Zscaler and Saviynt Connector Deployment Guide

November 2021

Version 1.1

Zscaler Business Development – Solutions Architecture Team

Table of Contents

About This Document

Zscaler Overview

Zscaler (Nasdaq: ZS), enables the world's leading organizations to securely transform their networks and applications for a mobile and cloud-first world. Its flagship Zscaler Internet Access (ZIA) and Zscaler Private Access (ZPA) services create fast, secure connections between users and applications, regardless of device, location, or network. Zscaler delivers its services 100% in the cloud and offers the simplicity, enhanced security, and improved user experience that traditional appliances or hybrid solutions can't match. Used in more than 185 countries, Zscaler operates a massive, global cloud security platform that protects thousands of enterprises and government agencies from cyberattacks and data loss. For more information on Zscaler, visit www.zscaler.com or follow Zscaler on Twitter @zscaler.

Saviynt Overview

Saviynt's cloud-architected identity and access governance platform helps modern enterprises scale cloud initiatives and solve the toughest security and compliance challenges in record time. The company brings together identity governance (IGA), granular application access, cloud security, and privileged access management (PAM) to secure the entire business ecosystem and provide a frictionless user experience. The world's largest brands trust Saviynt to accelerate business transformation, empower distributed workforces, and meet continuous compliance. For more information, please visit www.saviynt.com.

Audience

This guide is for network administrators, endpoint and IT administrators, and security analysts responsible for deploying, monitoring, and managing enterprise security systems. For additional product and company resources, please refer to:

- *Zscaler Resources*
- *Saviynt Resources*
- *Appendix A: Requesting Zscaler* Support

Software Versions

This document was authored using:

- ECM release version 6.0 and later
- Zscaler release version 11.3 and later

Request for Comments

- **For Prospects and Customers:** We value reader opinions and experiences. Please contact us at partner-doc-support@zscaler.com to offer feedback or corrections for this guide.
- **For Zscaler Employees:** Please contact z-bd-sa@zscaler.com to reach the team that validated and authored the integrations in this document.

Zscaler and Saviynt Introduction

Zscaler Overview

Below are overviews of the Zscaler and Saviynt applications described in this section.

Zscaler Internet Access (ZIA) Overview

Zscaler Internet Access (ZIA) is a secure Internet and web gateway delivered as a service from the cloud. Think of it as a secure Internet onramp—all you do is make Zscaler your next hop to the Internet via one of the following methods:

- Setting up a tunnel (GRE or IPSec) to the closest Zscaler data center (for offices).
- Forwarding traffic via our lightweight Zscaler Client Connector or PAC file (for mobile employees).

No matter where users connect—a coffee shop in Milan, a hotel in Hong Kong, or a VDI instance in South Korea—they get identical protection. ZIA sits between your users and the Internet and inspects every transaction inline across multiple security techniques (even within SSL).

You get full protection from web and Internet threats. The Zscaler cloud platform supports Cloud Firewall, IPS, Sandboxing, DLP, CASB, and Browser Isolation, allowing you start with the services you need now and activate others as your needs grow.

Zscaler Private Access (ZPA) Overview

Zscaler Private Access (ZPA) is a cloud service that provides secure remote access to internal applications running on cloud or data center using a zero trust framework. With ZPA, applications are never exposed to the internet, making them completely invisible to unauthorized users. The service enables the applications to connect to users via inside-out connectivity rather than extending the network to them.

ZPA provides a simple, secure, and effective way to access internal applications. Access is based on policies created by the IT administrator within the ZPA Admin Portal and hosted within the Zscaler cloud. On each user device, a piece of software called Zscaler Client Connector is installed. Zscaler Client Connector ensures the user's device posture and extends a secure micro-tunnel out to the Zscaler cloud when a user attempts to access an internal application.

Zscaler Resources

The following table contains links to Zscaler resources based on general topic areas.

Name and Link	*Description*
ZIA Help Portal	Help articles for ZIA.
ZPA Help Portal	Help articles for ZPA
ZPA Posture Profiles	Help link for how to configure ZPA posture profiles.
ZPA Access Policies	Help link for how to configure ZPA access policies with a set of configuration examples.
Zscaler Tools	Troubleshooting, security and analytics, and browser extensions that help Zscaler determine your security needs.

Zscaler Training and Certification	Training designed to help you maximize Zscaler products.
Submit a Zscaler Support Ticket	Zscaler support portal for submitting requests and issues.

Saviynt Overview

Saviynt Enterprise Identity Cloud Overview

The Saviynt Enterprise Identity Cloud (EIC) combines multiple identity management capabilities into a single cohesive platform. Unify controls and risk management for every identity, app, and cloud across your business. Onboard people, apps, and machines in minutes and selectively turn on access and governance functionality.

- **Designed for Rapid Deployment**. Configure without code and use our industry integrations, templates, and control libraries to deploy Saviynt in weeks.
- **Built on Zero Trust**. Automate dynamic access management and introduce just-in-time privilege elevation & time-bound access for any human or machine identity.
- **Continuous Compliance Ready**. Simplify audits with an assured compliance framework, reduce fraud with cross-application SoD management, and automated controls monitoring.
- **Trusted and Secure**. Meet evolving security and industry regulations with a SOC, ISO, and FedRAMP Moderate certified identity platform.

Saviynt Resources

The following table contains links to Saviynt support resources. You will need a Saviynt customer login to access help and support resources.

Name and Link	*Description*
Saviynt Enterprise Identity Cloud	Description of the EIC properties and value.
Saviynt Customer Support	Saviynt support portal for submitting requests and issues.
Saviynt Solution Guides	Solution guides help enterprises easily configure our products with their existing software solutions

Zscaler Connector Guide

This guide describes the Zscaler connector used to integrate Saviynt Enterprise Identity Cloud (EIC) with Zscaler Private Access (ZPA) and Zscaler Internet Access (ZIA).

Introduction

Zscaler is the creator of the Zero Trust Exchange platform that transforms and empowers an anywhere-workforce seamlessly and securely by embracing a zero-trust mindset. At a high-level, Zscaler comprises of elements such as users, groups, and policies.

The Zscaler connector creates an integration with ZPA and ZIA applications to manage Zscaler users and gain visibility of their groups and user-group memberships from EIC.

> **NOTE**
>
> This guide provides information about using the Zscaler (SCIM-based) connector for performing operations listed in the Supported Features section.

Supported Features

The Zscaler connector supports the following features:

		Import		*Provisioning*		
Zscaler Object	***EIC Object***	***Full Import***	***Incremental Import***	***Lifecycle Management***	***Add or Remove Access***	***Additional Configurations***
Users	Accounts	Yes	No	Support for creating and removing accounts		
Groups	Groups	Yes	No	Not applicable	Support for adding group members and removing group members	Groups

> **NOTE**
>
> The features listed above are currently supported in EIC. Any new enhancements will be communicated via the Release Notes.

Understanding the Integration Between Saviynt and Zscaler

You must integrate the EIC and the collaboration platform hosted by the target application (Zscaler, in this case) to execute import, provisioning, and deprovisioning tasks. The following components are involved in the integration:

- **Zscaler**, which is the target application for which EIC manages the identity lifecycle. Zscaler integrates with EIC through the connector to import, manage accounts, and access data.
- **Objects**, which are imported as entitlement types into EIC.
- A **Security System**, which represents the connection between EIC and the target application.
    - The security system is an endpoint that is the target application for which you want EIC to manage the identity repository.
    - The security system provides application instance abstraction from connectivity, including high-level metadata. You can select one connection for importing data from the target application and another connection for provisioning data to the target application. For more information about creating a security system, see Creating a Security System.
- An **Endpoint** is an instance of an application within the context of a security system.
    - Endpoints are the target applications that the connector to which the imports or exports data and performs provisioning or deprovisioning of identity objects such as users, accounts, and entitlements.
    - You must create an endpoint after creating the security system. You can associate a single security system with multiple endpoints if the deployment involves modelling of multiple isolated virtual applications (based on sets of specific entitlements according to certain categories) within a single application instance. For more information about creating an endpoint, see Creating an Endpoint for the Security System.
- A **Connector** is a software component that enables communication between the EIC and the target application. It provides a simplified integration mechanism where in some instances you only need to create a connection with minimal connectivity information for your target application. The REST Connector is used for importing, provisioning, and accessing accounts through the SCIM APIs. For more information about creating a connection, see Creating a Connection.
- The **Job Scheduler** is a software component that executes a job based on the configured schedule to perform import or provisioning operations from EIC. When a provisioning job is triggered, it creates provisioning tasks in EIC. When these tasks are completed, the provisioning action is performed on the target application through the configured connector. If you want to instantly provision requests for completing the tasks without running the provisioning job, you must enable Instant Provisioning at the security system level and the **Instant Provisioning Tasks** global configuration. For more information about the jobs used by the connectors in the Zscaler integration, see Using the Zscaler Connector.

Connector Architecture

The EIC uses a REST connection to integrate with Zscaler and import data, as well as for provisioning and deprovisioning tasks. The REST connection uses the System for Cross Identity Management (SCIM) protocol to communicate with Zscaler's SCIM interface.

The following diagram illustrates the connector architecture and communication with the target application.

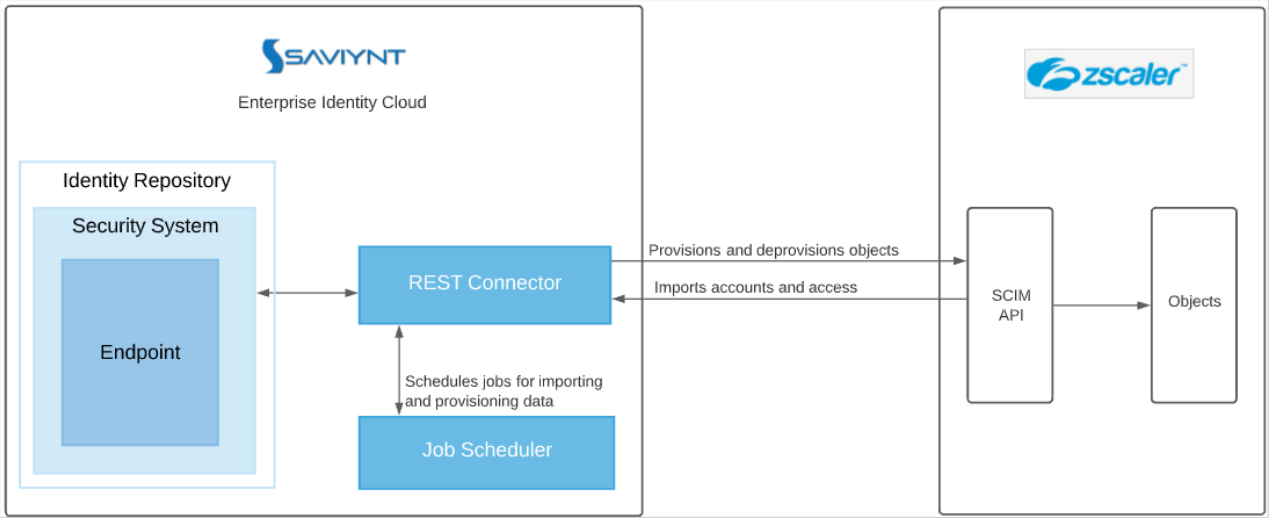

Figure 1. *Zscaler Connector Architecture*

Configuring a Connection

Prerequisites

The connector configuration parameters use an access token and URL for its initial authentication to ZPA and ZIA applications, and to authorize subsequent calls for performing additional transactions.

- Generate the access token and the URL for ZPA application. For more information, see the *Generating Access Token and URL for the ZPA Application* section in Prerequisites.
- Generate the access token and the URL for ZIA application. For more information, see the *Generating Access Token and URL for the ZIA Application* section in Prerequisites.

Generating Access Token and URL for the ZPA Application

Perform the following steps to generate access token and URL for the ZPA application:

1. Log in to ZPA Admin portal using administrator credentials.
2. Navigate to **Administration** > **IdP Configuration Settings**.

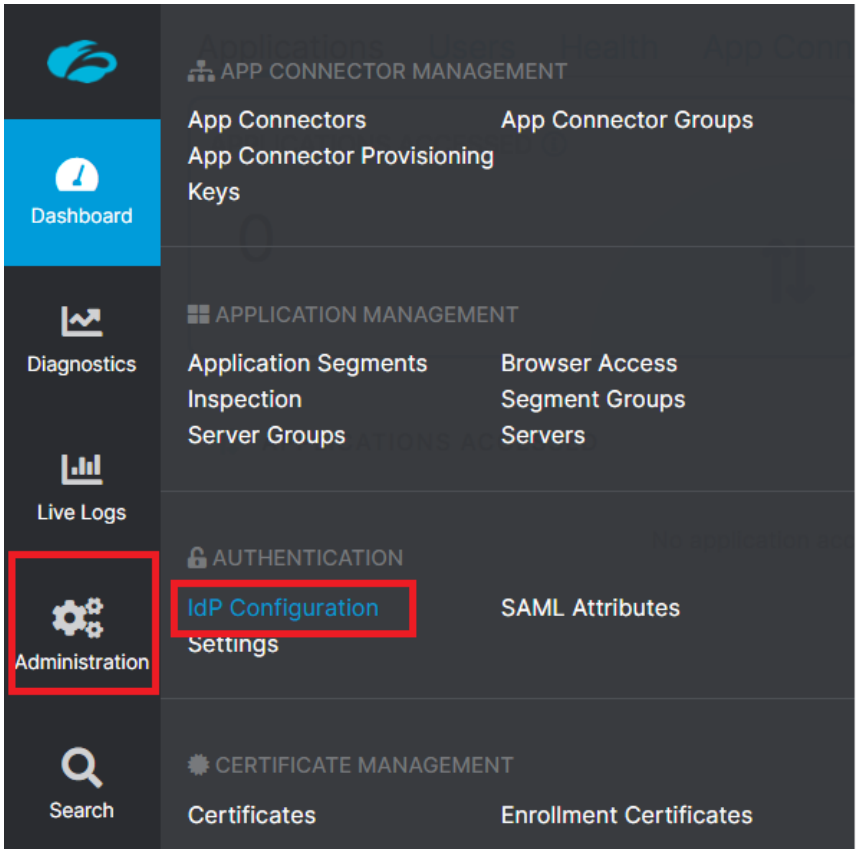

Figure 2. *IdP Configuration Settings*

3. Specify the **IdP Configuration details.** The SCIM Configuration page displays the endpoint URL and the access token.

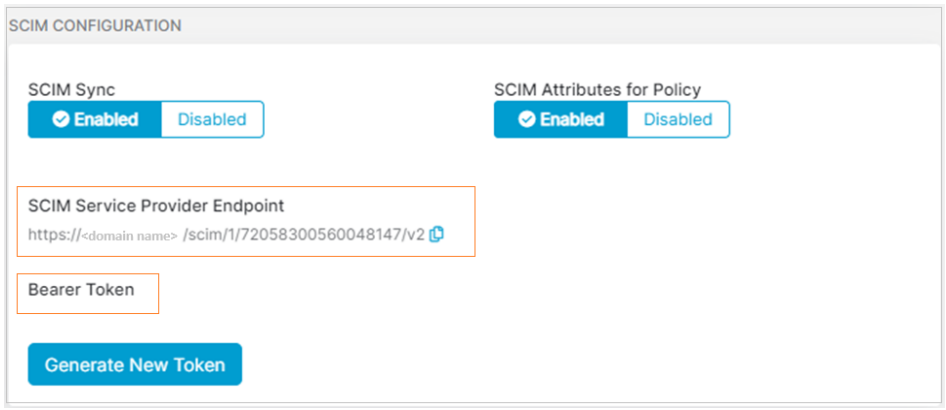

Figure 3. *IdP Configuration details*

In the figure above, 72058300560048147 is the ZPA account number.

Generating Access Token and URL for the ZIA Application

Perform the following steps to generate access token and URL for the ZIA application:

1. Log in to ZPA Admin portal using administrator credentials.
2. Navigate to **Administration** > **Authentication Settings**.

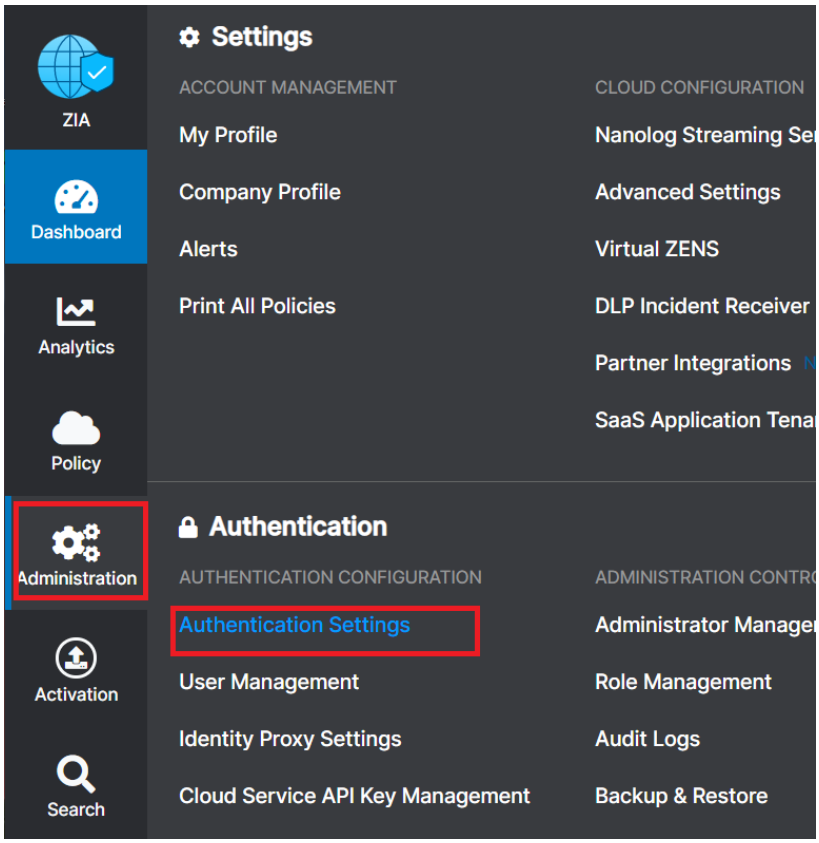

Figure 4. *Authentication Settings*

3. In the **Authentication Settings** page that displays, click **Add IdP**.

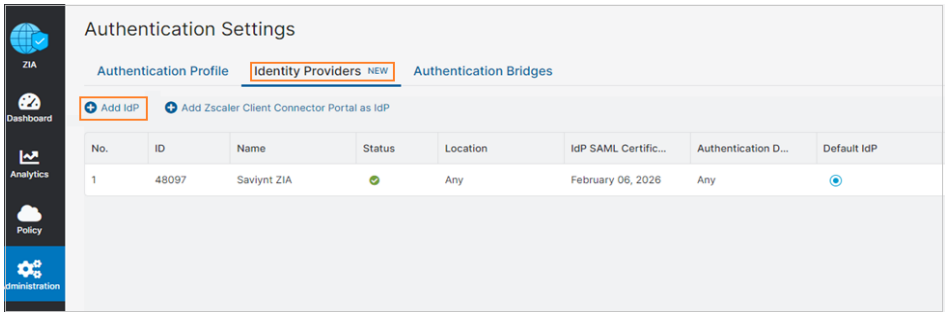

Figure 5. Add IdP

4. Specify the **IdP Configuration details.** The **SCIM Configuration** page displays the endpoint URL and the access token.

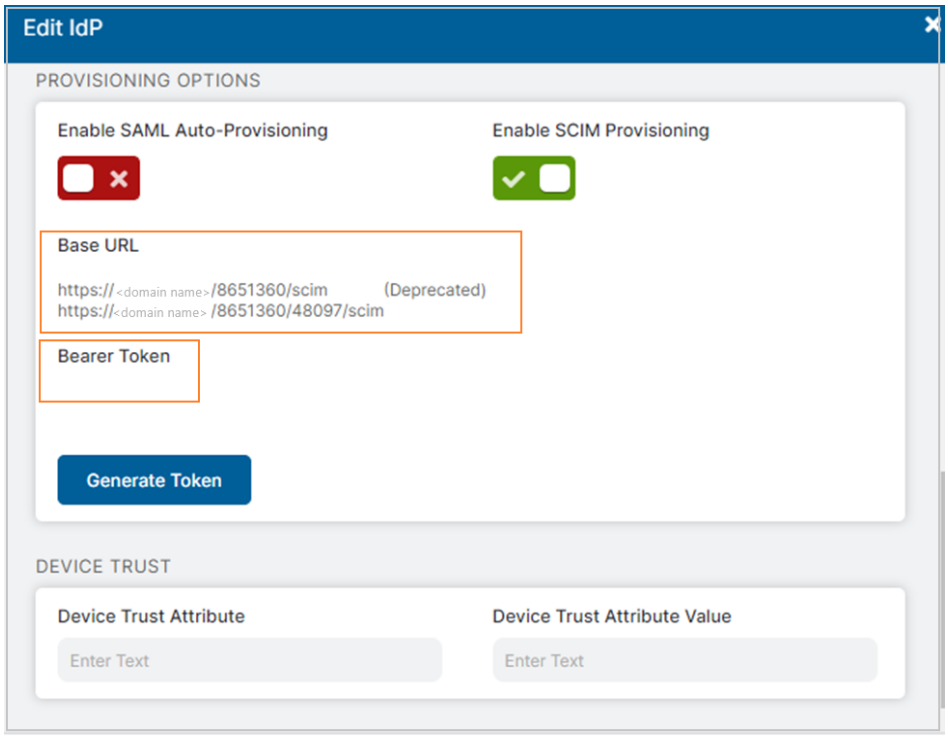

Figure 6. *IdP Configuration page*

In the figure above, 8651360/48096 is the ZIA account number.

Creating a Connection

A connection refers to the configuration setup for connecting EIC to target applications. For more information about the procedure to create a connection, see Creating Connections.

Understanding the Configuration Parameters

When creating a connection, you must specify connection parameters that the connector uses to connect with the target application, define the type of operations to perform, the target application objects against which those operations are performed, and the frequency of performing them. In addition, you can view and edit attribute mappings between the EIC and the target application, predefined correlation rules, and provisioning and import jobs.

Configuration Parameters for Account and Access Import

The connector uses the following parameters for creating a connection and for importing account and access from the target application:

Connection Parameters

Parameter	*Description*	*Example Configuration*	*Mandatory?*
Connection Name	Specify the name to identify the connection.	-	Yes
Connection Description	Specify the description for the connection.	-	No
Connection Type	Select the connection type as **REST**.	-	Yes
Default SAV Role	Specify this parameter to assign the SAV role for the connection. The SAV role is a role in EIC that assigns specific access to users. This parameter is valid only for importing users. **Sample value:** User assigned with the **ROLE_ADMIN** role, has access to all the sections of EIC.	-	No
Email Template	Specify this parameter to select an email template for sending notifications. Email templates provide immediate trigger of emails to a user based on actions performed. Email informs user about the action performed and if critical, needs immediate action from the user.	-	No

ConnectionJSON	Specify this parameter to create a connection.	Use the following format to connect to the ZPA application:
		
		NOTE - the base URL will need to be updated to reflect your Zscaler tenant.
		`{`
		`  "authentications": {`
		`  "acctAuth": {`
		`  "authType": "oauth2",`
		`  "httpHeaders": {`
		`  "contentType": "application/json"`
		`  },`
		`  "authError": [`
		`  "InvalidAuthenticationToken",`
		`  "AuthenticationFailed"`
		`  ],`
		`  "url": "https://<domain name>/scim/1/72058300560048147/v2",`
		`  "httpMethod": "POST",`
		`  "httpContentType": "application/json",`
		`  "errorPath": "error.code",`
		`  "maxRefreshTryCount": 5,`
		`  "tokenResponsePath": "access_token",`
		`  "tokenType": "Bearer",`
		`  "authHeaderName": "Authorization",`
		`  "accessToken": "<access token>",`
		`  "httpParams": "[object Object]",`
		`  "retryFailureStatusCode": []`
		`  }`
		`  }`
		`}`
		Use the following format to connect to the ZIA application:
		
		NOTE - the base URL will need to be updated to reflect your Zscaler tenant.
		
		`{`
		`  "authentications": {`
		`  "acctAuth": {`
		`  "authType": "oauth2",`
		`  "httpHeaders": {`
		`  "contentType": "application/json"`
		`  },`
		`  "authError": [`
		`  "InvalidAuthenticationToken",`
		`  "AuthenticationFailed"`
		`  ],`
		`  "url": "https://<domain name>/8651360/48096/scim",`
		`  "httpMethod": "POST",`
			Yes

Parameter	*Description*	*Example Configuration*	*Mandatory?*
		"httpContentType": "application/json", "errorPath": "error.code", "maxRefreshTryCount": 5, "tokenResponsePath": "access_token", "tokenType": "Bearer", "authHeaderName": "Authorization", "accessToken": "<access token>", "httpParams": "[object Object]", "retryFailureStatusCode": [] } } } For more information on description of attributes in this parameter, see REST Connector Guide.	

Import Parameters

Parameter	*Description*	*Example Configuration*	*Mandatory?*

ImportAccountEntJSON	Specify this parameter to map attributes of Zscaler application to attributes of EIC for account and entitlement import.	Use the following format to import accounts and entitlements using the ZPA application:
		
		NOTE - the base URL will need to be updated to reflect your Zscaler tenant.
		{
		"accountParams": {
		"processingType":"SequentialAndIterative",
		"connection": "acctAuth",
		"createUsers": true,
		"call": {
		"call1": {
		"http": {
		"url": "https://<domain name>/scim/1/72058300560048147/v2/Users",
		"basicUrl": "<domain name>",
		"hostUrl": "/72058300560048128/scim/Users",
		"httpContentType": "application/json",
		"httpMethod": "GET",
		"httpHeaders": {
		"Authorization": "${access_token}",
		"Accept": "application/json"
		}
		},
		"listField": "Resources",
			Yes

		"keyField": "name",
		"colsToPropsMap": {
		"accountID": "id~#~char",
		"name": "userName~#~char",
		"displayName": "displayName~#~char",
		"customproperty1": "id~#~char",
		"customproperty2": "department~#~char"
		}}}
		},
		"entitlementParams": {
		"processingType":"SequentialAndIterative",
		"connection": "acctAuth",
		"entTypes": {
		"Entitlement": {
		"call": {
		"call1": {
		"connection": "restconnectorscim",
		"http": {
		"url": "https://<domain name>/scim/1/72058300560048147/v2/Groups",
		"basicUrl": "<domain name>",
		"hostUrl": "/72058300560048128/scim/Groups",
		"httpContentType": "application/json",
		"httpMethod": "GET",
		"httpHeaders": {
		"Authorization": "${access_token}",
		"Accept": "application/json"
		}
		},
		"listField": "Resources",
		"keyField": "entitlementID",
		"colsToPropsMap": {
		"entitlementID": "id~#~char",
		"entitlement_value": "displayName~#~char",
		"customproperty1": "id~#~char",
		"acctEntMappingInfoColumnFromEnt":
		"STORE#ACC#ENT#MAPPINGINFO~#~char"
		}
		}},
		"acctEntMappings": {
		"listField": "members",
		"idPath": "value",
		"keyField": "entitlementID"
		}
		}}},
		"acctEntParams": {
		"processingType": "entToAcctMapping"
		}
		}
		Use the following format to import accounts and entitlements using the ZIA application:	

NOTE - the base URL will need to be updated to reflect your Zscaler tenant.

```
{
"accountParams": {
"processingType":"SequentialAndIterative",
"connection": "acctAuth",
"createUsers": true,
"call": {
"call1": {
"http": {
"url": "https://<domain
name>/8651360/48097/scim/Users",
"basicUrl": "<domain name>",
"hostUrl": "/8651360/48097/scim/Users",
"httpContentType": "application/json",
"httpMethod": "GET",
"httpHeaders": {
        "Authorization": "${access_token}",
        "Accept": "application/json"
      }
},
"listField": "Resources",
"keyField": "name",
"colsToPropsMap": {
"accountID": "id~#~char",
"name": "userName~#~char",
"displayName": "displayName~#~char",
"customproperty1": "id~#~char",
"customproperty2": "department~#~char"
}}}
},
"entitlementParams": {
"processingType":"SequentialAndIterative",
"connection": "acctAuth",
"entTypes": {
"Entitlement": {
"call": {
"call1": {
"connection": "restconnectorscim",
"http": {
"url": "https://<domain
name>/8651360/48097/scim/Groups",
"basicUrl": "<domain name>",
"hostUrl": "/8651360/48097/scim/Groups",
"httpContentType": "application/json",
"httpMethod": "GET",
"httpHeaders": {
        "Authorization": "${access_token}",
        "Accept": "application/json"
      }
},
"listField": "Resources",
```

		"keyField": "entitlementID",
		"colsToPropsMap": {
		"entitlementID": "id~#~char",
		"entitlement_value": "displayName~#~char",
		"customproperty1": "id~#~char",
		"acctEntMappingInfoColumnFromEnt":
		"STORE#ACC#ENT#MAPPINGINFO~#~char"
		}
		}},
		"acctEntMappings": {
		"listField": "members",
		"idPath": "value",
		"keyField": "entitlementID"
		}
		}}},
		"acctEntParams": {
		"processingType": "entToAcctMapping"
		}
		}
		For more information on description of attributes in this parameter, see REST Connector Guide.	

Configuration Parameters for Provisioning

Parameter	*Description*	*Recommended Configuration*	*Binding Variables?*	*Java Operations?*

CreateAccountJSON	Specify this parameter to create an account in the target application.	Use the following format to create accounts using the ZPA application:
		
		NOTE - the base URL will need to be updated to reflect your Zscaler tenant.
		
		{
		"accountIdPath":
		"Entitlement.message.id",
		"responseColsToPropsMap": {
		"name":
		"Entitlement.message.userName~#~char",
		"displayName":
		"Entitlement.message.displayName~#~char"
		},
		"call": [
		{
		"name": "Entitlement",
		"connection": "acctAuth",
			Yes. The bindings supported are:
			• ServiceAccountOwnerMap
			• endpoints
			• accountName
			• userManager
			• approvers
			• arsTasks/task
			• managerAccount
			• password
			• requestid
			• response
			• connection
			• userAccount
			• requestAccessAttributes/reqAttrs
			• businessJustification
			• user
				Yes

		"url": "https://<domain name>/scim/1/72058300560048147/v2/Users",
		"httpMethod": "POST",
		"httpParams": "{\"schemas\":[\"urn:ietf:params:scim:schemas:core:2.0:User\",\"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User\"],\"userName\":\"${user.email}\",\"displayName\":\"${user.username}\",\"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User\" : {\"department\":\"Saviynt Global\"}}",
		"httpHeaders": {
		"Authorization": "${access_token}",
		"Accept": "application/json"
		},
		"httpContentType": "application/json",
		"successResponses": {
		"statusCode": [201]
		}
		}
		]
		}
		
		Use the following format to create accounts using the ZIA application:
		
		NOTE - the base URL will need to be updated to reflect your Zscaler tenant.
		
		{
		"accountIdPath": "Entitlement.message.id",
		"responseColsToPropsMap": {
		"name": "Entitlement.message.userName~#~char",
		"displayName": "Entitlement.message.displayName~#~char"
		},
		"call": [
		{
		"name": "Entitlement",
		"connection": "acctAuth",		

		<pre>"url": "https://<domain name>/8651360/48097/scim/Users",
		"httpMethod": "POST",
		"httpParams": "{\"schemas\":[\"urn:ietf:params:scim:schemas:core:2.0:User\",\"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User\"],\"userName\":\"${user.email}\",\"displayName\":\"${user.username}\",\"urn:ietf:params:scim:schemas:extension:enterprise:2.0:User\" : {\"department\":\"Saviynt Global\"}}",
		"httpHeaders": {
		"Authorization": "${access_token}",
		"Accept": "application/json"
		},
		"httpContentType": "application/json",
		"successResponses": {
		"statusCode": [
		201
		]
		}
		}
		]
		}</pre>
		For more information on description of attributes in this parameter, see REST Connector Guide.		

AddAccess JSON	Specify this parameter to add access to an account.	Use the following format to add access using the ZPA application: NOTE - the base URL will need to be updated to reflect your Zscaler tenant. `{ "call": [{ "name": "Entitlement", "connection": "acctAuth", "url": "https://<domain name>/scim/1/72058300560048147/v2/Groups/${entitlementValue.entitlementID}", "httpMethod": "PATCH", "httpParams": "{\"schemas\":[\"urn:ietf:params:scim:api:messages:2.0:PatchOp\"],\"Operations\":[{\"op\":\"add\",\"value\":{\"members\":[{\"display\":\"${account.name}\",\"value\":\"${account.accountID}\"}]}}]}", "httpHeaders": { "Authorization": "${access_token}", "Accept": "application/json" }, "httpContentType": "application/json", "successResponses": { "statusCode": [204, 200, 201] } }] }`  Use the following format to add access using the ZIA application:  NOTE - the base URL will need to be updated to reflect your Zscaler tenant.  `{ "call": [{ "name": "Entitlement",`	Yes. The bindings supported are: • ServiceAccountOwnerMap • endpoints • userManager • approvers • arsTasks/task • managerAccount • requestid • response • connection • userAccount • requestAccessAttributes/reqAttrs • businessJustification • user • account • entitlementValue	Yes

		<pre>"connection": "acctAuth",
		"url": "https://<domain name>/8651360/48097/scim/Groups/${entitlementValue.entitlementID}",
		"httpMethod": "PATCH",
		"httpParams": "{\"schemas\":[\"urn:ietf:params:scim:api:messages:2.0:PatchOp\"],\"Operations\":[{\"op\":\"add\",\"value\":{\"members\":[{\"display\":\"${account.name}\",\"value\":\"${account.accountID}\"}]}}]}",
		"httpHeaders": {
		"Authorization": "${access_token}",
		"Accept": "application/json"
		},
		"httpContentType": "application/json",
		"successResponses": {
		"statusCode": [
		204,
		200,
		201
		]
		}
		}
		]
		}</pre>
		For more information on description of attributes in this parameter, see REST Connector Guide.		

RemoveAccessJSON	Specify this parameter to remove access from an account.	Use the following format to remove access using the ZPA application:
		
		NOTE - the base URL will need to be updated to reflect your Zscaler tenant.
		
		<pre>{
		"call": [
		{
		"name": "Entitlement",
		"connection": "acctAuth",
		"url": "https://<domain name>/scim/1/72058300560048147/v2/Groups/${entitlementValue.entitlementID}",
		"httpMethod": "PATCH",</pre>
			Yes. The bindings supported are:
			• ServiceAccountOwnerMap
			• endpoints
			• userManager
			• approvers
			• arsTasks/task
			• managerAccount
			• requestid
			• response
			• connection
			• userAccount
			• requestAccessAttributes/reqAttrs
				Yes

		"httpParams": "{\"schemas\":[\"urn:ietf:params:scim:api:messages:2.0:PatchOp\"],\"Operations\":[{\"op\":\"remove\",\"path\":\"members[value eq \\\"${account.accountID}\\\"]\"}]}",
		"httpHeaders": {
		"Authorization": "${access_token}",
		"Accept": "application/json"
		},
		"httpContentType": "application/json",
		"successResponses": {
		"statusCode": [204, 200, 201]
		}
		}
		]
		}
		
		Use the following format to remove access using the ZIA application:
		
		NOTE - the base URL will need to be updated to reflect your Zscaler tenant.
		
		{
		"call": [
		{
		"name": "Entitlement",
		"connection": "acctAuth",
		"url": "https://<domain name>/8651360/48097/scim/Groups/${entitlementValue.entitlementID}",
		"httpMethod": "PATCH",
		"httpParams": "{\"schemas\":[\"urn:ietf:params:scim:api:messages:2.0:PatchOp\"],\"Operations\":[{\"op\":\"remove\",\"path\":\"members[value eq \\\"${account.accountID}\\\"]\"}]}",
		"httpHeaders": {
		"Authorization": "${access_token}",
			• businessJustification
			• user
			• account
			• entitlementValue
			• account_entitlements	

		<pre>"Accept": "application/json" }, "httpContentType": "application/json", "successResponses": { "statusCode": [204, 200, 201] } }] }</pre> For more information on description of attributes in this parameter, see REST Connector Guide.		
RemoveAccountJSON	Specify this parameter to remove the account.	Use the following format to remove accounts using the ZPA application: NOTE - the base URL will need to be updated to reflect your Zscaler tenant. <pre>{ "call": [{ "name": "Call1", "connection": "acctAuth", "url": "https://<domain name>/scim/1/72058300560048147/v2/Users/${account.accountID}", "httpMethod": "DELETE", "httpHeaders": { "Authorization": "${access_token}", "Accept": "application/json" }, "httpContentType": "application/json", "successResponses": { "statusCode": [200, 201, 204] } }] }</pre>	Yes. The bindings supported are: • ServiceAccountOwnerMap • endpoints • userManager • approvers • arsTasks/task • managerAccount • requestid • response • connection • userAccount • businessJustification • user • account	Yes

		Use the following format to remove accounts using the ZIA application: NOTE - the base URL will need to be updated to reflect your Zscaler tenant. `{ "call": [{ "name": "Call1", "connection": "acctAuth", "url": "https://<domain name>/8651360/48097/scim/Users/${account.accountID}", "httpMethod": "DELETE", "httpHeaders": { "Authorization": "${access_token}", "Accept": "application/json" }, "httpContentType": "application/json", "successResponses": { "statusCode": [200, 201, 204] } }] }` For more information on description of attributes in this parameter, see REST Connector Guide.		

Creating a Security System

The security system represents the connection between the EIC and the target application. For more information on creating a security system, see Creating a Security System**.**

Creating an Endpoint for the Security System

Endpoint refers to the target application used to import accounts and entitlements (access) to EIC. For more information on creating an endpoint, see Creating Endpoints.

Using the Zscaler Connector

You can use the Zscaler connector for performing import and provisioning operations after configuring it to meet your requirements.

Guidelines for Using the Connector

You must apply the following guidelines for configuring import:

- Run account import before running the access import
- Map all Zscaler attributes to the EIC account attributes using **ImportAccountEntJSON**

You must apply the following guidelines for configuring provisioning:

- Use Java ternary operators if you want to add conditions in the provisioning parameters.

Configuring Import Operations

- **Full account import:** When configuring the connection for the first time, first perform a full import of all existing accounts from the target application to the EIC. To perform a full import, the invoke API gets a response from the target application and maps the attributes in the target application with attributes in the EIC. As part of this process, the deleted accounts are also identified and marked as suspended from import service.
- **Full Access import:** When configuring the connection for the first time, first perform a full import of all existing access from the target application to the EIC. To perform a full import, the invoke API gets a response from the target application and maps the attributes in the target application with attributes in the EIC. As part of this process, the deleted entitlements are also identified and marked as inactive.

The reconciliation jobs are automatically created in the EIC after you create a connection for Zscaler. For more information about creating jobs, see Data Jobs.

Importing Accounts and Accesses

You must import accounts after the users are available in the EIC.

To import accounts:

1. Specify the connection and import parameters. For more information, see the *Configuration Parameters for Account and Access Import* section in Creating a Connection.

   NOTE:

   Ensure that you select the REST connection type.

2. Configure the **Application Data Import (Single Threaded)** job to import accounts and access. For more information, see Data Jobs.

Configuring Provisioning and Deprovisioning

Provisioning is automatically enabled when a connection is configured. For detailed information about performing provisioning tasks, see Access Request System.

To provision objects to the target application:

1. Specify the connection and provisioning parameters. For more information, see the *Configuration Parameters for Provisioning* section in Creating a Connection.

> **NOTE:**
>
> Ensure that you select the REST connection type.

2. Configure the **Provisioning** job **(WSRETRY)**. For more information, see Provisioning Jobs.

When a provisioning job is triggered, it creates provisioning tasks in EIC. When these tasks are completed, the provisioning action is performed on the target application through the connector.

Appendix A: Requesting Zscaler Support

Gather Support Information

You might sometimes need Zscaler support for provisioning certain services, or to help troubleshoot configuration and service issues. Zscaler support is available 24/7 hours a day, year-round.

To contact Zscaler support, select **Administration** > **Settings** > **Company profile**.

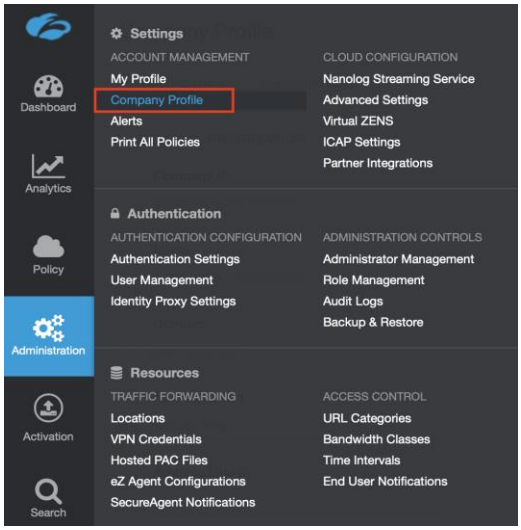

Figure 7. *Collecting details to open support case with Zscaler TAC*

Save Company ID

Copy the Company ID, as shown below.

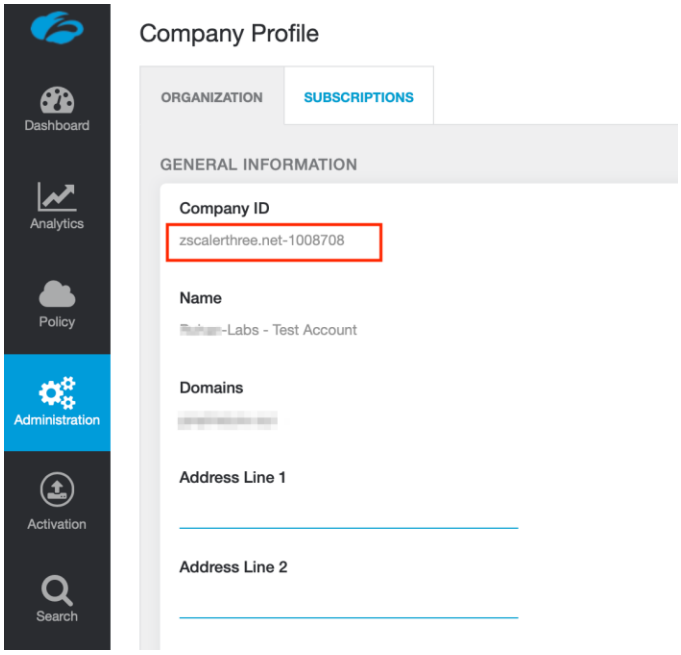

Figure 8. *Company ID*

Now that you have the company ID, you can open a support ticket. Navigate to **Dashboard** > **Support** > **Submit a Ticket**.

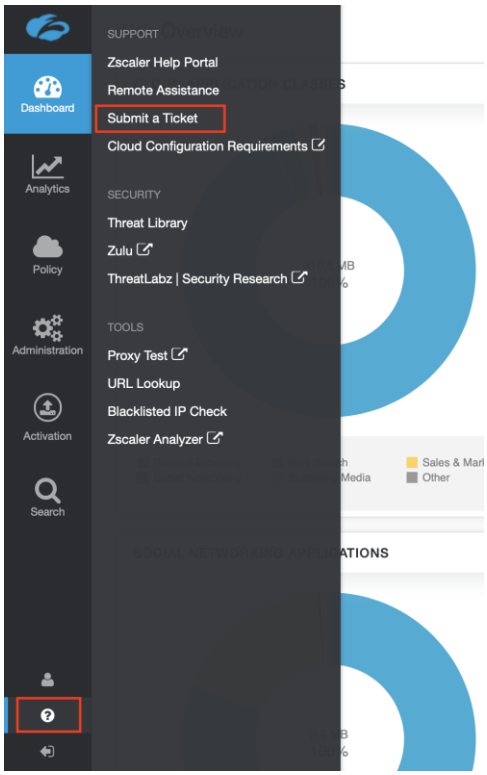

Figure 9. *Submit ticket*