Dynamic authentication for connectors in WebSphere Cast Iron, Part 1: Authenticating SAP and HTTP connectors dynamically

Madhavi Kota (madhkota@in.ibm.com)
Staff Software Engineer
IBM

Aparna Srinivasan (apasrini@in.ibm.com)
Software Engineer
IBM

15 October 2014

This article series shows you how to enable dynamic authentication connectors in WebSphere Cast Iron Studio. Part 1 shows you how to dynamically provide the connection parameters and connection pool options for authentication in WebSphere Cast Iron for SAP and HTTP connectors.

Introduction

IBM® WebSphere® Cast Iron provides a platform for integrating cloud-based applications from leading Software as a Service (SaaS) providers with on-premise applications from IBM and other companies. Cast Iron Studio is a development tool that you can use to design, test, and publish integration projects to a Cast Iron Integration Appliance. Cast Iron Studio provides numerous entities that you can drag into a workspace and configure as part of one or more business-process orchestrations that make up an integration project.

Dynamic authentication lets you connect to different endpoints or change endpoint configurations during runtime without stopping the orchestration. For example, you can change the user name and password dynamically. You enable this feature by adding optional parameters to the connection properties. If you do not provide a value for an element in the optional parameters of the connection properties, then no exception will be thrown during run time, because the value will be obtained from the Endpoint Configuration panel, which is mandatory. This feature lets you connect to different endpoint servers for different activities in an orchestration.

Objectives of this article

This article shows you how to dynamically authenticate various connectors in WebSphere Cast Iron Studio, including SAP, Domino, and others. The article also describes how to dynamically pass the values of the authentication in WebSphere Cast Iron Appliance and Cast Iron Live.
Prerequisites

To benefit from this article, you should be familiar with WebSphere Cast Iron Studio and WebSphere Cast Iron Appliance, including knowledge of the terminology and the different connectors used with Cast Iron Studio. Also, reviewing the developerWorks article Static and dynamic ways of providing input to connector activities using WebSphere Cast Iron Studio will help you understand this article.

System requirements

Configuring and deploying the module requires the following products:

1. WebSphere Cast Iron Studio V7.0.0.1 or later
2. WebSphere Cast Iron Appliance V 7.0.0.1 or later

Dynamic authentication for SAP connectors

SAP connectors support dynamic authentication for various authentication mechanisms to connect to different SAP servers. You enable this feature by adding optional parameters to sapConnectionProperties in the Map Inputs section of the activity. If you want to dynamically connect to an SAP server other than the one configured in the endpoint panel, or connect to the same SAP server with a different user name and password, or use an SSO token different from the one added in the endpoint panel, you can do so using dynamic authentication without stopping the project.

In Cast Iron, you can connect to SAP through the SAP Connector in the following ways:

- Dynamic authentication using Basic Authentication for SAP Connector
- Dynamic authentication using SSO Token for SAP Connector
- Dynamic authentication using Secure Network Connection (SNC) for SAP Connector.

Complete the following steps to create Dynamic Authentication for SAP connector:

1. Create a new project in Cast Iron Studio and create an endpoint for the SAP connector. Provide the following values of the SAP server in the created SAP endpoint: Host Name, System Number, SAP Client, RFC Program ID, Service, and user name and password of the SSO Token or SNC (Name, Partner, QOP, X509 Certificate). Create the configuration properties for each if needed.
2. Verify the test connection.
3. In the orchestration workspace, create an FTP endpoint using the XML file with input values (user name and password of the SAP server) that is already present. See Figure 1 below.
4. Drag and drop the Poll Directory activity of the FTP connector and then add the Read XML activity to read the XML input.
5. Drag and drop one of the SAP activities, such as the Invoke RFC activity. For more information on how to provide input to connector activities dynamically using WebSphere Cast Iron Studio, see Static and dynamic ways of providing input to connector activities using WebSphere Cast Iron Studio.
6. In the Configure panel of the activity, select the `BAPI_BANK_GETDETAIL` remote function call, as an example. In the Map Input section, provide the value of the BANK Country and BANK key value.

7. Right-click `BAPI BANK GETDETAIL` in the To Activity of the Map Input section and select `Show Optional Parameters`, as shown in Figure 2:

**Figure 2. Optional parameters option for SAP connector**

8. Once selected, the optional parameters are displayed. `SAPConnectionProperties` lists parameters for dynamic authentication along with the connection pool options, as shown in Figures 3 and 4:
9. Specify the Host Name, Client, System Number, Language, Basic Authentication, SSO Authentication, SNC Authentication, and connection pool options in the SAP connection properties:

**Figure 3. SAP connection properties option**

![Figure 3. SAP connection properties option](image)

10. Dynamically pass the values of User Name and Password (Basic Authentication) as obtained from the XML file from the FTP location, as shown in Figure 5. You can change the values of User Name and Password dynamically when the project is in running state:

**Figure 4. SAP connection properties (expanded view)**

![Figure 4. SAP connection properties (expanded view)](image)
11. Dynamically pass the values of the SSO token as obtained from the XML file from the FTP location, as shown below in Figure 6. The SSO token can be changed dynamically when the project is running in the appliance:

**Figure 6. Dynamically passing SSO authentication value for SAP connector**

12. Dynamically pass the SNC credentials – Name, Partner Name, QOP, and X509 Certificate -- as obtained from the XML file from the FTP location, as shown in Figure 7:

**Figure 7. Dynamically passing SNC authentication values for SAP connector**
13. You can also specify the connection pool options dynamically, as shown in Figure 8. The values for maximumConnections and maxidle Time can be changed dynamically when the project is running in the appliance:

**Figure 8. Dynamically passing connection pool options for SAP connector**

![Figure 8. Dynamically passing connection pool options for SAP connector](image)

Complete the entire orchestration with FTP Poll Directory, Read XML, and the SAP activity configured and well connected. Then deploy the project in the appliance. You can then specify the values dynamically in the FTP server, so that the Poll Directory activity picks up the values of the connection parameter dynamically and provides the connection parameters to the SAP activity.

**Dynamic authentication for HTTP connector**

HTTP connectors support dynamic authentication for various authentication mechanisms to connect to different HTTP servers. You enable this feature by adding optional parameters to httpConnParameter in the Map Inputs section of the activity. If you want to dynamically connect to an HTTP server other than the one configured in the endpoint panel, or connect to the same HTTP server with a different user name and password, or use OAUTH details different from those specified in the endpoint panel, you can do so using dynamic authentication without stopping the project.

In Cast Iron, you can connect to the HTTP server through the HTTP connector in the following ways:

- Dynamic authentication by passing the host name, port, user name, and password details in the Map Inputs section of the of HTTP server for HTTP connector.
- Dynamic authentication by passing the host name, port, and OAUTH details for HTTP connector.

In Cast Iron, to use dynamic authentication in the HTTP connector, you must specify the host name and the port details in the endpoint panel. Here are the steps to create an orchestration with the dynamic authentication feature for HTTP connector:

1. Create a new project in Cast Iron Studio and create an endpoint for HTTP connector.
2. Provide the values of the HTTP server in the HTTP endpoint.
3. Create configuration properties for each, if needed.
4. Verify the test connection.
5. In the orchestration workspace, create HTTP and FTP endpoints, using the XML file with input values (host name, port, user name, and password of the HTTP server) that is already present in the FTP location. See Figure 9 below.

6. Drag and drop the Poll Directory activity of the FTP connector, and then add the Read XML activity to read the XML input.

7. Drag and drop one of the HTTP connector activities, such as the Invoke Request activity. For more information on how to provide input to connector activities dynamically using WebSphere Cast Iron Studio, see Static and dynamic ways of providing input to connector activities using WebSphere Cast Iron Studio.

**Figure 9. Orchestration for dynamic authentication for HTTP connector**

8. In the Configure panel, under the Request section, specify the URL in the URL field.

9. On the Map Inputs tab in the checklist, expand httpConnParameter, so that you can see the parameters for dynamic authentication, as shown in Figure 10:
10. Map the values for host, port, isSecure, user name, and password obtained from the XML file from the FTP location, as shown in Figure 11. If you change the connection parameter in the XML file in the FTP location that is mapped as optional parameters, then the project in running state takes on the new connection parameter:

Figure 11. Dynamically passing basic authentication values for HTTP connector

11. Similarly, you can dynamically pass the values of the proxy server (to improve HTTP server performance) as obtained from the XML file in the FTP location when the project is in running state, as shown in Figure 12:
12. You can also dynamically pass the values of OAUTH credentials as obtained from the XML file in the FTP location when the project is in running state, as shown in Figure 13:

**Figure 13. Dynamically passing OAuth authentication values for HTTP connector**

Complete the entire orchestration with the FTP Poll Directory, Read XML, and HTTP activities configured and well connected. Then deploy the project in the appliance. You can then specify the values dynamically in the FTP server, so that the Poll Directory activity picks up the values of the connection parameter dynamically and provides the connection parameters to the HTTP activity.

**Conclusion**

Part 1 of this article series showed you how to dynamically provide the connection parameters and connection pool options for authentication in WebSphere Cast Iron Appliance for SAP and HTTP Connectors. Part 2 will show you how to dynamically authenticate connectors for Bulk API, JDE, and Salesforce.com.

**Acknowledgements**

The authors would like to thank IBM Software Developers C. Vamsi and Roshini Sunil for their help in reviewing this article and providing valuable input.
Resources

- **WebSphere Cast Iron resources**
  - WebSphere Cast Iron information center
  - Static and dynamic ways of providing input to connector activities using IBM WebSphere Cast Iron Studio
    This developerWorks article
  - WebSphere Cast Iron Studio information center
    A single portal to all WebSphere Cast Iron Cloud Integration documentation, with conceptual, task, and reference information on installing, configuring, and using WebSphere Partner Gateway.
  - WebSphere DataPower Cast Iron Management API guide
    Information on the Management API for the Cloud and Integration Appliances.
  - WebSphere Cast Iron Cloud Integration product library
    Product announcements, case studies, white papers, and more.
  - WebSphere Cast Iron Cloud Integration product page
    Product descriptions, product news, training information, support information, and more.
  - WebSphere Cast Iron Cloud Integration support
    A portal for support problems and their solutions, plus downloads, fixes, problem tracking, and more.
  - Cast Iron community forums
    Get answers to your technical questions and share your expertise with other WebSphere Cast Iron users.
  - IBM Software Services for WebSphere Cast Iron cloud integration
    Rapidly deploy cloud, on-premise, or hybrid applications with help from IBM Software Services for WebSphere. Our team of Cast Iron cloud integration experts has deep technical skills and experience from thousands of customer integrations.
  - IBM Redbook: Getting started with IBM WebSphere Cast Iron Cloud Integration
    Detailed introduction to the development and administrative interfaces for WebSphere Cast Iron.
  - Connect cloud and on-premise applications using IBM Cast Iron OmniConnect
    This IBM Redguide for business leaders shows you how to use IBM Cast Iron OmniConnect to connect cloud and on-premise applications quickly and easily using "configure not code" cloud integration with built-in connectivity and integration templates.

- **WebSphere resources**
  - developerWorks WebSphere
    Technical information and resources for developers who use WebSphere products. developerWorks WebSphere provides downloads, how-to information, support resources, and a free technical library of more than 2000 technical articles, tutorials, best practices, IBM Redbooks, and product manuals.
  - Most popular WebSphere trial downloads
    No-charge trial downloads for key WebSphere products.
  - WebSphere forums
Product-specific forums where you can get answers to your technical questions and share your expertise with other WebSphere users.

- **WebSphere demos**
  Download and watch these self-running demos, and learn how WebSphere products can provide business advantage for your company.

- **WebSphere-related articles on developerWorks**
  Over 3000 edited and categorized articles on WebSphere and related technologies by top practitioners and consultants inside and outside IBM. Search for what you need.

- **WebSphere-related books from IBM Press**
  Convenient online ordering through Barnes & Noble.

- **WebSphere-related events**
  Conferences, trade shows, Webcasts, and other events around the world of interest to WebSphere developers.

- **developerWorks resources**
  - **Trial downloads for IBM software products**
    No-charge trial downloads for selected IBM® DB2®, Lotus®, Rational®, Tivoli®, and WebSphere® products.
  - **developerWorks business process management developer resources**
    BPM how-to articles, downloads, tutorials, education, product info, and other resources to help you model, assemble, deploy, and manage business processes.
  - **developerWorks blogs**
    Join a conversation with developerWorks users and authors, and IBM editors and developers.
  - **developerWorks tech briefings**
    Free technical sessions by IBM experts to accelerate your learning curve and help you succeed in your most challenging software projects. Sessions range from one-hour virtual briefings to half-day and full-day live sessions in cities worldwide.
  - **developerWorks podcasts**
    Listen to interesting and offbeat interviews and discussions with software innovators.
  - **developerWorks on Twitter**
    Check out recent Twitter messages and URLs.
  - **IBM Education Assistant**
    A collection of multimedia educational modules that will help you better understand IBM software products and use them more effectively to meet your business requirements.
About the authors

Madhavi Kota

Madhavi Kota is a Staff Software Engineer on the WebSphere Cast Iron team. She has a Bachelor of Technology degree from Vignan Institute of Technology and Science in Chennai, India. She has worked with the IBM TXSeries for Multiplatforms and IBM WebSphere Adapters teams, and she currently works on the WebSphere Cast Iron team using technologies such as Java and Rational Functional Tester. You can contact Madhavi at madhkota@in.ibm.com.

Aparna Srinivasan

Aparna Srinivasan is a Software Engineer and Level-3 Support Engineer on the WebSphere Cast Iron Adapters Team. She has a Bachelor of Engineering degree from Anna University in Chennai, India, and she has three years of experience working with various technologies such as Java and Rational Functional Tester. You can contact Aparna atapasrini@in.ibm.com.

© Copyright IBM Corporation 2014

Trademarks
(www.ibm.com/developerworks/ibm/trademarks/)