&gt;&lt;entry&gt;&lt;key _jt=&quot;IBSSOperation&quot;
name=&quot;opMobileFavorites&quot;/&gt;&lt;value _jt=&quot;IBSSVerb&quot;
.
.
.
_jt=&quot;IBSSOperation&quot; name=&quot;opCustom01&quot;/&gt;&lt;value
_jt=&quot;IBSSVerb&quot; name=&quot;DENY&quot;/&gt;&lt;/
entry&gt;&lt;entry&gt;&lt;key _jt=&quot;IBSSOperation&quot;
name=&quot;opCustom02&quot;/&gt;&lt;value _jt=&quot;IBSSVerb&quot;
name=&quot;DENY&quot;/&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
_jt=&quot;IBSSOperation&quot; name=&quot;opCustom03&quot;/&gt;&lt;value
_jt=&quot;IBSSVerb&quot; name=&quot;DENY&quot;/&gt;&lt;/
entry&gt;&lt;entry&gt;&lt;key

```

```

    _jt="&quot;IBSSOperation&quot; name="&quot;opCustom04&quot; /&gt;&lt;value
    _jt="&quot;IBSSVerb&quot; name="&quot;DENY&quot; /&gt;&lt;/
    entry&gt;&lt;entry&gt;&lt;key _jt="&quot;IBSSOperation&quot;
    name="&quot;opCustom05&quot; /&gt;&lt;value _jt="&quot;IBSSVerb&quot;
    name="&quot;DENY&quot; /&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
    _jt="&quot;IBSSOperation&quot; name="&quot;opCustom06&quot; /&gt;&lt;value
    _jt="&quot;IBSSVerb&quot; name="&quot;DENY&quot; /&gt;&lt;/
    entry&gt;&lt;entry&gt;&lt;key _jt="&quot;IBSSOperation&quot;
    name="&quot;opCustom07&quot; /&gt;&lt;value _jt="&quot;IBSSVerb&quot;
    name="&quot;DENY&quot; /&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
    _jt="&quot;IBSSOperation&quot; name="&quot;opCustom08&quot; /&gt;&lt;value
    _jt="&quot;IBSSVerb&quot; name="&quot;DENY&quot; /&gt;&lt;/
    entry&gt;&lt;entry&gt;&lt;key _jt="&quot;IBSSOperation&quot;
    name="&quot;opCustom09&quot; /&gt;&lt;value _jt="&quot;IBSSVerb&quot;
    name="&quot;DENY&quot; /&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
    _jt="&quot;IBSSOperation&quot; name="&quot;opCustom10&quot; /&gt;&lt;value
    _jt="&quot;IBSSVerb&quot; name="&quot;DENY&quot; /&gt;&lt;/
    entry&gt;&lt;entry&gt;&lt;key _jt="&quot;IBSSOperation&quot;
    name="&quot;opCustom11&quot; /&gt;&lt;value _jt="&quot;IBSSVerb&quot;
    name="&quot;DENY&quot; /&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
    _jt="&quot;IBSSOperation&quot; name="&quot;opCustom12&quot; /&gt;&lt;value
    _jt="&quot;IBSSVerb&quot; name="&quot;DENY&quot; /&gt;&lt;/
    entry&gt;&lt;entry&gt;&lt;key _jt="&quot;IBSSOperation&quot;
    name="&quot;opCustom13&quot; /&gt;&lt;value _jt="&quot;IBSSVerb&quot;
    name="&quot;DENY&quot; /&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
    _jt="&quot;IBSSOperation&quot; name="&quot;opCustom14&quot; /&gt;&lt;value
    _jt="&quot;IBSSVerb&quot; name="&quot;DENY&quot; /&gt;&lt;/
    entry&gt;&lt;entry&gt;&lt;key _jt="&quot;IBSSOperation&quot;
    name="&quot;opCustom15&quot; /&gt;&lt;value _jt="&quot;IBSSVerb&quot;
    name="&quot;DENY&quot; /&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
    _jt="&quot;IBSSOperation&quot; name="&quot;opCustom16&quot; /&gt;&lt;value
    _jt="&quot;IBSSVerb&quot; name="&quot;DENY&quot; /&gt;&lt;/
    entry&gt;&lt;entry&gt;&lt;key _jt="&quot;IBSSOperation&quot;
    name="&quot;opCustom17&quot; /&gt;&lt;value _jt="&quot;IBSSVerb&quot;
    name="&quot;DENY&quot; /&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
    _jt="&quot;IBSSOperation&quot; name="&quot;opCustom18&quot; /&gt;&lt;value
    _jt="&quot;IBSSVerb&quot; name="&quot;DENY&quot; /&gt;&lt;/
    entry&gt;&lt;entry&gt;&lt;key _jt="&quot;IBSSOperation&quot;
    name="&quot;opCustom19&quot; /&gt;&lt;value _jt="&quot;IBSSVerb&quot;
    name="&quot;DENY&quot; /&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
    _jt="&quot;IBSSOperation&quot; name="&quot;opCustom20&quot; /&gt;&lt;value
    _jt="&quot;IBSSVerb&quot; name="&quot;DENY&quot; /&gt;&lt;/entry&gt;&lt;/
    policy&gt;&lt;/rootObject&gt;"/>
  </ibfsparams>
  <rootObject _jt="string">////D////fx/////+//////////4AAAA</rootObject>
</ibfsrpc>

```

Running a Resource Template

This RESTful web service request can be used to run a resource template, which will create predefined groups, roles, portals, and folders.

For more information on resource templates, see the *WebFOCUS Security and Administration* content (*Chapter 5, WebFOCUS Administration, Understanding Domains* topic).

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/templates`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Body Format:

`IBIRS_action=run&IBIRS_fileName=templateName&IBIRS_vars=object`

where:

templateName

Is the name of the resource template in the \WebFOCUSxx\config\resource_templates directory.

Object

Is the XML object defining the name and description of the group, role, portal, and folder that is created from running the template. The XML object uses the following format:

```
<object _jt="HashMap"><entry><key _jt="string" value="name" />
<value _jt="string" value="name" /></entry><entry><key _jt="string"
value="desc" /><value _jt="string" value="description" /></entry></object>
```

where:

name

Is the group, role, portal, and folder name.

description

Is the group, role, portal, and folder description.

Example:

In the following example, a template called *EnterpriseDomain* is being used, which will create a group and folder. The group and folder that are created will have a name of *Sales* with a description of *Sales Domain*.

Request:

`http://localhost:8080/ibi_apps/rs/templates`

Body:

```
IBIRS_action=run&IBIRS_fileName=EnterpriseDomain&IBIRS_vars=<object
_jt="HashMap"><entry><key _jt="string" value="name"/><value _jt="string"
value="Sales"/></entry><entry><key _jt="string" value="desc"/><value
_jt="string" value="Sales Domain"/></entry></object>
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="run"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="2">
    <entry key="IBIRS_vars" value="&lt;object
_jt=&quot;HashMap&quot;&gt;&lt;entry&gt;&lt;key _jt=&quot;string&quot;
value=&quot;name&quot;/&gt;&lt;value _jt=&quot;string&quot;
value=&quot;Sales&quot;/&gt;&lt;/entry&gt;&lt;entry&gt;&lt;key
_jt=&quot;string&quot; value=&quot;desc&quot;/&gt;&lt;value
_jt=&quot;string&quot; value=&quot;Sales Domain&quot;/&gt;&lt;/
entry&gt;&lt;/object&gt;"/>
    <entry key="IBIRS_fileName" value="EnterpriseDomain"/>
  </ibfsparams>
  <rootObject _jt="string"/>
</ibfsrpc>
```

If the value for the *returncode* attribute in the XML response is 10000, then the template ran successfully.

Changing a Password for a User

This RESTful web service request can be used to change the password for a user.

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Body Format:

`IBIRS_action=changePassword&IBIRS_userName=Userid&IBIRS_password=Password`

where:

Userid

Is the name of the user ID in which the password will be changed.

Password

Is the new password.

Example:

In the following example, the password for user ID *restid* is changed to *rest10*.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs`

Body:

`IBIRS_action=changePassword&IBIRS_userName=restid&IBIRS_password=rest10`

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="changePassword"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple"><ibfsparams size="2">
<entry key="IBIRS_password" value="****"/><entry key="IBIRS_userName"
value="restid"/></ibfsparams><rootObject _jt="IBFSUserObject"
description="Rest Userid" dummy="false"
email="restid@informationbuilders.com" fullPath="IBFS:/SSYS/USERS/restid"
handle="1811177469" length="0" name="restid" nameSpace="DB" policy="f//
3s///99H/7///9v/9///f//+AAAAA==" rsPath="/ibi_apps/rs/ibfs/SSYS/USERS/
restid" type="User">
<status _jt="IBSSUserStatus" name="ACTIVE"/><groups _jt="ArrayList"
size="0"/><pSetList _jt="ArrayList" size="0"/></rootObject></ibfsrpc>
```


This section describes the format and structure of ReportCaster RESTful web service requests.

In this chapter:

- [Retrieving Reports From the ReportCaster Library](#)
 - [Deleting a Version of a Report From the ReportCaster Library](#)
 - [Creating and Updating an Address Book](#)
 - [Creating and Updating a Library Access List](#)
 - [Deleting a Library Access List](#)
 - [Creating and Updating a Schedule](#)
 - [Running a Schedule](#)
 - [Retrieving a Schedule](#)
 - [Deleting a Schedule](#)
 - [Deleting an Address Book](#)
 - [Log Functionality](#)
 - [Console Functionality](#)
-

Retrieving Reports From the ReportCaster Library

This RESTful web service request can be used to retrieve a report for a specific version from the ReportCaster Library.

HTTP Method: POST

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ContentName`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder used for the stored WebFOCUS report. If the folder used for the stored WebFOCUS report exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

ContentName

Is the name of the stored WebFOCUS report as defined in the *name* attribute when listing the content of a folder. For more information, see [Listing Reports, Schedules, and Library Content Within the WebFOCUS Repository](#) on page 50.

Body Format:

`IBIRS_action=run&IBIRS_args=Object`

where:

Object

Is the XML object that defines the version of the report that is to be retrieved.

```
<object _jt="HashMap">
  <entry>
    <key _jt="string" value="IBFS_content_revision"/>
    <value _jt="intval" value="versionNumber"/>
  </entry>
</object>
```

where:

versionNumber

Is the version of the report that is to be retrieved.

Note: The `IBIRS_args` parameter is optional. If it is left out, then the last version of the report will be retrieved.

Example:

In the following example, the 12th version of a stored library report identified by L1748ltvgq02.lib within the Car_Reports folder is retrieved. The Car_Reports folder is a subfolder of the RESTful_Web_Services folder. L1748ltvgq02.lib is defined in the name attribute when listing the content of a folder. For more information, see [Listing Reports, Schedules, and Library Content Within the WebFOCUS Repository](#) on page 50.

POST Request URL:


```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/
Car_Reports/L1748ltvgq02.lib
```

Body:

```
IBIRS_action=run&IBIRS_args=<object _jt="HashMap"><entry><key _jt="string"
value="IBFS_content_revision"/><value _jt="intval" value="12"/></entry></object>
```

Response:

The response is a report in either HTML, Excel, PDF, active report, or a Graph.

Deleting a Version of a Report From the ReportCaster Library

This RESTful web service request can be used to delete a specific version of a report from the ReportCaster Library.

HTTP Method: POST

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ContentName
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder used for the stored WebFOCUS report. If the folder used for the stored WebFOCUS report exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

ContentName

Is the name of the stored WebFOCUS report as defined in the *name* attribute when listing the content of a folder. For more information, see [Listing Reports, Schedules, and Library Content Within the WebFOCUS Repository](#) on page 50.

Body Format:

```
IBIRS_action=run&IBIRS_args=Object
```

where:

Object

Is the XML object that defines the version of the report that is to be deleted.

```
<object _jt="HashMap">
  <entry>
    <key _jt="string" value="IBFS_content_revision"/>
    <value _jt="intval" value="deleteversions"/>
    <value _jt="boolval" value="true"/>
  </entry>
</object>
```

where:

deleteversions

Is the version of the report that is to be deleted.

Creating and Updating an Address Book

This RESTful web service request can be used to create or update a ReportCaster Address Book, which is used by ReportCaster schedules to distribute reports using email, FTP, or a printer.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/AddressBookName

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder where the ReportCaster Address Book is stored. If the folder that is used to store the Address Book exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

AddressBookName

Is the name of the ReportCaster Address Book to add or update, which must have a .adr extension.

Body Format:

```
IBIRS_action=put&IBIRS_object=Object&IBIRS_replace=ReplaceAddressBook&IBIRS_private=MakePrivate&
IBIRS_args=AddEntriesObject
```

where:

Object

Is the XML object that defines the ReportCaster Address Book.

```
<rootObject _jt="IBFSCasterObject"
description="AddressBookDescription"
type="CasterDistributionList"><casterObject _jt="CasterAddrBook"
access="AccessType" bookName="AddressBookName"
description="AddressBookDescription" method="Method"
owner="Owner"><destinationList _jt="array"
itemsClass="CasterAddrbookDestinationElement" size="numberOfItems"><item
_jt="CasterAddrbookDestinationElement" burstValue="BurstValue"
burstValueType="BurstValueType" index="indexValue"
location="Location"/></destinationList></casterObject></rootObject>
```

where:

AddressBookDescription

Is the title for the Address Book.

AccessType

Specifies the security level of an Address Book, which can be set to *PUBLIC* or *PRIVATE*. A public Address Book can be viewed by all users, while a private Address Book can be viewed only by the owner and the Administrator.

AddressBookName

Is the name of the Address Book to add or update, which must have a *.adr* extension. For example, *REST_Distribution_List.adr*.

Method

Specifies the distribution method for an Address Book, which can be set to *FTP*, *EMAIL*, or *PRINT*.

Owner

Indicates the owner of an Address Book. The user ID specified will be associated with the Address Book as the owner, and will have privileges to view and modify the Address Book.

numberOfItems

Is the number of members that will be added to the Address Book.

BurstValue

If *BurstValueType* is set to *P*, then *BurstValue* is the value used when bursting a report.

If *BurstValueType* is set to *W*, then an asterisk (*) and a question mark (?) can be used as wild cards to represent characters at the beginning, end, or middle of the burst values. For example:

`a?c*`

In this case, all values that start with letter *a* and have letter *c* as the third character are returned.

If *BurstValueType* is set to *R*, then Java regular expressions can be used to identify strings of text. Precede each instance of a burst value using a Java regular expression. For example:

`[bcr]at`

In this case, all values that are *bat*, *cat*, or *rat* are returned.

If *BurstValueType* is set to *E*, then *BurstValue* should not have a value.

BurstValueType

Specifies one of the following patterns that is used for *BurstValue*:

- P.** Plain Text
- W.** Wildcard
- R.** Regular Expression
- E.** Else Send

indexValue

Is a value that starts at 0 and increments by 1 for every member that is added to the Address Book.

Location

Depending on the distribution method, *Location* may contain an email address, printer destination, or FTP path.

ReplaceAddressbook

Determines whether to update an Address Book. Select one of the following options:

- true.** Updates an Address Book. To update an Address Book, the existing Address Book must be retrieved. The retrieved XML object would then be modified and then used as input. The following REST URL retrieves an existing Address Book:

```
http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/AddressBookName?
IBIRS_action=get
```

- false.** Does not update an Address Book.

MakePrivate

Determines whether to make an Address Book private. Specify *true* or *false*.

AddEntriesObject (optional)

Is the XML object that is used to indicate that additional entries are to be added to the Address Book. The Address Book must first be retrieved and the additional entries must be included within the *destinationList* tags as part of the *Object* definition for *IBIRS_object*. The existing entries in the Address Book do not have to be included within the *destinationList* tags. *IBIRS_replace* should be set to *true*.

```
<object _jt="HashMap">
  <entry>
    <key _jt="string" value="insertitems"/>
    <value _jt="string" value="true"/>
  </entry>
</object>
```

Example 1:

In this example:

- An Address Book called REST_Distribution_List.adr is added.
- The description for the Address Book is REST Distribution List.
- The Address Book is used for an email distribution.
- The Address Book will be private.
- For a burst value of JAPAN, the report will be emailed to rest@informationbuilders.com.
- For all burst values except for JAPAN, the report will be emailed to other@informationbuilders.com.

POST Request URL:

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_Distribution_List.adr

Body:

```
IBIRS_action=put&IBIRS_object=<rootObject
  _jt="IBFSCasterObject" description="REST Distribution List"
  type="CasterDistributionList"><casterObject _jt="CasterAddrBook"
  access="PRIVATE" bookName="REST_Distribution_List.adr" description="REST
  Distribution List" method="EMAIL" owner="admin"><destinationList
  _jt="array" itemsClass="CasterAddrbookDestinationElement" size="2"><item
  _jt="CasterAddrbookDestinationElement" burstValue="" burstValueType="E"
  index="0" location="other@informationbuilders.com"/><item
  _jt="CasterAddrbookDestinationElement" burstValue="JAPAN"
  burstValueType="P" index="1"
  location="rest@informationbuilders.com"/></destinationList></casterObject
></rootObject>&IBIRS_replace=false&IBIRS_private=true
```

Response:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="put"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="5">
    <entry key="IBIRS_replace" value="false"/>
    <entry key="IBIRS_private" value="true"/>
    <entry key="IBIRS_object" value="&lt;rootObject
_jt=&quot;IBFSCasterObject&quot; description=&quot;REST Distribution
List&quot; type=&quot;CasterDistributionList&quot;&gt;&lt;casterObject
_jt=&quot;CasterAddrBook&quot; access=&quot;PRIVATE&quot;
bookName=&quot;REST_Distribution_List.adr&quot; description=&quot;REST
Distribution List&quot; method=&quot;EMAIL&quot;
owner=&quot;admin&quot;&gt;&lt;destinationList _jt=&quot;array&quot;
itemsClass=&quot;CasterAddrbookDestinationElement&quot; size=&quot;
2&quot;&gt;&lt;item _jt=&quot;CasterAddrbookDestinationElement&quot;
burstValue=&quot;&quot; burstValueType=&quot;E&quot; index=&quot;0&quot;
location=&quot;other@informationbuilders.com&quot; /&gt;&lt;item
_jt=&quot;CasterAddrbookDestinationElement&quot;
burstValue=&quot;JAPAN&quot; burstValueType=&quot;P&quot; index=&quot;
1&quot; location=&quot;rest@informationbuilders.com&quot; /&gt;&lt;/
destinationLis
t&gt;&lt;/casterObject&gt;&lt;/rootObject&gt; "/>
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/WFC/Repository/RESTful_Web_Services/
Car_Reports/REST_Distribution_List.adr"/>

```

```

    </ibfsparams>
    <rootObject _jt="IBFSCasterObject" defaultLng="en_US" description="REST
Distribution List" dummy="false" extension="adr"
externalId="1a7ddf0eIff6aI4886Ibde9I77c691d280a0" fullPath="/WFC/Repository/
RESTful_Web_Services/
Car_Reports/REST_Distribution_List.adr"
handle="1a7ddf0eIff6aI4886Ibde9I77c691d280a0" length="0"
name="REST_Distribution_List.adr" policy="////D///9+f/////f/////////8AAAA="
rsPath="/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports
/REST_Distribution_List.adr" type="CasterDistributionList">
    <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
        <entry>
            <key _jt="string" value="en_US"/>
            <value _jt="ArrayList" size="1">
                <item _jt="string" index="0" value="REST Distribution
List"/>
            </value>
        </entry>
    </nlsValues>
    <properties size="3">
        <entry key="id" value="1a7ddf0eIff6aI4886Ibde9I77c691d280a0"/>
        <entry key="tool" value="addressbook"/>
        <entry key="method" value="EMAIL"/>
    </properties>
    <casterObject _jt="CasterAddrBook" access="PRIVATE"
bookName="REST_Distribution_List.adr" burstValue="false" description="REST
Distribution List" ibfsId="1a7ddf0eIff6aI4886Ibde9I77c691d280a0"
ibfsPath="" id="1a7ddf0eIff6aI4886Ibde9I77c691d280a0" method="EMAIL"
owner="admin" policy="open,delete,rename,|,security;makeRules;viewRules"
sendMethod="EMAIL" summary="">
        <destinationList _jt="array"
itemsClass="CasterAddrbookDestinationElement" size="2">
            <item _jt="CasterAddrbookDestinationElement" burstValue=""
burstValueType="E" index="0" location="other@informationbuilders.com"/>
            <item _jt="CasterAddrbookDestinationElement"
burstValue="JAPAN" burstValueType="P" index="1"
location="rest@informationbuilders.com"/>
        </destinationList>
    </casterObject>
</rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the Address Book was successfully added.

Example 2:

In this example:

- An Address Book called REST_Distribution_List.adr is updated.
- The description for the Address Book is REST Distribution List.

- The Address Book is used for an email distribution.
- The Address Book will be private.
- For a burst value of ITALY, the report will be emailed to rest@informationbuilders.com.
- For all burst values except for ITALY, the report will be emailed to other@informationbuilders.com.

The following REST URL retrieves an existing Address Book:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_Distribution_List.adr?IBIRS_action=get
```

POST Request URL:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_Distribution_List.adr
```

Body:

```
IBIRS_action=put&IBIRS_object=<rootObject
_jt="IBFSCasterObject" binary="false" createdOn="1350862349237"
defaultLng="en_US" description="REST Distribution List" dummy="false"
effectiveRSName="EDASERVE" extension="adr"
externalId="1a7ddf0eIff6aI4886Ibde9I77c691d280a0"
fullPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_Distribution_List.adr" handle="1a7ddf0eIff6aI4886Ibde9I77c691d280a0"
lastModified="1350862349237" lastaccessBy="admin"
lastaccessOn="1350862566520" length="0" name="REST_Distribution_List.adr"
ownerId="10001" ownerName="admin" ownerType="U"
policy="//3/D//9+f/////f/////////8AAAA=" returnedLng="en_US"
rsPath="/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_Distribution_List.adr" signedOn="true"
type="CasterDistributionList"><nlsValues _jt="HashMap" loadFactor="0.75"
threshold="12"><entry><key _jt="string" value="en_US"/><value
_jt="ArrayList" size="2"><item _jt="string" index="0" value="REST
Distribution List"/></value></entry></nlsValues><properties
size="3"><entry key="id"
value="1a7ddf0eIff6aI4886Ibde9I77c691d280a0"/><entry key="tool"
value="addressbook"/><entry key="method"
value="EMAIL"/></properties><casterObject _jt="CasterAddrBook"
access="PRIVATE" bookName="REST_Distribution_List.adr" burstValue="false"
description="REST Distribution List"
ibfsId="1a7ddf0eIff6aI4886Ibde9I77c691d280a0"
ibfsPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports"
id="1a7ddf0eIff6aI4886Ibde9I77c691d280a0" method="EMAIL" owner="admin"
policy="open,delete,rename,|,security;makeRules;viewRules"
```

```
sendMethod="EMAIL"><destinationList _jt="array"
itemsClass="CasterAddrbookDestinationElement" size="2"><item
_jt="CasterAddrbookDestinationElement" burstValue=" " burstValueType="E"
index="0" location="other@informationbuilders.com"/><item
_jt="CasterAddrbookDestinationElement" burstValue="ITALY"
burstValueType="P" index="1"
location="rest@informationbuilders.com"/></destinationList></casterObject
></rootObject>&IBIRS_replace=true&IBIRS_private=true
```

Response:

If the value for the *returncode* attribute in the XML response is 10000, then the Address Book was successfully updated.

Example 3:

In this example:

- Additional entries are added to the REST_Distribution_List.adr Address Book.
- For a burst value of ENGLAND, the report will be emailed to rest2@informationbuilders.com.
- For a burst value of FRANCE, the report will be emailed to rest3@informationbuilders.com.

The following REST URL retrieves an existing Address Book:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/
REST_Distribution_List.adr?IBIRS_action=get
```

POST Request URL:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/
REST_Distribution_List.adr
```

Body:

```

IBIRS_action=put&IBIRS_object= <rootObject _jt="IBFSCasterObject"
createdOn="1393510291277" defaultLng="en_US" description="REST Distribution List"
dummy="false" effectiveRSName="EDASERVE" extension="adr"
externalId="f7c08730I4adfI4c8aIb109I8e014fac5a23" fullPath="IBFS:/WFC/Repository/
RESTful_Web_Services/Car_Reports/REST_Distribution_List.adr"
handle="f7c08730I4adfI4c8aIb109I8e014fac5a23" inheritedPrivacy="true"
lastModified="1393510291277" lastaccessBy="admin" lastaccessOn="1393510324927"
length="0" name="REST_Distribution_List.adr" ownerId="10001" ownerName="admin"
ownerType="U" policy="//7/w///38f9///v9/////+AAAA" returnedLng="en_US" rsPath="/
ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/
REST_Distribution_List.adr" type="CasterDistributionList"><properties size="3"><entry
key="id" value="f7c08730I4adfI4c8aIb109I8e014fac5a23"/><entry key="tool"
value="addressbook"/><entry key="method" value="EMAIL"/></properties><nlsValues
_jt="HashMap" loadFactor="0.75" threshold="12"><entry><key _jt="string" value="en_US"/
><value _jt="ArrayList" size="2"><item _jt="string" index="0" value="REST Distribution
List"/></value></entry></nlsValues><casterObject _jt="CasterAddrBook" access="PRIVATE"
bookName="REST_Distribution_List.adr" burstValue="false" description="REST
Distribution List" ibfsId="f7c08730I4adfI4c8aIb109I8e014fac5a23" ibfsPath="IBFS:/WFC/
Repository/RESTful_Web_Services/Car_Reports" id="f7c08730I4adfI4c8aIb109I8e014fac5a23"
method="EMAIL" owner="admin"
policy="open,delete,rename,|,security;makeRules;viewRules"
sendMethod="EMAIL"><destinationList _jt="array"
itemsClass="CasterAddrbookDestinationElement" size="2"><item
_jt="CasterAddrbookDestinationElement" burstValue="ENGLAND" burstValueType="P"
index="0" location="rest2@informationbuilders.com"/><item
_jt="CasterAddrbookDestinationElement" burstValue="FRANCE" burstValueType="P"
index="1" location="rest3@informationbuilders.com"/></destinationList></casterObject></
rootObject>&IBIRS_replace=true&
IBIRS_private=true&IBIRS_args=<object _jt="HashMap">
<entry><key _jt="string" value="insertitems"/><value _jt="string" value="true"/></
entry></object>

```

Response:

If the value for the returncode attribute in the XML response is 10000, then the additional entries were successfully added to the Address Book.

Creating and Updating a Library Access List

This RESTful web service request creates or updates a ReportCaster Library Access List that can be used by ReportCaster schedules when the distribution is set to the Report Library. If a schedule is defined to use a Library Access List, then Users or Groups defined in the list are granted access to view the part of the report that is stored in the Library, which they have access to, based on a Burst Value. If a Burst Value is not supplied for a particular Access List entry, then the User or Group in the definition will be able to view the entire report.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/LibraryAccessListName

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder where the ReportCaster Library Access List is stored. If the folder that is used to store the Library Access List exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

LibraryAccessListName

Is the name of the ReportCaster Library Access List to add or update, which must have a .acl extension.

Body Format:

IBIRS_action=put&IBIRS_object=Object&IBIRS_replace=ReplaceAccessList&IBIRS_private=MakePrivate

where:

Object

Is the XML object that defines the ReportCaster Library Access List.

```
<rootObject _jt="IBFSCasterObject"
description="AccessListDescription"
type="CasterAccessList"><casterObject _jt="CasterLibraryAccessBook"
burstValue="burstValueFlag" description="AccessListDescription"
owner="Owner"><accessElementList _jt="array"
itemsClass="CasterLibAccessElement" size="numberOfItems">
<item _jt="CasterLibAccessElement" burstValue="burstValue"
index="indexValue" memberName="member" MemberType="memberType" />
</accessElementList></casterObject></rootObject>
```

where:

AccessListDescription

Is the title for the Library Access List.

burstValueFlag

Specify one of the following:

- true.** The Library Access List will be used to burst reports based on a value in each member definition.
- false.** The Library Access List will not be used to burst reports.

Owner

Is the owner of the Library Access List.

numberOfItems

Is the number of members that will be added to the Library Access List.

burstValue

Is the value used in bursting a report.

indexValue

Is a value that starts at 0 and increments by 1 for every member that is added to the Library Access List.

member

Is the user name or group that will be added as a member of the Library Access List.

memberType

Specify U for user or G for group.

ReplaceAccessList

Specify one of the following:

- true.** Update the Library Access List.

To update a Library Access List, the existing Library Access List must be retrieved. The retrieved XML object would then be modified and then used as input.

The following REST URL retrieves an existing Library Access List:

```
http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/  
LibraryAccessListName?IBIRS_action=get
```

- false.** Do not update Library Access List.

MakePrivate

Determines whether to make a Library Access List private. Specify *true* or *false*.

Example 1:

In this example:

- A Library Access List called *RESTAccessList.acl* is added.
- The description for the Access List is *REST Access List*.
- The Library Access List will be private.
- User ID *daniel* will view the part of the report where the first sort value is equal to *FRANCE*.
- User ID *david* will view the part of the report where the first sort value is equal to *JAPAN*.
- User ID *efrem* will view the part of the report where the first sort value is equal to *ENGLAND*.
- User ID *gerry* will view the part of the report where the first sort value is equal to *ITALY*.

POST Request URL:

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/RESTAccessList.acl

Body:

```
IBIRS_action=put&IBIRS_object=<rootObject
 _jt="IBFSCasterObject" description="REST Access List"
 type="CasterAccessList"><casterObject _jt="CasterLibraryAccessBook"
 burstValue="true" description="REST Access List"
 owner="admin"><accessElementList _jt="array"
 itemsClass="CasterLibAccessElement" size="4"><item
 _jt="CasterLibAccessElement" burstValue="FRANCE" index="0"
 memberName="daniel" memberType="U"/><item _jt="CasterLibAccessElement"
 burstValue="JAPAN" index="1" memberName="david" memberType="U"/><item
 _jt="CasterLibAccessElement" burstValue="ENGLAND" index="2"
 memberName="efrem" memberType="U"/><item _jt="CasterLibAccessElement"
 burstValue="ITALY" index="3" memberName="gerry"
 memberType="U"/></accessElementList></casterObject></rootObject>&IBIRS_re
 place=false&IBIRS_private=true
```

Response:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="put"
returncode="1000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="5">
    <entry key="IBIRS_replace" value="false"/>
    <entry key="IBIRS_private" value="true"/>
    <entry key="IBIRS_object" value="&lt;rootObject
_jt=&quot;IBFSCasterObject&quot; description=&quot;REST Access List&quot;
type=&quot;CasterAccessList&quot;&gt;&lt;casterObject
_jt=&quot;CasterLibraryAccessBook&quot; burstValue=&quot;true&quot;
description=&quot;REST Access List&quot;
owner=&quot;admin&quot;&gt;&lt;accessElementList _jt=&quot;array&quot;
itemsClass=&quot;CasterLibAccessElement&quot; size=&quot;
4&quot;&gt;&lt;item _jt=&quot;CasterLibAccessElement&quot;
burstValue=&quot;FRANCE&quot; index=&quot;0&quot;
memberName=&quot;daniel&quot; memberType=&quot;U&quot;/&gt;&lt;item
_jt=&quot;CasterLibAccessElement&quot; burstValue=&quot;JAPAN&quot;
index=&quot;1&quot; memberName=&quot;david&quot;

```

```

    memberType="U" /><item
    _jt="CasterLibAccessElement" burstValue="ENGLAND"
    index="2" memberName="efrem" memberType="U" />
    <item _jt="CasterLibAccessElement"
    burstValue="ITALY" index="3"
    memberName="gerry" memberType="U" />
    accessElementList></item>
  /casterObject></rootObject>"/>
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/WFC/Repository/RESTful_Web_Services/
    Car_Reports/
    RESTAccessList.acl"/>
  </ibfsparams>
  <rootObject _jt="IBFSCasterObject" defaultLng="en_US" description="REST
  Access List" dummy="false" extension="acl"
  externalId="C34ea5140c31c0c4f68c8534ca97cd4538363" fullPath="/WFC/
  Repository/RESTful_Web_Services/Car_Reports/RESTAccessList.acl"
  handle="5ca19e73I55f0I4c4cI9cd1I48340f7da5d5" length="0"
  name="RESTAccessList.acl" policy="///D///9+f///f/////////8AAAA="
  rsPath="/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/
  Car_Reports/RESTAccessList.acl" type="CasterAccessList">
    <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
      <entry>
        <key _jt="string" value="en_US"/>
        <value _jt="ArrayList" size="1">
          <item _jt="string" index="0" value="REST Access List"/>
        </value>
      </entry>
    </nlsValues>
    <properties size="2">
      <entry key="id" value="C34ea5140c31c0c4f68c8534ca97cd4538363"/>
      <entry key="tool" value="accesslist"/>
    </properties>
    <casterObject _jt="CasterLibraryAccessBook" burstValue="true"
    description="REST Access List"
    ibfsId="5ca19e73I55f0I4c4cI9cd1I48340f7da5d5" ibfsPath=""
    id="C34ea5140c31c0c4f68c8534ca97cd4538363" name="" owner="admin"
    policy="open,delete,rename,|,security;makeRules;viewRules" summary="">
      <accessElementList _jt="array"
      itemsClass="CasterLibAccessElement" size="4">
        <item _jt="CasterLibAccessElement" burstValue="FRANCE"
        index="0" memberName="daniel" memberType="U"/>
        <item _jt="CasterLibAccessElement" burstValue="JAPAN"
        index="1" memberName="david" memberType="U"/>
        <item _jt="CasterLibAccessElement" burstValue="ENGLAND"
        index="2" memberName="efrem" memberType="U"/>
        <item _jt="CasterLibAccessElement" burstValue="ITALY"
        index="3" memberName="gerry" memberType="U"/>
      </accessElementList>
    </casterObject>
  </rootObject>
</ibfsrpc>

```


If the value for the *returncode* attribute in the XML response is 10000, then the Library Access List was added successfully.

Example 2:

In this example:

- ❑ A Library Access List called *RESTAccessList.acl* is updated.
- ❑ The description for the Access List is *REST Access List*.
- ❑ The Library Access List will be private.
- ❑ User ID *daniel* will view the part of the report where the first sort value is equal to *FRANCE*.
- ❑ User ID *david* will view the part of the report where the first sort value is equal to *JAPAN*.
- ❑ User ID *efrem* will view the part of the report where the first sort value is equal to *ENGLAND*.
- ❑ User ID *gerry* will view the part of the report where the first sort value is equal to *ITALY*.

The following REST URL retrieves an existing Library Access List:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/  
Car_Reports/RESTAccessList.acl?IBIRS_action=get
```

POST Request URL:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/  
Car_Reports/RESTAccessList.acl
```

Body:

```

IBIRS_action=put&IBIRS_object=<rootObject
  _jt="IBFSCasterObject" binary="false" createdOn="1349797553600"
  defaultLng="en_US" description="REST Access List - Updated" dummy="false"
  effectiveRSName="EDASERVE" extension="acl"
  externalId="C34ea5140c31c0c4f68c8534ca97cd4538363"
  fullPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/RESTAccess
sList.acl" handle="5ca19e73155f014c4cI9cd1I48340f7da5d5"
  lastModified="1349797553600" lastaccessBy="admin"
  lastaccessOn="1349797663457" length="0" name="RESTAccessList.acl"
  ownerId="10001" ownerName="admin" ownerType="U"
  policy="//3/D///9+f/////f/////////8AAAA=" returnedLng="en_US"
  rsPath="/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports
/RESTAccessList.acl" type="CasterAccessList"><nlsValues _jt="HashMap"
  loadFactor="0.75" threshold="12"><entry><key _jt="string"
  value="en_US"/><value _jt="ArrayList" size="2"><item _jt="string"
  index="0" value="REST Access List -
Updated"/></value></entry></nlsValues><properties size="2"><entry
  key="id" value="C34ea5140c31c0c4f68c8534ca97cd4538363"/><entry key="tool"
  value="accesslist"/></properties><casterObject
  _jt="CasterLibraryAccessBook" burstValue="true" description="REST Access
List - Updated" ibfsId="5ca19e73155f014c4cI9cd1I48340f7da5d5"
  ibfsPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports"
  id="C34ea5140c31c0c4f68c8534ca97cd4538363" name="RESTAccessList.acl"
  owner="admin"
  policy="open,delete,rename,|,security;makeRules;viewRules"><accessElement
List _jt="array" itemsClass="CasterLibAccessElement" size="4"><item
  _jt="CasterLibAccessElement" burstValue="FRANCE" index="0"
  memberName="daniel" memberType="U"/><item _jt="CasterLibAccessElement"
  burstValue="JAPAN" index="1" memberName="david" memberType="U"/><item
  _jt="CasterLibAccessElement" burstValue="ENGLAND" index="2"
  memberName="efrem" memberType="U"/><item _jt="CasterLibAccessElement"
  burstValue="ITALY" index="3" memberName="gerry"
  memberType="U"/></accessElementList></casterObject></rootObject>&IBIRS_re
place=true&IBIRS_private=true

```

Response:

If the value for the *returncode* attribute in the XML response is 10000, then the Library Access List was updated successfully.

Deleting a Library Access List

This RESTful web service request can be used to delete a ReportCaster Library Access List.

HTTP Method: DELETE

REST URL Format:

```

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/AccessListName?
IBIRS_action=delete

```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder where the ReportCaster Library Access List is stored. If the folder that is used to store the Library Access List exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

AccessListName

Is the name of the ReportCaster Library Access List to delete, which must have a .acl extension.

Example:

In the following example, the ReportCaster Library Access List named RESTAccessList.acl is deleted from the Car_Reports folder, which is within the RESTful_Web_Services folder.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/
Car_Reports/RESTAccessList.acl?IBIRS_action=delete
```

Response:

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="delete"
returncode="10000" returndesc="SUCCESS" subreturncode="0" subsystem="SSYS"
type="simple">
  <ibfsparams size="2">
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/WFC/Repository/RESTful_Web_Services/
Car_Reports/
RESTAccessList.acl"/>
  </ibfsparams>
```

```

    <rootObject _jt="IBFSCasterObject" binary="false"
createdOn="1349435037177" defaultLng="en_US" description="REST Access List"
dummy="false" extension="acl"
externalId="C3222b6bcc30c0c4582c90fdcc4c403cd249c" fullPath="IBFS:/WFC/
Repository/RESTful_Web_Services/Car_Reports/RESTAccessList.acl"
handle="b60b3b27I4bd0I4a15I923cI7db3bd6ae555" lastModified="1349435037177"
lastaccessBy="admin" lastaccessOn="1349436904650" length="0"
name="RESTAccessList.acl" policy="////D//9+f/////f/////////8AAAA="
returnedLng="en_US" rsPath="/ibi_apps/rs/ibfs/WFC/Repository/
RESTful_Web_Services/
Car_Reports/RESTAccessList.acl" type="CasterAccessList">
    <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
        <entry>
            <key _jt="string" value="en_US"/>
            <value _jt="ArrayList" size="2">
                <item _jt="string" index="0" value="REST Access List"/>
            </value>
        </entry>
    </nlsValues>
    <properties size="2">
        <entry key="id" value="C3222b6bcc30c0c4582c90fdcc4c403cd249c"/>
        <entry key="tool" value="accesslist"/>
    </properties>
</rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the ReportCaster Library Access List was deleted successfully.

Creating and Updating a Schedule

This section describes the structure of the RESTful web service request that is used to create and update a ReportCaster Schedule.

HTTP Method: POST

REST URL Format:

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ScheduleName

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder that will contain the ReportCaster Schedule. If the folder used for the Schedule is a subfolder, the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName

ScheduleName

Is the name of the ReportCaster Schedule to be added or updated, which also must have a .sch extension.

Body Format:

IBIRS_action=put&IBIRS_replace=ReplaceSchedule&IBIRS_object=Object

where:

ReplaceSchedule

Specify one of the following options:

- True.** Updates the Schedule. To update a Schedule, the existing Schedule must be retrieved. The retrieved XML object would then be modified and then used as input.

The following REST URL retrieves an existing Schedule:

[http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ScheduleName?
IBIRS_action=get](http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ScheduleName?IBIRS_action=get)

- False.** Does not update the Schedule.

Object

Is the XML object that defines the ReportCaster Schedule. The XML object consists of seven components, which are concatenated in the following order:

- Schedule rootObject
- Schedule properties
- Notification
- Distribution
- Recurrence
- Task
- Closing tags

Schedule rootObject

This section describes the Schedule rootObject.

Body Format:

```
<rootObject _jt="IBFSCasterObject" description="ScheduleTitle"
type="CasterSchedule">
```

where:

ScheduleTitle

Is the text describing the job that is being scheduled. The maximum size of the description is 90 characters.

Schedule Properties

This section describes the Schedule properties.

Body Format:

```
<casterObject _jt="CasterSchedule" active="Active"
deleteJobAfterRun="DeleteJobAfterRun" description="ScheduleTitle"
owner="Owner" priority="Priority" traceType="TraceType">
```

where:

Active

Is the flag indicating whether or not a Schedule is active. If set to *true*, the Schedule is active. If set to *false*, the Schedule is inactive.

DeleteJobAfterRun

Is a flag indicating whether or not a Schedule is deleted after running the job. If set to *true*, the job is deleted after all tasks are completed. If set to *false*, the job is not deleted.

ScheduleTitle

Is the text describing the job that is being scheduled. The maximum size of the description is 90 characters.

Owner

Is the owner of this Schedule. The maximum size of the owner is 48 characters.

Priority

Is the priority level for the scheduled job. The value ranges from 1 (highest priority) to 5 (lowest priority).

TraceType

Specify one of the following types of tracing:

- 0 = Default Trace. Uses ReportCaster trace configuration setting.
- 1 = No Traces
- 2 = Trace Schedule
- 3 = Trace Schedule and Report

Notification

When scheduled reports are distributed, ReportCaster allows selected individuals to be notified with log information about the distribution. This notification feature can be altered on a per Schedule basis and can be set to *inactive*, *always notify*, or *notify only on error*. Each Schedule allows the following two types of notifications to be sent simultaneously:

- Brief.** Contains partial log information.
- Full.** Contains complete log information.

Body Format:

```
<notification _jt="CasterScheduleNotification"
addressForBriefNotification="BriefNotificationAddress"
  addressForFullNotification="FullNotificationAddress" description=""
from="FromAddress"
  subject="Subject" type="NotificationType" />
```

where:

BriefNotificationAddress

Is the email address where a brief notification message will be sent after running a Schedule in ReportCaster. The content of the brief notification email is the partial log information for a given Schedule run. The maximum size of the brief notification email address is 75 characters.

FullNotificationAddress

Is the email address where a full notification message will be sent upon running a Schedule in ReportCaster. The content of the full notification email is the complete log information for a given Schedule run. The maximum size of the full notification email address is 75 characters.

FromAddress

Is the email address linked to the From header to which the notification will be sent upon running a Schedule in ReportCaster. The maximum size of the From address is 75 characters.

Subject

Is the subject header in the email to which the notification will be sent upon running a Schedule in ReportCaster. The maximum size of the email subject is 255 characters.

NotificationType

Is the type of notification message to be sent upon the running of a ReportCaster Schedule. The three possible types are *ALWAYS*, *INACTIVE*, and *ONERROR*.

Distribution

There are five distribution types to choose from when creating a ReportCaster Schedule:

- Report Library
- Email
- FTP
- Printer
- WebFOCUS Repository

Report Library

This is used when the intended distribution method for the scheduled ReportCaster job is storage in the ReportCaster Library. The ReportCaster Library is a secure archiving environment that is configured in a database and provides common access. It is optionally available with the ReportCaster product.

Body Format:

```
<distributionList _jt="array" itemsClass="CasterScheduleDistribution"
size="1">
  <item accessListFullPath="AccessListPath" accessType="AccessType"
category="Category"
      compressionEnabled="CompressionEnabled"
      description="DistributionName" destinationPath="DestinationPath"
enabled="true">
```



```

        expirationData="ExpirationData" expirationMode="ExpirationMode"
index="0" valueonly="ValueOnly">
  <storageLibraryEmail authEnabled="AuthEnabled"
authPassword="AuthPassword" authUserId="AuthUserId"
  libraryURL="LibraryURL" mailFrom="MailFrom"
  mailMessage="MailMessage" mailReplyAddress="MailReply"
mailServerName="MailServer"
  mailSubject="MailSubject"
sendEmailAfterSaveReport="SendEmailAfterSaveReport" sslEnabled="SSLflag"
  tlsEnabled="TLSflag" />
</item>
</distributionList>

```

where:

AccessListPath

Is the full path to the Access List used to control the viewing of the library content when AccessType is set to ACCESS_LIST. For example, IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/RESTAccessList.acl.

AccessType

Is the access type for this library distribution. The access type contains the following three options for viewing a library report:

- PUBLIC
- OWNER
- ACCESS_LIST

Category

Is the library category associated with this report. Each category is a root directory within the library used to organize the distribution of reports.

Any task other than a WebFOCUS Repository report that is delivered to the library must be assigned a category when a Schedule is created. The scheduled report and all subsequent versions of the report are filed in subdirectories under the category assigned to it. If the category does not exist, a new category (for example, root directory) is created upon distribution.

The category accepts a maximum of 90 characters.

CompressionEnabled

Specify one of the following options:

- True.** The report is compressed before it is stored in the library.

-
- False.** The report is not compressed before it is stored in the library.

DistributionName

Is a name that is assigned to the distribution (for example, Report Library).

DestinationPath

Is the path to the folder where Library Content will be stored. For example:

`IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports`

ExpirationData

Is the expiration data used to calculate the expiration of a library resident report. For a given ExpirationMode, the expiration data represents a corresponding integer that, when combined with the ExpirationMode, determines if a report should expire. For example, if the ExpirationMode is set to D and the ExpirationData is set to 3, the report will expire in three days. For V, the ExpirationData represents the threshold number of versions that must exist prior to the report expiring from the library.

ExpirationMode

Is the basis of calculating when a library report will expire. There are seven expiration modes:

- D.** Day
- H.** Hour
- M.** Month
- V.** Version
- W.** Week
- Y.** Year
- N.** Never

When associated with a corresponding ExpirationData integer, ReportCaster can determine when a library report will expire. For example, if the ExpirationMode is set to D, and the ExpirationData is set to 3, the report will expire in three days.

For V, the ExpirationData represents the threshold number of versions that must exist prior to the report expiring from the library.

ValueOnly

Specify one of the following options:

- True.** The distribution to values are in the Access List is limited.
- False.** The distribution to values in the Access List is not limited.

AuthEnabled

Specify one of the following options:

- True.** The Mail Server requires authentication.
- False.** The Mail Server does not require authentication.

AuthPassword

Is the password used to authenticate to the Mail Server if *AuthEnabled* is set to *true*.

AuthUserid

Is the account name used to authenticate to the Mail Server if *AuthEnabled* is set to *true*.

LibraryURL

Is the base URL contained in a library email notification. When library notification is turned on using `SendEmailAfterSaveReport`, all users who have access to a library report are sent an email that contains message content, usually a notification that the report is available, and a URL that opens the report in the browser. The base URL can be set to a value that is accessible inside or outside of the ReportCaster environment.

The `LibraryURL` accepts a maximum of 128 characters. For example:

```
http://localhost:8080/ibi_apps/library/report.rc
```

MailFrom

Is the From email address in this library email notification (library email notification must be turned on through `SendEmailAfterSaveReport`). The maximum size of the From email address is 65 characters.

MailMessage

Is the email message content contained in the email message sent out as part of this notification email (library email notification must be turned on through `SendEmailAfterSaveReport`). The maximum size of the email message is 255 characters.

MailReply

Is the Reply email address sent in this library email notification (library email notification must be turned through `SendEmailAfterSaveReport`). The maximum size of the Reply email address is 65 characters.

MailServer

Is the mail server name used to send this library email notification (library email notification must be turned on through `SendEmailAfterSaveReport`). The maximum size of the mail server name is 65 characters.

MailSubject

Is the email subject sent in this library email notification (library email notification must be turned on through `SendEmailAfterSaveReport`). The maximum size of the Subject email header is 255 characters.

SendEmailAfterSaveReport

Specifies whether or not an email notification is sent after a report is saved to the library. If the value is set to *true*, an email notification is sent to users who have access to the report. If the value is set to *false*, no email notification is sent.

SSLflag

Specify one of the following options:

- True.** The Mail Server requires a secure SSL connection.
- False.** The Mail Server does not require a secure SSL connection.

TLSflag

Specify one of the following options:

- True.** The Mail Server requires a secure TLS connection.
- False.** The Mail Server does not require a secure TLS connection.

Email

This is used when the intended distribution method for the scheduled ReportCaster job is through email.

Body Format:

```

<distributionList _jt="array" itemsClass="CasterScheduleDistribution"
size="1">
  <item _jt="CasterScheduleDistributionEmail" authEnabled="AuthEnabled"
authPassword="AuthPassword" authUserId="AuthUserId"
  description="DistributionName" enabled="true" index="0"
inlineMessage="InlineMessage" inlineTaskIndex="InlineTaskIndex"
  mailFrom="MailFrom" mailReplyAddress="MailReply"
mailServerName="MailServer" mailSubject="MailSubject"
  sendingReportAsAttachment="AttachmentFlag" sslEnabled="SSLflag"
tlsEnabled="TLSflag" zipFileName="ZipFileName"
  zipResult="ZipFlag">
    <destination _jt="CasterScheduleDestination"
distributionFile="DistFile" distributionListFullPath="DistPath"
  singleAddress="SingleAddress" type="Type">
      <dynamicAddress _jt="CasterScheduleDynamicAddress"
password="Password" procedureName="ProcedureName"

          serverName="ServerName" userName="UserName" />
    </destination>
  </item>
</distributionList>

```

where:

AuthEnabled

Specify one of the following options:

- True.** The Mail Server requires authentication.
- False.** The Mail Server does not require authentication.

AuthPassword

Is the password used to authenticate to the Mail Server if *AuthEnabled* is set to *true*.

AuthUserId

Is the account name used to authenticate to the Mail Server if *AuthEnabled* is set to *true*.

DistributionName

Is a name that is assigned to the distribution (for example, Email).

InlineMessage

Is the inline message associated with an email report distribution. An inline message is the message contained in the body of the email when the report is sent as an attachment. If the report is sent inline, this should not be set. The size limit for an inline message is 255 characters.

InlineTaskIndex

Is the index of the task that is going to be inline (in the body of the email). ReportCaster Schedules can accept multiple tasks, with each task representing a report within the Schedule. These tasks will run sequentially. The task index is the sequential index number (from 0 to N) assigned to the tasks within a scheduled distribution. This is particularly important for inline email distribution because only one of the tasks can be an inline report (for example, a report whose contents are in the body of the email). The other reports are sent as an attachment.

MailFrom

Is the email address associated with the From header field of a scheduled email distribution. The size limit for MailFrom is 65 characters.

MailReply

Is the reply email address from the Reply Address header field of a scheduled email distribution. The size limit for mail reply address is 65 characters.

MailServer

Is an SMTP mail server name associated with scheduled email distribution. The size limit for mail server name is 65 characters.

MailSubject

Is an email subject corresponding to the Subject header field associated with scheduled email distribution. The size limit for mail subject is 90 characters.

AttachmentFlag

Specify one of the following options:

- True.** The report is sent as an attachment.
- False.** The report is sent within the body of the email.

SSLflag

Specify one of the following options:

- True.** The Mail Server requires a secure SSL connection.
- False.** The Mail Server does not require a secure SSL connection.

TLSflag

Specify one of the following options:

- True.** The Mail Server requires a secure TLS connection.

- False.** The Mail Server does not require a secure TLS connection.

ZipFileName

Is the name of the zip file associated with a scheduled email distribution. ZipFlag should be set to *true*. The size limit for a zip file name is 64 characters.

ZipFlag

Specify one of the following options:

- True.** The output is zipped.
- False.** The output is not zipped.

DistFile

Is a list of one or many recipients stored within a physical file accessible to the Distribution Server. The Type must be set to DISTRIBUTION_FILE.

DistPath

Is the full path to a ReportCaster Address Book which lists one or many recipients. For example:

```
IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_List.adr
```

The Type must be set to DISTRIBUTION_LIST.

SingleAddress

Are email addresses of the report recipients.

When the email addresses are separated by a comma (,) character, the report is distributed in one email.

When the email addresses are separated by a semicolon (;) character, the report is distributed in multiple emails (one address per email).

The Type must be set to SINGLE_ADDRESS.

Type

Is one of the following valid values:

- DISTRIBUTION_FILE
- DISTRIBUTION_LIST
- DYNAMIC_ADDRESS
- SINGLE_ADDRESS

Password

Is the value of the password required for authentication to the Reporting Server containing the WebFOCUS procedure that creates the dynamic distribution list.

Type must be set to DYNAMIC_ADDRESS.

ProcedureName

Is the name of the WebFOCUS procedure that produces the dynamic distribution list qualified by the application name. For example, ibisamp/getEmails.

Type must be set to DYNAMIC_ADDRESS.

ServerName

Is the name of the Reporting Server that contains the WebFOCUS procedure that creates the dynamic distribution list.

Type must be set to DYNAMIC_ADDRESS.

UserName

Is the user ID to the Reporting Server that contains the WebFOCUS procedure that creates the dynamic distribution list.

Type must be set to DYNAMIC_ADDRESS.

FTP

This is used to distribute a scheduled ReportCaster report through FTP.

Body Format:

```
<distributionList _jt="array" itemsClass="CasterScheduleDistribution"
size="1">
  <item compressionFormat="CompressionFormat" description="DistributionName"
        enabled="true" ftpLocation="FTPlocation" ftpPassword="FTPpass"
ftpServerName="FTPserver" ftpUserName="FTPuser"
        index="0" indexFile="IndexFile" passwordAuthEnabled="passwordFlag"
publicKeyAuthEnabled="publicFlag"
        sftpEnabled="SFTPflag" zipBurstReportsTogether="ZipBurstTogether"
zipFileName="ZipFileName"
        zipResult="ZipResult">
    <destination _jt="CasterScheduleDestination"
distributionFile="DistFile" distributionListFullPath="DistPath"
        singleAddress="SingleFile" type="Type">
      <dynamicAddress _jt="CasterScheduleDynamicAddress"
password="Password" procedureName="ProcedureName"
        serverName="ServerName" userName="UserName" />
    </destination>
  </item>
</distributionList>
```


where:

CompressionFormat

Are the options to compress the output before distribution. You can choose from either of the following:

- 0.** Choose this option if you want an archive file (.zip), a compressed file (.zip), or no compression.
- 1.** Choose this option if you want a compressed file (.gz).

DistributionName

Is a name that is assigned to the distribution (for example, FTP).

FTPlocation

Is the root directory for a report scheduled for distribution through FTP. The report will be sent to this target destination through FTP unless a Distribution List bursts sections of the report to subdirectories of this FTPlocation directory.

FTPpass

Is the password to the target FTP server needed to authenticate at the time of report distribution. The FTP password is part of the credentials necessary for the user to access the FTP server.

FTPserver

Is the name of the FTP server that is the target of the distribution.

FTPuser

Is the user name needed to authenticate to the target FTP server at the time of report distribution. The FTP user name is part of the credentials necessary for the user to have access to the FTP server.

IndexFile

Specifies the index file associated with report(s) scheduled for distribution through FTP, where bursting is activated. When bursting is activated, the index file specifies the name of the file where the corresponding index page will be created.

If bursting is activated (Burst=TRUE), and no index file is specified, the index file name is set to index.htm.

Note. It makes sense to burst a report in cases where the distribution type is: DISTRIBUTION LIST, DISTRIBUTION FILE, and DYNAMIC LIST. In the case where distribution type is SINGLE ADDRESS, there is no need to burst because the reports will be sent to a single address.

passwordFlag

Is the password authentication. If *SFTPflag* equals *true*, specify one of the following options:

- True.** Password authentication is enabled.
- False.** Password authentication is disabled.

publicFlag

Is the Public Key authentication. If *SFTPflag* is set to *true*, specify one of the following options:

- True.** Public Key authentication is enabled.
- False.** Public Key authentication is disabled.

SFTPflag

If set to *true*, the FTP server requires a secure SSH File Transfer Protocol (SFTP).

If set to *false*, the FTP server does not require a secure SSH File Transfer Protocol (SFTP).

ZipBurstTogether

Is the option to use a .zip for compression. If set to *true*, and when *ZipResult* equals *true*, an archive .zip file is created before distribution.

If set to *false*, and when *ZipResult* is set to *true*, a compressed .zip file is created before distribution.

The value will also be set to *false* if no compression is required or *CompressionFormat* is set to 1.

ZipFileName

Is the file name that will contain an archive or compressed .zip file.

CompressionFormat would be set to 0 and *ZipResult* would be set to *true*.

ZipResult

Is the option to use a compressed file. If set to *true*, an archive or compressed .zip file is created before distribution. A compressed .gz file is created before distribution.

CompressionFormat sets the type of compression and *ZipBurstTogether* sets whether an archive or compressed .zip file is created before distribution.

If set to *false*, no compression will occur before distribution.

DistFile

Is a list of one or many locations stored within a physical file accessible to the Distribution Server.

Type must be set to DISTRIBUTION_FILE.

DistPath

Is the full path to a ReportCaster Address Book which lists one or many locations. For example, IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_List.adr.

Type must be set to DISTRIBUTION_LIST

SingleFile

Is the single file name used if distribution is set to one location.

Type must be set to SINGLE_ADDRESS.

Type

The following is a list of valid values:

- DISTRIBUTION_FILE
- DISTRIBUTION_LIST
- DYNAMIC_ADDRESS
- SINGLE_ADDRESS

Password

Is the value of the password required for authentication to the Reporting Server containing the WebFOCUS procedure that creates the dynamic distribution list.

Type must be set to DYNAMIC_ADDRESS.

ProcedureName

Is the name of the WebFOCUS procedure that produces the dynamic distribution list qualified by the application name. For example, ibisamp/getEmails.

Type must be set to DYNAMIC_ADDRESS.

ServerName

Is the name of the Reporting Server that contains the WebFOCUS procedure that creates the dynamic distribution list.

Type must be set to DYNAMIC_ADDRESS.

UserName

Is the user ID to the Reporting Server that contains the WebFOCUS procedure that creates the dynamic distribution list.

Type must be set to DYNAMIC_ADDRESS.

Printer

This is used when the intended distribution method for the scheduled ReportCaster job is through a printer.

When using this distribution, the report format in the Task (SendFormat) must be set to either DOC, WP, PS or PDF (if the printer you are using supports PDF output sent without Adobe).

Body Format:

```
<distributionList _jt="array" itemsClass="CasterScheduleDistribution"
size="1">
  <item description="DistributionName" enabled="true" index="0">
    <destination _jt="CasterScheduleDestination"
distributionFile="DistFile" distributionListFullPath="DistPath"
    singleAddress="SinglePrinter" type="Type">
      <dynamicAddress _jt="CasterScheduleDynamicAddress"
password="Password" procedureName="ProcedureName"
      serverName="ServerName" userName="UserName" />
    </destination>
  </item>
</distributionList>
```

where:

DistributionName

Is a name that is assigned to the distribution (for example, Printer).

DistFile

Is a list of one or many printers stored within a physical file accessible to the Distribution Server.

Type must be set to DISTRIBUTION_FILE.

DistPath

Is the full path to a ReportCaster Address Book which lists one or many printers. For example, IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_List.adr.

Type must be set to DISTRIBUTION_LIST.

SinglePrinter

Is the single printer to print the distributed report. Type must be set to SINGLE_ADDRESS.

Type

The following is a list of valid values:

- DISTRIBUTION_FILE
- DISTRIBUTION_LIST
- DYNAMIC_ADDRESS
- SINGLE_ADDRESS

Password

Is the value of the password required for authentication to the Reporting Server containing the WebFOCUS procedure that creates the dynamic distribution list.

Type must be set to DYNAMIC_ADDRESS.

ProcedureName

Is the name of the WebFOCUS procedure that produces the dynamic distribution list qualified by the application name. For example, ibisamp/getEmails.

Type must be set to DYNAMIC_ADDRESS.

ServerName

Is the name of the Reporting Server that contains the WebFOCUS procedure that creates the dynamic distribution list.

Type must be set to DYNAMIC_ADDRESS.

UserName

Is the user ID to the Reporting Server that contains the WebFOCUS procedure that creates the dynamic distribution list.

Type must be set to DYNAMIC_ADDRESS.

WebFOCUS Repository

This is used when the intended distribution method for the scheduled ReportCaster job is to store the output in the WebFOCUS Repository.

Body Format:

```
<distributionList _jt="array" itemsClass="CasterScheduleDistribution"
size="1">
  <item description="DistributionName" enabled="true"
folderName="FolderName"
  index="0" />
</distributionList>
```

where:

DistributionName

Is a name that is assigned to the distribution (for example, WebFOCUS Repository).

FolderName

Is the full path to the WebFOCUS Repository folder where the report will be stored. For example, IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports.

Recurrence

There are seven recurrence types to choose from when creating a ReportCaster Schedule:

- Run Once
- Minutes
- Hourly
- Daily
- Weekly
- Monthly
- Yearly

Run Once

This is used to schedule jobs that are to run only once.

```
<timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
  <item description="" enabled="true" index="0" name="">
    <startTime _jt="calendar" time="StartTime"/>
  </item>
</timeInfoList>
```

where:

StartTime

Is designated as the first time a new Schedule is set to run.

Creating a new Schedule and altering any jobs that are to run in the future will create an entirely new start time.

The default start time is the current time.

The start time is expressed in Unix Time (number seconds that has elapsed since January 1 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The start time should also be expressed in UTC (Coordinated Universal Time).

For example, December 17, 2012, 15:00:00 UTC converts to 1355756400000.

Minutes

This is used to schedule jobs that run in intervals of minutes.

Body Format:

```
<timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
  <item description="" enabled="true" frequency="Frequency"
  friday="FridayFlag"
    index="0" monday="MondayFlag" name="" saturday="SaturdayFlag"
  sunday="SundayFlag" thursday="ThursdayFlag"
    tuesday="TuesdayFlag" wednesday="WednesdayFlag">
    <startTime _jt="calendar" time="StartTime"/>
    <endTime _jt="calendar" time="EndTime"/>
  </item>
</timeInfoList>
```

where:

Frequency

Is the frequency for a scheduled event in minutes.

For example, if an email report distribution is set to run every, five minutes, the frequency would be 5.

FridayFlag

Determines whether or not the ReportCaster job is scheduled for a Friday. If set to *true*, the job will run on a Friday.

MondayFlag

Determines whether or not the ReportCaster job is scheduled for a Monday. If set to *true*, the job will run on a Monday.

SaturdayFlag

Determines whether or not the ReportCaster job is scheduled for a Saturday. If set to *true*, the job will run on a Saturday.

SundayFlag

Determines whether or not the ReportCaster job is scheduled for a Sunday. If set to *true*, the job will run on a Sunday.

ThursdayFlag

Determines whether or not the ReportCaster job is scheduled for a Thursday. If set to *true*, the job will run on a Thursday.

TuesdayFlag

Determines whether or not the ReportCaster job is scheduled for a Tuesday. If set to *true*, the job will run on a Tuesday.

WednesdayFlag

Determines whether or not the ReportCaster job is scheduled for a Wednesday. If set to *true*, the job will run on a Wednesday.

StartTime

Is designated as the first time a new Schedule is set to run.

Creating a new Schedule and altering any jobs that are to run in the future will create an entirely new start time.

The default start time is the current time.

The start time is expressed in Unix Time (number seconds that has elapsed since January 1 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The start time should also be expressed in UTC (Coordinated Universal Time).

For example, December 17, 2012, 15:00:00 UTC converts to 1355756400000.

EndTime

Is designated as the last time a Schedule is set to run.

The end time is expressed in Unix Time (number seconds that has elapsed since January 1 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The end time should also be expressed in UTC (Coordinated Universal Time).

For example, December 31, 2013, 15:00:00 UTC converts to 1388502000000.

Hourly

This is used to schedule jobs that run in intervals of hours.

Body Format:


```
<timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
  <item description="" enabled="true" frequency="Frequency"
  friday="FridayFlag"
    index="0" monday="MondayFlag" name="" saturday="SaturdayFlag"
  sunday="SundayFlag" thursday="ThursdayFlag"
    tuesday="TuesdayFlag" wednesday="WednesdayFlag">
    <startTime _jt="calendar" time="StartTime"/>
    <endTime _jt="calendar" time="EndTime"/>
  </item>
</timeInfoList>
```

where:

Frequency

Is the frequency for a scheduled event in hours.

For example, if an email report distribution is set to run every five hours, the frequency would be 5.

FridayFlag

Determines whether or not the ReportCaster job is scheduled for a Friday. If set to *true*, the job will run on a Friday.

MondayFlag

Determines whether or not the ReportCaster job is scheduled for a Monday. If set to *true*, the job will run on a Monday.

SaturdayFlag

Determines whether or not the ReportCaster job is scheduled for a Saturday. If set to *true*, the job will run on a Saturday.

SundayFlag

Determines whether or not the ReportCaster job is scheduled for a Sunday. If set to *true*, the job will run on a Sunday.

ThursdayFlag

Determines whether or not the ReportCaster job is scheduled for a Thursday. If set to *true*, the job will run on a Thursday.

TuesdayFlag

Determines whether or not the ReportCaster job is scheduled for a Tuesday. If set to *true*, the job will run on a Tuesday.

WednesdayFlag

Determines whether or not the ReportCaster job is scheduled for a Wednesday. If set to *true*, the job will run on a Wednesday.

StartTime

Is designated as the first time a new Schedule is set to run.

Creating a new Schedule and altering any jobs that are to run in the future will create an entirely new start time.

The default start time is the current time.

The start time is expressed in Unix Time (number seconds that has elapsed since January 1, 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The start time should also be expressed in UTC (Coordinated Universal Time).

For example, December 17, 2012, 15:00:00 UTC converts to 1355756400000.

EndTime

Is designated as the last time a Schedule is set to run.

The end time is expressed in Unix Time (number seconds that has elapsed since January 1, 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The end time should also be expressed in UTC (Coordinated Universal Time).

For example, December 31, 2013, 15:00:00 UTC converts to 1388502000000.

Daily

This is used to schedule jobs that run in intervals of days.

Body Format:

```
<timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
  <item description="" enabled="true" frequency="Frequency" index="0"
name="">
    <startTime _jt="calendar" time="StartTime"/>
    <endTime _jt="calendar" time="EndTime"/>
    <secondaryRunInterval _jt="CasterScheduleTimeInterval"
duration="Duration" interval="Interval"
      isEnabled="SecondaryIntervalFlag"><untilTime _jt="calendar"
time="UntilTime"/>
    </secondaryRunInterval>
  </item>
</timeInfoList>
```

where:

Frequency

Is the frequency for a scheduled event in days. For example, if an email report distribution is set to run every five days, the frequency would be 5.

StartTime

Is the start time is designated as the first time a new Schedule is set to run.

Creating a new Schedule and altering any jobs that are to run in the future will create an entirely new start time.

The default start time is the current time.

The start time is expressed in Unix Time (number seconds that has elapsed since January 1 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The start time should also be expressed in UTC (Coordinated Universal Time).

For example, December 17, 2012, 15:00:00 UTC converts to 1355756400000.

EndTime

Is the end time is designated as the last time a Schedule is set to run.

The end time is expressed in Unix Time (number seconds that has elapsed since January 1 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The end time should also be expressed in UTC (Coordinated Universal Time).

For example, December 31, 2013, 15:00:00 UTC converts to 1388502000000.

Duration

Is the duration of *SecondaryIntervalFlag*. If *SecondaryIntervalFlag* is set to *true*, the duration specified in minutes during which the time interval will be applied. *UntilTime* must equal 18000000.

Interval

Is the time interval in minutes. If *SecondaryIntervalFlag* is set to *true*, the time interval is applied every *n* minutes.

SecondaryIntervalFlag

Is the time interval settings. If set to *true*, the time Interval settings are active. If set to *false*, the time Interval settings are inactive.

UntilTime

Is the end time of the time interval. If *SecondaryIntervalFlag* set to *true*, the end time for which the time interval will be applied. Duration must be set to -1.

The end time of the interval is expressed in Unix Time (number seconds that has elapsed since January 1 1970, 00:00:00).

It does not matter which date is used as only the time portion will be used in setting the end time for the time interval.

Typically, the date the Schedule is created or updated is used in setting *UntilTime*.

Three zeros are appended to the Unix Time to represent milliseconds.

The end time of the interval should also be expressed in UTC (Coordinated Universal Time).

For example, October 19, 2012, 20:38:00 UTC converts to 1350679080000. October 19, 2012 will be ignored when setting the end time of the interval.

Weekly

This is used to schedule jobs that run in intervals of weeks.

Body Format:

```
<timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
  <item _jt="CasterScheduleTimeInfoWeek" description="" enabled="true"
frequency="Frequency" friday="FridayFlag" index="0"
      monday="MondayFlag" name="" saturday="SaturdayFlag"
sunday="SundayFlag" thursday="ThursdayFlag"
      tuesday="TuesdayFlag" wednesday="WednesdayFlag">
    <startTime _jt="calendar" time="StartTime"/>
    <endTime _jt="calendar" time="EndTime"/>
    <secondaryRunInterval _jt="CasterScheduleTimeInterval"
duration="Duration" interval="Interval"
      isEnabled="SecondaryIntervalFlag"><untilTime _jt="calendar"
time="UntilTime"/>
    </secondaryRunInterval>
  </item>
</timeInfoList>
```

Frequency

Is the frequency for a scheduled event in weeks.

For example, if an email report distribution is set to run every five weeks, the frequency would be 5.

FridayFlag

Determines whether or not the ReportCaster job is scheduled for a Friday. If set to *true*, the job will run on a Friday.

MondayFlag

Determines whether or not the ReportCaster job is scheduled for a Monday. If set to *true*, the job will run on a Monday.

SaturdayFlag

Determines whether or not the ReportCaster job is scheduled for a Saturday. If set to *true*, the job will run on a Saturday.

SundayFlag

Determines whether or not the ReportCaster job is scheduled for a Sunday. If set to *true*, the job will run on a Sunday.

ThursdayFlag

Determines whether or not the ReportCaster job is scheduled for a Thursday. If set to *true*, the job will run on a Thursday.

TuesdayFlag

Determines whether or not the ReportCaster job is scheduled for a Tuesday. If set to *true*, the job will run on a Tuesday.

WednesdayFlag

Determines whether or not the ReportCaster job is scheduled for a Wednesday. If set to *true*, the job will run on a Wednesday.

StartTime

Is designated as the first time a new Schedule is set to run.

Creating a new Schedule and altering any jobs that are to run in the future will create an entirely new start time.

The default start time is the current time.

The start time is expressed in Unix Time (number seconds that has elapsed since January 1, 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The start time should also be expressed in UTC (Coordinated Universal Time).

For example, December 17, 2012, 15:00:00 UTC converts to 1355756400000.

EndTime

Is designated as the last time a Schedule is set to run.

The end time is expressed in Unix Time (number seconds that has elapsed since January 1, 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The end time should also be expressed in UTC (Coordinated Universal Time).

For example, December 31, 2013, 15:00:00 UTC converts to 1388502000000.

Duration

Is the duration of *SecondaryIntervalFlag*. If *SecondaryIntervalFlag* is set to *true*, the duration specified in minutes during which the time interval will be applied. *UntilTime* must be set to 18000000.

Interval

Is the time interval in minutes. If *SecondaryIntervalFlag* is set to *true*, the time interval is applied every *n* minutes.

SecondaryIntervalFlag

Are the time interval settings. If true, the time Interval settings are active. If set to *false*, the time Interval settings are inactive.

UntilTime

Is the end time of the time interval. If *SecondaryIntervalFlag* is set to *true*, the end time for which the time interval will be applied. Duration must set to -1.

The end time of the interval is expressed in Unix Time (number seconds that has elapsed since January 1 1970, 00:00:00).

It does not matter which date is used as only the time portion will be used in setting the end time for the time interval.

Typically, the date the Schedule is created or updated is used in setting *UntilTime*.

Three zeros are appended to the Unix Time to represent milliseconds.

The end time of the interval should also be expressed in UTC (Coordinated Universal Time).

For example, October 19, 2012, 20:38:00 UTC converts to 1350679080000. October 19, 2012 will be ignored when setting the end time of the interval.

Monthly

This is used to schedule jobs that run in intervals of months.

Body Format:

```
<timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
  <item dayOfWeek="DayOfWeek" dayOfWeekEnabled="DayOfWeekEnabled" description=""
    enabled="true" frequency="Frequency" index="0" lastDayOfMonth="LastDayOfMonth"
    name="" type="5" weekOfMonth="WeekOfMonth">
    <startTime _jt="calendar" time="StartTime"/>
    <endTime _jt="calendar" time="EndTime"/>
    <daysOfMonth _jt="array" size="31">
      <item _jt="boolval" index="0" value="false"/>
      <item _jt="boolval" index="1" value="false"/>
      <item _jt="boolval" index="2" value="false"/>
      <item _jt="boolval" index="3" value="false"/>
      <item _jt="boolval" index="4" value="false"/>
      <item _jt="boolval" index="5" value="false"/>
      <item _jt="boolval" index="6" value="false"/>
      <item _jt="boolval" index="7" value="false"/>
      <item _jt="boolval" index="8" value="false"/>
      <item _jt="boolval" index="9" value="false"/>
      <item _jt="boolval" index="10" value="false"/>
      <item _jt="boolval" index="11" value="false"/>
      <item _jt="boolval" index="12" value="false"/>
      <item _jt="boolval" index="13" value="false"/>
      <item _jt="boolval" index="14" value="false"/>
      <item _jt="boolval" index="15" value="false"/>
      <item _jt="boolval" index="16" value="false"/>

      <item _jt="boolval" index="17" value="false"/>
      <item _jt="boolval" index="18" value="false"/>
      <item _jt="boolval" index="19" value="false"/>
      <item _jt="boolval" index="20" value="false"/>
      <item _jt="boolval" index="21" value="false"/>
      <item _jt="boolval" index="22" value="false"/>
      <item _jt="boolval" index="23" value="false"/>
      <item _jt="boolval" index="24" value="false"/>
      <item _jt="boolval" index="25" value="false"/>
      <item _jt="boolval" index="26" value="false"/>
      <item _jt="boolval" index="27" value="false"/>
      <item _jt="boolval" index="28" value="false"/>
      <item _jt="boolval" index="29" value="false"/>
      <item _jt="boolval" index="30" value="false"/>
    </daysOfMonth>
    <secondaryRunInterval _jt="CasterScheduleTimeInterval" duration="Duration"
    interval="Interval" isEnabled="SecondaryIntervalFlag">
      <untilTime _jt="calendar" time="UntilTime"/>
    </secondaryRunInterval>
  </item>
</timeInfoList>
```

where:

DayOfWeek

Is the day of the week for the report to run. *DayOfWeekEnabled* must be set to *true*. The following list shows the valid values:

- 1.** Sunday
- 2.** Monday
- 3.** Tuesday
- 4.** Wednesday
- 5.** Thursday
- 6.** Friday
- 7.** Saturday

DayOfWeekEnabled

Is the day of the week or day of the month to be set. If set to *true*, the *DayOfWeek* and *WeekOfMonth* must be set.

If set to *false*, the *DaysOfMonth* and/or *LastDayOfMonth* must be set.

Frequency

Is the frequency for a scheduled event in months. For example, if an email report distribution is set to run every 2 months, the frequency would be 2.

LastDayOfMonth

Is an indicator whether or not the last day of the month flag is set. When this flag is set to *true*, ReportCaster runs a Schedule on the last day of the month regardless of what day it is.

For example, a Schedule set to run on February 28th will next run on March 31st if this flag is set to *true*. Otherwise, ReportCaster will run the job on the corresponding day of the next month.

If this flag is set to *false* then it will run the report on March 28th. If the corresponding day of the next month does not exist, then ReportCaster will not run the report.

WeekOfMonth

Week of the month for the report to run. *DayOfWeekEnabled* must be set to *true*. The following list shows the valid values.

- 1.** First week
- 2.** Second week
- 3.** Third week
- 4.** Fourth week
- 5.** Last week

StartTime

Is designated as the first time a new Schedule is set to run.

Creating a new Schedule and altering any jobs that are to run in the future will create an entirely new start time.

The default start time is the current time.

The start time is expressed in Unix Time (number seconds that has elapsed since January 1, 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The start time should also be expressed in UTC (Coordinated Universal Time).

For example, December 17, 2012, 15:00:00 UTC converts to 1355756400000.

EndTime

Is designated as the last time a Schedule is set to run.

The end time is expressed in Unix Time (number seconds that has elapsed since January 1, 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The end time should also be expressed in UTC (Coordinated Universal Time).

For example, December 31, 2013, 15:00:00 UTC converts to 1388502000000.

daysOfmonth

Is a 31 element array indicating which days of the month have been selected for a report to run. All array members are initialized to *false*.

Each array member has an index attribute associated with it.

The index starts at 0 and increments by 1 for each successive day of the month. For example, index=0 equates to the first day of the month.

Those members of the array that are then set to *true* are the days of the month the Schedule will run. *DayOfWeekEnabled* must be set to *false*.

Duration

Is the duration of *SecondaryIntervalFlag*. If *SecondaryIntervalFlag* is set to *true*, the duration specified in minutes during which the time interval will be applied. *UntilTime* must be set to 18000000.

Interval

Is the time interval in minutes. If *SecondaryIntervalFlag* is set to *true*, the time interval is applied every *n* minutes.

SecondaryIntervalFlag

Are the time interval settings. If set to *true*, the time Interval settings are active. If set to *false*, the time Interval settings are inactive.

UntilTime

Is the end time of the time interval. If *SecondaryIntervalFlag* is set to *true*, the end time for which the time interval will be applied. Duration must be set to -1.

The end time of the interval is expressed in Unix Time (number seconds that has elapsed since January 1 1970, 00:00:00).

It does not matter which date is used as only the time portion will be used in setting the end time for the time interval.

Typically, the date the Schedule is created or updated is used in setting *UntilTime*.

Three zeros are appended to the Unix Time to represent milliseconds.

The end time of the interval should also be expressed in UTC (Coordinated Universal Time).

For example, October 19, 2012, 20:38:00 UTC converts to 1350679080000. October 19, 2012 will be ignored when setting the end time of the interval.

Yearly

This is used to schedule jobs that run in intervals of years.

Body Format:

```

<timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
  <item description="" enabled="true" frequency="Frequency" index="0"
name="">
    <startTime _jt="calendar" time="StartTime"/>
    <endTime _jt="calendar" time="EndTime"/>
    <secondaryRunInterval _jt="CasterScheduleTimeInterval "
duration="Duration" interval="Interval"
      isEnabled="SecondaryIntervalFlag">
      <untilTime _jt="calendar" time="UntilTime"/>
    </secondaryRunInterval>
  </item>
</timeInfoList>

```

where:

Frequency

Is the frequency for a scheduled event in years.

For example, if an email report distribution is set to run every year, the frequency would be 1.

StartTime

Is designated as the first time a new Schedule is set to run.

Creating a new Schedule and altering any jobs that are to run in the future will create an entirely new start time.

The default start time is the current time.

The start time is expressed in Unix Time (number seconds that has elapsed since January 1, 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The start time should also be expressed in UTC (Coordinated Universal Time).

For example, December 17, 2012, 15:00:00 UTC converts to 1355756400000.

EndTime

Is designated as the last time a Schedule is set to run.

The end time is expressed in Unix Time (number seconds that has elapsed since January 1, 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The end time should also be expressed in UTC (Coordinated Universal Time).

For example, December 31, 2013, 15:00:00 UTC converts to 1388502000000.

Duration

Is the duration of *SecondaryIntervalFlag*. If *SecondaryIntervalFlag* is set to *true*, the duration specified in minutes during which the time interval will be applied. *UntilTime* must be set to 18000000.

Interval

Is the time interval in minutes. If *SecondaryIntervalFlag* is set to *true*, the time interval is applied every *n* minutes.

SecondaryIntervalFlag

Are the time interval settings. If set to *true*, the time Interval settings are active. If set to *false*, the time Interval settings are inactive.

UntilTime

Is the end time of the time interval. If *SecondaryIntervalFlag* is set to *true*, the end time for which the time interval will be applied. Duration must be set to -1.

The end time of the interval is expressed in Unix Time (number seconds that has elapsed since January 1 1970, 00:00:00).

It does not matter which date is used as only the time portion will be used in setting the end time for the time interval.

Typically, the date the Schedule is created or updated is used in setting *UntilTime*.

Three zeros are appended to the Unix Time to represent milliseconds.

The end time of the interval should also be expressed in UTC (Coordinated Universal Time).

For example, October 19, 2012, 20:38:00 UTC converts to 1350679080000. October 19, 2012 will be ignored when setting the end time of the interval.

Custom

This is used to schedule jobs that run on specific dates.

Body Format:

```

<timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
  <item description="" enabled="true" index="0" name="" type="7">
    <startTime _jt="calendar" time="StartTime"/>
    <endTime _jt="calendar" time="EndTime"/>
    <dateList _jt="array" itemsClass="java.util.Calendar"
size="numberOfItems">
      <item _jt="calendar" index="0" time="date1"/>
      <item _jt="calendar" index="1" time="date2"/>
    </dateList>
    <secondaryRunInterval _jt="CasterScheduleTimeInterval"
duration="Duration" interval="Interval"
      isEnabled="SecondaryIntervalFlag"><untilTime _jt="calendar"
time="UntilTime"/>
    </secondaryRunInterval>
  </item>
</timeInfoList>

```

where:

StartTime

Is designated as the first time a new Schedule is set to run.

Creating a new Schedule and altering any jobs that are to run in the future will create an entirely new start time.

The default start time is the current time.

The start time is expressed in Unix Time (number seconds that has elapsed since January 1, 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The start time should also be expressed in UTC (Coordinated Universal Time).

For example, December 17, 2012, 15:00:00 UTC converts to 1355756400000.

EndTime

Is designated as the last time a Schedule is set to run.

The end time is expressed in Unix Time (number seconds that has elapsed since January 1, 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The end time should also be expressed in UTC (Coordinated Universal Time).

For example, December 31, 2013, 15:00:00 UTC converts to 1388502000000.

dateList

Is an item array indicating which days have been selected for a report to run.

Each item has an index attribute associated with it.

The index starts at 0 and increments by 1 for each date.

Each item has date associated with it. (for example, date1, date2).

The date is expressed in Unix Time (number seconds that has elapsed since January 1, 1970, 00:00:00).

Three zeros are appended to the Unix Time to represent milliseconds.

The date should also be expressed in UTC (Coordinated Universal Time).

For example, December 17, 2012, 15:00:00 UTC converts to 1355756400000.

numberOfItems

Is the number of dates that will be defined for the schedule to run.

Duration

Is the duration of *SecondaryIntervalFlag*. If *SecondaryIntervalFlag* is set to *true*, the duration specified in minutes during which the time interval will be applied. *UntilTime* must be set to 18000000.

Interval

Is the time interval in minutes. If *SecondaryIntervalFlag* is set to *true*, the time interval is applied every *n* minutes.

SecondaryIntervalFlag

Are the time interval settings. If set to *true*, the time Interval settings are active. If set to *false*, the time Interval settings are inactive.

UntilTime

Is the end time of the time interval. If *SecondaryIntervalFlag* is set to *true*, the end time for which the time interval will be applied. Duration must be set to -1.

The end time of the interval is expressed in Unix Time (number seconds that has elapsed since January 1 1970, 00:00:00).

It does not matter which date is used as only the time portion will be used in setting the end time for the time interval.

Typically, the date the Schedule is created or updated is used in setting *UntilTime*.

Three zeros are appended to the Unix Time to represent milliseconds.

The end time of the interval should also be expressed in UTC (Coordinated Universal Time).

For example, October 19, 2012, 20:38:00 UTC converts to 1350679080000. October 19, 2012 will be ignored when setting the end time of the interval.

Task

There are five task types to choose from when creating a ReportCaster Schedule:

- WebFOCUS Report
- WebFOCUS Server Procedure
- File
- FTP
- URL

WebFOCUS Report

WebFOCUS Report enables you to schedule the distribution of reports that reside specifically within the WebFOCUS Repository. You can associate an alert with the report which allows you to schedule actions that are contingent upon specific alert conditions being triggered. If the report to be run is not an alert, the alert tag in the XML is not required.

Body Format:

```
<taskList _jt="array" itemsClass="CasterScheduleTask" size="1">
  <item alertEnabled="AlertFlag" burst="BurstFlag"
    description="TaskDescription" domainHREF="" enabled="true"
    execId="ServerUserid"
    execPassword="ServerPassword"
    firstPostProcessingProcedure="FirstPostProcedure"
    firstPreProcessingProcedure="FirstPreProcedure" index="0"
    procedureDescription="" procedureName="ProcedureName"
    reportName="ReportName"

    secondPostProcessingProcedure="SecondPostProcedure"
    secondPreProcessingProcedure="SecondPreProcedure"
    sendFormat="SendFormat" serverName="ServerName">
    <parameterList _jt="array" itemsClass="CasterScheduleParameter"
    size="NumberOfParameters">
      <item _jt="CasterScheduleParameter" enabled="true" index="IndexValue"
    name="ParameterName"
      value="ParameterValue"/>
    </parameterList>
    <alert resetInterval="ResetInterval" resetType="ResetType"/>
  </item>
</taskList>
```

where:

AlertFlag

Is the value that determines whether or not an alert is enabled.

If set to *true*, the alert is enabled. If set to *false*, it is disabled.

BurstFlag

Is the value that specifies whether or not report bursting is enabled. Report bursting allows you to segment a report into sections based upon a primary sort field. The report segments are then distributed as separate reports by the Distribution Server. Access to these report segments is based upon burst values (specific values of the primary sort field) that are associated with email addresses in distribution lists or user IDs in Library Access Lists.

TaskDescription

Is the text used to describe the task. The maximum size for the description is 255 characters.

ServerUserid

Is the user name needed to establish a connection to the WebFOCUS Reporting Server. The user name is one of the credentials necessary for a user to access a WebFOCUS procedure that resides on the WebFOCUS Reporting Server during scheduling, as well as to run this procedure at the time the job is run.

This setting must have a value even if using an unsecured Reporting Server.

ServerPassword

Is the password needed to establish a connection to the WebFOCUS Reporting Server. The password is one of the credentials necessary for a user to access a WebFOCUS procedure that resides on the WebFOCUS Reporting Server during scheduling, as well as run this procedure at the time the job is run. This setting must have a value even if using an unsecured Reporting Server.

FirstPostProcedure

Is the name of the first of two possible post-processing procedures. Post-processing procedures (available for WebFOCUS Server procedure and WebFOCUS Report) are non-reporting WebFOCUS procedures that run synchronously after the execution of their associated task. They are often used to reset computing or data environments.

FirstPreProcedure

Is the name of the first of two possible preprocessing procedures. Preprocessing procedures (available for WebFOCUS Server procedure and WebFOCUS Report) are non-reporting WebFOCUS procedures that run synchronously prior the execution of their associated task. They are often used to set or test conditions before the running of reports.

ProcedureName

Is the full path to the WebFOCUS Report that is to be run. For example, IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/Sales_Report_by_Country.fex.

ReportName

The name of the file when sending the output as an attachment. The maximum size for report name is 64 characters.

SecondPostProcedure

Is the name of the second of two possible post-processing procedures. Post-processing procedures (available for WebFOCUS Server procedure and WebFOCUS Report) are non-reporting WebFOCUS procedures that run synchronously after the execution of their associated task. They are often used to reset computing or data environments.

SecondPreProcedure

Is the name of the second of two possible preprocessing procedures. Preprocessing procedures (available for WebFOCUS Server procedure and WebFOCUS Report) are non-reporting WebFOCUS procedures that run synchronously prior the execution of their associated task. They are often used to set or test conditions before the running of reports.

SendFormat

Is the report format that will be generated by the WebFOCUS Reporting Server. For example, PDF, HTML, AHTML, EXL07, DFIX DELIMITER, and COM.

ServerName

Is the name of the WebFOCUS Reporting Server used to run the WebFOCUS reports and procedures associated with this task.

NumberOfParameters

Is the number of parameters that are to be passed to the WebFOCUS Report.

IndexValue

Is a value that starts at 0 and increments by 1 for every parameter that is to be sent to the WebFOCUS Report.

ParameterName

Is the name of a parameter passed to the WebFOCUS report. The maximum length of the name field is 64 characters.

ParameterValue

Is the value of a parameter associated with the *ParameterName* passed to the WebFOCUS report. The maximum length of the value field is 255 characters.

ResetInterval

If *AlertFlag* is set to *true*, *ResetInterval* represents the time interval (delay) between when an alert Schedule is run upon being triggered and when it is reactivated. The actual time period is based on the *ResetType*.

For example, if the *ResetType* is HOUR, a specified reset interval of three would represent a three hour delay.

ResetType

The following list shows the valid values for *ResetType* if *AlertFlag* is set to *true*.

- MINUTE**
- HOUR**
- DAY**
- WEEK**
- MONTH**
- YEAR**
- CONTINUE.** Reactivate the alert immediately.
- AUTO.** Reactivate the alert when the condition is no longer true.
- TERMINATE.** Deactivate the Schedule.

WebFOCUS Server Procedure

The WebFOCUS Server Procedure allows you to schedule the distribution of reports that reside specifically on a WebFOCUS Reporting Server. A WebFOCUS Server procedure is a WebFOCUS report (FOCEXEC) residing on a WebFOCUS Reporting Server that is accessible to the Distribution Server.

Body Format:

```

<taskList _jt="array" itemsClass="CasterScheduleTask" size="1">
  <item _jt="CasterScheduleWFServerProcedure" burst="BurstFlag"
description="TaskDescription" enabled="true"
  execId="ServerUserid" execPassword="ServerPassword"
firstPostProcessingProcedure="FirstPostProcedure"
  firstPreProcessingProcedure="FirstPreProcedure" index="0"
procedureName="ProcedureName" reportName="ReportName"
  secondPostProcessingProcedure="SecondPostProcedure"
secondPreProcessingProcedure="SecondPreProcedure"
  sendFormat="SendFormat" serverName="ServerName">
  <parameterList _jt="array" itemsClass="CasterScheduleParameter"
size="NumberOfParameters">
    <item _jt="CasterScheduleParameter" enabled="true"

index="IndexValue" name="ParameterName" value="ParameterValue" />
  </parameterList>
  </item>
</taskList>

```

where:

BurstFlag

Is the value that specifies whether or not report bursting is enabled. Report bursting allows you to segment a report into sections based upon a primary sort field. The report segments are then distributed as separate reports by the Distribution Server. Access to these report segments is based upon burst values (specific values of the primary sort field) that are associated with email addresses in distribution lists or user IDs in Library Access Lists.

TaskDescription

Is the text used to describe the task. The maximum size for the description is 255 characters.

ServerUserid

Is the user name needed to establish a connection to the WebFOCUS Reporting Server. The user name is one of the credentials necessary for a user to access a WebFOCUS procedure that resides on the WebFOCUS Reporting Server during scheduling, as well as to run this procedure at the time the job is run. This setting must have a value even if using an unsecured Reporting Server.

ServerPassword

Is the password needed to establish a connection to the WebFOCUS Reporting Server. The password is one of the credentials necessary for a user to access a WebFOCUS procedure that resides on the WebFOCUS Reporting Server during scheduling, as well as run this procedure at the time the job is run.

FirstPostProcedure

Is the name of the first of two possible post-processing procedures. Post-processing procedures (available for WebFOCUS Server procedure and WebFOCUS Report) are non-reporting WebFOCUS procedures that run synchronously after the execution of their associated task. They are often used to reset computing or data environments.

FirstPreProcedure

Is the name of the first of two possible preprocessing procedures. Preprocessing procedures (available for WebFOCUS Server procedure and WebFOCUS Report) are non-reporting WebFOCUS procedures that run synchronously prior the execution of their associated task. They are often used to set or test conditions before the running of reports.

ProcedureName

Is the full path to the WebFOCUS Server Procedure Report that is to be run. For example, ibisamp/carinst.

ReportName

Is the name of the file when sending the output as an attachment. The maximum size for report name is 64 characters.

SecondPostProcedure

Is the name of the second of two possible post-processing procedures. Post-processing procedures (available for WebFOCUS Server procedure and WebFOCUS Report) are non-reporting WebFOCUS procedures that run synchronously after the execution of their associated task. They are often used to reset computing or data environments.

SecondPreProcedure

Is the name of the second of two possible preprocessing procedures. Preprocessing procedures (available for WebFOCUS Server procedure and WebFOCUS Report) are non-reporting WebFOCUS procedures that run synchronously prior the execution of their associated task. They are often used to set or test conditions before the running of reports.

SendFormat

Is the report format that will be generated by the WebFOCUS Reporting Server. For example, PDF, HTML, AHTML, EXL07, DFIX DELIMITER, and COM.

ServerName

Is the name of the WebFOCUS Reporting Server used to run the WebFOCUS procedures associated with this task.

NumberOfParameters

Is the number of parameters that are to be passed to the WebFOCUS Report.

IndexValue

Is a value that starts at 0 and increments by 1 for every parameter that is to be sent to the WebFOCUS Report.

ParameterName

Is the name of a parameter passed to the WebFOCUS report. The maximum length of the name field is 64 characters.

ParameterValue

Is the value of a parameter associated with the ParameterName passed to the WebFOCUS report. The maximum length of the value field is 255 characters.

File

This allows you to schedule the distribution of a file, represented by a fully qualified path, to which the ReportCaster Distribution Server has read access.

Body Format:

```
<taskList _jt="array" itemsClass="CasterScheduleTask" size="1">
  <item deleteFileAfterRetrieval="DeleteFile" description="TaskDescription"
    enabled="true" index="0" procedureName="FileLocation"
    reportName="ReportName" />
</taskList>
```

where:

DeleteFile

Is used to either delete or not delete the file.

If set to *true*, the file is deleted, as identified by *ProcedureName*, after it is distributed.

If set to *false*, the file is not deleted, as identified by *ProcedureName*, after it is distributed.

TaskDescription

Is the text used to describe the task. The maximum size for the description is 255 characters.

FileLocation

Is the full path to the file being distributed. For example, C:\Documentation\HTML\REST_Documentation_version_2.html.

ReportName

Is the name of the file when sending the output as an attachment.

FTP

FTP allows you to schedule the distribution of a file that resides on any FTP Server.

Body Format:

```
<taskList _jt="array" itemsClass="CasterScheduleTask" size="1">
  <item deleteAfterRetrieval="DeleteFile" description="TaskDescription"
    enabled="true" index="0" password="FTPpass"
passwordAuthEnabled="passwordFlag" procedureName="FileLocation"
    publicKeyAuthEnabled="publicFlag" reportName="ReportName"
sendFormat="SendFormat"
    serverName="FTPserver" sftpEnabled="SFTPflag" userName="FTPuser" />
</taskList>
```

where:

DeleteFile

Is used to either delete or not delete the file.

If set to *true*, the file is deleted, as identified by *ProcedureName*, after it is distributed.

If set to *false*, the file is not deleted, as identified by *ProcedureName*, after it is distributed.

TaskDescription

Is the text used to describe the task. The maximum size for the description is 255 characters.

FTPpass

Is the password needed to authenticate to the FTP server. The FTP password is part of the credentials necessary for the Distribution Server to access the FTP server.

passwordFlag

Is the password authentication.

If *SFTPflag* is set to *true*, and *passwordFlag* is set to *true*, then the password authentication is enabled.

If *SFTPflag* is set to *true*, and *passwordFlag* is set to *false*, then the password authentication is disabled.

FileLocation

Is the full path to the file being distributed. For example, outgoing\HTML\REST_Documentation_version_2.html.

publicFlag

Is the Public Key authentication.

If *SFTPflag* equals *true*, and *passwordFlag* is set to *true*, then the Public Key authentication is enabled.

If *SFTPflag* equals *true*, and *passwordFlag* is set to *false*, then the Public Key authentication is disabled.

ReportName

Is the name of the file when sending the output as an attachment.

SendFormat

Is the report format that will be generated by the WebFOCUS Reporting Server. For example, PDF, HTML, AHTML, EXL07, DFIX DELIMITER, and COM.

FTPserver

Is the name of the FTP server where the file being distributed exists.

SFTPflag

Is the secure SSH File Transfer Protocol (SFTP).

If set to *true*, the FTP server requires a secure SSH File Transfer Protocol (SFTP).

If set to *false*, the FTP server does not require a secure SSH File Transfer Protocol (SFTP).

FTPuser

The user name needed to authenticate to the FTP server. The FTP password is part of the credentials necessary for the Distribution Server to access the FTP server.

URL

This allows ReportCaster to connect to a specified URL at execution time, retrieve the pages returned by that URL and distributes them. This task can be used to call any type of URL, including programs that are executed by JSP and ASP pages, as well as reports generated by other reporting products.

Body Format:

```
<taskList _jt="array" itemsClass="CasterScheduleTask" size="1">
  <item description="TaskDescription" enabled="true" index="0"
password="Password" reportName="ReportName" urlString="URLstring"
userName="Username"> <parameterList _jt="array"
itemsClass="CasterScheduleParameter" size="1">
  <item _jt="CasterScheduleParameter" enabled="true" index="IndexValue"
name="ParameterName" value="ParameterValue" />
  </parameterList>
</item>
</taskList>
```

where:

TaskDescription

Is the text used to describe the task. The maximum size for the description is 255 characters.

Password

Is the value of the password necessary for access to the web server of the URL. This password is submitted within the HTTP header.

ReportName

Is the name of the file when sending the output as an attachment.

URLstring

Is the URL of the webpage to be distributed.

Username

Is the value of the user name necessary for access to the web server of the URL. This user name is submitted within the HTTP header.

IndexValue

Is a value that starts at 0 and increments by 1 for every parameter that is to be sent to the webpage.

ParameterName

Is the name of a parameter passed to the webpage. The maximum length of the name field is 64 characters.

ParameterValue

Is the value of a parameter associated with the ParameterName passed to the webpage. The maximum length of the value field is 255 characters.

Closing Tag

The following closing tag must be used:


```
</casterObject></rootObject>
```

Example 1: Creating a Schedule

This example creates a Schedule called REST_Schedule that runs the Sales_Report_by_Country WebFOCUS report once on December 17th, 2012 at 15:00:00 UTC and distributes the output to the Report Library. The report will run with the COUNTRY parameter set to ENGLAND and the DEALER_COST parameter set to 10000.

POST Request URL Format:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_Schedule.sch
```

Body Format:

```
IBIRS_action=put&IBIRS_replace=false&IBIRS_object=
<rootObject _jt="IBFSCasterObject" description="Schedule Created through REST"
type="CasterSchedule">
  <casterObject _jt="CasterSchedule" active="true" deleteJobAfterRun="false"
description="Schedule Created through REST" owner="admin" priority="3" traceType="0">

    <notification _jt="CasterScheduleNotification" addressForBriefNotification=""
addressForFullNotification="" description="" from="" subject="" type="INACTIVE"/>
    <distributionList _jt="array" itemsClass="CasterScheduleDistribution" size="1">
      <item accessListFullPath="" accessType="OWNER" category=""
compressionEnabled="false"
        description="Report Library" destinationPath="IBFS:/WFC/Repository/
RESTful_Web_Services/Car_Reports" enabled="true" expirationData="1"
        expirationMode="N" index="0" valueonly="false">
        <storageLibraryEmail authEnabled="false" authPassword="" authUserId=""
        libraryURL="http://localhost:8080/ibi_apps/library/report.rc" mailFrom=""
mailMessage="" mailReplyAddress="" mailServerName="ibismtp.ibi.com"
        mailSubject="" sendEmailAfterSaveReport="false" sslEnabled="false"
tlsEnabled="false"/>
      </item>
    </distributionList>
    <timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
      <item description="" enabled="true" index="0" name="">
        <startTime _jt="calendar" time="1355756400000"/>
      </item>
    </timeInfoList>
    <taskList _jt="array" itemsClass="CasterScheduleTask" size="1">
      <item alertEnabled="false" burst="true" description="Task 1" domainHREF=""
enabled="true"
        execId="guest" execPassword="guest" firstPostProcessingProcedure=""
firstPreProcessingProcedure="" index="0" procedureDescription=""
procedureName="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/
Sales_Report_by_Country.fex" reportName="car_sales.htm"
        secondPostProcessingProcedure="" secondPreProcessingProcedure=""
sendFormat="HTML" serverName="EDASERVE">
        <parameterList _jt="array" itemsClass="CasterScheduleParameter" size="2">
```

```
        <item _jt="CasterScheduleParameter" enabled="true" index="0" name="COUNTRY"
value="ENGLAND" />
        <item _jt="CasterScheduleParameter" enabled="true" index="1"
name="DEALER_COST" value="10000" />
    </parameterList>
</item>
</taskList>
</casterObject>
</rootObject>
```

Response:

```

<?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="put" returncode="10000"
returndesc="SUCCESS" subreturncode="0"
  subsystem="SSYS" type="simple">
  <ibfsparams size="5">
    <entry key="IBIRS_replace" value="false"/>
    <entry key="IBIRS_private" value="__null"/>
    <entry key="IBIRS_object" value="&lt;rootObject _jt=&quot;IBFSCasterObject&quot;
description=&quot;Schedule Created through REST&quot;
type=&quot;CasterSchedule&quot;;&gt;
      &lt;casterObject _jt=&quot;CasterSchedule&quot; active=&quot;true&quot;
deleteJobAfterRun=&quot;false&quot;
description=&quot;Schedule Created through REST&quot;
owner=&quot;admin&quot; priority=&quot;3&quot;
traceType=&quot;0&quot;;&gt; &lt;notification
_jt=&quot;CasterScheduleNotification&quot; addressForBriefNotification=&quot;&quot;
addressForFullNotification=&quot;&quot; description=&quot;&quot;
from=&quot;&quot; subject=&quot;&quot;
type=&quot;INACTIVE&quot;;&gt; &lt;distributionList _jt=&quot;array&quot;
itemsClass=&quot;CasterScheduleDistribution&quot;
size=&quot;1&quot;;&gt; &lt;item accessListFullPath=&quot;&quot;
accessType=&quot;OWNER&quot; category=&quot;&quot;
class=&quot;ibi.broker.api.data.schedule.StorageLibrary&quot;
compressionEnabled=&quot;false&quot;
description=&quot;Report
Library&quot;destinationPath=&quot;IBFS:/WFC/Repository/RESTful_Web_Services/
Car_Reports&quot;
enabled=&quot;true&quot; expirationData=&quot;1&quot;
expirationMode=&quot;N&quot; index=&quot;0&quot;
valueonly=&quot;false&quot;;&gt; &lt;storageLibraryEmail
authEnabled=&quot;false&quot; authPassword=&quot;&quot;
authUserId=&quot;&quot;
class=&quot;ibi.broker.api.data.schedule.StorageLibraryEmail&quot;
libraryURL=&quot;http://localhost:8080/ibi_apps/library/report.rc&quot;
mailFrom=&quot;&quot; mailMessage=&quot;&quot;
mailReplyAddress=&quot;&quot; mailServerName=&quot;ibismtp.ibi.com&quot;
mailSubject=&quot;&quot;
sendEmailAfterSaveReport=&quot;false&quot; sslEnabled=&quot;false&quot;
tlsEnabled=&quot;false&quot;;&gt; &lt;/item&gt; &lt;/distributionList&gt;
&lt;timeInfoList _jt=&quot;array&quot;
itemsClass=&quot;CasterScheduleTimeInfo&quot; size=&quot;
1&quot;;&gt; &lt;item class=&quot;ibi.broker.api.data.schedule.TimeInfoOnce&quot;
description=&quot;&quot; enabled=&quot;true&quot; index=&quot;0&quot;
name=&quot;&quot;;&gt; &lt;startTime _jt=&quot;calendar&quot;

```

```

time=&quot;1355756400000&quot; /&gt;&lt;/item&gt;&lt;/timeInfoList&gt; &lt;/taskList
_jt=&quot;array&quot;
  itemsClass=&quot;CasterScheduleTask&quot; size=&quot;1&quot;&gt;&lt;/item
alertEnabled=&quot;false&quot; burst=&quot;true&quot;
  class=&quot;ibi.broker.api.data.schedule.TaskStandardReport&quot;
description=&quot;Task 1&quot; domainHref=&quot;&quot;
  enabled=&quot;true&quot; execId=&quot;guest&quot;
  execPassword=&quot;guest&quot; firstPostProcessingProcedure=&quot;&quot;
firstPreProcessingProcedure=&quot;&quot;
  index=&quot;0&quot; procedureDescription=&quot;&quot;
  procedureName=&quot;IBFS:/WFC/Repository/RESTful_Web_Services/
Car_Reports/Sales_Report_by_Country.fex&quot;
  reportName=&quot;car_sales.htm&quot;
secondPostProcessingProcedure=&quot;&quot; secondPreProcessingProcedure=&quot;&quot;
  sendFormat=&quot;HTML&quot;
serverName=&quot;EDASERVE&quot;&gt;&lt;/parameterList _jt=&quot;array&quot;
  itemsClass=&quot;CasterScheduleParameter&quot; size=&quot;
2&quot;&gt;&lt;/item _jt=&quot;CasterScheduleParameter&quot;
  enabled=&quot;true&quot; index=&quot;0&quot; name=&quot;COUNTRY&quot;
  value=&quot;ENGLAND&quot; /&gt;&lt;/item
_jt=&quot;CasterScheduleParameter&quot; enabled=&quot;true&quot; index=&quot;1&quot;
  name=&quot;DEALER_COST&quot; value=&quot;10000&quot; /&gt;&lt;/
parameterList&gt;&lt;/item&gt;&lt;/taskList&gt;
  &lt;/casterObject&gt;&lt;/rootObject&gt; "/>
  <entry key="IBIRS_args" value="__null"/>
  <entry key="IBIRS_" value="/WFC/Repository/RESTful_Web_Services/Car_Reports/
REST_Schedule.sch"/>
  </ibfsparams>
  <rootObject _jt="IBFSCasterObject" defaultLng="en_US" description="Schedule Created
through REST" dummy="false" extension="sch"
    externalId="Sebc72ee3sd148s41ees8a8fs9c92340b99bb" fullPath="/WFC/Repository/
RESTful_Web_Services/Car_Reports/REST_Schedule.sch"
    handle="9f013bcaI357fI4c69Ib7ceIle96775f72cb" length="0"
name="REST_Schedule.sch" policy="///D///9+f/////f/////////8AAAA="
  rsPath="/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services
/Car_Reports/REST_Schedule.sch" type="CasterSchedule">
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
  <entry>
    <key _jt="string" value="en_US"/>
    <value _jt="ArrayList" size="1">
      <item _jt="string" index="0" value="Schedule Created through REST"/>
    </value>
  </entry>
  </nlsValues>
  <properties size="2">
    <entry key="id" value="Sebc72ee3sd148s41ees8a8fs9c92340b99bb"/>
    <entry key="tool" value="schedule"/>

```

```

</properties>
<casterObject _jt="CasterSchedule" active="false"
compressedReport="false" deleteJobAfterRun="false"
description="Schedule Created through REST" destinationAddress="OWNER"
ibfsId="9f013bca1357f14c691b7ce11e96775f72cb"
  ibfsPath="" id="Sebc72ee3sd148s41ees8a8fs9c92340b99bb" name=""
nextRunTime="disabled" notification="INACTIVE"
  owner="admin"
policy="open,delete,rename,|,run,|,security;makeRules;viewRules" priority="3"
recurrence="0"
  scheduleId="Sebc72ee3sd148s41ees8a8fs9c92340b99bb" scheduleTitle="Schedule
Created through REST" sendMethod="LIBRARY"
  statusLastExecuted="" summary="" taskType="1" traceType="0"><notification
_jt="CasterScheduleNotification"
  addressForBriefNotification="" addressForFullNotification="" description=""
from="" subject="" type="INACTIVE"/>
  <distributionList _jt="array" itemsClass="CasterScheduleDistribution" size="1">
    <item accessList="" accessListFullPath="" accessType="OWNER" category=""
      compressionEnabled="false" counter="0" description="Report Library"
destinationIbfsId="c60b1f9a_05ef_4e72_a737_e869917607db"
      destinationPath="IBFS:/WFC/Repository/RESTful_Web_Services
/Car_Reports" enabled="true" expirationData="1" expirationMode="N"
      id="De465359cddf8fd41d2da9f3d1fd0080f2220" index="0" type="LIBRARY"
valueonly="false">
      <storageLibraryEmail authEnabled="false" authPassword="" authUserId=""
        libraryURL="http://localhost:8080/ibi_apps/library/report.rc"
mailFrom="" mailMessage="" mailReplyAddress=""
        mailServerName="ibismtp.ibi.com" mailSubject=""
sendEmailAfterSaveReport="false" sslEnabled="false" tlsEnabled="false"/>
    </item>
  </distributionList>
  <timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
    <item description="" enabled="true" id="Iace3d448i9197i4611i927di2969f6607559"
      index="0" name="" type="0">
      <nextRunTime _jt="calendar" time="1355756400000" timeZone="America/New_York"/>

```

```

    <startTime _jt="calendar" time="1355756400000" timeZone="America/New_York"/>
  </item>
</timeInfoList>
<taskList _jt="array" itemsClass="CasterScheduleTask" size="1">
  <item alertEnabled="false" burst="true" description="Task 1" domainHREF=""
    enabled="true" execId="guest" execPassword="guest"
    firstPostProcessingProcedure="" firstPreProcessingProcedure=""
    folderHREF="" id="T65819f8at8felt4db5t9c10t07c10277175b" index="0"
    procedureDescription=""
    procedureId="64e971c8_fd80_4d07_99a7_a2356743010b"
    procedureName="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/
    Sales_Report_by_Country.fex" reportName=
    "car_sales.htm"
    secondPostProcessingProcedure="" secondPreProcessingProcedure=""
    sendFormat="HTML" serverName="EDASERVE" type="1">
    <parameterList _jt="array" itemsClass="CasterScheduleParameter" size="2">
      <item _jt="CasterScheduleParameter" enabled="true" index="0"
        name="COUNTRY" type="0" value="ENGLAND"/>
      <item _jt="CasterScheduleParameter" enabled="true" index="1"
        name="DEALER_COST" type="0" value="10000"/>
    </parameterList>
  </item>
</taskList>
<lastTimeExecuted _jt="calendar" time="18000000" timeZone="America/New_York"/>
</casterObject>
</rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the Schedule was successfully added.

Example 2: Updating a Schedule

This example updates a Schedule called REST_Schedule that runs the Sales_Report_by_Country WebFOCUS report once on December 17th, 2012 at 15:00:00 UTC and distributes the output to the Report Library. The report will run with the COUNTRY parameter set to ENGLAND and the DEALER_COST parameter set to 10000.

The following REST URL retrieves an existing Schedule:

```

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/
REST_Schedule.sch?IBIRS_action=get

```

POST Request URL Format:

```

http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/
RESTful_Web_Services/Car_Reports/REST_Schedule.sch

```

Body Format:

```

IBIRS_action=put&IBIRS_replace=true&IBIRS_object=
  <rootObject _jt="IBFSCasterObject" binary="false" createdOn="1350492747568"
defaultLng="en_US" description="Schedule Created through REST"
  dummy="false" effectiveRSName="EDASERVE" extension="sch"
externalId="Sebc72ee3sd148s41ees8a8fs9c92340b99bb"
fullPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_Schedule.sch"
handle="9f013bcaI357fI4c69Ib7ceI1e96775f72cb"
  lastModified="1350492747568" lastaccessBy="admin"
lastaccessOn="1350492822549" length="0" name="REST_Schedule.sch" ownerId="10001"
  ownerName="admin" ownerType="U" policy="//3/D//9+f7//f/////////8AAAA="
returnedLng="en_US"
  rsPath="/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/
Car_Reports/REST_Schedule.sch" signedOn="true" type="CasterSchedule">
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="Schedule Created through REST"/>
      </value>
    </entry>
  </nlsValues>
  <properties size="2"><
    <entry key="id" value="Sebc72ee3sd148s41ees8a8fs9c92340b99bb"/>
    <entry key="tool" value="schedule"/>
  </properties>
  <casterObject _jt="CasterSchedule" active="false"
compressedReport="false" deleteJobAfterRun="false"
  description="Schedule Created through REST" destinationAddress="OWNER"
ibfsId="9f013bcaI357fI4c69Ib7ceI1e96775f72cb"
  ibfsPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports"
id="Sebc72ee3sd148s41ees8a8fs9c92340b99bb"
  name="REST_Schedule.sch" nextRunTime="disabled" notification="INACTIVE"
owner="admin"
policy="open,delete,rename,|,run,|,security;makeRules;viewRules" priority="3"
recurrence="0"
  scheduleId="Sebc72ee3sd148s41ees8a8fs9c92340b99bb" scheduleTitle="Schedule
Created through REST" sendMethod="LIBRARY"
  statusLastExecuted="" taskType="1" traceType="0">
  <notification _jt="CasterScheduleNotification" addressForBriefNotification=""
addressForFullNotification="" description=""
  from="" id="" subject="" type="INACTIVE"/>
  <distributionList _jt="array" itemsClass="CasterScheduleDistribution" size="1">

```

```

        <item accessList="" accessListFullPath="" accessType="OWNER" category=""
            compressionEnabled="false" counter="0" description="Report Library"
destinationIbfsId="c60blf9a_05ef_4e72_a737_e869917607db"
destinationPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports" enabled="true"
expirationData="1" expirationMode="N"
            id="De465359cddf8fd41d2da9f3d1fd0080f2220" index="0" type="LIBRARY"
valueonly="false">
            <storageLibraryEmail authEnabled="false" authPassword="" authUserId=""
                libraryURL="http://localhost:8080/ibi_apps/library/report.rc"
mailFrom="" mailMessage="" mailReplyAddress=""
                mailServerName="ibismtp.ibi.com" mailSubject=""
sendEmailAfterSaveReport="false" sslEnabled="false" tlsEnabled="false"/>
        </item>
    </distributionList>
    <timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
        <item description="" enabled="true" id="Iace3d448i9197i4611i927di2969f6607559"
            index="0" name="" type="0">
            <nextRunTime _jt="calendar" time="1355756400000" timeZone="America/New_York"/>
        <startTime _jt="calendar" time="1355756400000" timeZone="America/New_York"/>
        </item>
    </timeInfoList>
    <taskList _jt="array" itemsClass="CasterScheduleTask" size="1">
        <item alertEnabled="false" burst="true" description="Task 1" domainHREF=""
            enabled="true" execId="guest" execPassword=""
firstPostProcessingProcedure="" firstPreProcessingProcedure="" folderHREF=""
            id="T65819f8at8felt4db5t9c10t07c10277175b" index="0"
procedureDescription="" procedureId="64e971c8_fd80_4d07_99a7_a2356743010b"
            procedureName="IBFS:/WFC/Repository/RESTful_Web_Services/
Car_Reports/Sales_Report_by_Country.fex" reportName="car_sales.htm"
            secondPostProcessingProcedure="" secondPreProcessingProcedure=""
sendFormat="HTML" serverName="EDASERVE" type="1">
            <parameterList _jt="array" itemsClass="CasterScheduleParameter" size="2">
                <item _jt="CasterScheduleParameter" enabled="true" index="0"
name="COUNTRY" type="0" value="ENGLAND"/>
                <item _jt="CasterScheduleParameter" enabled="true" index="1"
name="DEALER_COST" type="0" value="10000"/>
            </parameterList>
        </item>
    </taskList>
    <lastTimeExecuted _jt="calendar" time="18000000" timeZone="America/New_York"/>
</casterObject>
</rootObject>

```

Response:

If the value for the *returncode* attribute in the XML response is 10000, then the Schedule was successfully added.

Running a Schedule

This RESTful web service request can be used to run an existing ReportCaster Schedule.

HTTP Method: POST

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ScheduleName
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder used for the stored ReportCaster Schedule. If the folder used for the stored ReportCaster Schedule exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

ScheduleName

Is the name of the ReportCaster Schedule to run, which also must have a .sch extension.

Body Format:

```
IBIRS_action=run
```

Example:

The following example demonstrates how to run a ReportCaster Schedule called REST_Schedule.

POST Request URL:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/  
Car_Reports/REST_Schedule.sch
```

Body:

```
IBIRS_action=run
```

Response:

A job number is returned in HTML format. For example:

```
J453ce7a4je11bj48ffj832ej9053e5377495
```

Retrieving a Schedule

This RESTful web service request can be used to retrieve an existing ReportCaster Schedule.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ScheduleNameIBIRS_action=get`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder used for the stored ReportCaster Schedule. If the folder used for the stored ReportCaster Schedule exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

ScheduleName

Is the name of the ReportCaster Schedule to retrieve, which also must have a .sch extension.

Example:

In the following example, a schedule called REST_Schedule.sch is retrieved from the Car_Reports folder, which is within the RESTful_Web_Services folder.

Request:

`http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_Schedule.sch?IBIRS_action=get`

Response:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="get" returncode="10000"
returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfparams size="2">
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/WFC/Repository/RESTful_Web_Services/Car_Reports/
REST_Schedule.sch"/>
  </ibfparams>
  <rootObject _jt="IBFSCasterObject" binary="false" createdOn="1356718595487"
defaultLng="en_US" description="Schedule Created through REST"
  dummy="false" effectiveRSName="EDASERVE" extension="sch"
externalId="S1995b2ecsa8f6s4096sa62es1867fa2d7a85"
fullPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports/REST_Schedule.sch"
handle="7c2fd2a3I2dbcI400dIb666I3512e8d8b89f"
  lastModified="1356718595487" lastaccessBy="admin" lastaccessOn="1356719962891"
length="0" name="REST_Schedule.sch" ownerId="10001"
  ownerName="admin" ownerType="U" policy="//3/D///9+P9///v////////+AAAA="
returnedLng="en_US"
rsPath="/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/
REST_Schedule.sch" type="CasterSchedule">
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="Schedule Created through REST"/>
      </value>
    </entry>
  </nlsValues>
  <properties size="2">
    <entry key="id" value="S1995b2ecsa8f6s4096sa62es1867fa2d7a85"/>
    <entry key="tool" value="schedule"/>
  </properties>
  <casterObject _jt="CasterSchedule" active="false" compressedReport="false"
deleteJobAfterRun="false"
  description="Schedule Created through REST" destinationAddress="OWNER"
ibfsId="7c2fd2a3I2dbcI400dIb666I3512e8d8b89f"
  ibfsPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports"
id="S1995b2ecsa8f6s4096sa62es1867fa2d7a85"
  name="REST_Schedule.sch" nextRunTime="disabled" notification="INACTIVE"
owner="admin"
  policy="open,delete,rename,|,run,|,security;makeRules;viewRules"
priority="3" recurrence="0"
  scheduleId="S1995b2ecsa8f6s4096sa62es1867fa2d7a85" scheduleTitle="Schedule
Created through REST" sendMethod="LIBRARY"
  statusLastExecuted="" taskType="1" traceType="0">
  <notification _jt="CasterScheduleNotification"

```

```

addressForBriefNotification="" addressForFullNotification="" description=""
  from="" id="" subject="" type="INACTIVE"/>
  <distributionList _jt="array" itemsClass="CasterScheduleDistribution" size="1">
    <item accessList="" accessListFullPath="" accessType="OWNER" category=""
      compressionEnabled="false" counter="0" description="Report Library"
      destinationIbfsId="c60b1f9a_05ef_4e72_a737_e869917607db"
      destinationPath="IBFS:/WFC/Repository/RESTful_Web_Services/Car_Reports"
      disabled="false" expirationData="1" expirationMode="N"
      id="D58215579d4885d4b5eda023d9f44d1b4da01" index="0" type="LIBRARY"
      valueonly="false">
      <storageLibraryEmail authEnabled="false" authPassword="" authUserId=""
        libraryURL="http://localhost:8080/ibi_apps/library/report.rc"
        mailFrom="" mailMessage="" mailReplyAddress=""
          mailServerName="ibismtp.ibi.com" mailSubject=""
        sendEmailAfterSaveReport="false" sslEnabled="false" tlsEnabled="false"/>
    </item>
  </distributionList>
  <timeInfoList _jt="array" itemsClass="CasterScheduleTimeInfo" size="1">
    <item description="" disabled="false"
      id="Ifc777178ilab0i42faibd06i81df82c234e7"
      index="0" name="" type="0">
      <nextRunTime _jt="calendar" time="1355756400000" timeZone="America/New_York"/>
      <startTime _jt="calendar" time="1355756400000" timeZone="America/New_York"/>
    </item>
  </timeInfoList>
  <taskList _jt="array" itemsClass="CasterScheduleTask" size="1">
    <item alertEnabled="false" burst="true" description="Task 1" disabled="false"
      domainHREF="" execId="guest" execPassword=""
      firstPostProcessingProcedure="" firstPreProcessingProcedure="" folderHREF=""
      id="Tcdde20bdt3305t436ata200tecd3367ad16f" index="0"
      procedureDescription=""
      procedureId="64e971c8_fd80_4d07_99a7_a2356743010b"
      procedureName="IBFS:/WFC/Repository/RESTful_Web_Services/
      Car_Reports/Sales_Report_by_Country.fex" reportName="car_sales.htm"
      secondPostProcessingProcedure="" secondPreProcessingProcedure=""
      sendFormat="HTML" serverName="EDASERVE" type="1">
      <parameterList _jt="array" itemsClass="CasterScheduleParameter" size="2">
        <item _jt="CasterScheduleParameter" index="0" name="COUNTRY" type="0"
          useDefaultValue="false" value="ENGLAND"/>
        <item _jt="CasterScheduleParameter" index="1" name="DEALER_COST" type="0"
          useDefaultValue="false" value="10000"/>
      </parameterList>
    </item>
  </taskList>
  <lastTimeExecuted _jt="calendar" time="18000000" timeZone="America/New_York"/>
</casterObject>
</rootObject>
</ibfsrpc>

```

Deleting a Schedule

This RESTful web service request can be used to delete an existing ReportCaster Schedule.

HTTP Method: DELETE

REST URL Format:

```
http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/ScheduleName?  
IBIRS_action=delete
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder used for the stored ReportCaster Schedule. If the folder used for the stored ReportCaster Schedule exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

ScheduleName

Is the name of the ReportCaster Schedule to delete, which also must have a .sch extension.

Example:

In the following example, the ReportCaster Schedule named REST_schedule.sch is deleted from the Car_Reports folder, which is within the RESTful_Web_Services folder.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/  
Car_Reports/REST_schedule.sch?IBIRS_action=delete
```

Response:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="delete" returncode="10000"
returndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfparams size="2">
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/WFC/Repository/RESTful_Web_Services/Car_Reports/
REST_schedule.sch"/>
  </ibfparams>
  <rootObject _jt="IBFSCasterObject" binary="false" createdOn="1355156594727"
defaultLng="en_US" description="Schedule Created through REST"
  dummy="false" extension="sch" externalId="Sdc748ba8s5ff1s4390sb3c1s3777d7686d9f"
  fullPath="IBFS:/WFC/Repository/RESTful_Web_Services/
Car_Reports/REST_Schedule.sch" handle="4b4c8010Ib22cI4609I9c41Ie7102db522b1"
  lastModified="1356617719033" lastaccessBy="admin" lastaccessOn="1356623237446"
length="0" name="REST_Schedule.sch" ownerId="10001"
  ownerName="admin" ownerType="U" policy="//3/D///9+P9///v/////////+AAAA="
returnedLng="en_US"
rsPath="/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/Car_Reports/
REST_Schedule.sch" type="CasterSchedule">
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="Schedule Created through REST"/>
      </value>
    </entry>
  </nlsValues>
  <properties size="2">
    <entry key="id" value="Sdc748ba8s5ff1s4390sb3c1s3777d7686d9f"/>
    <entry key="tool" value="schedule"/>
  </properties>
</rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the ReportCaster Schedule was deleted successfully.

Deleting an Address Book

This RESTful web service request can be used to delete an existing ReportCaster Address Book.

HTTP Method: DELETE

REST URL Format:

```

http://host:port/ibi_apps/rs/ibfs/WFC/Repository/FolderName/AddressBookName?
IBIRS_action=delete

```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

FolderName

Is the name of the folder used for the stored ReportCaster Address Book. If the folder used for the stored ReportCaster Address Book exists as a subfolder, then the path to the subfolder name must be included in the REST URL. For example, TopFolderName/SubFolderName.

AddressBookName

Is the name of the ReportCaster Address Book to delete, which also must have a .adr extension.

Example:

In the following example, the ReportCaster Address Book named REST_Distribution_List.adr is deleted from the Car_Reports folder, which is within the RESTful_Web_Services folder.

Request:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/
Car_Reports/REST_Distribution_List.adr?IBIRS_action=delete
```

Response:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<ibfsrpc _jt="IBFSResponseObject" language="EN" name="delete" returncode="10000"
returnndesc="SUCCESS" subreturncode="0" subsystem="SSYS" type="simple">
  <ibfsparams size="2">
    <entry key="IBIRS_args" value="__null"/>
    <entry key="IBIRS_" value="/WFC/Repository/RESTful_Web_Services/
Car_Reports/REST_Distribution_List.adr"/>
  </ibfsparams>
  <rootObject _jt="IBFSCasterObject" binary="false" createdOn="1350862349237"
defaultLng="en_US" description="REST Distribution List"
dummy="false" extension="adr" externalId="1a7ddf0eIff6aI4886Ibde9I77c691d280a0"
fullPath="IBFS:/WFC/Repository/RESTful_Web_Services/
Car_Reports/REST_Distribution_List.adr" handle="1a7ddf0eIff6aI4886Ibde9I77c691d280a0"
```

```

        lastModified="1350862989380" lastaccessBy="admin"
lastaccessOn="1356623807376" length="0" name="REST_Distribution_List.adr"
ownerId="10001"
        ownerName="admin" ownerType="U" policy="//3/D//9+P////v////////+AAAA="
returnedLng="en_US"
        rsPath="/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_Services/
Car_Reports/REST_Distribution_List.adr" type="CasterDistributionList">
  <nlsValues _jt="HashMap" loadFactor="0.75" threshold="12">
    <entry>
      <key _jt="string" value="en_US"/>
      <value _jt="ArrayList" size="2">
        <item _jt="string" index="0" value="REST Distribution List"/>
      </value>
    </entry>
  </nlsValues>
<properties size="3">
  <entry key="id" value="1a7ddf0eIff6aI4886Ibde9I77c691d280a0"/>
  <entry key="tool" value="addressbook"/>
  <entry key="method" value="EMAIL"/>
</properties>
</rootObject>
</ibfsrpc>

```

If the value for the *returncode* attribute in the XML response is 10000, then the ReportCaster Address Book was deleted successfully.

Log Functionality

This section describes the format and structure of RESTful web service requests that are used for a variety of ReportCaster logging functionality.

Deleting a Specific Log

This RESTful web service request can be used to delete a ReportCaster log for a specific job.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/services/LogServiceREST/deleteLogByJobId?jobId=jobId`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

jobId

Is a unique identifier for the ReportCaster job.

Example:

In the following example, the ReportCaster log for job ID J34558adaj1b4dj4e6cjaddcj3745b2688f2c is deleted.

Request:

```
http://localhost:8080/ibi_apps/services/LogServiceREST/deleteLogByJobId?
jobId=J34558adaj1b4dj4e6cjaddcj3745b2688f2c
```

Response:

```
<ns:deleteLogByJobIdResponse xmlns:ns="http://ws.api.broker.ibi">
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/>
</ns:deleteLogByJobIdResponse>
```

Deleting Logs for a Specific Time Period

This RESTful web service request can be used to delete ReportCaster logs for all schedules that were run between a specific time interval.

If there is no value for the start time of the time interval, then all log records before the end time of the time interval are deleted.

If there is no value for the end time of the time interval, then all log records after the start time of the time interval are deleted.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/services/LogServiceREST/deleteLogList?startTime=startTime&
endTime=endTime
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

startTime

Is the start time for when the logs are to be deleted. The following format must be used:

YYYY/MM/DD%20HH:MM:SS

endTime

Is the end time for when the logs are to be deleted. The following format must be used:

YYYY/MM/DD%20HH:MM:SS

Example:

In the following example, all ReportCaster logs between 2014-02-20 11:00:00 and 2014-02-20 13:00:00 are to be deleted.

Request:

```
http://localhost:8080/ibi_apps/services/LogServiceREST/deleteLogList?
startTime=2014/02/20%2011:00:00&
endTime=2014/02/20%2013:00:00
```

Response:

```
<ns:deleteLogListResponse xmlns:ns="http://ws.api.broker.ibi">
  <ns:return>2</ns:return>
</ns:deleteLogListResponse>
```

The value within the <return> element indicates the number of ReportCaster logs that were deleted.

Deleting Logs for an Owner

This RESTful web service request can be used to delete ReportCaster logs for all schedules that were run between a specific time interval for a specific log owner.

If there is no value for the start time of the time interval, then all log records before the end time of the time interval are deleted.

If there is no value for the end time of the time interval, then all log records after the start time of the time interval are deleted.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/services/LogServiceREST/deleteLogListByOwner?owner=owner&
startTime=startTime&endTime=endTime
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

owner

Is the owner of the ReportCaster log.

startTime

Is the start time for when the logs are to be deleted. The following format must be used:

YYYY/MM/DD%20HH:MM:SS

endTime

Is the end time for when the logs are to be deleted. The following format must be used:

YYYY/MM/DD%20HH:MM:SS

Example:

In the following example, all ReportCaster logs between 2014-02-21 09:00:00 and 2014-02-21 10:00:00 for the log owner named *admin* are to be deleted.

Request:

```
http://localhost:8080/ibi_apps/services/LogServiceREST/deleteLogListByOwner?
owner=admin&
startTime=2014/02/21%2009:00:00&endTime=2014/02/21%2010:00:00
```

Response:

```
<ns:deleteLogListByOwnerResponse xmlns:ns="http://ws.api.broker.ibi">
  <ns:return>2</ns:return>
</ns:deleteLogListByOwnerResponse>
```

The value within the <return> element indicates the number of ReportCaster logs that were deleted.

Deleting Logs for a Schedule ID

This RESTful web service request can be used to delete all ReportCaster logs a specific schedule identified by a schedule ID.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/services/LogServiceREST/deleteLogListByScheduleId?
scheduleId=scheduleId
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

scheduleId

Is the schedule ID for the ReportCaster schedule.

Example:

In the following example, all ReportCaster logs for schedule ID S23f65030s728as482asa632s879fd9f6a727 are to be deleted.

Request:

```
http://localhost:8080/ibi_apps/services/LogServiceREST/deleteLogListByScheduleId?scheduleId=S23f65030s728as482asa632s879fd9f6a727
```

Response:

```
<ns:deleteLogListByScheduleIdResponse xmlns:ns="http://ws.api.broker.ibi">
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/>
</ns:deleteLogListByScheduleIdResponse>
```

Deleting Logs for a Schedule ID Within a Time Period

This RESTful web service request can be used to delete all ReportCaster logs that were run between a specific time interval for a specific schedule identified by the schedule ID.

If there is no value for the start time of the time interval, then all log records before the end time of the time interval are deleted.

If there is no value for the end time of the time interval, then all log records after the start time of the time interval are deleted.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/services/LogServiceREST/deleteLogListByScheduleIdByCalendar?scheduleId=scheduleId&startTime=startTime&endTime=endTime
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

scheduleId

Is the schedule ID for the ReportCaster schedule.

startTime

Is the start time for when the logs are to be deleted. The following format must be used:

YYYY/MM/DD%20HH:MM:SS

endTime

Is the end time for when the logs are to be deleted. The following format must be used:

YYYY/MM/DD%20HH:MM:SS

Example:

In the following example, all ReportCaster logs between 2014-02-21 12:00:00 and 2014-02-21 13:00:00 for schedule ID Sca76e628s892as43a4sbddcs10875ff7f188 are to be deleted.

Request:

```
http://localhost:8080/ibi_apps/services/LogServiceREST/
deleteLogListByScheduleIdByCalendar?
scheduleId=Sca76e628s892as43a4sbddcs10875ff7f188&
startTime=2014/02/21%2012:00:00&endTime=2014/02/21%2013:00:00
```

Response:

```
<ns:deleteLogListByScheduleIdByCalendarResponse xmlns:ns="http://ws.api.broker.ibi">
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true"/>
</ns:deleteLogListByScheduleIdByCalendarResponse>
```

Retrieving Last Log for a Schedule ID

This RESTful web service request can be used to retrieve the last ReportCaster log for a specific schedule identified by the schedule ID.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/services/LogServiceREST/getLastLogByScheduleId?
scheduleId=scheduleId
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

scheduleId

Is the schedule ID for the ReportCaster schedule.

Example:

In the following example, the last ReportCaster log for schedule ID Sca76e628s892as43a4sbddcs10875ff7f188 is to be retrieved.

Request:

http://localhost:8080/ibi_apps/services/LogServiceREST/getLastLogByScheduleId?scheduleId=Sca76e628s892as43a4sbddcs10875ff7f188

Response:

```

<ns:getLastLogByScheduleIdResponse xmlns:ns="http://ws.api.broker.ibi">
  <ns:return xmlns:ax264="http://io.java/xsd" xmlns:ax263="http://rmi.java/xsd"
xmlns:ax267="http://dslog.data.api.broker.ibi/xsd" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" xmlns:ax261="http://schedule.data.api.broker.ibi/xsd"
xsi:type="ax267:DsLog">
  <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
  <ax267:description xsi:nil="true"/>
  <ax267:endTime>2014-02-24T09:00:25.861-05:00</ax267:endTime>
  <ax267:errorType>0</ax267:errorType>
  <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
  <ax267:ibfsPath/>
  <ax267:id>J4ce5d61ejf6b2j441dja02ej084628360372</ax267:id>
  <ax267:jobId>J4ce5d61ejf6b2j441dja02ej084628360372</ax267:jobId>
  <ax267:logElementList xsi:type="ax267:DsLogElement">
    <ax267:error>false</ax267:error>
    <ax267:execId xsi:nil="true"/>
    <ax267:message>Schedule Executed On Demand at IBI-Laptop:8201 (IBI-Laptop/
172.44.18.74)</ax267:message>
    <ax267:messageCode>BTP1010</ax267:messageCode>
    <ax267:taskDescription xsi:nil="true"/>
    <ax267:time>2014-02-24T09:00:23.126-05:00</ax267:time>
    <ax267:warning>false</ax267:warning>
  </ax267:logElementList>
  <ax267:logElementList xsi:type="ax267:DsLogElement">
    <ax267:error>false</ax267:error>
    <ax267:execId xsi:nil="true"/>
    <ax267:message>Job placed in the waiting queue at 2014-02-24
09:00:23.104-0500 (1,393,250,423,104)</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
    <ax267:taskDescription xsi:nil="true"/>
    <ax267:time>2014-02-24T09:00:23.140-05:00</ax267:time>
    <ax267:warning>false</ax267:warning>
  </ax267:logElementList>
  <ax267:logElementList xsi:type="ax267:DsLogElement">
    <ax267:error>false</ax267:error>
    <ax267:execId xsi:nil="true"/>
    <ax267:message>Job started running at 2014-02-24 09:00:23.122-0500
(1,393,250,423,122)</ax267:message>

```

```

    <ax267:messageCode>BTP1010</ax267:messageCode>
    <ax267:taskDescription xsi:nil="true"/>
    <ax267:time>2014-02-24T09:00:23.141-05:00</ax267:time>
    <ax267:warning>>false</ax267:warning>
  </ax267:logElementList>
  <ax267:logElementList xsi:type="ax267:DsLogElement">
    <ax267:error>>false</ax267:error>
    <ax267:execId xsi:nil="true"/>
    <ax267:message>Job remained in waiting queue for 0.018 seconds</
ax267:message>
    <ax267:messageCode>BTP1010</ax267:messageCode>
    <ax267:taskDescription xsi:nil="true"/>
    <ax267:time>2014-02-24T09:00:23.141-05:00</ax267:time>
    <ax267:warning>>false</ax267:warning>
  </ax267:logElementList>
  <ax267:logElementList xsi:type="ax267:DsLogElement">
    <ax267:error>>false</ax267:error>
    <ax267:execId>guest</ax267:execId>
    <ax267:message>Starting task: Task 1</ax267:message>
    <ax267:messageCode>BTP1020</ax267:messageCode>
    <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
    <ax267:time>2014-02-24T09:00:23.752-05:00</ax267:time>
    <ax267:warning>>false</ax267:warning>
  </ax267:logElementList>
  <ax267:logElementList xsi:type="ax267:DsLogElement">
    <ax267:error>>false</ax267:error>
    <ax267:execId>guest</ax267:execId>
    <ax267:message>Task type: EDA RPC</ax267:message>
    <ax267:messageCode>BTP1020</ax267:messageCode>
    <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
    <ax267:time>2014-02-24T09:00:23.752-05:00</ax267:time>
    <ax267:warning>>false</ax267:warning>
  </ax267:logElementList>
  <ax267:logElementList xsi:type="ax267:DsLogElement">
    <ax267:error>>false</ax267:error>
    <ax267:execId>guest</ax267:execId>
    <ax267:message>Procedure name: ibisamp/carinst</ax267:message>
    <ax267:messageCode>BTP1020</ax267:messageCode>
    <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
    <ax267:time>2014-02-24T09:00:23.752-05:00</ax267:time>
    <ax267:warning>>false</ax267:warning>
  </ax267:logElementList>

```



```

<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Connecting to server EDASERVE with execution id
guest at 2014-02-24 09:00:23.753-0500 (1,393,250,423,753)</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:23.753-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Connection to the Reporting Server EDASERVE
established at 2014-02-24 09:00:24.323-0500 (1,393,250,424,323)</
ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:24.323-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>The time to establish a connection to the
Reporting Server EDASERVE was 0.57 seconds</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:24.325-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Executing focexec.</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:24.351-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>

```

```

<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>0 HOLDING HTML FILE ON PC DISK .../</
ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:24.974-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Connection to the Reporting Server EDASERVE
closed at 2014-02-24 09:00:25.040-0500 (1,393,250,425,040)</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.040-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Job ran on the Reporting Server EDASERVE for
0.717 seconds</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.040-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Task finished.</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.040-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Starting distribution: Report Library</
ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription xsi:nil="true"/>
  <ax267:time>2014-02-24T09:00:25.060-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>

```

```

<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Distribution method: Report Library</
ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription xsi:nil="true"/>
  <ax267:time>2014-02-24T09:00:25.060-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Report distributed to ReportLibrary
(Ld7e0274610c0514dc11a18f10d3d1d20d9a9) with the group id
Lc8f723df65ca6cebd2b86b264f4cfc14 at the version 15 </ax267:message>
  <ax267:messageCode>DS10001</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.309-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Library access option "Private" is applied.</
ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.309-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Retrieving email address list for library watch
list</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.312-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>

```

```

<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Library Notification email sent to
myEmail@ibi.com</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.639-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Distribution finished.</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.642-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Job finished at 2014-02-24 09:00:25.642-0500
(1,393,250,425,642)</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.643-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Job time on distribution server after the report
completed was 0.582 seconds</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.643-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>

```

```

<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Total running time was 2.52 seconds</
ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.644-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Total elapsed time (including the queue time)
was 2.538 seconds</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-24T09:00:25.645-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Brief notification successfully sent to
myEmail@ibi.com.</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription xsi:nil="true"/>
  <ax267:time>2014-02-24T09:00:25.751-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Log report notification successfully sent to
myEmail@ibi.com.</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription xsi:nil="true"/>
  <ax267:time>2014-02-24T09:00:25.861-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:name xsi:nil="true"/>
<ax267:owner>admin</ax267:owner>
<ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
  <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
  <ax267:startTime>2014-02-24T09:00:23.126-05:00</ax267:startTime>
  <ax267:summary xsi:nil="true"/>
</ns:return>
</ns:getLastLogByScheduleIdResponse>

```

The following table lists and describes the possible *errorType* code values that are returned in the XML response document.

errorType Code Value	Description
0	No Error
1	Error
2	Warning
6	Running
7	Running With Error

Retrieving the Log for a Job ID

This RESTful web service request can be used to retrieve the last ReportCaster log for a specific job identified by the job ID.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/services/LogServiceREST/getLogByJobId?processId=jobId`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

jobId

Is a unique identifier for the ReportCaster job.

Example:

In the following example, the ReportCaster log for job ID J0c6828cfj96f0j4363ja81ejd41e3782cff2 is to be retrieved.

Request:

`http://localhost:8080/ibi_apps/services/LogServiceREST/getLogByJobId?processId=J0c6828cfj96f0j4363ja81ejd41e3782cff2`

Response:

```

<ns:getLogByJobIdResponse xmlns:ns="http://ws.api.broker.ibi">
  <ns:return xmlns:ax264="http://io.java/xsd" xmlns:ax263="http://rmi.java/xsd"
xmlns:ax267="http://dslog.data.api.broker.ibi/xsd" xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" xmlns:ax261="http://schedule.data.api.broker.ibi/xsd"
xsi:type="ax267:DsLog">
  <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
  <ax267:description xsi:nil="true"/>
  <ax267:endTime>2014-02-19T16:19:08.674-05:00</ax267:endTime>
  <ax267:errorType>0</ax267:errorType>
  <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
  <ax267:ibfsPath/>
  <ax267:id>J0c6828cfj96f0j4363ja81ejd41e3782cff2</ax267:id>
  <ax267:jobId>J0c6828cfj96f0j4363ja81ejd41e3782cff2</ax267:jobId>
  <ax267:logElementList xsi:type="ax267:DsLogElement">
    <ax267:error>false</ax267:error>
    <ax267:execId xsi:nil="true"/>
    <ax267:message>Schedule Executed On Demand at IBI-Laptop:8201 (IBI-Laptop/
172.44.18.74)</ax267:message>
    <ax267:messageCode>BTP1010</ax267:messageCode>
    <ax267:taskDescription xsi:nil="true"/>
    <ax267:time>2014-02-19T16:19:08.061-05:00</ax267:time>
    <ax267:warning>false</ax267:warning>
  </ax267:logElementList>
  <ax267:logElementList xsi:type="ax267:DsLogElement">
    <ax267:error>false</ax267:error>
    <ax267:execId xsi:nil="true"/>
    <ax267:message>Job placed in the waiting queue at 2014-02-19
16:19:08.055-0500 (1,392,844,748,055)</ax267:message>
    <ax267:messageCode>BTP1010</ax267:messageCode>
    <ax267:taskDescription xsi:nil="true"/>
    <ax267:time>2014-02-19T16:19:08.061-05:00</ax267:time>
    <ax267:warning>false</ax267:warning>
  </ax267:logElementList>
  <ax267:logElementList xsi:type="ax267:DsLogElement">
    <ax267:error>false</ax267:error>
    <ax267:execId xsi:nil="true"/>
    <ax267:message>Job started running at 2014-02-19 16:19:08.056-0500
(1,392,844,748,056)</ax267:message>
    <ax267:messageCode>BTP1010</ax267:messageCode>
    <ax267:taskDescription xsi:nil="true"/>
    <ax267:time>2014-02-19T16:19:08.061-05:00</ax267:time>
    <ax267:warning>false</ax267:warning>
  </ax267:logElementList>

```

```

<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Job remained in waiting queue for 0.0010 seconds</
ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription xsi:nil="true"/>
  <ax267:time>2014-02-19T16:19:08.061-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Starting task: Task 1</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.129-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Task type: EDA RPC</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.130-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Procedure name: ibisamp/carinst</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.130-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>

```



```

<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Connecting to server EDASERVE with execution id
guest at 2014-02-19 16:19:08.130-0500 (1,392,844,748,130)</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.130-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Connection to the Reporting Server EDASERVE
established at 2014-02-19 16:19:08.157-0500 (1,392,844,748,157)</
ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.158-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>The time to establish a connection to the
Reporting Server EDASERVE was 0.027 seconds</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.158-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Executing focexec.</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.158-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>0 HOLDING HTML FILE ON PC DISK ...</
ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.215-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>

```

```

<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Connection to the Reporting Server EDASERVE
closed at 2014-02-19 16:19:08.217-0500 (1,392,844,748,217)</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.217-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Job ran on the Reporting Server EDASERVE for
0.06 seconds</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.218-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId>guest</ax267:execId>
  <ax267:message>Task finished.</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription>ibisamp/carinst</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.218-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Starting distribution: Report Library</
ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription xsi:nil="true"/>
  <ax267:time>2014-02-19T16:19:08.231-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>

```

```

<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Distribution method: Report Library</
ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription xsi:nil="true"/>
  <ax267:time>2014-02-19T16:19:08.231-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Report distributed to ReportLibrary
(L0bbafb2a137ef149941834c13b614c4afeaf) with the group id
Lc8f723df65ca6cebd2b86b264f4cfc14 at the version 3 </ax267:message>
  <ax267:messageCode>DS10001</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.327-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Library access option "Private" is applied.</
ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.327-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Retrieving email address list for library watch
list</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.327-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Library Notification email sent to
myEmail@ibi.com</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.454-05:00</ax267:time>
  <ax267:warning>>false</ax267:warning>
</ax267:logElementList>

```

```

<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Distribution finished.</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.456-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Job finished at 2014-02-19 16:19:08.456-0500
(1,392,844,748,456)</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.456-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Job time on distribution server after the report
completed was 0.225 seconds</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.457-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Total running time was 0.4 seconds</
ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.457-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>

```

```

<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Total elapsed time (including the queue time)
was 0.401 seconds</ax267:message>
  <ax267:messageCode>BTP1010</ax267:messageCode>
  <ax267:taskDescription>Distribute</ax267:taskDescription>
  <ax267:time>2014-02-19T16:19:08.457-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Brief notification successfully sent to
myEmail@ibi.com.</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription xsi:nil="true"/>
  <ax267:time>2014-02-19T16:19:08.565-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:logElementList xsi:type="ax267:DsLogElement">
  <ax267:error>false</ax267:error>
  <ax267:execId xsi:nil="true"/>
  <ax267:message>Log report notification successfully sent to
myEmail@ibi.com.</ax267:message>
  <ax267:messageCode>BTP1020</ax267:messageCode>
  <ax267:taskDescription xsi:nil="true"/>
  <ax267:time>2014-02-19T16:19:08.674-05:00</ax267:time>
  <ax267:warning>false</ax267:warning>
</ax267:logElementList>
<ax267:name xsi:nil="true"/>
<ax267:owner>admin</ax267:owner>
<ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
  <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
  <ax267:startTime>2014-02-19T16:19:08.061-05:00</ax267:startTime>
  <ax267:summary xsi:nil="true"/>
</ns:return>
</ns:getLogByJobIdResponse>

```

The following table lists and describes the possible *errorType* code values that are returned in the XML response document.

errorType Code Value	Description
0	No Error
1	Error
2	Warning

errorType Code Value	Description
6	Running
7	Running With Error

Retrieving the Log List for an Owner

This RESTful web service request can be used to retrieve a list of log information for a specific owner. The details for each log are not returned.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/services/LogServiceREST/getLogInfoListByOwner?owner=owner`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

owner

Is the owner of the log.

Example:

In the following example, a list of log information for the owner named *admin* is to be retrieved.

Request:

`http://localhost:8080/ibi_apps/services/LogServiceREST/getLogInfoListByOwner?
owner=admin`

Response:

```

<ns:getLogInfoListByOwnerResponse xmlns:ns="http://ws.api.broker.ibi"
xmlns:ax264="http://io.java/xsd" xmlns:ax263="http://rmi.java/xsd"
xmlns:ax267="http://dslog.data.api.broker.ibi/xsd" xmlns:ax261="http://
schedule.data.api.broker.ibi/xsd">
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
    <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
    <ax267:description xsi:nil="true"/>
    <ax267:endTime>2014-02-19T16:14:42.279-05:00</ax267:endTime>
    <ax267:errorType>2</ax267:errorType>
    <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
    <ax267:ibfsPath/>
    <ax267:id>J73498ee4j33caj409bjacbbj47ab9f66920d</ax267:id>
    <ax267:jobId>J73498ee4j33caj409bjacbbj47ab9f66920d</ax267:jobId>
    <ax267:logElementList xsi:nil="true"/>
    <ax267:name xsi:nil="true"/>
    <ax267:owner>admin</ax267:owner>
    <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
    <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
    <ax267:startTime>2014-02-19T16:14:41.146-05:00</ax267:startTime>
    <ax267:summary xsi:nil="true"/>
  </ns:return>
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
    <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
    <ax267:description xsi:nil="true"/>
    <ax267:endTime>2014-02-19T16:16:22.945-05:00</ax267:endTime>
    <ax267:errorType>2</ax267:errorType>
    <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
    <ax267:ibfsPath/>
    <ax267:id>J5004dec7j6c9cj4009jab87j97b8fa04831a</ax267:id>
    <ax267:jobId>J5004dec7j6c9cj4009jab87j97b8fa04831a</ax267:jobId>
    <ax267:logElementList xsi:nil="true"/>
    <ax267:name xsi:nil="true"/>
    <ax267:owner>admin</ax267:owner>
    <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
    <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
    <ax267:startTime>2014-02-19T16:16:22.298-05:00</ax267:startTime>
    <ax267:summary xsi:nil="true"/>
  </ns:return>

```

```

    <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
    <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
    <ax267:description xsi:nil="true"/>
    <ax267:endTime>2014-02-19T16:19:08.674-05:00</ax267:endTime>
    <ax267:errorType>0</ax267:errorType>
    <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
    <ax267:ibfsPath/>
    <ax267:id>J0c6828cfj96f0j4363ja81ejd41e3782cff2</ax267:id>
    <ax267:jobId>J0c6828cfj96f0j4363ja81ejd41e3782cff2</ax267:jobId>
    <ax267:logElementList xsi:nil="true"/>
    <ax267:name xsi:nil="true"/>
    <ax267:owner>admin</ax267:owner>
    <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
    <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
    <ax267:startTime>2014-02-19T16:19:08.061-05:00</ax267:startTime>
    <ax267:summary xsi:nil="true"/>
    </ns:return>
    <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
    <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
    <ax267:description xsi:nil="true"/>
    <ax267:endTime>2014-02-19T16:22:16.729-05:00</ax267:endTime>
    <ax267:errorType>0</ax267:errorType>
    <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
    <ax267:ibfsPath/>
    <ax267:id>Jd5ae6d5cj3283j4bc2ja36ejc29fd6895419</ax267:id>
    <ax267:jobId>Jd5ae6d5cj3283j4bc2ja36ejc29fd6895419</ax267:jobId>
    <ax267:logElementList xsi:nil="true"/>
    <ax267:name xsi:nil="true"/>
    <ax267:owner>admin</ax267:owner>
    <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
    <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
    <ax267:startTime>2014-02-19T16:22:16.030-05:00</ax267:startTime>
    <ax267:summary xsi:nil="true"/>
    </ns:return>

```



```

    <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
    <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
    <ax267:description xsi:nil="true"/>
    <ax267:endTime>2014-02-19T16:29:31.724-05:00</ax267:endTime>
    <ax267:errorType>0</ax267:errorType>
    <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
    <ax267:ibfsPath/>
    <ax267:id>Jbcdb8429jab89j4ed8j95d2j25dd04279f5c</ax267:id>
    <ax267:jobId>Jbcdb8429jab89j4ed8j95d2j25dd04279f5c</ax267:jobId>
    <ax267:logElementList xsi:nil="true"/>
    <ax267:name xsi:nil="true"/>
    <ax267:owner>admin</ax267:owner>
    <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
    <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
    <ax267:startTime>2014-02-19T16:29:31.083-05:00</ax267:startTime>
    <ax267:summary xsi:nil="true"/>
    </ns:return>
    <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
    <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
    <ax267:description xsi:nil="true"/>
    <ax267:endTime>2014-02-22T21:26:23.251-05:00</ax267:endTime>
    <ax267:errorType>0</ax267:errorType>
    <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
    <ax267:ibfsPath/>
    <ax267:id>J06ea07cbj164cj4c0djad92jabc788ea8f76</ax267:id>
    <ax267:jobId>J06ea07cbj164cj4c0djad92jabc788ea8f76</ax267:jobId>
    <ax267:logElementList xsi:nil="true"/>
    <ax267:name xsi:nil="true"/>
    <ax267:owner>admin</ax267:owner>
    <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
    <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
    <ax267:startTime>2014-02-22T21:26:18.515-05:00</ax267:startTime>
    <ax267:summary xsi:nil="true"/>
    </ns:return>

```

```

<ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
  <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
  <ax267:description xsi:nil="true"/>
  <ax267:endTime>2014-02-24T09:00:25.861-05:00</ax267:endTime>
  <ax267:errorType>0</ax267:errorType>
  <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
  <ax267:ibfsPath/>
  <ax267:id>J4ce5d61ejf6b2j441dja02ej084628360372</ax267:id>
  <ax267:jobId>J4ce5d61ejf6b2j441dja02ej084628360372</ax267:jobId>
  <ax267:logElementList xsi:nil="true"/>
  <ax267:name xsi:nil="true"/>
  <ax267:owner>admin</ax267:owner>
  <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
  <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
  <ax267:startTime>2014-02-24T09:00:23.126-05:00</ax267:startTime>
  <ax267:summary xsi:nil="true"/>
</ns:return>
</ns:getLogInfoListByOwnerResponse>

```

The following table lists and describes the possible *errorType* code values that are returned in the XML response document.

errorType Code Value	Description
0	No Error
1	Error
2	Warning
6	Running
7	Running With Error

Retrieving the Log List for an Owner Within a Time Period

This RESTful web service request can be used to retrieve a list of log information for a specific owner that was run between a specific time interval.

If there is no value for the start time of the time interval, then log list information before the end time of the time interval is retrieved.

If there is no value for the end time of the time interval, then log list information after the start time of the time interval is retrieved.

The details for each log are not returned.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/services/LogServiceREST/getLogInfoListByOwnerByCalendar?  
owner=owner&  
startTime=startTime&endTime=endTime
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

owner

Is the owner of the ReportCaster log.

startTime

Is the start time for when the logs are to be retrieved. The following format must be used:

YYYY/MM/DD%20HH:MM:SS

endTime

Is the end time for when the logs are to be retrieved. The following format must be used:

YYYY/MM/DD%20HH:MM:SS

Example:

In the following example, a list of log information for the owner named *admin* between 2014-02-19 00:00:00 and 2014-02-19 23:59:59 is to be retrieved.

Request:

```
http://localhost:8080/ibi_apps/services/LogServiceREST/getLogInfoListByOwnerByCalendar?  
owner=admin&  
startTime=2014/02/19%2000:00:00&endTime=2014/02/19%2023:59:59
```

Response:

```

<ns:getLogInfoListByOwnerByCalendarResponse xmlns:ns="http://
ws.api.broker.ibi" xmlns:ax264="http://io.java/xsd" xmlns:ax263="http://
rmi.java/xsd" xmlns:ax267="http://dslog.data.api.broker.ibi/xsd"
xmlns:ax261="http://schedule.data.api.broker.ibi/xsd">
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
    <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
    <ax267:description xsi:nil="true"/>
    <ax267:endTime>2014-02-19T16:14:42.279-05:00</ax267:endTime>
    <ax267:errorType>2</ax267:errorType>
    <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
    <ax267:ibfsPath/>
    <ax267:id>J73498ee4j33caj409bjacbbj47ab9f66920d</ax267:id>
    <ax267:jobId>J73498ee4j33caj409bjacbbj47ab9f66920d</ax267:jobId>
    <ax267:logElementList xsi:nil="true"/>
    <ax267:name xsi:nil="true"/>
    <ax267:owner>admin</ax267:owner>
    <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
    <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
    <ax267:startTime>2014-02-19T16:14:41.146-05:00</ax267:startTime>
    <ax267:summary xsi:nil="true"/>
  </ns:return>
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
    <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
    <ax267:description xsi:nil="true"/>
    <ax267:endTime>2014-02-19T16:16:22.945-05:00</ax267:endTime>
    <ax267:errorType>2</ax267:errorType>
    <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
    <ax267:ibfsPath/>
    <ax267:id>J5004dec7j6c9cj4009jab87j97b8fa04831a</ax267:id>
    <ax267:jobId>J5004dec7j6c9cj4009jab87j97b8fa04831a</ax267:jobId>

```

```

        <ax267:logElementList xsi:nil="true"/>
        <ax267:name xsi:nil="true"/>
        <ax267:owner>admin</ax267:owner>
        <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
        <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
        <ax267:startTime>2014-02-19T16:16:22.298-05:00</ax267:startTime>
        <ax267:summary xsi:nil="true"/>
    </ns:return>
    <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
        <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
        <ax267:description xsi:nil="true"/>
        <ax267:endTime>2014-02-19T16:19:08.674-05:00</ax267:endTime>
        <ax267:errorType>0</ax267:errorType>
        <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
        <ax267:ibfsPath/>
        <ax267:id>J0c6828cfj96f0j4363ja81ejd41e3782cff2</ax267:id>
        <ax267:jobId>J0c6828cfj96f0j4363ja81ejd41e3782cff2</ax267:jobId>
        <ax267:logElementList xsi:nil="true"/>
        <ax267:name xsi:nil="true"/>
        <ax267:owner>admin</ax267:owner>
        <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
        <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
        <ax267:startTime>2014-02-19T16:19:08.061-05:00</ax267:startTime>
        <ax267:summary xsi:nil="true"/>
    </ns:return>
    <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
        <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
        <ax267:description xsi:nil="true"/>
        <ax267:endTime>2014-02-19T16:22:16.729-05:00</ax267:endTime>
        <ax267:errorType>0</ax267:errorType>
        <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
        <ax267:ibfsPath/>
        <ax267:id>Jd5ae6d5cj3283j4bc2ja36ejc29fd6895419</ax267:id>
        <ax267:jobId>Jd5ae6d5cj3283j4bc2ja36ejc29fd6895419</ax267:jobId>
        <ax267:logElementList xsi:nil="true"/>
        <ax267:name xsi:nil="true"/>
        <ax267:owner>admin</ax267:owner>
        <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
        <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
        <ax267:startTime>2014-02-19T16:22:16.030-05:00</ax267:startTime>
        <ax267:summary xsi:nil="true"/>
    </ns:return>

```

```

    <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
    <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
    <ax267:description xsi:nil="true"/>
    <ax267:endTime>2014-02-19T16:29:31.724-05:00</ax267:endTime>
    <ax267:errorType>0</ax267:errorType>
    <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
    <ax267:ibfsPath/>
    <ax267:id>Jbcdb8429jab89j4ed8j95d2j25dd04279f5c</ax267:id>
    <ax267:jobId>Jbcdb8429jab89j4ed8j95d2j25dd04279f5c</ax267:jobId>
    <ax267:logElementList xsi:nil="true"/>
    <ax267:name xsi:nil="true"/>
    <ax267:owner>admin</ax267:owner>
    <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
    <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
    <ax267:startTime>2014-02-19T16:29:31.083-05:00</ax267:startTime>
    <ax267:summary xsi:nil="true"/>
    </ns:return>
</ns:getLogInfoListByOwnerByCalendarResponse>

```

The following table lists and describes the possible *errorType* code values that are returned in the XML response document.

errorType Code Value	Description
0	No Error
1	Error
2	Warning
6	Running
7	Running With Error

Retrieving the Log List for a Schedule

This RESTful web service request can be used to retrieve a list of log information for a specific schedule identified by the schedule ID. The details for each log are not returned.

HTTP Method: GET

REST URL Format:

http://host:port/ibi_apps/services/LogServiceREST/getLogInfoListByScheduleId?scheduleId=scheduleId

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

scheduleId

Is the schedule ID for the ReportCaster schedule.

Example:

In the following example, a list of log information for schedule ID Sca76e628s892as43a4sbddcs10875ff7f188 is to be retrieved.

Request:

```
http://localhost:8080/ibi_apps/services/LogServiceREST/getLogInfoListByScheduleId?
scheduleId=Sca76e628s892as43a4sbddcs10875ff7f188
```

Response:

```

<ns:getLogInfoListByScheduleIdResponse xmlns:ns="http://ws.api.broker.ibi"
xmlns:ax264="http://io.java/xsd" xmlns:ax263="http://rmi.java/xsd"
xmlns:ax267="http://dslog.data.api.broker.ibi/xsd" xmlns:ax261="http://
schedule.data.api.broker.ibi/xsd">
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
    <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
    <ax267:description xsi:nil="true"/>
    <ax267:endTime>2014-02-19T16:14:42.279-05:00</ax267:endTime>
    <ax267:errorType>2</ax267:errorType>
    <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
    <ax267:ibfsPath/>
    <ax267:id>J73498ee4j33caj409bjacbbj47ab9f66920d</ax267:id>
    <ax267:jobId>J73498ee4j33caj409bjacbbj47ab9f66920d</ax267:jobId>
    <ax267:logElementList xsi:nil="true"/>
    <ax267:name xsi:nil="true"/>
    <ax267:owner>admin</ax267:owner>
    <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
    <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
    <ax267:startTime>2014-02-19T16:14:41.146-05:00</ax267:startTime>
    <ax267:summary xsi:nil="true"/>
  </ns:return>
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
    <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
    <ax267:description xsi:nil="true"/>
    <ax267:endTime>2014-02-19T16:16:22.945-05:00</ax267:endTime>
    <ax267:errorType>2</ax267:errorType>
    <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
    <ax267:ibfsPath/>
    <ax267:id>J5004dec7j6c9cj4009jab87j97b8fa04831a</ax267:id>
    <ax267:jobId>J5004dec7j6c9cj4009jab87j97b8fa04831a</ax267:jobId>
    <ax267:logElementList xsi:nil="true"/>
    <ax267:name xsi:nil="true"/>
    <ax267:owner>admin</ax267:owner>
    <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>

```



```

<ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:scheduleId>
  <ax267:startTime>2014-02-19T16:16:22.298-05:00</ax267:startTime>
  <ax267:summary xsi:nil="true" />
</ns:return>
<ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
  <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
  <ax267:description xsi:nil="true" />
  <ax267:endTime>2014-02-19T16:19:08.674-05:00</ax267:endTime>
  <ax267:errorType>0</ax267:errorType>
  <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
  <ax267:ibfsPath/>
  <ax267:id>J0c6828cfj96f0j4363ja81ejd41e3782cff2</ax267:id>
  <ax267:jobId>J0c6828cfj96f0j4363ja81ejd41e3782cff2</ax267:jobId>
  <ax267:logElementList xsi:nil="true" />
  <ax267:name xsi:nil="true" />
  <ax267:owner>admin</ax267:owner>
  <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
  <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
  <ax267:startTime>2014-02-19T16:19:08.061-05:00</ax267:startTime>
  <ax267:summary xsi:nil="true" />
</ns:return>
<ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
  <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
  <ax267:description xsi:nil="true" />
  <ax267:endTime>2014-02-19T16:22:16.729-05:00</ax267:endTime>
  <ax267:errorType>0</ax267:errorType>
  <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
  <ax267:ibfsPath/>
  <ax267:id>Jd5ae6d5cj3283j4bc2ja36ejc29fd6895419</ax267:id>
  <ax267:jobId>Jd5ae6d5cj3283j4bc2ja36ejc29fd6895419</ax267:jobId>
  <ax267:logElementList xsi:nil="true" />
  <ax267:name xsi:nil="true" />
  <ax267:owner>admin</ax267:owner>
  <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
  <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
  <ax267:startTime>2014-02-19T16:22:16.030-05:00</ax267:startTime>
  <ax267:summary xsi:nil="true" />
</ns:return>

```

```

<ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
  <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
  <ax267:description xsi:nil="true"/>
  <ax267:endTime>2014-02-19T16:29:31.724-05:00</ax267:endTime>
  <ax267:errorType>0</ax267:errorType>
  <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
  <ax267:ibfsPath/>
  <ax267:id>Jbcdb8429jab89j4ed8j95d2j25dd04279f5c</ax267:id>
  <ax267:jobId>Jbcdb8429jab89j4ed8j95d2j25dd04279f5c</ax267:jobId>
  <ax267:logElementList xsi:nil="true"/>
  <ax267:name xsi:nil="true"/>
  <ax267:owner>admin</ax267:owner>
  <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
  <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
  <ax267:startTime>2014-02-19T16:29:31.083-05:00</ax267:startTime>
  <ax267:summary xsi:nil="true"/>
</ns:return>
<ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
  <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
  <ax267:description xsi:nil="true"/>
  <ax267:endTime>2014-02-22T21:26:23.251-05:00</ax267:endTime>
  <ax267:errorType>0</ax267:errorType>
  <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
  <ax267:ibfsPath/>
  <ax267:id>J06ea07cbj164cj4c0djad92jabc788ea8f76</ax267:id>
  <ax267:jobId>J06ea07cbj164cj4c0djad92jabc788ea8f76</ax267:jobId>
  <ax267:logElementList xsi:nil="true"/>
  <ax267:name xsi:nil="true"/>
  <ax267:owner>admin</ax267:owner>
  <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
  <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
  <ax267:startTime>2014-02-22T21:26:18.515-05:00</ax267:startTime>
  <ax267:summary xsi:nil="true"/>
</ns:return>

```

```

<ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax267:DsLog">
  <ax267:IBFSObjectType>0</ax267:IBFSObjectType>
  <ax267:description xsi:nil="true"/>
  <ax267:endTime>2014-02-24T09:00:25.861-05:00</ax267:endTime>
  <ax267:errorType>0</ax267:errorType>
  <ax267:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</ax267:ibfsId>
  <ax267:ibfsPath/>
  <ax267:id>J4ce5d61ejf6b2j441dja02ej084628360372</ax267:id>
  <ax267:jobId>J4ce5d61ejf6b2j441dja02ej084628360372</ax267:jobId>
  <ax267:logElementList xsi:nil="true"/>
  <ax267:name xsi:nil="true"/>
  <ax267:owner>admin</ax267:owner>
  <ax267:scheduleDescription>Carinst Report</
ax267:scheduleDescription>
  <ax267:scheduleId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax267:scheduleId>
  <ax267:startTime>2014-02-24T09:00:23.126-05:00</ax267:startTime>
  <ax267:summary xsi:nil="true"/>
</ns:return>
</ns:getLogInfoListByScheduleIdResponse>

```

The following table lists and describes the possible *errorType* code values that are returned in the XML response document.

errorType Code Value	Description
0	No Error
1	Error
2	Warning
6	Running
7	Running With Error

Retrieving a List of Schedule Owners

This RESTful web service request can be used to retrieve a list of schedule owners.

HTTP Method: GET

REST URL Format:

http://host:port/ibi_apps/services/LogServiceREST/getOwnerList

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Example:

In the following example, a list of schedule owners is to be retrieved.

Request:

`http://localhost:8080/ibi_apps/services/LogServiceREST/getOwnerList`

Response:

```
<ns:getOwnerListResponse xmlns:ns="http://ws.api.broker.ibi"
xmlns:ax264="http://io.java/xsd" xmlns:ax263="http://rmi.java/xsd"
xmlns:ax267="http://dslog.data.api.broker.ibi/xsd" xmlns:ax261="http://
schedule.data.api.broker.ibi/xsd">
  <ns:return>admin</ns:return>
  <ns:return>system</ns:return>
</ns:getOwnerListResponse>
```

The value within each <return> element indicates a schedule owner.

Console Functionality

This section describes the format and structure of RESTful web service requests that are used for a variety of ReportCaster Console functionality.

Changing Job Priority

This RESTful web service request can be used to change the priority of a job awaiting execution in the job queue.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/services/ConsoleServiceREST/changeJobPriority?
jobId=jobId&priority=priority`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

jobId

Is a unique identifier for the ReportCaster job.

priority

The priority of a job awaiting execution in the job queue. A value of 1 is the highest and a value of 5 is the lowest.

Example:

In the following example, the priority for the ReportCaster job identified by job ID Jc12b4443jb1f8j4c19j90aaj7ba31ac4dbf5 is changed to 1.

Request:

```
http://localhost:8080/ibi_apps/services/ConsoleServiceREST/changeJobPriority?
jobId=Jc12b4443jb1f8j4c19j90aaj7ba31ac4dbf5&priority=1
```

Response:

```
<ns:changeJobPriorityResponse xmlns:ns="http://ws.api.broker.ibi">
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:nil="true"/>
</ns:changeJobPriorityResponse>
```

Retrieving Job Status

This RESTful web service request can be used to retrieve the status of a current ReportCaster job in the queue.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/services/ConsoleServiceREST/getJobStatus?jobId=jobId
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

jobId

Is a unique identifier for the ReportCaster job.

Example:

In the following example, the status of the ReportCaster job identified by job ID Jc12b4443jb1f8j4c19j90aaj7ba31ac4dbf5 is returned.

Request:

```
http://localhost:8080/ibi_apps/services/ConsoleServiceREST/getJobStatus?jobId=Jc12b4443jb1f8j4c19j90aaj7ba31ac4dbf5
```

Response:

```
<ns:getJobStatusResponse xmlns:ns="http://ws.api.broker.ibi">
  <ns:return>1</ns:return>
</ns:getJobStatusResponse>
```

The following table lists and describes the ReportCaster job status return codes.

getJobStatus Return Code	Description
-1	The specified ReportCaster job identified by <i>jobId</i> does not exist.
0	The specified ReportCaster job identified by <i>jobId</i> exists, but the calling user is not authorized to view the status of this job.
1	The specified ReportCaster job identified by <i>jobId</i> is in the waiting queue.
2	The specified ReportCaster job identified by <i>jobId</i> is running.
3	The specified ReportCaster job identified by <i>jobId</i> has completed.

Listing Jobs in the Queue

This RESTful web service request can be used to list the ReportCaster jobs awaiting execution in the job queue.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/services/ConsoleServiceREST/getJobsInQueue
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Example:

In the following example, a list of ReportCaster jobs awaiting execution in the job queue is retrieved.

Request:

```
http://localhost:8080/ibi_apps/services/ConsoleServiceREST/getJobsInQueue
```

Response:

```

<ns:getJobsInQueueResponse xmlns:ns="http://ws.api.broker.ibi"
xmlns:ax220="http://rmi.java.xsd" xmlns:ax221="http://io.java.xsd"
xmlns:ax224="http://schedule.data.api.broker.ibi/xsd" xmlns:ax226="http://
console.data.api.broker.ibi/xsd">
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax226:Job">
    <ax226:distributionServerName xsi:nil="true"/>
    <ax226:fullyQualifiedServerName xsi:nil="true"/>
    <ax226:id>J6d1eb46fj9a2dj46e0jb532j711fa60ec7e1</ax226:id>
    <ax226:schedule xsi:type="ax224:Schedule">
      <ax224:IBFSObjectType>113</ax224:IBFSObjectType>
      <ax224:active>true</ax224:active>
      <ax224:compressedReport>false</ax224:compressedReport>
      <ax224:deleteJobAfterRun>false</ax224:deleteJobAfterRun>
      <ax224:description>Carinst Report 2</ax224:description>
      <ax224:distribution xsi:type="ax224:DistributionEmail">
        <ax224:description/>
        <ax224:disabled>false</ax224:disabled>
        <ax224:id/>
        <ax224:type>EMAIL</ax224:type>
        <ax224:authEnabled>false</ax224:authEnabled>
        <ax224:authPassword/>
        <ax224:authUserId/>
        <ax224:destination xsi:type="ax224:Destination">
          <ax224:distributionFile/>
          <ax224:distributionList/>
          <ax224:distributionListFullPath/>
          <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
            <ax224:password/>
            <ax224:procedureName/>
            <ax224:serverName/>
            <ax224:userName/>
          </ax224:dynamicAddress>
          <ax224:singleAddress/>
          <ax224:type>DISTRIBUTION_LIST</ax224:type>
        </ax224:destination>
        <ax224:inlineMessage/>
        <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
        <ax224:mailFrom/>
        <ax224:mailReplyAddress/>
        <ax224:mailServerName/>
        <ax224:mailSubject/>
      </ax224:distribution>
    </ax226:schedule>
  </ns:return>
</ns:getJobsInQueueResponse>

```



```

<ax224:sendingReportAsAttachment>true</ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distribution>
<ax224:distributionList xsi:type="ax224:DistributionEmail">
  <ax224:description/>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:type>EMAIL</ax224:type>
  <ax224:authEnabled>>false</ax224:authEnabled>
  <ax224:authPassword/>
  <ax224:authUserId/>
  <ax224:destination xsi:type="ax224:Destination">
    <ax224:distributionFile/>
    <ax224:distributionList/>
    <ax224:distributionListFullPath/>
    <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
      <ax224:password/>
      <ax224:procedureName/>
      <ax224:serverName/>
      <ax224:userName/>
    </ax224:dynamicAddress>
    <ax224:singleAddress/>
    <ax224:type>DISTRIBUTION_LIST</ax224:type>
  </ax224:destination>
  <ax224:inlineMessage/>
  <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
  <ax224:mailFrom/>
  <ax224:mailReplyAddress/>
  <ax224:mailServerName/>
  <ax224:mailSubject/>
  <ax224:sendingReportAsAttachment>true</
ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distributionList>

```

```

<ax224:firstTask xsi:type="ax224:TaskWFServerProcedure">
  <ax224:description>WebFocus Server Procedure task</
ax224:description>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:procedureId/>
  <ax224:procedureName/>
  <ax224:reportName/>
  <ax224:taskRetry xsi:nil="true"/>
  <ax224:type>0</ax224:type>
  <ax224:allowFormatList xsi:nil="true"/>
  <ax224:burst>>false</ax224:burst>
  <ax224:execId/>
  <ax224:execPassword/>
  <ax224:firstPostProcessingProcedure/>
  <ax224:firstPreProcessingProcedure/>
  <ax224:formatInFex>>false</ax224:formatInFex>
  <ax224:parameterList xsi:nil="true"/>
  <ax224:secondPostProcessingProcedure/>
  <ax224:secondPreProcessingProcedure/>
  <ax224:sendFormat>HTML</ax224:sendFormat>
  <ax224:serverName>EDASERVE</ax224:serverName>
</ax224:firstTask>
<ax224:ibfsId>6dff2b49I8245I4638I9e9fIc5900a9a12d5</
ax224:ibfsId>
  <ax224:ibfsPath/>
  <ax224:id>S23f65030s728as482asa632s879fd9f6a727</ax224:id>
  <ax224:lastModified>2014-03-11T18:26:05.912-04:00</
ax224:lastModified>
  <ax224:lastTimeExecuted>1970-01-01T00:00:00.000-05:00</
ax224:lastTimeExecuted>
  <ax224:name/>
  <ax224:notification xsi:type="ax224:Notification">
    <ax224:addressForBriefNotification/>
    <ax224:addressForFullNotification/>
    <ax224:description/>
    <ax224:from/>
    <ax224:id/>
    <ax224:subject/>
    <ax224:type>INACTIVE</ax224:type>
  </ax224:notification>

```

```

<ax224:owner>admin</ax224:owner>
  <ax224:priority>3</ax224:priority>
  <ax224:statusLastExecuted/>
  <ax224:summary/>
  <ax224:taskList xsi:type="ax224:TaskWFServerProcedure">
    <ax224:description>WebFocus Server Procedure task</
ax224:description>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:procedureId/>
    <ax224:procedureName/>
    <ax224:reportName/>
    <ax224:taskRetry xsi:nil="true"/>
    <ax224:type>0</ax224:type>
    <ax224:allowFormatList xsi:nil="true"/>
    <ax224:burst>>false</ax224:burst>
    <ax224:execId/>
    <ax224:execPassword/>
    <ax224:firstPostProcessingProcedure/>
    <ax224:firstPreProcessingProcedure/>
    <ax224:formatInFex>>false</ax224:formatInFex>
    <ax224:parameterList xsi:nil="true"/>
    <ax224:secondPostProcessingProcedure/>
    <ax224:secondPreProcessingProcedure/>
    <ax224:sendFormat>HTML</ax224:sendFormat>
    <ax224:serverName>EDASERVE</ax224:serverName>
  </ax224:taskList>
  <ax224:timeInfo xsi:type="ax224:TimeInfoOnce">
    <ax224:description/>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:name/>
    <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
    <ax224:startTime>2014-03-11T18:26:05.912-04:00</
ax224:startTime>
    <ax224:type>0</ax224:type>
  </ax224:timeInfo>
  <ax224:timeInfoList xsi:type="ax224:TimeInfoOnce">
    <ax224:description/>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:name/>
    <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
    <ax224:startTime>2014-03-11T18:26:05.912-04:00</
ax224:startTime>
    <ax224:type>0</ax224:type>
  </ax224:timeInfoList>

```

```
<ax224:traceType>0</ax224:traceType>
  </ax226:schedule>
  <ax226:startTime>2014-03-11T18:25:53.738-04:00</ax226:startTime>
  <ax226:status>0</ax226:status>
</ns:return>
</ns:getJobsInQueueResponse>
```

Listing Jobs in the Queue for an Owner

This RESTful web service request can be used to list the ReportCaster jobs awaiting execution in the job queue for a specific schedule owner.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/services/ConsoleServiceREST/getJobsInQueueByOwner?owner=owner
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

owner

Is the owner of the ReportCaster schedule.

Example:

In the following example, a list of ReportCaster jobs awaiting execution in the job queue for the owner named *admin* is retrieved.

Request:

```
http://localhost:8080/ibi_apps/services/ConsoleServiceREST/getJobsInQueueByOwner?
owner=admin
```

Response:

```

<ns:getJobsInQueueByOwnerResponse xmlns:ns="http://ws.api.broker.ibi"
xmlns:ax220="http://rmi.java/xsd" xmlns:ax221="http://io.java/xsd"
xmlns:ax224="http://schedule.data.api.broker.ibi/xsd" xmlns:ax226="http://
console.data.api.broker.ibi/xsd">
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax226:Job">
    <ax226:distributionServerName xsi:nil="true"/>
    <ax226:fullyQualifiedServerName xsi:nil="true"/>
    <ax226:id>Jc12b4443jblf8j4c19j90aa7ba31ac4dbf5</ax226:id>
    <ax226:schedule xsi:type="ax224:Schedule">
      <ax224:IBFSObjectType>113</ax224:IBFSObjectType>
      <ax224:active>true</ax224:active>
      <ax224:compressedReport>false</ax224:compressedReport>
      <ax224:deleteJobAfterRun>false</ax224:deleteJobAfterRun>
      <ax224:description>Carinst Report</ax224:description>
      <ax224:distribution xsi:type="ax224:DistributionEmail">
        <ax224:description/>
        <ax224:disabled>false</ax224:disabled>
        <ax224:id/>
        <ax224:type>EMAIL</ax224:type>
        <ax224:authEnabled>false</ax224:authEnabled>
        <ax224:authPassword/>
        <ax224:authUserId/>
        <ax224:destination xsi:type="ax224:Destination">
          <ax224:distributionFile/>
          <ax224:distributionList/>
          <ax224:distributionListFullPath/>
          <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
            <ax224:password/>
            <ax224:procedureName/>
            <ax224:serverName/>
            <ax224:userName/>
          </ax224:dynamicAddress>
          <ax224:singleAddress/>
          <ax224:type>DISTRIBUTION_LIST</ax224:type>
        </ax224:destination>
        <ax224:inlineMessage/>
        <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
        <ax224:mailFrom/>
        <ax224:mailReplyAddress/>
        <ax224:mailServerName/>
        <ax224:mailSubject/>
      </ax224:distribution>
    </ax226:schedule>
  </ns:return>
</ns:getJobsInQueueByOwnerResponse>

```

```

<ax224:sendingReportAsAttachment>>true</ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distribution>
<ax224:distributionList xsi:type="ax224:DistributionEmail">
  <ax224:description/>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:type>EMAIL</ax224:type>
  <ax224:authEnabled>>false</ax224:authEnabled>
  <ax224:authPassword/>
  <ax224:authUserId/>
  <ax224:destination xsi:type="ax224:Destination">
    <ax224:distributionFile/>
    <ax224:distributionList/>
    <ax224:distributionListFullPath/>
    <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
      <ax224:password/>
      <ax224:procedureName/>
      <ax224:serverName/>
      <ax224:userName/>
    </ax224:dynamicAddress>
    <ax224:singleAddress/>
    <ax224:type>DISTRIBUTION_LIST</ax224:type>
  </ax224:destination>
  <ax224:inlineMessage/>
  <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
  <ax224:mailFrom/>
  <ax224:mailReplyAddress/>
  <ax224:mailServerName/>
  <ax224:mailSubject/>
  <ax224:sendingReportAsAttachment>true</
ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distributionList>

```

```

    <ax224:firstTask xsi:type="ax224:TaskWFServerProcedure">
      <ax224:description>WebFocus Server Procedure task</
ax224:description>
      <ax224:disabled>>false</ax224:disabled>
      <ax224:id/>
      <ax224:procedureId/>
      <ax224:procedureName/>
      <ax224:reportName/>
      <ax224:taskRetry xsi:nil="true"/>
      <ax224:type>0</ax224:type>
      <ax224:allowFormatList xsi:nil="true"/>
      <ax224:burst>>false</ax224:burst>
      <ax224:execId/>
      <ax224:execPassword/>
      <ax224:firstPostProcessingProcedure/>
      <ax224:firstPreProcessingProcedure/>
      <ax224:formatInFex>>false</ax224:formatInFex>
      <ax224:parameterList xsi:nil="true"/>
      <ax224:secondPostProcessingProcedure/>
      <ax224:secondPreProcessingProcedure/>
      <ax224:sendFormat>HTML</ax224:sendFormat>
      <ax224:serverName>EDASERVE</ax224:serverName>
    </ax224:firstTask>
    <ax224:ibfsId>Sca76e628s892as43a4sbddcs10875ff7f188</
ax224:ibfsId>
    <ax224:ibfsPath/>
    <ax224:id>Sca76e628s892as43a4sbddcs10875ff7f188</ax224:id>
    <ax224:lastModified>2014-02-24T14:25:09.581-05:00</
ax224:lastModified>
    <ax224:lastTimeExecuted>1970-01-01T00:00:00.000-05:00</
ax224:lastTimeExecuted>
    <ax224:name/>
    <ax224:notification xsi:type="ax224:Notification">
      <ax224:addressForBriefNotification/>
      <ax224:addressForFullNotification/>
      <ax224:description/>
      <ax224:from/>
      <ax224:id/>
      <ax224:subject/>
      <ax224:type>INACTIVE</ax224:type>
    </ax224:notification>

```

```

<ax224:owner>admin</ax224:owner>
  <ax224:priority>3</ax224:priority>
  <ax224:statusLastExecuted/>
  <ax224:summary/>
  <ax224:taskList xsi:type="ax224:TaskWFServerProcedure">
    <ax224:description>WebFocus Server Procedure task</
ax224:description>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:procedureId/>
    <ax224:procedureName/>
    <ax224:reportName/>
    <ax224:taskRetry xsi:nil="true"/>
    <ax224:type>0</ax224:type>
    <ax224:allowFormatList xsi:nil="true"/>
    <ax224:burst>>false</ax224:burst>
    <ax224:execId/>
    <ax224:execPassword/>
    <ax224:firstPostProcessingProcedure/>
    <ax224:firstPreProcessingProcedure/>
    <ax224:formatInFex>>false</ax224:formatInFex>
    <ax224:parameterList xsi:nil="true"/>
    <ax224:secondPostProcessingProcedure/>
    <ax224:secondPreProcessingProcedure/>
    <ax224:sendFormat>HTML</ax224:sendFormat>
    <ax224:serverName>EDASERVE</ax224:serverName>
  </ax224:taskList>
  <ax224:timeInfo xsi:type="ax224:TimeInfoOnce">
    <ax224:description/>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:name/>
    <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
    <ax224:startTime>2014-02-24T14:25:09.581-05:00</
ax224:startTime>
    <ax224:type>0</ax224:type>
  </ax224:timeInfo>
  <ax224:timeInfoList xsi:type="ax224:TimeInfoOnce">
    <ax224:description/>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:name/>
    <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
    <ax224:startTime>2014-02-24T14:25:09.581-05:00</
ax224:startTime>
    <ax224:type>0</ax224:type>
  </ax224:timeInfoList>

```



```
<ax224:traceType>0</ax224:traceType>
  </ax226:schedule>
  <ax226:startTime>2014-02-24T14:13:25.760-05:00</ax226:startTime>
  <ax226:status>0</ax226:status>
</ns:return>
</ns:getJobsInQueueByOwnerResponse>
```

Listing Running Jobs

This RESTful web service request can be used to list the ReportCaster jobs that are running.

HTTP Method: GET

REST URL Format:

```
http://host:port/ibi_apps/services/ConsoleServiceREST/getRunningJobs
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

Example:

In the following example, a list of ReportCaster jobs that are running is retrieved.

Request:

```
http://localhost:8080/ibi_apps/services/ConsoleServiceREST/getRunningJobs
```

Response:

```

<ns:getRunningJobsResponse xmlns:ns="http://ws.api.broker.ibi"
xmlns:ax220="http://rmi.java.xsd" xmlns:ax221="http://io.java.xsd"
xmlns:ax224="http://schedule.data.api.broker.ibi/xsd" xmlns:ax226="http://
console.data.api.broker.ibi/xsd">
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax226:Job">
    <ax226:distributionServerName xsi:nil="true"/>
    <ax226:fullyQualifiedServerName xsi:nil="true"/>
    <ax226:id>J41d8861bj53f5j4a60j8568jeddf39416a88</ax226:id>
    <ax226:schedule xsi:type="ax224:Schedule">
      <ax224:IBFSObjectType>113</ax224:IBFSObjectType>
      <ax224:active>true</ax224:active>
      <ax224:compressedReport>false</ax224:compressedReport>
      <ax224:deleteJobAfterRun>false</ax224:deleteJobAfterRun>
      <ax224:description>Carinst Report 2</ax224:description>
      <ax224:distribution xsi:type="ax224:DistributionEmail">
        <ax224:description/>
        <ax224:disabled>false</ax224:disabled>
        <ax224:id/>
        <ax224:type>EMAIL</ax224:type>
        <ax224:authEnabled>false</ax224:authEnabled>
        <ax224:authPassword/>
        <ax224:authUserId/>
        <ax224:destination xsi:type="ax224:Destination">
          <ax224:distributionFile/>
          <ax224:distributionList/>
          <ax224:distributionListFullPath/>
          <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
            <ax224:password/>
            <ax224:procedureName/>
            <ax224:serverName/>
            <ax224:userName/>
          </ax224:dynamicAddress>
          <ax224:singleAddress/>
          <ax224:type>DISTRIBUTION_LIST</ax224:type>
        </ax224:destination>
        <ax224:inlineMessage/>
        <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
        <ax224:mailFrom/>
        <ax224:mailReplyAddress/>
        <ax224:mailServerName/>
        <ax224:mailSubject/>
      </ax224:distribution>
    </ax226:schedule>
  </ns:return>
</ns:getRunningJobsResponse>

```

```

<ax224:sendingReportAsAttachment>true</ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distribution>
<ax224:distributionList xsi:type="ax224:DistributionEmail">
  <ax224:description/>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:type>EMAIL</ax224:type>
  <ax224:authEnabled>>false</ax224:authEnabled>
  <ax224:authPassword/>
  <ax224:authUserId/>
  <ax224:destination xsi:type="ax224:Destination">
    <ax224:distributionFile/>
    <ax224:distributionList/>
    <ax224:distributionListFullPath/>
    <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
      <ax224:password/>
      <ax224:procedureName/>
      <ax224:serverName/>
      <ax224:userName/>
    </ax224:dynamicAddress>
    <ax224:singleAddress/>
    <ax224:type>DISTRIBUTION_LIST</ax224:type>
  </ax224:destination>
  <ax224:inlineMessage/>
  <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
  <ax224:mailFrom/>
  <ax224:mailReplyAddress/>
  <ax224:mailServerName/>
  <ax224:mailSubject/>
  <ax224:sendingReportAsAttachment>true</
ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distributionList>

```

```

    <ax224:firstTask xsi:type="ax224:TaskWFServerProcedure">
      <ax224:description>WebFocus Server Procedure task</
ax224:description>
      <ax224:disabled>>false</ax224:disabled>
      <ax224:id/>
      <ax224:procedureId/>
      <ax224:procedureName/>
      <ax224:reportName/>
      <ax224:taskRetry xsi:nil="true"/>
      <ax224:type>0</ax224:type>
      <ax224:allowFormatList xsi:nil="true"/>
      <ax224:burst>>false</ax224:burst>
      <ax224:execId/>
      <ax224:execPassword/>
      <ax224:firstPostProcessingProcedure/>
      <ax224:firstPreProcessingProcedure/>
      <ax224:formatInFex>>false</ax224:formatInFex>
      <ax224:parameterList xsi:nil="true"/>
      <ax224:secondPostProcessingProcedure/>
      <ax224:secondPreProcessingProcedure/>
      <ax224:sendFormat>HTML</ax224:sendFormat>
      <ax224:serverName>EDASERVE</ax224:serverName>
    </ax224:firstTask>
    <ax224:ibfsId>6dff2b49I8245I4638I9e9fIc5900a9a12d5</
ax224:ibfsId>
    <ax224:ibfsPath/>
    <ax224:id>S23f65030s728as482asa632s879fd9f6a727</ax224:id>
    <ax224:lastModified>2014-03-11T18:27:46.710-04:00</
ax224:lastModified>
    <ax224:lastTimeExecuted>1970-01-01T00:00:00.000-05:00</
ax224:lastTimeExecuted>
    <ax224:name/>
    <ax224:notification xsi:type="ax224:Notification">
      <ax224:addressForBriefNotification/>
      <ax224:addressForFullNotification/>
      <ax224:description/>
      <ax224:from/>
      <ax224:id/>
      <ax224:subject/>
      <ax224:type>INACTIVE</ax224:type>
    </ax224:notification>

```

```

<ax224:owner>admin</ax224:owner>
  <ax224:priority>3</ax224:priority>
  <ax224:statusLastExecuted/>
  <ax224:summary/>
  <ax224:taskList xsi:type="ax224:TaskWFServerProcedure">
    <ax224:description>WebFocus Server Procedure task</
ax224:description>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:procedureId/>
    <ax224:procedureName/>
    <ax224:reportName/>
    <ax224:taskRetry xsi:nil="true"/>
    <ax224:type>0</ax224:type>
    <ax224:allowFormatList xsi:nil="true"/>
    <ax224:burst>>false</ax224:burst>
    <ax224:execId/>
    <ax224:execPassword/>
    <ax224:firstPostProcessingProcedure/>
    <ax224:firstPreProcessingProcedure/>
    <ax224:formatInFex>>false</ax224:formatInFex>
    <ax224:parameterList xsi:nil="true"/>
    <ax224:secondPostProcessingProcedure/>
    <ax224:secondPreProcessingProcedure/>
    <ax224:sendFormat>HTML</ax224:sendFormat>
    <ax224:serverName>EDASERVE</ax224:serverName>
  </ax224:taskList>
  <ax224:timeInfo xsi:type="ax224:TimeInfoOnce">
    <ax224:description/>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:name/>
    <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
    <ax224:startTime>2014-03-11T18:27:46.710-04:00</
ax224:startTime>
    <ax224:type>0</ax224:type>
  </ax224:timeInfo>
  <ax224:timeInfoList xsi:type="ax224:TimeInfoOnce">
    <ax224:description/>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:name/>
    <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
    <ax224:startTime>2014-03-11T18:27:46.710-04:00</
ax224:startTime>
    <ax224:type>0</ax224:type>
  </ax224:timeInfoList>

```

```

<ax224:traceType>0</ax224:traceType>
  </ax226:schedule>
  <ax226:startTime>2014-03-11T18:25:35.827-04:00</ax226:startTime>
  <ax226:status>1</ax226:status>
</ns:return>
<ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax226:Job">
  <ax226:distributionServerName xsi:nil="true"/>
  <ax226:fullyQualifiedServerName xsi:nil="true"/>
  <ax226:id>Jbc069445jb73dj4b41j9051j5dd542f3074e</ax226:id>
  <ax226:schedule xsi:type="ax224:Schedule">
    <ax224:IBFSObjectType>113</ax224:IBFSObjectType>
    <ax224:active>true</ax224:active>
    <ax224:compressedReport>false</ax224:compressedReport>
    <ax224:deleteJobAfterRun>false</ax224:deleteJobAfterRun>
    <ax224:description>Carinst Report 2</ax224:description>
    <ax224:distribution xsi:type="ax224:DistributionEmail">
      <ax224:description/>
      <ax224:disabled>false</ax224:disabled>
      <ax224:id/>
      <ax224:type>EMAIL</ax224:type>
      <ax224:authEnabled>false</ax224:authEnabled>
      <ax224:authPassword/>
      <ax224:authUserId/>
      <ax224:destination xsi:type="ax224:Destination">
        <ax224:distributionFile/>
        <ax224:distributionList/>
        <ax224:distributionListFullPath/>
        <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
          <ax224:password/>
          <ax224:procedureName/>
          <ax224:serverName/>
          <ax224:userName/>
        </ax224:dynamicAddress>
        <ax224:singleAddress/>
        <ax224:type>DISTRIBUTION_LIST</ax224:type>
      </ax224:destination>
    </ax224:distribution>
  </ax226:schedule>
</ns:return>

```

```

<ax224:inlineMessage/>
  <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
  <ax224:mailFrom/>
  <ax224:mailReplyAddress/>
  <ax224:mailServerName/>
  <ax224:mailSubject/>
  <ax224:sendingReportAsAttachment>true</
ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distribution>
<ax224:distributionList xsi:type="ax224:DistributionEmail">
  <ax224:description/>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:type>EMAIL</ax224:type>
  <ax224:authEnabled>>false</ax224:authEnabled>
  <ax224:authPassword/>
  <ax224:authUserId/>
  <ax224:destination xsi:type="ax224:Destination">
    <ax224:distributionFile/>
    <ax224:distributionList/>
    <ax224:distributionListFullPath/>
    <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
      <ax224:password/>
      <ax224:procedureName/>
      <ax224:serverName/>
      <ax224:userName/>
    </ax224:dynamicAddress>
    <ax224:singleAddress/>
    <ax224:type>DISTRIBUTION_LIST</ax224:type>
  </ax224:destination>
  <ax224:inlineMessage/>
  <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
  <ax224:mailFrom/>
  <ax224:mailReplyAddress/>
  <ax224:mailServerName/>
  <ax224:mailSubject/>
  <ax224:sendingReportAsAttachment>true</
ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distributionList>

```

```

<ax224:firstTask xsi:type="ax224:TaskWFServerProcedure">
  <ax224:description>WebFocus Server Procedure task</
ax224:description>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:procedureId/>
  <ax224:procedureName/>
  <ax224:reportName/>
  <ax224:taskRetry xsi:nil="true"/>
  <ax224:type>0</ax224:type>
  <ax224:allowFormatList xsi:nil="true"/>
  <ax224:burst>>false</ax224:burst>
  <ax224:execId/>
  <ax224:execPassword/>
  <ax224:firstPostProcessingProcedure/>
  <ax224:firstPreProcessingProcedure/>
  <ax224:formatInFex>>false</ax224:formatInFex>
  <ax224:parameterList xsi:nil="true"/>
  <ax224:secondPostProcessingProcedure/>
  <ax224:secondPreProcessingProcedure/>
  <ax224:sendFormat>HTML</ax224:sendFormat>
  <ax224:serverName>EDASERVE</ax224:serverName>
</ax224:firstTask>
<ax224:ibfsId>6dff2b49I8245I4638I9e9fIc5900a9a12d5</
ax224:ibfsId>
  <ax224:ibfsPath/>
  <ax224:id>S23f65030s728as482asa632s879fd9f6a727</ax224:id>
  <ax224:lastModified>2014-03-11T18:27:46.710-04:00</
ax224:lastModified>
  <ax224:lastTimeExecuted>1970-01-01T00:00:00.000-05:00</
ax224:lastTimeExecuted>
  <ax224:name/>
  <ax224:notification xsi:type="ax224:Notification">
    <ax224:addressForBriefNotification/>
    <ax224:addressForFullNotification/>
    <ax224:description/>
    <ax224:from/>
    <ax224:id/>
    <ax224:subject/>
    <ax224:type>INACTIVE</ax224:type>
  </ax224:notification>

```



```

<ax224:owner>admin</ax224:owner>
  <ax224:priority>3</ax224:priority>
  <ax224:statusLastExecuted/>
  <ax224:summary/>
  <ax224:taskList xsi:type="ax224:TaskWFServerProcedure">
    <ax224:description>WebFocus Server Procedure task</
ax224:description>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:procedureId/>
    <ax224:procedureName/>
    <ax224:reportName/>
    <ax224:taskRetry xsi:nil="true"/>
    <ax224:type>0</ax224:type>
    <ax224:allowFormatList xsi:nil="true"/>
    <ax224:burst>>false</ax224:burst>
    <ax224:execId/>
    <ax224:execPassword/>
    <ax224:firstPostProcessingProcedure/>
    <ax224:firstPreProcessingProcedure/>
    <ax224:formatInFex>>false</ax224:formatInFex>
    <ax224:parameterList xsi:nil="true"/>
    <ax224:secondPostProcessingProcedure/>
    <ax224:secondPreProcessingProcedure/>
    <ax224:sendFormat>HTML</ax224:sendFormat>
    <ax224:serverName>EDASERVE</ax224:serverName>
  </ax224:taskList>
  <ax224:timeInfo xsi:type="ax224:TimeInfoOnce">
    <ax224:description/>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:name/>
    <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
    <ax224:startTime>2014-03-11T18:27:46.710-04:00</
ax224:startTime>
    <ax224:type>0</ax224:type>
  </ax224:timeInfo>
  <ax224:timeInfoList xsi:type="ax224:TimeInfoOnce">
    <ax224:description/>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:name/>
    <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
    <ax224:startTime>2014-03-11T18:27:46.710-04:00</
ax224:startTime>
    <ax224:type>0</ax224:type>
  </ax224:timeInfoList>

```

```

<ax224:traceType>0</ax224:traceType>
  </ax226:schedule>
  <ax226:startTime>2014-03-11T18:25:40.622-04:00</ax226:startTime>
  <ax226:status>1</ax226:status>
</ns:return>
<ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax226:Job">
  <ax226:distributionServerName xsi:nil="true"/>
  <ax226:fullyQualifiedServerName xsi:nil="true"/>
  <ax226:id>J07e3e5b8j0608j49bfj823ajc00d8768a7ba</ax226:id>
  <ax226:schedule xsi:type="ax224:Schedule">
    <ax224:IBFSObjectType>113</ax224:IBFSObjectType>
    <ax224:active>true</ax224:active>
    <ax224:compressedReport>false</ax224:compressedReport>
    <ax224:deleteJobAfterRun>false</ax224:deleteJobAfterRun>
    <ax224:description>Carinst Report 2</ax224:description>
    <ax224:distribution xsi:type="ax224:DistributionEmail">
      <ax224:description/>
      <ax224:disabled>false</ax224:disabled>
      <ax224:id/>
      <ax224:type>EMAIL</ax224:type>
      <ax224:authEnabled>false</ax224:authEnabled>
      <ax224:authPassword/>
      <ax224:authUserId/>
      <ax224:destination xsi:type="ax224:Destination">
        <ax224:distributionFile/>
        <ax224:distributionList/>
        <ax224:distributionListFullPath/>
        <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
          <ax224:password/>
          <ax224:procedureName/>
          <ax224:serverName/>
          <ax224:userName/>
        </ax224:dynamicAddress>
        <ax224:singleAddress/>
        <ax224:type>DISTRIBUTION_LIST</ax224:type>
      </ax224:destination>
    </ax224:distribution>
  </ax226:schedule>
</ns:return>

```

```

<ax224:inlineMessage/>
  <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
  <ax224:mailFrom/>
  <ax224:mailReplyAddress/>
  <ax224:mailServerName/>
  <ax224:mailSubject/>
  <ax224:sendingReportAsAttachment>true</
ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distribution>
<ax224:distributionList xsi:type="ax224:DistributionEmail">
  <ax224:description/>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:type>EMAIL</ax224:type>
  <ax224:authEnabled>>false</ax224:authEnabled>
  <ax224:authPassword/>
  <ax224:authUserId/>
  <ax224:destination xsi:type="ax224:Destination">
    <ax224:distributionFile/>
    <ax224:distributionList/>
    <ax224:distributionListFullPath/>
    <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
      <ax224:password/>
      <ax224:procedureName/>
      <ax224:serverName/>
      <ax224:userName/>
    </ax224:dynamicAddress>
    <ax224:singleAddress/>
    <ax224:type>DISTRIBUTION_LIST</ax224:type>
  </ax224:destination>
  <ax224:inlineMessage/>
  <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
  <ax224:mailFrom/>
  <ax224:mailReplyAddress/>
  <ax224:mailServerName/>
  <ax224:mailSubject/>
  <ax224:sendingReportAsAttachment>true</
ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distributionList>

```

```

<ax224:firstTask xsi:type="ax224:TaskWFServerProcedure">
  <ax224:description>WebFocus Server Procedure task</
ax224:description>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:procedureId/>
  <ax224:procedureName/>
  <ax224:reportName/>
  <ax224:taskRetry xsi:nil="true"/>
  <ax224:type>0</ax224:type>
  <ax224:allowFormatList xsi:nil="true"/>
  <ax224:burst>>false</ax224:burst>
  <ax224:execId/>
  <ax224:execPassword/>
  <ax224:firstPostProcessingProcedure/>
  <ax224:firstPreProcessingProcedure/>
  <ax224:formatInFex>>false</ax224:formatInFex>
  <ax224:parameterList xsi:nil="true"/>
  <ax224:secondPostProcessingProcedure/>
  <ax224:secondPreProcessingProcedure/>
  <ax224:sendFormat>HTML</ax224:sendFormat>
  <ax224:serverName>EDASERVE</ax224:serverName>
</ax224:firstTask>
<ax224:ibfsId>6dff2b49I8245I4638I9e9fIc5900a9a12d5</
ax224:ibfsId>
  <ax224:ibfsPath/>
  <ax224:id>S23f65030s728as482asa632s879fd9f6a727</ax224:id>
  <ax224:lastModified>2014-03-11T18:27:46.710-04:00</
ax224:lastModified>
  <ax224:lastTimeExecuted>1970-01-01T00:00:00.000-05:00</
ax224:lastTimeExecuted>
  <ax224:name/>
  <ax224:notification xsi:type="ax224:Notification">
    <ax224:addressForBriefNotification/>
    <ax224:addressForFullNotification/>
    <ax224:description/>
    <ax224:from/>
    <ax224:id/>
    <ax224:subject/>
    <ax224:type>INACTIVE</ax224:type>
  </ax224:notification>

```

```

<ax224:owner>admin</ax224:owner>
  <ax224:priority>3</ax224:priority>
  <ax224:statusLastExecuted/>
  <ax224:summary/>
  <ax224:taskList xsi:type="ax224:TaskWFServerProcedure">
    <ax224:description>WebFocus Server Procedure task</
ax224:description>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:procedureId/>
    <ax224:procedureName/>
    <ax224:reportName/>
    <ax224:taskRetry xsi:nil="true"/>
    <ax224:type>0</ax224:type>
    <ax224:allowFormatList xsi:nil="true"/>
    <ax224:burst>>false</ax224:burst>
    <ax224:execId/>
    <ax224:execPassword/>
    <ax224:firstPostProcessingProcedure/>
    <ax224:firstPreProcessingProcedure/>
    <ax224:formatInFex>>false</ax224:formatInFex>
    <ax224:parameterList xsi:nil="true"/>
    <ax224:secondPostProcessingProcedure/>
    <ax224:secondPreProcessingProcedure/>
    <ax224:sendFormat>HTML</ax224:sendFormat>
    <ax224:serverName>EDASERVE</ax224:serverName>
  </ax224:taskList>
  <ax224:timeInfo xsi:type="ax224:TimeInfoOnce">
    <ax224:description/>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:name/>
    <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
    <ax224:startTime>2014-03-11T18:27:46.710-04:00</
ax224:startTime>
    <ax224:type>0</ax224:type>
  </ax224:timeInfo>

```

```

<ax224:timeInfoList xsi:type="ax224:TimeInfoOnce">
  <ax224:description/>
  <ax224:disabled>false</ax224:disabled>
  <ax224:id/>
  <ax224:name/>
  <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
  <ax224:startTime>2014-03-11T18:27:46.710-04:00</
ax224:startTime>
  <ax224:type>0</ax224:type>
</ax224:timeInfoList>
  <ax224:traceType>0</ax224:traceType>
</ax226:schedule>
  <ax226:startTime>2014-03-11T18:25:47.863-04:00</ax226:startTime>
  <ax226:status>1</ax226:status>
</ns:return>
</ns:getRunningJobsResponse>

```

Listing Running Jobs for an Owner

This RESTful web service request can be used to list the ReportCaster jobs that are running for a specific schedule owner.

HTTP Method: GET

REST URL Format:

`http://host:port/ibi_apps/services/ConsoleServiceREST/getRunningJobsByOwner?owner=owner`

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

owner

Is the owner of the ReportCaster schedule.

Example:

In the following example, a list of ReportCaster jobs that are running for the owner named *admin* is retrieved.

Request:

`http://localhost:8080/ibi_apps/services/ConsoleServiceREST/getRunningJobsByOwner?owner=admin`

Response:

```

<ns:getRunningJobsByOwnerResponse xmlns:ns="http://ws.api.broker.ibi"
xmlns:ax220="http://rmi.java.xsd" xmlns:ax221="http://io.java.xsd"
xmlns:ax224="http://schedule.data.api.broker.ibi/xsd" xmlns:ax226="http://
console.data.api.broker.ibi/xsd">
  <ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax226:Job">
    <ax226:distributionServerName xsi:nil="true"/>
    <ax226:fullyQualifiedServerName xsi:nil="true"/>
    <ax226:id>J00709cc0jdeddj4115ja3d9j353dcd5bc11e</ax226:id>
    <ax226:schedule xsi:type="ax224:Schedule">
      <ax224:IBFSObjectType>113</ax224:IBFSObjectType>
      <ax224:active>true</ax224:active>
      <ax224:compressedReport>false</ax224:compressedReport>
      <ax224:deleteJobAfterRun>false</ax224:deleteJobAfterRun>
      <ax224:description>Carinst Report 2</ax224:description>
      <ax224:distribution xsi:type="ax224:DistributionEmail">
        <ax224:description/>
        <ax224:disabled>false</ax224:disabled>
        <ax224:id/>
        <ax224:type>EMAIL</ax224:type>
        <ax224:authEnabled>false</ax224:authEnabled>
        <ax224:authPassword/>
        <ax224:authUserId/>
        <ax224:destination xsi:type="ax224:Destination">
          <ax224:distributionFile/>
          <ax224:distributionList/>
          <ax224:distributionListFullPath/>
          <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
            <ax224:password/>
            <ax224:procedureName/>
            <ax224:serverName/>
            <ax224:userName/>
          </ax224:dynamicAddress>
          <ax224:singleAddress/>
          <ax224:type>DISTRIBUTION_LIST</ax224:type>
        </ax224:destination>
        <ax224:inlineMessage/>
        <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
        <ax224:mailFrom/>
        <ax224:mailReplyAddress/>
        <ax224:mailServerName/>
        <ax224:mailSubject/>
      </ax224:distribution>
    </ax226:schedule>
  </ns:return>
</ns:getRunningJobsByOwnerResponse>

```

```

        <ax224:sendingReportAsAttachment>true</
ax224:sendingReportAsAttachment>
        <ax224:sslEnabled>>false</ax224:sslEnabled>
        <ax224:tlsEnabled>>false</ax224:tlsEnabled>
        <ax224:zipFileName/>
        <ax224:zipResult>>false</ax224:zipResult>
    </ax224:distribution>
    <ax224:distributionList xsi:type="ax224:DistributionEmail">
        <ax224:description/>
        <ax224:disabled>>false</ax224:disabled>
        <ax224:id/>
        <ax224:type>EMAIL</ax224:type>
        <ax224:authEnabled>>false</ax224:authEnabled>
        <ax224:authPassword/>
        <ax224:authUserId/>
        <ax224:destination xsi:type="ax224:Destination">
            <ax224:distributionFile/>
            <ax224:distributionList/>
            <ax224:distributionListFullPath/>
            <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
                <ax224:password/>
                <ax224:procedureName/>
                <ax224:serverName/>
                <ax224:userName/>
            </ax224:dynamicAddress>
            <ax224:singleAddress/>
            <ax224:type>DISTRIBUTION_LIST</ax224:type>
        </ax224:destination>
        <ax224:inlineMessage/>
        <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
        <ax224:mailFrom/>
        <ax224:mailReplyAddress/>
        <ax224:mailServerName/>
        <ax224:mailSubject/>
        <ax224:sendingReportAsAttachment>true</
ax224:sendingReportAsAttachment>
        <ax224:sslEnabled>>false</ax224:sslEnabled>
        <ax224:tlsEnabled>>false</ax224:tlsEnabled>
        <ax224:zipFileName/>
        <ax224:zipResult>>false</ax224:zipResult>
    </ax224:distributionList>

```



```

    <ax224:firstTask xsi:type="ax224:TaskWFServerProcedure">
      <ax224:description>WebFocus Server Procedure task</
ax224:description>
      <ax224:disabled>>false</ax224:disabled>
      <ax224:id/>
      <ax224:procedureId/>
      <ax224:procedureName/>
      <ax224:reportName/>
      <ax224:taskRetry xsi:nil="true"/>
      <ax224:type>0</ax224:type>
      <ax224:allowFormatList xsi:nil="true"/>
      <ax224:burst>>false</ax224:burst>
      <ax224:execId/>
      <ax224:execPassword/>
      <ax224:firstPostProcessingProcedure/>
      <ax224:firstPreProcessingProcedure/>
      <ax224:formatInFex>>false</ax224:formatInFex>
      <ax224:parameterList xsi:nil="true"/>
      <ax224:secondPostProcessingProcedure/>
      <ax224:secondPreProcessingProcedure/>
      <ax224:sendFormat>HTML</ax224:sendFormat>
      <ax224:serverName>EDASERVE</ax224:serverName>
    </ax224:firstTask>
    <ax224:ibfsId>6dff2b49I8245I4638I9e9fIc5900a9a12d5</
ax224:ibfsId>
    <ax224:ibfsPath/>
    <ax224:id>S23f65030s728as482asa632s879fd9f6a727</ax224:id>
    <ax224:lastModified>2014-02-24T14:42:43.031-05:00</
ax224:lastModified>
    <ax224:lastTimeExecuted>1970-01-01T00:00:00.000-05:00</
ax224:lastTimeExecuted>
    <ax224:name/>
    <ax224:notification xsi:type="ax224:Notification">
      <ax224:addressForBriefNotification/>
      <ax224:addressForFullNotification/>
      <ax224:description/>
      <ax224:from/>
      <ax224:id/>
      <ax224:subject/>
      <ax224:type>INACTIVE</ax224:type>
    </ax224:notification>
    <ax224:owner>admin</ax224:owner>
    <ax224:priority>3</ax224:priority>
    <ax224:statusLastExecuted/>
    <ax224:summary/>

```

```

<ax224:taskList xsi:type="ax224:TaskWFServerProcedure">
  <ax224:description>WebFocus Server Procedure task</
ax224:description>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:procedureId/>
  <ax224:procedureName/>
  <ax224:reportName/>
  <ax224:taskRetry xsi:nil="true"/>
  <ax224:type>0</ax224:type>
  <ax224:allowFormatList xsi:nil="true"/>
  <ax224:burst>>false</ax224:burst>
  <ax224:execId/>
  <ax224:execPassword/>
  <ax224:firstPostProcessingProcedure/>
  <ax224:firstPreProcessingProcedure/>
  <ax224:formatInFex>>false</ax224:formatInFex>
  <ax224:parameterList xsi:nil="true"/>
  <ax224:secondPostProcessingProcedure/>
  <ax224:secondPreProcessingProcedure/>
  <ax224:sendFormat>HTML</ax224:sendFormat>
  <ax224:serverName>EDASERVE</ax224:serverName>
</ax224:taskList>
<ax224:timeInfo xsi:type="ax224:TimeInfoOnce">
  <ax224:description/>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:name/>
  <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
  <ax224:startTime>2014-02-24T14:42:43.031-05:00</
ax224:startTime>
  <ax224:type>0</ax224:type>
</ax224:timeInfo>
<ax224:timeInfoList xsi:type="ax224:TimeInfoOnce">
  <ax224:description/>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:name/>
  <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
  <ax224:startTime>2014-02-24T14:42:43.031-05:00</
ax224:startTime>
  <ax224:type>0</ax224:type>
</ax224:timeInfoList>
<ax224:traceType>0</ax224:traceType>
</ax226:schedule>

```

```

<ax226:startTime>2014-02-24T14:24:35.685-05:00</ax226:startTime>
  <ax226:status>1</ax226:status>
</ns:return>
<ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax226:Job">
  <ax226:distributionServerName xsi:nil="true"/>
  <ax226:fullyQualifiedServerName xsi:nil="true"/>
  <ax226:id>Jfbd27992j5c60j4f48ja110jea71ff6ae996</ax226:id>
  <ax226:schedule xsi:type="ax224:Schedule">
    <ax224:IBFSObjectType>113</ax224:IBFSObjectType>
    <ax224:active>true</ax224:active>
    <ax224:compressedReport>false</ax224:compressedReport>
    <ax224:deleteJobAfterRun>false</ax224:deleteJobAfterRun>
    <ax224:description>Carinst Report 2</ax224:description>
    <ax224:distribution xsi:type="ax224:DistributionEmail">
      <ax224:description/>
      <ax224:disabled>false</ax224:disabled>
      <ax224:id/>
      <ax224:type>EMAIL</ax224:type>
      <ax224:authEnabled>false</ax224:authEnabled>
      <ax224:authPassword/>
      <ax224:authUserId/>
      <ax224:destination xsi:type="ax224:Destination">
        <ax224:distributionFile/>
        <ax224:distributionList/>
        <ax224:distributionListFullPath/>
        <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
          <ax224:password/>
          <ax224:procedureName/>
          <ax224:serverName/>
          <ax224:userName/>
        </ax224:dynamicAddress>
        <ax224:singleAddress/>
        <ax224:type>DISTRIBUTION_LIST</ax224:type>
      </ax224:destination>
      <ax224:inlineMessage/>
      <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
      <ax224:mailFrom/>
      <ax224:mailReplyAddress/>
      <ax224:mailServerName/>
      <ax224:mailSubject/>
      <ax224:sendingReportAsAttachment>true</
ax224:sendingReportAsAttachment>
      <ax224:sslEnabled>false</ax224:sslEnabled>
      <ax224:tlsEnabled>false</ax224:tlsEnabled>
      <ax224:zipFileName/>
      <ax224:zipResult>false</ax224:zipResult>
    </ax224:distribution>
  </ax226:schedule>
</ns:return>

```

```

<ax224:distributionList xsi:type="ax224:DistributionEmail">
  <ax224:description/>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:type>EMAIL</ax224:type>
  <ax224:authEnabled>>false</ax224:authEnabled>
  <ax224:authPassword/>
  <ax224:authUserId/>
  <ax224:destination xsi:type="ax224:Destination">
    <ax224:distributionFile/>
    <ax224:distributionList/>
    <ax224:distributionListFullPath/>
    <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
      <ax224:password/>
      <ax224:procedureName/>
      <ax224:serverName/>
      <ax224:userName/>
    </ax224:dynamicAddress>
    <ax224:singleAddress/>
    <ax224:type>DISTRIBUTION_LIST</ax224:type>
  </ax224:destination>
  <ax224:inlineMessage/>
  <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
  <ax224:mailFrom/>
  <ax224:mailReplyAddress/>
  <ax224:mailServerName/>
  <ax224:mailSubject/>
  <ax224:sendingReportAsAttachment>>true</
ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distributionList>

```

```

<ax224:firstTask xsi:type="ax224:TaskWFServerProcedure">
  <ax224:description>WebFocus Server Procedure task</
ax224:description>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:procedureId/>
  <ax224:procedureName/>
  <ax224:reportName/>
  <ax224:taskRetry xsi:nil="true"/>
  <ax224:type>0</ax224:type>
  <ax224:allowFormatList xsi:nil="true"/>
  <ax224:burst>>false</ax224:burst>
  <ax224:execId/>
  <ax224:execPassword/>
  <ax224:firstPostProcessingProcedure/>
  <ax224:firstPreProcessingProcedure/>
  <ax224:formatInFex>>false</ax224:formatInFex>
  <ax224:parameterList xsi:nil="true"/>
  <ax224:secondPostProcessingProcedure/>
  <ax224:secondPreProcessingProcedure/>
  <ax224:sendFormat>HTML</ax224:sendFormat>
  <ax224:serverName>EDASERVE</ax224:serverName>
</ax224:firstTask>
<ax224:ibfsId>6dff2b49I8245I4638I9e9fIc5900a9a12d5</
ax224:ibfsId>
  <ax224:ibfsPath/>
  <ax224:id>S23f65030s728as482asa632s879fd9f6a727</ax224:id>
  <ax224:lastModified>2014-02-24T14:42:43.032-05:00</
ax224:lastModified>
  <ax224:lastTimeExecuted>1970-01-01T00:00:00.000-05:00</
ax224:lastTimeExecuted>
  <ax224:name/>
  <ax224:notification xsi:type="ax224:Notification">
    <ax224:addressForBriefNotification/>
    <ax224:addressForFullNotification/>
    <ax224:description/>
    <ax224:from/>
    <ax224:id/>
    <ax224:subject/>
    <ax224:type>INACTIVE</ax224:type>
  </ax224:notification>
  <ax224:owner>admin</ax224:owner>
  <ax224:priority>3</ax224:priority>
  <ax224:statusLastExecuted/>
  <ax224:summary/>

```

```

    <ax224:taskList xsi:type="ax224:TaskWFServerProcedure">
      <ax224:description>WebFocus Server Procedure task</
ax224:description>
      <ax224:disabled>>false</ax224:disabled>
      <ax224:id/>
      <ax224:procedureId/>
      <ax224:procedureName/>
      <ax224:reportName/>
      <ax224:taskRetry xsi:nil="true"/>
      <ax224:type>0</ax224:type>
      <ax224:allowFormatList xsi:nil="true"/>
      <ax224:burst>>false</ax224:burst>
      <ax224:execId/>
      <ax224:execPassword/>
      <ax224:firstPostProcessingProcedure/>
      <ax224:firstPreProcessingProcedure/>
      <ax224:formatInFex>>false</ax224:formatInFex>
      <ax224:parameterList xsi:nil="true"/>
      <ax224:secondPostProcessingProcedure/>
      <ax224:secondPreProcessingProcedure/>
      <ax224:sendFormat>HTML</ax224:sendFormat>
      <ax224:serverName>EDASERVE</ax224:serverName>
    </ax224:taskList>
    <ax224:timeInfo xsi:type="ax224:TimeInfoOnce">
      <ax224:description/>
      <ax224:disabled>>false</ax224:disabled>
      <ax224:id/>
      <ax224:name/>
      <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
      <ax224:startTime>2014-02-24T14:42:43.031-05:00</
ax224:startTime>
      <ax224:type>0</ax224:type>
    </ax224:timeInfo>
    <ax224:timeInfoList xsi:type="ax224:TimeInfoOnce">
      <ax224:description/>
      <ax224:disabled>>false</ax224:disabled>
      <ax224:id/>
      <ax224:name/>
      <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
      <ax224:startTime>2014-02-24T14:42:43.031-05:00</
ax224:startTime>
      <ax224:type>0</ax224:type>
    </ax224:timeInfoList>
    <ax224:traceType>0</ax224:traceType>
  </ax226:schedule>

```

```

<ax226:startTime>2014-02-24T14:24:35.817-05:00</ax226:startTime>
  <ax226:status>1</ax226:status>
</ns:return>
<ns:return xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ax226:Job">
  <ax226:distributionServerName xsi:nil="true"/>
  <ax226:fullyQualifiedServerName xsi:nil="true"/>
  <ax226:id>J57602256jc2bfj4523j8492jl38f33deb40f</ax226:id>
  <ax226:schedule xsi:type="ax224:Schedule">
    <ax224:IBFSObjectType>113</ax224:IBFSObjectType>
    <ax224:active>true</ax224:active>
    <ax224:compressedReport>false</ax224:compressedReport>
    <ax224:deleteJobAfterRun>false</ax224:deleteJobAfterRun>
    <ax224:description>Carinst Report 2</ax224:description>
    <ax224:distribution xsi:type="ax224:DistributionEmail">
      <ax224:description/>
      <ax224:disabled>false</ax224:disabled>
      <ax224:id/>
      <ax224:type>EMAIL</ax224:type>
      <ax224:authEnabled>false</ax224:authEnabled>
      <ax224:authPassword/>
      <ax224:authUserId/>
      <ax224:destination xsi:type="ax224:Destination">
        <ax224:distributionFile/>
        <ax224:distributionList/>
        <ax224:distributionListFullPath/>
        <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
          <ax224:password/>
          <ax224:procedureName/>
          <ax224:serverName/>
          <ax224:userName/>
        </ax224:dynamicAddress>
        <ax224:singleAddress/>
        <ax224:type>DISTRIBUTION_LIST</ax224:type>
      </ax224:destination>
      <ax224:inlineMessage/>
      <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
      <ax224:mailFrom/>
      <ax224:mailReplyAddress/>
      <ax224:mailServerName/>
      <ax224:mailSubject/>
      <ax224:sendingReportAsAttachment>true</
ax224:sendingReportAsAttachment>
      <ax224:sslEnabled>false</ax224:sslEnabled>
      <ax224:tlsEnabled>false</ax224:tlsEnabled>
      <ax224:zipFileName/>
      <ax224:zipResult>false</ax224:zipResult>
    </ax224:distribution>
  </ax226:schedule>
</ns:return>

```

```

<ax224:distributionList xsi:type="ax224:DistributionEmail">
  <ax224:description/>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:type>EMAIL</ax224:type>
  <ax224:authEnabled>>false</ax224:authEnabled>
  <ax224:authPassword/>
  <ax224:authUserId/>
  <ax224:destination xsi:type="ax224:Destination">
    <ax224:distributionFile/>
    <ax224:distributionList/>
    <ax224:distributionListFullPath/>
    <ax224:dynamicAddress xsi:type="ax224:DynamicAddress">
      <ax224:password/>
      <ax224:procedureName/>
      <ax224:serverName/>
      <ax224:userName/>
    </ax224:dynamicAddress>
    <ax224:singleAddress/>
    <ax224:type>DISTRIBUTION_LIST</ax224:type>
  </ax224:destination>
  <ax224:inlineMessage/>
  <ax224:inlineTaskIndex>0</ax224:inlineTaskIndex>
  <ax224:mailFrom/>
  <ax224:mailReplyAddress/>
  <ax224:mailServerName/>
  <ax224:mailSubject/>
  <ax224:sendingReportAsAttachment>>true</
ax224:sendingReportAsAttachment>
  <ax224:sslEnabled>>false</ax224:sslEnabled>
  <ax224:tlsEnabled>>false</ax224:tlsEnabled>
  <ax224:zipFileName/>
  <ax224:zipResult>>false</ax224:zipResult>
</ax224:distributionList>
<ax224:firstTask xsi:type="ax224:TaskWFServerProcedure">
  <ax224:description>WebFocus Server Procedure task</
ax224:description>
  <ax224:disabled>>false</ax224:disabled>
  <ax224:id/>
  <ax224:procedureId/>
  <ax224:procedureName/>
  <ax224:reportName/>
  <ax224:taskRetry xsi:nil="true"/>
  <ax224:type>0</ax224:type>
  <ax224:allowFormatList xsi:nil="true"/>

```



```

<ax224:burst>>false</ax224:burst>
  <ax224:execId/>
  <ax224:execPassword/>
  <ax224:firstPostProcessingProcedure/>
  <ax224:firstPreProcessingProcedure/>
  <ax224:formatInFex>>false</ax224:formatInFex>
  <ax224:parameterList xsi:nil="true"/>
  <ax224:secondPostProcessingProcedure/>
  <ax224:secondPreProcessingProcedure/>
  <ax224:sendFormat>HTML</ax224:sendFormat>
  <ax224:serverName>EDASERVE</ax224:serverName>
</ax224:firstTask>
<ax224:ibfsId>6dff2b49I8245I4638I9e9fIc5900a9a12d5</
ax224:ibfsId>
  <ax224:ibfsPath/>
  <ax224:id>S23f65030s728as482asa632s879fd9f6a727</ax224:id>
  <ax224:lastModified>2014-02-24T14:42:43.032-05:00</
ax224:lastModified>
  <ax224:lastTimeExecuted>1970-01-01T00:00:00.000-05:00</
ax224:lastTimeExecuted>
  <ax224:name/>
  <ax224:notification xsi:type="ax224:Notification">
    <ax224:addressForBriefNotification/>
    <ax224:addressForFullNotification/>
    <ax224:description/>
    <ax224:from/>
    <ax224:id/>
    <ax224:subject/>
    <ax224:type>INACTIVE</ax224:type>
  </ax224:notification>
  <ax224:owner>admin</ax224:owner>
  <ax224:priority>3</ax224:priority>
  <ax224:statusLastExecuted/>
  <ax224:summary/>
  <ax224:taskList xsi:type="ax224:TaskWFServerProcedure">
    <ax224:description>WebFocus Server Procedure task</
ax224:description>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:procedureId/>
    <ax224:procedureName/>
    <ax224:reportName/>
    <ax224:taskRetry xsi:nil="true"/>
    <ax224:type>0</ax224:type>
    <ax224:allowFormatList xsi:nil="true"/>
    <ax224:burst>>false</ax224:burst>

```

```

<ax224:execId/>
    <ax224:execPassword/>
    <ax224:firstPostProcessingProcedure/>
    <ax224:firstPreProcessingProcedure/>
    <ax224:formatInFex>>false</ax224:formatInFex>
    <ax224:parameterList xsi:nil="true" />
    <ax224:secondPostProcessingProcedure/>
    <ax224:secondPreProcessingProcedure/>
    <ax224:sendFormat>HTML</ax224:sendFormat>
    <ax224:serverName>EDASERVE</ax224:serverName>
</ax224:taskList>
<ax224:timeInfo xsi:type="ax224:TimeInfoOnce">
    <ax224:description/>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:name/>
    <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
    <ax224:startTime>2014-02-24T14:42:43.032-05:00</
ax224:startTime>
    <ax224:type>0</ax224:type>
</ax224:timeInfo>
<ax224:timeInfoList xsi:type="ax224:TimeInfoOnce">
    <ax224:description/>
    <ax224:disabled>>false</ax224:disabled>
    <ax224:id/>
    <ax224:name/>
    <ax224:nextRunTime>1970-01-01T00:00:00.000-05:00</
ax224:nextRunTime>
    <ax224:startTime>2014-02-24T14:42:43.032-05:00</
ax224:startTime>
    <ax224:type>0</ax224:type>
</ax224:timeInfoList>
<ax224:traceType>0</ax224:traceType>
</ax226:schedule>
<ax226:startTime>2014-02-24T14:24:36.070-05:00</ax226:startTime>
<ax226:status>1</ax226:status>
</ns:return>
</ns:getRunningJobsByOwnerResponse>

```

Removing a Job From the Job Queue

This RESTful web service request can be used to remove a specific ReportCaster job from the job queue.

HTTP Method: GET

REST URL Format:

http://host:port/ibi_apps/services/ConsoleServiceREST/removeJobFromQueue?jobId=jobId

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

jobId

Is a unique identifier for the ReportCaster job.

Example:

In the following example, the ReportCaster job with a job ID of Jc12b4443jb1f8j4c19j90aaj7ba31ac4dbf5 is removed from the job queue.

Request:

```
http://localhost:8080/ibi_apps/services/ConsoleServiceREST/removeJobFromQueue?
jobId=Jc12b4443jb1f8j4c19j90aaj7ba31ac4dbf5
```

Response:

```
<ns:removeJobFromQueueResponse xmlns:ns="http://ws.api.broker.ibi">
  <ns:return>1</ns:return>
</ns:removeJobFromQueueResponse>
```


Using the RESTful Web Services Test Page

This appendix describes how to use the Test page to test and debug the functionality of RESTful web services.

In this chapter:

- ❑ [Accessing the Test Page](#)
- ❑ [Using the Test Page](#)

Accessing the Test Page

Enter the following URL in your browser to access the Test page:

http://host:port/ibi_apps/rs?IBIRS_action=TEST

where:

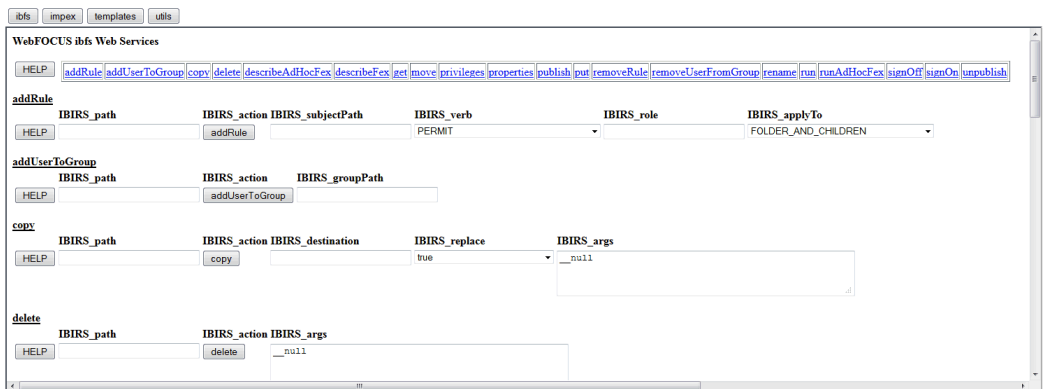
host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

The Test page opens in your browser, as shown in the following image.



Using the Test Page

Four buttons (ibfs, impex, templates, and utils) are available at the top of the Test page. Clicking a specific button will display a list of RESTful web services functionality that can be tested.

- ibfs.** The majority of the RESTful web services functionality is included in this category. Change Management and Template functionality are excluded.
- impex.** Functions that support Change Management are included in this category. Change Management Export and Import are included.
- templates.** Functions that support Templates are included in this category. Creating and Running a Template are included.
- utils.** Functions that support Utilities are included in this category. Expanding and Compacting a Policy are included.

Within each category of RESTful web services, each of the RESTful web services functionality tests are based on the IBIRS_action parameter value. Every RESTful web service includes this parameter. The label of the button on each test within the Test page represents the parameter value for the IBIRS_action parameter. To display the form for a specific test based on IBIRS_action, click the appropriate link at the top of the Test page (for example, put). Note that the parameter name is displayed above each field or drop-down list that is required to test the particular functionality.

When a particular parameter requires a path, the URL from the Test page, excluding IBIRS_action=TEST along with the category name is assumed in making the REST request. In Example 1 from [Listing Folders and Subfolders](#) on page 44, the path can be represented in the field from the Test page test as /WFC/Repository.

In order for any of the tests to work successfully, you must be authenticated to WebFOCUS. This can be done by logging on to the WebFOCUS Business Intelligence Portal or by entering the credentials in the signOn section of the Test page and clicking the signOn button.

Example:

In the following example, a test is performed to add a Group based on the example from [Adding and Updating a Group](#) on page 117.

The screenshot displays the RESTful Web Services Test Page interface. At the top, there are navigation tabs: 'ibfs', 'impex', 'templates', and 'utils'. Below these is a 'publish' button. The main area is divided into sections for different HTTP methods: 'put', 'removeRule', 'removeUserFromGroup', and 'rename'. Each section has a 'HELP' button and input fields for various parameters. The 'put' section is currently active, showing the following configuration:

IBIRS_path	IBIRS_action	IBIRS_object	IBIRS_private	IBIRS_replace	IBIRS_args
/SSYS/GROUPS/RestUsers	put	<object _id="IBFSGroupObject" contains="true" description="RESTful Web Services Users" type="Group"></object>	__null	true	__null

The 'removeRule' section has input fields for 'IBIRS_path', 'IBIRS_action', 'IBIRS_subjectPath', and 'IBIRS_role'. The 'removeUserFromGroup' section has input fields for 'IBIRS_path' and 'IBIRS_groupPath'. The 'rename' section has input fields for 'IBIRS_path', 'IBIRS_action', 'IBIRS_newName', and 'IBIRS_args'.

Alternative Method of Calling WebFOCUS RESTful Web Service Requests

This appendix describes an alternative method that can be used to call WebFOCUS RESTful web service requests.

In this chapter:

- ❑ [Calling WebFOCUS RESTful Web Service Requests](#)
-

Calling WebFOCUS RESTful Web Service Requests

For each WebFOCUS RESTful web service request, the portion of the URL path following *rs* can be represented as a parameter. *IBIRS_service* represents the parameter for the category and *IBIRS_path* represents the path to the specific functionality that is being performed.

Example:

In the following example, the REST URL for Example 1 from [Listing Folders and Subfolders](#) on page 44 shows the REST URL as:

```
http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository?IBIRS_action=get
```

This request can also be sent as follows:

```
http://localhost:8080/ibi_apps/rs?IBIRS_action=get&IBIRS_path=/WFC/Repository&IBIRS_service=ibfs
```


Chapter 9

Visual Basic .NET, Java, HTML and jQuery Code Examples

This appendix provides Visual Basic .NET, Java, HTML and jQuery code examples on how to create WebFOCUS RESTful web service requests.

In this chapter:

- ❑ [Signing In to WebFOCUS](#)
- ❑ [Listing Folders From WebFOCUS](#)
- ❑ [Running a WebFOCUS Report](#)
- ❑ [Handling Drill-downs, Active Cache, and On-Demand Paging Reports](#)
- ❑ [Parsing the XML Response of a SignOn Request to Obtain the CSRF Name and Value](#)

Signing In to WebFOCUS

This section provides code examples that demonstrate how to sign in to WebFOCUS.

Visual Basic .NET Example

```
Imports System.Net
Imports System.IO
Imports System.Text
Dim cookies As New CookieContainer
Dim webStream As Stream
Dim webResponse As String = ""
Dim request As HttpWebRequest
Dim response As HttpWebResponse
Dim postData As String
request = WebRequest.Create("http://localhost:8080/ibi_apps/rs/ibfs")
request.Method = "POST"
postData = "IBIRS_action=signOn&IBIRS_userName=admin&IBIRS_password=admin"
request.CookieContainer = cookies
Dim byteArray As Byte() = Encoding.UTF8.GetBytes(postData)
request.ContentType = "application/x-www-form-urlencoded"
request.ContentLength = byteArray.Length
Dim dataStream As Stream = request.GetRequestStream()
dataStream.Write(byteArray, 0, byteArray.Length)
dataStream.Close()
response = request.GetResponse()
webStream = response.GetResponseStream()
Dim webStreamReader As New StreamReader(webStream)
While webStreamReader.Peek >= 0
    webResponse = webStreamReader.ReadToEnd()
End While
```

Java Example

```
import java.awt.Frame;
import java.io.BufferedReader;
import java.io.InputStream;
import java.io.InputStreamReader;
import org.apache.commons.httpclient.*;
import org.apache.commons.httpclient.methods.*;
String request = "http://localhost:8080/ibi_apps/rs/ibfs";
HttpClient client = new HttpClient();
PostMethod method = new PostMethod(request);

method.addParameter("IBIRS_action", "signOn");
method.addParameter("IBIRS_userName", "admin");
method.addParameter("IBIRS_password", "admin");

int statusCode = client.executeMethod(method);
Header[] cookies = null;
InputStream rstream = null;

rstream = method.getResponseBodyAsStream();
cookies = method.getResponseHeaders("Set-Cookie");

BufferedReader br = new BufferedReader(new InputStreamReader(rstream));
String line;
while ((line = br.readLine()) != null) {
    System.out.println(line);
}
br.close();
```

HTML and jQuery Example

```

<!DOCTYPE html>
<html>
<head>
  <title></title>
  <meta charset="utf-8" />
  <script type="text/javascript" src="http://code.jquery.com/jquery-3.1.0.js"> </
script>
  <script type='text/javascript' src="http://cdnjs.cloudflare.com/ajax/libs/jquery-
ajaxtransport-xdomainrequest/1.0.1/
jquery.xdomainrequest.min.js"></script>

  <script type="text/javascript">
    var csrf_name;
    var csrf_value;
    $(document).ready(function (IBIRS_action, IBIRS_userName, IBIRS_password) {
      var contentType = "application/x-www-form-urlencoded; charset=utf-8";
      if (window.XDomainRequest)
        contentType = "text/plain";
      var divToBeWorkedOn = "#AjaxPlaceHolder";
      var webMethod = "http://machine:port/ibi_apps/rs/ibfs";
      var IBIRS_action = "signOn";
      var IBIRS_userName = "admin";
      var IBIRS_password = "admin";
      var parameters = 'IBIRS_action=' + IBIRS_action + '&IBIRS_userName=' +
IBIRS_userName + '&IBIRS_password=' + IBIRS_password;
      $.ajax({
        type: "POST",
        url: webMethod,
        data: parameters,
        dataType: "xml",
        contentType: contentType,
        success: alert("success"),
        complete: function(xhr,status) {
          alert(xhr.responseText);
          alert(xhr.getAllResponseHeaders());
        },
        error:function(jqXHR,textStatus,errorThrown)
        {
          alert("You can not send Cross Domain AJAX requests: " +
errorThrown);
        }
      });
    });
  
```

```

    </script>
</head>
<body>
    <div id="AjaxPlaceholder">
        <div align="center"></div>
    </div>

    <div></div>
</body>
</html>

```

Listing Folders From WebFOCUS

This section provides code examples that demonstrate how to retrieve a list of the top-level folders from WebFOCUS. A successful sign-on request is a prerequisite for running this example, including retrieving the HTTP Header cookies from its response.

Visual Basic .NET Example

```

Imports System.Net
Imports System.IO
Imports System.Text
Dim request3 As HttpWebRequest
Dim response3 As HttpWebResponse
Dim webStream3 As Stream
Dim webResponse3 As String = ""
Dim tempfile As String
request3 = WebRequest.Create("http://localhost:8080/ibi_apps/rs/ibfs/WFC/
Repository?IBIRS_action=get")
request3.Method = "GET"
'cookies is defined as CookieContainer in the Signing-On to WebFOCUS example
request3.CookieContainer = cookies
response3 = request3.GetResponse()
webStream3 = response3.GetResponseStream()
Dim webStreamReader3 As New StreamReader(webStream3)
tempfile = "c:\temp\Folders.xml"
FileOpen(1, tempfile, OpenMode.Output)
While webStreamReader3.Peek >= 0
    webResponse3 = webStreamReader3.ReadToEnd()
    PrintLine(1, webResponse3)
End While
FileClose(1)
Dim xmlElem = XElement.Parse(webResponse3)

```

Java Example

```

import java.awt.Frame;
import java.io.BufferedReader;
import java.io.File;
import java.io.FileOutputStream;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import org.apache.commons.httpclient.*;
import org.apache.commons.httpclient.methods.*;
String request3 = "http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository?
IBIRS_action=get";

GetMethod method_getFolders = new GetMethod(request3);
// cookies is defined as Header[] in the Signing-On to WebFOCUS example
for(int h=0; h<cookies.length; h++){
    method_getFolders.addRequestHeader(cookies[h].getName(), cookies[h].getValue());
}
// client is defined as HttpClient in the Signing-On to WebFOCUS example
int statusCode3 = client.executeMethod(method_getFolders);

InputStream rstream3 = null;
rstream3 = method_getFolders.getResponseBodyAsStream();
File tempfile = new File("c:\\temp\\Folders.xml");
FileOutputStream fos = new FileOutputStream(tempfile);
PrintWriter out=new PrintWriter(fos);
BufferedReader br3 = new BufferedReader(new InputStreamReader(rstream3));
String line3;
String newOutput = null;
while ((line3 = br3.readLine()) != null) {
    newOutput = line3;
    out.println(newOutput);
    System.out.println(line3);
}
br3.close();
out.close();

```

HTML and jQuery Example

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
  <meta charset="utf-8" />
  <script type="text/javascript" src="http://code.jquery.com/jquery-3.1.0.js"> </
script>
  <script type='text/javascript' src="http://cdnjs.cloudflare.com/ajax/libs/jquery-
ajaxtransport-xdomainrequest/1.0.1/
jquery.xdomainrequest.min.js"></script>
  <script type="text/javascript">
    var csrf_name;
    var csrf_value;
    var frameToBeWorkedOn = "#AjaxPlaceHolder";
    var contentType = "application/x-www-form-urlencoded; charset=utf-8";
    $(document).ready(function (IBIRS_action, IBIRS_userName, IBIRS_password) {
      if (window.XDomainRequest)
        contentType = "text/plain";
      var webMethod = "http://machine:port/ibi_apps/rs/ibfs";
      var IBIRS_action = "signOn";
      var IBIRS_userName = "admin";
      var IBIRS_password = "admin";
      var parameters = 'IBIRS_action=' + IBIRS_action + '&IBIRS_userName=' +
IBIRS_userName + '&IBIRS_password=' + IBIRS_password;
      $.ajax({
        type: "POST",
        url: webMethod,
        data: parameters,
        dataType: "xml",
        xhrFields: {
          withCredentials: true
        },
        crossDomain: true,
        contentType: contentType,
        success: listFolders,
        error:function(jqXHR,textStatus,errorThrown)
          {
            alert("You can not send Cross Domain AJAX requests: " +
errorThrown);
          }
      });
    });
    function listFolders() {
      if (window.XDomainRequest)
        contentType = "text";
    }
  }
</script>
</head>
</html>
```



```

var webMethod = "http://machine:port/ibi_apps/rs/ibfs/WFC/Repository";
var IBIRS_action = "get";
var parameters = 'IBIRS_action=' + IBIRS_action;

$.ajax({
  type: "GET",
  url: webMethod,
  data: parameters,
  dataType: "xml",
  xhrFields: {
    withCredentials: true
  },
  crossDomain: true,
  success: xmlParse,
  //complete: function(xhr,status) {

  //  alert(xhr.responseText);
  //  AjaxPlaceHolder.innerText = xhr.responseText;
  //},
  error: function (jqXHR, textStatus, errorThrown) {
    alert("You can not send Cross Domain AJAX requests: " +
errorThrown);
  }
})
}

function xmlParse(xml) {
  $(xml).find("item").each(function () {
    if ($(this).attr("type") == "MRFolder") {
      folder_name = $(this).attr("name");
      AjaxPlaceHolder.appendChild(document.createTextNode(folder_name +
"\n"));
    }
  });
}
}
</script>
</head>
<body>
  <h1>These are the top-level folders under the Respository</h1>
  <textarea id="AjaxPlaceHolder" name="AjaxPlaceHolder"
style="position:absolute; width:500px; height:500px;" ></textarea>
</body>
</html>

```

Running a WebFOCUS Report

This section provides code examples that demonstrate how to run the *Sales_for_a_Specific_Country* WebFOCUS report, which resides in the *RESTful_Web_Services/Car_Reports* folder. A successful sign-on request is a prerequisite for running this example, including retrieving the HTTP Header cookies from its response.

Visual Basic .NET Example

```
Imports System.Net
Imports System.IO
Imports System.Text
Dim request2 As HttpWebRequest
Dim response2 As HttpWebResponse
Dim webStream2 As Stream
Dim webResponse2 As String = ""
request2 =
WebRequest.Create("http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/
RESTful_Web_Services/Car_Reports/Sales_for_a_Specific_Country.fex")
request2.Method = "POST"
'cookies is defined as CookieContainer in the Signing-On to WebFOCUS example
request2.CookieContainer = cookies
postData = "IBIRS_action=run&COUNTRY=ENGLAND"
Dim byteArray2 As Byte() = Encoding.UTF8.GetBytes(postData)
request2.ContentType = "application/x-www-form-urlencoded"
request2.ContentLength = byteArray2.Length
Dim dataStream2 As Stream = request2.GetRequestStream()
dataStream2.Write(byteArray2, 0, byteArray2.Length)
dataStream2.Close()
response2 = request2.GetResponse()
webStream2 = response2.GetResponseStream()
Dim webStreamReader2 As New StreamReader(webStream2)
While webStreamReader2.Peek >= 0
    webResponse2 = webStreamReader2.ReadToEnd()
End While
WebBrowser1.DocumentText = webResponse2
```

Java Example

```

import java.awt.Frame;
import java.io.BufferedReader;
import java.io.File;
import java.io.FileOutputStream;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.PrintWriter;
import org.apache.commons.httpclient.*;
import org.apache.commons.httpclient.methods.*;
String request2 = "http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/RESTful_Web_
Services/Car_Reports/Sales_for_a_Specific_Country.fex";

PostMethod method_report = new PostMethod(request2);

method_report.addParameter("IBIRS_action", "run");
method_report.addParameter("COUNTRY", "ENGLAND");
// cookies is defined as Header[] in the Signing-On to WebFOCUS example
for(int h=0; h<cookies.length; h++){
    System.out.println(cookies[h]);
    method_report.setRequestHeader(cookies[h].getName(), cookies[h].getValue());
}

// client is defined as HttpClient in the Signing-On to WebFOCUS example
int statusCode2 = client.executeMethod(method_report);
InputStream rstream2 = null;

rstream2 = method_report.getResponseBodyAsStream();

File tempfile = new File("c:\\temp\\Report.htm");
FileOutputStream fos = new FileOutputStream(tempfile);
PrintWriter out=new PrintWriter(fos);
BufferedReader br2 = new BufferedReader(new InputStreamReader(rstream2));
String line2;
String newOutput = null;

while ((line2 = br2.readLine()) != null) {
    newOutput = line2;
    out.println(newOutput);
    System.out.println(line2);
}
br2.close();
out.close();

```

HTML and jQuery Example

```
<!DOCTYPE html>
<html>
<head>
  <title></title>
  <meta charset="utf-8" />
  <script type="text/javascript" src="http://code.jquery.com/jquery-3.1.0.js"> </
script>
  <script type='text/javascript' src="http://cdnjs.cloudflare.com/ajax/libs/jquery-
ajaxtransport-xdomainrequest/1.0.1/
jquery.xdomainrequest.min.js"></script>

  <script type="text/javascript">
    var csrf_name;
    var csrf_value;
    var frameToBeWorkedOn = "#AjaxPlaceHolder";
    var contentType = "application/x-www-form-urlencoded; charset=utf-8";
    $(document).ready(function (IBIRS_action, IBIRS_userName, IBIRS_password) {

      if (window.XDomainRequest)
        contentType = "text/plain";
      var webMethod = "http://machine:port/ibi_apps/rs";
      var IBIRS_action = "signOn";
      var IBIRS_userName = "admin";
      var IBIRS_password = "admin";
      var parameters = 'IBIRS_action=' + IBIRS_action + '&IBIRS_userName=' +
IBIRS_userName + '&IBIRS_password=' + IBIRS_password;
      $.ajax({
        type: "POST",
        url: webMethod,
        data: parameters,
        dataType: "xml",
        xhrFields: {
          withCredentials: true
        },
        crossDomain: true,
        contentType: contentType,
        success: xmlParser,
        error:function(jqXHR,textStatus,errorThrown)
          {
            alert("You can not send Cross Domain AJAX requests: " +
errorThrown);
          }
      })
    });
    function xmlParser(xml) {
```

```

$(xml).find("entry").each(function () {
    if ($(this).attr("key") == "IBI_CSRF_Token_Name") {
        csrf_name = $(this).attr("value");
    }
    if ($(this).attr("key") == "IBI_CSRF_Token_Value") {
        csrf_value = $(this).attr("value");
    }
});
runReport();
}
function runReport() {
    if (window.XDomainRequest)
        contentType = "text/plain";
    var webMethod = "http://machine:port/ibi_apps/rs/ibfs/WFC/Repository/Tests/
Revenue_by_Product_Category.fex";
    var IBIRS_action = "run";
    var BUSINESS_REGION = "'North America'";
    var BUSINESS_SUB_REGION = "'MidWest'";
    var parameters = 'IBIRS_action=' + IBIRS_action + '&BUSINESS_REGION=' +
BUSINESS_REGION + '&BUSINESS_SUB_REGION=' + BUSINESS_SUB_REGION + '&' + csrf_name +
'=' + csrf_value;
    $.ajax({
        type: "POST",
        url: webMethod,
        data: parameters,
        dataType: "html",
        xhrFields: {
            withCredentials: true
        },
        crossDomain: true,
        contentType: contentType,
        success: alert("success"), /*
        complete: function(xhr,status) {
        alert(xhr.responseText); /*
        $("AjaxPlaceHolder".html(xhr.responseText)); /*
        document.AjaxPlaceHolder.document.body.innerHTML =
xhr.responseText;
    },
    error: function (jqXHR, textStatus, errorThrown) {
        alert("You can not send Cross Domain AJAX requests: " +
errorThrown);
    }
    })
}
</script>
</head>
<body>
<iframe id="AjaxPlaceHolder" name="AjaxPlaceHolder" height="600" width="900"
align="middle" style="position:absolute; top: 5px; left: 5px"></iframe>
</body>
</html>

```

Handling Drill-downs, Active Cache, and On-Demand Paging Reports

This section provides code examples that demonstrate how to run an On-Demand Paging report called `ODP_Report.fex`, which resides in the `RESTful_Web_Services/Car_Reports` folder.

The examples include:

- ❑ A `signOn` page, which is used to run the initial request.
- ❑ A `WebForm2` page, which is used to make the additional RESTful Web Services requests required for the paging within the WebFOCUS report.

The `WebForm2` page can also be used as is to handle Drill-down and Active Cache paging requests.

The `signOn` page contains the RESTful Web Service request to run the initial WebFOCUS report. The `IBIRS_clientPath` parameter is set so that all additional RESTful Web Services requests needed, whether paging, image retrieval, or paging will be routed through the client application. For example:

```
IBIRS_clientPath=http://localhost:51970/WebForm2.aspx
```

Visual Basic .NET Example (`signOn.aspx` and `WebForm2.aspx`)

`signOn.aspx`

```
Imports System.Net
Imports System.IO
Public Class signOn
    Inherits System.Web.UI.Page
    Dim cookies As New CookieContainer
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load
    Dim webStream As Stream
    Dim webResponse As String = ""
    Dim request As HttpWebRequest
    Dim response1 As HttpWebResponse
    Dim postData As String
    request = WebRequest.Create("http://localhost.:8080/ibi_apps/rs/ibfs")

    request.Method = "POST"
    postData = "IBIRS_action=signOn&IBIRS_userName=admin&IBIRS_password=admin"
    request.CookieContainer = cookies
```

```

Dim byteArray As Byte() = Encoding.UTF8.GetBytes(postData)
request.ContentType = "application/x-www-form-urlencoded"
request.ContentLength = byteArray.Length
Dim dataStream As Stream = request.GetRequestStream()
dataStream.Write(byteArray, 0, byteArray.Length)
dataStream.Close()
responsel = request.GetResponse()
webStream = responsel.GetResponseStream()
Dim request2 As HttpWebRequest
Dim response2 As HttpWebResponse
Dim webStream2 As Stream
Dim webResponse2 As String = ""
Dim uri As New System.Uri("http://localhost.:8080/ibi_apps/rs")

request2 = WebRequest.Create(uri)
request2.Method = "POST"
request2.CookieContainer = cookies
postData = "IBIRS_action=run" + _
           "&IBIRS_clientPath=/WebForm2.aspx" + _
"&IBIRS_path=/WFC/Repository/RESTful_Web_Services/Car_Reports/ODP_Report.fex" + _
           "&IBIRS_service=ibfs" + _
           "&IBIRS_htmlPath=http://localhost:8080/ibi_apps/ibi_html"

Dim byteArray2 As Byte() = Encoding.UTF8.GetBytes(postData)
request2.ContentType = "application/x-www-form-urlencoded"

request2.ContentLength = byteArray2.Length
Dim dataStream2 As Stream = request2.GetRequestStream()
dataStream2.Write(byteArray2, 0, byteArray2.Length)
dataStream2.Close()
response2 = request2.GetResponse()
Dim i As Integer
Dim cookieArray As New CookieCollection
cookieArray = cookies.GetCookies(uri)
For i = 0 To cookies.Count - 1
    Dim aCookie As New HttpCookie(cookieArray(i).Name)
    aCookie.Value = cookieArray(i).Value
    Response.Cookies.Add(aCookie)
Next i
webStream2 = response2.GetResponseStream()
Dim webStreamReader2 As New StreamReader(webStream2)
While webStreamReader2.Peek >= 0
    webResponse2 = webStreamReader2.ReadToEnd()
End While
Response.Output.Write(webResponse2)
End Sub
End Class

```

WebForm2.aspx

```

Imports System.Net
Imports System.IO
Public Class WebForm2
    Inherits System.Web.UI.Page
    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load
    Dim tDrillURL As String = Request.ServerVariables("QUERY_STRING")
    Dim i As Integer
    Dim qParm As String
    Dim qValue As String
    Dim IBIRS_path As String = ""
    Dim Clicked_On As String = ""
    Dim cookies As New CookieContainer
    Dim request3 As HttpWebRequest
    Dim response3 As HttpWebResponse
    Dim webStream3 As Stream
    Dim webResponse3 As String = ""
    Dim getData As String
    Dim uris As String = "http://localhost.:8080/ibi_apps/rs"
    Dim uri As New System.Uri(uris)

getData = "http://localhost.:8080/ibi_apps/rs?" + _
        tDrillURL + _
        "&IBIRS_clientPath=/WebForm2.aspx" + _
        "&IBIRS_htmlPath=http://localhost:8080/ibi_apps/ibi_html"
request3 = WebRequest.Create(getData)
    request3.Method = "GET"

    Dim j As Integer
    For j = 0 To Request.Cookies.Count - 1
        Dim rCookie As New System.Net.Cookie
        rCookie.Name = Request.Cookies(j).Name
        rCookie.Value = Request.Cookies(j).Value
        cookies.Add(uri, rCookie)
        Dim aCookie As New HttpCookie(Request.Cookies(j).Name)
        aCookie.Value = Request.Cookies(j).Value
        Response.Cookies.Add(aCookie)
    Next j
    request3.CookieContainer = cookies
    response3 = request3.GetResponse()
    webStream3 = response3.GetResponseStream()
    Dim binaryReader3 As New BinaryReader(webStream3)
    Dim readData() As Byte = Nothing
    Dim byteArray() As Byte = Nothing
    Dim byteStart As Integer = 0
    Dim byteLength As Integer

```



```
While (True)
    readData = binaryReader3.ReadBytes(4096)
    If (readData.Length = 0) Then
        Exit While
    End If
    byteLength = readData.Length
    ReDim Preserve byteArray(byteLength + byteStart - 1)
    Array.Copy(readData, 0, byteArray, byteStart, byteLength)
    byteStart = byteStart + byteLength
End While
Response.OutputStream.Write(byteArray, 0, byteArray.Length)

End Sub
End Class
```

Java Example (signOn.jsp and WebForm2.jsp)

signOn.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1" session="true"
import="
java.io.BufferedReader,
java.io.IOException,
java.io.InputStream,
java.io.InputStreamReader,
java.io.File,
java.io.FileOutputStream,
java.io.PrintWriter,
java.net.URI,
java.net.URISyntaxException,
org.apache.commons.httpclient.*,
org.apache.commons.httpclient.methods.*,
sax.xml.parser.SaxHandler,
javax.xml.parsers.ParserConfigurationException,
javax.xml.parsers.SAXParser,
javax.xml.parsers.SAXParserFactory,
org.xml.sax.SAXException
"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/
html4/loose.dtd">
<%
String request1 = "http://localhost:8080/ibi_apps/rs/ibfs";
HttpClient client = new HttpClient();
PostMethod method = new PostMethod(request1);

method.addParameter("IBIRS_action", "signOn");
method.addParameter("IBIRS_userName", "admin");
method.addParameter("IBIRS_password", "admin");

client.executeMethod(method);
Header[] cookies = null;
InputStream responsel = null;

responsel = method.getResponseBodyAsStream();
cookies = method.getResponseHeaders("Set-Cookie");
SAXParserFactory factory = SAXParserFactory.newInstance();
SAXParser parser = factory.newSAXParser();
SaxHandler handler = new SaxHandler();
parser.parse(responsel, handler);
String csrfName = handler.getResults()[0].toString();
String csrfValue = handler.getResults()[1].toString();
// System.out.println("csrfName = " + csrfName);
// System.out.println("csrfValue = " + csrfValue);

```

```

String request2 = "http://localhost:8080/ibi_apps/rs";
PostMethod method_report = new PostMethod(request2);
method_report.addParameter("IBIRS_action", "run");
method_report.addParameter("IBIRS_clientPath", "/drillDownJSP/WebForm2.jsp");
method_report.addParameter("IBIRS_path", "/EDA/EDASERVE/ibisamp/carinst.fex");
method_report.addParameter("IBIRS_service", "ibfs");
method_report.addParameter("IBIRS_htmlPath", "http://localhost:8080/ibi_apps/
ibi_html");
method_report.addParameter(csrfName, csrfValue);
// cookies is defined as Header[] in the Signing-On to WebFOCUS example
for(int h=0; h<cookies.length; h++){
//   System.out.println(cookies[h]);
method_report.setRequestHeader(cookies[h].getName(), cookies[h].getValue());
String str = cookies[h].getName() + cookies[h].getValue();
//write cookie to a disk file and then read it back in the next JSP
String nameOfTextFile = "c:/temp/jsessionid.txt";
try {
    PrintWriter pw = new PrintWriter(new FileOutputStream(nameOfTextFile));
    pw.println(str);
    //clean up
    pw.close();
} catch(IOException e) {
    out.println(e.getMessage());
}
}

method_report.setRequestHeader("Content-type", "application/x-www-form-
urlencoded");
// client is defined as HttpClient in the Signing-On to WebFOCUS example
client.executeMethod(method_report);
InputStream response2 = null;
response2 = method_report.getResponseBodyAsStream();
BufferedReader br2 = new BufferedReader(new InputStreamReader(response2));
String line2;
String newOutput = null;
while ((line2 = br2.readLine()) != null) {
newOutput = line2;
out.println(newOutput);
//   System.out.println(line2);
}

%>

```

WebForm2.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"
import="
    java.io.BufferedReader,
    java.io.IOException,
    java.io.InputStream,
    java.io.InputStreamReader,
    java.io.File,
    java.io.FileOutputStream,
    java.io.PrintWriter,
    java.io.FileReader,
    java.net.URI,
    java.net.URISyntaxException,
    org.apache.commons.httpclient.*,
    org.apache.commons.httpclient.methods.*,
    sax.xml.parser.SaxHandler,
    javax.xml.parsers.ParserConfigurationException,
    javax.xml.parsers.SAXParser,
    javax.xml.parsers.SAXParserFactory,
    org.xml.sax.SAXException
"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/
html4/loose.dtd">
<%
    String tDrillURL = "";
    int i;
    String qParm;
    String qValue;
    String IBIRS_path = "";
    String Clicked_On = "";
    Header[] cookies = null;
    HttpClient client = new HttpClient();
    tDrillURL = request.getQueryString();
//    read saved cookie from text file
    String txtFilePath = "c:/temp/jsessionId.txt";
    BufferedReader reader = new BufferedReader(new FileReader(txtFilePath));
    StringBuilder sb = new StringBuilder();
    String line;
    while((line = reader.readLine())!= null){
        sb.append(line);
    }
//    System.out.println(sb.toString());
    reader.close();

```

```

String request3 = "http://localhost:8080/ibi_apps/rs";

GetMethod method_report2 = new GetMethod(request3);
method_report2.setQueryString(tDrillURL);
method_report2.getParams().setParameter("IBIRS_clientPath", "/drillDownJSP/
WebForm2.jsp");
method_report2.getParams().setParameter("IBIRS_htmlPath", "http://localhost:8080/
ibi_apps/ibi_html");
String cookie=sb.toString();
// System.out.println("webform2 cookie before replace " + cookie);
cookie = cookie.replace("Set-Cookie","");
// System.out.println("webform2 cookie after replace " + cookie);
method_report2.setRequestHeader("Cookie", cookie);

// }
method_report2.setRequestHeader("Content-type", "application/x-www-form-
urlencoded");

int statusCode = client.executeMethod(method_report2);
System.out.println(statusCode);
InputStream response3 = null;
response3 = method_report2.getResponseBodyAsStream();
BufferedReader br2 = new BufferedReader(new InputStreamReader(response3));
String line3;
String newOutput = null;
while ((line3 = br2.readLine()) != null) {
newOutput = line3;
out.println(newOutput);
}
%>

```

HTML and jQuery Example (drillOne.html and drillTwo.html)

drillOne.html

```

<!DOCTYPE html>
<html>
<head>
  <title></title>
  <meta charset="utf-8" />
  <script type="text/javascript" src="http://code.jquery.com/jquery-3.1.0.js"> </
script>
  <script type='text/javascript' src="http://cdnjs.cloudflare.com/ajax/libs/jquery-
ajaxtransport-xdomainrequest/1.0.1/
jquery.xdomainrequest.min.js"></script>
  <script type="text/javascript">
    var csrf_name;
    var csrf_value;
    var frameToBeWorkedOn = "#AjaxPlaceholder";
    var contentType = "application/x-www-form-urlencoded; charset=utf-8";
    $(document).ready(function (IBIRS_action, IBIRS_userName, IBIRS_password) {
      if (window.XDomainRequest)
        contentType = "text/plain";
      var webMethod = "http://machine:port/ibi_apps/rs";
      var IBIRS_action = "signOn";
      var IBIRS_userName = "admin";
      var IBIRS_password = "admin";
      var parameters = 'IBIRS_action=' + IBIRS_action + '&IBIRS_userName=' +
IBIRS_userName + '&IBIRS_password=' + IBIRS_password;
      $.ajax({
        type: "POST",
        url: webMethod,
        data: parameters,
        dataType: "xml",
        xhrFields: {
          withCredentials: true
        },
        crossDomain: true,
        contentType: contentType,
        success: xmlParser,
        error:function(jqXHR,textStatus,errorThrown)
        {
          alert("You can not send Cross Domain AJAX requests: " +
errorThrown);
        }
      })
    });
    function xmlParser(xml) {
      $(xml).find("entry").each(function () {

```

```

        if ($(this).attr("key") == "IBI_CSRF_Token_Name") {
            csrf_name = $(this).attr("value");
        }
        if ($(this).attr("key") == "IBI_CSRF_Token_Value") {
            csrf_value = $(this).attr("value");
        }
    });
    runReport();
}
function runReport() {
    if (window.XDomainRequest)
        contentType = "text/plain";
    var webMethod = "http://machine:port/ibi_apps/rs";
    var IBIRS_action = "run";
    var IBIRS_clientPath = "/src/drillTwo.html";
    var IBIRS_path = "/EDA/EDASERVE/ibisamp/carinst.fex";
    var IBIRS_service = "ibfs";
    var IBIRS_htmlPath = "http://machine:port/ibi_apps/ibi_html";
    var parameters = 'IBIRS_action=' + IBIRS_action + '&IBIRS_clientPath=' +
IBIRS_clientPath + '&IBIRS_path=' + IBIRS_path
        + '&IBIRS_service=' + IBIRS_service + '&IBIRS_htmlPath=' +
IBIRS_htmlPath + '&' + csrf_name + '=' + csrf_value;
    $.ajax({
        type: "POST",
        url: webMethod,
        data: parameters,
        dataType: "html",
        xhrFields: {
            withCredentials: true
        },
        crossDomain: true,
        contentType: contentType,
        success: alert("success"), /*
    /*
    complete: function(xhr,status) {
        alert(xhr.responseText); /*
    /*
        $("AjaxPlaceHolder".html(xhr.responseText)); /*
        document.AjaxPlaceHolder.document.body.innerHTML =
xhr.responseText;
    },
    error: function (jqXHR, textStatus, errorThrown) {
        alert("You can not send Cross Domain AJAX requests: " +
errorThrown);
    }
    })
}
</script>
</head>
<body>
    <iframe id="AjaxPlaceHolder" name="AjaxPlaceHolder" height="600" width="900"
align="middle" style="position:absolute; top: 5px; left: 5px"></iframe>
</body>
</html>

```

drillTwo.html

```

<!DOCTYPE html>
<html>
<head>
  <title></title>
  <meta charset="utf-8" />
  <script type="text/javascript" src="http://code.jquery.com/jquery-3.1.0.js"> </
script>
  <script type='text/javascript' src="http://cdnjs.cloudflare.com/ajax/libs/jquery-
ajaxtransport-xdomainrequest/1.0.1/
jquery.xdomainrequest.min.js"></script>
  <script type="text/javascript">
    var frameToBeWorkedOn = "#AjaxPlaceHolder";
    var contentType = "application/x-www-form-urlencoded; charset=utf-8";
    var tDrillURLx = window.location.search;
    var tDrillURL = tDrillURLx.slice(1);

    $(document).ready(function () {
      if (window.XDomainRequest)
        contentType = "text/plain";
      var webMethod = "http://machine:port/ibi_apps/rs";
      var IBIRS_action = "get";
      var IBIRS_clientPath = "/src/drillTwo.html";
      var IBIRS_htmlPath = "http://machine:port/ibi_apps/ibi_html";
      var parameters = tDrillURL + '&IBIRS_clientPath=' + IBIRS_clientPath +
'&IBIRS_htmlPath=' + IBIRS_htmlPath;
      $.ajax({
        type: "GET",
        url: webMethod,
        data: parameters,
        dataType: "html",
        xhrFields: {
          withCredentials: true
        },
        crossDomain: true,
        contentType: contentType,
        success: alert("success"), /*
        complete: function(xhr,status) {

```



```

/*          alert(xhr.responseText); */
/*          $("AjaxPlaceHolder".html(xhr.responseText)); */
document.AjaxPlaceHolder.document.body.innerHTML =
xhr.responseText;
    },
    error: function (jqXHR, textStatus, errorThrown)
{
    alert("You can not send Cross Domain AJAX requests: " +
errorThrown);
}
    })
})
</script>
</head>
<body>
    <iframe id="AjaxPlaceHolder" name="AjaxPlaceHolder" height="600" width="900"
align="middle" style="position:absolute; top: 5px; left: 5px"></iframe>
</body>
</html>

```

Parsing the XML Response of a SignOn Request to Obtain the CSRF Name and Value

This section provides code examples that demonstrate how to parse the XML response of a SignOn request to obtain the Cross-Site Request Forgery (CSRF) name and value. The CSRF name and value can then be sent to subsequent POST requests.

Java Example

```
import java.awt.Desktop;
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.File;
import java.io.FileOutputStream;
import java.io.PrintWriter;
import java.net.URI;
import java.net.URISyntaxException;
import javax.xml.parsers.ParserConfigurationException;
import javax.xml.parsers.SAXParser;
import javax.xml.parsers.SAXParserFactory;
import org.apache.commons.httpclient.Header;
import org.apache.commons.httpclient.HttpClient;
import org.apache.commons.httpclient.HttpException;
import org.apache.commons.httpclient.methods.PostMethod;
import org.xml.sax.SAXException;
/**
 * @author
 *
 */
public class runReport
{
    /**
     * @param args
     * @throws IOException
     * @throws HttpException
     * @throws SAXException
     * @throws ParserConfigurationException
     * @throws URISyntaxException
     */
    public static void main(String[] args) throws HttpException, IOException,
    ParserConfigurationException, SAXException, URISyntaxException
    {
        String request = "http://localhost:8080/ibi_apps/rs/ibfs";
        HttpClient client = new HttpClient();
        PostMethod method = new PostMethod(request);
        method.addParameter("IBIRS_action", "signOn");
        method.addParameter("IBIRS_userName", "admin");
        method.addParameter("IBIRS_password", "admin");
    }
}
```

```

client.executeMethod(method);
Header[] cookies = null;
InputStream rstream = null;
rstream = method.getResponseBodyAsStream();
cookies = method.getResponseHeaders("Set-Cookie");
/* parse rstream XML for csrf token */
SAXParserFactory factory = SAXParserFactory.newInstance();
SAXParser parser = factory.newSAXParser();
SaxHandler handler = new SaxHandler();
parser.parse(rstream, handler);
String csrfName = SaxHandler.results[0];
String csrfValue = SaxHandler.results[1];
System.out.println("csrfName = " + csrfName);
System.out.println("csrfValue = " + csrfValue);
String request2 = "http://localhost:8080/ibi_apps/rs/ibfs/WFC/Repository/Tests/
car_param.fex";
PostMethod method_report = new PostMethod(request2);
method_report.addParameter("IBIRS_action", "run");
method_report.addParameter("COUNTRY", "ENGLAND");
method_report.addParameter("CAR", "JAGUAR");
method_report.addParameter("MODEL", "XJ12L AUTO");
method_report.addParameter(csrfName, csrfValue);
// cookies is defined as Header[] in the Signing-On to WebFOCUS example
for(int h=0; h<cookies.length; h++){
System.out.println(cookies[h]);
method_report.setRequestHeader(cookies[h].getName(), cookies[h].getValue());
}
// client is defined as HttpClient in the Signing-On to WebFOCUS example
int statusCode2 = client.executeMethod(method_report);
InputStream rstream2 = null;
rstream2 = method_report.getResponseBodyAsStream();
File tempfile = new File("c:\\temp\\Report.htm");
FileOutputStream fos = new FileOutputStream(tempfile);
PrintWriter out=new PrintWriter(fos);
BufferedReader br2 = new BufferedReader(new InputStreamReader(rstream2));
String line2;
String newOutput = null;
while ((line2 = br2.readLine()) != null) {
newOutput = line2;
out.println(newOutput);
System.out.println(line2);
}
// bring up the HTML report in the default browser
URI xtempfile = new URI ("file:/c:/temp/Report.htm");
Desktop.getDesktop().browse(xtempfile);
br2.close();
out.close();
}
}

```

XML Parser Class

The XML Parser class is called *SaxHandler* and is in a separate class file

```

import org.xml.sax.Attributes;
import org.xml.sax.SAXException;
import org.xml.sax.helpers.DefaultHandler;
public class SaxHandler extends DefaultHandler {

    static String[] results = new String[2];

    public void startElement(String uri, String localName, String qName,
Attributes attributes)
throws SAXException {
        if (qName.equals("entry")) {
            String keyName = attributes.getValue("key");
            if (keyName.equals("IBI_CSRF_Token_Name")) {
                String tokenNameKeyValue =
attributes.getValue("value");
                System.out.println("key value is " +
tokenNameKeyValue);
                results[0] = tokenNameKeyValue;
            }
            if (keyName.equals("IBI_CSRF_Token_Value")) {
                String tokenValueKeyValue =
attributes.getValue("value");
                System.out.println("key value is " +
tokenValueKeyValue)
                results[1] = tokenValueKeyValue;
            }
        }
    }
}

```

Visual Basic .NET Example

```
Imports System.Net
Imports System.Text
Imports System.IO
Module Module1
    Sub Main()
        Dim cookies As New CookieContainer
        Dim webStream As Stream
        Dim webResponse As String = ""
        Dim request As HttpWebRequest
        Dim response As HttpWebResponse
        Dim postData As String
        Dim csrf(2) As String
        request = WebRequest.Create("http://localhost:8080/ibi_apps/rs/ibfs")
        request.Method = "POST"
        postData = "IBIRS_action=signOn&IBIRS_userName=admin&IBIRS_password=admin"
        request.CookieContainer = cookies
        Dim byteArray As Byte() = Encoding.UTF8.GetBytes(postData)
        request.ContentType = "application/x-www-form-urlencoded"
        request.ContentLength = byteArray.Length
        Dim dataStream As Stream = request.GetRequestStream()
        dataStream.Write(byteArray, 0, byteArray.Length)
        dataStream.Close()
        response = request.GetResponse()
        webStream = response.GetResponseStream()
        Dim webStreamReader As New StreamReader(webStream)
        While webStreamReader.Peek >= 0
            webResponse = webStreamReader.ReadToEnd()
        End While
        csrf = XMLParse.XMLParseCSRF.doParseXML(webResponse)
        Console.WriteLine("csrf token name is " + csrf(0))
        Console.WriteLine("csrf key value is " + csrf(1))
        Console.ReadKey()
        Dim request2 As HttpWebRequest
        Dim response2 As HttpWebResponse
        Dim webStream2 As Stream
        Dim webResponse2 As String = ""
        request2 = WebRequest.Create("http://localhost:8080/ibi_apps/rs/ibfs/WFC/
Repository/Tests/car_param.fex")
        request2.Method = "POST"
        'cookies is defined as CookieContainer in the Signing-On to WebFOCUS example
        request2.CookieContainer = cookies
        postData = "IBIRS_action=run&COUNTRY=ENGLAND&CAR=JAGUAR&MODEL=XJ12L%20AUTO" +
"&" + csrf(0) + "=" + csrf(1)
        Dim byteArray2 As Byte() = Encoding.UTF8.GetBytes(postData)
        request2.ContentType = "application/x-www-form-urlencoded"
        request2.ContentLength = byteArray2.Length
```

```
Dim dataStream2 As Stream = request2.GetRequestStream()
dataStream2.Write(byteArray2, 0, byteArray2.Length)
dataStream2.Close()
response2 = request2.GetResponse()
webStream2 = response2.GetResponseStream()
'Write to disk
Dim fs As New FileStream("c:\temp\output.htm", FileMode.Create)
Dim read As Byte() = New Byte(255) {}
Dim count As Integer = webStream2.Read(read, 0, read.Length)
While count > 0
    fs.Write(read, 0, count)
    count = webStream2.Read(read, 0, read.Length)
End While
'Close everything
fs.Close()
webStream2.Close()
Process.Start("c:\temp\output.htm")
End Sub
End Module
```

XML Parser Function

The XML Parser function is called *doParseXML* and is located in a separate class file named *XMLParseCSRF.vb*, which is located in a separate project for reusability.

```
Imports System.IO
Imports System.Xml
Public Class XMLParseCSRF
    Public Shared Function doParseXML(inResponse As String) As String()
        Dim results(2) As String
        Dim m_xmlr As XmlTextReader = New XmlTextReader(New StringReader(inResponse))
        While m_xmlr.Read()
            If (m_xmlr.NodeType = XmlNodeType.Element) Then
                If m_xmlr.Name = "entry" Then
                    Dim keyName As String = m_xmlr.GetAttribute("key")
                    If (keyName = "IBI_CSRF_Token_Name") Then
                        Dim tokenKeyNameValue As String = m_xmlr.GetAttribute("value")
                        Console.WriteLine("tokenKeyName value is " + tokenKeyNameValue)
                        results(0) = tokenKeyNameValue
                    End If
                    If (keyName = "IBI_CSRF_Token_Value") Then
                        Dim tokenValueKeyValue As String = m_xmlr.GetAttribute("value")
                        Console.WriteLine("tokenValueKey value is " +
                            tokenValueKeyValue)
                        results(1) = tokenValueKeyValue
                    End If
                End If
            End If
        End While
        'close the reader
        m_xmlr.Close()
        Return results
    End Function
End Class
```


This section describes the format and structure of URL calls that can be used to directly access WebFOCUS InfoAssist.

In this chapter:

- ❑ [Starting InfoAssist](#)

Starting InfoAssist

This URL call can be used to start InfoAssist externally.

URL Format:

```
http://host:port/ibi_apps/ia?tool=<tool_value>&is508=<true/false>&master=<master_name>&item=<item_ibfs_path>
```

where:

host

Is the name of the system where WebFOCUS is installed.

port

Is the port number used by WebFOCUS.

item (required)

Is the IBFS path to one of the following:

- ❑ Folder
- ❑ Chart
- ❑ Table
- ❑ InfoDiscovery Fex
- ❑ Link to a Folder
- ❑ Link to a Chart
- ❑ Link to a Table
- ❑ Link to an InfoDiscovery Fex

Note: The specified value for the *item* parameter must be encoded using UTF-8.

tool (optional)

If the *item* parameter is set to a folder or a link to a folder, then this specified value is the tool (mode) to start when InfoAssist is launched:

report (default)

Starts in *Report* mode.

chart

Starts in *Chart* mode.

idis

Starts in *DataVisualization* mode.

document

Starts in *Compose* mode.

dashboard

Starts in *DashBoard* mode.

sample

Used to create samples. In this case, the *item* parameter must be set to a folder and a specific Master File must be specified for the *master* parameter.

is508 (Optional)

Determines whether to start InfoAssist in 508-compliance mode. Specify *true* or *false*.

master (Optional)

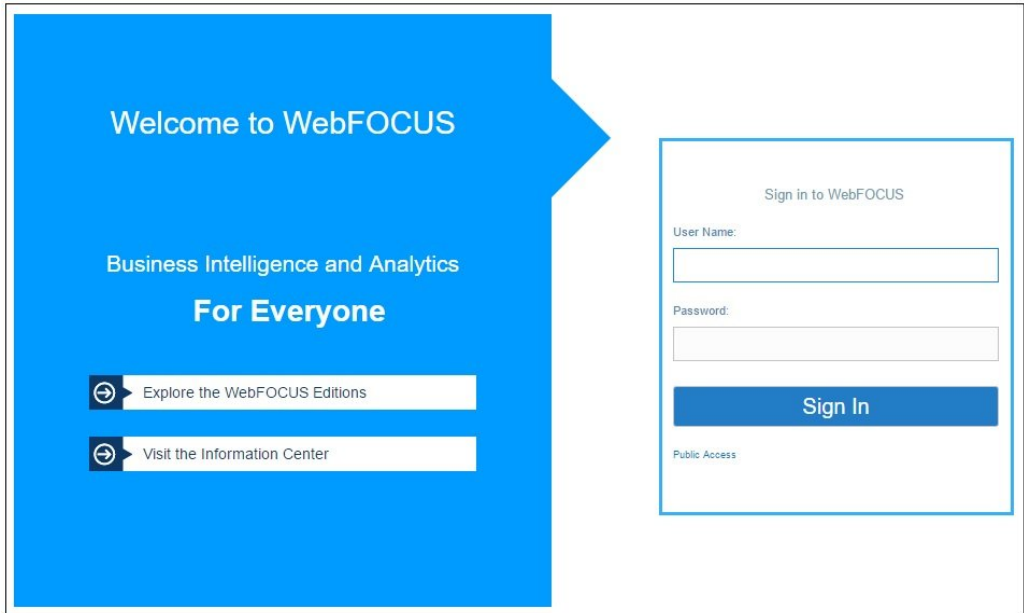
If the *item* parameter is set to a folder or a link to a folder, then this specified value is the Master File to use.

Example:

```
http://host:port/ibi_apps/ia?tool=chart&master=CAR&item=IBFS%3A%2FWFC%2FRepository%2FPublic%2Fbig14%2FChart1.fex
```

In this example, InfoAssist is started in Chart mode using the CAR Master File. Since this is a secure URL, the WebFOCUS Sign In page is initially displayed, as shown in the following image. The user must specify a valid user name and password before proceeding.

Note: The WebFOCUS Sign In page only displays if the user is not already signed in. If the user is already signed in to WebFOCUS, then this page is not displayed.



To bypass the WebFOCUS Sign In page, the developer of the application can use an alternate sign on procedure, such as a web service signOn call, or any SSO option. For more information, see the *WebFOCUS Security and Administration* content.

WebFOCUS Open Portal Services

WebFOCUS Open Portal Services provides seamless integration to Enterprise Information Portals (EIPs) through a single sign-on (SSO) so that users can consume and interact with WebFOCUS content in an easy and secure way.

Information Builders offers a way to leverage your EIP investment by extending access to your enterprise data. With WebFOCUS Open Portal Services, you can deploy WebFOCUS business intelligence across the enterprise by incorporating reporting structures and structured content within supported third-party EIPs.

This section describes how to install and use WebFOCUS Open Portal Services and its portal components in the Microsoft® SharePoint 2016 and 2013 Portal Server environments. It is intended for administrators who are installing and configuring the WebFOCUS Portal components on a specific EIP to enable the usage and delivery of WebFOCUS business intelligence content in third-party applications.

Introducing WebFOCUS Open Portal Services

WebFOCUS Open Portal Services for WebFOCUS 8 provides seamless integration to Enterprise Information Portals (EIPs) through a Single Sign On so that users can consume and interact with WebFOCUS content in an easy and secure way.

Information Builders offers a way to leverage your EIP investment by extending access to your enterprise data. With WebFOCUS Open Portal Services, you can deploy WebFOCUS business intelligence across the enterprise by incorporating reporting structures and structured content within supported third-party EIPs.

Information Builders WebFOCUS business intelligence technology is the most usable, deployable, and scalable business intelligence software solution for accessing more than 90+ data sources, including legacy, data warehouse, enterprise resource planning (ERP), and customer relationship management (CRM), on over 35 platforms, including S/390 mainframe. Because it can access and integrate data from any source, it reduces the complexity of a given data environment.

The following section provides an overview of the features and benefits of WebFOCUS Open Portal Services.

In this chapter:

- ❑ [WebFOCUS Open Portal Services](#)
 - ❑ [Benefits of Using WebFOCUS Open Portal Services](#)
 - ❑ [Java Portlet Specification 2.0 \(JSR 286\) Support](#)
-

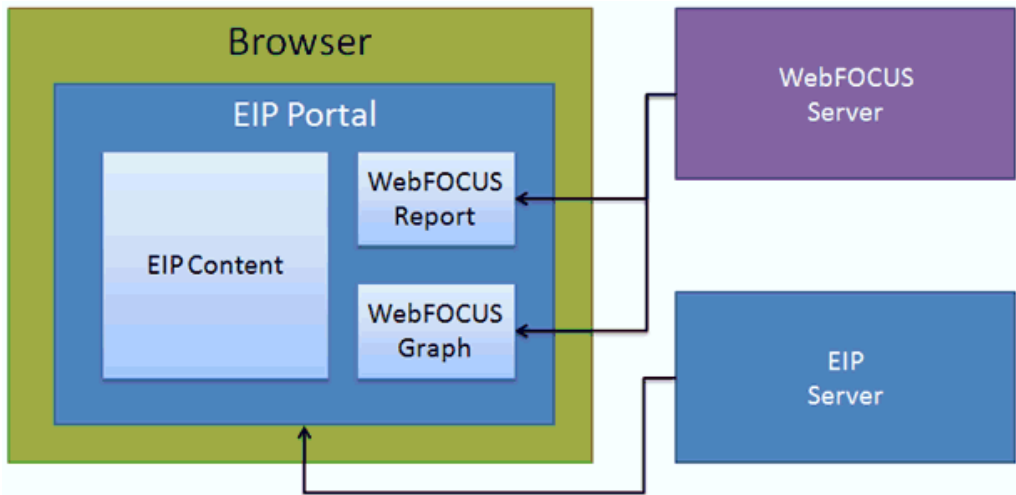
WebFOCUS Open Portal Services

WebFOCUS Open Portal Services enables you to extend WebFOCUS business intelligence capabilities to end users within an existing EIP framework. Users can:

- ❑ Personalize the way they view, store, and retrieve business intelligence content for optimum job efficiency.
- ❑ Decide what content to include in each content window and how that content is displayed and organized.
- ❑ Combine graphics and reports in the same portal page.

- ❑ Employ analytic tools for all types of reporting and query, including ad hoc and OLAP.

The following diagram illustrates WebFOCUS content being displayed inside an EIP:



Benefits of Using WebFOCUS Open Portal Services

WebFOCUS Open Portal Services offer many benefits to the users within the enterprise by:

- ❑ Providing immediate access to critical enterprise-wide data through a personalized portal page.
- ❑ Incorporating comprehensive reporting and analysis capabilities within the portal and enhancing the user experience by delivering relevant, real-time information.
- ❑ Enabling users to display, locate, share, visualize, and analyze business intelligence information based on their roles within the enterprise.
- ❑ Supporting internal and external security for delivering timely and accurate business reports to authorized users.

Java Portlet Specification 2.0 (JSR 286) Support

Java Portlet Specification 2.0 (JSR 286) establishes a standard API for creating portlets, the integration component between applications and portals that enables delivery of an application through a portal.

Released in June 2008, JSR 286 is the successor to the Java Portlet Specification 1.0 (JSR 168), which was originally released in October 2003. JSR 286 provides new features and improvements and fills any gaps that were identified with JSR 168.

As of WebFOCUS Release 8.2 Version 01, WebFOCUS portlets are available for JSR 286 compliant portal environments, such as IBM WebSphere Portal Server.

Using WebFOCUS Portal Components

This section describes the types of WebFOCUS components that are provided by WebFOCUS Open Portal Services (OPS). In addition, information on using WebFOCUS components is provided.

In this chapter:

- [WebFOCUS Open Portal Services Components Overview](#)
 - [Using WebFOCUS Open Portal Services Components](#)
 - [Usage Considerations](#)
-

WebFOCUS Open Portal Services Components Overview

WebFOCUS Open Portal Services (OPS) provides the following set of WebFOCUS components:

- Report
- Deferred Status
- Resource Tree
- Portal
- Portal Tree

These components enable integration between existing Enterprise Information Portals (EIPs) and the WebFOCUS business intelligence platform using Single Sign-On (SSO) functionality. As a result, users are automatically authenticated with WebFOCUS once they log on to their EIP. After receiving the user ID from the EIP through a secure channel, WebFOCUS OPS initiates a trusted Managed Reporting logon on behalf of the user to avoid the subsequent Managed Reporting logon prompt.

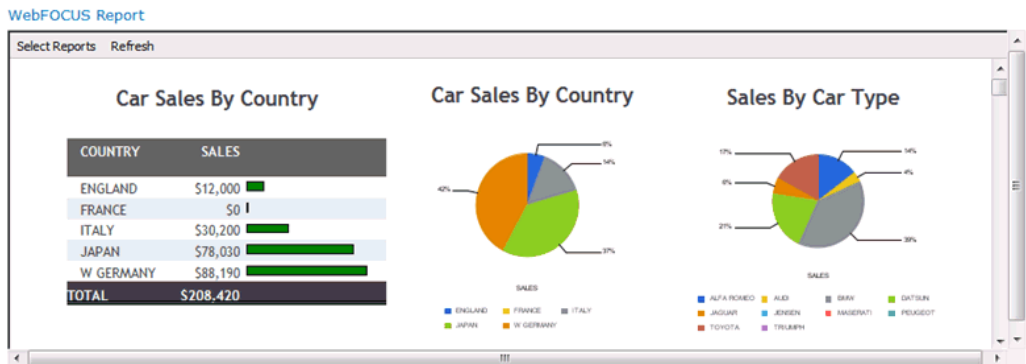
For more information about the trusted Managed Reporting logon feature, see the *WebFOCUS Security and Administration* content.

WebFOCUS Report Component

The WebFOCUS Report component allows portal users to access WebFOCUS content items that include reports, charts, dashboards, documents, and URLs. Depending on security privileges, a user has the following options:

- Select their own WebFOCUS content items to be displayed.
- Select the specific WebFOCUS content items to be displayed to other users.
- View only WebFOCUS content items already selected by another user.

The following image shows an example of the WebFOCUS Report component.



WebFOCUS Deferred Status Component

The WebFOCUS Deferred Status component allows users to check the status of any report submitted for deferred execution. The following image shows an example of the WebFOCUS Deferred Status component.

WebFOCUS Deferred Status

WebFOCUS Deferred Report Status as of Monday, August 26, 2013 11:25:37 AM Information Builders.

Refresh Sort By Date Delete Help

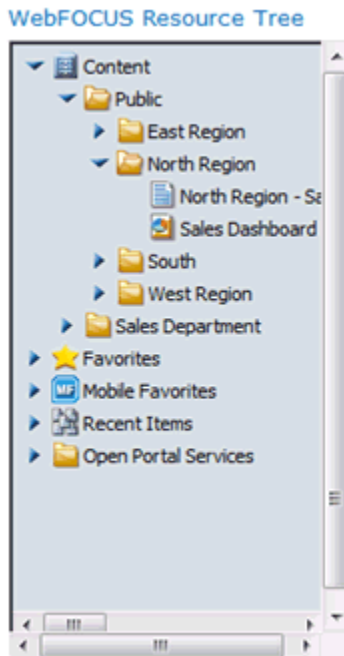
Refresh every seconds. (min. 5 seconds) Enable Refresh:

Date/Time Submitted	Domains	Description	Expires In	Options
Wednesday, August 21, 2013 2:23:38 PM	Public	Deferred ticket of 'Country Sales'	25 days	Delete View Save Run
Wednesday, August 21, 2013 2:20:57 PM	Public	Deferred ticket of 'Country Sales'	25 days	Delete View Save Run
Thursday, August 01, 2013 12:08:40 AM	Public	Deferred ticket of 'Summary Sales Chart'	4 days	Delete View Save Run

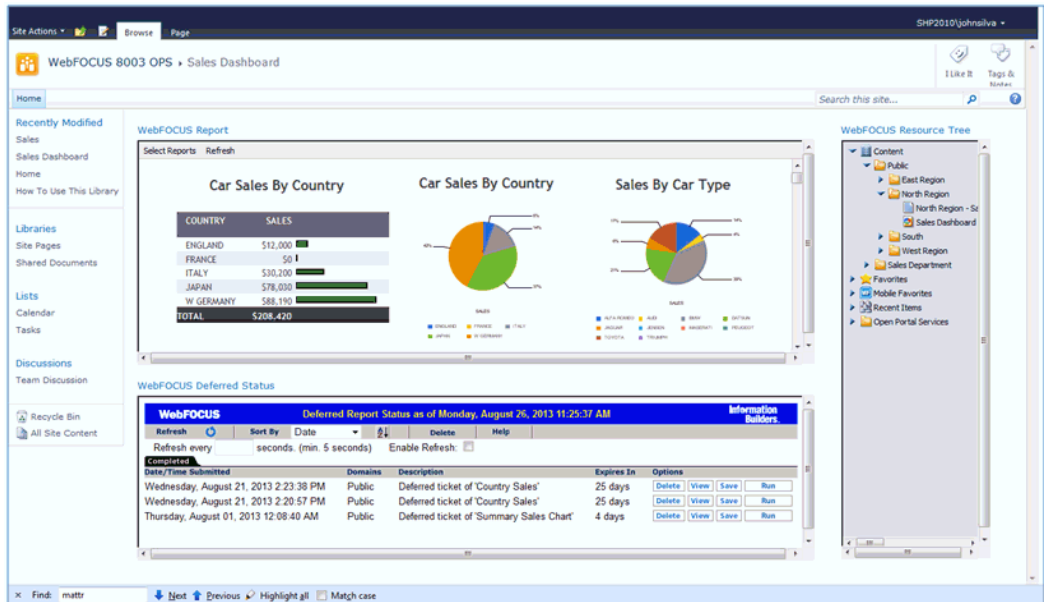
WebFOCUS Resource Tree Component

The WebFOCUS Resource Tree component is a modified version of the standard WebFOCUS Resource Tree located in the WebFOCUS Business Intelligence (BI) Portal. It allows EIP users to run, build, and modify WebFOCUS content items. In addition, functionality such as scheduling reports, enabling content to be viewed on an iPad is provided. If WebFOCUS ReportCaster is installed, then links to the ReportCaster Library and additional ReportCaster resources are available.

The following image shows an example of the WebFOCUS Resource Tree component.



The following image shows an example of the WebFOCUS Report, Deferred Status, and Resource Tree components being used on a single website page in Microsoft SharePoint.



WebFOCUS Portal Component

The WebFOCUS Portal component allows users to access their WebFOCUS Business Intelligence (BI) portal content in a portal environment (for example, IBM WebSphere Portal Server). The portal launches in the WebFOCUS Portal component so the portal is open and ready to be used.

WebFOCUS Portal Tree Component

The WebFOCUS Portal Tree component shows any basic WebFOCUS portals in a portal environment (for example, IBM WebSphere Portal Server).

Using WebFOCUS Open Portal Services Components

This section describes how to configure and use the WebFOCUS Report, Deferred Status, and Resource Tree components once the *Source URL* parameter is set.

For more information on how to add WebFOCUS Open Portal Services (OPS) components to a portal page and access its properties, see [Installing WebFOCUS Web Parts for Microsoft SharePoint 2013](#) on page 385.

Setting the Source URL Parameter

After you add a component to the portal page, the first common step that is required for all three WebFOCUS components is to set the *Source URL* parameter for the component to the URL of the WebFOCUS Client to be used. Once this URL is set, the component is able to communicate with WebFOCUS and respond with the initial screen of the component.

Using the WebFOCUS Report Component

Once the WebFOCUS Report component is added to a portal page and the WebFOCUS Client is selected as described in [Setting the Source URL Parameter](#) on page 359, the component content must be defined. This is done by a user who has permission to select WebFOCUS Report component content items in one of the following display modes:

- Launch
- Folder
- List

For more information on configuring WebFOCUS Report component security, see [WebFOCUS Report Component Configurations](#) on page 369.

For more information on adding the WebFOCUS Report component to a portal page, see [Installing WebFOCUS Web Parts for Microsoft SharePoint 2013](#) on page 385.

Launch Mode

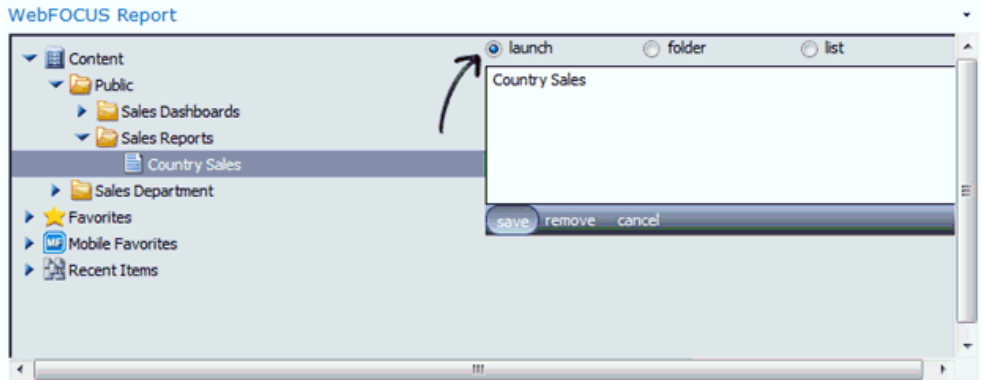
In this mode, you can set a WebFOCUS content item to be displayed by default each time a user visits the portal page containing the WebFOCUS Report component.

Procedure: How to Configure Launch Mode for the WebFOCUS Report Component

1. Click *Select Reports* from the Component menu bar, as shown in the following image.



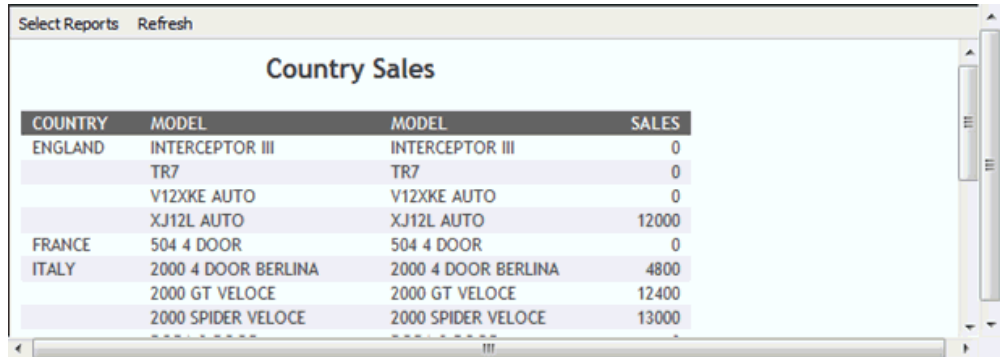
2. Select the *launch* option from the display mode options in the upper-right corner, as shown in the following image.



3. Select a content item (report) to be displayed from the Content node on the left pane and then click save, as shown in the following image.



Subsequent visits to the WebFOCUS Report component will display the selected report by default, as shown in the following image.



COUNTRY	MODEL	MODEL	SALES
ENGLAND	INTERCEPTOR III	INTERCEPTOR III	0
	TR7	TR7	0
	V12XKE AUTO	V12XKE AUTO	0
FRANCE	XJ12L AUTO	XJ12L AUTO	12000
	504 4 DOOR	504 4 DOOR	0
ITALY	2000 4 DOOR BERLINA	2000 4 DOOR BERLINA	4800
	2000 GT VELOCE	2000 GT VELOCE	12400
	2000 SPIDER VELOCE	2000 SPIDER VELOCE	13000

Folder Mode

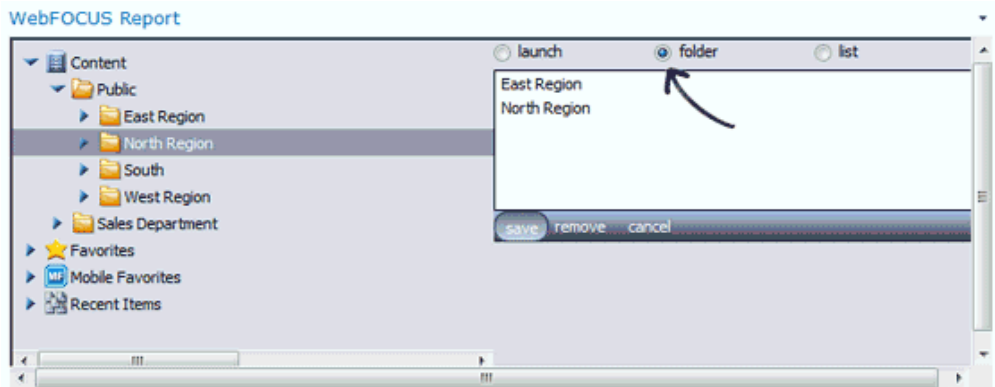
In this mode, you can set a specific WebFOCUS folder to be displayed by default each time a user visits the portal page containing the WebFOCUS Report component. Users can then select the WebFOCUS content item they want to run by expanding the displayed folder.

Procedure: How to Configure Folder Mode for the WebFOCUS Report Component

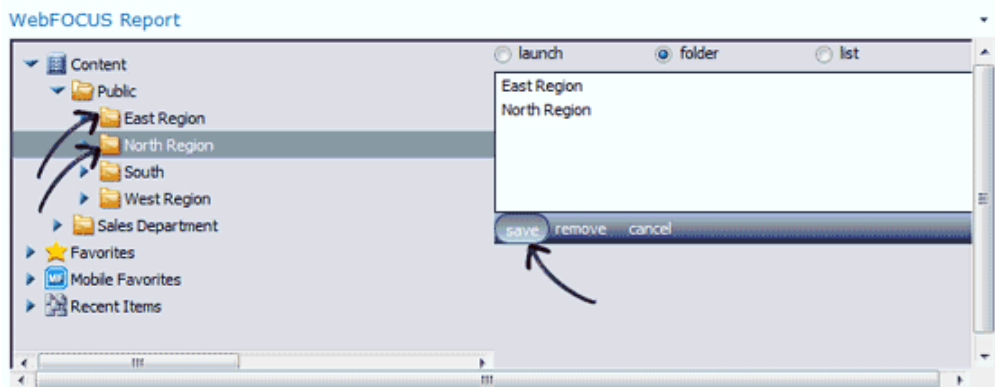
1. Click *Select Reports* from the Component menu bar, as shown in the following image.



2. Select the *folder* option from the display mode options in the upper-right corner, as shown in the following image.



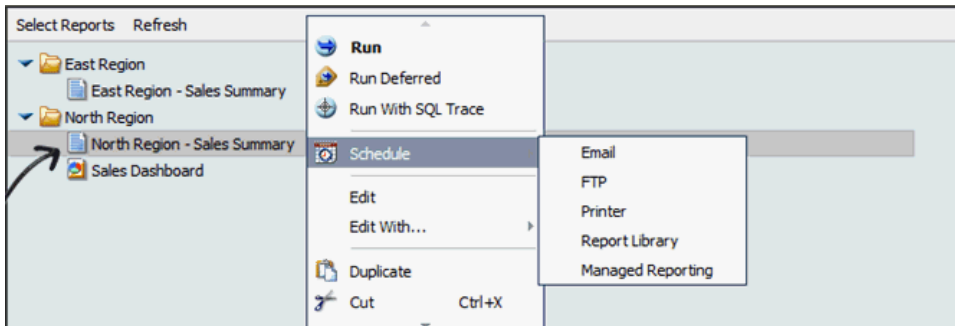
3. Select one or more folders from the Content node on the left pane and click save, as shown in the following image.



Subsequent visits to the WebFOCUS Report component will display the selected folders and respective content items by default, as shown in the following image.



4. Double-click a content item to run it or right-click a content item to select different options (such as Run, Edit, and Schedule), as shown in the following image.



List Mode

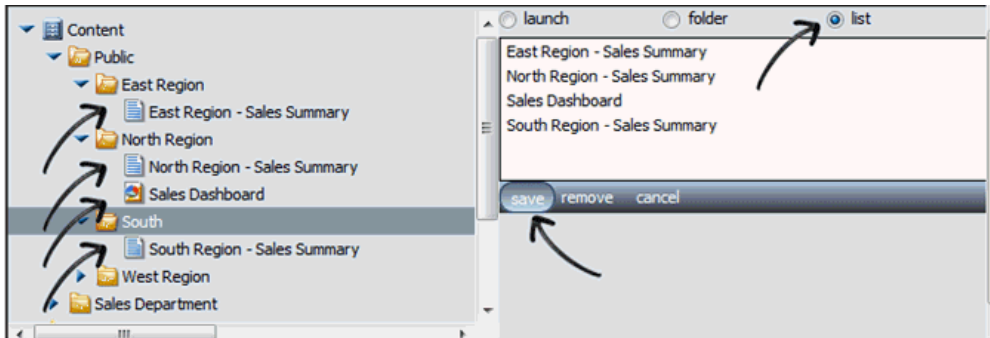
In this mode, you can set a list of WebFOCUS content items to be displayed by default each time a user visits the portal page containing the WebFOCUS Report component. Users can then select one of the content items to run.

Procedure: How to Configure List Mode for the WebFOCUS Report Component

1. Click *Select Reports* from the Component menu bar, as shown in the following image.

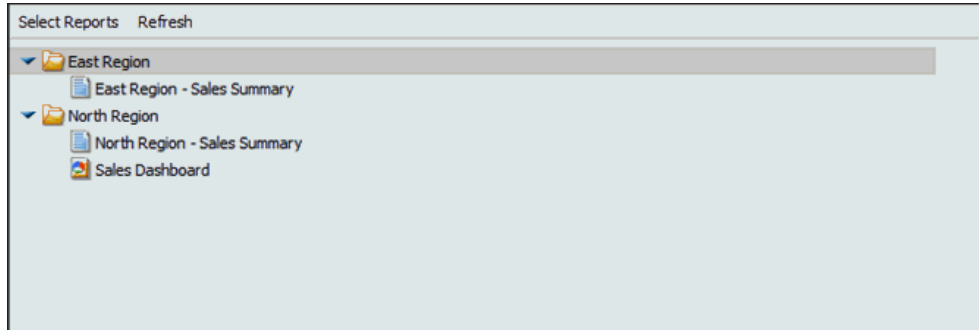


2. Select the *list* option from the display mode options in the upper-right corner, as shown in the following image.

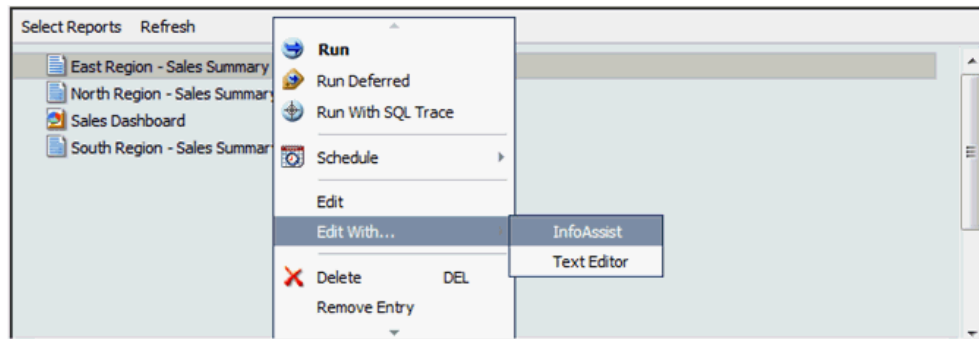


3. Select one or more nodes from the Content node on the left pane, and click save.

Subsequent visits to the WebFOCUS Report component will display the selected content items by default, as shown in the following image.



4. Double-click a content item to run it or right-click a content item to select different options (such as Run, Edit, and Schedule), as shown in the following image.



WebFOCUS Report Component Parameters

The following table lists and describes the available WebFOCUS Report component parameters.

Parameters	Description
WebFOCUS Connection	
Source URL	URL and port number to the WebFOCUS client used to retrieve WebFOCUS content items.

Parameters	Description
User run only	<p>Used to control whether or not a user is allowed to change the content displayed in the Report component.</p> <p>If the <i>User run only</i> check box is not selected, then the user will be able to select a WebFOCUS content item and its display mode. This is the default setting.</p> <p>If the <i>User run only</i> check box is selected, then the Report Component will display a default content item, such as a report, to any user that does not have Managed Reporting administrator privileges, and the <i>Select Report</i> menu option will not be available, preventing the user from changing the content item selected by default.</p> <p>Note: The <i>User run only</i> parameter works in conjunction with the following Open Portal Services security privileges:</p> <ul style="list-style-type: none"> <input type="checkbox"/> <i>Edit OPS Portlet</i> <input type="checkbox"/> <i>Save OPS Portlet Customization</i> <p>For more information, see WebFOCUS Report Component Configurations on page 369.</p>
Show Refresh	<p>If the <i>Show Refresh</i> check box is not selected, then the <i>Refresh</i> option will not be available on the Report Component menu and the user will not be able to manually trigger the content item to refresh.</p> <p>If <i>Show Refresh</i> is selected, then the <i>Refresh</i> option will be displayed on the Report Component menu and the user can manually trigger the content item to refresh. This is the default setting.</p>

Parameters	Description
Use scroll bars	<p>If the <i>Use scroll bars</i> check box is not selected, then the Report Component will not display scrollbars for the content item displayed.</p> <p>If the <i>Use scroll bars</i> check box is selected, then the Report Component will display scrollbars for the content item displayed. This is the default setting.</p> <p>This is particularly useful for content items that are larger than the component width and height, such as reports with several rows.</p>
Show time stamp	<p>If the <i>Show time stamp</i> check box is not selected, then the Report Component will not display the time stamp. This is the default setting.</p> <p>If the <i>Show time stamp</i> check box is selected, then the Report Component will display the time stamp.</p>
Gn	<p>The <i>gn</i> parameter is used to distinguish between multiple instances of the WebFOCUS Report component and retain its attributes (for example, width, height, and refresh rate).</p> <p>If you add three Report components to your portal page, then you must specify a unique <i>gn</i> value for each instance of that Report component.</p>
Number of columns	<p>Represents the number columns displayed inside the content mode box. For example, if a user is selecting reports to be placed in a list and the <i>Number of columns</i> parameter is set to 2, then items will be organized into two columns.</p> <p>The default value is 1.</p>
Refresh	<p>The amount of time in seconds before the content in the WebFOCUS component is refreshed.</p> <p>If nothing is specified, the default value is 0, meaning the component will not automatically refresh itself.</p>

Parameters	Description
Scale	The time scale used to determine the refresh rate. Specify <i>m</i> for minutes (default), <i>s</i> for seconds, or <i>h</i> for hours.
Content	
Context path	The application context path. For example: <code>/ibi_apps</code>
Prefix fba usernames	
Trace	Displays debugging information.
Appearance	
Title	The title for the Report Component to appear on the top left of the component box.
Height	Represents the height of the component. Select either <i>Yes</i> and provide the fixed height of the Web Part, or <i>No. Adjust height to fit zone.</i>
Width	Represents the width of the component. Select either <i>Yes</i> and provide the fixed width of the Web Part, or <i>No. Adjust width to fit zone.</i>
Chrome State	It can be set to <i>Minimized</i> , meaning the component will be minimized on the page, or <i>Normal</i> , meaning the component will be displayed with the set dimensions.

WebFOCUS Report Component Configurations

There is one WebFOCUS Report component parameter and two Open Portal Service security privileges that control the ability of the user to set and access content in the WebFOCUS Report component, as indicated in the following table.

UseRunOnly (Parameter)	OPS Customize (Privilege)	OPS Edit (Privilege)	Access Type
No	No	No	Select Report is not displayed. Access Denied.
No	No	Yes	Select Report is not displayed. Access Denied.
No	Yes	No	Select Report is displayed for customization. The user can customize, select, and add tree items. OPS Customize has precedence over OPS Edit.
No	Yes	Yes	Select Report is displayed for customization. The user can customize, select, and add tree items. OPS customize has precedence over OPS edit.
Yes	No	No	Select Report is not displayed. The user sees Fixed Report configured by a user with OPS Edit.
Yes	No	Yes	Select Report is displayed for global configuration. The user sees Fixed Report configured globally by a user with OPS Edit.
Yes	Yes	No	Select Report is NOT Displayed. The user sees Fixed Report configured by a user with OPS Edit.

UseRunOnly (Parameter)	OPS Customize (Privilege)	OPS Edit (Privilege)	Access Type
Yes	Yes	Yes	Select Report is displayed for global configuration. The user sees Fixed Report configured globally by a user with OPS Edit.

For more information on how to configure the WebFOCUS component privileges, see the *WebFOCUS Security and Administration* content.

Using the WebFOCUS Deferred Status Component

The WebFOCUS Deferred Status component allows users to check the status of any report submitted for deferred execution. An example of the WebFOCUS Deferred Status component is shown in the following image.

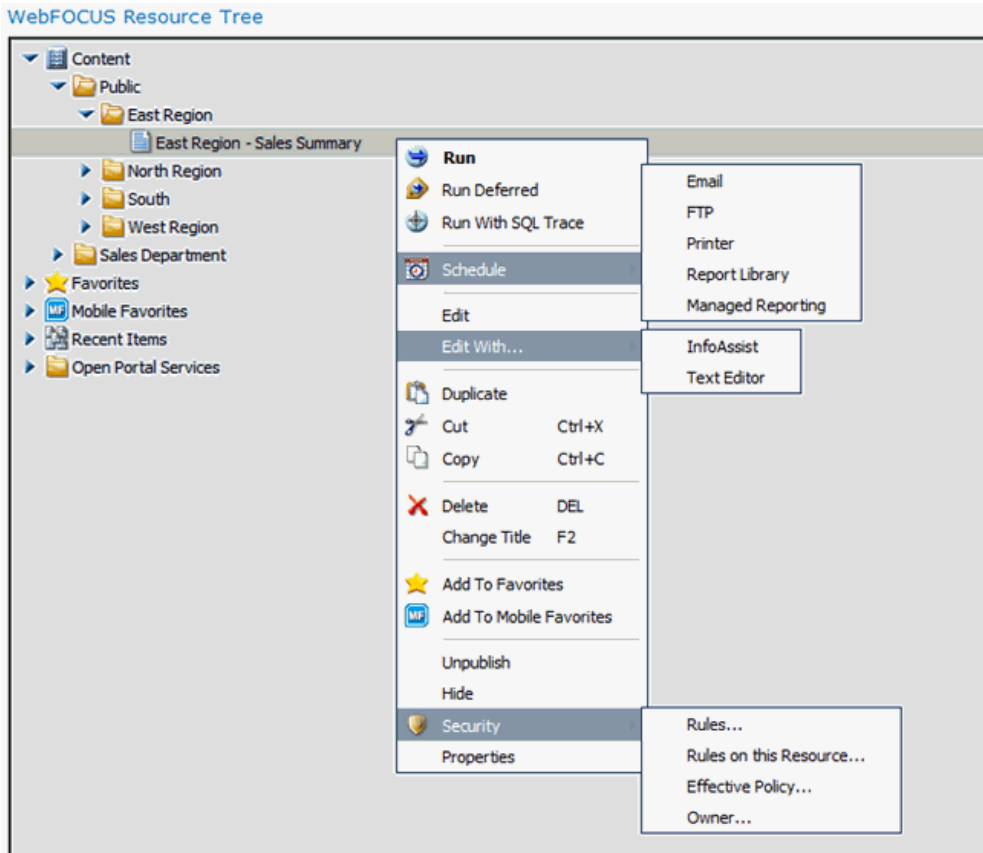


Using the WebFOCUS Resource Tree Component

Once the WebFOCUS Resource Tree Component is added to the portal page, users can navigate through the tree nodes and perform different operations as described in the following sections.

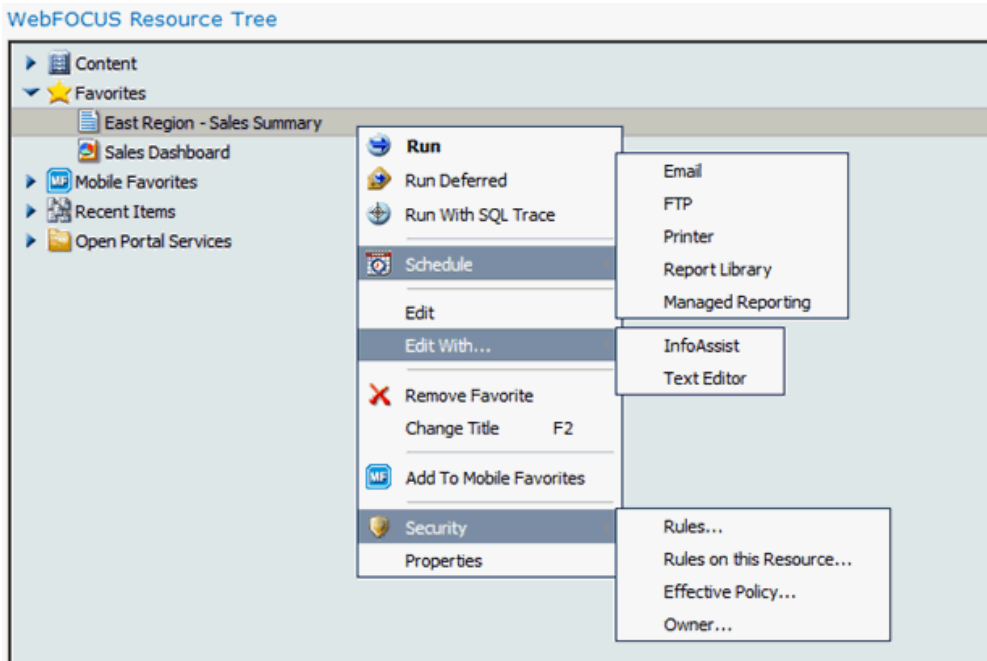
Content Node

The Content node allows users to interact with WebFOCUS content items based on the security privileges of the user by either double-clicking on the item to run it or by right-clicking on the item and selecting an option from the menu that appears, as shown in the following image.



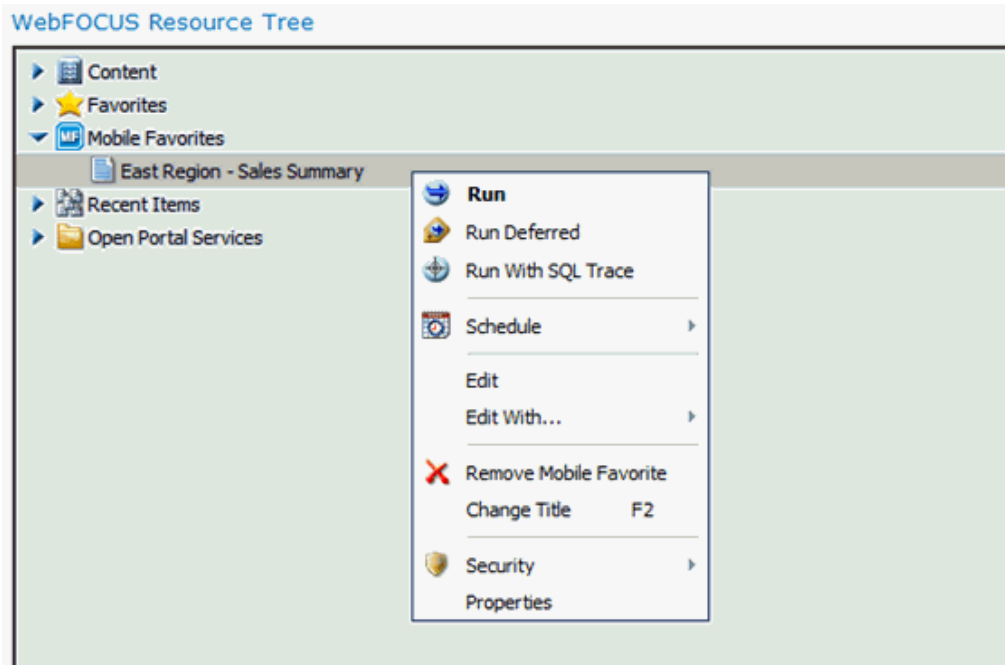
Favorites Node

The Favorites node allows users to interact with WebFOCUS content items based on the security privileges of the user by either double-clicking on an item to run it or by right-clicking an item and selecting an option from the menu that appears, as shown in the following image.



Mobile Favorites Node

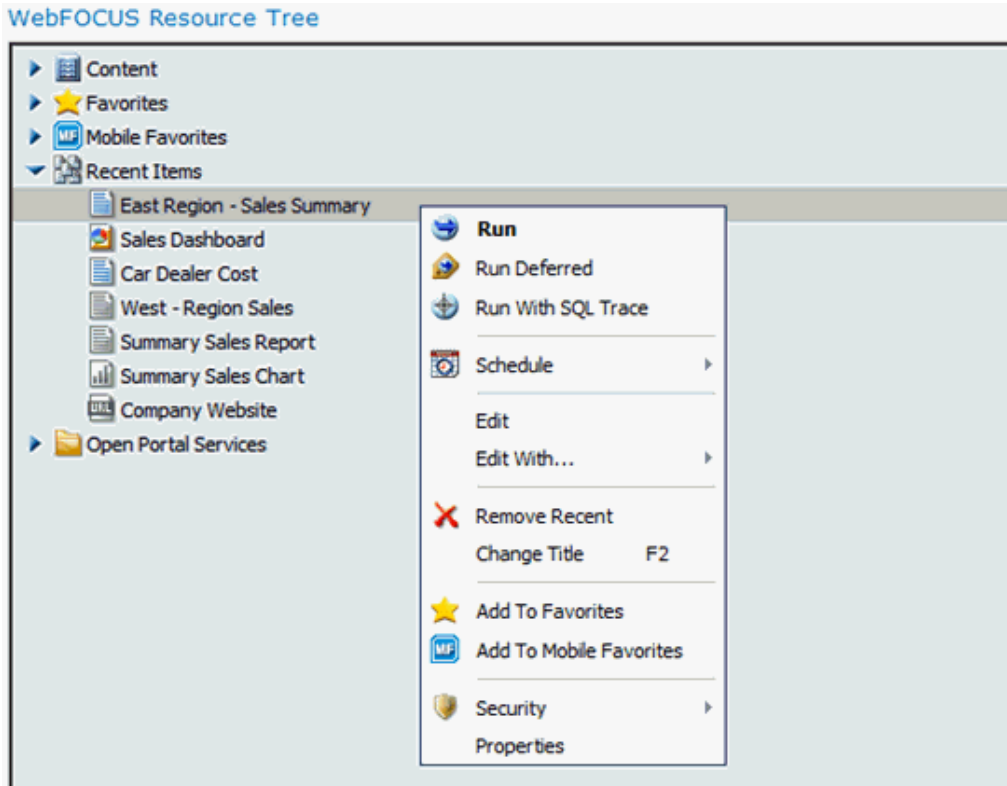
The Mobile Favorites node includes items that will be displayed in the Mobile Favorites application. It also allows users to interact with WebFOCUS content items based on the security privileges of the user by either double-clicking on the item to run it or by right-clicking on the item and selecting an option from the menu that appears, as shown in the following image.



For more information about Mobile Favorites, see the *WebFOCUS Business Intelligence Portal* content.

Recent Items Node

The Recent Items node displays recently run content items. It also allows users to interact with WebFOCUS content items based on the security privileges of the user by either double-clicking on the item to run it or by right-clicking on the item and selecting an option from the menu that appears, as shown in the following image.



WebFOCUS Resource Tree Component Parameters

The following table lists and describes the available WebFOCUS Resource Tree component parameters.

Parameter	Description
WebFOCUS Connection	

Parameter	Description
Source URL	The URL and port number to the WebFOCUS client used to retrieve WebFOCUS content items.
Proxy URL	The Proxy URL and port number to the WebFOCUS client.
User run only	<p>Used to control whether or not a user is allowed to change the content displayed in the Report Component.</p> <p>If the <i>User run only</i> check box is not selected, then the user will be able to select a WebFOCUS content item and its display mode. This is the default setting.</p> <p>If the <i>User run only</i> check box is selected, then the Report Component will display a default content item, such as a report, to any user that does not have Managed Reporting administrator privileges, and the <i>Select Report</i> menu option will not be available, preventing the user from changing the content item selected by default.</p> <p>Note: The <i>User run only</i> parameter works in conjunction with the following Open Portal Services security privileges:</p> <ul style="list-style-type: none"> <input type="checkbox"/> <i>Edit OPS Portlet</i> <input type="checkbox"/> <i>Save OPS Portlet Customization</i> <p>For more information, see WebFOCUS Report Component Configurations on page 369.</p>
Show Refresh	<p>If the <i>Show Refresh</i> check box is not selected, then the <i>Refresh</i> option will not be available on the Report Component menu and the user will not be able to manually trigger the content item to refresh.</p> <p>If <i>Show Refresh</i> is selected, then the <i>Refresh</i> option will be displayed on the Report Component menu and the user can manually trigger the content item to refresh. This is the default setting.</p>

Parameter	Description
Use scroll bars	<p>If the <i>Use scroll bars</i> check box is not selected, then the Report Component will not display scrollbars for the content item displayed.</p> <p>If the <i>Use scroll bars</i> check box is selected, then the Report Component will display scrollbars for the content item displayed. This is the default setting.</p> <p>This is particularly useful for content items that are larger than the component width and height, such as reports with several rows.</p>
Show time stamp	<p>If the <i>Show time stamp</i> check box is not selected, then the Report Component will not display the time stamp. This is the default setting.</p> <p>If the <i>Show time stamp</i> check box is selected, then the Report Component will display the time stamp.</p>
Gn	<p>The <i>gn</i> parameter is used to distinguish between multiple instances of the WebFOCUS Report component and retain its attributes (for example, width, height, and refresh rate).</p> <p>If you add three Report components to your portal page, specify a unique <i>gn</i> value for each instance of that Report component.</p>
Number of columns	<p>Represents the number of columns displayed inside the content mode box. For example, if a user is selecting reports to be placed in a list and the <i>Number of columns</i> parameter is set to 2, then items will be organized into two columns.</p> <p>The default value is <i>1</i>.</p>
Refresh	<p>The amount of time in seconds before the content in the WebFOCUS component is refreshed.</p> <p>If nothing is specified, the default value is <i>0</i>, meaning the component will not automatically refresh itself.</p>
Scale	<p>The time scale used to determine the refresh rate. Specify <i>m</i> for minutes (default), <i>s</i> for seconds, or <i>h</i> for hours.</p>

Parameter	Description
Content	
Context path	The application context path. For example: <code>/ibi_apps</code>
Prefix fba usernames	
Trace	Displays debugging information.
Appearance	
Title	The title for the Report Component that appears on the top left of the component pane.
Height	Represents the height of the component. Select either <i>Yes</i> and provide the fixed height of the Web Part, or <i>No. Adjust height to fit zone.</i>
Width	Represents the width of the component. Select either <i>Yes</i> and provide the fixed width of the Web Part, or <i>No. Adjust width to fit zone.</i>
Chrome State	This can be set to <i>Minimized</i> , meaning that the component will be minimized on the page, or <i>Normal</i> , meaning the component will be displayed with the set dimensions.

Usage Considerations

This section describes several considerations when using and configuring WebFOCUS Open Portal Services.

Right-Click Context Menu Persists When Working in Another Portlet

When working between two or more portlets, a right-click context menu that appears in one portlet can also appear in another portlet. For example, if you right-click on a folder in the Resource Tree portlet to perform an action (for example, to run a procedure), you can also do the same in the Report portlet. Context menus in both portlets are displayed/enabled. The context menus do not disappear unless you click anywhere within the respective portlet where a context menu is displayed.

Portlet Menu Options to Avoid

When embedding a WebFOCUS portal, developers should avoid using the *Sign In*, *Sign Out*, and *Close* menu options in their WebFOCUS portlets. Including these menu options may leave the portlet in an unstable state.

This section describes how administrators can install the WebFOCUS App Parts for the Microsoft SharePoint Portal Server 2016.

In this chapter:

- ❑ [On-Premise SharePoint Server](#)
- ❑ [Using Apps \(Add-ins\) With a Developer Site and Other Site Types Through the App Catalog.](#)
- ❑ [Using SharePoint on Microsoft Office 365 and Azure \(In the Cloud\)](#)

On-Premise SharePoint Server

This section describes how to configure the On-Premise SharePoint Server.

As of Microsoft SharePoint Portal Server 2016, the WebFOCUS App Parts are packaged as a SharePoint Add-in (*WfApp.app*), which is located in the following directory of your WebFOCUS installation:

```
drive:\ibi\WebFOCUS82\utilities\ops\sharepoint-addin
```

where:

```
drive:
```

Is the drive letter corresponding to the location where WebFOCUS is installed.

In terms of security, ensure that the same security paradigm is used on both environments (WebFOCUS and SharePoint).

1. Add the domain user ID you intend to use for apps (Add-ins) as a Windows administrator.
2. Use Central Administration to ensure that the *Application Management Service* and *User Profile Service Application* are configured, by selecting *Manage Service Applications* under Application Management.
3. Verify that your domain user ID has a user profile in Central Administration. If it does not, then create a new user profile for it by clicking the *User Profile Service* link under Manage Service Applications.
4. Under People, select *Manage User Profiles*, and search for the specific name.
5. Add the domain user ID to SQL Server as a login with the *sysadmin* role and *User Mapping* of *db_owner*, and *Sharepoint_shell_access* for the *SharePoint_config* database.
6. Open SharePoint Management Shell using the *Run as Administrator* option.

7. Load the PowerShell snap-ins for the session/script using the following command:

```
Add-PSSnapin Microsoft.Sharepoint.Powershell
```

8. Add user(s) as spshell administrators using the following command:

```
Add-SPShellAdmin -UserName domain\username
```

9. Set the Add-ins subdomain using the following command:

```
Set-SPAppDomain "wfapp.ibi.com"
```

10. Set the account to run Add-ins using the following command:

```
$account = New-SPManagedAccount
```

11. Set the account, app pool, and database settings using the following commands:

```
a. $account = Get-SPManagedAccount "domain\user"
```

```
b. $appPoolSubSvc = New-SPServiceApplicationPool -Name  
SettingsServiceAppPool -Account $account
```

```
c. $appPoolAppSvc = New-SPServiceApplicationPool -Name  
AppServiceAppPool -Account $account
```

```
d. $appSubSvc = New-SPSubscriptionSettingsServiceApplication -  
ApplicationPool $appPoolSubSvc -Name SettingsServiceApp -DatabaseName  
SettingsServiceDB  
(Note: The user ID must be sysadmin or dbcreator for this command.)
```

```
e. $proxySubSvc = New-SPSubscriptionSettingsServiceApplicationProxy -  
ServiceApplication $appSubSvc
```

```
f. $appAppSvc = New-SPAppManagementServiceApplication -  
ApplicationPool $appPoolAppSvc -Name AppServiceApp -DatabaseName  
AppServiceDB
```

```
g. $proxyAppSvc = New-SPAppManagementServiceApplicationProxy -  
ServiceApplication $appAppSvc
```

12. Set the Add-in prefix using the following command:

```
Set-SPAppSiteSubscriptionName -Name "add-in" -Confirm:$false
```

The full app (Add-in) name will appear as shown in the following example:

```
http://add-in-61844c031b3e0d.wfapp.ibi.com/sites/WFapp/
```

Note: Providing the support for wfapp.ibi.com (or other) domain is the responsibility of the SharePoint administrator.

The SharePoint Server is now configured to support apps (Add-ins).

Using Apps (Add-ins) With a Developer Site and Other Site Types Through the App Catalog.

This section describes how to use apps (Add-ins) with a developer site and other site types through the app catalog.

Procedure: How to Use Apps (Add-ins) With a Developer Site

To use apps (Add-ins) with a developer site:

1. Using Central Administration, create a Developer site with *Create Site Collections*.
2. From another machine, use your browser to go to the following new developer website:
<http://sharepointServer/sites/developerSite>
3. Pre-register the WebFOCUS App using the following:
http://sharepointServer/sites/developerSite/_layouts/15/appregnew.aspx
4. For Client ID, enter the following:
`40f398ee-1f96-4e4e-96dd-9f7218a3b880`
5. Next to Client Secret, click *Generate*.
6. Provide a title in the Title field.
7. In the App Domain field, enter a non-existent host, for example:
<http://www.webfocusdavid.com>
8. In the Redirect URL field, enter the following:
<http://www.webfocusdavid.com/default.aspx>
9. Click *Create*.
10. In the left pane, select *Apps in Testing*.
11. Click the *New app to deploy* link.
12. From the dialog that appears, click *Upload* and navigate to *WFApp.app* to upload it.
13. Click *Deploy*, and then when you are prompted, click *Trust it*.
14. Refresh your browser until the *Installing* message disappears, which indicates that the app is ready for use.
15. Create a page in the site.
16. From the Insert tab at the top of the page, select *App Parts* so that the WebFOCUS App Parts will be available for use.

When you insert the first app part, the frame will get an error which shows you the host that needs to be added to the Windows Hosts file in order to access the Add-ins.

17. Use the real IP address of the SharePoint Server machines in the hosts file.
18. Refresh the browser.
The frame may show a *403 - Access Denied* message which means you need to log on to WebFOCUS in another browser tab.

Procedure: How to Use Apps (Add-ins) With Other Site Types Through the App Catalog

To use apps (Add-ins) with other site types through the app catalog:

1. Using the Central Administration with a Farm Administrators account, click *Apps* from the left side of the pane, then under App Management, click *Manage App Catalog*.
2. Select the *Create a new app catalog site* radio button, and click *Ok*.
3. On the Create App Catalog page, enter a title for the App Catalog site and an optional description.
4. In the URL field, enter the URL to use for this site, for example:

<http://sharepointServer/sites/mycatalog>

5. In the Primary Site Collection Administrator section, type the name of the user who will manage the catalog, and then click *check names* to validate it.
6. In the End Users section, type of the names of the users or groups that you wish to be able to browse the catalog, and then click *check names* to validate them.
7. Select an optional quota that is needed, and then click *Ok* to create the catalog.

Once the catalog is created, the Manage App Catalog page displays the App Catalog site URL.

8. Using your browser, go to the App Catalog website and click *Apps for SharePoint* on the left side of the page.
9. Click *New* and in the *Add a document* dialog that appears, click *Choose files* and navigate to the folder that has the *app* file and select it for upload.
10. Click *Ok*.
11. Use the Central Administration to create a non-Developer site with *Create Site Collections*.
12. Using your browser in another machine, go to the following new website:

<http://sharepointServer/sites/testSite>

13. Pre-register the WebFOCUS App using the following:

http://sharepointServer/sites/testSite/_layouts/15/appregnew.aspx

14. In the Client ID field, enter the following:

40f398ee-1f96-4e4e-96dd-9f7218a3b880

15. Next to Client Secret, click *Generate*.
16. Enter a title in the Title field.
17. In the App Domain field, enter a non-existent host, for example:
<http://www.webfocusdavid.com>
18. In the Redirect URL field, enter the following:
<http://www.webfocusdavid.com/default.aspx>
19. Click *Create*, and then click *Ok*.
20. From the gear menu at the top right of the pane, click *Add an App* to display the Your Apps page.
21. Click *Information Builders WebFOCUS applets*.
22. In the *Do you trust WebFOCUS?* dialog that appears, click the *Trust it* button.
 This will add the app to your site.
23. Create a page in the site.
24. From the Insert tab at the top of the page, select *App Parts* and you will see the WebFOCUS App Parts available for use.
 When you insert the first app part, the frame will get an error which shows you the host that needs to be added to the Windows Hosts file in order to access the Add-ins. You can use the real IP address of the SharePoint Server machines in the hosts file.
25. Refresh the browser.
 The frame may show a *403 - Access Denied* message, indicating that you must log on to WebFOCUS in another browser tab.

Using SharePoint on Microsoft Office 365 and Azure (In the Cloud)

Secure Sockets Layer (SSL) must be configured for WebFOCUS in order to use SharePoint in the cloud.

Note: The following procedure uses Microsoft Office 365 as an example, but the general steps would also apply in a Microsoft Azure environment.

1. Sign in to Office 365 with your SharePoint online account.
2. From the SharePoint Admin Center (the tile with *A* and *Admin* on it), select *Apps* from the left pane, and then click *App Catalog*.
3. Select *Create a new app catalog site* and then click *Ok*.
4. On the Create App Catalog Site Collection page, enter the required information, and then click *Ok*.

-
5. Navigate to the App Catalog site within the Admin Center, click *Apps*, and then click *App Catalog*.
 6. On the home page of the App Catalog site, select the tile labeled *Distribute apps for SharePoint*, and click *new app*.
 7. Navigate and select the folder that contains the app you wish to upload, and then click *Open*.
 8. In the *Add a document* dialog box, add any optional comments about this version of the app, and then click *Ok*.
 9. Ensure that the *Enabled* check box is selected so that users are able to add this app to their sites.
 10. Click *Save*.
 11. On the App Catalog site, pre-register the WebFOCUS app using:
http://sharepointServer/sites/testSite/_layouts/15/appregnew.aspx
 12. On the App Catalog site, click *Settings* and then click *Add an App*.
 13. Select the app you want to add, and when you are prompted, select *Trust it*.
 14. Create your new site collection.
 15. Using your browser, go to the URL of the new site collection and select *Add an App* from the *Settings* menu.
 16. Select the WebFOCUS app to add it to the new site.

Once it is installed, you can create pages and use *App Part* on the ribbon to add the WebFOCUS App Parts.
 17. In the *Edit web part* pane, ensure to specify an HTTPS URL to WebFOCUS and that your WebFOCUS environment is configured for SSL.

This section describes how administrators can install the WebFOCUS Web Parts for the Microsoft SharePoint Portal Server 2013.

In this chapter:

- ❑ [Microsoft SharePoint Portal Server 2013](#)
-

Microsoft SharePoint Portal Server 2013

The WebFOCUS Web Parts for Microsoft SharePoint Portal Server 2013 are included as a WebFOCUS Solution Package (webfocus.wsp).

The webfocus.wsp solution package for Microsoft SharePoint and script files are located in the following directory where WebFOCUS is installed:

```
drive:\ibi\WebFOCUS80\utilities\ops\sharepoint
```

where:

```
drive:
```

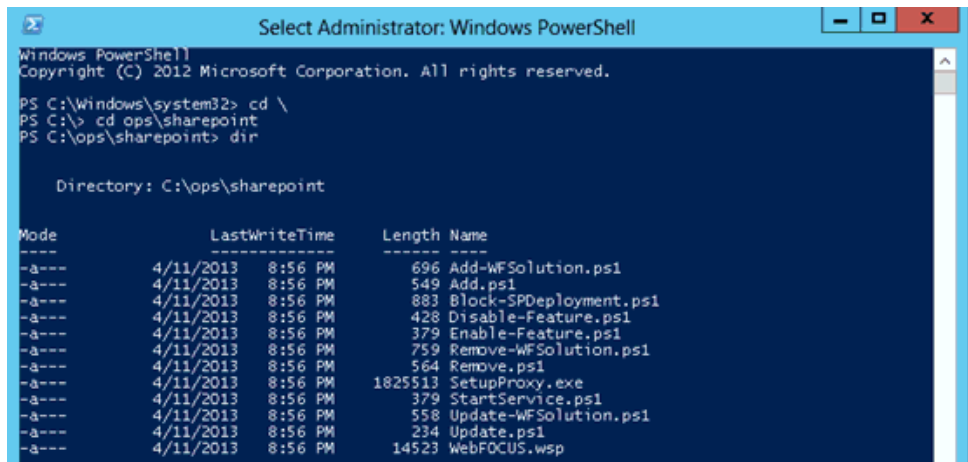
Is the drive letter corresponding to the location where WebFOCUS is installed.

Procedure: How to Install and Verify the WebFOCUS Solution Package for Microsoft SharePoint

1. Navigate to the following Microsoft SharePoint directory where WebFOCUS is installed:

```
drive:\ibi\WebFOCUS80\utilities\ops\sharepoint
```
2. Copy the *sharepoint* directory to the system where Microsoft SharePoint Portal Server 2013 is being hosted.
3. Launch the SharePoint Management Shell as an Administrator.
4. Navigate to the location of the WebFOCUS files in the *sharepoint* directory that was copied. For example:

c:\ops\sharepoint



```
Windows PowerShell
Copyright (C) 2012 Microsoft Corporation. All rights reserved.

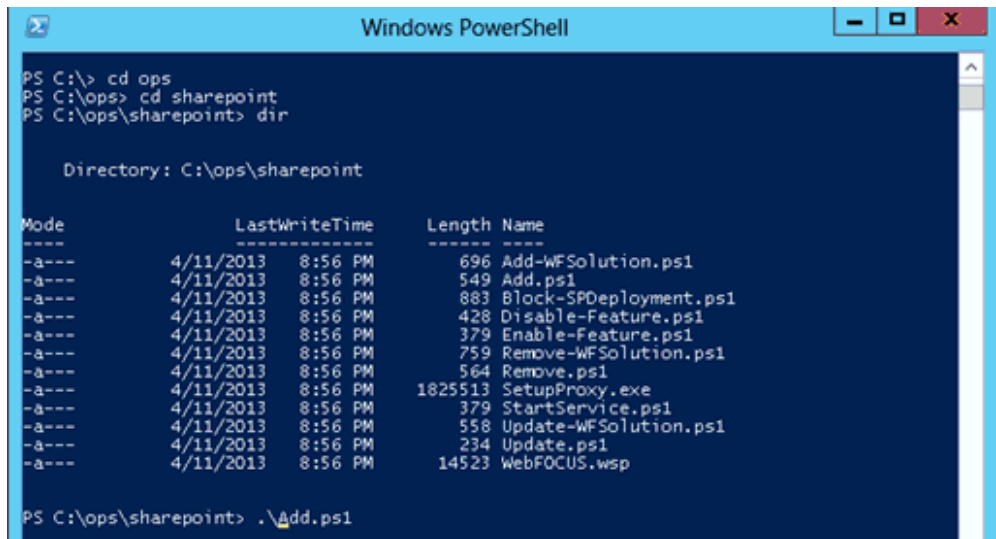
PS C:\Windows\system32> cd \
PS C:\> cd ops\sharepoint
PS C:\ops\sharepoint> dir

Directory: C:\ops\sharepoint

Mode                LastWriteTime         Length Name
----                -
-a---             4/11/2013  8:56 PM           696 Add-WFSolution.ps1
-a---             4/11/2013  8:56 PM           549 Add.ps1
-a---             4/11/2013  8:56 PM           883 Block-SPDeployment.ps1
-a---             4/11/2013  8:56 PM           428 Disable-Feature.ps1
-a---             4/11/2013  8:56 PM           379 Enable-Feature.ps1
-a---             4/11/2013  8:56 PM           759 Remove-WFSolution.ps1
-a---             4/11/2013  8:56 PM           564 Remove.ps1
-a---             4/11/2013  8:56 PM       1825513 SetupProxy.exe
-a---             4/11/2013  8:56 PM           379 StartService.ps1
-a---             4/11/2013  8:56 PM           558 Update-WFSolution.ps1
-a---             4/11/2013  8:56 PM           234 Update.ps1
-a---             4/11/2013  8:56 PM       14523 WebFOCUS.wsp
```

5. Enter the following command from the Power Shell command prompt:

```
PS C:\ops\sharepoint> .\Add.ps1
```



```
Windows PowerShell

PS C:\> cd ops
PS C:\ops> cd sharepoint
PS C:\ops\sharepoint> dir

Directory: C:\ops\sharepoint

Mode                LastWriteTime         Length Name
----                -
-a---             4/11/2013  8:56 PM           696 Add-WFSolution.ps1
-a---             4/11/2013  8:56 PM           549 Add.ps1
-a---             4/11/2013  8:56 PM           883 Block-SPDeployment.ps1
-a---             4/11/2013  8:56 PM           428 Disable-Feature.ps1
-a---             4/11/2013  8:56 PM           379 Enable-Feature.ps1
-a---             4/11/2013  8:56 PM           759 Remove-WFSolution.ps1
-a---             4/11/2013  8:56 PM           564 Remove.ps1
-a---             4/11/2013  8:56 PM       1825513 SetupProxy.exe
-a---             4/11/2013  8:56 PM           379 StartService.ps1
-a---             4/11/2013  8:56 PM           558 Update-WFSolution.ps1
-a---             4/11/2013  8:56 PM           234 Update.ps1
-a---             4/11/2013  8:56 PM       14523 WebFOCUS.wsp

PS C:\ops\sharepoint> .\Add.ps1
```

The following prompt is displayed:

Enter the Web Application full url? (http://localhost):

```

Administrator: Windows PowerShell
PS C:\ibi\webparts> dir

Directory: C:\ibi\webparts

Mode                LastWriteTime         Length Name
----                -
-ar--              3/11/2013  5:01 PM           641 Add-WFSolution.ps1
-ar--              3/11/2013  5:01 PM           549 Add.ps1
-ar--              3/11/2013  5:02 PM           883 Block-SPDeployment.ps1
-ar--              9/6/2012   1:35 PM          9991 Deploy-SPSolutions.ps1
-ar--              3/11/2013  5:02 PM           428 Disable-Feature.ps1
-ar--              3/11/2013  5:02 PM           379 Enable-Feature.ps1
-ar--              3/11/2013  5:02 PM           759 Remove-WFSolution.ps1
-ar--              3/11/2013  5:02 PM           564 Remove.ps1
-ar--              3/11/2013  5:02 PM           379 StartService.ps1
-ar--              3/11/2013  5:02 PM           558 Update-WFSolution.ps1
-ar--              3/11/2013  5:02 PM           234 Update.ps1
-a---              3/14/2013  2:19 PM          14769 WebFOCUS.wsp
-ar--             10/16/2012 11:53 AM            47 _sp.cmd

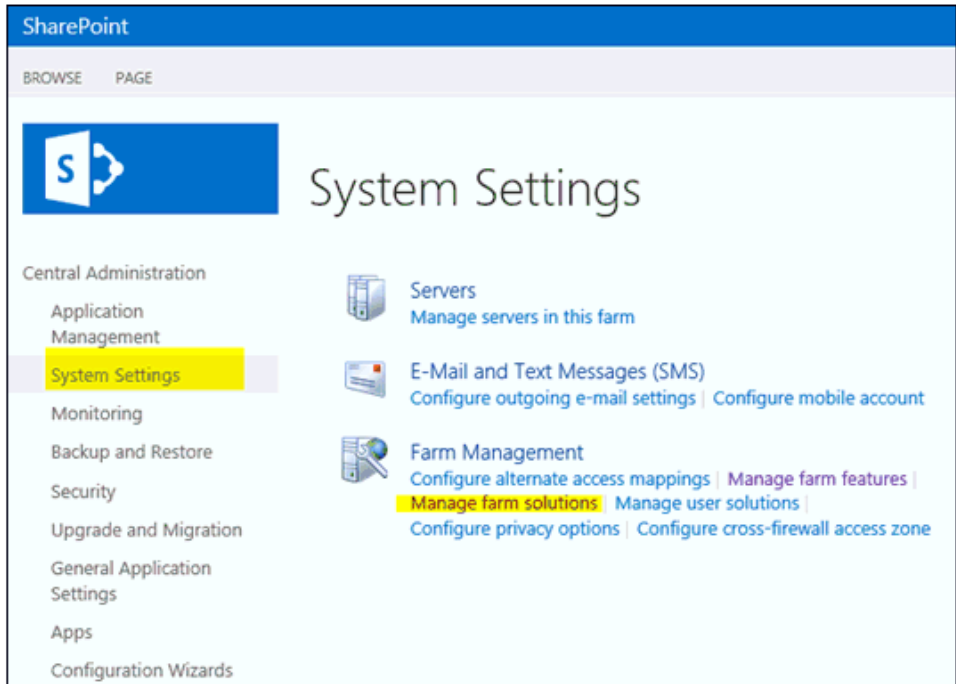
PS C:\ibi\webparts> .\Add.ps1
Enter the Web Application full url ? (http://localhost): _
  
```

- Specify the web application where the webfocus.wsp solution package for Microsoft SharePoint is to be deployed (or use the default localhost) and press Enter.

The webfocus.wsp solution package for Microsoft SharePoint is deployed and the Web Parts are enabled.

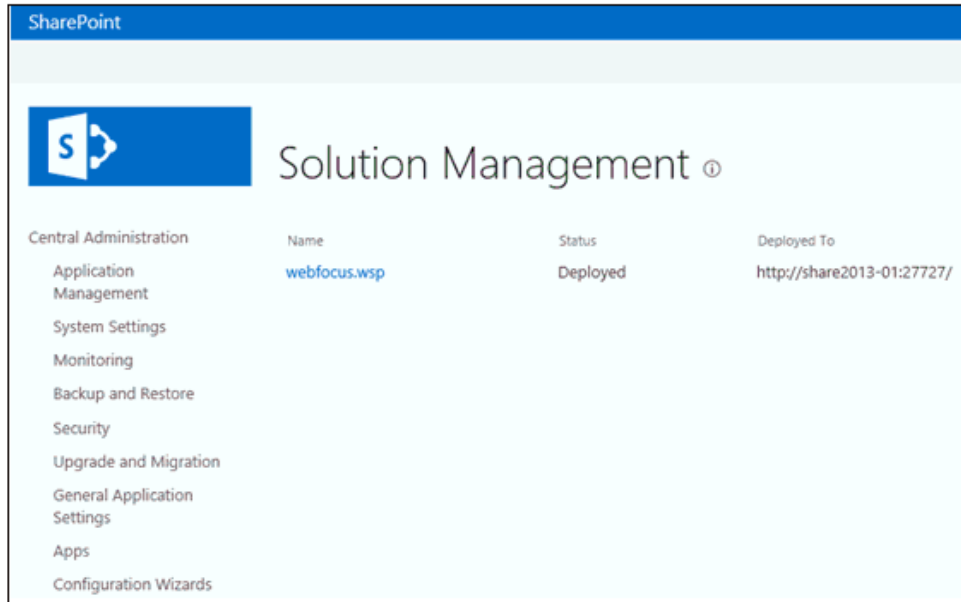
- To verify if the webfocus.wsp solution package for Microsoft SharePoint is successfully deployed, access the Central Administration page.
- Click *System Settings* in the left pane.

The System Settings page opens, as shown in the following image.



9. Click *Manage farm solutions*.

The Solution Management page opens, as shown in the following image.



Notice that the webfocus.wsp solution package for Microsoft SharePoint is listed with a status of Deployed.

Procedure: How to Uninstall the WebFOCUS Solution Package for Microsoft SharePoint

To uninstall the WebFOCUS Solution Package for Microsoft SharePoint (webfocus.wsp) and the Web Parts from the configuration, enter the following command from the Power Shell command prompt as an Administrator:

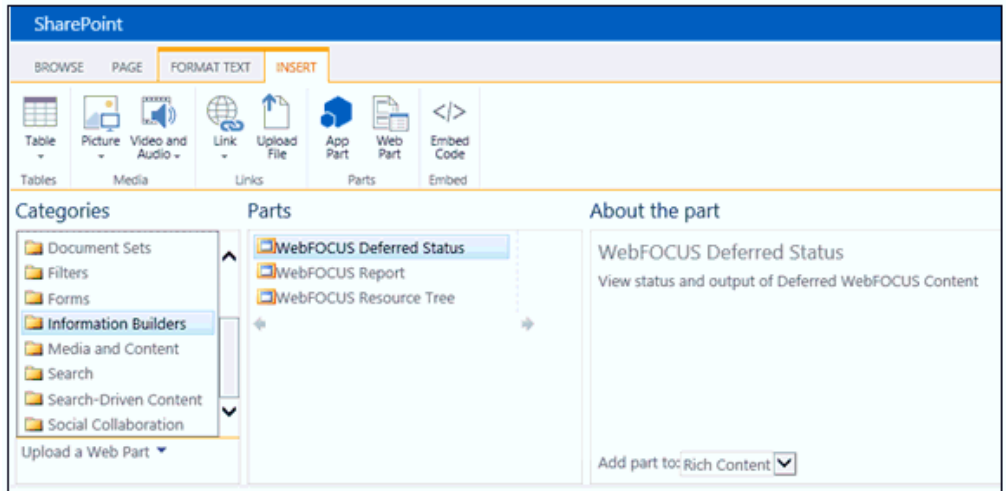
```
PS c:\ops\sharepoint> .\Remove.ps1
```

Procedure: How to Add and Configure WebFOCUS Web Parts

To add WebFOCUS Web Parts to the Microsoft SharePoint Portal Server:

1. Log on to the Microsoft SharePoint Portal Server 2013.
2. Navigate to the page where you want to add WebFOCUS Web Parts.
3. Edit the page.

4. From the Tools menu, click *Insert*.



5. Click *Web Part* in the menu bar.
6. From the Categories area in the left pane, click the WebFOCUS Web Parts category (for example, Information Builders).
7. From the Parts area, select the specific WebFOCUS Web Part and click *Add*.
8. To display properties for the Web Part, move your cursor over the right corner of the Web Part to display a drop-down menu.
9. Click *Edit Web Part*.
10. In the WebFOCUS Connection category, edit the Source URL by entering the machine name and port number that points to the WebFOCUS Client.
11. Expand the *Content* category.
12. Edit the context path based on your WebFOCUS Client installation.
13. Click *Apply* and then *OK*.

Repeat steps 8 through 13 for each WebFOCUS Web Part that you added to a page.

Embedding WebFOCUS Business Intelligence Content Into Salesforce.com

This section describes how to embed WebFOCUS Business Intelligence (BI) content into Salesforce.com. Salesforce.com is a cloud computing company, which develops a leading Customer Relationship Management (CRM) platform with cloud-based applications for sales, service, and marketing.

In this chapter:

- ❑ [Embedding a URL to Run a WebFOCUS Report](#)
- ❑ [Configuring SAML Authentication](#)
- ❑ [Programming Solutions](#)

Embedding a URL to Run a WebFOCUS Report

This section describes a simple example of embedding a URL into Salesforce.com (SFDC) that runs a WebFOCUS report.

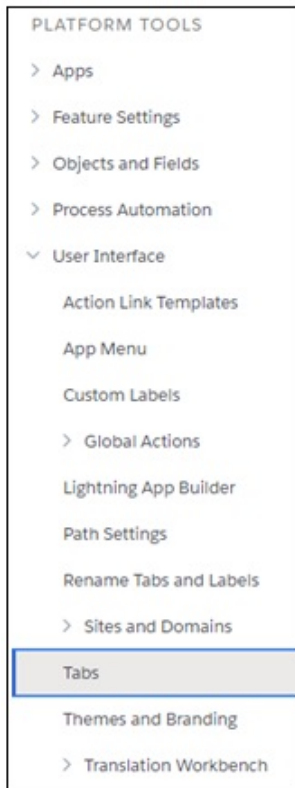
For example:

https://hostname:port/ibi_apps/run/ibfs/WFC/Repository/Tests/Car_Report.fex

1. Ensure that your WebFOCUS environment is configured for Secure Sockets Layer (SSL).
2. Login to Salesforce.com.
3. Click the gear icon to access the Setup menu, as shown in the following image.



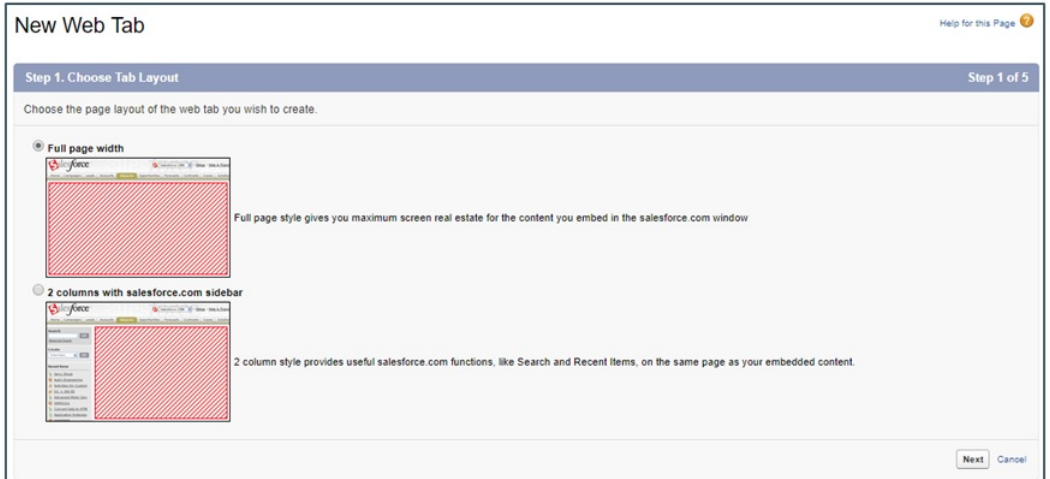
4. In the left pane under PLATFORM TOOLS, expand *User Interface* and then click *Tabs*, as shown in the following image.



5. Under Web Tabs, click *New*, as shown in the following image.

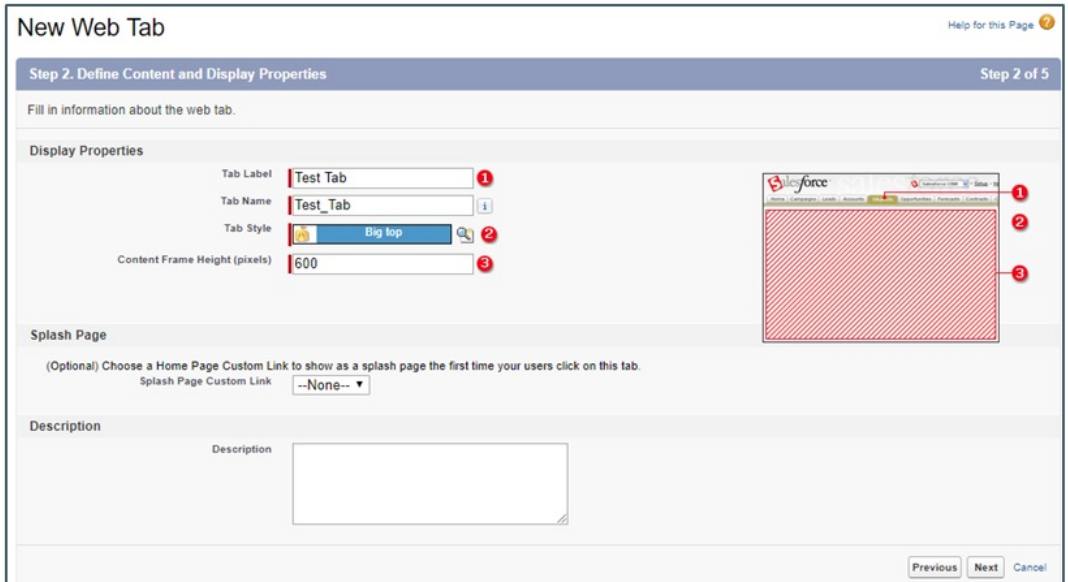


The New Web Tab dialog opens showing the *Step 1. Choose Tab Layout* pane, as shown in the following image.



6. Ensure the *Full page width* layout is selected and then click *Next*.

The *Step 2. Define Content and Display Properties* pane opens, as shown in the following image.



7. In the Display Properties area, enter a value in the Tab Label field and select a Tab Style from the drop-down list.

8. Change the default value (600) in the Content Frame Height (pixels) field if needed, and then click **Next**.

The *Step 3. Enter the URL Details* pane opens, as shown in the following image.

Step 3. Enter the URL Details Step 3 of 5

Enter the web page address in the Link URL field. You can enter a simple URL just as it appears in the browser address bar, or you can use one or more merge fields to insert organization-specific data from salesforce.com into URL parameters.

Some sites may not work in a Web tab because of browser security settings, or because the site has prevented itself from being displayed in a frame. For more information, visit the [Salesforce Help](#).

Examples:

Simple	https://yoursite.com
With Merge Field	https://yoursite.com/search?q={Org_Name}

Available Merge Fields

Select Field Type

Organization Fields ▾

Select Field

Copy Merge Field Value

Copy and paste the merge field value into your template below.

Button or Link URL Required information

https://as200.ibi.com:8443/ibi_apps/run/ibfs/WFC/Repository/Tests/Car_Report.fex

Preview Web Tab

Encoding Unicode (UTF-8)

Previous Next Cancel

9. Enter a URL that runs a WebFOCUS report in the bottom area of the screen.

For example:

https://hostname:port/ibi_apps/run/ibfs/WFC/Repository/Tests/Car_Report.fex

10. Click **Next**.

11. Do not make any changes for the *Add to Profiles* and *Add to Custom Apps* steps that follow.

12. Click **Save**.

The new tab you created (for example, Test Tab) is now listed under the Web Tabs area, as shown in the following image.

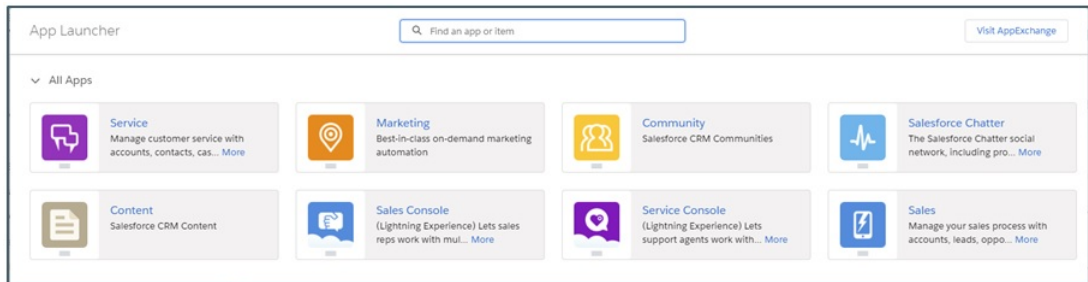
Web Tabs New What Is This?

Edit | Del Test Tab Big top

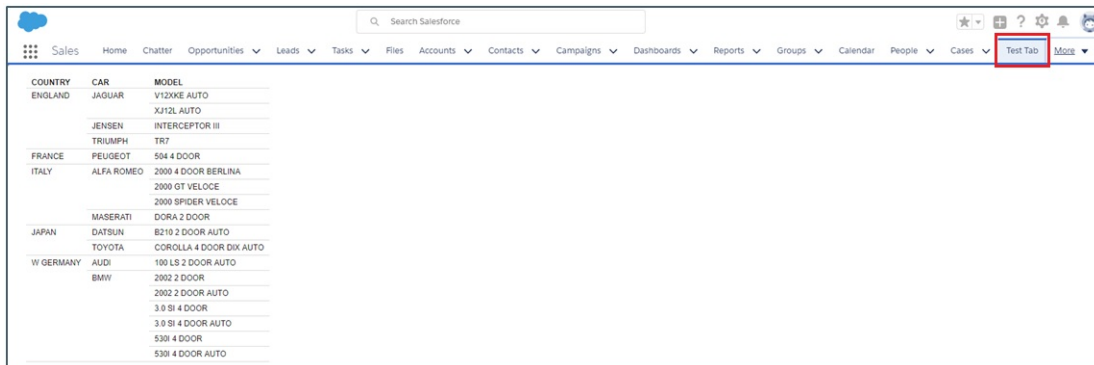
13. Click the Tiles menu, as shown in the following image.



14. Open a Salesforce.com application by clicking the corresponding tile, as shown in the following image.



15. Select the tab you created (for example, Test Tab) to run the WebFOCUS report you specified in the URL and display its output, as shown in the following image.



Configuring SAML Authentication

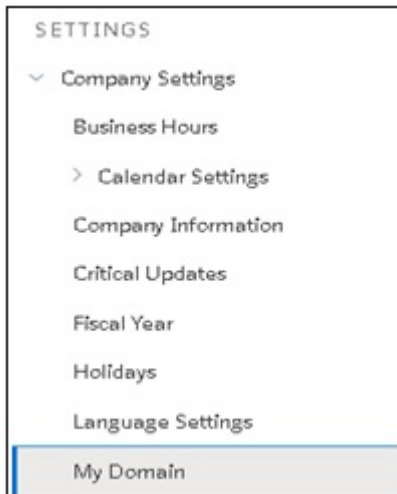
This section describes how to configure Security Assertion Markup Language (SAML) authentication as a single sign-on (SSO) login between Salesforce.com and WebFOCUS. Doing so prevents you from having to log on to Salesforce.com and WebFOCUS separately.

Enabling the Identity Provider

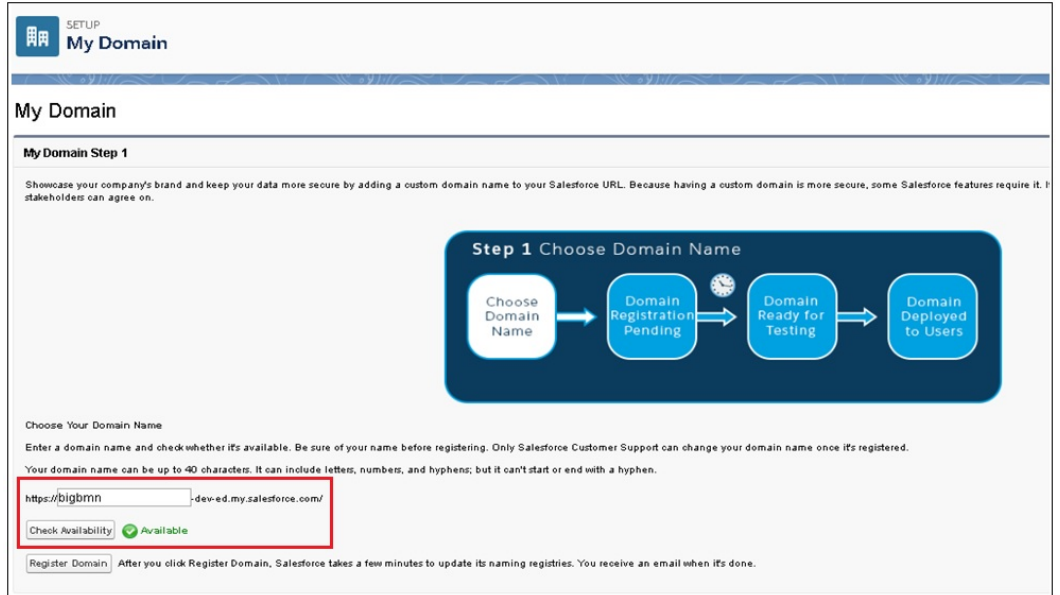
1. Log on to Salesforce.com.
2. Click the gear icon to access the Setup menu, as shown in the following image.



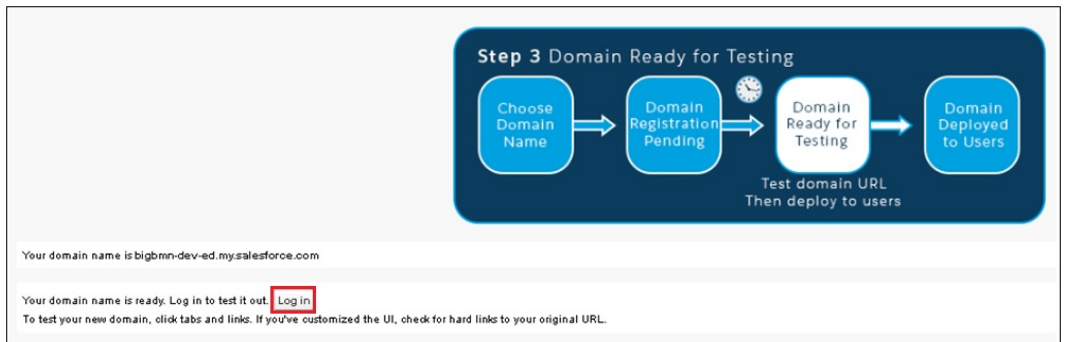
3. In the left pane under SETTINGS, expand *Company Settings* and then click *My Domain*, as shown in the following image.



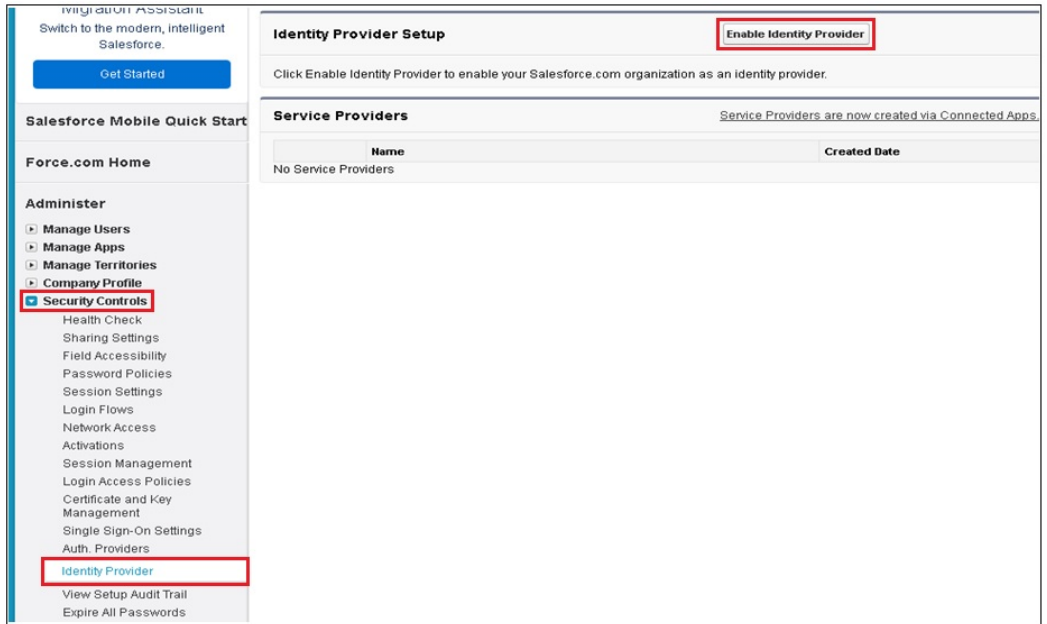
The My Domain pane opens, as shown in the following image.



4. Specify your domain name, and then click *Check Availability*.
5. Once your domain has been verified, click *Register Domain*.
6. Once your domain has been registered, log in to the domain by clicking *Log in*, as shown in the following image.

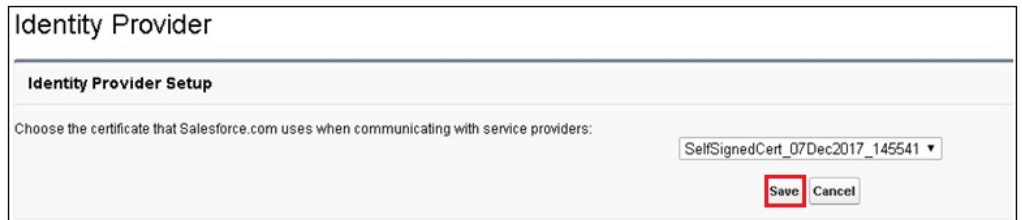


7. In the left pane, expand *Security Controls* and select *Identity Provider*, as shown in the following image.



8. Click *Enable Identity Provider*.

The following screen is displayed.



9. Click *Save*.

The following screen is displayed, which provides details regarding the Identity Provider, metadata, and the certificate.

Identity Provider

Enable Salesforce.com as an identity provider so you can use single sign-on with other web sites, and define the appropriate service providers whose applications support single sign-on. You can switch to different service providers without having to log in again. [Learn more...](#)

Identity Provider Setup Edit Disable Download Certificate Download Metadata

▼ **Details**

Issuer	https://bigbmn-dev-ed.my.salesforce.com		
--------	---	--	--

▼ **Currently chosen certificate details**

Label	SelfSignedCert_07Dec2017_145541	Unique Name	SelfSignedCert_07Dec2017_145541
Created Date	12/7/2017 6:55 AM	Expiration Date	12/7/2018 4:00 AM
Key Size	2048		

▼ **SAML Metadata Discovery Endpoints**

Salesforce Identity	https://bigbmn-dev-ed.my.salesforce.com/well-known/samlidp.xml
---------------------	--

Service Providers [Service Providers are now created via Connected Apps. Click here.](#)

Name	Created Date
No Service Providers	

10. Click *Download Metadata*.

Copy the metadata into the following WebFOCUS directory:

`\ibi\WebFOCUS82\config\was\saml`

This metadata will be used to configure WebFOCUS for the XML file that is used by the Identity Provider.

The specific names of the files are not important, but configuration values in the *securitysettings.xml* file must reference the correct metadata file.

SAMLIp-00D11000003pWq6.xml	12/7/2017 10:02 AM	XML Document	3 KB
SelfSignedCert_07Dec2017_145541.crt	12/7/2017 10:02 AM	Security Certificate	2 KB

Configuring WebFOCUS and Generating the *wfspMetadata.xml* File

At a high level, this process consists of the following steps:

- Creating a key pair to be used for WebFOCUS signing and encrypting (*keytool* command).
- Importing a Salesforce.com certificate (*keytool* command).
- Generating the *wfspMetadata.xml* file.

- ❑ Configuring WebFOCUS to utilize the *wfspMetadata.xml* and *SalesforceMetadata.xml* files.

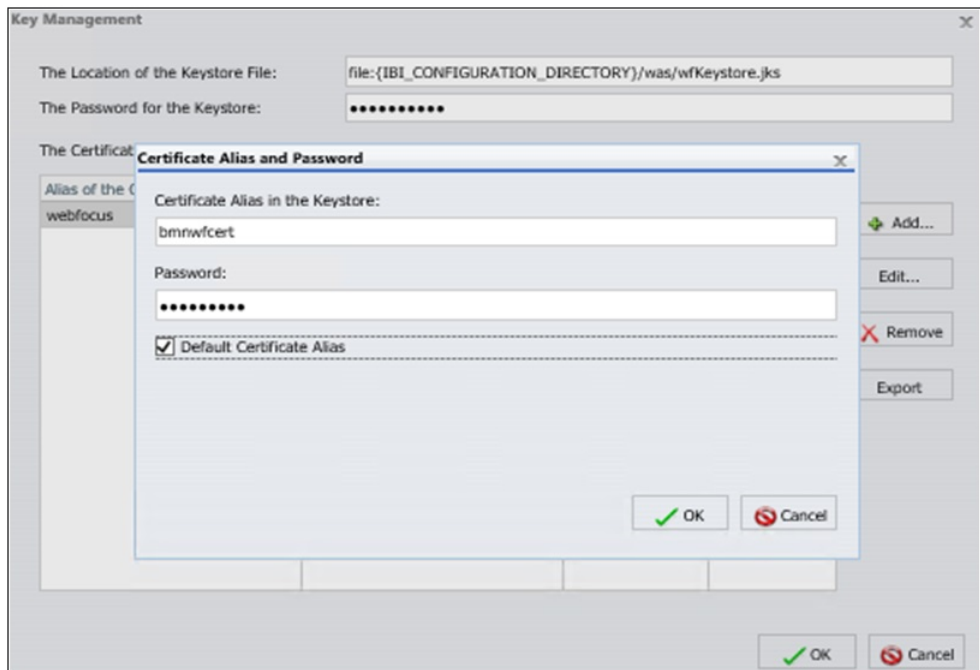
To configure WebFOCUS and generate the *wfspMetadata.xml* file:

1. Navigate to the WebFOCUS Administration Console and click the *Security* tab.
2. In the left pane, expand *Security Zones, Default*, and then click *Authentication*.
3. In the right pane, click *Key Management*.

The Key Management dialog opens.

4. Specify the password for the Keystore.
5. Click *Add*.

The Certificate Alias and Password dialog opens, as shown in the following image.

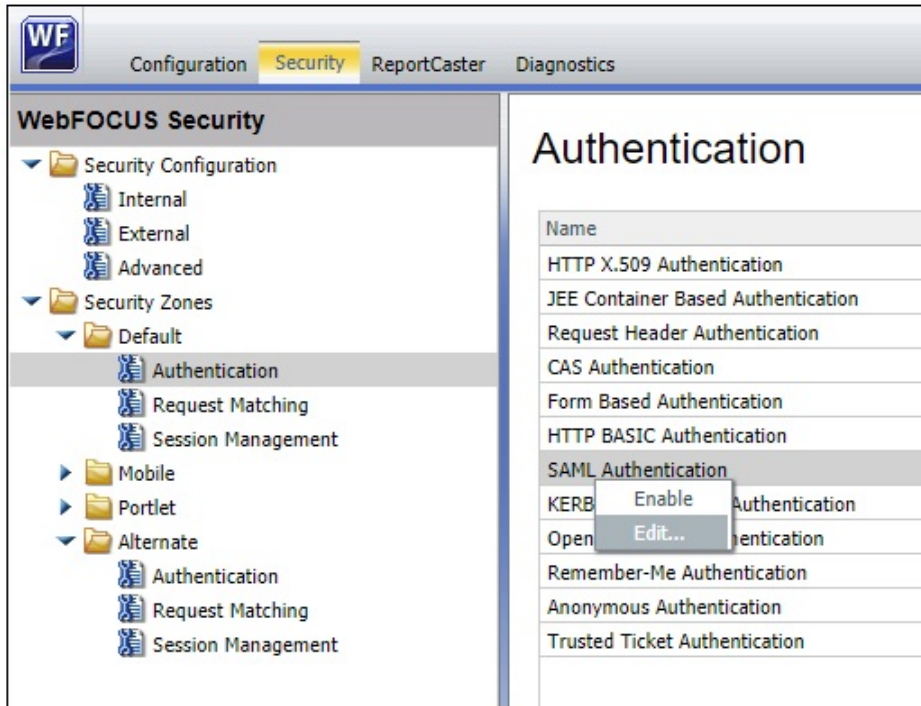


6. Specify the alias and password for the certificate in the corresponding fields, and then click the *Default Certificate Alias* check box.

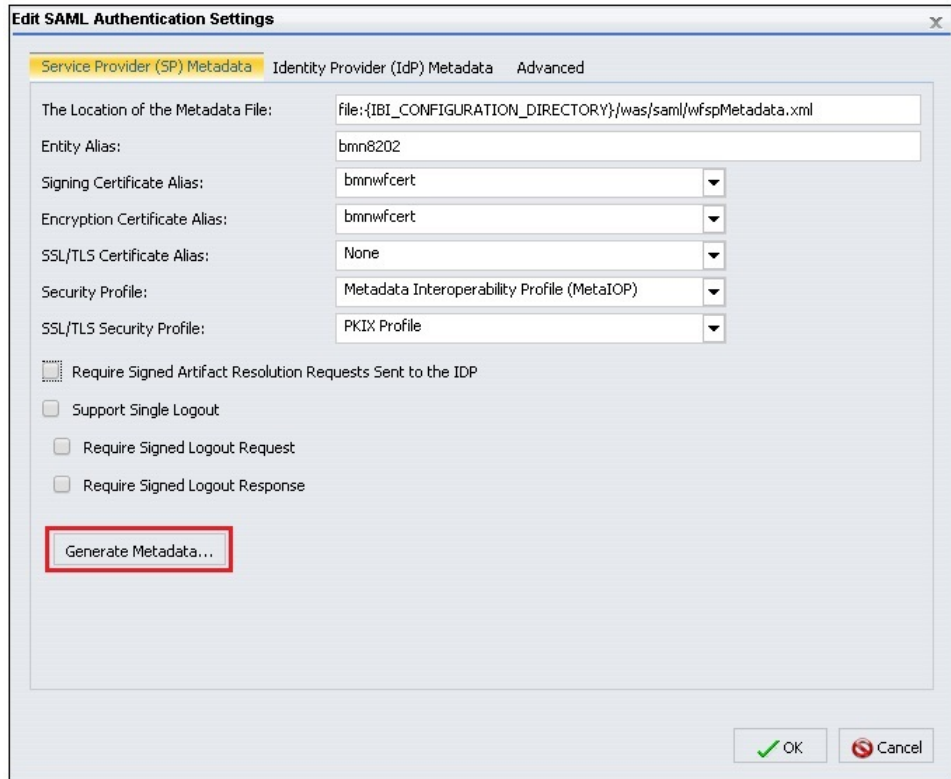
The password for the Keystore, alias of the certificate, and password for the certificate, are all in the keytool step for generating the key.

7. Click *OK* to save your changes and close the Certificate Alias and Password dialog.
8. Click *OK* to save your changes and close the Key Management dialog.

9. From the Authentication pane, right-click *SAML Authentication* and select *Edit* from the context menu, as shown in the following image.



The Edit SAML Authentication Settings dialog opens, as shown in the following image.



10. Specify values for the following parameters:

- Entity Alias
- Signing Certificate Alias
- Encryption Certificate Alias

11. Deselect the *Support Single Logout* check box.

If this option remains selected, logging out from WebFOCUS will automatically log you out from Salesforce.com.

12. Leave the default values for all of the remaining parameters.

Note: The same signing and encryption certificates are used in this example, but two different certificates could also be used, if configured originally using the keytool command.

13. Click *Generate Metadata*.

The Service Provider (SP) Metadata Generation dialog opens, as shown in the following image.

Service Provider (SP) Metadata Generation

Entity ID:

Entity Base URL:

Entity Alias:

Signing Certificate Alias:

Encryption Certificate Alias:

SSL/TLS Certificate Alias:

Security Profile:

SSL/TLS Security Profile:

The Service Signs Authentication Requests

Require Signed Authentication Assertion

Require Signed Artifact Resolution Requests Sent to the IDP

Require Signed Logout Request

Require Signed Logout Response

Single Sign-On Bindings:	Default	Included	Name
<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="checkbox"/>	SSO HTTP-POST
<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	SSO Artifact
<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	SSO PAOS
<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	HoK SSO HTTP-POST
<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	HoK SSO Artifact

Supported NameIDs:

Unspecified

E-Mail

Transient

Persistent

X509 Subject

14. Click *Generate*.

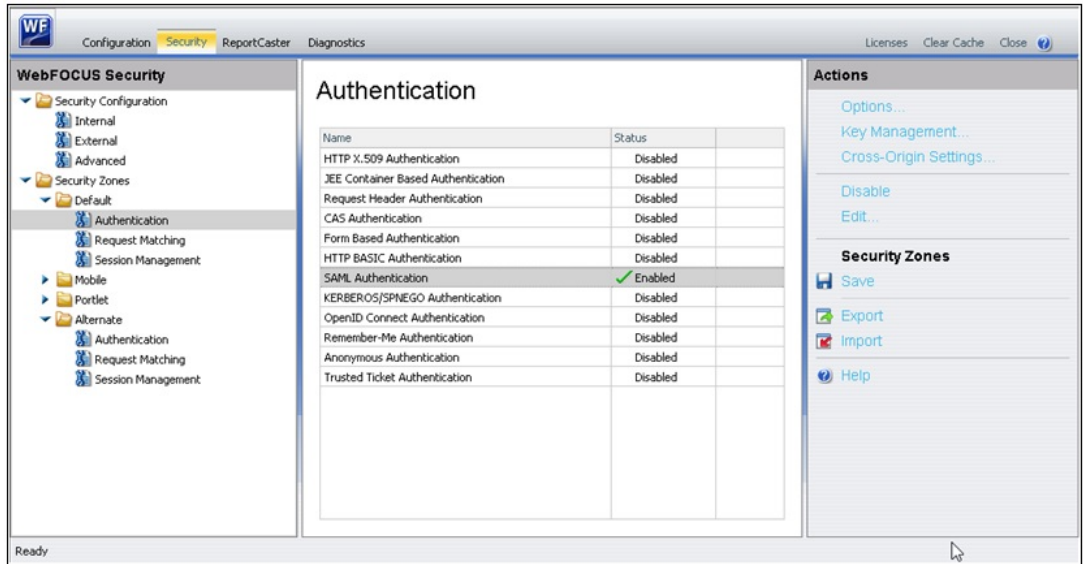
Note: If there were any issues with the passwords for the Keystore or certificate, a JSON file is returned and not the *wfspMetadata.xml* file.

15. Copy the *wfspMetadata.xml* file to the following WebFOCUS directory:

config\was\saml

Provide this file to your ADFS administrator for their use in the configuration of ADFS.

16. Enable the Alternate Authentication Zone to allow you to sign in to WebFOCUS from the local machine using forms-based authentication and not SAML. Doing so will allow you to fix any configuration issues.
17. In the Default Authentication Zone, disable Form Based Authentication, Anonymous Authentication, and enable *SAML Authentication*, as shown in the following image.



18. Click Save.

19. Restart your application server so these changes can take effect.

However, do not attempt to sign in until completing the steps described in the next section.

Configuring WebFOCUS as a Service Provider for Salesforce.com

To configure WebFOCUS as a service provider for Salesforce.com:

1. Log on to the new domain you created for Salesforce.com.
2. In the left pane under Administer, expand *Security Controls* and select *Identity Provider*.

3. Click *Service Providers are now created via Connected Apps. Click here*, as shown in the following image.

Identity Provider [Help for this Page](#)

Enable Salesforce.com as an identity provider so you can use single sign-on with other web sites, and define the appropriate service providers whose applications support single sign-on. You can switch to different service providers without having to log in again. [Learn more...](#)

Quick Tips

- Certificates and Keys
- About Single Sign-On
- My Domain

Identity Provider Setup [Edit](#) [Disable](#) [Download Certificate](#) [Download Metadata](#)

▼ **Details**

Issuer	https://bigmn-dev-ed.my.salesforce.com		
--------	--	--	--

▼ **Currently chosen certificate details**

Label	SelfSignedCert_07Dec2017_145541	Unique Name	SelfSignedCert_07Dec2017_145541
Created Date	12/7/2017 6:55 AM	Expiration Date	12/7/2018 4:00 AM
Key Size	2048		

▼ **SAML Metadata Discovery Endpoints**

Salesforce Identity	https://bigmn-dev-ed.my.salesforce.com/weil-known/samlidp.xml
---------------------	---

Service Providers [Service Providers are now created via Connected Apps. Click here](#)

Name	Created Date
No Service Providers	

The New Connected App dialog opens, as shown in the following image.

Note: If you select the *Enable SAML* check box in the Web App Settings section of this dialog, then the SAML information that must be entered is displayed. You should copy this information from the *wfspMetadata.xml* file, as Salesforce.com does not provide the ability to upload WebFOCUS metadata as some other Identity Providers.

4. Values for the Assertion Consumer Service (ACS) URL and Entity ID should be copied from the *wfspMetadata.xml* file.


You can search for *entityID* within the *wfspMetadata.xml* file.

- Click Save to save these configuration settings.
- Click the gear icon to access the Setup menu, as shown in the following image.



- In the left pane under PLATFORM TOOLS, expand *Apps*, *Connected Apps*, and then *Manage Connected Apps*.
- Select the *Label* of the Connected App that was created (for example, WebFOCUS).

The configuration pane for the Connected App (WebFOCUS) is displayed, as shown in the following image.

Connected App		WebFOCUS		Printable View Help
Back to List: Connected Apps				
Connected App Detail		Edit Policies		
		Version 1 Description WebFOCUS 8202		
System Info				
Installed By	Ben Naphthali	Installed Date	12/8/2017 9:04 AM	
Last Modified By	Ben Naphthali	Last Modified Date	12/8/2017 9:36 AM	
Basic Information				
Info URL			Start URL	https://bmn-8202.ibi.com/ibi_apps/
			Mobile Start URL	
SAML Service Provider Settings				
Entity Id	https://bmn-8202.ibi.com/ibi_apps/tp	ACS URL	https://bmn-8202.ibi.com/ibi_apps/tp/saml/SSO/alias/bmn8202	
Subject Type	Username	Issuer	https://bmn-dev-ed.my.salesforce.com	
Idp Certificate	SelfSignedCert_07Dec2017_145541			
Name ID Format	urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified			
Verify Request Signatures				
Enable Single Logout	Disabled			
SAML Login Information				
View and download SAML endpoint metadata for your organization, communities, or custom domains.				
Your Organization	Download Metadata			
IDP-Initiated Login URL	https://bmn-dev-ed.my.salesforce.com/idp/login?app=0sp1000000CaJk			
SP-Initiated POST Endpoint	https://bmn-dev-ed.my.salesforce.com/idp/endpoint/HttpPost			
SP-Initiated Redirect Endpoint	https://bmn-dev-ed.my.salesforce.com/idp/endpoint/HttpRedirect			
Metadata Discovery Endpoint	https://bmn-dev-ed.my.salesforce.com/well-known/saml/dfp/WebFOCUS.xml			
Single Logout Endpoint	https://bmn-dev-ed.my.salesforce.com/services/auth/dfp/saml2/logout			

To allow users to be able to access WebFOCUS as a Service Provider, you must add profiles to this Connected App.

9. Scroll down and select *Manage Profiles*, as shown in the following image.

The image shows a screenshot of the Salesforce Connected App configuration page. The page is divided into several sections:

- Custom Connected App Handler:** Includes fields for "Apex Plugin Class" and "Run As".
- User Provisioning Settings:** Contains a checkbox for "Enable User Provisioning" with a help icon.
- Trusted IP Range for OAuth Web server flow:** Shows "No application-defined IP ranges".
- Profiles:** Displays "No profiles associated with this app." and a button labeled "Manage Profiles" which is highlighted with a red rectangular box.
- Permission Sets:** Shows "No permission sets associated with this app." and a button labeled "Manage Permission Sets".
- Custom Attributes:** Shows "No Custom Attributes" and a button labeled "New".

The Application Profile Assignment dialog opens, as shown in the following image.

Application Profile Assignment
[Back to Connected App Details](#)

Select the appropriate profiles to choose which users have access to this application.

Select	Profiles	Description
<input type="checkbox"/>	Analytics Cloud Integration User	
<input type="checkbox"/>	Analytics Cloud Security User	
<input type="checkbox"/>	Authenticated Website	
<input type="checkbox"/>	Authenticated Website	
<input type="checkbox"/>	Chatter External User	
<input type="checkbox"/>	Chatter Free User	
<input type="checkbox"/>	Chatter Moderator User	
<input type="checkbox"/>	Contract Manager	
<input type="checkbox"/>	Cross Org Data Proxy User	
<input type="checkbox"/>	Custom: Marketing Profile	
<input type="checkbox"/>	Custom: Sales Profile	
<input type="checkbox"/>	Custom: Support Profile	
<input type="checkbox"/>	Customer Community Login User	
<input type="checkbox"/>	Customer Community Plus Login User	
<input type="checkbox"/>	Customer Community Plus User	
<input type="checkbox"/>	Customer Community User	
<input type="checkbox"/>	Customer Portal Manager Custom	
<input type="checkbox"/>	Customer Portal Manager Standard	
<input type="checkbox"/>	External Identity User	
<input type="checkbox"/>	Force.com - App Subscription User	
<input checked="" type="checkbox"/>	Force.com - Free User	
<input type="checkbox"/>	Gold Partner User	
<input type="checkbox"/>	High Volume Customer Portal	
<input type="checkbox"/>	High Volume Customer Portal User	
<input type="checkbox"/>	Identity User	
<input type="checkbox"/>	Marketing User	
<input type="checkbox"/>	Partner App Subscription User	
<input type="checkbox"/>	Partner Community Login User	
<input type="checkbox"/>	Partner Community User	
<input type="checkbox"/>	Read Only	
<input type="checkbox"/>	Silver Partner User	
<input type="checkbox"/>	Solution Manager	
<input type="checkbox"/>	Standard Platform User	
<input type="checkbox"/>	Standard User	
<input checked="" type="checkbox"/>	System Administrator	
<input type="checkbox"/>	Work.com Only User	

10. Select the user(s) that will access WebFOCUS using Salesforce.com as a SAML authentication provider, and then click Save.

11. Return to the configuration pane for the Connected App (WebFOCUS), as shown in step 8.

The profiles you assigned to the Connected App (WebFOCUS) are now listed under the Profiles section, as shown in the following image.



12. You can test logging on to WebFOCUS by using the URL you configured.

https://server.ibi.com/ibi_apps/

Redirect to your Salesforce.com login page.

13. Enter your Salesforce.com user credentials.

You are redirected to WebFOCUS. This is a Service Provider initiated login (login from WebFOCUS).

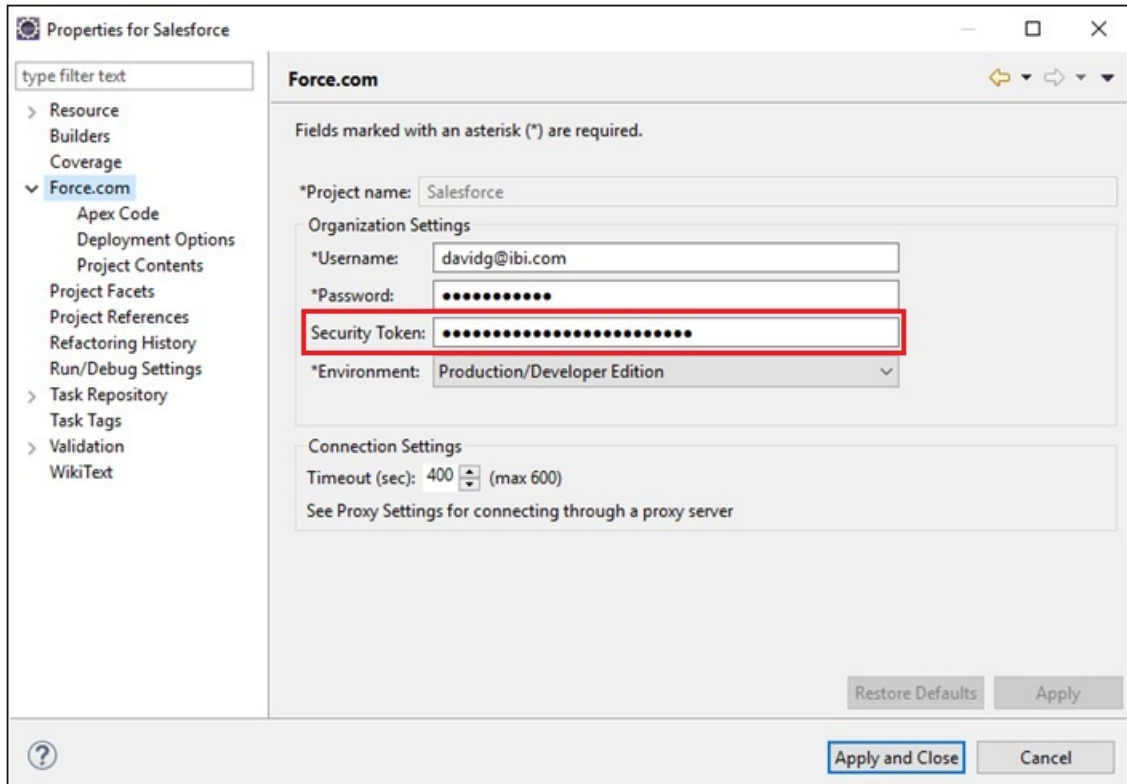
14. If you log on to Salesforce.com using its URL, then any WebFOCUS content will also be logged on from the back-end.

This is an Identity Provider initiated login.

Programming Solutions

Embedding a URL in Salesforce.com (SFDC) to run a WebFOCUS report is a simple example of embedding WebFOCUS BI content. To accomplish more advanced embedding techniques in SFDC (for example, calling multiple web services, checking cookies, and so on), some programming will be required.

The Security Token is generated on the SFDC website and is required for any API program to access SFDC, such as Force.com IDE.



Apex classes are similar to Java classes, but have special codes for SFDC. Likewise, Visualforce pages are similar to Java Server Pages (JSPs).

If you do not want to integrate SAML, and would prefer to do something like a web service signOn or have a need to run multiple web services, then you need to use what SFDC calls *Chained Callouts*.

A *Callout* is the term used by SFDC for calling an external web service.

Chained Callouts means that one Callout will need the response from the previous instance as in a signOn and then an execution of a procedure, schedule, and so on.

For example:

```

public with sharing class ChainedContinuationController {
    // Unique label for the initial callout request
    public String requestLabel1;
    // Unique label for the chained callout request
    public String requestLabel2;
    // Result of initial callout
    public String result1 {get;set;}
    // Result of chained callout
    public String result2 {get;set;}
    // Endpoint of long-running service
    private static final String LONG_RUNNING_SERVICE_URL1 =
        'http://pmdev.ibi.com/ibi_apps/rs/ibfs';
    private static final String LONG_RUNNING_SERVICE_URL2 =
        'http://pmdev.ibi.com/ibi_apps/rs/ibfs/WFC/Repository/David_SFDC/
Car_Report.fex';
    // Action method
    public Object invokeInitialRequest() {
        // Create continuation with a timeout
        Continuation con = new Continuation(60);
        // Set callback method
        con.continuationMethod='processInitialResponse';
        // Create first callout request
        String body1 =
'IBIRS_action=signOn&IBIRS_userName=david&IBIRS_password=david';
        HttpRequest req = new HttpRequest();
        req.setMethod('POST');
        req.setBody(body1);
        req.setHeader('Content-Type', 'application/x-www-form-urlencoded');
        req.setEndpoint(LONG_RUNNING_SERVICE_URL1);
        // Add initial callout request to continuation
        this.requestLabel1 = con.addHttpRequest(req);
        // Return the continuation
    }
}

```

```

// Callback method for initial request
public Object processInitialResponse() {
    // Get the response by using the unique label
    HttpResponse response = Continuation.getResponse(this.requestLabel1);
    // Set the result variable that is displayed on the Visualforce page
    this.result1 = response.getBody();
    String cookie = response.getHeader('Set-Cookie');
    Continuation chainedContinuation = null;
    // Chain continuation if some condition is met
    //if (response.getBody().toLowerCase().contains('expired')) {
    // Create a second continuation
    chainedContinuation = new Continuation(60);
    // Set callback method
    chainedContinuation.continuationMethod='processChainedResponse';
    // Create callout request
    HttpRequest req = new HttpRequest();
    req.setMethod('GET');
    req.setHeader('Content-Type', 'application/x-www-form-urlencoded');
    req.setHeader('Cookie', cookie);
    req.setEndpoint(LONG_RUNNING_SERVICE_URL2 + '?IBIRS_action=run');
    // Add callout request to continuation
    this.requestLabel2 = chainedContinuation.addHttpRequest(req);
    //}
    // Start another continuation
    return chainedContinuation;
}
// Callback method for chained request
public Object processChainedResponse() {
    // Get the response for the chained request
    HttpResponse response = Continuation.getResponse(this.requestLabel2);
    // Set the result variable that is displayed on the Visualforce page
    this.result2 = response.getBody();
    // Return null to re-render the original Visualforce page
    return null;
}
}

```

The following is an example of the syntax used to define a Visualforce page.

```

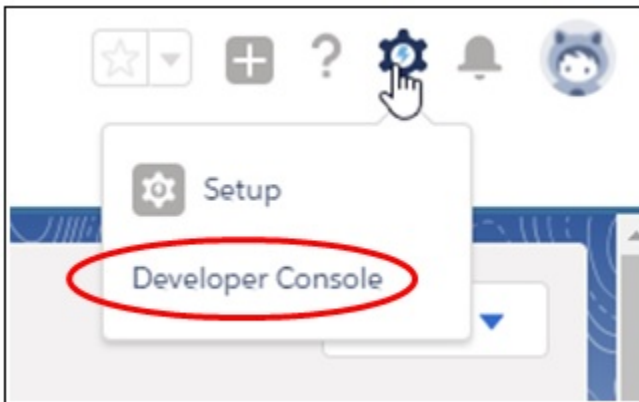
<apex:page controller="ChainedContinuationController" showChat="false"
showHeader="false">
  <apex:form >
    <!-- Invokes the action method when the user clicks this button. -->
    <apex:commandButton action="{!invokeInitialRequest}" value="Start
Request" reRender="panel" />
  </apex:form>

  <apex:outputPanel id="panel">
    <!-- Displays the response body of the initial callout. -->
    <apex:outputText value="{!result1}" />

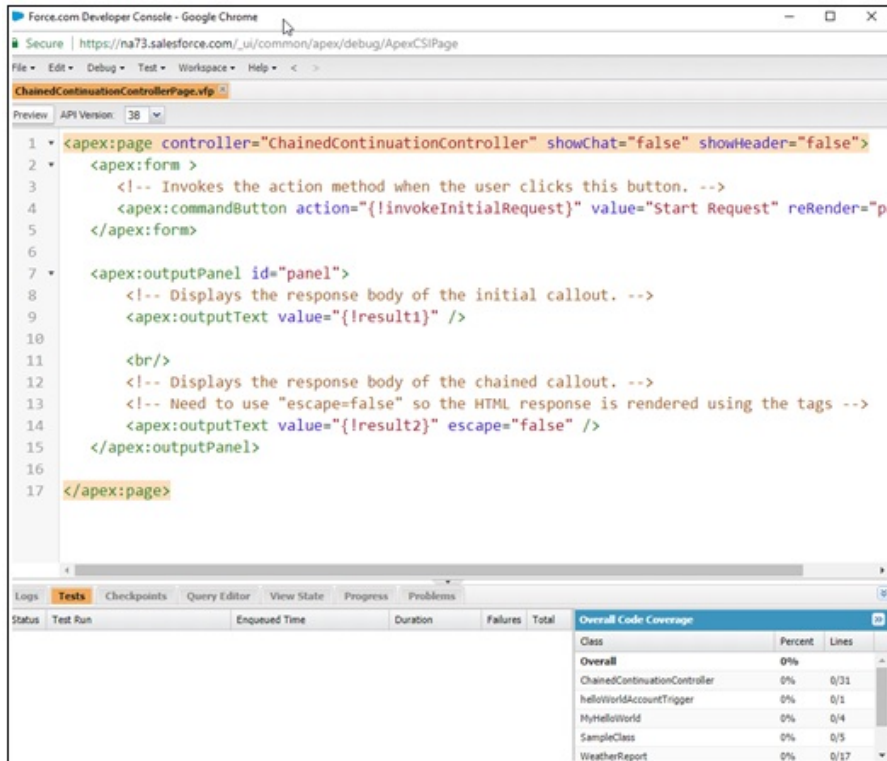
    <br/>
    <!-- Displays the response body of the chained callout. -->
    <!-- Need to use "escape=false" so the HTML response is rendered
using the tags -->
    <apex:outputText value="{!result2}" escape="false" />
  </apex:outputPanel>
</apex:page>

```

You can access the Developer Console, as shown in the following image.



The Developer Console is where you define and configure a Visualforce page, as shown in the following image.



The following image shows sample output that is generated using the Developer Console.

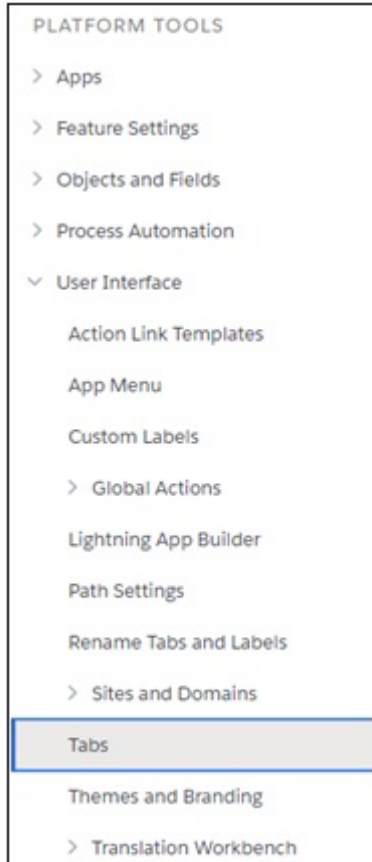
The screenshot shows a browser window with the URL `https://c.na73.visual.force.com/apex/ChainedContinuationControllerPage?core.apexpages.request.devconsole=1`. Below the browser window, the raw REST API response is displayed as XML. The XML contains a table of car data. The table has the following columns: COUNTRY, CAR, MODEL, DEALER_COST, RETAIL_COST, and SALES. The data is grouped by country: ENGLAND, FRANCE, ITALY, JAPAN, and W GERMANY.

COUNTRY	CAR	MODEL	DEALER_COST	RETAIL_COST	SALES
ENGLAND	JAGUAR	V12XKE AUTO	7,427	8,878	0
		XJ12L AUTO	11,194	13,491	12000
	JENSEN	INTERCEPTOR III	14,940	17,850	0
	TRIUMPH	TR7	4,292	5,100	0
FRANCE	PEUGEOT	504 4 DOOR	4,631	5,610	0
ITALY	ALFA ROMEO	2000 4 DOOR BERLINA	4,915	5,925	4800
		2000 GT VELOCE	5,660	6,820	12400
		2000 SPIDER VELOCE	5,660	6,820	13000
	MASERATI	DORA 2 DOOR	25,000	31,500	0
JAPAN	DATSUN	B210 2 DOOR AUTO	2,626	3,139	43000
	TOYOTA	COROLLA 4 DOOR DIX AUTO	2,886	3,339	35030
W GERMANY	AUDI	100 LS 2 DOOR AUTO	5,063	5,970	7800
	BMW	2002 2 DOOR	5,800	5,940	8950
		2002 2 DOOR AUTO	6,000	6,355	8900
		3.0 Si 4 DOOR	10,000	13,752	14000
		3.0 Si 4 DOOR AUTO	11,000	14,123	18940
	530i 4 DOOR	8,300	9,097	14000	
	530i 4 DOOR AUTO	8,400	9,495	15600	

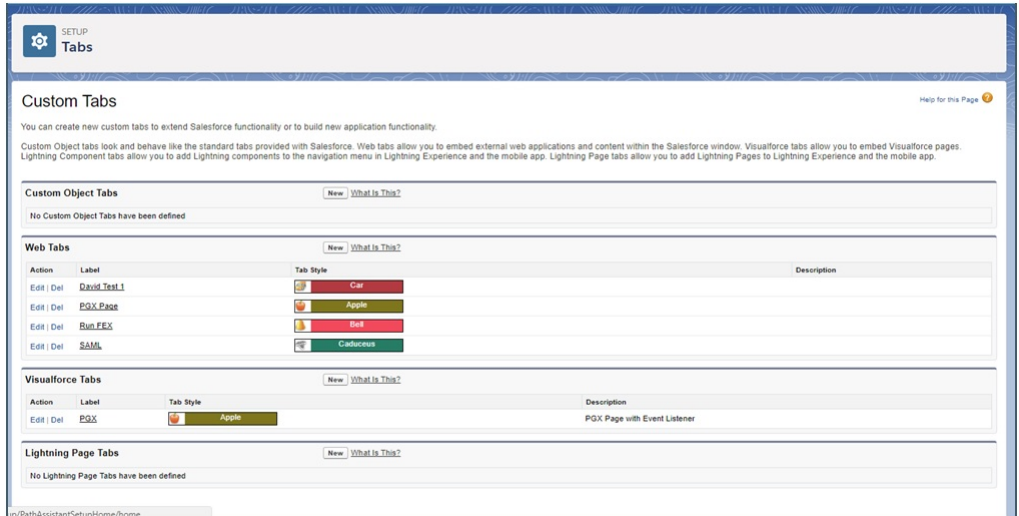
At the bottom of the console, the following text is visible: `SfdcApp Visualforce.viewstate ViewstateSender.resend();`

Adding a Visualforce page to your SFDC dashboard is simple:

1. In the left pane under PLATFORM TOOLS, expand *User Interface* and then click *Tabs*, as shown in the following image.

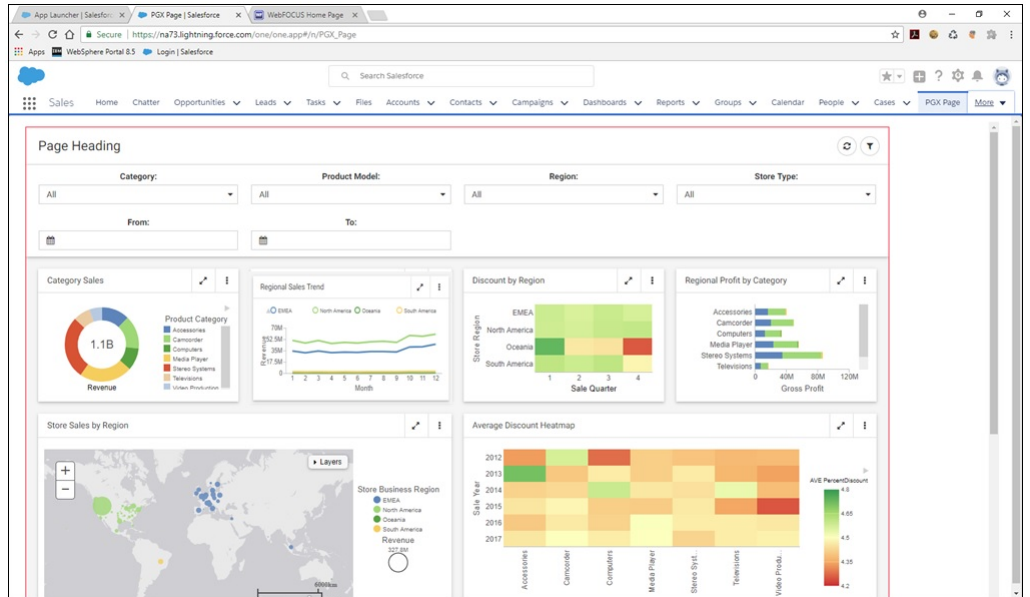


The Tabs pane opens, as shown in the following image.



2. On the right side, under the Visualforce Tabs section, click *New*.
3. Select the name of your Visualforce page from the Visualforce Page drop-down list.
4. In the Display Properties area, enter a value in the Tab Label field, Tab Name field, and then select a Tab Style from the drop-down list.
5. Click *Next*.
6. Do not make any changes for the *Add to Profiles* and *Add to Custom Apps* steps that follow.
7. Click *Save*.

- You can now select this new tab containing your Visualforce page when you open a SFDC application in the dashboard, as shown in the following image.



Note: A Visualforce page may also contain HTML and JavaScript. For example:

```

<apex:page >
<html lang="en" xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta charset="utf-8" />
<title></title>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
<script type="text/javascript">
    var csrf_name;
    var csrf_value;

    $(document).ready(function () {
        var contentType = "application/x-www-form-urlencoded; charset=utf-8";

        var webMethod = "https://as8200.ibi.com:8443/ibi_apps/rs/ibfs";
        var IBIRS_action = "signOn";
        var IBIRS_userName = "admin";
        var IBIRS_password = "admin";
        var parameters = 'IBIRS_action=' + IBIRS_action + '&IBIRS_userName=' + IBIRS_userName + '&IBIRS_password=' + IBIRS_password;

        if (window.XDomainRequest)
            contentType = "text/plain";

        $.ajax({
            type: "POST",
            url: webMethod,
            data: parameters,
            dataType: "xml",
            async: false,
            xhrFields: {
                withCredentials: true
            },
            crossDomain: true,
            contentType: contentType,
            success: function () {
                $('<div class="wrapper">').attr('src', 'https://as8200.ibi.com:8443/ibi_apps/rs/ibfs/WFC/Repository/Retail_Samples/I
            },
            error: function (jqXHR, textStatus, errorThrown) {
                alert("You can not send Cross Domain AJAX requests: " + errorThrown);
            }
        });
    });

    window.addEventListener('message', onHeightChanged, false);

    function onHeightChanged(event)
    {
        if (event.data && event.data.message_name == 'height_changed')
        {
            $('<div class="wrapper">').innerHeight(event.data.page_height + 10); // add 10 to account for different things, like bor
        }
    }
</script>
<style>
<div class="wrapper" {
    width: 90%;
    height: 100px;
    border: 1px solid red;
}
</style>
</head>
<body>

    <div class="wrapper"></div>

</body>
</html>
</apex:page>

```


This appendix describes the format and structure of URL calls that can be used to directly access WebFOCUS components (for example, in an application or outside of a third-party portal environment).

In this chapter:

- Report Component
- Deferred Status Component
- Resource Tree Component
- Portal Component
- Portal Tree Component

Report Component

Use the following URL to directly access the WebFOCUS Report component:

```
http://hostname:port/context/report.ops?userrunonly=yes|no&showrefresh=yes|no&gn=n&usescrollbars=yes|no&showtimestamp=yes|no
```

The following table lists and describes the parameters for the WebFOCUS Report component:

Parameter	Description
hostname	Host name (or IP address) of the machine that is hosting the WebFOCUS client.
port	Port number to the WebFOCUS client.
context	The application context path. For example: <code>/ibi_apps</code>

Parameter	Description
userrunonly	<p>Used to control whether or not a user is allowed to change the content displayed in the Report component.</p> <p>If this parameter is set to <i>no</i>, which is the default, then the user will be able to select a WebFOCUS content item and its display mode.</p> <p>If this parameter is set to <i>yes</i>, then the Report component will display a default content item, such as a report, to any user that does not have Managed Reporting administrator privileges, and the Select Report menu option will not be available, preventing the user from changing the content item selected by default.</p>
showrefresh	<p>If this parameter is set to <i>no</i>, then the Refresh option will not be available on the Report component menu and the user will not be able to manually trigger the content item to refresh.</p> <p>If this parameter is set to <i>yes</i>, which is the default, then the Refresh option will be displayed on the Report component menu and the user can manually trigger the content item to refresh.</p>
gn	<p>This parameter is used to distinguish between multiple instances of the WebFOCUS Report component and retain its attributes (for example, width, height, and refresh rate).</p> <p>If you add three Report components to your portal page, then you must specify a unique gn value for each instance of that Report component.</p>

Parameter	Description
usescrollbars	<p>If this parameter is set to <i>no</i>, then the Report component will not display scrollbars for the content item displayed.</p> <p>If this parameter is set to <i>yes</i>, which is the default, then the Report component will display scrollbars for the content item displayed.</p> <p>This is particularly useful for content items that are larger than the component width and height, such as reports with several rows.</p>
showtimestamp	<p>If this parameter is set to <i>no</i>, which is the default, then the Report component will not display the time stamp.</p> <p>If this parameter is set to <i>yes</i>, then the Report component will display the time stamp.</p>

Deferred Status Component

Use the following URL to directly access the WebFOCUS Deferred Status component:

<http://hostname:port/context/deferstatus.ops>

The following table lists and describes the parameters for the WebFOCUS Deferred Status component:

Parameter	Description
hostname	Host name (or IP address) of the machine that is hosting the WebFOCUS client.
port	Port number to the WebFOCUS client.

Parameter	Description
context	The application context path. For example: <code>/ibi_apps</code>

Resource Tree Component

Use the following URL to directly access the WebFOCUS Resource Tree component:

<http://hostname:port/context/domain.ops>

The following table lists and describes the parameters for the WebFOCUS Resource Tree component:

Parameter	Description
hostname	Host name (or IP address) of the machine that is hosting the WebFOCUS client.
port	Port number to the WebFOCUS client.
context	The application context path. For example: <code>/ibi_apps</code>

Portal Component

Use the following URL to directly access the WebFOCUS Portal component:

<http://hostname:port/context/portal.ops?portalPath=>

The following table lists and describes the parameters for the WebFOCUS Portal component:

Parameter	Description
hostname	Host name (or IP address) of the machine that is hosting the WebFOCUS client.
port	Port number to the WebFOCUS client.

Parameter	Description
context	The application context path. For example: <code>/ibi_apps</code>
portalPath	The path to the WebFOCUS Business Intelligence (BI) portal.

Portal Tree Component

Use the following URL to directly access the WebFOCUS Portal Tree component:

`http://hostname:port/context/portaltree.ops`

The following table lists and describes the parameters for the WebFOCUS Portal Tree component:

Parameter	Description
hostname	Host name (or IP address) of the machine that is hosting the WebFOCUS client.
port	Port number to the WebFOCUS client.
context	The application context path. For example: <code>/ibi_apps</code>

WebFOCUS Embedded Business Intelligence Demonstration Application

In WebFOCUS Release 8.2 Version 03, a demonstration (“demo”) application is available for deployment that allows you to embed WebFOCUS Business Intelligence (BI) content, analytics, and functionality (features) as an external application. This embedded BI demo application that is packaged with WebFOCUS enables you to explore the iframe and web services embedding options.

During the configuration process, you will also learn how the Trusted Ticket Authentication feature in WebFOCUS Release 8.2 Version 03 can be used to implement single sign-on (SSO), which is an important developer consideration for embedded BI applications.

This section serves as a getting started guide, which describes how to install, configure, and use the embedded Business Intelligence (BI) demonstration (“demo”) application that is packaged with WebFOCUS Release 8.2 Version 03.

In this chapter:

- [Installing the Embedded Business Intelligence Demonstration Application](#)
 - [Configuring the Embedded Business Intelligence Demonstration Application](#)
 - [Using the Embedded Business Intelligence Demonstration Application \(Fintoso Financial\)](#)
 - [Additional Considerations for Embedded Business Intelligence](#)
 - [Customizing the Embedded Business Intelligence Demonstration Application \(Fintoso Financial\)](#)
 - [Troubleshooting](#)
 - [Appendix: Detailed Request/Response Flow for the Embedded Business Intelligence Demonstration Application](#)
-

Installing the Embedded Business Intelligence Demonstration Application

This section describes how to install the sample embedded content, sample user (ffadv), and the embedded business intelligence (BI) demo application (*Fintoso Financial*).

In WebFOCUS Release 8.2 Version 03, the Fintoso Financial embedded BI demo is automatically installed for you into the following directory:

```
<drive>:\ibi\WebFOCUS82\samples\embedded_demo
```

Here you will find the following components that are referenced and used by the embedded BI demo:

- embeddemo.war.** A redesigned application .war file that you must deploy to your application server (for example, Apache Tomcat).
- fintoso_domain_CM_v01.zip.** A new Change Management package that deploys the Fintoso Financial domain into your WebFOCUS environment along with sample financial data.

- ❑ **fintoso_users.csv.** A CSV file containing the predefined sample user ID (ffadv), which is used to log in to the Fintoso Financial embedded BI demo application.

These components must be installed in the following order:

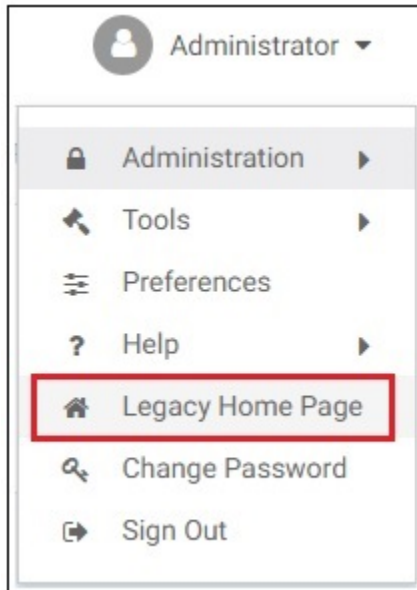
1. Sample embedded content (fintoso_domain_CM_v01.zip).
2. Sample predefined user (ffadv) (fintoso_users.csv).
3. Application .war file (embeddemo.war).

Installing the Sample Embedded Content

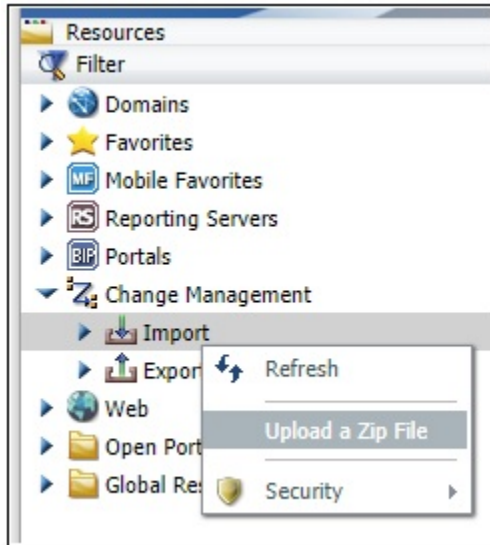
For improved performance and quicker deployment, a new financial data set has been packaged with the Fintoso Financial embedded BI demo. You are no longer required to build WF RETAIL as a data source for the embedded demo application. Instead, a set of FTM files representing sample financial data is included with the new Change Management package (fintoso_domain_CM_v01.zip). As a result, simply deploying the new Change Management package will also deploy all of the required data that is used by the new embedded BI demo at the same time.

Procedure: How to Install the Sample Embedded Content

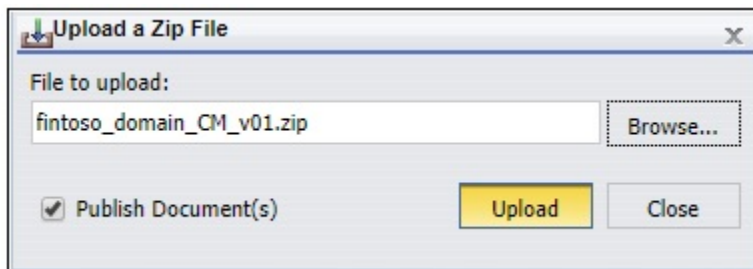
1. Sign in to WebFOCUS Release 8.2 Version 03 as an *administrator*.
2. From the WebFOCUS Home Page, click *Administrator* in the upper-right and then click *Legacy Home Page* from the menu, as shown in the following image.



3. In the Resources tree on the left pane, expand *Change Management*, right-click the *Import* node, and then select *Upload a Zip File* from the context menu, as shown in the following image.

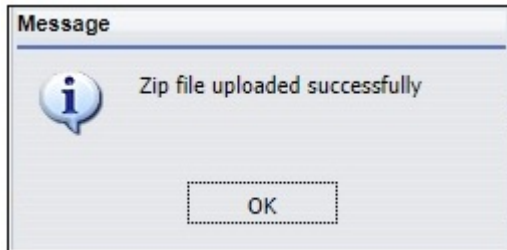


The Upload a Zip File dialog opens, as shown in the following image.

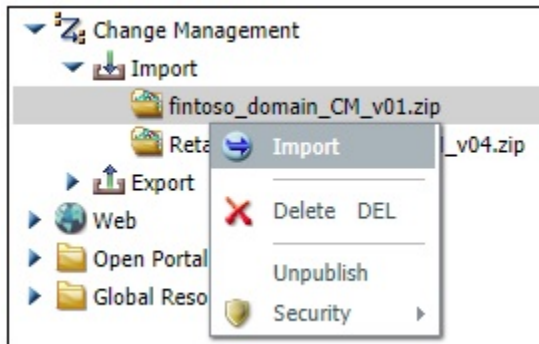


4. Browse to the following directory and select the *fintoso_domain_CM_v01.zip* file.
`<drive>:\WebFOCUS82\samples\embedded_demo\fintoso_domain_CM_v01.zip`
5. Click *Upload*.

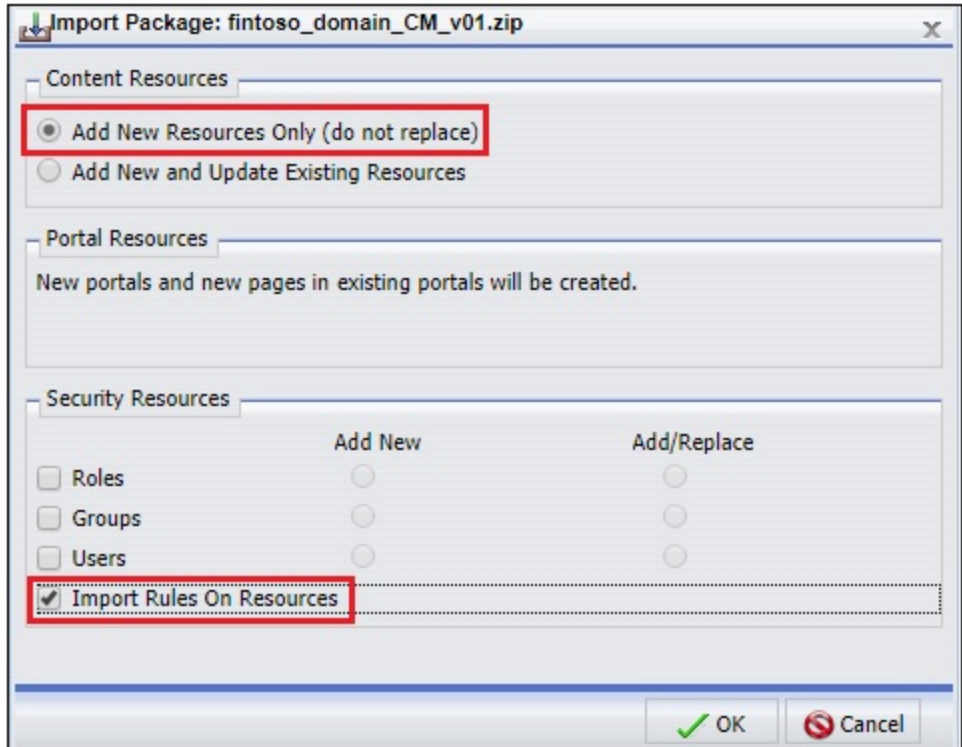
The following message is displayed.



6. Click *OK*.
7. Under the *Import* node, right-click the *fintoso_domain_CM_v01.zip* file and select *Import* from the context menu, as shown in the following image.

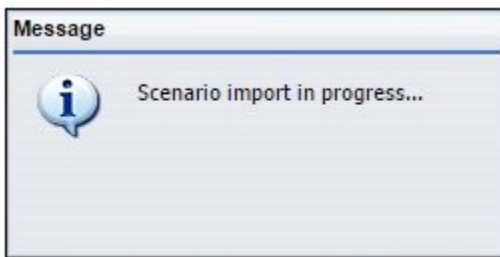


The Import Package: fintoso_domain_CM_v01.zip file dialog opens, as shown in the following image

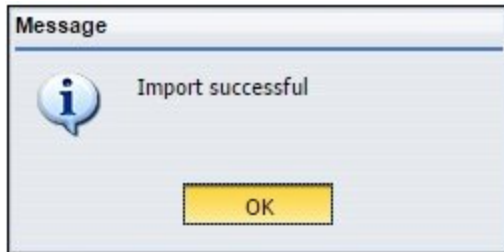


8. Perform the following steps:
 - a. In the Content Resources area, select *Add New Resources Only (do not replace)*.
 - b. In the Security Resources area, select *Import Rules On Resources*.
9. Click OK.

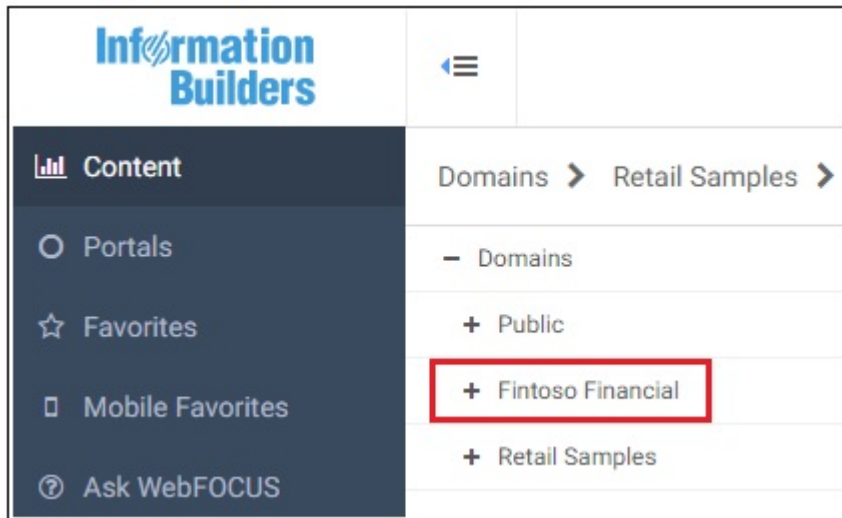
The *Scenario import in progress* message is displayed, as shown in the following image.



Once this process has completed, an *Import successful* message is displayed, as shown in the following image.



10. Click *OK*.
11. Return to the WebFOCUS Home Page where you will now find Fintoso Financial listed as a new domain, as shown in the following image.

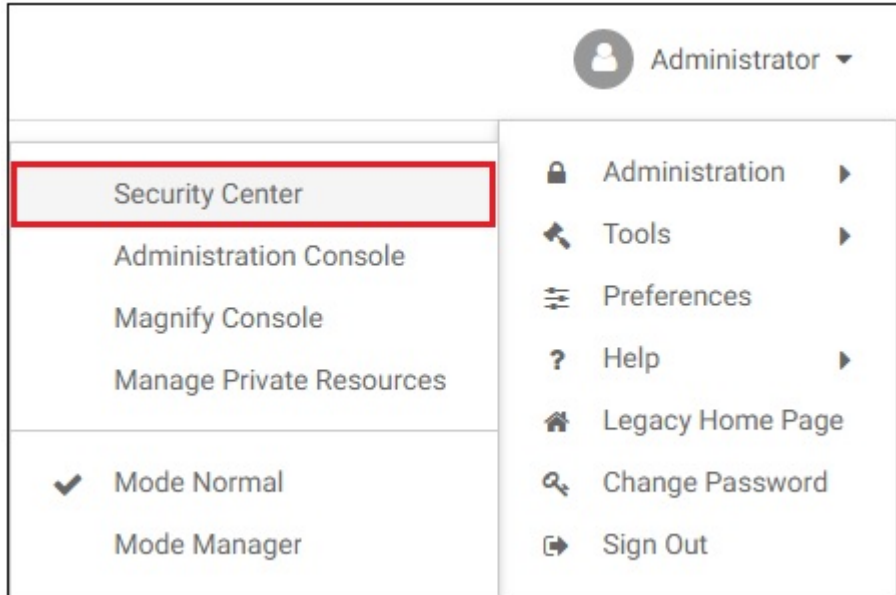


Importing the Sample User (ffadv)

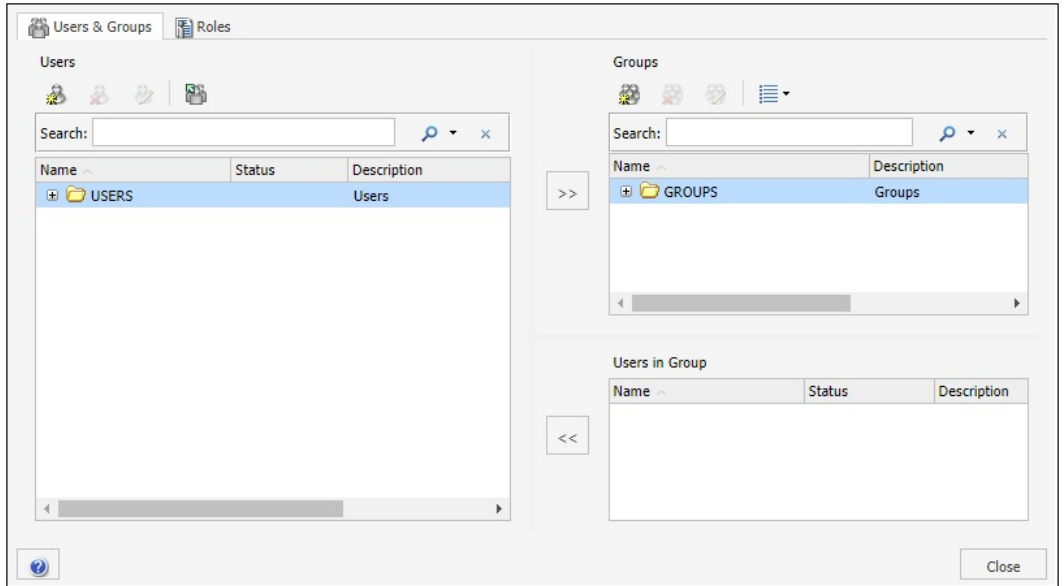
The embedded BI demo application (*Fintoso Financial*) includes a CSV file that contains a predefined sample user ID (ffadv). You must use the ffadv user ID to log in to the Fintoso Financial embedded BI demo.

Procedure: How to Import the Sample User (ffadv) Using the Security Center

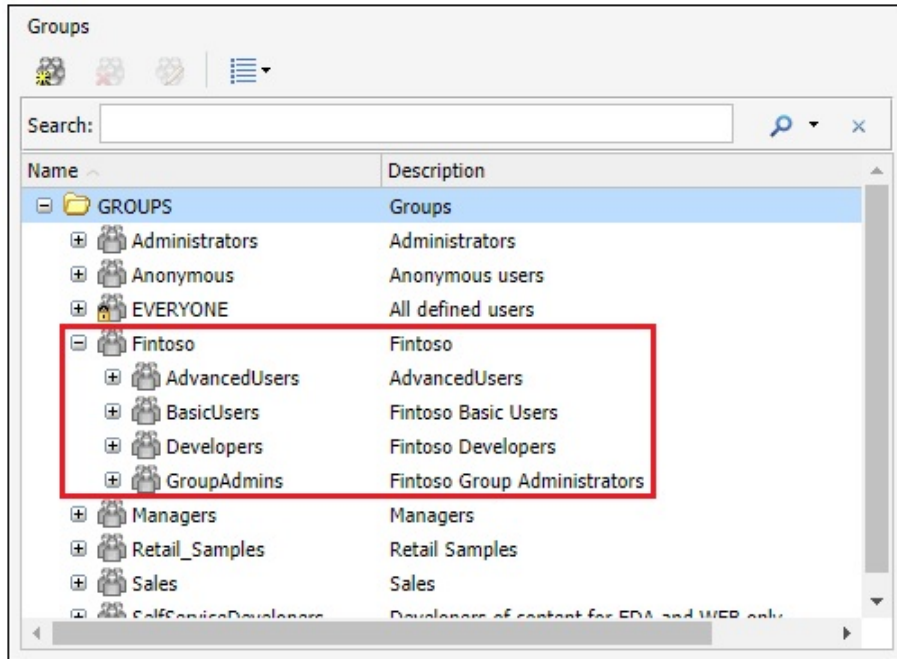
1. From the WebFOCUS Home Page, click *Administrator* in the upper-right, select *Administration*, and then click *Security Center* from the menu, as shown in the following image.



The Security Center opens, as shown in the following image.



2. In the Groups section, expand the main *Fintoso* group node, as shown in the following image.

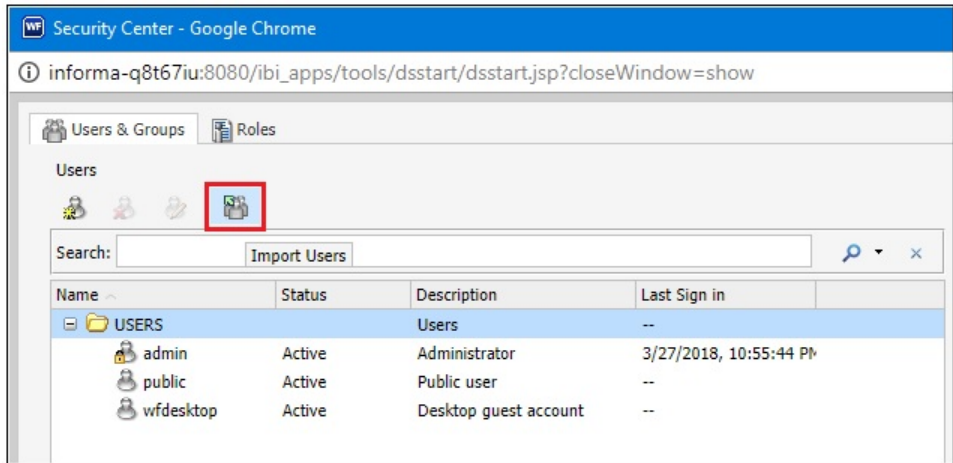


Ensure that the following groups are available:

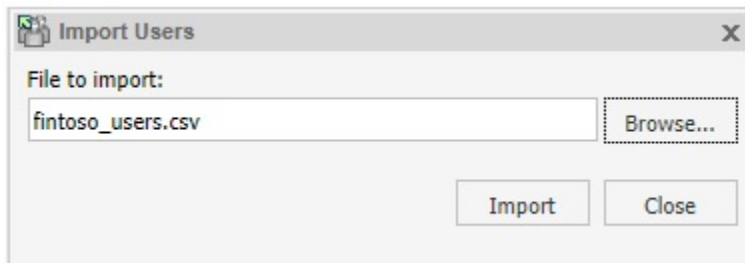
- AdvancedUsers
- BasicUsers
- Developers
- GroupAdmins

Note: These new groups are automatically created when you import the Change Management package (fintoso_domain_CM_v01.zip) into your WebFOCUS environment.

3. In the Users section, click *Import Users*, as shown in the following image.



The Import Users dialog opens, as shown in the following image.



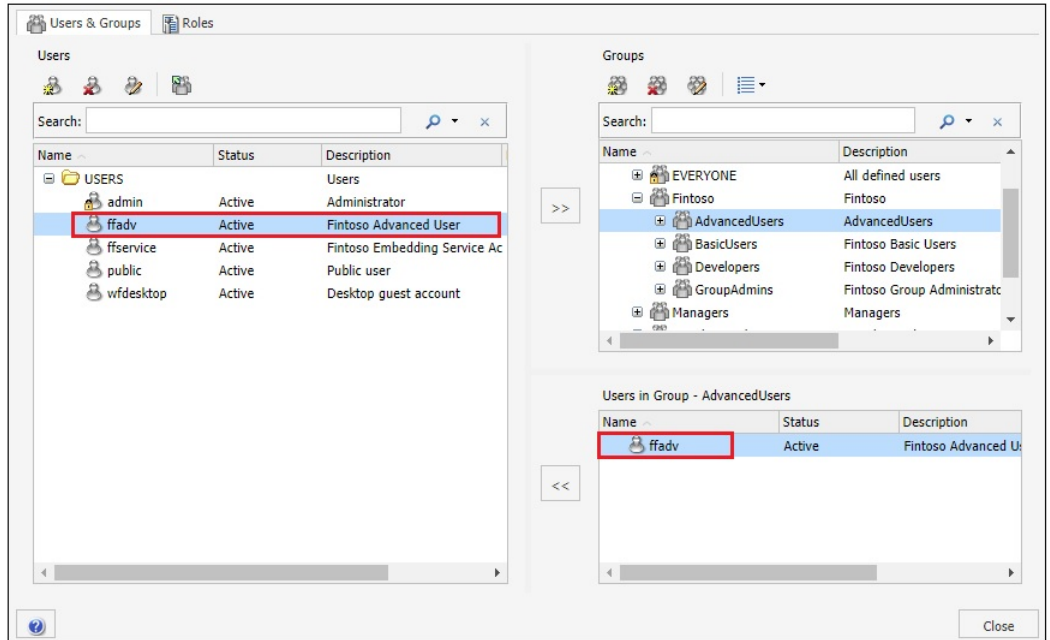
4. Browse to the following directory and select the *fintoso_users.csv* file.
`<drive>:\WebFOCUS82\samples\embedded_demo\fintoso_users.csv`
5. Click *Import*.

The following message is displayed.



- Click OK.

The Security Center is refreshed and now lists the new user ID (ffadv) that you imported under the Users section and Groups section accordingly, as shown in the following image.



- Click Close in the lower-right to exit from the Security Center.

Installing the Embedded BI Demo Application (Fintoso Financial)

The embedded BI demo application (*Fintoso Financial*) was developed using Bootstrap, JSP, Java, and JavaScript. The entire application is packaged as a web archive file (*embeddemo.war*), which you can easily deploy to any servlet container or Java application server. You can also modify the application to explore various embedding scenarios. For more information, see [Customizing the Embedded Business Intelligence Demonstration Application \(Fintoso Financial\)](#) on page 472.

Procedure: How to Install the Embedded BI Demo Application (Fintoso Financial)

- Navigate to the following folder in your WebFOCUS installation where the *embeddemo.war* file is located:

`<drive>:\ibi\WebFOCUS82\samples\embedded_demo\embeddemo.war`

- Deploy the *embeddemo.war* file on your application server (for example, Apache Tomcat) by following the corresponding content for your application server.

Note: It is recommended that you successfully deploy the embedded BI demo application in a same origin scenario even if your goal is to explore the cross-origin scenario. This means that you should first deploy the *embeddemo.war* file on the application server where WebFOCUS Release 8.2 Version 03 is being hosted.

The following steps describe how to deploy the *embeddemo.war* file on the instance of the Apache Tomcat application server that is available with the WebFOCUS installation:

- a. Copy the *embeddemo.war* file to the following folder in your WebFOCUS installation:

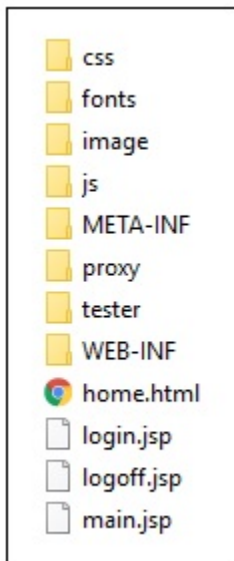
```
<drive>:\ibi\tomcat\webapps
```

You are not required to stop or restart Apache Tomcat.

- b. Within a few seconds you will see a new subfolder called *embeddemo* that is automatically created. For example:

```
<drive>:\ibi\tomcat\webapps\embeddemo
```

This subfolder represents the deployed copy of the *embeddemo.war* file and is structured as shown in the following image.

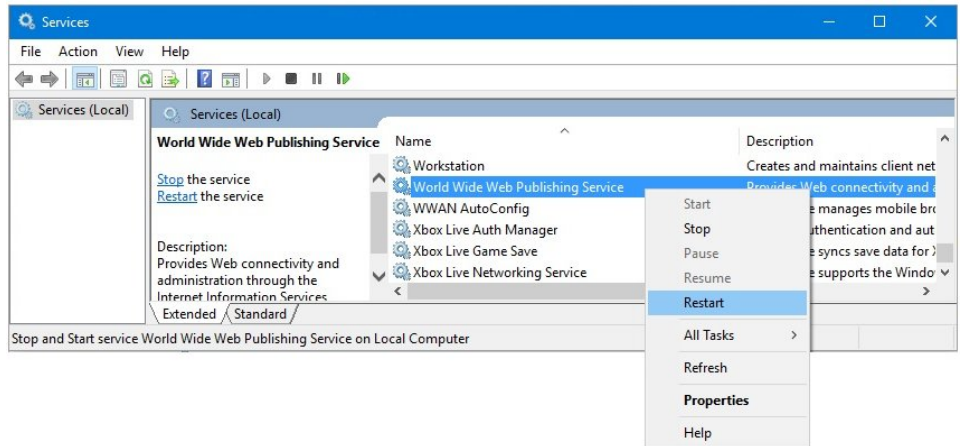


3. If you deployed WebFOCUS on the Apache Tomcat application server behind Microsoft Internet Information Services (IIS), then you will need to modify the Jakarta Connector configuration to pass requests to the embedded BI demo application through IIS to Apache Tomcat.
 - a. Navigate to the following folder on your Windows operating system:

```
C:\Program Files\Apache Software Foundation\Jakarta Isapi Redirector
\conf
```

- b. Open the `uriworkermap.properties` file using a text editor.
- c. Add the following two lines anywhere in this file:


```
/embeddemo/*=ajp13w
/embeddemo=ajp13w
```
- d. Save the `uriworkermap.properties` file.
- e. Restart the World Wide Web Publishing Service on Windows, as shown in the following image.



Required HTML 5 Chart Extensions

The Fintoso Financial embedded BI demo requires the following HTML 5 chart extensions to be installed on your WebFOCUS environment for several charts to display correctly:

- Sparkline KPI (com.ibi.kpi.sparkline)
- World Choropleth & Bubble Map (com.ibi.map.world)

These extensions can be downloaded and installed from the Information Builders public extension GitHub page. For more information on installing HTML 5 chart extensions, see the *WebFOCUS Security and Administration* documentation.

Configuring the Embedded Business Intelligence Demonstration Application

This section describes the configuration steps for the embedded business intelligence (BI) demo application (*Fintoso Financial*).

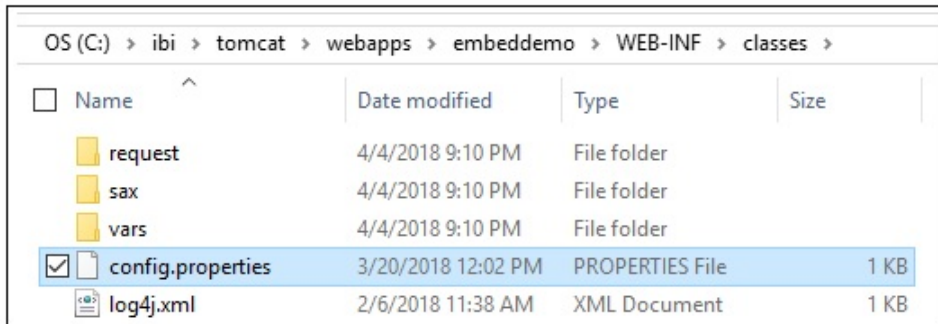
Tip: If you plan to configure the cross-origin scenario (dual web hosts), then it is recommended that you first successfully configure the same origin scenario (single web host).

Configuring a Back Channel Ticket Request

The configuration for the back channel ticket request made by the embedded BI demo application is specified in the `config.properties` file, which is located in the following folder:

`embeddemo\WEB-INF\classes\config.properties`

For example, if you deployed the embedded BI demo application (`embeddemo.war`) file on the Apache Tomcat application server that is available with the WebFOCUS installation, then the `config.properties` file is located in the folder path, as shown in the following image.



The default file configuration is shown below:

```
WF_TICKET_PROVIDER_URL=http://localhost:8080/ibi_apps
WF_TRUSTED_APPLICATION_NAME=IBIEmbeddingDemo
WF_HOST=http://localhost:8080/ibi_apps
USERIDS=ffadv
```

Make the following changes to the `WF_TICKET_PROVIDER_URL` setting, as required for your installation:

1. If your back channel request will need to use HTTPS, rather than HTTP, to access WebFOCUS, then change the protocol value accordingly.
2. If you are deploying the embedded BI demo application (*Fintoso Finacial*) on a different machine from WebFOCUS, then change `localhost` to the web host that the embedded BI demo application will use to access WebFOCUS on the back channel.

Note: If you are deploying a same origin scenario, then you can leave the host set to `localhost`.

3. If your back channel request will need to use a port other than 8080 to access WebFOCUS, then change the port value accordingly.

For example, if you are deploying a same origin scenario and you installed Apache Tomcat on port 80, then remove `:8080` from the value.

4. If you deployed WebFOCUS on a non-standard context path (for example, `/ibi_apps82`), then change the context path value accordingly.
5. Save any changes you made to the `config.properties` file.

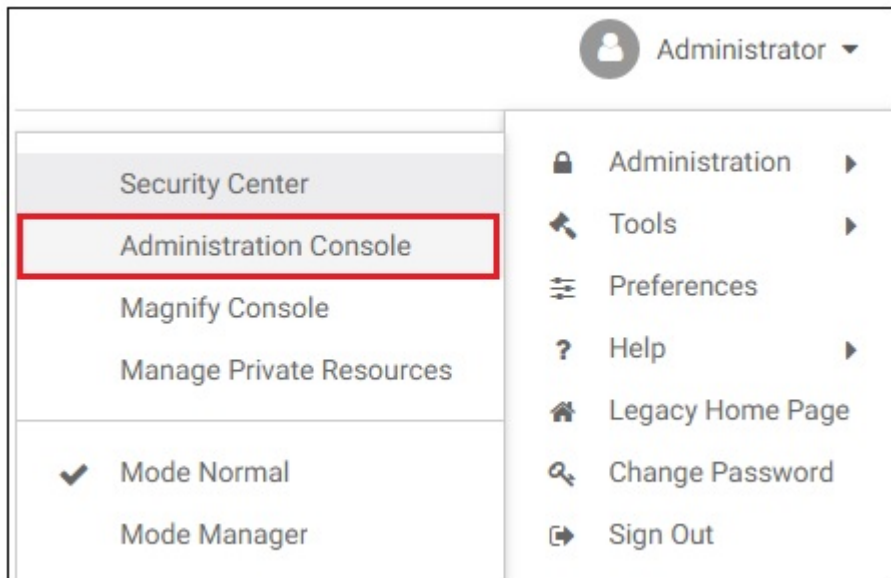
For now, leave the value of `WF_TRUSTED_APPLICATION_NAME` set to `IBIEmbeddingDemo`. A trusted application name is passed on the ticket request so WebFOCUS knows which trusted host configuration to reference when verifying the IP address of the request.

Configuring WebFOCUS

This section describes how to enable Trusted Ticket Authentication in WebFOCUS. Depending on your deployment scenario, you may also need to reconfigure Trusted Ticket Authentication. If you are configuring a cross-origin (dual web host) scenario, then you will need to configure additional cross-origin settings. For more information on configuring and using cross-origin settings in WebFOCUS, see the *WebFOCUS Security and Administration* content.

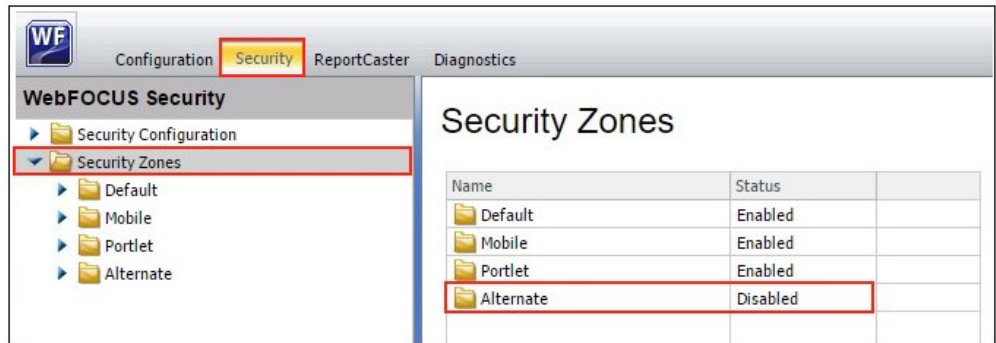
Procedure: How to Configure WebFOCUS

1. Sign in to WebFOCUS as an *Administrator*.
2. Click *Administration* in the top menu and then click *Administration Console* from the drop-down list, as shown in the following image.



The WebFOCUS Administration Console opens.

3. Click the *Security* tab and then the *Security Zones* folder node in the left pane, as shown in the following image.



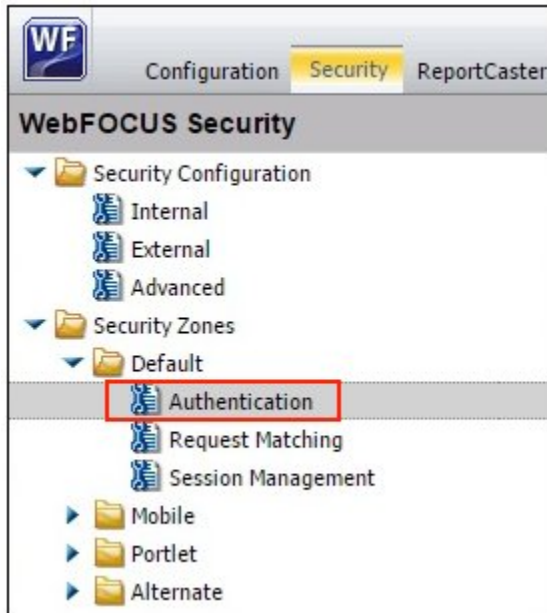
4. Verify whether the Alternate security zone (highlighted in the above image) is *Disabled* or *Enabled*.

There is no reason to enable the Alternate Zone to support WebFOCUS embedded BI, but this zone is sometimes enabled to provide an alternative way for administrators to access WebFOCUS.

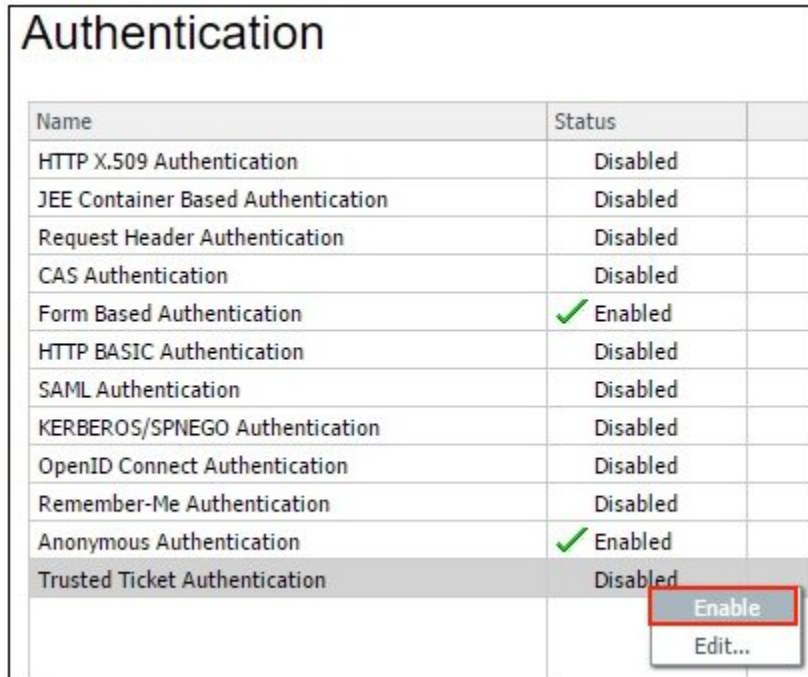
If the status of your Alternate security zone is *Disabled*, then skip to Step 5.

If the status of your Alternate security zone is *Enabled*, then you need to determine if the Alternate security zone will be used to process trusted ticket and trusted sign-on requests. If this is the case, then you need to enable Trusted Ticket Authentication on the Alternate security zone and ensure that it is configured properly to process these requests. The remaining configuration steps in this procedure are provided for the Default security zone. These steps can be applied to the Alternate Zone as well, if it is enabled, and processing requests related to the embedded BI demo application.

5. Expand *Security Zones, Default*, and then click *Authentication*, as shown in the following image.



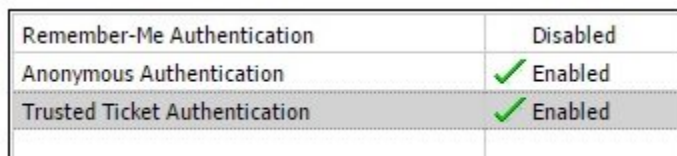
6. In the Authentication pane, right-click *Trusted Ticket Authentication* and select *Enable* from the context menu, as shown in the following image.



Name	Status	
HTTP X.509 Authentication	Disabled	
JEE Container Based Authentication	Disabled	
Request Header Authentication	Disabled	
CAS Authentication	Disabled	
Form Based Authentication	✓ Enabled	
HTTP BASIC Authentication	Disabled	
SAML Authentication	Disabled	
KERBEROS/SPNEGO Authentication	Disabled	
OpenID Connect Authentication	Disabled	
Remember-Me Authentication	Disabled	
Anonymous Authentication	✓ Enabled	
Trusted Ticket Authentication	Disabled	

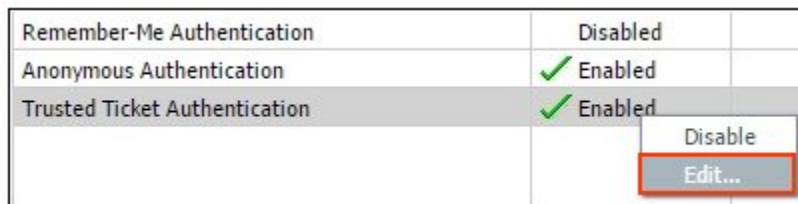
Enable
Edit...

The status of *Trusted Ticket Authentication* is now *Enabled*, as shown in the following image.



Remember-Me Authentication	Disabled	
Anonymous Authentication	✓ Enabled	
Trusted Ticket Authentication	✓ Enabled	

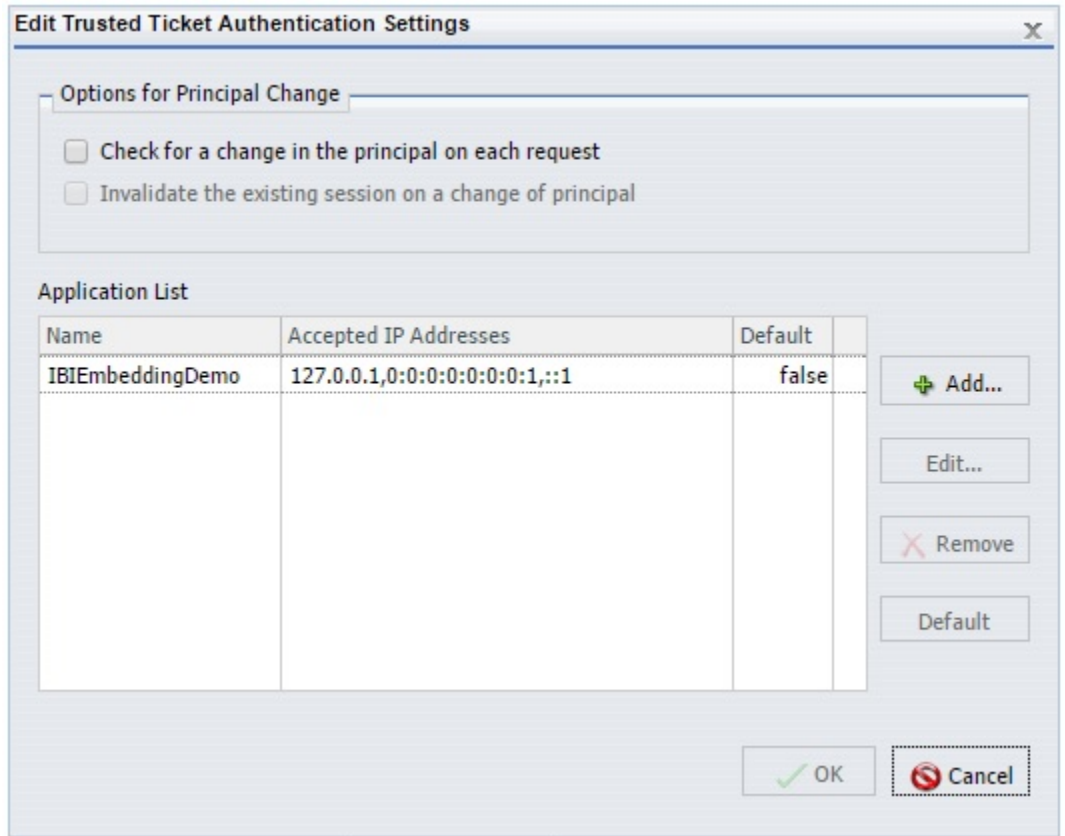
7. Right-click *Trusted Ticket Authentication* again and select *Edit* from the context menu, as shown in the following image.



Remember-Me Authentication	Disabled	
Anonymous Authentication	✓ Enabled	
Trusted Ticket Authentication	✓ Enabled	

Disable
Edit...

The Edit Trusted Ticket Authentication Settings dialog opens, as shown in the following image.



Notice that the trusted application name is *IBIEmbeddingDemo* and three versions of the localhost IP address (version 4 (IPv4), version 6 (IPv6), and IPv6 loopback) are defined in the Accepted IP Addresses list. These settings are used by WebFOCUS to determine if a trusted ticket request is originating from an authorized server.

8. If your *config.properties* file (as described in [Configuring a Back Channel Ticket Request](#) on page 444) has `localhost` specified as the value for the `WF_TICKET_PROVIDER_URL` setting and if `IBIEmbeddingDemo` is specified as the value for the `WF_TRUSTED_APPLICATION_NAME` setting, then you do not need to make any changes in the Edit Trusted Ticket Authentication Settings dialog.

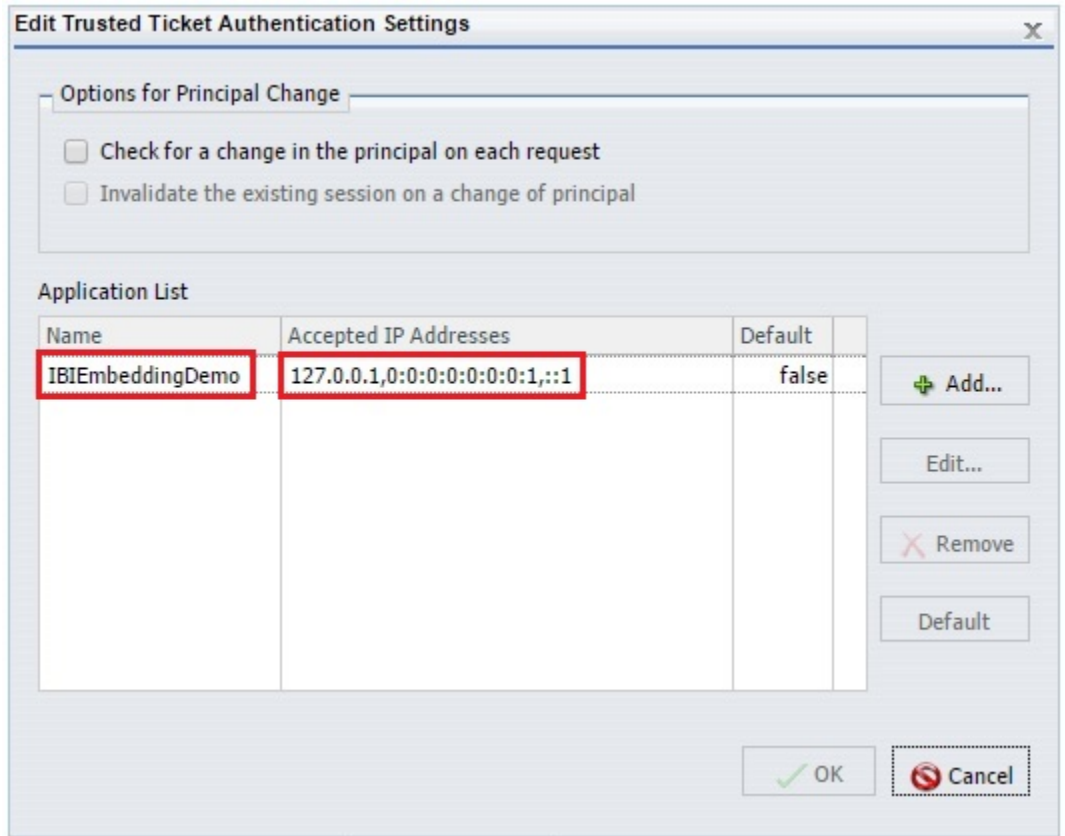
Click *Cancel* and skip to Step 10.

-
9. If you do not have `localhost` specified as the value for the `WF_TICKET_PROVIDER_URL` setting, then you must determine what IP address the embedded BI demo application will appear to be running on and add it to the Accepted IP Addresses list in the Edit Trusted Ticket Authentication Settings dialog.

Tip: If you are not sure what the IP address is, and the wrong value is configured, a message will be displayed after signing on to the embedded BI demo application (*Fintoso Financial*) indicating that the trusted ticket was not granted. To investigate, you can check the WebFOCUS *websecurity* log. If the log message indicates that the IP address of the requesting server was not in the Accepted IP Addresses list, then copy the IP address from the log message to the Accepted IP Addresses list in the Edit Trusted Ticket Authentication Settings dialog and try again.

10. Click *OK* to acknowledge the requirement to restart your application server (for example, Apache Tomcat).
11. Before closing the WebFOCUS Administration Console, double-click the Trusted Ticket Authentication provider you just enabled.

The Edit Trusted Ticket Authentication Settings dialog opens, as shown in the following image.



Notice that a trusted application called *IBIEmbeddingDemo* is pre-configured for you and that this matches the value in the *config.properties* file of the embedded BI demo application.

While this trusted application information is preconfigured for you in WebFOCUS, it is not used unless you enable it as described in this procedure.

12. Click *Cancel* and then close the WebFOCUS Administration Console.
13. Restart your application server (for example, Apache Tomcat).

WebFOCUS is now ready to accept trusted ticket connections.

Using the Trusted Ticket Test Pages

This section describes how to use the trusted ticket test pages by using valid arguments and then using an invalid argument to test the functionality of the page.

Procedure: How to Configure the Trusted Ticket Test Page

1. Run the Reporting Server on the jagsig VMWare image.
2. From the WebFOCUS Administration Console, click the Security tab and select the *Trusted Ticket Authentication* check box to ensure it is enabled for the Default Zone.

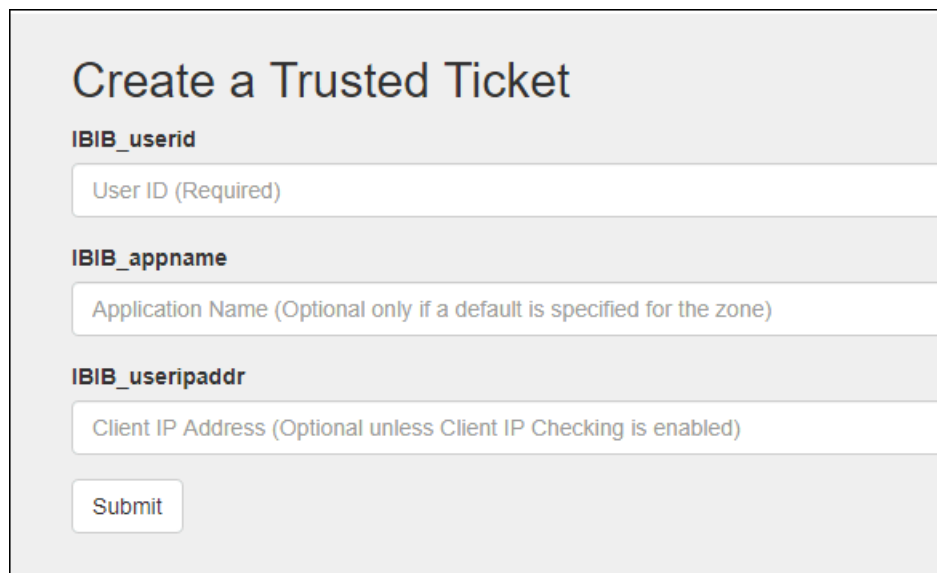
If you are using the Alternate Zone, make sure the option is also enabled there.

3. In a new browser tab, enter the following URL:

http://localhost:8080/embeddemo/tester/create_trusted_ticket.jsp

4. Enter the following values as shown in the table below:

Parameter	Value
UserID:	<i><use CTRL-V to paste it from the clipboard></i>
IBIB_appname:	<i>IBIEmbeddingDemo</i>
IBIB_destination:	<i><leave blank></i>



The screenshot shows a web form titled "Create a Trusted Ticket". It contains three input fields and a submit button. The first field is labeled "IBIB_userid" and contains the text "User ID (Required)". The second field is labeled "IBIB_appname" and contains the text "Application Name (Optional only if a default is specified for the zone)". The third field is labeled "IBIB_useripaddr" and contains the text "Client IP Address (Optional unless Client IP Checking is enabled)". A "Submit" button is located at the bottom left of the form.

A trusted ticket is returned and displayed in the browser.

5. Copy the ticket to the clipboard by pressing CTRL-C.
6. In your browser, enter the following URL:

http://localhost:8080/embeddemo/tester/test_trusted_ticket.jsp

7. Enter the following values as shown in the table below:

Parameter	Value
IBIB_ticket:	<i><use CTRL-V to paste it from the clipboard></i>
IBIB_appname:	<i>IBIEmbeddingDemo</i>
IBIB_destination:	<i><leave blank></i>

This returns a browser page that shows an XML response with an ibfssession, which means the sign on was successful.

Procedure: How to Test the Trusted Ticket Page Using an Invalid Argument

To test the trusted ticket page using an invalid argument:

1. While running the create_trusted_ticket.jsp, enter an invalid name into the IBIB_appname field, as shown in the following image.

Test a Trusted Ticket Signon Request

IBIB_ticket

Trusted Ticket Value (Required)

IBIB_appname

Application Name (Optional only if a default is specified for the zone)

IBIB_Destination

Redirect URL (Optionally specify a full or a /relative URL)

Your IP Address is 172.30.234.118.

The result, A "-1" will be returned.

2. In the next test, in `securitysettings.xml`, set `userIPAddrCheck` to `true` for the `TrustedTicketPreferences` section, and then recycle Apache Tomcat.
3. Enter a non-localhost IP address for `IBIB_useripaddr`.

Note:

- The ticket is returned as expected.
 - You can use CTRL-C to copy the ticket to the clipboard.
4. Run `test_trusted_ticket.jsp` with that ticket.

You will receive a `403 return` because it failed. The `websecurity.log` file will show the following message:

```
[2017-08-16 13:23:52,512] WARN [http-nio-8080-exec-8:wfsecurity]
:unknown: - [Zone: main]Trusted sign on request rejected, Key
verification failure
```

Using the Embedded Business Intelligence Demonstration Application (Fintoso Financial)

This section describes how to access and use the embedded business intelligence (BI) demo application (*Fintoso Financial*), and reviews the internal (back-end) functionality of the application.

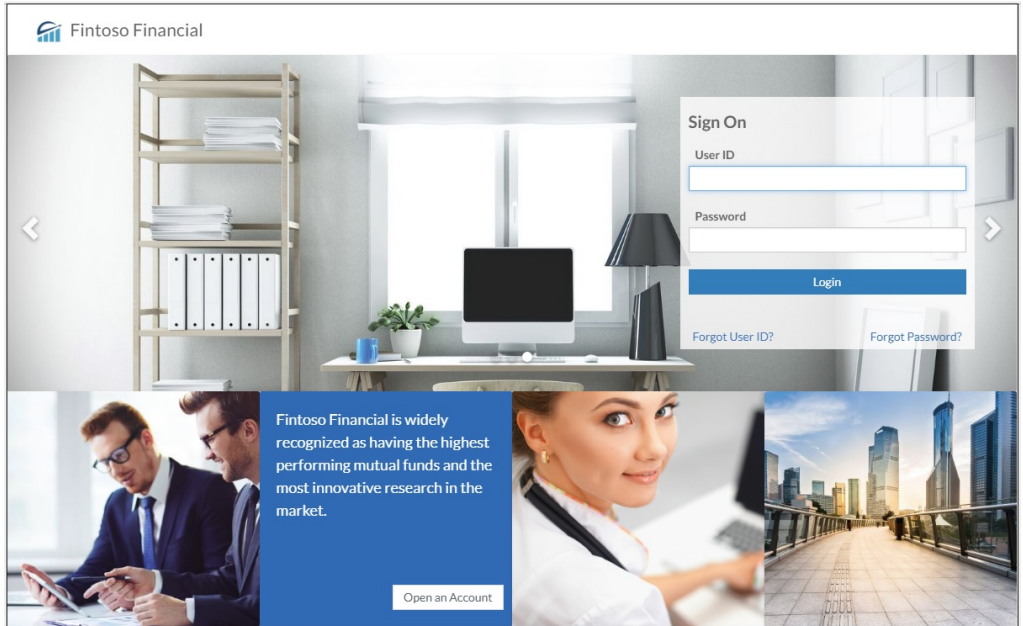
Accessing and Running the Embedded BI Demo Application

1. Open a web browser and enter the URL for the embedded BI demo application (*Fintoso Financial*).

You can access the demo application in many ways, depending on where your web browser is located. The following examples (with port numbers, where required) are supported:

- `http://localhost/embeddemo`
- `http://host/embeddemo`
- `http://host.domain.com/embeddemo`

You will be redirected to the Fintoso Financial sign in page, as shown in the following image, because you do not yet have a session established with the Fintoso Financial application.

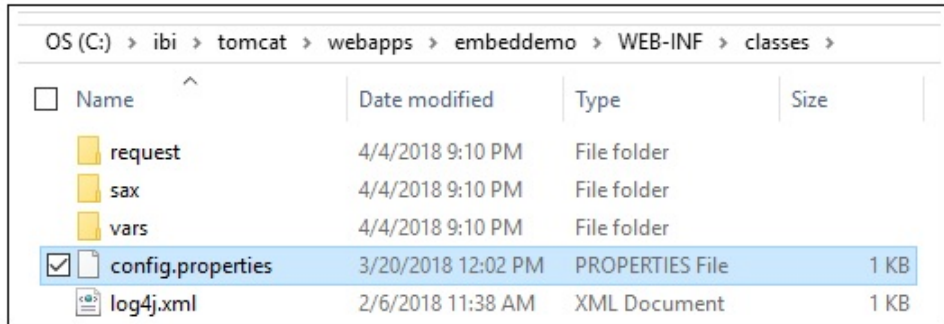


2. Enter a valid user name (for example, *ffadv*).

Note: This is not a WebFOCUS sign in and anything you enter in the Password field is ignored by the demo application. The user name that is entered is simply verified as being in the array defined in line 5 of the *login.jsp* file, which is located in the following folder:

`embeddemo\login.jsp`

For example, if you deployed the embedded BI demo application (*embeddemo.war*) file on the Apache Tomcat application server that is available with the WebFOCUS installation, then the *config.properties* file is located in the folder path, as shown in the following image.



You may add to or change the list of users in the *config.properties* file. For example:

```
WF_TICKET_PROVIDER_URL=http://localhost:8080/ibi_apps
WF_TRUSTED_APPLICATION_NAME=IBIEmbeddingDemo
WF_HOST=http://localhost:8080/ibi_apps
USERIDS=ffadv
```

Changes are applied immediately and an application server restart is not required.

The user name you enter must be a valid WebFOCUS user account with access to the BI Portal and web service content configured in the embedded BI demo application.

3. Click *Login*.

The application requests a trusted ticket for the user name you entered and then makes a WebFOCUS trusted sign-on request with this ticket in order to obtain a WebFOCUS session cookie for your web browser. For more information, see [Appendix: Detailed Request/Response Flow for the Embedded Business Intelligence Demonstration Application](#) on page 478.

The home page of the Fintoso Financial embedded BI demo application is displayed, as shown in the following image.

Fintoso Financial ffadv Sign Out

HOME MY INVESTMENTS RESEARCH

Current Forecast

Your current strategy is likely to produce retirement income that meets 80%-94% of your goal.

Estimated Household Income	\$53,100
Household Income Goal	\$64,600
Income Gap	-\$11,500

Me

Savings Rate	10%
Account Balance	\$187,379.26

Your Retirement Outlook® 82%
(percent of income goal achieved)

People My Age

Average Savings Rate	6%
Account Balance	\$33,402

Your Retirement Outlook® 114%
(percent of income goal achieved)

Analytics Workstation Premium

Use state of the art data, reporting and visualization tools to analyze our fund data and your account data.

[Launch](#) [Learn more](#)

Fund Analyzer New!

Our new FundAnalyzer allows you to analyze and compare hundreds of mutual funds and investment strategies.

[Launch](#) [Learn more](#)

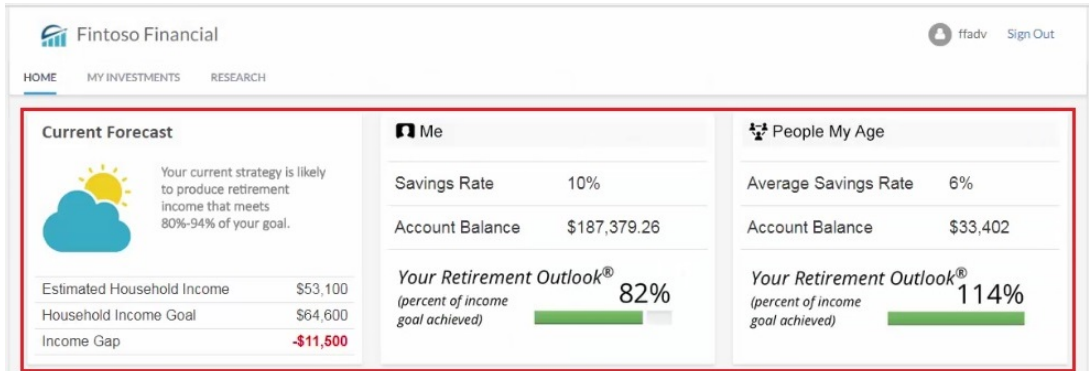
Go Paperless! Free!

Enroll in our innovative new Interactive eStatement product that provides what-if analysis on your investments.

[Enroll](#) [View statement](#)

The Fintoso Financial embedded BI demo application features responsive web design. Resizing the browser or viewing on mobile devices will automatically adjust the dimensions and layout of the application accordingly. This behavior is important to understand because it is the result of coordination between WebFOCUS and the embedded BI demo application. For more information see, [Responsive Web Design](#) on page 470.

Three account widgets are displayed across the top, as shown in the following image.

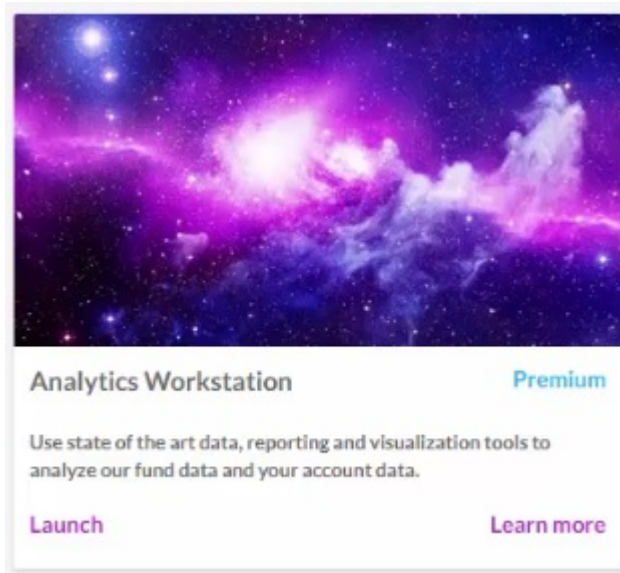


The middle and right-hand frames contain widgets that run WebFOCUS reports (Fund Analyzer and Go Paperless!), as shown in the following image.

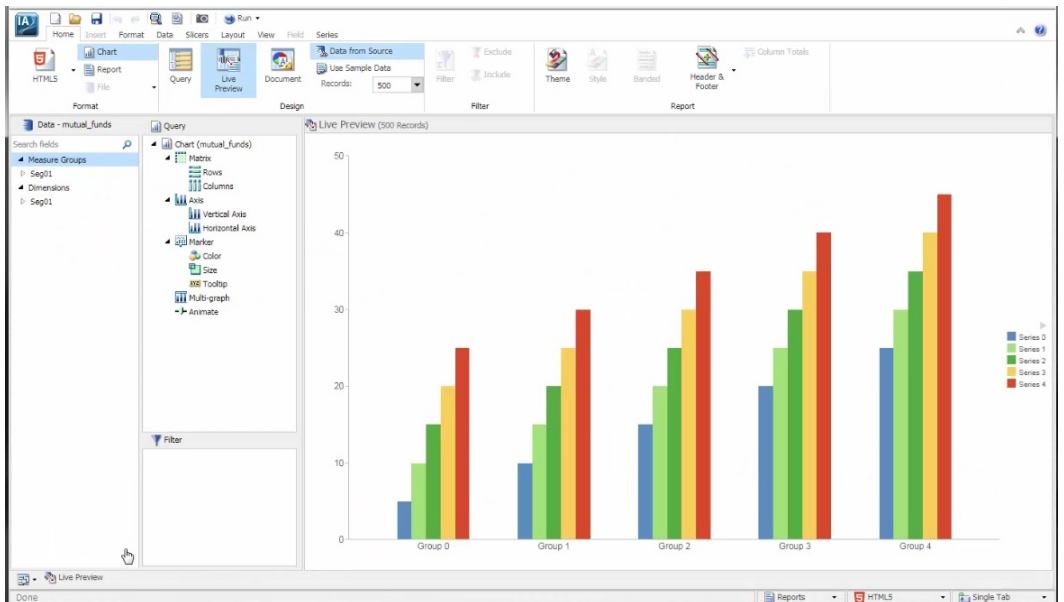
The image shows two promotional widgets. The 'Fund Analyzer' widget features a background of glowing light bulbs and includes a 'Launch' button and a 'Learn more' link. The 'Go Paperless!' widget features a background of a hand interacting with a futuristic interface and includes an 'Enroll' button and a 'View statement' link.

They are included by making a RESTful web service call to WebFOCUS, which is within the context of the user ID, and using a trusted ticket approach.

On the lower-left, the Analytics Workstation widget includes a *Launch* hyperlink, as shown in the following image.

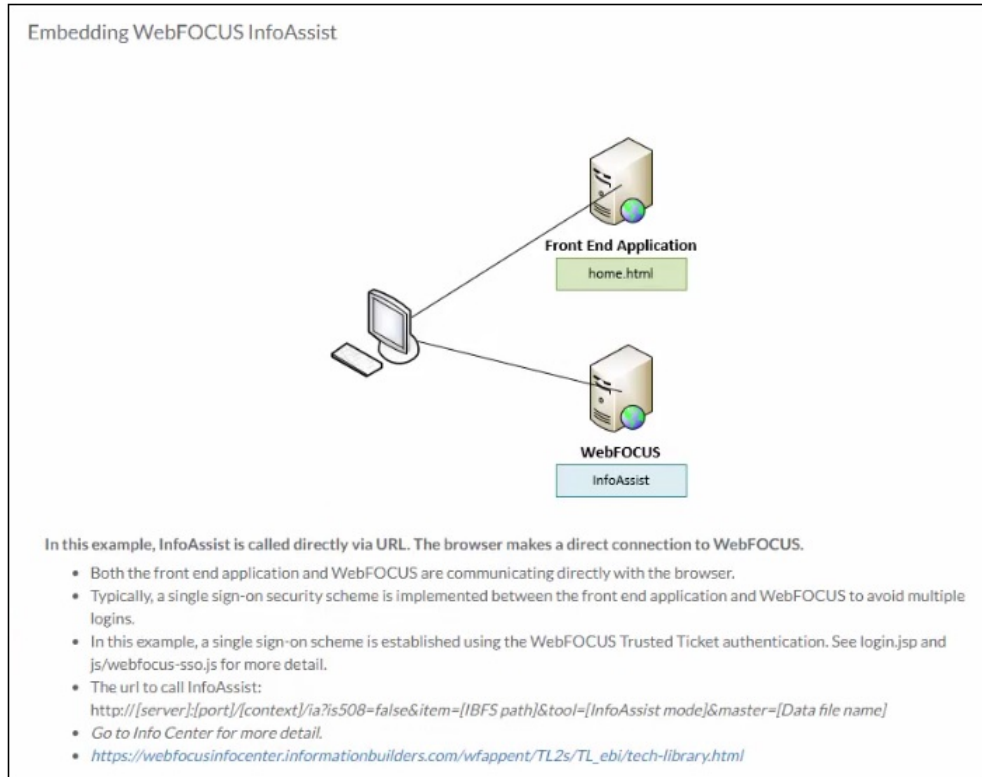


Clicking *Launch* opens WebFOCUS InfoAssist, as shown in the following image.



This is a good example of URL embedding.

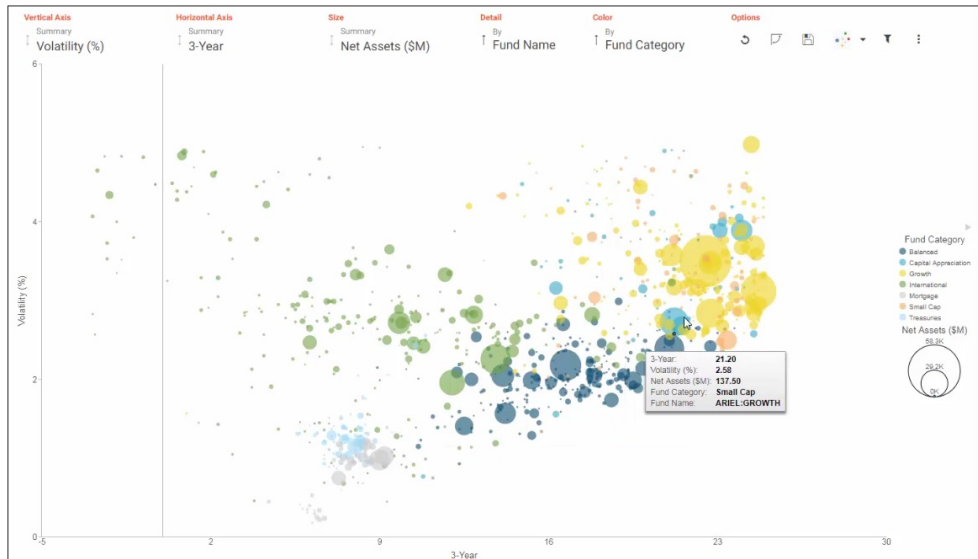
Clicking *Learn more* from the Analytics Workstation widget opens a pop-up window, which includes a diagram and provides a brief overview on URL embedding, as shown in the following image.



The Fund Analyzer widget also includes a *Launch* hyperlink, as shown in the following image.

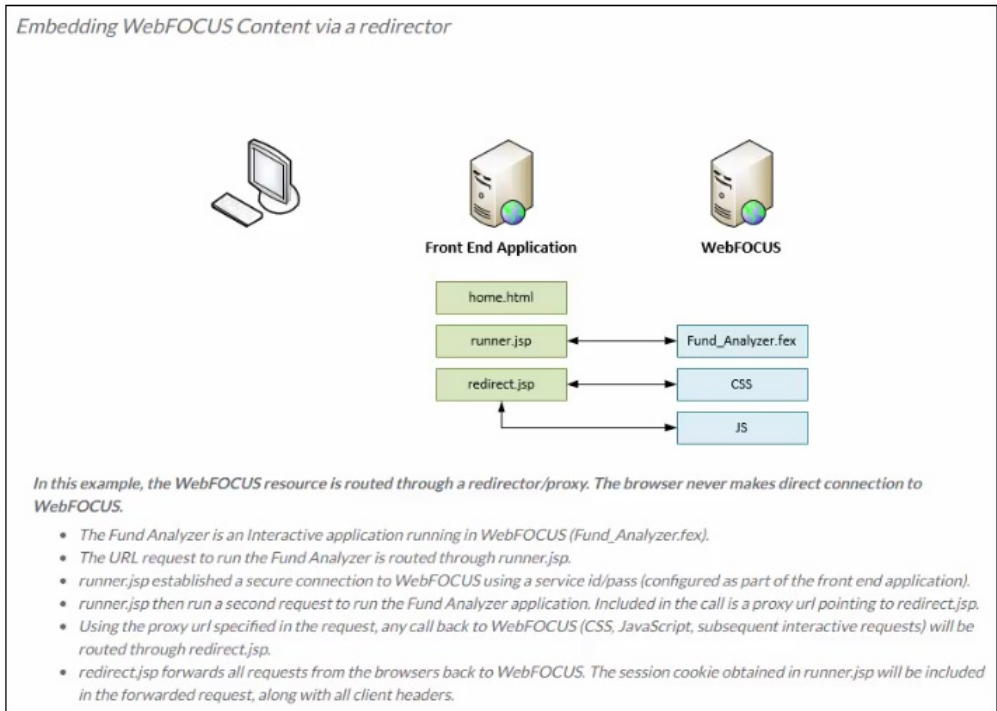


Clicking this link opens a WebFOCUS Insight report, as shown in the following image.



A Java proxy is being used here to call this report. This is a good example, as many WebFOCUS customers prefer all URL calls from their applications to initially go through a Java proxy before reaching WebFOCUS.

Clicking *Learn more* from the Fund Analyzer widget also opens a pop-up window, which includes a diagram and provides a brief overview on using a Java proxy to redirect URL calls being made to WebFOCUS, as shown in the following image.



On the lower-right, the Go Paperless widget includes a *View Statement* hyperlink, as shown in the following image.



Clicking this link opens a WebFOCUS In-Document Analytics (Active) report, as shown in the following image.

Accct Activity
Fund Activity
Allocations
Trends
Other Activity
Education
Contact Us

Information Builders
2 Penn Plaza
New York, NY 10121-2898

Allison Smith
444 Cedar Street
St. Paul, MN 55101

RETIREMENT ACCOUNT STATEMENT
July 01, 2017 - September 30, 2017

Your Account Number: RCMAJ1234567
Your Contribution Rates: Pre-tax 2.3%
After-tax 5.0%

Your Account Activity

TOTAL PORTFOLIO VALUE

Activity	Portfolio Value
Balance as of July 1, 2015	\$44,500
Your Contributions This Period	\$62
Employer Contributions This Period	\$62
Other Deposits	\$5
Total Contributions	\$125
Withdrawals	-\$2,310
Expenses	-\$125
Transfer:	-\$2,548.77
Investment Earnings	\$2,161.19
Balance as of September 30, 2015	\$46,953.17
Vested Balance	\$25,500.00
Total Change in Value	\$2,453.17
Number of Loans	1
Total Outstanding Loan Balance	\$8,327.91

ARE YOU READY FOR RETIREMENT?

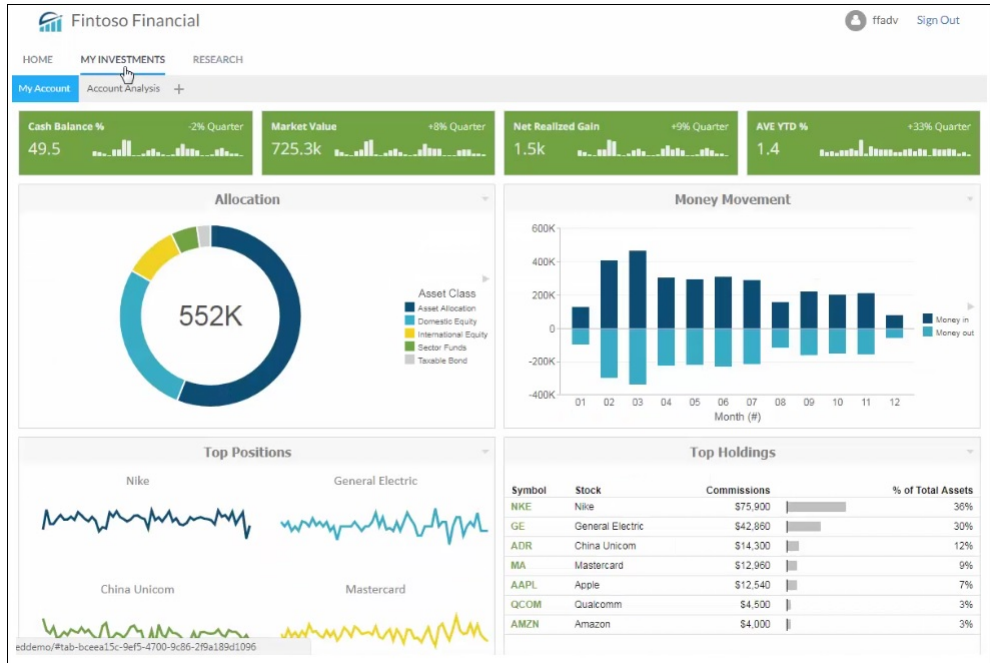
Current Age = 44 Retirement Age = 65 Contribution = 1 %

at Age 44, Projected Balance at Age 65, New Projected Balance at Age 65 BY Contribution Rate

■ Current Balance at Age 44
 ■ Projected Balance at Age 65
■ New Projected Balance at Age 65

This illustration is a projection based on your Ending Balance for this period and the assumptions found at www.informationbuilders.com. It is intended to give you a starting point for retirement planning discussions with your financial advisor. Your actual results may vary.

- Click the *MY INVESTMENTS* tab, which launches a redesigned Collaborative Portal, as shown in the following image.



5. Click the *RESEARCH* tab, which provides a good example showing how WebFOCUS Designer pages can be used in an application, as shown in the following image.

The screenshot displays the Fintoso Financial Mutual Fund Research Center. At the top, the Fintoso Financial logo is on the left, and a user profile icon with the name 'ffadv' and a 'Sign Out' link is on the right. Below the logo, navigation tabs for 'HOME', 'MY INVESTMENTS', and 'RESEARCH' are visible, with 'RESEARCH' being the active tab. The main heading is 'Mutual Fund Research Center'. The central area features a 'Mutual Fund Selections' section with filters for 'Fund Category', 'Fund Family', and 'Options', all set to 'All'. Below these are filters for 'Returns' (All, Low, Avg, High), 'Rating' (All, Low, Avg, High), and 'Expenses' (All, Low, Avg, High). A 'Risk' slider is positioned at 0, with a range from 0 to 4. A 'See Results' button is located at the bottom right of the filter section. To the right of the filters, a green box displays 'Mutual Fund Selections' with a large '646' and a table of metrics: Rating (5.63), Return (12.70), Expenses (1.24), and Risk (2.48). The bottom of the page is divided into four columns: 'Investment Advice' (with a gold medal icon), 'Innovative Research' (with a brain icon), 'Why choose' (with a hand icon), and 'Contact us' (with a headset icon, phone number 1-800-736-6130, and a 'Chat' link). The footer contains 'About Fintoso Financial' on the left and 'Follow us:' with social media icons for Facebook, Twitter, and YouTube on the right.

Specifically, two Designer pages are linked together through the Global Name feature.

Fund Category	Fund Name	Type	YTD	1-Year	3-Year	5-Year	Rating	Buy
All	AARP BAL STOCK & BOND		15.20	25.00	17.30	.	★★★★★	Buy
	AARP GNMA & US TREAS		4.50	7.50	7.30	5.40	★★★★★	Buy
	ACCESSOR.MORTGAGE SEC		5.50	9.30	8.60	6.30	★★★★★	Buy
	ACORN FUND		15.40	25.00	17.90	19.60	★★★★★	Buy
	ADVANCE CAP I BALANCED		13.20	23.40	19.30	.	★★★★★	Buy
	AETNA:AETNA FUND,SEL		14.40	23.50	16.20	12.80	★★★★★	Buy
	AIM EQ.WNGARTEN,RTL A		20.90	32.90	24.00	15.70	★★★★★	Buy
	AIM BALANCED FUND;A		16.30	27.60	22.20	17.70	★★★★★	Buy
	AIM BALANCED FUND;B R		15.70	26.50	21.20	.	★★★★★	Buy
	AIM VALUE A		18.60	31.70	22.20	20.60	★★★★★	Buy
	AIM VALUE B R		18.00	30.70	21.20	.	★★★★★	Buy
	ALLIANCE BALANCED;A		16.40	26.00	16.50	11.50	★★★★★	Buy
	ALLIANCE BD;US GOVT;A		4.10	7.00	6.40	5.20	★★★★★	Buy
	ALLIANCE BD;US GOVT;B R		3.60	6.20	5.60	4.50	★★★★★	Buy
	ALLIANCE BD;US GOVT;C R		3.60	6.20	5.70	.	★★★★★	Buy
	ALLIANCE FUND;A		21.60	37.90	23.80	19.30	★★★★★	Buy
	ALLIANCE MTGE INC;A		5.20	8.50	7.90	5.80	★★★★★	Buy
	ALLIANCE MTGE INC;B R		4.70	7.60	7.10	5.00	★★★★★	Buy

6. Click *Sign Out* in the upper-right corner of the application.

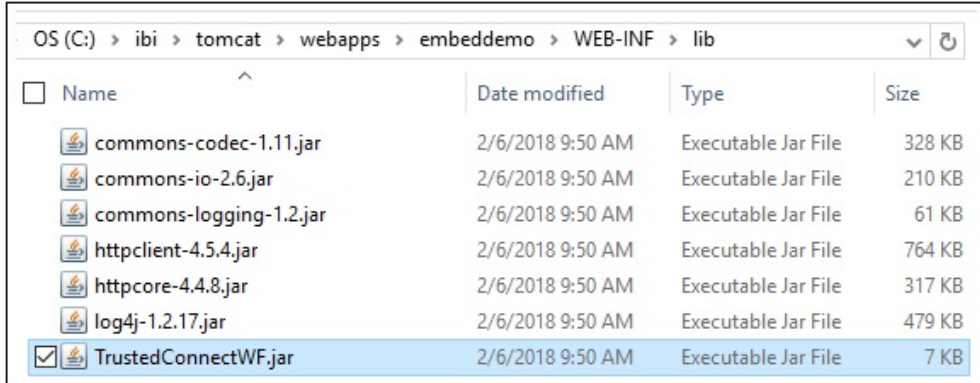
This signs you out of the Fintoso Financial embedded BI demo application.








Reviewing the Internal (Back-End) Functionality of the Embedded BI Demo Application

After authenticating the user on line 5 in the *login.jsp* file (*embeddemo/login.jsp*) the embedded BI demo application makes a request to obtain a trusted ticket for the user. This is done through a Java Bean that is defined on lines 11-14 in the *index.jsp* file (*embeddemo/index.jsp*) and executed on line 75.

Note: The Java Bean call is passed the authenticated user ID and the HTTP request object.

The trusted ticket request is made by the `generateTicket()` method. This method is located in the `TrustedConnectWF` class, which is located in the `TrustedConnectWF.jar` file (`embeddemo\WEB-INF\lib\TrustedConnectWF.jar`). For example:



Name	Date modified	Type	Size
 commons-codec-1.11.jar	2/6/2018 9:50 AM	Executable Jar File	328 KB
 commons-io-2.6.jar	2/6/2018 9:50 AM	Executable Jar File	210 KB
 commons-logging-1.2.jar	2/6/2018 9:50 AM	Executable Jar File	61 KB
 httpclient-4.5.4.jar	2/6/2018 9:50 AM	Executable Jar File	764 KB
 httpcore-4.4.8.jar	2/6/2018 9:50 AM	Executable Jar File	317 KB
 log4j-1.2.17.jar	2/6/2018 9:50 AM	Executable Jar File	479 KB
<input checked="" type="checkbox"/>  TrustedConnectWF.jar	2/6/2018 9:50 AM	Executable Jar File	7 KB

The *Fintoso Financial* sample web application is a realistic looking but simple demonstration of an actual embedded BI application. You can review the code in the `embeddemo` folder as well as the Java source for the `TrustedConnectWF` method inside the `IBITrustedTicket.jar` file (`embeddemo\WEB-INF\lib\IBITrustedTicket.jar`). To access the Java source code for this method, use a utility such as WinZip to open this .jar file and then extract the file from the `com\ibi\example` folder structure.

When you run the embedded BI demo application, you are running the `index.jsp` file (`embeddemo\index.jsp`), which builds the HTML view:

```
trustedTicket = encodeURIComponent('<%=TrustedConnectWF.generateTicket()  
%>');
```

This Java code uses the two properties in the `config.properties` file (`embeddemo\WEB-INF\classes\config.properties`) to determine where to make the ticket request and what trusted application name to provide on the call.

```
WF_TICKET_PROVIDER_URL=http://localhost/ibi_apps  
WF_TRUSTED_APPLICATION_NAME=IBIEmbeddingDemo
```

In addition, the user ID and web browser IP address are passed in the request to WebFOCUS. For more information about these HTTP requests/responses, see [Appendix: Detailed Request/Response Flow for the Embedded Business Intelligence Demonstration Application](#) on page 478.

Once the ticket is obtained successfully, the embedded BI demo application makes a trusted sign-on request to WebFOCUS using information defined in lines 6-8 in the *bip-page ext.js* file (*embeddemo\js\bip-page ext.js*):

```
var webfocusHost = '';  
var trustedAppName = 'IBIEmbeddingDemo';  
var webfocusContext = '/ibi_apps';
```

In a cross-origin (dual web host) configuration, you must edit the `webfocusHost` setting according to your environment.

Additional Considerations for Embedded Business Intelligence

To provide the best user experience in your embedded BI application you should also consider the following topics.

Hiding BI Portal Features

Generally you want to hide the BI Portal banner, since the embedding application usually has its own banner. If you only want to embed a single BI Portal page you can also choose to disable the BI Portal Navigation bar from the ribbon in Portal Designer. If you have a multi-page portal you should leave the navigation bar enabled and consider styling the portal page tabs with a custom CSS file, as explained below. When the navigation bar is shown you also have a choice to show or hide the New Page icon to users who have the Customize Portal privilege.

Branding and Rebranding

Generally speaking you want the embedded BI application and the content it is hosting to appear like a single, well styled, application. WebFOCUS has excellent rebranding capabilities to address this requirement. There are two aspects to styling that you need to consider:

- Styling the content (for example, reports, charts, and so on) rendered by the portal.
- Styling the portal *chrome* that appears around the content.

You can style the content to match your embedding requirements by developing a custom WebFOCUS stylesheet and selecting it from the Theme button in the InfoAssist ribbon.

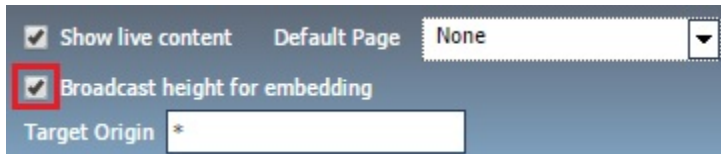
You can also use the BI Portal Dynamic Styling option to specify a custom stylesheet override to all the portal content. The embedded BI demo application uses this approach, as shown in the following image.



You can also create a custom CSS theme to style things like the panel borders, title bar text, portal page tab appearance, and even images like the easy selector Add Content or New Page icons. Store your custom CSS theme in the repository and select it from the Portal Theme Files UI, which is accessible by clicking the Theme button on the ribbon in Portal Designer.

Responsive Web Design

If you plan to embed responsive BI Portal content in your application you need to enable the *Broadcast height for embedding* option, on the portal properties panel in Portal Designer, as shown in the following image.



This option causes the portal to broadcast JavaScript events that return the height (in pixels) that the hosting iframe should be reset to in order for the portal content to fit without extra space or an inner scrollbar.

If the portal is only used in iframe embedding scenarios you can leave the Target Origin set to *. If the portal will be used in multiple scenarios and you want to limit the broadcast messages to only a single application you can set Target Origin to a specific host. For example:

<http://embeddinghost.domain.com>

You then need to modify your embedding application to listen for the following two events:

- portal_loaded.** This event is broadcast once the portal page has loaded.
- height_changed.** This event is broadcast each time the portal page height changes.

The following is an example which you will find in the embedded BI demo application on line 818 in `embeddemo/js/bip-page-ext.js`:

```

/* Add Message Listener */
window.addEventListener('message', function(e) {

var data;

if (typeof e.data === 'string')
data = JSON.parse(e.data);
else
data = e.data;

var pageType = typeof data.portal_path !== 'undefined' ? 'portal' : 'page';
console.log(data, pageType, data.page_path);
var portalNode = pageType == 'portal' ? getPortalNode(data.portal_path,
'path') : getPortalNode(data.page_path, 'path');

if (data) {
var message_name = data["message_name"];

if (message_name == "height_changed"){
if (pageType == "portal" && portalNode.loaded)
$('#'+portalNode.id).parent().height( parseInt(data["portal_height"]) );
else
$('#'+portalNode.id).parent().height( parseInt(data["page_height"]) );
}
else if (message_name == "portal_loaded"){
portalNode.loaded = true;
$('#'+portalNode.id).parent().height( parseInt(data["portal_height"]) );
}
}
window.scrollTo(0,0);
});
}
}

```

Alternate Security Zone

In general, there is no requirement to enable the Alternate security zone to support embedded BI deployments. If it is not required, then the Alternate zone should remain disabled because it does complicate troubleshooting of trusted ticket authentication configurations. The guidelines in this section are provided to assist users that will need to enable the Alternate zone while supporting trusted ticket authentication.

When enabled, WebFOCUS will first determine if a request should be processed by the Alternate zone configuration. By default, the Alternate zone is configured to capture requests made to `127.0.0.1`, `0:0:0:0:0:0:1`, and `::1`. As a result, if you are testing a same origin (single web host) configuration, then your trusted ticket request may be processed by the Alternate zone. In this case, you must enable Trusted Ticket Authentication on the Alternate zone in addition to on your Default zone.

You can enable and configure the Alternate zone to process trusted ticket requests in cross-origin (dual web host) configurations, but there is no requirement to do so. If you want to configure this, simply enable Trusted Ticket Authentication on the Alternate zone and add the IP address of the host where your embedded BI application resides.

Customizing the Embedded Business Intelligence Demonstration Application (Fintoso Financial)

This section describes how to customize the embedded demo application (*Fintoso Financial*).

Registering User Names

The embedded BI demo application makes trusted authentication requests to WebFOCUS. In a typical use case scenario, the application would authenticate users to some system that is external to WebFOCUS. However, to simplify the demonstration, this embedded application only checks to see if the user name entered on the Sign-on page is found in a list defined on line 5 in `embeddemo\login.jsp`:

```
Set<String> users = new HashSet<String>(Arrays.asList( userids ));
```

You may edit the list to include any ID that exists in your WebFOCUS repository. Changes take effect immediately upon reloading the application and do not require an application server restart.

Important: Only include WebFOCUS user IDs in this file that have carefully defined access to content and features. This is because anyone with access to the embedded BI demo application will be able to obtain a WebFOCUS session for these accounts without knowledge of the password for the account.

Using Different BI Portal Content

The HOME, MY INVESTMENTS, and RESEARCH tabs in the embedded BI demo application load specially configured BI portals into an iframe below the tabs. You can reconfigure these tabs to load different BI portals.



Simply modify the path and corresponding URL values in the `\bi\tomcat\webapps\embeddemo\js\webfocus-sso.js` file. For example:


```
// Array containing the list of pages
var pages = [
  { text: 'Home', type: 'page', url: 'home.html', loaded: false },
  { text: 'My Investments', type: 'url', path: 'IBFS:/WFC/Repository/Fintoso/
Fintoso_Financial.prtl', url: '/portal/Fintoso/Fintoso_Financial', loaded:
false },
  { text: 'Research', type: 'url', path: 'IBFS:/WFC/Repository/Fintoso/
Page_Designer_content/selections/page.man', url: '/rs/ibfs/WFC/Repository/
Fintoso/Page_Designer_content/selections', loaded: false }
];
```

Troubleshooting

This section provides troubleshooting information for the embedded business intelligence (BI) demo application (*Fintoso Finacial*) and workarounds where applicable.

If you require additional support or assistance with the embedded BI demo application, open a support ticket on the Information Builders Technical Support Center:

<http://techsupport.informationbuilders.com>

Pop-up Message: Failed to Obtain a Trusted Ticket From WebFOCUS

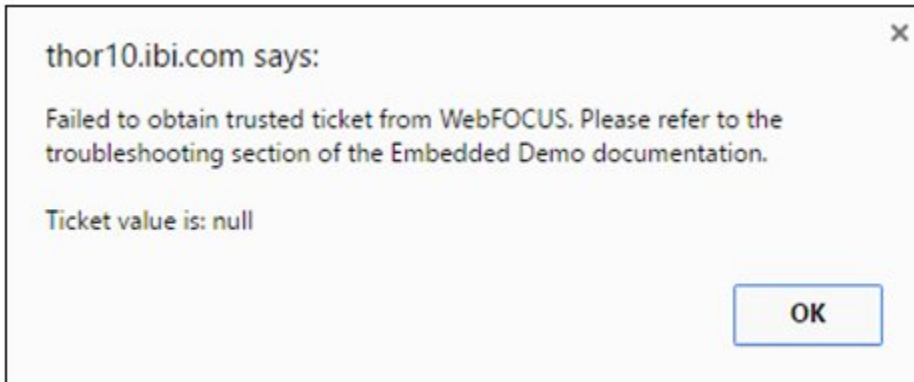
After signing on to the embedded BI demo application (*Fintoso Finacial*), you may encounter a pop-up message indicating that the application was unable to obtain a trusted ticket from WebFOCUS. If this occurs, you must resolve the issue before continuing because the demo application will not make the trusted sign-on request to WebFOCUS. This section describes several suggestions and workarounds to resolve the issue based on the information in the message.

The demo application checks for the result of the trusted ticket request and determines if it appears to be a ticket. This is done with a simple check in the `trustedWFSignOn()` function on line 21 in `embeddemo\js\bip-page-ext.js` to see if the result is longer than 40 characters. This indicates that a ticket was returned rather than a -1 status code, a null value, or other non-ticket response. For example:

```
function trustedWFSignOn(){
  // if we do not get a trusted ticket back from the TrustedConnectWF
  bean call in index.jsp, popup a message and do not attempt the trusted
  signon
  if (trustedTicket.length < 40) {
    alert('Failed to obtain trusted ticket from WebFOCUS. Please refer
to the troubleshooting section of the Embedded Demo documentation.\n
\nTicket value is: ' + trustedTicket);
    return;
  }
}
```

Ticket Value is: null

A *null* value indicates that the trusted ticket request was not processed by WebFOCUS. For example:

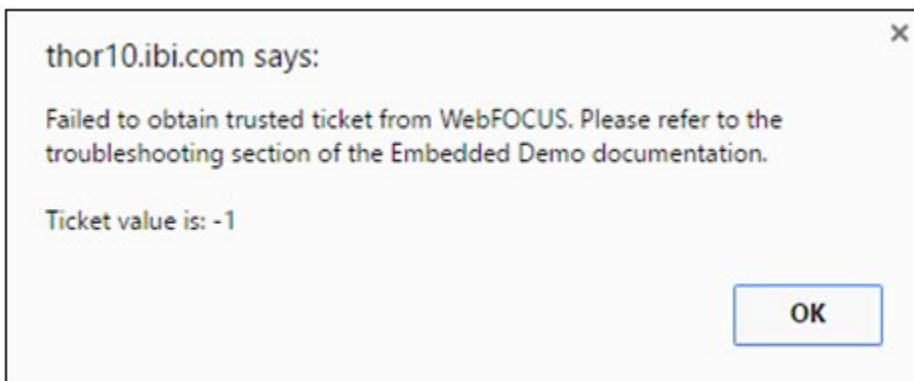


Check to ensure that you enabled trusted ticket authentication on the Default zone (and on the Alternate zone, if enabled).



Ticket Value is: -1

A *-1* value indicates that the trusted ticket request was processed, but WebFOCUS refused to create the ticket. For example:



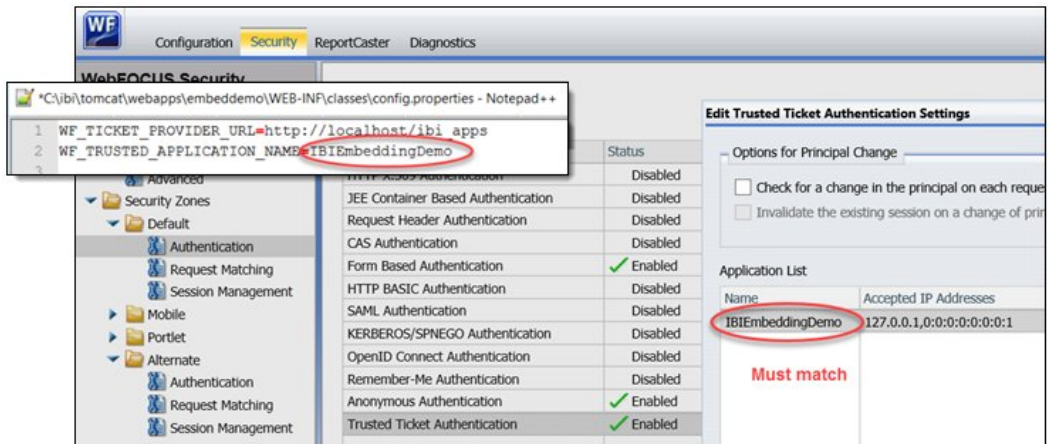
This can result from various reasons and the *websecurity.date.log* file will have additional information to help determine the cause. The *websecurity.date.log* file is located in the following folder in your WebFOCUS installation:

```
<drive>:\ibi\WebFOCUS82\logs
```

The following message indicates that the trusted application name sent by the embedded BI demo application does not match the value found in the WebFOCUS trusted ticket authentication configuration:

```
WARN [http-nio-80-exec-1:wfsecurity] :unknown: - [Zone: main]Invalid
application name: 'IBIEmbeddingDmo'
```

Check the value of the `WF_TRUSTED_APPLICATION_NAME` setting in the *config.properties* file of the embedded BI demo application and ensure that it matches the value of the Trusted Ticket Authentication zone configuration in the WebFOCUS Administration Console (Security tab), as shown in the following image.

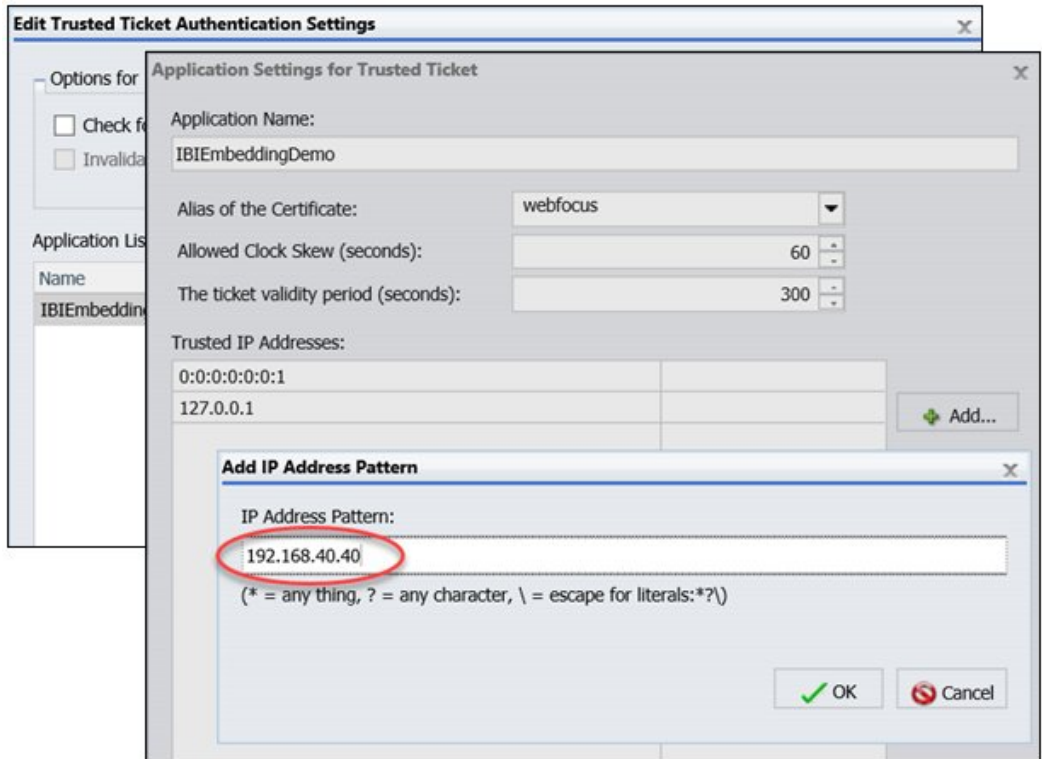


The following message indicates that the IP address of the host making the trusted ticket request does not match the IP address in the WebFOCUS Trusted Ticket Authentication zone configuration:

```
WARN [http-nio-80-exec-9:wfsecurity] :unknown: - [Zone: main] Trusted
ticket request rejected, the host IP address '192.168.40.40' is not in the
accepted host list.
```

This might be the case in a cross-origin (dual web host) configuration where you forgot to add the host IP of the embedded BI demo application to the WebFOCUS configuration.

Add the IP address shown in the log to the Trusted Ticket Authentication zone configuration in the WebFOCUS Administration Console (Security tab), as shown in the following image.



After you make this change, restart the application server where WebFOCUS is deployed.

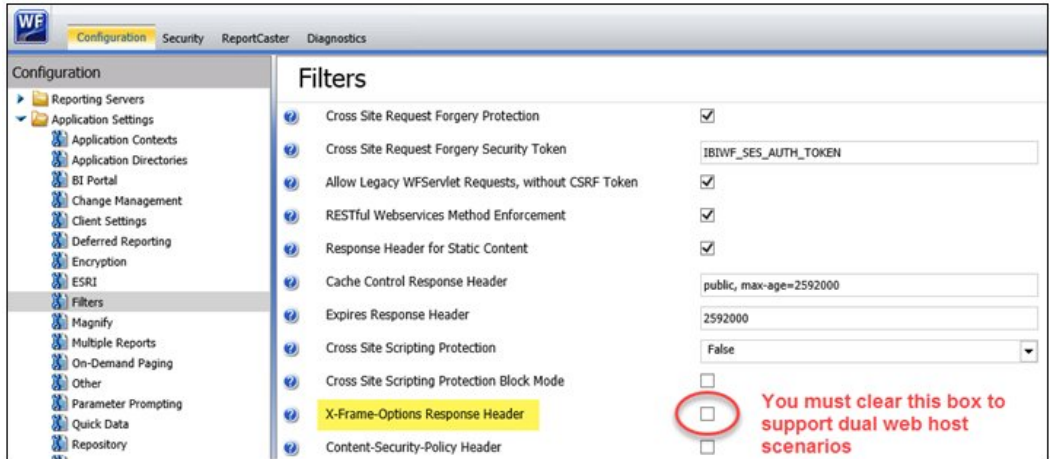
BI Portal Tabs Display an Error or are Blank

In cross-origin configurations, if you forget to disable the *X-Frame-Options Response Header* setting in the WebFOCUS Administration Console, the trusted ticket request and trusted sign-on calls may be successful, but the browser will refuse to embed the BI Portal in the iframe of the embedded BI demo application (*Fintoso Financial*).

Google Chrome browsers will simply leave the iframe blank. However, if you press *F12*, which opens the Developer Console, you will see that the error is caused by WebFOCUS sending a *SAMEORIGIN* requirement to the browser in the *X-Frame-Options* header.

Internet Explorer provides a clearer error message and no errors in the Developer Console (*F12*).

To resolve this issue, deselect the *X-Frame-Options Response Header* check box in the Application Settings, Filters area of the WebFOCUS Administration Console and click Save, as shown in the following image.

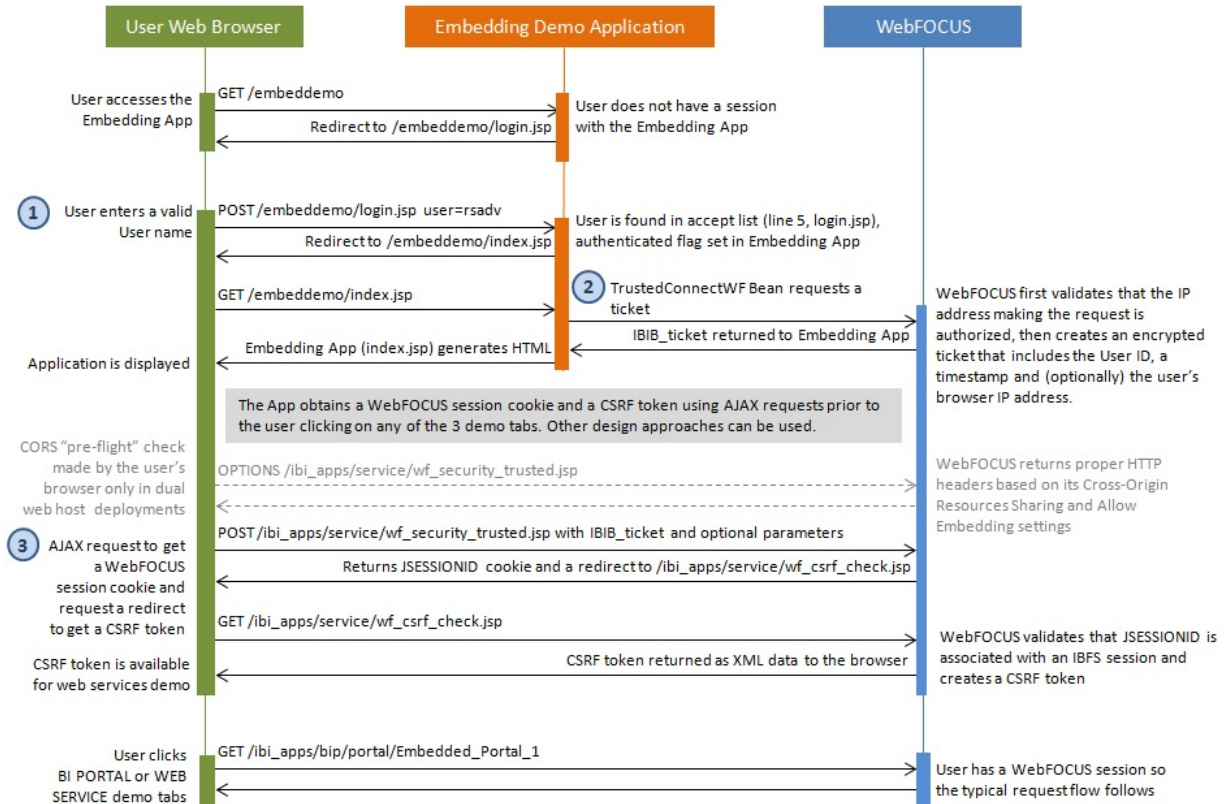


This action does not require an application server restart.

After making this change, reload/refresh the embedded BI demo application (*Fintoso Financial*) in your browser.

Appendix: Detailed Request/Response Flow for the Embedded Business Intelligence Demonstration Application

The following diagram illustrates the request/response flow for the embedded business intelligence (BI) demo application (*Fintoso Financial*), which also serves as a useful reference.





Feedback

Customer success is our top priority. Connect with us today!

Information Builders Technical Content Management team is comprised of many talented individuals who work together to design and deliver quality technical documentation products. Your feedback supports our ongoing efforts!

You can also preview new innovations to get an early look at new content products and services. Your participation helps us create great experiences for every customer.

To send us feedback or make a connection, contact Sarah Buccellato, Technical Editor, Technical Content Management at Sarah_Buccellato@ibi.com.

To request permission to repurpose copyrighted material, please contact Frances Gambino, Vice President, Technical Content Management at Frances_Gambino@ibi.com.



WebFOCUS

WebFOCUS Embedded Business Intelligence User's Guide
Release 8.2 Version 03



DN4501684.0518

**Information
Builders**

Information Builders, Inc.
Two Penn Plaza
New York, NY 10121-2898