IBM Security Identity Governance and Intelligence
Version 5.2.3

Reference Topics

IBM Security Identity Governance and Intelligence
Version 5.2.3

Reference Topics

Table of contents

Table list

Chapter 1. Application programming interfaces

Application programming interfaces (APIs) are part of a plug-in model that you can use to add applications without disrupting existing applications.

Remote application programs run outside of the Identity Governance and Intelligence Java™ virtual machine (JVM). Classes outside of the application packages are not intended to be started by a remote application. Classes in remote applications are documented under the Identity Governance and Intelligence application packages. Server extensions, which run in the Identity Governance and Intelligence JVM, can use any of the classes that are listed in the published API documentation (Javadoc). They are Java classes that run in the same JVM of the caller. These APIs are used to develop Identity Governance and Intelligence customization and extensions that can plug into Identity Governance and Intelligence.

Several application APIs can be started by a remote application. A few server extension APIs in the data services package are also included. The following application APIs are intended to be started by a remote application:

Identity Governance and Intelligence Software Development Kit (SDK)

The Identity Governance and Intelligence SDK can be accessed directly from the Virtual Appliance Dashboard. The SDK contains the following elements.

Table 1. Identity Governance and Intelligence SDK contents

Folders	Contains the following files
`customization`	Files used to customize Identity Governance and Intelligence. For example, adding a custom application in the desk, changing the labels and descriptions of the applications, and setting the date and time format for the entire product". See Customization features.
`javaDocAGCore`	The Javadoc, which provides the documentation for the Identity Governance and Intelligence EJB.
`lib`	The binary versions of the IBM® Security Identity Governance libraries and WebSphere® Application Server client to compile the SDK source.
`Readme`	A `README.txt` file.
`RESTDoc`	Documentation to create REST API calls to the Identity Governance and Intelligence services. See “REST APIs” on page 2.
`RESTExamples`	Examples of the REST API calls.
`src`	The source code of the SDK.
`sas.client.props`	The WebSphere Application Server access configuration information.
`ssl.client.props`	The SSL information.

EJB APIs

The Javadoc includes a set of Java packages that contains:

- The interfaces and methods for managing a certain set of functions, and
- The EJB that these interfaces and methods use

These packages allow a third-party application to establish interoperability with Identity Governance and Intelligence and calls a large set of functions, through EJB technology.

The main package of this set is `com.engiweb.profilemanager.common.interfaces`. It contains:

- The main set of interfaces for the interoperability with the Access Governance Core module
- The Interface ISec API, for managing the authorization function

For more information, see the Identity Governance and Intelligence Javadoc

REST APIs

The REST APIs provide third-party applications some functionality and the interface for operating with Identity Governance and Intelligence. Identity Governance and Intelligence client components send the queries to these REST APIs.

Identity Governance and Intelligence External Authorization Services API
: Accepts or refuses the received request. Use these REST APIs when the RESTful web server returns `3 = WAITING_ASYNCHRONOUS`. The RESTful web service must meet the requirements that are specified in the `IGI External Authorization Services.html` file. Otherwise, external authorization cannot work.

External Authorization Services API
: Manages the request related to a list of permissions or roles that can be added, removed, or renewed according to the request type. See `External Authorization Services.html` for complete information about creating the correct RESTful web service for external authorization.

External SoD APIs
: Checks if the entitlement, group, or user presents Segregation of Duties risks. See `ExtSODServices.html`.

External authorization

Virtual appliance REST APIs
You can develop custom applications with the REST application programming interfaces (APIs) that are supported by IBM Security Identity Governance and Intelligence virtual appliance. The REST APIs are web services that are available so you can administer tasks outside of the virtual appliance user interface.

Identity Brokerage REST APIs
The Identity Brokerage provides a REST API for managing accounts, targets, target profiles, groups membership (modify only), and permissions. The API implements the Simple Cloud Identity Management (SCIM) standard Version 2.0 with custom schema extensions. This implementation enables developers to access and manage identity resources directly by developing client applications that can be invoked from anywhere within the network.

Virtual appliance REST APIs

You can develop custom applications with the REST application programming interfaces (APIs) that are supported by IBM Security Identity Governance and Intelligence virtual appliance. The REST APIs are web services that are available so you can administer tasks outside of the virtual appliance user interface.

The REST APIs are separated into a set of functional components of the virtual appliance. The following list describes the components

Analysis and Diagnostics Monitoring
View information about analysis and diagnostic tools such as SNMP monitoring, storage, CPU usage, memory statistics, event logs, and appliance status.

System Settings Management
View information about export or import settings, network settings, system settings, maintenance, and other aspects.

Configuration Management
View server-setting configuration information, which includes custom files, certificates, mail server, and external entities configuration information about the directory and database servers.

Dashboard
View information about quick links, interfaces, middleware and server monitoring, notifications, partition information, and disk usage.

Download REST API documentation

The REST API documentation for IBM Security Identity Governance and Intelligence virtual appliance is packaged in a compressed file.

Complete these steps.

1. Access http://www.ibm.com/support/docview.wss?uid=swg27046896.
2. Download the `RAPI_DOCS.zip` file to a folder on your local computer.
3. Extract the `RAPI_DOCS.zip` file.
4. Open the `index.html` file to view the REST API documentation.

Identity Brokerage REST APIs

The Identity Brokerage provides a REST API for managing accounts, targets, target profiles, groups membership (modify only), and permissions. The API implements the Simple Cloud Identity Management (SCIM) standard Version 2.0 with custom schema extensions. This implementation enables developers to access and manage identity resources directly by developing client applications that can be invoked from anywhere within the network.

Supported REST APIs

The following table lists the Identity Governance and Intelligence supported Identity Brokerage REST APIs.

Note:

- For resource search, limited filtering capability is supported for User and Group resources. All other Identity Brokerage resources do not support filtering.

- For resource search, sorting and pagination are not supported. A search limit is specified in the Identity Brokerage properties file to specify the maximum number of returns that are supported by Identity Brokerage. This search limit applies to all Identity Brokerage managed resources.
- For resource search, the attributes query parameter is supported for User and Group resources to adjust the information that is returned.

Table 2. Supported Identity Brokerage REST APIs

Category	Resource	Endpoint	Operations	Description
Target Profile Management	TargetProfile	`/TargetProfiles`	POST	Loads or updates a target profile that contains metadata for supported targets, including service provider configuration, resource types, and schemas.
	TargetProfile	`/TargetProfileJar`	POST	Loads or updates a connector profile that contains metadata for supported targets, including service provider configuration, resource types, and schemas.
	LanguagePack	`/LanguagePack`	POST	Loads or updates the language pack JAR file that contains labels that are used for localized messages.
	TargetProfile	`/TargetProfiles`	GET	Returns a list of all target profiles that are loaded.
	TargetProfile	`/TargetProfiles/{profile}`	GET	Returns the information for the specified profile.
	Schema	`/TargetProfiles/{profile}/Schema`	GET	Returns the schema definitions for the targets of the specified profile.
Target Management	Target	`/Targets`	POST	Defines a target to Identity Brokerage.

Table 2. Supported Identity Brokerage REST APIs (continued)

Category	Resource	Endpoint	Operations	Description
	Target	`/Targets`	GET	Returns a list of all targets that are managed by Identity Brokerage.
	Target	`/Targets/{targetId}`	GET	Returns the information, which includes connection status for the specified target.
	ServiceProviderConfig	`/Targets/{id}/ServiceProviderConfig`	GET	Returns the service provider definition for the specified target.
	Schema	`/Targets/{id}/Schema`	GET	Returns the schema definitions for the specified target.
	Target	`/Targets/{id}`	PUT	Modifies the target information in Identity Brokerage.
	Target	`/Targets/{id}`	DELETE	Deletes the specified target from Identity Brokerage.
User Management	User	`/Targets/{targetId}/Users`	POST	Adds a user to the specified target.
	User	`/Targets/{targetId}/Users`	GET	Returns a list of all users for the specified target.
	User	`/Targets/{targetId}/Users/{userId}`	GET	Returns the information for the specified user.
	Users	`/Targets/{targetId}/Users/{userId}`	PATCH	Adds, modifies, or deletes attribute values for the specified user.
	Users	`/Targets/{targetId}/Users/{userId}`	PUT	Replaces attributes of the specified user with the specified values.
	Users	`/Targets/{targetId}/Users/{userId}`	DELETE	Deletes the specified user from the target.

Table 2. Supported Identity Brokerage REST APIs (continued)

Category	Resource	Endpoint	Operations	Description
Group Management	Group	`/Targets/{targetId}/Groups`	GET	Returns a list of all groups for the specified target.
	Group	`/Targets/{targetId}/Groups/{groupId}`	GET	Returns the information for the specified group.
	Group	`/Targets/{targetId}/Groups/{groupId}`	PATCH	Adds, modifies, or deletes attribute values for the specified group.
		`/labels?profile={profile}&key={key}`	GET	Returns all the localized labels that are currently available for the specified profile and label key.*
		`/Forms/{profile}/Target`	GET	Returns the information that is needed to create a form to configure a connector for the specified profile type. The information includes the attributes for the connector along with metadata for each attribute.*

Note: The `/labels` and the `/Forms` APIs do not implement SCIM v2.0. The differences between the APIs that are not SCIM-compliant as compared to the SCIM-compliant APIs are listed:

- The data that is returned by the non-compliant APIs are not SCIM resources and cannot be managed through the Identity Brokerage.
- The APIs are accessed through the `/config` context, not the `/identity` context. For example, `https://{host}:{port}/BrokerageService/config/Forms/{profile}/Target`
- The response messages that are returned by these APIs are in JSON format, but they are not SCIM-compliant. Therefore, the caller must provide an Accept header of "application/json" instead of "application/scim+json".

Procedure

1. The Identity Brokerage REST API is disabled by default. To enable it, see "Enabling or disabling the Identity Brokerage REST API" on page 7.

2. Set up basic authentication to access REST APIs by creating Identity Brokerage users. See “Managing Identity Brokerage users and passwords” to create the authentication credentials.
3. Enable HTTPS communication to the Identity Brokerage. Only HTTPS communication is supported. See Managing certificates to enable secure communication.

   Note:

   - The Identity Brokerage profile uses the same certificate store as the Identity Governance and Intelligence.
   - Use port 8443 for the external client to use the Identity Brokerage REST API. This port is blocked by default. To enable it, see “Enabling or disabling the Identity Brokerage REST API.”
4. View and run the sample client.

API documentation

To access the REST APIs documentation:

1. Download the `Brokerage Provider SCIM APIs.zip` file from http://www.ibm.com/support/docview.wss?uid=swg27048142 into a folder on your local computer.
2. Extract the `Brokerage Provider SCIM APIs.zip` file.
3. Open the `index.html` file.

Limitations

Attribute values are case-sensitive. When you delete an attribute value, make sure to specify the value in its exact case when it was added to the account. Otherwise, the delete request fails.

Enabling or disabling the Identity Brokerage REST API

The Identity Brokerage REST API is disabled by default. Enable it to use the REST API for managing accounts, groups membership, and permissions.

Procedure

1. Access the command line interface console.
2. To enable:
   a. Enter **igi utilities ib_settings ib_api enable**.
   b. Enter **YES** to confirm the request.
3. To disable:
   a. Enter **igi utilities ib_settings ib_api disable**.
   b. Enter **YES** to confirm the request.

Managing Identity Brokerage users and passwords

Use the **IBPasswordSetter** utility to add, update, or deactivate an Identity Brokerage user and its password from the authentication table.

About this task

Add the Identity Brokerage user with password in the authentication table so that the user can create a client application to communicate with the Identity Brokerage.

Note: If the user is already created, you cannot create it again even if the user is deactivated.

Update the Identity Brokerage password whenever applicable.

Deactivate the Identity Brokerage user and account password if the user no longer needs access to the Identity Brokerage REST APIs.

This procedure is intended only for external users who wants to access the Identity Brokerage REST APIs.

Procedure

1. Access the command line interface console.
2. Enter **igi utilities ib_settings users**.
3. To add an Identity Brokerage user:
   a. Enter **create**.
   b. Enter the user name.
   c. Enter the password.
   d. Re-enter the password for confirmation.
4. To change the Identity Brokerage user password:
   a. Enter **change_password**.
   b. Enter the index corresponding to the user.
   c. Enter the new password.
   d. Re-enter the password for confirmation.

   Note:

   Create or update the password based on the password policy.

   The password must be at least 8 characters. It must contain one upper case, one lower case, one numerical, and one special character. The special character cannot be <, >, `, $, |, ;, or &.
5. To deactivate a user:
   a. Enter **deactivate** to display the list of available Identity Brokerage users.
   b. Enter the index corresponding to the user.
   c. Enter **YES** to confirm the request.
6. To reactivate a user:
   a. Enter **reactivate** to display the list of deactivated Identity Brokerage users.
   b. Enter the index corresponding to the user.
   c. Enter the new password.
   d. Re-enter the password for confirmation.
7. To view the list of available Identity Brokerage users, enter **list_users**.

Chapter 2. Rules overview

You can use *rules* to define event management that is based on an event type. You can also use rules to automate particular policies.

Configuring rules for Access Risk Controls for SAP

Configuring rules for Access Governance Core
Rules are used to manage different types of events or for the automation of particular policies.

DeferredEvents_IN_ORGUNIT_ADD

This flow of rules, named **ORGUNIT_ADD**, adds an organizational unit (OU) to the data model. The flow has only one rule: Add OU.

Rule Class = *Deferred Events*

Queue = *IN*

Note: The structure and the content of the flow could be the same as the flow LifeEvents_IN_ORGUNIT_ADD. It's always possible to change the content of the flow for addressing specific needs of the customer.

The flow LifeEvents_IN_ORGUNIT_ADD is run in "real time", while the flow DeferredEvents_IN_ORGUNIT_ADD must be associated to a task that is scheduled with Task Planner.

In this case, see Enabling a flow of rules to be deferred.

LifeEvents_IN_ORGUNIT_ADD

This flow of rules, named **ORGUNIT_ADD**, adds an organizational unit (OU) to the data model. The flow has only one rule: Add OU.

Rule Class = *Live Events*

Queue = *IN*

Note: If you change the sequence of rules in the flow, the behavior of the flow becomes unpredictable. Only expert administrators can change the sequence.

Add OU

In input, are provided the OU to add and the needed boolean parameters.

You can decide to inherit the roles (false) and the OU resources of the parent OU.

```
when
orgUnit : OrgUnitBean( )
orgUnitErcBean : OrgUnitErcBean( )

then
```

```
// [ V1.1 - 2014-05-26 ]
OrgUnitAction.add(sql, orgUnit, false, false);
logger.debug("OU created : " + orgUnit);
```

Index

R

Printed in USA