



# **SailPoint Direct Connectors**

Version 7.1

# **Administration and Configuration Guide**

**Copyright © 2016 SailPoint Technologies, Inc., All Rights Reserved.**

SailPoint Technologies, Inc. makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. SailPoint Technologies shall not be liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

**Restricted Rights Legend.** All rights are reserved. No part of this document may be published, distributed, reproduced, publicly displayed, used to create derivative works, or translated to another language, without the prior written consent of SailPoint Technologies. The information contained in this document is subject to change without notice.

Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DOD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

**Regulatory/Export Compliance.** The export and re-export of this software is controlled for export purposes by the U.S. Government. By accepting this software and/or documentation, licensee agrees to comply with all U.S. and foreign export laws and regulations as they relate to software and related documentation. Licensee will not export or re-export outside the United States software or documentation, whether directly or indirectly, to any Prohibited Party and will not cause, approve or otherwise intentionally facilitate others in so doing. A Prohibited Party includes: a party in a U.S. embargoed country or country the United States has named as a supporter of international terrorism; a party involved in proliferation; a party identified by the U.S. Government as a Denied Party; a party named on the U.S. Government's Specially Designated Nationals (SDN) List; a party prohibited from participation in export or re-export transactions by a U.S. Government General Order; a party listed by the U.S. Government's Office of Foreign Assets Control as ineligible to participate in transactions subject to U.S. jurisdiction; or any party that licensee knows or has reason to know has violated or plans to violate U.S. or foreign export laws or regulations. Licensee shall ensure that each of its software users complies with U.S. and foreign export laws and regulations as they relate to software and related documentation.

**Copyright and Trademark Notices.** Copyright © 2016 SailPoint Technologies, Inc. All Rights Reserved. All logos, text, content, including underlying HTML code, designs, and graphics used and/or depicted on these written materials or in this Internet web site are protected under United States and international copyright and trademark laws and treaties, and may not be used or reproduced without the prior express written permission of SailPoint Technologies, Inc.

“SailPoint Technologies & Design,” “IdentityIQ,” “IdentityNow,” “AccessIQ,” “Identity Cube,” “Managing the Business of Identity” and the SailPoint logo are registered trademarks of SailPoint Technologies, Inc. “SecurityIQ,” “SailPoint,” “Identity is Everything” and “The Power of Identity” are trademarks of SailPoint Technologies, Inc. None of the foregoing marks may be used without the prior express written permission of SailPoint Technologies, Inc. All other trademarks shown herein are owned by the respective companies or persons indicated.

# Revision History

---

The following table describes the revision history of *SailPoint Direct Connectors Administration and Configuration Guide* for version 7.1:

Revision	Version	Description
1	6.3	First draft
2	6.3 Patch 3	Includes the following important changes: <ul style="list-style-type: none"><li>• New connector: Cerner Connector</li><li>• Session Management in ServiceNow Connector</li><li>• SharePoint Stealthbits</li></ul>
3	6.4	Includes the following important changes: <ul style="list-style-type: none"><li>• New connector: Duo Connector</li><li>• Connector Categorization</li><li>• Connector updates and addition to the following: Active Directory, Oracle HRMS, PeopleSoft HRMS, Salesforce, Oracle DB, Oracle EBS, JDBC, Delimited File, SQL Loader and Duo Security.</li></ul>
4	6.4 Patch 4	Includes the following important changes: <ul style="list-style-type: none"><li>• ServiceNow Fuji support</li><li>• Additions to LDAP Pre-requisites</li><li>• Oracle HRMS updates for DEFINER and INVOKER rights</li></ul>
5	6.4 Patch 5	Includes the following important changes: <ul style="list-style-type: none"><li>• Administrator Permissions updated for Oracle E-Business Suite</li><li>• Optional attributes added under Schema Attributes of Workday Connector</li><li>• SAP Connector Permission change</li></ul>
6	7.0	Includes the following important changes: <ul style="list-style-type: none"><li>• Changes in Workday Connector</li><li>• Azure Active Directory Connector</li><li>• Administrator permissions of Oracle E-Business Connector</li><li>• Deprecating support for Microsoft SharePoint Connector, Microsoft Office 365 Connector, and Microsoft Office 365 Exchange Online Connector</li></ul>
7	7.0 Patch 2	Includes the following important changes: <ul style="list-style-type: none"><li>• Deprecating support for Rule Based Logical Connector</li><li>• AUTL support for IBM i Connector</li><li>• Geneva support for ServiceNow Connector</li><li>• Support for SAP NetWeaver version 7.5</li><li>• Support for PeopleTool 8.55</li><li>• Support for Tenrox Software version 2015 and 2014</li></ul>

Revision	Version	Description
8	7.1	<p>Includes the following important changes:</p> <ul style="list-style-type: none"> <li>• Deprecating support for Tenrox, Rally and ALES Connector</li> <li>• IQService: Support for Windows FIPS compliant mode</li> <li>• New Connectors: SCIM 2, SAP HANA, Web Services and RACF via LDAP</li> <li>• Workday Connector Enhancements: <ul style="list-style-type: none"> <li>- It is a Standard Deployment Connector now</li> <li>- Future data</li> <li>- Delta Aggregation</li> <li>- Extensible account schema including schema attributes</li> </ul> </li> <li>• Administrator Permissions changes for SAP HANA Connector</li> <li>• Deleted the Password Interceptor appendix from the guide. For more information about Password Interceptor for Active Directory and IBM i, see <a href="https://community.sailpoint.com/docs/DOC-3257">https://community.sailpoint.com/docs/DOC-3257</a></li> <li>• Multiple group support in Tivoli, SunOne, and OpenLDAP LDAP Connectors</li> <li>• RSA Connector: Support for Extended Attributes such as mobile number</li> <li>• SAP HR/HCM Connector Enhancements <ul style="list-style-type: none"> <li>- support for future hire and future data</li> <li>- enhancement for supporting different models to detect employee's manager</li> <li>- Administrator permissions changes</li> </ul> </li> </ul>

# Table of Contents

---

<b>Revision History .....</b>	<b>3</b>
<b>Overview .....</b>	<b>1</b>
Connector basics .....	1
Connector Licensing .....	1
Working of Connectors .....	1
Agent Connectors .....	2
Retryable mechanism .....	2
What are Direct Connectors .....	3
Application Types for Connectors .....	3
Viewing the available connectors .....	7
Connector selection .....	7
<b>Standard Deployment Connectors .....</b>	<b>9</b>
<b>Chapter 1: SailPoint Active Directory Connector .....</b>	<b>11</b>
Overview .....	11
Supported features .....	12
Supported Managed System .....	13
Pre-requisites .....	13
Administrator permissions .....	14
Configuration parameters .....	14
Additional configuration parameter .....	15
Configuring Domain Settings .....	16
Configuring searchDNs .....	16
Schema attributes .....	17
Account attributes .....	17
Group attributes .....	23
Provisioning Policy attributes .....	24
Active Directory Recycle Bin .....	28
Pre-requisites .....	28
Configuring Recycle Bin .....	29
Additional information .....	29
Unstructured Target Collector .....	29
Troubleshooting .....	32
<b>Chapter 2: SailPoint AIX Connector .....</b>	<b>35</b>
Overview .....	35
Supported features .....	35
Supported Managed Systems .....	36
Pre-requisites .....	36
Administrator permissions .....	36
Configuration parameters .....	37
Additional configuration parameters for SSH configuration .....	37
Public key authentication configuration .....	38
Schema attributes .....	38
Account attributes .....	38
Group attributes .....	44
Provisioning policy attributes .....	44

Account attributes .....	44
Group attributes .....	45
Additional information .....	45
Unstructured Target Collector .....	45
Troubleshooting .....	46
<b>Chapter 3: SailPoint Azure Active Directory Connector .....</b>	<b>49</b>
Overview .....	49
Supported features .....	50
Pre-requisites .....	50
Administrator permissions .....	51
Configuration parameters .....	52
Additional configuration parameters .....	52
Schema attributes .....	53
Account attributes .....	53
Group attributes .....	54
Provisioning Policy attributes .....	55
Create Account Policy .....	55
Create Group Policy .....	56
Update Group Policy .....	57
Additional information .....	57
Managing licenses .....	57
Connector Reconfigure .....	57
<b>Chapter 4: SailPoint BMC Remedy Connector .....</b>	<b>59</b>
Overview .....	59
Supported features .....	59
Supported Managed Systems .....	60
Pre-requisites .....	60
Administrator permission .....	60
Configuration parameters .....	60
Schema attributes .....	60
Account attributes .....	60
Group attributes .....	61
Provisioning policy attributes .....	62
Create account attributes .....	62
Create group attributes .....	62
Update policies .....	62
Additional information .....	62
Enable/Disable Account .....	63
Troubleshooting .....	63
<b>Chapter 5: SailPoint BMC Remedy IT Service Management Suite Connector .....</b>	<b>65</b>
Overview .....	65
Supported features .....	65
Supported Managed Systems .....	66
Pre-requisites .....	66
Administrator permission .....	66
Configuration parameters .....	66
Schema attributes .....	67
Account attributes .....	67
Group attributes .....	68
Provisioning policy attributes .....	68

Create account attributes .....	68
Create group attributes .....	69
Update policies .....	70
Additional information .....	70
Enable/Disable Account .....	70
Add Entitlement operation for ITSM .....	70
Troubleshooting .....	71
<b>Chapter 6: SailPoint DB2 Windows Connector .....</b>	<b>73</b>
Overview .....	73
Supported features .....	74
Supported Managed Systems .....	74
Pre-requisites .....	74
Administrator permissions .....	74
Configuration parameters .....	75
Schema Attributes .....	75
Account attributes .....	75
Roles attributes .....	76
Provisioning Policy attributes .....	77
Additional information .....	77
Create user .....	78
Delete user .....	78
Delete Role .....	78
Troubleshooting .....	78
<b>Chapter 7: SailPoint Delimited File Connector .....</b>	<b>81</b>
Overview .....	81
Configuration parameters .....	81
Schema attributes .....	83
<b>Chapter 8: SailPoint JDBC Connector .....</b>	<b>85</b>
Overview .....	85
Supported features .....	85
Supported Managed Systems .....	86
Pre-requisites .....	86
Administrator permissions .....	86
Configuration parameters .....	86
Additional configuration parameters .....	88
Schema Attributes .....	89
Troubleshooting .....	89
<b>Chapter 9: SailPoint Jive Connector .....</b>	<b>91</b>
Overview .....	91
Supported features .....	91
Pre-requisites .....	92
Administrator permission .....	92
Configuration parameters .....	92
Schema attributes .....	92
Account attributes .....	92
Group attributes .....	94
Provisioning Policy attributes .....	94
Create account attributes .....	94
Create group attributes .....	95

Additional information .....	95
Configuration settings .....	95
Troubleshooting .....	95
<b>Chapter 10: SailPoint LDAP Connector .....</b>	<b>97</b>
Overview .....	97
Supported features .....	97
Supported Managed Systems .....	98
Pre-requisites .....	100
Administrator permissions .....	100
Configuration parameters .....	100
Additional configuration parameter .....	101
Configuring Account Search Scope .....	101
Configuring Group Search Scope .....	103
Schema attributes .....	103
Account attributes .....	103
Group attributes .....	107
posixgroup and nisNetgroup Attributes .....	108
Group Membership attribute .....	109
Group Entitlement attribute .....	109
Provisioning Policy attributes .....	110
Configuring group provisioning policy for new group .....	111
Additional information .....	111
Adding additional group types .....	111
Managing Revoke-Restore for SunOne .....	113
Using Novell eDirectory as a Pass-through Authentication Source .....	113
Troubleshooting .....	113
<b>Chapter 11: SailPoint LDIF Connector .....</b>	<b>115</b>
Overview .....	115
Configuration parameters .....	115
Schema Attributes .....	116
Account attributes .....	116
Group attributes .....	120
<b>Chapter 12: SailPoint Logical Connector .....</b>	<b>121</b>
Overview .....	121
Configuration parameters .....	121
Schema attributes .....	121
Additional information .....	122
Logical Connector - Tiers Tab .....	122
Defining Logical Connectors .....	124
Logical Application Filtering .....	124
<b>Chapter 13: SailPoint Lotus Domino Connector .....</b>	<b>127</b>
Overview .....	127
Supported features .....	127
Supported Managed Systems .....	128
Pre-requisites .....	128
Administrator permissions .....	129
Configuration parameters .....	129
Additional configuration parameters .....	130
Schema attributes .....	130

Account attributes .....	.130
Group attributes .....	.132
Provisioning policy attributes .....	.133
Create account attributes .....	.133
Create group attributes .....	.136
Update policies .....	.136
Additional information .....	.138
ID Vault functionalities .....	.138
Password management .....	.138
Troubleshooting .....	.139
<b>Chapter 14: SailPoint Linux Connector .....</b>	<b>141</b>
Overview .....	.141
Supported features .....	.141
Supported Managed Systems .....	.142
Pre-requisites .....	.142
Administrator permissions .....	.142
Configuration parameters .....	.143
Additional configuration parameters for SSH configuration .....	.143
Public key authentication configuration .....	.144
Schema attributes .....	.144
Account attributes .....	.144
Group attributes .....	.145
Provisioning policy attributes .....	.145
Account attributes .....	.146
Group attributes .....	.146
Additional information .....	.147
Unstructured Target Collector .....	.147
Troubleshooting .....	.148
<b>Chapter 15: SailPoint Mainframe Connector .....</b>	<b>153</b>
Overview .....	.153
Configuration parameters .....	.153
Schema attributes .....	.154
Account attributes .....	.154
<b>Chapter 16: SailPoint Microsoft SQL Server .....</b>	<b>155</b>
Overview .....	.155
Supported features .....	.156
Supported Managed Systems .....	.156
Pre-requisites .....	.156
Administrator permissions .....	.156
Configuration parameters .....	.159
Schema attributes .....	.159
Account attributes .....	.159
Group attributes .....	.160
Provisioning Policy attributes .....	.161
Additional information .....	.161
Delete login .....	.161
Direct permission .....	.162
Identity and Entitlement representation .....	.162
Troubleshooting .....	.163

<b>Chapter 17: SailPoint Oracle Connector .....</b>	<b>165</b>
Overview .....	165
Supported features .....	165
Supported Managed Systems .....	166
Pre-requisites .....	166
Administrator permissions .....	166
Configuration parameters .....	167
Additional configuration parameter .....	168
Schema attributes .....	168
Account attributes .....	168
Group attributes .....	169
Provisioning policy attributes .....	169
Troubleshooting .....	170
<b>Chapter 18: SailPoint PeopleSoft Connector .....</b>	<b>171</b>
Overview .....	171
Supported features .....	171
Supported Managed Systems .....	172
Pre-requisites .....	172
Administrator permission .....	172
Configuration parameters .....	172
Schema attributes .....	173
Account attributes .....	173
Group attributes .....	175
Additional information .....	175
Creating the Component Interfaces .....	175
Creating the Component interface jar file .....	175
Configuring the Component Interface Security .....	176
Troubleshooting .....	177
<b>Chapter 19: SailPoint RACF Connector .....</b>	<b>179</b>
Overview .....	179
Supported features .....	179
Configuration parameters .....	179
Schema Attributes .....	181
Account attributes .....	181
Group attributes .....	185
<b>Chapter 20: SailPoint RACF LDAP Connector .....</b>	<b>187</b>
Overview .....	187
Supported features .....	187
Supported Managed Systems .....	188
Pre-requisites .....	188
Administrator permissions .....	189
Configuration parameters .....	189
Schema Attributes .....	190
Account attributes .....	190
Group attributes .....	192
Provisioning Policy Attributes .....	193
Additional information .....	194
Support for PassPhrase .....	194
Support for Connection Attributes .....	194
Troubleshooting .....	194

<b>Chapter 21: SailPoint Salesforce Connector .....</b>	<b>195</b>
Overview .....	195
Supported features .....	195
Administrator permissions .....	196
Configuration parameters .....	196
Additional configuration parameters .....	197
Schema attributes .....	198
Account attributes .....	198
Profile attributes .....	200
Provisioning Policy attributes .....	201
Troubleshooting .....	202
<b>Chapter 22: SailPoint SAP Portal-User Management Web Service Connector ....</b>	<b>207</b>
Overview .....	207
Supported features .....	207
Supported Managed Systems .....	208
Pre-requisite .....	208
Administrator permission .....	208
Configuration parameters .....	208
Schema attributes .....	209
Account attributes .....	209
Group attributes .....	210
Provisioning Policy attributes .....	211
Create account attributes .....	211
Create Group attributes .....	212
Additional information .....	212
Undeploy .sda file .....	212
<b>Chapter 23: SailPoint SAP HR/HCM Connector .....</b>	<b>213</b>
Overview .....	213
Supported features .....	213
Supported Managed Systems .....	214
Pre-requisites .....	214
Administrator permissions .....	214
Configuration parameters .....	217
Schema Attributes .....	219
Account attributes .....	219
Additional information .....	225
Upgraded Application .....	225
Troubleshooting .....	226
<b>Chapter 24: SailPoint SAP HANA Connector .....</b>	<b>229</b>
Overview .....	229
Supported features .....	229
Supported Managed Systems .....	230
Pre-requisites .....	230
Administrator permissions .....	230
Configuration parameters .....	231
Schema Attributes .....	232
Account attributes .....	232
Group attributes .....	233
Provisioning Policy attributes .....	233
Create account attributes .....	234

Additional information .....	234
Enabling SSL connection to SAP HANA database through IdentityIQ .....	234
Troubleshooting .....	234
<b>Chapter 25: SailPoint System for Cross-Domain Identity Management Connector 2.0</b>	<b>237</b>
Overview .....	237
Supported features .....	237
Administrator permissions .....	238
Supported Managed System .....	238
Configuration parameters .....	238
Schema attributes .....	238
Provisioning Policy attributes .....	239
Create account attributes .....	239
Update account attributes .....	239
Provisioning of extended attributes .....	239
Troubleshooting .....	240
<b>Chapter 26: SailPoint ServiceNow Connector .....</b>	<b>241</b>
Overview .....	241
Supported features .....	241
Supported Managed System .....	242
Pre-requisites .....	242
User permissions .....	243
Configuration parameters .....	243
Schema attributes .....	244
Account attributes .....	245
Group attributes .....	246
Role attributes .....	247
Provisioning Policy attributes .....	248
Additional information .....	249
Upgrade .....	249
Session management .....	249
Troubleshooting .....	250
<b>Chapter 27: SailPoint Siebel Connector .....</b>	<b>253</b>
Overview .....	253
Supported features .....	253
Supported Managed Systems .....	254
Pre-requisites .....	254
Administrator permission .....	254
Configuration parameters .....	254
Schema attributes .....	255
Account attributes .....	256
Account Group attributes .....	256
Adding new custom attributes in schema .....	257
Provisioning policy attributes .....	257
Troubleshooting .....	258
<b>Chapter 28: SailPoint Solaris Connector .....</b>	<b>259</b>
Overview .....	259
Supported features .....	259
Supported Managed Systems .....	260

Pre-requisites .....	.260
Administrator permissions .....	.260
Configuration parameters .....	.261
Additional configuration parameters for SSH configuration .....	.261
Public key authentication configuration .....	.262
Schema attributes .....	.262
Account attributes .....	.262
Group attributes .....	.264
Provisioning policy attributes .....	.264
Account attributes .....	.264
Group attributes .....	.265
Additional information .....	.266
Unstructured Target Collector .....	.266
Troubleshooting .....	.267
<b>Chapter 29: SailPoint SQL Loader Connector .....</b>	<b>271</b>
Overview .....	.271
Supported features .....	.271
Supported Managed Systems .....	.272
Administrator permissions .....	.272
Configuration parameters .....	.272
Schema Attributes .....	.274
Troubleshooting .....	.274
<b>Chapter 30: SailPoint Sybase Connector .....</b>	<b>277</b>
Overview .....	.277
Supported features .....	.278
Supported Managed Systems .....	.278
Pre-requisites .....	.278
Administrator permissions .....	.279
Configuration parameters .....	.279
Schema attributes .....	.280
Account attributes .....	.280
Group attributes .....	.281
Provisioning policy attributes .....	.281
Additional information .....	.282
Identity and Entitlement representation .....	.282
Troubleshooting .....	.282
<b>Chapter 31: SailPoint Tivoli Access Manager Connector .....</b>	<b>283</b>
Overview .....	.283
Supported features .....	.283
Supported Managed System .....	.284
Pre-requisites .....	.284
Configuration parameters .....	.286
Schema attributes .....	.286
Account attributes .....	.286
Group attributes .....	.287
Provisioning Policy attributes .....	.287
Create account attributes .....	.287
Create group attributes .....	.288
Additional information .....	.288
Unstructured Target Collector .....	.288

Troubleshooting .....	.289
<b>Chapter 32: SailPoint Top Secret Connector .....</b>	<b>291</b>
Overview .....	.291
Supported features .....	.291
Configuration parameters .....	.292
Schema Attributes .....	.293
<b>Chapter 33: SailPoint UNIX Connector .....</b>	<b>305</b>
Overview .....	.305
Supported features .....	.305
Configuration parameters .....	.305
Schema attributes .....	.306
Account attributes .....	.306
Group attributes .....	.306
<b>Chapter 34: SailPoint Web Services Connector .....</b>	<b>309</b>
Overview .....	.309
Supported features .....	.309
Supported Managed Systems .....	.310
Pre-requisites .....	.310
Administrator permissions .....	.310
Configuration parameters .....	.310
(General Settings) Basic configuration parameters .....	.310
(Connector Operations) Operation specific configuration parameters .....	.311
Schema attributes .....	.312
Additional information .....	.312
Pagination .....	.312
Configuration for Response .....	.315
Configuration for Multiple endpoints .....	.316
Other Operations .....	.317
<b>Chapter 35: SailPoint Windows Local Connector .....</b>	<b>321</b>
Overview .....	.321
Supported features .....	.321
Supported Managed Systems .....	.322
Pre-requisites .....	.322
Administrator permissions .....	.322
Configuration parameters .....	.322
Additional configuration parameters .....	.323
Schema attributes .....	.323
Account attributes .....	.323
Group attributes .....	.324
Provisioning Policy attributes .....	.325
Install and register IQService .....	.325
Additional information .....	.326
Unstructured Target Collector .....	.326
Troubleshooting .....	.327
<b>Chapter 36: SailPoint Workday Connector .....</b>	<b>329</b>
Overview .....	.329
Supported features .....	.329
Prerequisites .....	.330

Administrator permissions .....	.330
Configuration parameters .....	.332
Configuring an Integration System in Workday .....	.333
Workday Worker Custom fields .....	.334
Schema attributes .....	.336
Account attributes .....	.336
Troubleshooting .....	.343
<b>Chapter 37: SailPoint XML Connector .....</b>	<b>347</b>
Overview .....	.347
Supported features .....	.347
Configuration parameters .....	.347
Additional information .....	.348
1 - Using XML Schema Definition (XSD) .....	.348
2 - Using Document Type Definition (DTD) .....	.349
<b>Assisted Deployment Connectors .....</b>	<b>363</b>
<b>Chapter 38: SailPoint Amazon Web Services Identity and Access Management Connector .....</b>	<b>351</b>
Overview .....	.351
Supported features .....	.352
Pre-requisites .....	.352
Administrator permissions .....	.353
Schema attributes .....	.357
Account schema .....	.357
Group schema .....	.357
Provisioning Policy attributes .....	.358
Account .....	.358
Account-Group .....	.358
Additional information .....	.358
Amazon Web Services Identity and Access Management API's .....	.358
Troubleshooting .....	.360
<b>Chapter 39: SailPoint Box Connector .....</b>	<b>365</b>
Overview .....	.365
Supported features .....	.365
Supported Managed Systems .....	.366
Pre-requisites .....	.366
Administrator permissions .....	.366
Configuration parameter .....	.366
Schema attributes .....	.366
Account attributes .....	.367
Group attributes .....	.367
Provisioning Policy attributes .....	.368
Troubleshooting .....	.368
<b>Chapter 40: SailPoint CyberArk Connector .....</b>	<b>371</b>
Overview .....	.371
Supported features .....	.371
Pre-requisites .....	.371
Configuration parameters .....	.372
Schema attributes .....	.373

Account attributes .....	.373
Additional information .....	.373
Direct permission .....	.373
<b>Chapter 41: SailPoint Duo Connector .....</b>	<b>375</b>
Overview .....	375
Supported features .....	.375
Pre-requisites .....	.375
Administrator permissions .....	.376
Configuration parameters .....	.376
Schema Attributes .....	.376
Account attributes .....	.376
Group attributes .....	.377
Provisioning Policy attributes .....	.378
Account attributes .....	.378
Behavioral changes .....	.379
<b>Chapter 42: SailPoint Dropbox Connector .....</b>	<b>381</b>
Overview .....	381
Supported features .....	.381
Supported Managed System .....	.382
Pre-requisites .....	.382
Administrator permissions .....	.382
Configuration parameters .....	.382
Schema attributes .....	.382
Account attributes .....	.383
Group attributes .....	.383
Provisioning Policy attributes .....	.383
Account attributes .....	.383
Group attributes .....	.384
Delete Provisioning Policy .....	.384
<b>Chapter 43: SailPoint Google Apps Connector .....</b>	<b>387</b>
Overview .....	387
Supported features .....	.388
Pre-requisites .....	.388
Administrator permissions .....	.389
Configuration parameters .....	.389
Schema attributes .....	.389
Account attributes .....	.389
Group attributes .....	.390
Provisioning Policy attributes .....	.394
Troubleshooting .....	.399
<b>Chapter 44: SailPoint GoToMeeting Connector .....</b>	<b>403</b>
Overview .....	403
Supported features .....	.403
Pre-requisites .....	.403
Administrator permissions .....	.404
Configuration parameter .....	.404
Schema attributes .....	.404
Account attributes .....	.404
Group attributes .....	.404

Provisioning Policy attributes .....	405
<b>Chapter 45: SailPoint IBM i Connector .....</b>	<b>407</b>
Overview .....	407
Supported features .....	408
Supported Managed Systems .....	409
Pre-requisites .....	409
Administrator permissions .....	409
Configuration parameters .....	410
Schema Attributes .....	410
Account and Account - Group attributes .....	410
Provisioning policy attributes .....	412
Additional information .....	413
Direct Permissions .....	413
Upgrade Consideration .....	414
Create SSL Communication between IdentityIQ and IBM i system .....	414
<b>Chapter 46: SailPoint Microsoft SharePoint Server Connector .....</b>	<b>415</b>
Overview .....	415
Supported features .....	415
Supported Managed system .....	416
Pre-requisites .....	416
Application Account permissions .....	417
Configuration parameters .....	417
Additional configuration parameter .....	418
Schema attributes .....	418
Account attributes .....	418
Group attributes .....	418
Provisioning Policy attributes .....	419
Additional information .....	420
Certifications .....	420
Troubleshooting .....	420
<b>Chapter 47: SailPoint Microsoft SharePoint Online Connector .....</b>	<b>423</b>
Overview .....	423
Supported features .....	423
Prerequisites .....	424
Administrator permissions .....	425
Configuration parameters .....	425
Schema attributes .....	425
Account attributes .....	426
Group attributes .....	426
Provisioning Policy attributes .....	426
Additional information .....	427
Unstructured Target Collector .....	427
<b>Chapter 48: SailPoint Microsoft Project Server Connector .....</b>	<b>429</b>
Overview .....	429
Supported features .....	429
Supported Managed system .....	430
Pre-requisites .....	430
Administrator permissions .....	430
Configuration parameters .....	430

Schema attributes .....	431
Account attributes .....	431
Group attributes .....	431
Provisioning Policy attributes .....	432
Troubleshooting .....	432
<b>Chapter 49: SailPoint NetSuite Connector .....</b>	<b>435</b>
Overview .....	435
Supported features .....	436
Supported Managed Systems .....	436
Administrator permissions .....	436
Configuration parameters .....	437
Schema attributes .....	437
Account attributes .....	437
Group attributes .....	438
Schema extension and custom attributes .....	438
Provisioning Policy attributes .....	439
Additional information .....	440
NetSuite Application Program Interface (API) .....	440
<b>Chapter 50: SailPoint Oracle HRMS Connector .....</b>	<b>441</b>
Overview .....	441
Supported features .....	441
Supported Managed Systems .....	441
Pre-requisites .....	441
Administrator permissions .....	442
Configuration parameters .....	444
Schema attributes .....	444
Account attributes .....	444
<b>Chapter 51: SailPoint Oracle E-Business Suite Connector .....</b>	<b>447</b>
Overview .....	447
Supported features .....	448
Supported Managed Systems .....	448
Pre-requisites .....	448
Administrator permissions .....	448
Configuration parameters .....	451
Additional configuration parameter .....	451
Schema attributes .....	452
Account attributes .....	452
Group attributes .....	453
Provisioning Policy attributes .....	454
Create account attributes .....	454
Delete account attributes .....	454
Create group attributes .....	454
Deleting Group (Responsibility) .....	455
Troubleshooting .....	455
<b>Chapter 52: SailPoint PeopleSoft HCM Database Connector .....</b>	<b>457</b>
Overview .....	457
Supported features .....	457
Supported Managed Systems .....	458
Pre-requisites .....	458

Administrator permission .....	.458
Configuration parameters .....	.459
Schema attributes .....	.460
Account attributes .....	.460
Additional information .....	.462
Configuring Component Interface Security .....	.462
Creating PeopleSoft HRMS Jar File .....	.462
Troubleshooting .....	.464
<b>Chapter 53: SailPoint RSA Authentication Manager Connector .....</b>	<b>465</b>
Overview .....	.465
Supported features .....	.466
Supported Managed Systems .....	.466
Pre-requisites .....	.466
Administrator permissions .....	.467
RSA Token PIN Reset .....	.467
Configuration parameters .....	.468
Schema attributes .....	.469
Account attributes .....	.469
Group attributes .....	.470
Provisioning Policy attributes .....	.470
Additional information .....	.471
Active Directory configured as an identity source .....	.471
<b>Chapter 54: SailPoint Remedyforce Connector .....</b>	<b>473</b>
Overview .....	.473
Supported features .....	.473
Configuration parameters .....	.474
Schema attributes .....	.475
Account attributes .....	.476
Profile attributes .....	.478
Provisioning Policy attributes .....	.478
Troubleshooting .....	.479
<b>Chapter 55: SailPoint SAP Connector .....</b>	<b>481</b>
Overview .....	.481
Supported features .....	.481
Supported Managed Systems .....	.482
Pre-requisites .....	.483
Administrator permissions .....	.483
Configuration parameters .....	.487
Schema attributes .....	.489
Account attributes .....	.489
Group attributes .....	.493
Schema extension and custom attributes .....	.494
Upgrade considerations .....	.494
Provisioning Policy attributes .....	.494
Create account attributes .....	.494
Additional information .....	.495
Entitlement validity period .....	.495
CUA support .....	.495
Entitlement Data .....	.495
Password Change .....	.495

Logon and Communication Language attributes .....	.496
Troubleshooting .....	.497
<b>Chapter 56: SailPoint System for Cross-Domain Identity Management Connector</b>	<b>501</b>
Overview .....	501
Supported features .....	501
Administrator permissions .....	502
Configuration parameters .....	502
Additional configuration parameters .....	502
Schema attributes .....	504
Account attributes .....	504
Group attributes .....	506
Provisioning Policy attributes .....	506
Create account attributes .....	507
Update group attributes .....	507
Troubleshooting .....	507
<b>Chapter 57: SailPoint WebEx Connector</b> .....	<b>509</b>
Overview .....	509
Supported features .....	509
Pre-requisites .....	510
Administrator permissions .....	510
Configuration parameters .....	510
Schema attributes .....	510
Account attributes .....	510
Group attributes .....	512
Provisioning Policy attributes .....	513
<b>Chapter 58: SailPoint Yammer Connector</b> .....	<b>515</b>
Overview .....	515
Supported features .....	515
Pre-requisites .....	515
Administrator permissions .....	516
Configuration parameter .....	516
Schema attributes .....	516
Account attributes .....	516
Group attributes .....	517
<b>Collaborative Deployment Connectors</b> .....	<b>519</b>
<b>Chapter 59: SailPoint Cerner Connector</b> .....	<b>521</b>
Overview .....	521
Supported features .....	521
Pre-requisites .....	522
Configuration parameters .....	522
Schema attributes .....	523
Account attributes .....	523
Group attributes .....	524
Provisioning Policy attributes .....	524
Troubleshooting .....	524
<b>Chapter 60: SailPoint Epic Connector</b> .....	<b>527</b>
Overview .....	527

Supported features .....	.527
Supported Managed System .....	.528
Pre-requisites .....	.528
Administrator permissions .....	.528
Configuration parameters .....	.528
Schema Attributes .....	.529
Account attributes .....	.529
Group attributes .....	.530
Provisioning Policy attributes .....	.530
Troubleshooting .....	.532
<b>Chapter 61: SailPoint GE Centricity Connector .....</b>	<b>535</b>
Overview .....	.535
Supported features .....	.536
Prerequisites .....	.536
Administrator permissions .....	.536
Configuration parameters .....	.536
Additional configuration parameters .....	.537
Schema attributes .....	.537
Account attributes .....	.537
Group attributes .....	.538
Provisioning Policy attributes .....	.538
Troubleshooting .....	.539
<b>Appendix .....</b>	<b>541</b>
<b>Appendix A: Delta Aggregation .....</b>	<b>543</b>
Overview .....	.543
Delta aggregation for Microsoft Active Directory, ADAM, SunOne and Tivoli .....	.543
Configuring server for Delta Aggregation .....	.545
Testing Delta Aggregation .....	.545
Delta aggregation for JDBC .....	.546
Delta aggregation for Lotus Domino .....	.547
Pre-requisites .....	.547
Delta aggregation for SAP .....	.547
Supported attributes .....	.548
Importing the BAPI's .....	.548
Verification .....	.550
<b>Appendix B: Component Interface .....</b>	<b>551</b>
Creating component interface for Peoplesoft financials .....	.551
Basic structure of Custom Component (CI) from USERMAINT component for Users .....	.551
Basic structure of Custom Component (CI) from ROLEMAINT component for Roles .....	.556
Basic structure of Custom Component (CI) from PURGE_USR_PROFILE component for Delete User .....	.558
Basic structure of Component Interface (CI) from PURGE_ROLEDEFN component for Delete Role .....	.559
Deleting the component interface .....	.560
<b>Appendix C: Partitioning Aggregation .....</b>	<b>563</b>
Overview .....	.563
Partitioning Aggregation for ERP Connectors (SAP and PeopleSoft) .....	.564
Partitioning Aggregation for JDBC Connector .....	.564
Partitioning Aggregation for Active Directory and LDAP Connectors .....	.565
Partitioning Aggregation for Delimited and CyberArk Connectors .....	.566
Partitioning Aggregation for IBM i Connector .....	.566
Partitioning Aggregation for GoogleApps .....	.566

Partitioning Aggregation for Cerner .....	567
Partitioning Aggregation for Tivoli Access Manager .....	568
Partitioning Aggregation for Azure Active Directory Connector .....	568
Partitioning Aggregation for RACF LDAP Connector .....	569
<b>Appendix D: Before and After Provisioning Action .....</b>	<b>571</b>
Overview .....	571
Before and After Provisioning Action for AIX/Linux/Solaris Connectors .....	571
Pre-requisite .....	571
Creating Before and After Provisioning Action .....	571
Before and After Provisioning Action for IBM i Connector .....	573
Pre-requisites .....	573
Creating CL scripts .....	573
<b>Appendix E: IQService.....</b>	<b>577</b>
Install and register the IQService for Windows .....	577
Installing and registering IQService .....	578
IQService Public Key Exchange Task .....	579
IQService Before/After Scripts .....	580
Writing a script .....	580
Creating a Rule .....	582

# Overview

---

The following topics are discussed:

Connector basics .....	1
Retryable mechanism .....	2
What are Direct Connectors .....	3
Application Types for Connectors.....	3
Viewing the available connectors.....	7

Connectivity is critical to successful IAM deployments. SailPoint is committed to providing design, configuration, troubleshooting and best practice information to deploy and maintain connectivity to target systems. SailPoint has modified the structure of this document to aid customers and partner deployments. The focus of this document is product configuration and integration. For more details on design, troubleshooting and deployment best practices, refer to the Connector and Integration Deployment Center in Compass.

This document describes the different types of connectors available for integration with external applications.

## Connector basics

---

There are several different types of connectors. Connectors are commonly grouped by the ways in which they can communicate. There are:

- read-only connectors that can only communicate data from an external application (Governance)
- read-write connectors that can read data from external applications and write data out to them (Gateway and Direct)

## Connector Licensing

---

Customers who licensed IdentityIQ on or after July 15, 2013 are entitled to all IdentityIQ connectors for aggregation, provisioning and password management use cases.

Customers who licensed IdentityIQ before July 15, 2013 are entitled to use connectors for reading data (aggregation). However, entitlement to provisioning and password management functionality through the IdentityIQ connectors (whether Direct, Gateway or Agent based) requires the purchase of the SailPoint Provisioning Engine.

## Working of Connectors

---

This section describes how the connectors work. The Direct Connectors are of the following types:

- **Read/Write Connectors:** These connectors have read and write capabilities on external application and allow data to send in both directions.
- **Read only Connectors:** These connectors are very simple in design, they make a direct read-only connection to the external application through the connection parameters specified on the Application Definition.

## Retryable mechanism

---

### Account rename operation

IdentityIQ does not fully support processing requests to Move/Rename accounts or groups on native system. Connectors supporting Move/Rename accounts via attributes update require customization in IdentityIQ to initiate such requests and update the native identity of the link once the request is successfully processed by the connector.

The Account Move/Rename requests must not be merged with any other requests/updates.

### Group provisioning

To enable group provisioning for existing application after upgrading to version 7.1, perform the following:

- Add **GROUP\_PROVISIONING** to the featureString from debug page
- Define **CreateGroup** and **EditGroup** provisioning policies.

For more information on the provisioning policies defined for the connector from `connectorRegistry.xml`, see the “Provisioning Policy” section of the respective connectors for the required attributes.

## Agent Connectors

---

The targeted systems for Agent connectors are centralized mainframe security systems; Agents are the simplest and most secure way to connect to those systems. Agents residing on mainframe security systems communicate through the Connector Gateway.

Following are the Agent Connectors:

- ACF2 Full
- RACF Full
- TopSecret Full

For more information, see *SailPoint Quick Reference Guide for Gateway Connectors*.

### Target permissions support (RACF, ACF2, and Top Secret)

The Target permissions feature is supported for Mainframe based connectors that include RACF, ACF2 and Top Secret.

For more information on target permissions, see *SailPoint Quick Reference Guide for Gateway Connectors*.

### Password Interceptor or Online Interceptor support (RACF, ACF2, and Top Secret)

Password Interception or Online Interceptor is not enabled by default. There are few steps that need to be performed in order to enable it on Gateway Connector and IdentityIQ.

For more information on enabling the Password Interceptor or Online Interceptor, see the “Settings for configuring Password Interceptor and Online Interceptor” section in the *SailPoint Quick Reference Guide for Gateway Connectors*.

## Retryable mechanism

---

For availing the advantage of some of the logic around retryable situations, add the retryable error messages list to the attributes map on an application. The **retryableErrors** entry is a list of strings through which the connector searches when it receives a message from the managed application. If one of the strings in the entry exists in the

error, the connector attempts to retry the connection. When the configured error string is not a part of the error message returned from the connector, then IdentityIQ will not attempt a retry.

Here is an example of this entry:

```
<entry key="retryableErrors">
  <value>
    <List>
      <String>Server is not operational</String>
    </List>
  </value>
</entry>
```

**Precaution:** Avoid using error messages which contain a date/time, sequence id, SM packets/messages, and so on, as these are very specific. Error codes or error message substrings would be good candidates for inclusion.

## What are Direct Connectors

---

- **Simple to configure and use:** Direct connectors are simple to configure, few configuration details required to use the connector and no extra steps to deploy agents on the end managed systems.
- **Less moving parts:** Direct connectors do not require Connector Gateway (CG), Connector Manager (CM), Provisioning Modules (PM) to be deployed to get the setup done. Installing and (Re)configuring each component is not required. Data caching and sequencing on transactions not required.
- **Increased performance:** The performance of direct connectors is improved compared to the old FULL or Gateway based connectors. It is recommended to move to direct connector to get maximum benefit of per transactions.
- **No single point of failure:** Earlier if one component failed in the connector model, then it required re-cycling of the Connector and Connector Gateway. Such issues do not exist in direct connector architecture.
- **Less hardware:** New direct connectors do not require any agent installation on end managed system or other computer. The overall hardware requirement for connectors setup is reduced due to this new architecture.

## Application Types for Connectors

---

The following table lists the application type of Connectors:

**Table 1—Application Types for Connectors**

Connector	Application Type	Details
<b>Standard Deployment Connectors</b>		
Active Directory	Active Directory - Direct	<a href="#">Chapter 1:SailPoint Active Directory Connector</a> 11
AIX	AIX - Direct	<a href="#">Chapter 2:SailPoint AIX Connector</a> 35
Azure Active Directory	Azure Active Directory	<a href="#">Chapter 3:SailPoint Azure Active Directory Connector</a> 49
Remedy	BMC Remedy - Direct	<a href="#">Chapter 4:SailPoint BMC Remedy Connector</a> 59

## Application Types for Connectors

**Table 1—Application Types for Connectors**

Connector	Application Type	Details
Remedy ITSM	BMC ITSM - Direct	<a href="#">Chapter 5:SailPoint BMC Remedy IT Service Management Suite Connector</a> 65
DB2 Windows	DB2 Windows - Direct	<a href="#">Chapter 6:SailPoint DB2 Windows Connector</a> 73
Delimited	DelimitedFile	<a href="#">Chapter 7:SailPoint Delimited File Connector</a> 81
JDBC	JDBC	<a href="#">Chapter 8:SailPoint JDBC Connector</a> 85
Jive	JIVE	<a href="#">Chapter 9:SailPoint Jive Connector</a> 91
LDAP	LDAP	<a href="#">Chapter 10:SailPoint LDAP Connector</a> 97
LDIF	LDIF	<a href="#">Chapter 11:SailPoint LDIF Connector</a> 115
Logical	Logical	<a href="#">Chapter 12:SailPoint Logical Connector</a> 121
Lotus Domino	IBM Lotus Domino - Direct	<a href="#">Chapter 13:SailPoint Lotus Domino Connector</a> 127
Linux	Linux - Direct	<a href="#">Chapter 14:SailPoint Linux Connector</a> 141
Mainframe	Mainframe	<a href="#">Chapter 15:SailPoint Mainframe Connector</a> 153
Microsoft SQL Server	Microsoft SQL Server - Direct	<a href="#">Chapter 16:SailPoint Microsoft SQL Server</a> 155
Oracle	Oracle Database - Direct	<a href="#">Chapter 17:SailPoint Oracle Connector</a> 165
PeopleSoft	PeopleSoft - Direct	<a href="#">Chapter 18:SailPoint PeopleSoft Connector</a> 171
RACF	RACF	<a href="#">Chapter 19:SailPoint RACF Connector</a> 179
RACF LDAP	RACF LDAP	<a href="#">Chapter 20:SailPoint RACF LDAP Connector</a> 187
Salesforce/Remedyforce	RemedyForce	<a href="#">Chapter 21:SailPoint Salesforce Connector</a> 195
SAP EP	SAP Portal - UMWebService	<a href="#">Chapter 22:SailPoint SAP Portal-User Management Web Service Connector</a> 207
SAP HR/HCM	SAP HR/HCM	<a href="#">Chapter 23:SailPoint SAP HR/HCM Connector</a> 213
SAP HANA	SAP HANA Database	<a href="#">Chapter 24:SailPoint SAP HANA Connector</a> 229

**Table 1—Application Types for Connectors**

Connector	Application Type	Details
System for Cross-Domain Identity Management 2.0	SCIM 2	<a href="#">Chapter 25:SailPoint System for Cross-Domain Identity Management Connector 2.0</a> 237
ServiceNow	ServiceNow	<a href="#">Chapter 26:SailPoint ServiceNow Connector</a> 241
Siebel	Siebel	<a href="#">Chapter 27:SailPoint Siebel Connector</a> 253
Solaris	Solaris - Direct	<a href="#">Chapter 28:SailPoint Solaris Connector</a> 259
SQL Loader	SQLLoader	<a href="#">Chapter 29:SailPoint SQL Loader Connector</a> 271
Sybase	Sybase - Direct	<a href="#">Chapter 30:SailPoint Sybase Connector</a> 277
Tivoli Access Manager	IBM Tivoli Access Manager	<a href="#">Chapter 31:SailPoint Tivoli Access Manager Connector</a> 283
Top Secret	TopSecret	<a href="#">Chapter 32:SailPoint Top Secret Connector</a> 291
UNIX	Unix	<a href="#">Chapter 33:SailPoint UNIX Connector</a> 305
Web Service	Web Services	<a href="#">Chapter 34:SailPoint Web Services Connector</a> 309
Windows Local	Windows Local - Direct	<a href="#">Chapter 35:SailPoint Windows Local Connector</a> 321
Workday	Workday	<a href="#">Chapter 36:SailPoint Workday Connector</a> 329
XML	XML	<a href="#">Chapter 37:SailPoint XML Connector</a> 347
<b>Assisted Deployment Connectors</b>		
Amazon Web Services Identity and Access Management	AWS IAM	<a href="#">Chapter 38:SailPoint Amazon Web Services Identity and Access Management Connector</a> 351
Box	Box	<a href="#">Chapter 39:SailPoint Box Connector</a> 365
CyberArk	CyberArk	<a href="#">Chapter 40:SailPoint CyberArk Connector</a> 371
Duo Connector	Duo	<a href="#">Chapter 41:SailPoint Duo Connector</a> 375
Google Apps	GoogleApps - Direct	<a href="#">Chapter 43:SailPoint Google Apps Connector</a> 387
GoToMeeting	GoToMeeting	<a href="#">Chapter 44:SailPoint GoToMeeting Connector</a> 403

## Application Types for Connectors

**Table 1—Application Types for Connectors**

Connector	Application Type	Details
IBM i	IBM i	<a href="#">Chapter 45:SailPoint IBM i Connector 407</a>
Microsoft SharePoint Server	Microsoft SharePoint Server	<a href="#">Chapter 46:SailPoint Microsoft SharePoint Server Connector 415</a>
Microsoft SharePoint Online	Microsoft SharePoint Online	<a href="#">Chapter 47:SailPoint Microsoft SharePoint Online Connector 423</a>
Microsoft Project Server 2013	Microsoft Project Server	<a href="#">Chapter 48:SailPoint Microsoft Project Server Connector 429</a>
NetSuite	NetSuite	<a href="#">Chapter 49:SailPoint NetSuite Connector 435</a>
Oracle HRMS	Oracle HRMS	<a href="#">Chapter 50:SailPoint Oracle HRMS Connector 441</a>
Oracle EBS	Oracle E-Business Suite	<a href="#">Chapter 51:SailPoint Oracle E-Business Suite Connector 447</a>
PeopleSoft HCM Database	PeopleSoft HCM Database	<a href="#">Chapter 52:SailPoint PeopleSoft HCM Database Connector 457</a>
RSA Authentication Manager	RSA Authentication Manager - Direct	<a href="#">Chapter 53:SailPoint RSA Authentication Manager Connector 465</a>
Remedyforce	Remedyforce	<a href="#">Chapter 54:SailPoint Remedyforce Connector 473</a>
SAP	SAP - Direct	<a href="#">Chapter 55:SailPoint SAP Connector 481</a>
System for Cross-Domain Identity Management	SCIM	<a href="#">Chapter 56:SailPoint System for Cross-Domain Identity Management Connector 501</a>
Webex	Webex	<a href="#">Chapter 57:SailPoint WebEx Connector 509</a>
Yammer	Yammer	<a href="#">Chapter 58:SailPoint Yammer Connector 515</a>
<b>Collaborative Deployment Connectors</b>		
Cerner	Cerner	<a href="#">Chapter 59:SailPoint Cerner Connector 521</a>
Epic	EPIC	<a href="#">Chapter 60:SailPoint Epic Connector 527</a>
GE Centricity	GE Centricity	<a href="#">Chapter 61:SailPoint GE Centricity Connector 535</a>

## Viewing the available connectors

---

Connectors may be added, removed, or modified in any release, including patch releases. Existing defined applications will continue to use the connector specified during their initial creation, and changes to the connector will not affect existing applications unless those changes are manually applied to the application definition. However, the **ConnectorRegistry** entry for the connectors does change with new releases. The list of available connectors, with their current set of available features, can be retrieved from the Connector Registry within the Debug Pages.

Select **Configuration** in the Objects list and click **List**, then select **ConnectorRegistry** to view the XML for all the connectors.

The **featuresString** value on each connector indicates the functionality that connector is capable of providing; when **PROVISIONING** is specified in the **featuresString**, the connector is a write-capable connector. The "`<entry key="MscsType" value="[MSCS-Type-Name]" />`" attribute requests the name for specific connector's MSCS Type value (also listed in the previous section here).

The out-of-the-box connector specifications can also be found in the **ConnectorRegistry.xml** file in the [IdentityIQ Installation Directory]/WEB-INF/config directory.

## Connector selection

---

Often there is more than one connector that can communicate with a single external application, which may raise questions as to which one is the best choice. When multiple connectors exist for a single application, they are always of different types. The best choice is dictated by the needs (and license limitations) of the organization. More information on connector selection is provided in the introductory section for each application within this document.

## **Viewing the available connectors**

# **Standard Deployment Connectors**

Many SailPoint customers have deployed the Connectors in this section. SailPoint still encourages deployment teams to obtain the latest troubleshooting and best practice information beyond what is contained in this document. For more specific information, refer to the Connector and Integration Deployment Center in Compass, SailPoint's Online customer portal.

This section contains information on the following:

- "SailPoint Active Directory Connector" on page 11
- "SailPoint AIX Connector" on page 35
- "SailPoint Azure Active Directory Connector" on page 49
- "SailPoint BMC Remedy Connector" on page 59
- "SailPoint BMC Remedy IT Service Management Suite Connector" on page 65
- "SailPoint DB2 Windows Connector" on page 73
- "SailPoint Delimited File Connector" on page 81
- "SailPoint JDBC Connector" on page 85
- "SailPoint Jive Connector" on page 91
- "SailPoint LDAP Connector" on page 97
- "SailPoint LDIF Connector" on page 115
- "SailPoint Logical Connector" on page 121
- "SailPoint Lotus Domino Connector" on page 127
- "SailPoint Linux Connector" on page 141
- "SailPoint Mainframe Connector" on page 153
- "SailPoint Microsoft SQL Server" on page 155
- "SailPoint Oracle Connector" on page 165
- "SailPoint PeopleSoft Connector" on page 171
- "SailPoint RACF Connector" on page 179
- "SailPoint RACF LDAP Connector" on page 187
- "SailPoint Salesforce Connector" on page 195
- "SailPoint SAP Portal-User Management Web Service Connector" on page 207
- "SailPoint SAP HR/HCM Connector" on page 213
- "SailPoint SAP HANA Connector" on page 229
- "SailPoint System for Cross-Domain Identity Management Connector 2.0" on page 237
- "SailPoint ServiceNow Connector" on page 241
- "SailPoint Siebel Connector" on page 253
- "SailPoint Solaris Connector" on page 259
- "SailPoint SQL Loader Connector" on page 271
- "SailPoint Sybase Connector" on page 277
- "SailPoint Tivoli Access Manager Connector" on page 283
- "SailPoint Top Secret Connector" on page 291
- "SailPoint UNIX Connector" on page 305
- "SailPoint Web Services Connector" on page 309
- "SailPoint Windows Local Connector" on page 321
- "SailPoint Workday Connector" on page 329
- "SailPoint XML Connector" on page 347

# Chapter 1: SailPoint Active Directory Connector

---

The following topics are discussed in this chapter:

Overview .....	11
Supported features .....	12
Supported Managed System .....	13
Pre-requisites .....	13
Administrator permissions .....	14
Configuration parameters .....	14
Additional configuration parameter .....	15
Configuring Domain Settings .....	16
Configuring searchDNs .....	16
Schema attributes .....	17
Account attributes .....	17
Group attributes .....	23
Provisioning Policy attributes .....	24
Active Directory Recycle Bin .....	28
Additional information .....	29
Unstructured Target Collector .....	29
Troubleshooting .....	32

## Overview

---

This connector mainly uses the LDAP and ADSI interfaces to communicate with an Active Directory Domain Controller. The connector supports reading, provisioning Active Directory users and groups, and defining the scope to be managed in terms of multiple containers from a single domain or multiple domains from a forest or LDAP filters. The connector manages group membership, the primary group concept and terminal services attributes. The connector supports restoring deleted objects from Active Directory Recycle Bin.

This connector directly connects to the domain controllers to read all the attributes except reading the terminal services attributes and all provisioning operations which requires IQService. For large environments, for faster aggregation of the accounts, the connector supports aggregating multiple partitions defined on the application in parallel. For more information on parallel aggregation, see Appendix C: Partitioning Aggregation.

Active Directory connector supports Delta Aggregation. When Delta Aggregation is performed, the connector aggregates only the changes made to the Active Directory accounts and groups since last aggregation using Active Directory synchronization (DirSync) control.

In addition, the connector supports **Microsoft Exchange Server** to manage user mailboxes and distribution lists. While this connector reads Exchange Server attributes by connecting to the Domain Controller, for provisioning on Exchange Server 2010 and above, the IQService must be installed on a any Windows system within same domain having Windows Powershell 2.0 or above.

The connector supports enabling, updating, removing a mailbox for a user and enabling an Active Directory group as a distribution list.

The connector now supports **Microsoft Lync\Skype for Business Server** user management. This requires the IQService to be installed on a Windows server running **Microsoft Lync\Skype for Business Server** administrative

## Overview

tools to read and provision on Microsoft Lync\Skype for Business Server. The connector supports create, update, delete, enable/disable, setting policies, and managing PIN for Microsoft Lync\Skype for Business user.

## Supported features

---

The Active Directory connector provides the ability to provision users, groups, and entitlements. The connector supports the following features:

- Account Management
  - Manages Active Directory Users as Accounts
  - Aggregation, Delta Aggregation, Partitioning Aggregation, Refresh Account, Pass Through Authentication
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements
  - Manages Microsoft Lync\Skype for Business Server attributes, Terminal Services, Dial-in Attributes
  - Create, Update, Delete Exchange User Mailbox
  - Password Interceptor
- Account - Group Management
  - Manages Active Directory Groups as Account-Groups
  - Aggregation, Delta Aggregation, Refresh Group
  - Create, Update, Delete
  - Create, Delete Exchange Distribution List
- Permission Management
  - Application can be configured for following unstructured target collectors to read permissions from respective end systems -
    - Windows File Share:** Read Windows File Share permissions directly assigned to accounts and groups.
    - SharePoint:** Read SharePoint permissions directly assigned to accounts and groups.
  - Supports automated revocation of the aggregated permissions and creates work items for requests only when the default provisioning action is overridden and **Manual Work Item** is selected as the provisioning action.
- Other
  - Supports one application managing multiple domains from single forest
  - Restore deleted objects (Active Directory Accounts and Groups) using 'Active Directory Recycle Bin'
  - Supports executing native before/after scripts for provisioning requests

## References

- "Active Directory Recycle Bin" on page 28
- "Unstructured Target Collector" on page 29
- "Appendix A: Delta Aggregation"
- "Appendix C: Partitioning Aggregation"
- "Appendix E: IQService"

## Supported Managed System

---

- Supported Active Directory Domain Services (AD DS) functional levels
  - Microsoft Windows Server 2016
  - Microsoft Windows Server 2012 R2
  - Microsoft Windows Server 2012
  - Microsoft Windows Server 2008 R2
  - Microsoft Windows Server 2008
  - Microsoft Windows Server 2003
- Supported Microsoft Exchange Servers
  - Microsoft Exchange Server 2016
  - Microsoft Exchange Server 2013
  - Microsoft Exchange Server 2010
- Supported Microsoft Lync\Skype for Business Servers
  - Microsoft Skype for Business 2015 Server
  - Microsoft Lync Server 2013

## Pre-requisites

---

1. Before you can use the provisioning feature of the connector, the IQService must be installed and registered on any Windows system with any of the supported Operating System. For more information on installing and registering IQService, see "Appendix E: IQService".
2. For managing Terminal Services (Remote Desktop Services profile) attributes, install the IQService on a Server class Windows Operating System.
3. For managing multiple domains in a forest having different domain tree structure in a forest there must be two way trust relationship between the domain trees to get the cross domain group memberships. For example, if there are two domain trees with their root domain as ABC . COM and XYZ . COM then there must be two way trust relationship between ABC . COM and XYZ . COM domains in order to get the group membership of a user present in ABC . COM which is having the membership on a group present in XYZ . COM and vice-versa.

## Administrator permissions

---

- For user provisioning through IQService, required that the administrator have the appropriate rights on the Active Directory. The Domain Controller should be accessible from the IQService host computer.

**Note:** The rights discussed in the following section grant limited account creation privileges to a user. This user can create and modify most accounts. It cannot manage the Administrator user account, the user accounts of administrators, the Server Operators, Account Operators, Backup Operators, and Print Operators. To manage these user types you must assign the appropriate security rights or add the user to groups having higher permissions. For example, domain administrators.

The administrative user specified in the application configuration will need additional rights for provisioning. These rights can be assigned by adding the user to the Account Operators group.

More granular rights can be assigned to users for specific portions of the directory, but this is discouraged by Microsoft best practices for Active Directory access control. The required rights will depend on the use cases that are implemented, but could include

- Read All Properties
  - Write All Properties
  - Create User Objects
  - Delete User Objects
  - Change Password
  - Reset Password
  - Read Members
  - Write Members
- For managing Exchange Server 2010 and above, the application user must be a member of Recipient Management group.

**Note:** Application user for provisioning of Exchange version 2010 and above must be Remote shell enabled. To enable remote Shell for a user, set the ‘RemotePowerShellEnabled’ parameter to \$True on the Set-User cmdlet.  
For example, Set-User UserName -RemotePowerShellEnabled \$True

- For Microsoft Lync\Skype for Business Server user management, application user must be a member of **RTCUniversalServerAdmins** and **CSAdministrator** domain groups. The user must also be a member of local Administrator group where IQService is installed.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Active Directory connector uses the following connection parameters:

Attributes	Description
IQService Host	Host name of the system where IQService is installed.

Attributes	Description
IQService Port	Port number used by the IQService.
Page Size	The number of objects to get, per page, when iterating over large numbers of objects. The default is 100.
Authentication Search Attributes	The list of attributes that will be used when authenticating application using pass-through authentication. By default <b>distinguishedName</b> , <b>sAMAccountName</b> , <b>uid</b> and <b>mail</b> are defined in the application configuration. It is recommended to modify the default values to have minimum number of attributes and it should be single attribute.
Exchange Version	The version of the Exchange Server if it needs to be managed by the application.
Exchange Host	(Applicable only for Exchange Server version 2010 or above) Fully qualified domain name (FQDN) or IP of Exchange Servers in priority order.  For example, <pre>&lt;entry key="ExchHost"&gt;     &lt;value&gt;         &lt;List&gt;             &lt;String&gt;VMEXCH2010.SPDomain.local&lt;/String&gt;         &lt;/List&gt;     &lt;/value&gt; &lt;/entry&gt;</pre>
Manage Skype for Business	The checkbox must be selected if the Microsoft Lync\Skype for Business Server is to be managed by the application.  <b>Note:</b> To optimize the performance of account aggregation with Microsoft Lync\Skype for Business Server attributes, run IQService with the Application user.
Delta Aggregation Mode	The following types of delta aggregators are provided: <ul style="list-style-type: none"> <li>• <b>uSNChanged:</b> Based on uSNCchanged attribute of Active Directory.</li> <li>• <b>DirSync:</b> Based on DirSync feature of Active Directory.</li> </ul>

## Additional configuration parameter

---

Attributes	Description
unlockOnChangePassword	The default behavior of unlocking the account on change password can be turned off by setting the unlockOnChangePassword attribute to <b>false</b> . Default: <b>true</b>
allowPartialResultException	During aggregation, if this parameter is set to true, it will ignore any PartialResultException.
setAttributeLevelResult	Set it to true to enable attribute request level results. Default: <b>False</b>  <b>Note:</b> Enabling this parameter would marginally increase the time taken to process the request.

## Configuration parameters

Attributes	Description
aggregationMaxRetries	Count of maximum retry attempts for Active Directory aggregation. Default: <b>5</b>
aggregationRetryThreshold	Delay in seconds between each retry attempt of aggregation Default: <b>10 seconds</b>

## Configuring Domain Settings

---

Domain Settings consist of details to connect domain/s such as Domain DN, Username, Password, Domain Controllers to connect and so on. Domain settings must be configured for all domains that this application is expected to contact.

The **Discover Domains** option provides way to enumerate all the domains present in the forest when the Global Catalog and the credentials of a user with permissions mentioned in “Administrator permissions” are provided.

Configuring the Global Catalog details also helps improve the pass-through authentication performance. The Active Directory Connector provides preference to connect to the Global Catalog if details are provided, else uses Server configured for respective domains to authenticate the users.

The following table lists the attributes that must be configured for each domain that the application is managing.

Attributes	Description
Domain*	Distinguished name of the domain.
User*	User of the domain in <b>Domain\User</b> format with appropriate rights required to read and provision.
Password*	Password of the user mentioned for <b>User</b> field.
Servers	The list of host name or IP address of the Domain Controller/s in the domain. The list of multiple Domain Controllers will be used by the connector to achieve failover. Click the browse button next to the list to enumerate servers available for this domain. Connector uses the Global Catalog and its credentials provided under <b>Discover Domains</b> option to enumerate servers. If no Servers are listed here, connector would perform serverless connection to the Active Directory Domain.
SSL	Indicates whether this is a SSL communication

## Configuring searchDNs

---

The searchDNs define list of distinguished names of the containers along with other relevant attributes which defines scope for this application. Each of these searchDNs is considered as a partition for parallel aggregation. Accounts and Groups can have different set of searchDNs to define different scope for each of them. In case the scope is not defined for Groups, it follows Accounts scope. Defining one search DN to the minimum is required to successfully configure application.

Attributes to be defined for searchDNs are as follows:

Attributes	Description
Search DN*	Distinguished Name of the container.

Attributes	Description
Iterate Search Filter	LDAP filter that defines scope for accounts/groups from this container.
Primary Group Search DN	(Applicable only for account search scope) Distinguished Name of the container that defines search scope when looking for primary group for the accounts. Defaults to Search DN
Group Membership Search DN	(Applicable only for account search scope) Distinguished Names of the containers/domains separated by semicolon (;) that defines search scope when looking for group membership for the accounts. If not defined, connector considers <b>groupMemberSearchDN</b> defined on the application or this searchDN in order.
Group Member Filter String	(Optional and applicable only for account search scope) LDAP Search filter to apply while fetching user's group membership.

**Note:** To increase the scope of search for Exchange Servers, add the following parameter in the application debug page:

```
<entry key="setViewEntireForest" value="True" />
```

## Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports two types of objects, account and group. Account objects are used when building identities Link objects. The group schema is used when building AccountGroup objects which are used to hold entitlements shared across identities.

## Account attributes

---

Name	Description
businessCategory	The types of business performed by an organization. Each type is one value of this multi-valued attribute. Examples: “engineering”, “finance”, and “sales”.
carLicense	This attribute type contains the license plate or vehicle registration number associated with the user.
cn	This attribute type contains names of an object. Each name is one value of this multi-valued attribute. If the object corresponds to a person, it is typically the person's full name. Examples: “Martin K Smith”, “Marty Smith” and “printer12”.
dn	This attribute contains the distinguished name by which the user is known.
departmentNumber	This attribute contains a numerical designation for a department within your enterprise.
description	This attribute type contains human-readable descriptive phrases about the object. Each description is one value of this multi-valued attribute. Examples: “Updates are done every Saturday, at 1am.”, and “distribution list for sales”.

## Schema attributes

Name	Description
destinationIndicator	<p>This attribute type contains country and city strings associated with the object (the addressee) needed to provide the Public Telegram Service. The strings are composed in accordance with CCITT Recommendations F.1 [F.1] and F.31 [F.31]. Each string is one value of this multi-valued attribute.</p> <p>Examples: "AASD" as a destination indicator for Sydney, Australia. "GBLD" as a destination indicator for London, United Kingdom.</p> <p><b>Note: The directory will not ensure that values of this attribute conform to the F.1 and F.31 CCITT Recommendations. It is the application's responsibility to ensure destination indicators that it stores in this attribute are appropriately constructed.</b></p>
displayName	This attribute contains the preferred name to be used for this person throughout the application.
employeeNumber	This attribute contains the numerical identification key for this person within your enterprise.
employeeType	This attribute contains a descriptive type for this user, for example, contractor, full time, or part time.
facsimileTelephoneNumber	This attribute type contains telephone numbers and any required parameters for facsimile terminals. Each telephone number is one value of this multi-valued attribute.
givenName	<p>This attribute type contains name strings that are the part of a person's name that is not their surname. Each string is one value of this multi-valued attribute.</p> <p>Examples: "John", "Sue", and "David".</p>
homePhone	This attribute contains the employee's home phone number.
homePostalAddress	This attribute contains the employee's mailing address.
homeMDB	Exchange mailbox store DN. Required for mailbox creation.
initials	<p>This attribute type contains strings of initials of some or all of an individual's names, except the surname(s). Each string is one value of this multi-valued attribute.</p> <p>Examples: "J. A." and "J"</p>
internationalISDNNumber	<p>This attribute type contains Integrated Services Digital Network (ISDN) addresses, as defined in the International Telecommunication Union (ITU) Recommendation E.164 [E.164]. Each address is one value of this multi-valued attribute.</p> <p>Example: "0198 444 444".</p>
l	<p>This attribute type contains names of a locality or place, such as a city, county, or other geographic region. Each name is one value of this multi-valued attribute.</p> <p>Examples: "Austin", "Chicago", and "Brisbane".</p>
mail	This attribute type contains the RFC822 mailbox for the user.

Name	Description
manager	This attribute type contains the distinguished name of the manager to whom this person reports.
mailNickname	Exchange Alias.
mobile	This attribute type contains the mobile telephone number of this person.
msExchHideFromAddressLists	Hide from Exchange address lists.
msNPAllowDialin	Indicates whether the account has permission to dial in to the RAS server.
msNPCallingStationID	If this property is enabled, the server verifies the caller's phone number. If the caller's phone number does not match the configured phone number, the connection attempt is denied.
msRADIUSCallbackNumber	The phone number that is used by the server is set by either the caller or the network administrator. If this property is enabled, the server calls the caller back during the connection process.
msRADIUSFramedRoute	Define a series of static IP routes that are added to the routing table of the server running the Routing and Remote Access service when a connection is made.
msRADIUSFramedIPAddress	Use this property to assign a specific IP address to a user when a connection is made.
o	This attribute type contains the names of an organization. Each name is one value of this multi-valued attribute.
ou	This attribute type contains the names of an organizational unit. Each name is one value of this multi-valued attribute. Examples: "Sales", "Human Resources", and "Information Technologies".
objectguid	Globally unique identifier of the object.
pager	This attribute type contains the telephone number of this persons pager.
physicalDeliveryOfficeName	This attribute type contains names that a Postal Service uses to identify a specific post office. Examples: "Austin, Downtown Austin" and "Chicago, Finance Station E".
postOfficeBox	This attribute type contains postal box identifiers use by a postal service to locate a box on the premises of the Postal Service rather than a physical street address. Each postal box identifier is a single value of this multi-valued attribute. Example: "Box 27".
postalAddress	This attribute type contains addresses used by a Postal Service to perform services for the object. Each address is one value of this multi-valued attribute. Example: "1111 Elm St.\$Austin\$Texas\$USA".

## Schema attributes

Name	Description
postalCode	This attribute type contains codes used by a Postal Service to identify postal service zones. Each code is one value of this multi-valued attribute. Example: "78664", to identify Pflugerville, TX, in the USA.
preferredDeliveryMethod	This attribute type contains an indication of the preferred method of getting a message to the object. Example: If the mhs-delivery Delivery Method is preferred over telephone-delivery, which is preferred over all other methods, the value would be: "mhs \$ telephone".
preferredLanguage	This attribute type contains the preferred written or spoken language of this person.
registeredAddress	This attribute type contains postal addresses to be used for deliveries that must be signed for or require a physical recipient. Each address is one value of this multi-valued attribute. Example: "Receptionist\$XYZ Technologies\$6034 Courtyard Dr. \$Austin, TX\$USA".
roomNumber	This attribute type contains the room or office number of this persons normal work location.
secretary	This attribute type contains the distinguished name of this persons secretary.
seeAlso	This attribute type contains the distinguished names of objects that are related to the subject object. Each related object name is one value of this multi-valued attribute. Example: The person object "cn=Elvis Presley,ou=employee,o=XYZ\, Inc." is related to the role objects "cn=Bowling Team Captain,ou=sponsored activities,o=XYZ\, Inc." and "cn=Dart Team,ou=sponsored activities,o=XYZ\, Inc.". Since the role objects are related to the person object, the 'seeAlso' attribute will contain the distinguished name of each role object as separate values.
sn	This attribute type contains name strings for surnames, or family names. Each string is one value of this multi-valued attribute. Example: "Smith".
st	This attribute type contains the full names of states or provinces. Each name is one value of this multi-valued attribute. Example: "Texas".
street	This attribute type contains site information from a postal address (that is, the street name, place, avenue, and the house number). Each street is one value of this multi-valued attribute. Example: "15 Main St.".
telephoneNumber	This attribute type contains telephone numbers that comply with the ITU Recommendation E.123 [E.123]. Each number is one value of this multi-valued attribute.
teletexTerminalIdentifier	The withdrawal of Recommendation F.200 has resulted in the withdrawal of this attribute.

Name	Description
telexNumber	This attribute type contains sets of strings that are a telex number, country code, and answer back code of a telex terminal. Each set is one value of this multi-valued attribute
title	This attribute type contains the persons job title. Each title is one value of this multi-valued attribute. Examples: "Vice President", "Software Engineer", and "CEO".
uid	This attribute type contains computer system login names associated with the object. Each name is one value of this multi-valued attribute. Examples: "s9709015", "admin", and "Administrator".
objectClass	The values of the objectClass attribute describe the kind of object which an entry represents. The objectClass attribute is present in every entry, with at least two values. One of the values is either "top" or "alias".
memberOf	This attribute type contains the account group membership for this person on the application.
objectSid	Windows Security Identifier
sAMAccountName	This attribute type contains the sAMAccoutName for this user.
primaryGroupId	This attribute type contains the RID of the this users primary group.
primaryGroupDN	This attribute type contains the distinguished name of this users primary group.
msDS-PrincipalName	Name of the entity in the following format:  NetBIOS domain name\sAMAccountName
TS_TerminalServicesProfilePath*	The roaming or mandatory profile path to be used when the user logs on to the RD Session Host server.
TS_TerminalServicesHomeDrive*	The root drive for the user.
TS_TerminalServicesHomeDirectory*	The root directory for the user.
TS_TerminalServicesInitialProgram*	The path and file name of the application that the user wants to start automatically when the user logs on to the RD Session Host server.
TS_TerminalServicesWorkDirectory*	The working directory path for the user.
TS_EnableRemoteControl*	A value that specifies whether to allow remote observation or remote control of the user's Remote Desktop Services session.
TS_AllowLogon*	A value that specifies whether the user is allowed to log on to the RD Session Host server.
TS_BrokenConnectionAction*	A value that specifies the action to be taken when a Remote Desktop Services session limit is reached.
TS_ReconnectionAction*	A value that specifies if reconnection to a disconnected Remote Desktop Services session is allowed.

## Schema attributes

Name	Description
TS_ConnectClientDrivesAtLogon*	A value that specifies if mapped client drives should be reconnected when a Remote Desktop Services session is started.
TS_ConnectClientPrintersAtLogon*	A value that specifies whether to reconnect to mapped client printers at logon. The value is one if reconnection is enabled, and zero if reconnection is disabled.
TS_DefaultToMainPrinter*	A value that specifies whether to print automatically to the client's default printer. The value is one if printing to the client's default printer is enabled, and zero if it is disabled.
TS_MaxConnectionTime*	The maximum duration of the Remote Desktop Services session, in minutes. After the specified number of minutes have elapsed, the session can be disconnected or terminated.
TS_MaxDisconnectionTime*	The maximum amount of time, in minutes, that a disconnected Remote Desktop Services session remains active on the RD Session Host server. After the specified number of minutes have elapsed, the session is terminated.
TS_MaxIdleTime*	The maximum amount of time that the Remote Desktop Services session can remain idle, in minutes. After the specified number of minutes has elapsed, the session can be disconnected or terminated. <sup>a</sup>
<b>Microsoft Lync\Skype for Business Server attributes</b>	
msRTCSIP-UserEnabled	Whether the user is currently enabled for Microsoft Lync\Skype for Business Server.
LineServerURI	The line server URL.
EnabledForFederation	Whether a user is enabled for federation.
PublicNetworkEnabled	Whether a user is enabled for access outside network.
EnterpriseVoiceEnabled	Whether a user EnterpriseVoiceEnabled service is enabled.
LineURI	The line Uniform Resource Identifier (URI).
SipAddress	This attribute contains the SIP address of a given user.
VoicePolicy	The name of Voice Policy.
MobilityPolicy	The name of Mobility Policy.
ConferencingPolicy	The name of Conferencing Policy.
PresencePolicy	The name of Presence Policy.
VoiceRoutingPolicy	The name of VoiceRouting Policy.
RegistrarPool	The name of registrar pool.
LocationPolicy	The name of Location Policy.
ClientVersionPolicy	The name of ClientVersion Policy.
ClientPolicy	The name of Conferencing Policy.
ExternalAccessPolicy	The name of ExternalAccess Policy.
HostedVoicemailPolicy	The name of HostedVoicemail Policy.

Name	Description
PersistentChatPolicy	The name of PersistentChat Policy.
UserServicesPolicy	The name of UserServices Policy.
ExperiencePolicy	The name of Experience Policy.
ArchivingPolicy	The name of Archiving Policy.
LegalInterceptPolicy	The name of LegalIntercept Policy.
PinPolicy	The name of Pin Policy.
LyncPinSet	Whether a user pin is set.
LyncPinLockedOut	Whether a user pin is locked.

a. Attributes with asterisk mark (\*) are the Terminal Services/Remote Desktop Services attributes. By default these attributes are not added to the schema and provisioning policy for performance optimization. To manage Terminal Services attributes add these attributes to schema and provisioning policy. Alternatively, you can uncomment these attributes from the connector registry and import it again.

## Group attributes

---

**Table 1—Active Directory Connector - Group Attributes**

Name	Description
cn	This attribute type contains names of an object. Each name is one value of this multi-valued attribute. If the object corresponds to a person, it is typically the person's full name. Examples: "Martin K Smith", "Marty Smith" and "printer12".
dn	This attribute type contains the directory path to the object.
owner	This attribute type contains the name of the owner of the object.
objectguid	Globally unique identifier of the object.
description	This attribute type contains human-readable descriptive phrases about the object. Each description is one value of this multi-valued attribute. Examples: "Updates are done every Saturday, at 1am.", and "distribution list for sales".
mailNickname	Exchange Alias.
memberOf	This attribute type contains the account group membership for this person on the application.
objectSid	This attribute type contains the Windows Security Identifier for this user.
sAMAccountName	sAMAccountName
groupType	Group Type. Allowed values are: 1. Security 2. Distribution

## Provisioning Policy attributes

**Table 1—Active Directory Connector - Group Attributes**

Name	Description
groupScope	Group Scope. Allowed values are: 1. Domain local 2. Global 3. Universal.
msDS-PrincipalName	Name of the entity in the following format: NetBIOS domain name\sAMAccountName

## Provisioning Policy attributes

---

The following table lists the provisioning policy attributes:

Attribute	Description
<b>Provisioning policy attributes for Account Creation</b>	
ObjectType	Type of the user to be created. Default value: User.
distinguishedName	Distinguished name of the user to be created.
sAMAccountName	sAMAccountName of the user to be created.
*password*	Password of the user to be created.
IIQDisabled	A boolean attribute, set to true to create a disabled user.
PrimaryGroupDN	Default group of the user to be created.
description	Description of the user to be created.
msNPAllowDialin	Indicates whether the account has permission to dial in to the RAS server.
msNPCallingStationID	If this property is enabled, the server verifies the caller's phone number. If the caller's phone number does not match the configured phone number, the connection attempt is denied.
msRADIUSCallbackNumber	The phone number that is used by the server is set by either the caller or the network administrator. If this property is enabled, the server calls the caller back during the connection process.
msRADIUSFramedRoute	Define a series of static IP routes that are added to the routing table of the server running the Routing and Remote Access service when a connection is made.
msRADIUSFramedIPAddress	Use this property to assign a specific IP address to a user when a connection is made.

Attribute	Description
TS_TerminalServicesProfilePath*	The roaming or mandatory profile path to be used when the user logs on to the RD Session Host server.
TS_TerminalServicesHomeDrive*	The root drive for the user.
TS_TerminalServicesHomeDirectory*	The root directory for the user.
TS_TerminalServicesInitialProgram*	The path and file name of the application that the user wants to start automatically when the user logs on to the RD Session Host server.
TS_TerminalServicesWorkDirectory*	The working directory path for the user.
TS_EnableRemoteControl*	A value that specifies whether to allow remote observation or remote control of the user's Remote Desktop Services session.
TS_AllowLogon*	A value that specifies whether the user is allowed to log on to the RD Session Host server.
TS_BrokenConnectionAction*	A value that specifies the action to be taken when a Remote Desktop Services session limit is reached.
TS_ReconnectionAction*	A value that specifies if reconnection to a disconnected Remote Desktop Services session is allowed.
TS_ConnectClientDrivesAtLogon*	A value that specifies if mapped client drives should be reconnected when a Remote Desktop Services session is started.
TS_ConnectClientPrintersAtLogon*	A value that specifies whether to reconnect to mapped client printers at logon. The value is one if reconnection is enabled, and zero if reconnection is disabled.
TS_DefaultToMainPrinter*	A value that specifies whether to print automatically to the client's default printer. The value is one if printing to the client's default printer is enabled, and zero if it is disabled.
TS_MaxConnectionTime*	The maximum duration of the Remote Desktop Services session, in minutes. After the specified number of minutes have elapsed, the session can be disconnected or terminated.
TS_MaxDisconnectionTime*	The maximum amount of time, in minutes, that a disconnected Remote Desktop Services session remains active on the RD Session Host server. After the specified number of minutes have elapsed, the session is terminated.
TS_MaxIdleTime*	The maximum amount of time that the Remote Desktop Services session can remain idle, in minutes. After the specified number of minutes has elapsed, the session can be disconnected or terminated. <sup>a</sup>
preferredServer	The preferred server (Domain Controller) on which this request must be executed. Connector considers the list of servers mentioned in domainSettings in case this server is unavailable.

## Provisioning Policy attributes

Attribute	Description
<b>Delete provisioning policy attribute for non-leaf user object</b>	
deleteSubTree	<p>To delete the non-leaf user objects set the value of the attribute to true (boolean).</p> <p>For example,</p> <pre>&lt;ProvisioningPlan nativeIdentity="Adam" targetIntegration="AD-Direct"&gt;     &lt;AccountRequest application="AD-Direct" nativeIdentity="CN=Adam,CN=Users,DC=SPDomain,DC=local" op="Delete"&gt;         &lt;AttributeRequest name="deleteSubTree" op="Add"&gt;             &lt;Value&gt;                 &lt;Boolean&gt;true&lt;/Boolean&gt;             &lt;/Value&gt;         &lt;/AttributeRequest&gt;     &lt;/AccountRequest&gt; &lt;/ProvisioningPlan&gt;</pre>
<b>Special provisioning attributes for Move/Rename request</b>	
AC_NewName	A string attribute to rename the user. For example, CN=abc
AC_NewParent	A string attribute to move the user to new OU. For example, OU=xyz,DC=pqr,DC=com
<b>Provisioning Exchange mailbox</b>	
homeMDB	Exchange mailbox store DN. Required for mailbox creation. Optional for Exchange 2010 and above. Send this attribute with new mailbox store DN to move the mailbox to other mailbox store.
mailNickname	Exchange alias. Required for mailbox creation and to update or disable the mailbox. Send this attribute with no value to disable the mailbox.
msExchHideFromAddressLists	(Optional) Hide from Exchange address lists.
DomainController	(Optional) Fully qualified domain name (FQDN) of the domain controller that writes this configuration change to Active Directory.
<p><b>Note:</b> Connector supports updating any other Exchange mailbox attributes supported by <b>set-mailbox</b> cmdlet. To set any such parameter, prefix the parameter name of the <b>set-mailbox</b> cmdlet with <b>Exch_</b> while adding the attribute to the provisioning policy.</p> <p>Alternatively, edit application xml file to add an application attribute named <b>exchangeAttributes</b> of string type with comma separated name of Exchange attributes added in provisioning policy.</p>	
<b>Provisioning policy attributes for Microsoft Lync\Skype for Business user</b>	
SipAddress	To assign the user a specific SIP address.

Attribute	Description
RegistrarPool	Registrar pool where the user's Microsoft Lync\Skype for Business Server account will be homed. Mandatory attribute. Send this attribute with no value to remove the user from Microsoft Lync\Skype for Business Server.
SipAddressType	Select one of the SipAddressType from supported values: SamAccountName, FirstName, LastName, EmailAddress. Microsoft Lync\Skype for Business Server generates a SIP address for the new user when SipAddressType is provided in combination with SipDomain.
SipDomain	The SIP domain for the user account being enabled. Microsoft Lync\Skype for Business Server generates a SIP address for the new user when SipAddressType is provided in combination with SipDomain.
msRTCSIP-UserEnabled	Send this attribute with true/false to enable/disable Microsoft Lync\Skype for Business.
Pin	Dial-in Conferencing PIN number to be set for the Microsoft Lync\Skype for Business.
LyncPinLockedOut	Send this attribute with true/false to lock/unlock Microsoft Lync\Skype for Business user's Dial-in Conferencing PIN.

**Note:** Active Directory Connector also supports provisioning other Microsoft Lync\Skype for Business attributes other than mentioned above.

To set any other Microsoft Lync\Skype for Business attributes, edit application xml file to add **lyncAttributes** application attribute of string type with comma separated name of Microsoft Lync\Skype for Business attributes added in provisioning policy.

Provisioning policy attributes for CreateGroup	
distinguishedName	Group in the distinguished name format.
sAMAccountName	sAMAccountName
Provisioning policy attributes for UpdateGroup	
description	A description of the group.
groupType	Group Type. Allowed values are: 1. Security 2. Distribution
groupScope	Group Scope. Allowed values are: 1. Domain local 2. Global 3. Universal.
mailNickname	Alias and is required if want to create Distribution Group on exchange. Only Universal type of group can be created on exchange.

## Active Directory Recycle Bin

a. \* - Attributes with asterisk mark (\*) are the Terminal Services/Remote Desktop Services attributes. By default these attributes are not added to the schema and provisioning policy for performance optimization. To manage Terminal Services attributes add these attributes to schema and provisioning policy. Alternatively, you can uncomment these attributes from the connector registry and import it again.

**Note:** **To skip plan attributes getting processed by connector, add excludeAttributesFromProvisioning attribute to application with value listing names of such attributes. For example,**

```
<entry key="excludeAttributesFromProvisioning">
  <value>
    <List>
      <String>region</String>
    </List>
  </value>
</entry>
```

## Active Directory Recycle Bin

---

A new feature ‘Recycle Bin’ introduced by Microsoft in Windows Server 2008 R2 provides support for restoring deleted users, groups with all their attributes and group memberships. SailPoint Active Directory Connector support this feature. Using this feature, any deleted objects (Accounts and Groups) can be restored.

### Pre-requisites

---

**Note:** **Recycle Bin feature should be enabled on Active Directory.**

1. IQService can be installed on Windows system with one of the following Operating System:
  - Microsoft Windows Server 2008 R2
  - Microsoft Windows Server 2012For more information on installing and registering IQService, see “Appendix E: IQService”.
2. Install **Active Directory module for Windows PowerShell** on the computer where IQService is installed.

**Note:** **By default this module is installed on all DCs.**

**For non-DC but server class Operating System computer, open Windows PowerShell Console and execute the following commands:**

- Import-Module ServerManager
- Add-WindowsFeature -Name "RSAT-AD-PowerShell" -IncludeAllSubFeature

3. Run the following PowerShell command on all domain controllers (DCs) in the forest which must be managed:  
Enable-PSRemoting

**Note:** **If multiple servers are managed, run the above command on all the servers present under the “domainSettings”.**

## Configuring Recycle Bin

---

1. Open the Console and import `IIQHOME\WEB-INF\config\configManageDeletedObjects.xml` file. The `configManageDeletedObjects.xml` file creates the **Manage Recycle Bin** quick link on the dashboard and adds the **Restore Deleted Objects** workflow.
2. Modify **manageRecycleBin** attribute in the Active Directory application with the value set to **true**.
 

```
<entry key="manageRecycleBin">
    <value>
        <Boolean>true</Boolean>
    </value>
</entry>
```
3. After account and account-group aggregation, the deleted object would be visible under the **Manage Recycle Bin** quick link. Accounts/Groups can be restored individually or all together.
4. The **DirSync** delta aggregator also supports detecting deleted objects.

## Additional information

---

This section describes the additional information related to the Active Directory Connector.

### Unstructured Target Collector

---

Unstructured target information is used to define unstructured data sources from which the connector is to extract data. Unstructured data is any data that is stored in a format that is not easily readable by a machine. For example, information contained in an Excel spread sheet, the body of an email, a Microsoft Word document, or an HTML file is considered unstructured data. Unstructured targets pose a number of challenges for connectors, because not only is the data stored in a format that is hard to extract from, the systems and directory structures in which the files reside are often difficult to access.

The unstructured target collectors those can be configured with Active Directory application are as follows:

- Windows file share  
For more information, see “Windows File Share”.
- SharePoint  
For more information, see “SharePoint Target Collector”.

**Note:** Active Directory Connector supports automated revocation of the Target Permissions.

### Windows File Share

Windows file share target collector can be configured on Active Directory application to read and correlate file share permissions on Active Directory entities. To correlate the aggregated permissions, ensure that the following attribute is marked as Correlation Key in respective schema:

- **objectSid** for Accounts and Groups

This target collector requires a the IQService to be installed on a machine that has visibility to the directory or share to include in the target scan. Refer to the Installation Guide for information on installing and registering the IQService.

## Additional information

The unstructured targets defined on this tab are used by the Target Aggregation task to correlate targets with permissions assigned to identities and account groups for use in certifications.

The Unstructured Targets tab contains the following information:

**Table 2—Application Configuration - Unstructured Targets Tab field descriptions**

Field	Description
<b>Attributes:</b> The required settings for connecting to the IQService.	
IQService Host	The host on which the IQService resides.
IQService Port	The TCP/IP port where the IQService is listening for requests.
Number of targets per block	Number or targets (files) to include in each block of data returned.
<b>File Shares:</b> The required information for each share.	
Path	UNC Style path to a share or local directory. You can target a specific file or a directory and its sub-directories containing multiple files from which to extract the required data. If you target a directory, use the <b>Wildcard</b> and <b>Directory Depth</b> fields to narrow the query if possible.
Directories Only	Use to instruct to the collector to ignore files and just report back directory permission information.
Directory Depth	The sub-directory depth from which to extract data. The <b>Directory Depth</b> field enables you to extend your query up to ten (10) sub-directories below the one specified in the <b>Path</b> field.
Wildcard	Use wild cards to target a particular file type of naming scheme. For example, to search only Excel spread sheets, use * .xls or to search only files with names beginning with finance_, use finance_*.*
Include Inherited Permissions	Use to instruct the collector to not report permissions unless they are directly assigned. Only directly assigned permissions will be returned
Administrator	The administrator that has access to this share so you can collect permissions. This value should be the users principal user@xyz.com name or a fully qualified domain user name in the domain\\user format.
Password	The password associated with the specified administrator. <b>Note:</b> The service will be running as System or can be configured to be run as any user, so the Administrator/Password fields may not be required in all cases.
<b>Rules:</b> Specify the rules used to transform and correlate the targets. <b>Note:</b> Click the “...” icon to launch the Rule Editor to make changes to your rules if needed.	
Creation Rule	The rule used to determine how the unstructured data extracted from data source is transformed into data that can be read by IdentityIQ.
Correlation Rule	The rule used to determine how to correlate account information from the application with identity cubes in IdentityIQ.
<b>Provisioning related Attributes:</b> Select the settings for provisioning to the share.	
Override Default Provisioning	Select it to override the default provisioning action for the collector.

**Table 2—Application Configuration - Unstructured Targets Tab field descriptions**

Field	Description
Provisioning Action	The overriding provisioning action for the collector.

To revoke permissions for Active Directory users and/or groups using Windows File Share Target Collector, perform the following:

1. Add the following attributes under target source configuration:  

```
<entry key="searchAttrForAcct" value="msDS-PrincipalName" />
<entry key="searchAttrForGrp" value=" msDS-PrincipalName " />
```
2. Remove the NO\_PERMISSIONS\_PROVISIONING feature string from the application configuration.

## SharePoint Target Collector

SharePoint target collector can be configured on Active Directory application to read and correlate SharePoint permissions on Active Directory entities. To correlate the aggregated permissions, ensure that following attributes are marked as Correlation Key in respective schema as follows:

- For SharePoint 2007 and 2010: **sAMAccountName** for Accounts and Groups.
- For SharePoint 2013: **sAMAccountName** for Accounts and **objectSid** for Groups

Multiple target sources can be specified and configured for applications which supports unstructured targets. This will be useful in the applications where the target permissions can be fetched from multiple target sources. For example, as stated in overview section, SharePoint connector does not manage domain groups. For assigning the SharePoint target permission to domain groups, SharePoint target collector can be configured for Active Directory application along with Windows file share target collector.

SharePoint Target Collector supports aggregation for Sites, Lists, List Items, Folders and Files. The objects can be filtered based on various filters configured on the Unstructured Targets Tab.

Attribute	Description	Possible values
IQService Host	Host name/IP Address of the computer where IQService is installed. The IQService needs to be installed on the SharePoint server.	
IQService Port	Port number used by the IQService.	Default: 5050
SharePoint Server Version	Version of the SharePoint Server	Default: 2007
Site Collection URL	URL of Site or Site Collection for target aggregation.	URL. Cannot be blank
UserName	User with Site Collection Administrator permission.	The user name format should be as present in SharePoint.  For Windows Claim based authentication, the user name should be in encoding format. For example, <b>i:0#.w contoso\chris</b>

## Troubleshooting

Attribute	Description	Possible values
Password	Password for UserName	
Target Types Filter	As mentioned above, the Target Collector supports aggregating Sites, Lists, List Items and Files. Using this filter, any of these target types can be selectively aggregated.	Any combination of following separated by comma: Sites,Files,Lists,ListItems,Folders,Files  List Item specific filtration, for example, Document, Picture, Wiki Page and so on.  If not specified, all target types would be aggregated.  Default – Not specified
Site Filter Type	This is used in combination to the Site Filter. This tells whether the Site Filter define the inclusion filter or exclusion filter.	Include/Exclude  Default: Include
Site Filter	Targets with path containing Words / phrases mentioned here can be selectively included or excluded depending on the Site Filter Type	Words/phrases separated by comma.  If not specified all the targets would be aggregated.  Default: Not specified

To revoke permissions for Active Directory users and/or groups using SharePoint Target Collector, perform the following:

1. Add the following attributes under target source configuration:  
`<entry key="searchAttrForAcct" value="msDS-PrincipalName" />`  
`<entry key="searchAttrForGrp" value="msDS-PrincipalName" />`
2. Remove the NO\_PERMISSIONS\_PROVISIONING feature string from the application configuration.  
`NO_PERMISSIONS_PROVISIONING`

## Troubleshooting

---

### 1 - The account aggregation fails in a short time with the error “Error processing control” when multiple partitions are processed in parallel

The account aggregation fails in a short time with the following error when multiple partitions are processed in parallel:

Exception during aggregation. Reason:

```
java.lang.RuntimeException: java.lang.RuntimeException:  
javax.naming.OperationNotSupportedException: [LDAP: error code 12 - 00000057:  
LdapErr: DSID-0C090747, comment: Error processing control, data 0, v1772];
```

**Resolution:** The sort control can be disabled for the aggregation by adding the following to application when the aggregation fails with **OperationNotSupportedException** while processing multiple partitions:

```
<entry key="disableSort" value="true"/>
```

## 2 - While managing Exchange 2010 and Microsoft Lync\Skype for Business Server, IQService fails after renaming "App.config" file in the IQService directory to "IQService.exe.config"

While managing Exchange 2010 and Microsoft Lync\Skype for Business Server, IQService fails after renaming App.config file in the IQService directory to IQService.exe.config.

Any provisioning operations involving IQService fails with the following error:

```
ProvisionExchange [] DEBUG : "Executing command :Get-mailbox"
ADConnectorServices [] ERROR : "Caught exception in Modify. Value cannot be null.
Parameter name: serverSettingsValue cannot be null.
Parameter name: serverSettings"
```

**Resolution:** Modify IQService.exe.config file to have the following startup element attribute:

```
<startup useLegacyV2RuntimeActivationPolicy="true">
```

After modifying the IQService.exe.config file, the configuration must look as follows:

```
<configuration>
  <startup useLegacyV2RuntimeActivationPolicy="true">
    <supportedRuntime version="v4.0" />
  </startup>
</configuration>
```

## 3 - After upgrading .NET from 4.0 to 4.5.2 provisioning to Exchange crashes IQService

IQService fails after upgrading .NET from 4.0 to 4.5.2 and provisioning an Active Directory account with Exchange.

**Resolution:** Install .NET version 4.0 and perform provisioning.

## **Troubleshooting**

# Chapter 2: SailPoint AIX Connector

---

The following topics are discussed in this chapter:

Overview .....	35
Supported features .....	35
Supported Managed Systems .....	36
Pre-requisites .....	36
Administrator permissions .....	36
Configuration parameters .....	37
Additional configuration parameters for SSH configuration .....	37
Public key authentication configuration .....	38
Schema attributes .....	38
Account attributes .....	38
Group attributes .....	44
Provisioning policy attributes .....	44
Account attributes .....	44
Group attributes .....	45
Additional information .....	45
Additional information .....	45
Troubleshooting .....	46

## Overview

---

The SailPoint AIX Connector manages the accounts and groups on AIX computer.

## Supported features

---

SailPoint AIX Connector provides support for the following features:

- Account Management
  - Manages AIX Users as Accounts
  - Aggregation, Refresh Account
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements
- Account - Group Management
  - Manages AIX Groups as Account-Groups
  - Aggregation, Refresh Group
  - Create, Update, Delete

## Overview

- Permissions Management
  - Application can be configured to read file permissions directly assigned to accounts and groups using Unstructured Target Collector.
  - The connector supports automated revocation of the aggregated permissions for accounts and groups.

**Note:** AIX connector supports MD5, SHA-1, and SHA-2 cryptographic hash functions.

## References

- "Unstructured Target Collector" on page 45
- "Appendix D: Before and After Provisioning Action"

## Supported Managed Systems

---

The AIX connector supports the following versions of the operating system:

- AIX 7.2
- AIX 7.1

## Pre-requisites

---

SSH should be installed on AIX computer.

## Administrator permissions

---

- You can use root user for managing your applications.
- If you want to use sudo user to perform the provisioning operations, the sudo user must be configured with the following rights and permissions:

Rights to execute the following commands with root privilege:

```
/usr/sbin/lsuser, /usr/sbin/lsgroup, /usr/bin/chmod, /usr/bin/mkuser,  
/usr/sbin/userdel, /usr/bin/chuser, /usr/bin/chgroup, /usr/bin/mkgroup,  
/usr/sbin/rmgroup, /usr/bin/passwd, /bin/rm, /bin/echo, /usr/bin/find,  
/usr/bin/pwdadm
```

An entry in /etc/sudoers file should look similar to the following:

```
username ALL = (root) PASSWD : /usr/sbin/lsuser, /usr/sbin/lsgroup,  
/usr/bin/chmod, /usr/bin/mkuser, /usr/sbin/userdel, /usr/bin/chuser,  
/usr/bin/chgroup, /usr/bin/mkgroup, /usr/sbin/rmgroup, /usr/bin/passwd, /bin/rm,  
/bin/echo, /usr/bin/find, /usr/bin/pwdadm
```

**Note:** All commands mentioned above are for default configuration. If any of the command is modified in application definition, then the respective changes in /etc/sudoers file entry should also be performed. Verify command paths on AIX computers as they might differ from the values mentioned here.

**Note:** If you want to use sudo user to perform the provisioning operations ensure to configure home directory with proper write access for this sudo user. In case sudo user is using Guest home directory then ensure it has proper write access over this directory.

# Configuration parameters

---

The following table lists the configuration parameters of AIX Connector:

Parameters	Description
Unix Server Host	Host Name/IP address of AIX computer.  <b>Note: For IdentityIQ version 6.4 Patch 4 and above, the format of the application XML has been changed from</b> <code>&lt;entry key="UnixServerHost" value="<hostname>" /&gt;</hostname></code> <b>to</b> <code>&lt;entry key="host" value="<hostname>" /&gt;</hostname></code>
SSH Port	SSH port configured. Default value: 22
Not a 'root' user	If User ID specified is not root, check this parameter.
User Name	User ID on AIX computer that you want to use for connector operations.
User Password	Password of the target system user account that you want to use for connector operations.
Private Key File Path	Path to Private Key File. Private/Public key authentication will have precedence over password authentication.
Passphrase For Private Key	Passphrase provided for creating Private Key.

## Additional configuration parameters for SSH configuration

---

The following procedure provides the steps for adding the additional configuration parameters for SSH configuration in Application or Target Source debug page.

**Note: These additional configuration parameters must be added in the Application/Target Source debug page.**

1. Following is the default command for setting shell prompt on UNIX computer:

```
<entry key="SetPrompt" value="PS1='SAILPOINT>' />
```

In the above command, "SetPrompt" is the application/target source attribute and PS1='SAILPOINT' is the value of the application/target source attribute.

If the command for setting shell prompt is different than the default command, change the value of the application/target source attribute. If the application/target source attribute is not found, add the following new entry in the application/target source debug page:

For example: For tcsh shell, the entry value would be:

```
<entry key="SetPrompt" value="set prompt='SAILPOINT>' />
```

2. For executing the commands, verify that the default shell is present on your system.

If the default shell present on your UNIX system is different, modify the value of the application/target source attribute. If the application/target source attribute is not found, add the following new entry in the application/target source debug page:

```
<entry key="DEFAULT_SSH_SHELL" value="tcsh" />
```

## Public key authentication configuration

---

This is an alternative security method to using passwords. To use public key authentication, you must generate a public and a private key (that is, a key pair). The public key is stored on the remote hosts on which you have accounts. The private key is saved on the computer you use to connect to those remote hosts. This method allows you to log into those remote hosts, and transfer files to them, without using your account passwords.

Perform the following configuration steps to make the UNIX computer as the server and IdentityIQ computer as client:

1. Generate Private and Public key's. For more information of the standard steps, see "5 - Test connection fails for key based authentication with an error." on page 47.
2. Append contents of public key file to `~/.ssh/authorized_keys` as shown below.  
`cat <public key file> >> ~/.ssh/authorized_keys`
3. Copy private key file to a location which is accessible by IdentityIQ server.
4. Provide path of private key file in application configuration.

## Schema attributes

---

This section describes the different schema attributes.

### Account attributes

---

The following table lists the account attributes:

Attributes	Description
User Name	Name of the user on AIX computer that you want to use for connector operations.
gecos	The General Electric Comprehensive Operating System (GECOS) information for User. The user's name, phone numbers, and other generic personal information are stored here.
id	User ID
pgrp	Primary group of user.
groups	Secondary groups of user.
home	Home directory of user.
shell	Default shell of user.
login	Indicates whether the user can log in to the system with the <code>login</code> command. Possible values are: <ul style="list-style-type: none"><li>• <b>true</b>: The user can log in to the system. Default.</li><li>• <b>false</b>: The user cannot log in to the system.</li></ul>
su	Indicates whether another user can switch to the specified user account with the <code>su</code> command. Possible values are: <ul style="list-style-type: none"><li>• <b>true</b>: Another user can switch to the specified account. Default</li><li>• <b>false</b>: Another user cannot switch to the specified account.</li></ul>

Attributes	Description
rlogin	<p>Permits access to the account from a remote location with the <b>telnet</b> or <b>rlogin</b> commands. Possible values are:</p> <ul style="list-style-type: none"> <li>• <b>true</b>: The user account can be accessed remotely. Default</li> <li>• <b>false</b>: The user account cannot be accessed remotely.</li> </ul>
daemon	<p>Indicates whether the user specified by the <i>Name</i> parameter can execute programs using the <b>cron</b> daemon or the <b>src</b> (system resource controller) daemon. Possible values are:</p> <ul style="list-style-type: none"> <li>• <b>true</b>: The user can initiate <b>cron</b> and <b>src</b> sessions. Default</li> <li>• <b>false</b>: The user cannot initiate <b>cron</b> and <b>src</b> sessions.</li> </ul>
admin	<p>Defines the administrative status of the user. Possible values are:</p> <ul style="list-style-type: none"> <li>• <b>true</b>: The user is an administrator. Only the root user can change the attributes of users defined as administrators.</li> <li>• <b>false</b>: The user is not an administrator. Default</li> </ul>
dce_export	<p>Allows the DCE registry to overwrite the local user information with the DCE user information during a DCE export operation. Possible values are:</p> <ul style="list-style-type: none"> <li>• <b>true</b>: Local user information will be overwritten</li> <li>• <b>false</b>: Local user information will not be overwritten</li> </ul>
sugroups	<p>Lists the groups that can use the su command to switch to the specified user account. The <i>Value</i> parameter is a comma-separated list of group names, or a value of ALL to indicate all groups. An ! (exclamation point) in front of a group name excludes that group. If this attribute is not specified, all groups can switch to this user account with the su command.</p>
admgroups	<p>Lists the groups the user administers. The <i>Value</i> parameter is a comma-separated list of group names. For additional information on group names, see the <b>adms</b> attribute of the /etc/security/group file.</p>
tpath	<p>Indicates the user's trusted path status. The possible values are:</p> <ul style="list-style-type: none"> <li>• <b>always</b>: The user can only execute trusted processes. This implies that the user's initial program is in the trusted shell or some other trusted process.</li> <li>• <b>notsh</b>: The user cannot invoke the trusted shell on a trusted path. If the user enters the secure attention key (SAK) after logging in, the login session ends.</li> <li>• <b>nosak</b>: The secure attention key (SAK) is disabled for all processes run by the user. Use this value if the user transfers binary data that may contain the SAK sequence. Default</li> <li>• <b>on</b>: The user has normal trusted path characteristics and can invoke a trusted path (enter a trusted shell) with the secure attention key (SAK).</li> </ul>
ttys	<p>Lists the terminals that can access the account specified by the <i>Name</i> parameter. The <i>Value</i> parameter is a comma-separated list of full path names, or a value of ALL to indicate all terminals. The values of RSH and REXEC also can be used as terminal names. An ! (exclamation point) in front of a terminal name excludes that terminal. If this attribute is not specified, all terminals can access the user account. If the <i>Value</i> parameter is not ALL, then /dev/pts must be specified for network logins to work.</p>

## Schema attributes

Attributes	Description
expires	Identifies the expiration date of the account. The <i>Value</i> parameter is a 10-character string in the <i>MMDDhhmmyy</i> form, where <i>MM</i> = month, <i>DD</i> = day, <i>hh</i> = hour, <i>mm</i> = minute, and <i>yy</i> = last 2 digits of the years 1939 through 2038. All characters are numeric. If the <i>Value</i> parameter is 0, the account does not expire. The default is 0. See the <b>date</b> command for more information.
auth1	<p>Lists additional mandatory methods for authenticating the user. The <b>auth1</b> attribute has been deprecated and may not be supported in a future release. The <b>SYSTEM</b> attribute should be used instead. The authentication process will fail if any of the methods specified by the <b>auth1</b> attribute fail.</p> <p>The <i>Value</i> parameter is a comma-separated list of <i>Method;Name</i> pairs. The <i>Method</i> parameter is the name of the authentication method. The <i>Name</i> parameter is the user to authenticate. If you do not specify a <i>Name</i> parameter, the name of the user being authenticated is used. Valid authentication methods for the <b>auth1</b> and <b>auth2</b> attributes are defined in the <i>/etc/security/login.cfg</i> file.</p>
auth2	<p>Lists additional optional methods for authenticating the user. The <b>auth2</b> attribute has been deprecated and may not be supported in a future release. The <b>SYSTEM</b> attribute should be used instead. The authentication process will not fail if any of the methods specified by the <b>auth2</b> attribute fail.</p> <ul style="list-style-type: none"> <li>• The <i>Value</i> parameter is a comma-separated list of <i>Method;Name</i> pairs.</li> <li>• The <i>Method</i> parameter is the name of the authentication method.</li> <li>• The <i>Name</i> parameter is the user to authenticate. If you do not specify a <i>Name</i> parameter, the name of the user being authenticated is used.</li> </ul>
umask	Determines file permissions. This value, along with the permissions of the creating process, determines a file's permissions when the file is created. The default is 022.
registry	Defines the authentication registry where the user is administered. It is used to resolve a remotely administered user to the local administered domain. This situation may occur when network services unexpectedly fail or network databases are replicated locally. Example values are files or NIS or DCE.
loginretries	<p>Defines the number of unsuccessful login attempts allowed after the last successful login before the system locks the account. The value is a decimal integer string. A zero or negative value indicates that no limit exists. Once the user's account is locked, the user will not be able to log in until the system administrator resets the user's <i>unsuccessful_login_count</i> attribute in the <i>/etc/security/lastlog</i> file to be less than the value of <b>loginretries</b>. To do this, enter the following:</p> <pre>chsec -f /etc/security/lastlog -s username -a \ unsuccessful_login_count=0</pre>
pwdwarntime	Defines the number of days before the system issues a warning that a password change is required. The value is a decimal integer string. A zero or negative value indicates that no message is issued. The value must be less than the difference of the <b>maxage</b> and <b>minage</b> attributes. Values greater than this difference are ignored, and a message is issued when the <b>minage</b> value is reached.

Attributes	Description
account_locked	<p>Indicates if the user account is locked. Possible values include:</p> <ul style="list-style-type: none"> <li>• <b>true</b>: The user's account is locked. The values yes, true, and always are equivalent. The user is denied access to the system.</li> <li>• <b>false</b>: The user's account is not locked. The values no, false, and never are equivalent. The user is allowed access to the system. Default</li> </ul>
minage	<p>Defines the minimum age (in weeks) a password must be before it can be changed. The value is a decimal integer string. The default is a value of 0, indicating no minimum age.</p>
SYSTEM	<p>Defines the system authentication mechanism for the user. The value may be an expression describing which authentication methods are to be used or it may be the keyword NONE.</p> <p>The <b>SYSTEM</b> mechanism is always used to authenticate the user, regardless of the value of the <b>auth1</b> and <b>auth2</b> attributes. If the <b>SYSTEM</b> attribute is set to NONE, authentication is only performed using the <b>auth1</b> and <b>auth2</b> attributes. If the <b>auth1</b> and <b>auth2</b> attributes are blank or ignored, as with the TCP socket daemons (<b>ftpd</b>, <b>rexecd</b> and <b>rshd</b>), no authentication will be performed.</p> <p>The method names <b>compat</b>, <b>files</b> and <b>NIS</b> are provided by the security library. Additional methods may be defined in the <code>/usr/lib/security/methods.cfg</code> file.</p> <p>Specify the value for <b>SYSTEM</b> using the following grammar:</p> <pre> "SYSTEM"      ::= EXPRESSION EXPRESSION    ::= PRIMITIVE                       "(" EXPRESSION ")"                       EXPRESSION OPERATOR EXPRESSION PRIMITIVE    ::= METHOD                         METHOD "[" "RESULT" "]" RESULT        ::= "SUCCESS"    "FAILURE"    "NOTFOUND"                      "UNAVAIL"     "*" OPERATOR      ::= "AND"        "OR" METHOD        ::= "compat"     "files"      "NONE"                        [a-z,A-Z,0-9]* </pre> <p>An example of the syntax is:</p> <pre>SYSTEM = "DCE OR DCE[UNAVAIL] AND compat"</pre>
maxage	<p>Defines the maximum age (in weeks) of a password. The password must be changed by this time. The value is a decimal integer string. The default is a value of 0, indicating no maximum age.</p>
maxexpired	<p>Defines the maximum time (in weeks) beyond the <b>maxage</b> value that a user can change an expired password. After this defined time, only an administrative user can change the password. The value is a decimal integer string. The default is -1, indicating no restriction is set. If the <b>maxexpired</b> attribute is 0, the password expires when the <b>maxage</b> value is met. If the <b>maxage</b> attribute is 0, the <b>maxexpired</b> attribute is ignored.</p>
minalpha	<p>Defines the minimum number of alphabetic characters that must be in a new password. The value is a decimal integer string. The default is a value of 0, indicating no minimum number.</p>

## Schema attributes

Attributes	Description
minother	Defines the minimum number of non-alphabetic characters that must be in a new password. The value is a decimal integer string. The default is a value of 0, indicating no minimum number.
logintimes	<p>Specifies the times, days, or both, the user is allowed to access the system. The value is a comma-separated list of entries of the following form:</p> <p>[!]:time-time          -or-          [!]day[-day][:time-time]          -or-          [!]date[-date][:time-time]</p> <p>The <i>day</i> variable must be one digit between 0 and 6 that represents one of the days of the week. A 0 (zero) indicates Sunday and a 6 indicates Saturday.</p> <p>The <i>time</i> variable is 24-hour military time (1700 is 5:00 p.m.). Leading zeroes are required. For example, you must enter 0800, not 800. The <i>time</i> variable must be four characters in length, and there must be a leading colon (:). An entry consisting of only a time specification applies to every day. The start hour of a time value must be less than the end hour.</p> <p>The <i>date</i> variable is a four digit string in the form <i>mmdd</i>. <i>mm</i> represents the calendar month and <i>dd</i> represents the day number. For example 0001 represents January 1. <i>dd</i> may be 00 to indicate the entire month, if the entry is not a range, or indicating the first or last day of the month depending on whether it appears as part of the start or end of a range. For example, 0000 indicates the entire month of January. 0600 indicates the entire month of June. 0311-0500 indicates April 11 through the last day of June.</p> <p>Entries in this list specify times that a user is allowed or denied access to the system. Entries not preceded by an ! (exclamation point) allow access and are called ALLOW entries. Entries prefixed with an ! (exclamation point) deny access to the system and are called DENY entries. The ! operator applies to only one entry, not the whole restriction list. It must appear at the beginning of each entry.</p>
mindiff	Defines the minimum number of characters required in a new password that were not in the old password. The value is a decimal integer string. The default is a value of 0, indicating no minimum number.
maxrepeats	Defines the maximum number of times a character can be repeated in a new password. Since a value of 0 is meaningless, the default value of 8 indicates that there is no maximum number. The value is a decimal integer string.
minlen	Defines the minimum length of a password. The value is a decimal integer string. The default is a value of 0, indicating no minimum length. The maximum value allowed is 8. This attribute is determined by the <b>minalpha</b> attribute value added to the <b>minother</b> attribute value. If the sum of these values is greater than the <b>minlen</b> attribute value, the minimum length is set to the result.
histexpire	Designates the period of time (in weeks) that a user cannot reuse a password. The value is a decimal integer string. The default is 0, indicating that no time limit is set.
histsize	Designates the number of previous passwords a user cannot reuse. The value is a decimal integer string. The default is 0.

Attributes	Description
pwdchecks	Defines the password restriction methods enforced on new passwords. The value is a list of comma-separated method names and is evaluated from left to right. A method name is either an absolute path name or a path name relative to /usr/lib of an executable load module.
dictionlist	<p>Defines the password dictionaries used by the composition restrictions when checking new passwords.</p> <p>The password dictionaries are a list of comma-separated, absolute path names that are evaluated from left to right. All dictionary files and directories must be write-protected from all users except root. The dictionary files are formatted one word per line. The word begins in the first column and terminates with a new-line character. Only 7-bit ASCII words are supported for passwords. If text processing is installed on your system, the recommended dictionary file is the /usr/share/dict/words file.</p>
default_roles	Specifies the default roles for the user. The Value parameter, a comma-separated list of valid role names, can only contain roles assigned to the user in the roles attribute. You can use the ALL keyword to signify that the default roles for the user are all their assigned roles.
fsize	Identifies the soft limit for the largest file a user process can create or extend.
cpu	Sets the soft limit for the largest amount of system unit time (in seconds) that a user process can use.
data	Identifies the soft limit for the largest process data segment for a user process.
stack	Specifies the soft limit for the largest process stack segment for a user process.
core	Specifies the soft limit for the largest core file a user process can create.
rss	Sets the soft limit for the largest amount of physical memory a user process can allocate. This limit is not enforced by the system.
nofiles	Sets the soft limit for the number of file descriptors a user process may have open at one time.
stack_hard	Specifies the largest process stack segment for a user process.
roles	Contains the list of roles for each user.
time_last_login	Specifies the number of seconds since the epoch (00:00:00 GMT, January 1, 1970) since the last successful login. The value is a decimal integer.
tty_last_login	Specifies the terminal on which the user last logged in. The value is a character string.
host_last_login	Specifies the host from which the user last logged in. The value is a character string.
unsuccessful_login_count	<p>Specifies the number of unsuccessful login attempts since the last successful login. The value is a decimal integer. This attribute works in conjunction with the user's loginretries attribute, specified in the /etc/security/user file, to lock the user's account after a specified number of consecutive unsuccessful login attempts. Once the user's account is locked, the user will not be able to log in until the system administrator resets the user's unsuccessful_login_count attribute to be less than the value of loginretries. To do this, enter the following:</p> <pre>chsec -f /etc/security/lastlog -s username -a \ unsuccessful_login_count=0</pre>

## Group attributes

---

The following table lists the group attributes:

Attributes	Description
users	Identifies a list of one or more users which are associated with group.
admin	Specifies whether administrative group or not.
registry	Specifies where the user or group identification information is administrated.
Group Name	Name of group
id	Group ID

## Provisioning policy attributes

---

This section lists the different policy attributes of AIX Connector.

## Account attributes

---

The following table lists the provisioning policy attributes for Create and Update Account:

Attributes	Description
User Name	(Only for Create Account) User ID on AIX computer that you want to use for connector operations.
id	User ID
pgrp	Primary group of user
home	Home directory of user
shell	Default shell of user
login	Indicates whether the user can log in to the system with the login command. Possible values are: <ul style="list-style-type: none"><li>• <b>true</b>: The user can log in to the system. Default.</li><li>• <b>false</b>: The user cannot log in to the system.</li></ul>
su	Indicates whether another user can switch to the specified user account with the <b>su</b> command. Possible values are: <ul style="list-style-type: none"><li>• <b>true</b>: Another user can switch to the specified account. Default</li><li>• <b>false</b>: Another user cannot switch to the specified account.</li></ul>
rlogin	Permits access to the account from a remote location with the <b>telnet</b> or <b>rlogin</b> commands. Possible values are: <ul style="list-style-type: none"><li>• <b>true</b>: The user account can be accessed remotely. Default</li><li>• <b>false</b>: The user account cannot be accessed remotely.</li></ul>

Attributes	Description
admin	Defines the administrative status of the user. Possible values are: <ul style="list-style-type: none"> <li><b>true</b>: The user is an administrator. Only the root user can change the attributes of users defined as administrators.</li> <li><b>false</b>: The user is not an administrator. Default</li> </ul>
sugroups	Lists the groups that can use the <b>su</b> command to switch to the specified user account. The <i>Value</i> parameter is a comma-separated list of group names, or a value of ALL to indicate all groups. An ! (exclamation point) in front of a group name excludes that group. If this attribute is not specified, all groups can switch to this user account with the <b>su</b> command.
admgroups	Lists the groups the user administers. The <i>Value</i> parameter is a comma-separated list of group names. For additional information on group names, see the <b>adms</b> attribute of the <i>/etc/security/group</i> file.
umask	Determines file permissions. This value, along with the permissions of the creating process, determines a file's permissions when the file is created. The default is 022.
default_roles	Specifies the default roles for the user. The <i>Value</i> parameter, a comma-separated list of valid role names, can only contain roles assigned to the user in the <i>roles</i> attribute. You can use the ALL keyword to signify that the default roles for the user are all their assigned roles.
Password	Initial password for newly created user account.

## Group attributes

---

The following table lists the provisioning policy attributes for Create and Update Group:

Attributes	Description
Group Name	(Only for create group) Name of group
users	Identifies a list of one or more users which are associated with group.
Id	Group ID

## Additional information

---

This section describes the additional information related to the AIX Connector.

### Unstructured Target Collector

---

AIX uses a data structure which requires the configuration in the **Unstructured Targets** tab to collect targeted data and correlate it with account **identityAttribute** for Accounts and group **identityAttribute** for Account Groups. For more information on the **Unstructured Targets** tab, see “Unstructured Targets Tab” section of the *SailPoint IdentityIQ User’s Guide*.

For AIX target permission, the Unstructured Targets functionality will be enabled if **UNSTRUCTURED\_TARGETS** feature string is present in the application.

## Troubleshooting

Multiple target sources can be specified and configured for an application which supports unstructured targets. This will be useful for applications which want to fetch resource information from multiple target sources.

AIX Target Collector support aggregation of file/directories under specified file system path(s). Only direct access permissions will be correlated Users and Groups. For UNIX platforms direct access means ownership of file or directory.

**Table 1—Unstructured Target Configuration parameters**

Attributes	Description	Possible values
Unix File System Path(s)*	Absolute path(s) which are to be scanned for resources.	Multiple paths can be mentioned with comma separated values. For example, /etc/tmp
Application Name*	Name of the application with which Unstructured Target will be correlated.	

**Note:** Attributes marked with \* sign are the mandatory attributes.

**Note:** If Unstructured Configuration is configured before upgrading to version 7.1 from version 6.0 Patch 5 or 6.0 Patch 6, then update the configuration and specify the Connector Application Name.

## Rule configuration parameters

The rule configuration parameters are used to transform and correlate the targets.

**Correlation Rule:** The rule used to determine how to correlate account and group information from the application with identity cubes.

**Note:** For version 6.2 onwards, the default schema does not have correlation keys defined. Update correlation rule in Unstructured Target Configuration accordingly.

## Provisioning related parameters

Select the settings for provisioning to the box.

- **Override Default Provisioning:** Overrides the default provisioning action for the collector.
- **Provisioning Action:** The overriding provisioning action for the collector.

# Troubleshooting

---

## 1 - Aggregation fails on AIX 5.3 when number of users exceeds 1000

Aggregation fails on AIX version 5.3.

**Resolution:** Set the **LDR\_CNTRL** environment variable as follows in run scripts of default shell for AIX connector administrator user:

```
export LDR_CNTRL =MAXDATA=0x60000000
```

Increase the number of stack segments from 1 to 8 depending on the number of Users, Groups and Connections present on the AIX computer.

```
MAXDATA=0xN0000000@DSA
```

where  $N$  is the number of stack segments.

## 2 - Test connection fails for managed systems

Test Connection fails for managed systems with the following error when SSH login prompt appears with some delay:

```
Test Connection failed. Login failed. 'sh' is not set on your machine. Please set 'sh'
```

The above error occurs when connector tries to login to target managed system with SSH and execute the **sh** command. The **sh** command fails because of delay on target managed system for SSHLogin prompt to appear.

**Resolution:** To resolve this issue, tune the following time out parameters according to your need in the Application Debug page:

- **SSHLoginTimeout:** Default value: 1000 ms

This time out parameter is responsible to tune time taken to connect to target host through ssh.

This is also effective in tuning time taken between actual login process start and actual appearance of first prompt.

- **SSHTimeOut:** Default value: 120000 ms

This time out parameter is responsible to tune maximum time for which a ssh command execution should be allowed. After this time out even if the command execution is in progress on target host, the connection will be dropped out and the operation will be timed out.

## 3 - Aggregation/test connection fails with timeout error

Aggregation/test connection fails with the following timeout error:

Exception during aggregation. Reason: sailpoint.connector.ConnectorException: Account aggregation failed. Timeout occurred.

**Resolution:** Change the value of the **SSHLoginTimeout (in millisecond)** application attribute as per your requirement in the debug page of the application:

```
<entry key="SSHLoginTimeout" value="1000" />
```

## 4 - After target aggregation resources are not getting correlated with Account Groups

After target aggregation the resources are not getting correlated with Account Groups.

**Resolution:** Ensure that your correlation rule populates "Correlator.RULE\_RETURN\_GROUP\_ATTRIBUTE" as follows:

```
....  
if ( isGroup ) {  
    returnMap.put(Correlator.RULE_RETURN_GROUP_ATTRIBUTE, "nativeIdentity");  
    returnMap.put(Correlator.RULE_RETURN_GROUP_ATTRIBUTE_VALUE, nativeId);  
}  
....
```

## 5 - Test connection fails for key based authentication with an error

Test connection fails for key based authentication with the following error.

```
Login failed. Error while connecting to host:<hostname>. Cannot read key file.
```

## Troubleshooting

**Resolution:** Perform the following steps to generate/convert private/public keys in format which is supported by UNIX direct connectors.

- Generate keys using open ssl. This method can be used for any version of SSH.

- a. Create private key using the following command:

```
openssl <gendsa/genrsa> -des3 -out <private_key> 1024
```

- b. Change the permission on the <private\_key> file as follows:

```
chmod 0600 <private_key>
```

- c. Create public key from private\_key

```
ssh-keygen -y -f <private_key> > <public_key>
```

- d. Use the <private\_key> and <public\_key> files for authentication.

- Generate keys using ssh-keygen. (OpenSSH 5.8 or above)

- a. Create private and public key using the following command

```
ssh-keygen -t <dsa/rsa> -b 1024
```

By default files with name id\_dsa/id\_rsa and id\_dsa.pub/id\_rsa.pub will be created.

- b. Convert <private key> to have DES-EDE3-CBC encryption algorithm by using the following command:

```
openssl <dsa/rsa> -in <private_key> -out <new_private_key> -des3
```

- c. Change the permission on the <new\_private\_key> file as follows:

```
chmod 0600 <new_private_key>
```

- d. Create public key file using the new private key as follows:

```
ssh-keygen -y -f <new_private_key> > <new_public_key>
```

- e. Use the <new\_private\_key> and <new\_public\_key> files for authentication.

## 6 - Test connection fails with an error when sudo user is configured for public key authentication

Test connection fails with the following error when sudo user is configured for public key authentication:

Test SSH communication failed over host: xxxxxxxx. Error while executing command: sudo -p %SAILPOINTSUDO echo TestConnection over host: xxxxxxxx. Invalid sudo user password.

**Resolution:** Verify the sudo user's password specified in application configuration, password should be correct for certificate based authentication.

# Chapter 3: SailPoint Azure Active Directory Connector

---

The following topics are discussed in this chapter:

Overview .....	49
Supported features .....	50
Pre-requisites .....	50
Administrator permissions .....	51
Configuration parameters .....	52
Additional configuration parameters .....	52
Schema attributes .....	53
Account attributes .....	53
Group attributes .....	54
Provisioning Policy attributes .....	55
Create Account Policy .....	55
Create Group Policy .....	56
Update Group Policy .....	57
Additional information .....	57
Managing licenses .....	57
Connector Reconfigure .....	57

## Overview

---

SailPoint Azure Active Directory connector manages the users and groups in Windows Azure Active Directory. SailPoint Azure Active Directory Connector can be used to manage Users and Groups as Windows Azure Active Directory is the directory for all cloud based organizational Microsoft directory services including Microsoft Office365.

SailPoint Azure Active Directory connector can also be used to provision users into a federated domain in Azure Active Directory.

The SailPoint Azure Active Directory connector uses **Azure AD Graph API** to manage users, groups and licenses.

## Supported features

---

The Azure Active Directory connector supports the following features:

- Account Management
  - Aggregation, Get Account, Partitioning Aggregation
  - Create user in Azure Activity Directory,
  - Create user in a federated domain in Azure Active Directory.
  - Update, Delete users in Azure Active Directory
  - Enable\Disable users,
  - Set password
  - Add\Remove Entitlements:
    - Add\Remove individual license plans
    - Add\Remove license packs
    - Add\Remove Roles
    - Add\Remove user's group membership
- Account - Group Management
  - Aggregation, get operation for Security groups and Mail Enabled Security Groups
  - Create, Update Security Groups
  - Delete Security Groups, Mail Enabled Security Groups
- Other
  - Supports executing native before/after scripts for provisioning requests

### References

- “Appendix C: Partitioning Aggregation”
- “Appendix E: IQService”

## Pre-requisites

---

To use Graph API, a client application must be registered on the Azure management portal. This application is responsible for calling Web APIs on behalf of the connector. The application's client ID and client secret key are required while configuring the application. To register an application on Azure, perform the following:

1. Sign in to **Azure Management Portal**.
2. Click on **Active Directory** in the left pane and select the directory domain which is to be managed.
3. Click on **Applications** tab and select ADD option at the bottom of the page.  
A new window would open.
4. Click on **Add an application my organization is developing** and enter the name for the application.  
For example, SailPointAzureADManagement.
5. Select Web application and/or Web API.

6. For the sign-on URL, enter the base URL. For example, `https://localhost:44320`  
For Application ID URL, enter url like, `domain name>/GraphWebapp`  
Connector does not use these URLs, the above value are just place holders and do not impact connector functionality.
7. Select **complete** mark.

After registering application, perform the following to get client ID, secret key and to provide permissions to application:

1. Select the application from the list and click on **Configure** tab.  
This displays application properties. Locate Client Id property and note its value.
2. Add a key: Select long duration (couple of years). Saving this page displays the Key. Record this key as it is not retrievable again on Azure.

## Administrator permissions

---

Following permissions must be granted to the client application created in Azure:

- Read Directory data
- Read and Write Directory data

To grant permissions to the client application:

- Click on the client application in Azure Active Directory console.
- Select **Configure** tab and search for the **Permissions to other applications** section on the page.
- Verify **Read Directory Data** and **Read and Write Directory Data** from the application permissions drop down.

Above permissions do not allow connector to manage users with administrative roles. To manage such users, the application created on Azure must have **Company Administrator** role assigned. This role can be assigned via PowerShell commands. Following are the prerequisite for executing the PowerShell commands.

**Note:** These prerequisites are not required for the connector to function. These can be installed on any system for temporary use to give required role to the application on Azure.

- Microsoft Online Services Sign-In Assistant for IT Professionals RTW
- Windows Azure Active Directory Module for Windows PowerShell

After installing the pre-requisites, open **Windows Azure Active Directory Module for Windows PowerShell** console and execute the following commands:

- Connect-msolservice, press enter, provide Azure administrator credentials.
- Execute `Get-MsolServicePrincipal | ft DisplayName, AppPrincipalId -Autosize`
- Locate your application name and copy the **ObjectId** value.
- Execute `$ClientObjID = <copied objectId of the application in the previous step>`
- Execute `$webApp = Get-MsolServicePrincipal -AppPrincipalId $ClientObjID`
- Execute `Add-MsolRoleMember -RoleName "Company Administrator" -RoleMemberType ServicePrincipal -RoleMemberObjectId $webapp.ObjectId`

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Azure Active Directory connector uses the following connection parameters:

Attributes	Description
Azure AD application client ID*	ID of the application created on the Azure Active Directory for using Graph REST API.
Azure AD application client secret key*	Client secret of the Azure Active Directory application.
Azure AD domain name*	Name of the Azure Active Directory domain to be managed. For example, contoso.onmicrosoft.com
Page Size	Number of records per page. Default: 500
IQService Host	Host name of the system where IQService is installed.
IQService Port	Port number used by the IQService.

**Note:** To enable native before/after script execution for provisioning requests, IQService Host and IQService Port parameters must be configured.

## Additional configuration parameters

---

Attributes	Description
createAccountTimelag	Time in seconds to wait after create account and before calling get account. Default: 20 seconds  For example, <entry key="createAccountTimelag" value="20">
maxReadTimeout	Time in seconds to wait for getting response from the REST call, in the read operation, before operation gets timed out. Default: 180 seconds.  For example, <entry key=" maxReadTimeout" value="200">
maxRetryCount	Indicates the number of time read operation must be retried on retry errors for the read operations. Default: 5  For example, <entry key=" maxRetryCount" value="6">
retryableErrorsOnAgg	List of error which must be retried if occurred during aggregation or get operation. Type: List of strings

Attributes	Description
userPartitions	<p>List of filters to be applied during account aggregation to limit set of data.</p> <p>For more information, see "Partitioning Aggregation for Azure Active Directory Connector" on page 568.</p>
groupPartitions	<p>List of filters to be applied during account-group aggregation to limit set of data.</p> <p>For more information on how to form filters, see "Partitioning Aggregation for Azure Active Directory Connector" on page 568.</p>

## Schema attributes

---

This section describes the different schema attributes.

**Note:** In addition to the schema attributes listed in the following tables, the connector supports managing the extended attributes that are registered on the client application on Azure.

### Account attributes

---

The following table lists the account attributes:

Name	Description
accountEnabled	True if the account is enabled; otherwise, false
assignedLicenses	List of the licenses that are assigned to the user
assignedPlans	Plans that are assigned to the user (Entitlement).
city	City in which the user is located.
country	Country/region in which the user is located.
department	Name for the department in which the user works.
dirSyncEnabled	Indicates whether this object was synced from the on-premises directory.
disabledPlans	Plans that are not assigned to user.
displayName	Name displayed in the address book for the user.
facsimileTelephoneNumber	Telephone number of the user's business fax machine.
givenName	First name of user.
groups	Groups assigned to a user (Entitlement).
immutableId	Property used to associate an on-premises Active Directory user account to their Azure AD user object.
jobTitle	User's job title.

## Schema attributes

Name	Description
lastDirSyncTime	Indicates the last time at which the object was synchronized with the on-premises directory.
mail	The SMTP address for the user. For example, john@contoso.onmicrosoft.com
mailNickname	The mail alias for the user.
manager	Manager of the user. (Type: String) By default this attribute is not added to the schema for performance optimization.
mobile	Primary cellular telephone number for the user.
objectId	Unique identifier for the user.
onPremisesSecurityIdentifier	Contains the on-premises security identifier (SID) for the user that was synchronized from on-premises to the cloud.
otherMails	A list of additional email addresses for the user.
passwordPolicies	Specifies password policies for the user.
physicalDeliveryOfficeName	Office location in the user's place of business.
postalCode	ZIP OR postal code for the user's postal address.
preferredLanguage	Preferred written or spoken language for a person.
proxyAddresses	Proxy addresses. For example, [ "SMTP: bob@contoso.com" , "smtp: bob@sales.contoso.com" ]
roles	Administrator Role assigned to user (Entitlement).
sipProxyAddress	Specifies the voice over IP (VOIP) session initiation protocol (SIP) address for the user.
state	The state or province in the user's address.
streetAddress	The street address of the user's place of business.
surname	Last name of the user.
telephoneNumber	Primary telephone number of the user's place of business.
usageLocation	A two letter country code indicating usage location.
userPrincipalName	The user principal name (UPN) of the user.
userType	Type of the user.

## Group attributes

---

Name	Description
description	Description for the group.
dirSyncEnabled	Indicates whether this object was synced from the on-premises directory.
displayName	Display name for the group.

Name	Description
lastDirSyncTime	Indicates the last time at which the object was synced with the on-premises directory.
mail	SMTP address for the group.
mailEnabled	Specifies whether the group is mail-enabled
mailNickname	The mail alias for the group.
objectId	Group ID.
onPremisesSecurityIdentifier	Contains the on-premises security identifier (SID) for the group that was synchronized from on-premises to the cloud.
owners	Owner of the group. By default not present in the schema, Type: String, Multi-Valued
proxyAddresses	Proxy addresses of the group.
securityEnabled	Specifies whether the group is a security group.

## Provisioning Policy attributes

---

This section lists different policy attributes for Azure Active Directory Connector.

**Note:** The attributes marked with \* sign are required attributes.

### Create Account Policy

---

Following table describes various attributes in the create account policy.

Attribute	Description
userPrincipalName*	user principal name (UPN) of the user. For example, jeff@contoso.onmicrosoft.com
password*	Password for the new user.
displayName*	Display name of the user.
mailNickname*	Mail alias for the user.
accountEnabled	Set it to false to create disabled account. Default: True
forceChangePasswordNextLogin	If true, asks user to change password on next login. Default: True
department	Department in which the user works.
jobTitle	User's job title.
isFederatedDomain	Set it true to create federated domain user. If this is checked and <b>immutableId</b> is not set then random <b>immutableId</b> value will be used.

## Provisioning Policy attributes

Attribute	Description
immutableId	This property is used to associate an on-premises Active Directory user account to their Azure AD user object; Populate this attribute with objectGUID of account from on-premises Active Directory to create federated user synchronized with on-premises Active Directory user.
passwordPolicies	Specifies password policies for the user  For example: DisablePasswordExpiration, DisableStrongPassword
otherMails	Additional email addresses for the user.
givenName	First name of the user.
surname	Surname of the user.
usageLocation	A two letter country code (ISO standard 3166). Required for users that will be assigned licenses.
country	The country/region in which the user is located. For example, <b>US</b> or <b>UK</b>
state	The state or province in the user's address.
city	The city in which the user is located.
streetAddress	The street address of the user's place of business.
postalCode	The postal code for the user's postal address.
physicalDeliveryOfficeName	The office location in the user's place of business.
preferredLanguage	Preferred language for the user. Should follow ISO 639-1 Code. For example, <b>en-US</b>
telephoneNumber	Primary telephone number of the user's place of business.
mobile	Primary cellular telephone number for the user.
facsimileTelephoneNumber	Telephone number of the user's business fax machine.
userType	A string value that can be used to classify user types in your directory, such as <b>Member</b> and <b>Guest</b> .

## Create Group Policy

Following table describes various attributes in the create group policy.

Attribute	Description
displayName*	Display name of the group.
mailNickname*	The mail alias for the group.

## Update Group Policy

---

Following table describes various attributes in the update group policy.

Attribute	Description
description	Description of the group.
owners	Owner of the group. Read only.
mailEnabled	True if it is mail enabled security group. Read only.

## Additional information

---

This section describes the additional information related to the Azure Active Directory Connector.

### Managing licenses

---

Azure Active Directory Connector supports assigning different Azure services licenses to the users. Connector provides options to assign license either by individual plan or as a whole license pack.

- **Assigning license plan:** Office 365 license pack consist of licenses for individual services. For example, Exchange Online, SharePoint Online and so on.
- The connector models **assignedPlans** attribute from account schema as an entitlement. It can be requested as an entitlement during **Create** or **Update** operations for Identities.
- **Assigning license pack:** To assign license pack, set **assignedLicenses** attribute from account schema as **Managed, Entitlement, Multi-Valued**, So that it request able as an entitlement.

**Note:** It is recommended that 'assignedPlans' or 'assignedLicenses' must be promoted as an entitlement to avoid conflicts.

### Connector Reconfigure

---

Existing Microsoft Office365 application can be reconfigured to Azure Active Directory application to preserve the data present in the IdentityIQ.

## **Additional information**

# Chapter 4: SailPoint BMC Remedy Connector

---

The following topics are discussed in this chapter:

Overview .....	59
Supported features .....	59
Supported Managed Systems .....	60
Pre-requisites .....	60
Administrator permission .....	60
Configuration parameters .....	60
Schema attributes .....	60
Account attributes .....	60
Group attributes .....	61
Provisioning policy attributes .....	62
Create account attributes .....	62
Create group attributes .....	62
Update policies .....	62
Additional information .....	62
Enable/Disable Account .....	63
Troubleshooting .....	63

## Overview

---

SailPoint BMC Remedy Connector manages the accounts and groups contained in BMC Remedy Action Request System.

## Supported features

---

SailPoint BMC Remedy Connector supports the following features:

- Account Management
  - Manage BMC Remedy Users as Accounts
  - Aggregation, Refresh Account, Pass Through Authentication
  - Create, Update, Delete
  - Enable, Disable, Change Password
  - Add/Remove Entitlements
- Account - Group Management
  - Manage BMC Remedy Groups as Account - Groups
  - Aggregation, Refresh Group
  - Create, Update, Delete

## Configuration parameters

### References

- “Enable/Disable Account” on page 63

## Supported Managed Systems

---

- BMC Remedy Action Request System Server version 9.1
- BMC Remedy Action Request System Server version 9.0

## Pre-requisites

---

- You must copy the `arapi<v>.jar` file from the location where the server is installed (`install-Folder\BMC Software\ARSystem\midtier\WEB-INF\lib`) to the lib folder of the connector installation (`\webapps\identityiq\WEB-INF\lib`).
- Add the location of `arapi<v>.jar` file to the CLASSPATH system variable (`\webapps\identityiq\WEB-INF\lib\arapi<v>.jar`) of the computer where IdentityIQ installed.
- Provide the appropriate read and write permissions to the Administrator to perform the user and group provisioning operations.

## Administrator permission

---

The Application User should be a member of the **Administrator** group.

## Configuration parameters

---

The following table lists the configuration parameters of BMC Remedy Connector:

Parameters	Description
Remedy Server name or IP Address	IP address of the computer on which the Remedy server is installed.
Administrator Name	Name of the Remedy administrator.
Administrator Password	Password of the administrator.
Server Port	Remedy Server port number.

## Schema attributes

---

This section describes the different schema attributes.

## Account attributes

---

The following table lists the account attributes:

Attributes	Description
RequestID	RequestID of the user.
LoginName	Remedy login name.
ForcePasswordChangeOnLogin	Set to Yes if the user should be asked to change his password on next login else to No.
FullName	Full name of the user.
Status	Status of the user.
AccountDisabledDate	Account disabled date of user.
ApplicationLicense	Application license of user.
AppliedDaysAfterExpirationUntilDisablement	Applied days after expiration until disablement of user.
AppliedNewUserMustChangePassword	Is set to Yes if the new user must change password.
AppliedNo.DaysbeforeExpiration	The number of days before expiration.
AppliedNumberofWarningDays	The number of warning days.
AppliedPasswordEnforcementEnabled	Is set to Yes if password enforcement is enabled.
Creator	Creator of the user.
LastModifiedBy	Name of the user who last modified the user.
LicenseType	License Type of user.
UniquelIdentifier	Unique Identifier of the user.
Groups	Groups connected to the user.

## Group attributes

---

The following table lists the group attributes:

Attributes	Description
RequestID	RequestID of group.
Comments	Comments about group.
GroupCategory	Category of group.
GroupID	ID of group.
GroupName	Name of the group.
GroupType	Type of group.
LastModifiedBy	Name of the user who last modified the group.
LongGroupName	Long name of the group.
ParentGroup	Parent group of the group.
Status	Status of the group.
UniquelIdentifier	Unique Identifier of group.

## Provisioning policy attributes

---

This section lists the different policy attributes of BMC Remedy Connector.

### Create account attributes

---

The following table lists the provisioning policy attributes for Create Accounts:

Attributes	Description
Login Name	Remedy login name of the user.
Full Name	Full name of the user.
Force Password Change On Login	Is set to Yes if the user should be asked to change his password on next login.
License Type	License Type of the user.
Password	Password of the user.

### Create group attributes

---

The following table lists the provisioning policy attributes for Create Group:

Attributes	Description
Group Name	Name of the group to be created.
Group ID	ID of the group. It should be a numeric value.
Group Type	Type of the group.
Long Group Name	Long name of the group.
Group Category	Category of the group. If the category of the group is <b>Computed</b> , <b>ComputedGroupDefinition</b> needs to be added in the provisioning policy.

### Update policies

---

The following table lists the attributes for enable/disable a user:

Attributes	Description
ResetPassword	The new password to be set.

## Additional information

---

This section describes the additional information related to the BMC Remedy Connector.

## Enable/Disable Account

---

For disabling a user, a password not known to the user should be provided by the administrator. The **Status** attribute of the user will be set to **Disabled**. All users which have a status other than **Current** will be marked as **Disabled**.

For enabling a user, a password should be provided by the administrator which can be communicated to the user after successful password change. The **Status** attribute of the user will be set to **Current**.

## Troubleshooting

---

- When an attribute is to be added to the schema, the attributes ID should be added as an **internalName** of the attribute in the schema.
- When an attribute (which is not present in the schema) is to be added to the provisioning policy, the ID of the attribute should be provided as the **name** of the attribute.
- While creating a Remedy Group having GroupType value **Computed**, ensure that the **ComputedGroupDefinition** attribute is added to the provisioning policy.

For example, `<Field displayName="ComputedGroupDefinition" name="121" type="string"/>`

## **Troubleshooting**

# Chapter 5: SailPoint BMC Remedy IT Service Management Suite Connector

---

The following topics are discussed in this chapter:

Overview .....	65
Supported features .....	65
Supported Managed Systems .....	66
Pre-requisites .....	66
Administrator permission .....	66
Configuration parameters .....	66
Schema attributes .....	67
Account attributes .....	67
Group attributes .....	68
Provisioning policy attributes .....	68
Create account attributes .....	68
Create group attributes .....	69
Update policies .....	70
Additional information .....	70
Enable/Disable Account .....	70
Add Entitlement operation for ITSM .....	70
Troubleshooting .....	71

## Overview

---

SailPoint BMC Remedy IT Service Management Suite Connector manages the accounts and groups contained in a BMC Remedy IT Service Management Suite (ITSM).

## Supported features

---

SailPoint BMC Remedy ITSM Connector supports the following features:

- Account Management
  - Manage BMC Remedy ITSM Users as Accounts
  - Aggregation, Refresh Account, Pass Through Authentication
  - Create, Update, Delete
  - Enable, Disable, Change Password
  - Add/Remove Entitlements

## Configuration parameters

- Account - Group Management
  - Manage BMC Remedy ITSM Support Groups as Account - Groups
  - Aggregation, Refresh Group
  - Create, Update, Delete

### References

- “Add Entitlement operation for ITSM” on page 70
- “Enable/Disable Account” on page 70

## Supported Managed Systems

---

- BMC Remedy IT Service Management Suite version 9.1
- BMC Remedy IT Service Management Suite version 9.0

## Pre-requisites

---

1. You must copy the **arapi<v>.jar** file from the location where the server is installed (*installFolder\BMC Software\ARSystem\midtier\WEB-INF\lib*) to the lib folder of the connector installation (*\webapps\identityiq\WEB-INF\lib*).
2. Add the location of **arapi<v>.jar** file to the CLASSPATH system variable (*\webapps\identityiq\WEB-INF\lib\arapi<v>.jar*) of the computer where IdentityIQ installed.
3. Provide the appropriate read and write permissions to the Administrator to perform the user and group provisioning operations.

## Administrator permission

---

The Application User must be a member of the **Administrator** group.

## Configuration parameters

---

The following table lists the configuration parameters of Remedy ITSM Connector:

Parameters	Description
Remedy Server name or IP Address	IP address of the computer on which the Remedy ITSM server is installed.
Administrator Name	Name of the Remedy ITSM administrator.
Administrator Password	Password of the administrator.
Server Port	Remedy ITSM Server port number.

# Schema attributes

---

This section describes the different schema attributes.

## Account attributes

---

The following table lists the account attributes:

Attributes	Description
PersonID	PersonID of the user
RemedyLoginID	Remedy Login Name
FirstName	First name of the user
LastName	Last name of the user
InternetEmail	Internet Email of user
Status	Status of the user
AccountingNumber	Accounting Number of user
ClientSensitivity	Sensitivity of client
ClientType	Type of Client
Company	Company of user
CorporateID	Corporate ID of user
BusinessPhoneNumber	Business Phone Number of the user
FullTextLicenseType	Full Text License Type of user
JobTitle	Job Title of user
LastModifiedBy	Name of the user who last modified the user attributes
LicenseType	Type of License
Region	Region information
Site	Site information
SiteAddress	Site Address information
SiteGroup	Site Group information
Submitter	Name of the submitter
SupportStaff	If user is part of Support Staff
VIP	If user is VIP
UnrestrictedAccess	If user has unrestricted access
Groups	Groups connected to user

## Group attributes

---

The following table lists the group attributes:

Attributes	Description
SupportGroupID	Support Group ID
Company	Support Company information
Description	Group description
DisableGroupNotification	If group notification is disabled
GroupNotificationEmail	Group notification email id
instanceId	Instance id of group
LastModifiedBy	Name of the user who last modified the group
Status	Status of the group
Creator	Creator of the group
SupportGroupName	Name of the group
SupportGroupRole	Support Group role name
SupportOrganization	Support Group organization name
UsesOLA	If the group uses OLAs
UsesSLA	If the group uses SLAs
VendorGroup	If group is a Vendor Group
OnCallGroup	If group is a On Call Group

## Provisioning policy attributes

---

This section lists the different policy attributes of Remedy ITSM Connector.

### Create account attributes

---

The following table lists the provisioning policy attributes for Create Accounts:

Attributes	Description
FirstName	First name of the user
LastName	Last name of the user

Attributes	Description
ClientType	Type of Client. Following are the allowed values: <ul style="list-style-type: none"> <li>• Office-Based Employee</li> <li>• Field-Based Employee</li> <li>• Home-Based Employee</li> <li>• Contractor</li> <li>• Customer</li> <li>• Prospect</li> <li>• Vendor</li> </ul>
ClientSensitivity	Sensitivity of Client. Following are the allowed values: <ul style="list-style-type: none"> <li>• Sensitive</li> <li>• Standard</li> </ul>
VIP	Following are the allowed values: <ul style="list-style-type: none"> <li>• Yes</li> <li>• No</li> </ul>
Company	Company name of the user
BusinessPhoneNumber	Business phone number of the user
RemedyLoginID	Remedy login ID
Password	Remedy Password for the login ID
SupportStaff	Following are the allowed values: <ul style="list-style-type: none"> <li>• Yes</li> <li>• No</li> </ul> <p>If value is Yes, AssignmentAvailability attribute must be added. Allowed values: Yes or No.</p>
UnrestrictedAccess	Following are the allowed values: <ul style="list-style-type: none"> <li>• Yes</li> <li>• No</li> </ul>

## Create group attributes

---

The following table lists the provisioning policy attributes for Create Group:

Attributes	Description
SupportCompany	Support Company name of group
SupportOrganization	Support organization of group
SupportGroupName	Support group name
SupportGroupRole	Support Group role name

## Additional information

Attributes	Description
VendorGroup	Following are the allowed values: <ul style="list-style-type: none"><li>• Yes</li><li>• No</li></ul>
OnCallGroup	Following are the allowed values: <ul style="list-style-type: none"><li>• Yes</li><li>• No</li></ul>

## Update policies

The following table lists the attributes for different update policies:

Attributes	Description
<b>Enable/Disable a user</b>	
ResetPassword	The new password to be set.
<b>Create an ITSM Account and Group Connection</b>	
AC_1000000017	Full name of the user.
AC_4	Remedy Login ID of the user.
AC_1000000401	Support Group Association Role name.

**Note:** The connection attributes should have 'AC\_' prefixed to the field id of the attribute.

## Additional information

This section describes the additional information related to the BMC Remedy ITSM Suite Connector.

### Enable/Disable Account

For disabling a user, a password not known to the user should be provided by the administrator. The **Profile Status** attribute of the user will be set to **Obsolete**. All users which have a status other than **Enabled** will be marked as **Disabled**.

For enabling a user, a password should be provided by the administrator which can be communicated to the user after successful password change. The **Profile Status** attribute of the user will be set to **Enabled**.

### Add Entitlement operation for ITSM

To add a user to a group in BMC Remedy ITSM, there are some mandatory attributes to be provided which are a part of the connection between the user and the group. Hence, for Remedy ITSM, an entitlement will have mandatory attributes which will be a part of the update provisioning policy. All entitlements added will have the same connection attributes.

# Troubleshooting

---

- When an attribute is to be added to the schema, the attributes ID should be added as an **internalName** of the attribute in the schema.
- When an attribute (which is not present in the schema) is to be added to the provisioning policy, the ID of the attribute should be provided as the **name** of the attribute.
- For connection attributes, ensure that the ID of the attribute is prefixed with **AC\_**.
- While creating an ITSM Account having SupportStaff value **Yes**, ensure that the **AssignmentAvailability** attribute is added to the provisioning policy.

For example,

```
<Field displayName="AssignmentAvailability" name="1000000346"
reviewRequired="true" type="string">
    <AllowedValues>
        <String>Yes</String>
        <String>No</String>
    </AllowedValues>
</Field>
```

- For account creation in BMC Remedy ITSM Suite version 7.5.00.001 required mandatory attribute **InternetEmail**.

## **Troubleshooting**

# Chapter 6: SailPoint DB2 Windows Connector

---

The following topics are discussed in this chapter:

Overview .....	73
Supported features .....	74
Supported Managed Systems .....	74
Pre-requisites .....	74
Administrator permissions .....	74
Configuration parameters .....	75
Schema Attributes .....	75
Account attributes .....	75
Roles attributes .....	76
Provisioning Policy attributes .....	77
Additional information .....	77
Create user .....	78
Delete user .....	78
Delete Role .....	78
Troubleshooting .....	78

## Overview

---

IBM DB2 is a relational model database server developed by IBM. As a database, it is a software product whose primary function is to store and retrieve data as requested by other software applications. Following are the main products of the DB2 family:

- DB2 for LUW (Linux, Unix, and Windows)
- DB2 for z/OS (mainframe)
- DB2 for iSeries (formerly OS/400)

The DB2 LUW product runs on multiple Linux, UNIX distributions (such as Red Hat Linux, SUSE Linux, AIX, HP/UX, and Solaris) and Windows systems. DB2 also powers the IBM InfoSphere Warehouse edition, which is DB2 LUW with DPF (Database Partitioning Feature), a massive parallel share-nothing data warehousing architecture.

Basically DB2 server manages all access on DB2 server for Windows users and the Windows authentication are also used to login to DB2 server.

**Note:** **SailPoint DB2 Connector supports DB2 Windows flavor only.**

SailPoint DB2 Windows Connector manages the following entities of DB2 Server:

- Database Users
- Roles

## **Supported features**

---

SailPoint DB2 Windows Connector supports the following features:

- Account Management
  - Manage DB2 Users as Accounts
  - Aggregate, Refresh Account, Discover Schema
  - Create, Update, Delete
  - Add/Remove Entitlement
- Account - Group Management
  - Manage DB2 Roles as Account Group
  - Aggregate, Refresh Group
  - Delete
- Permissions Management
  - Permissions directly assigned to accounts and groups as direct permissions during account and group aggregation.
  - Automated revocation of the aggregated permissions.

## **Supported Managed Systems**

---

SailPoint DB2 Windows Connector supports the following versions of DB2 Server:

- IBM DB2 Enterprise Server version 11.1
- IBM DB2 Enterprise Server version 10.5
- IBM DB2 Enterprise Server version 10.1

## **Pre-requisites**

---

The compatible DB2 Windows JDBC drivers must be used in the classpath of IdentityIQ for connecting to DB2 Server. For example, db2jcc.jar

## **Administrator permissions**

---

The Administrator login must have the SYSADM (Authority) as the minimum privilege and must be able to perform the following operations on DB User and Roles:

- Search
- Create
- Update
- Delete

**Note:** To run the **CREATE ROLE rolename** and **DROP ROLE rolename** query in the following DB2 versions, the respective specified authorities are required:

- 10.1 and 10.5: SECADM, SYSCTRL, or SECADM authority

# Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The DB2 Windows Connector uses the following connection attributes:

Attribute	Description
URL*	A valid URL of DB2 Server in the following format: <code>jdbc:db2:// [serverName] [:portNumber] / [databaseName]</code> where: <ul style="list-style-type: none"><li>• <b>jdbc:db2://</b> is known as the sub-protocol and is constant.</li><li>• <b>serverName</b>: address of the server to connect to. This could be a DNS, IP address, a localhost, or 127.0.0.1 for the local computer.</li><li>• <b>portNumber</b>: is the port to connect to on <i>serverName</i>. Default is 50000.</li><li>• <b>databaseName</b>: is the database to which you want to connect.</li></ul>
User*	A Domain or Local login to the operating system through which you want to connect the DB2 Server. The login name should have minimum privileges to log in to the DB2 Server.
Password*	Authentication details of login and should have a valid password of that login.
JDBC Driver*	Name of the driver class supported by JDBC. For example, <code>com.ibm.db2.jcc.DB2Driver</code>

# Schema Attributes

---

This section describes the different schema attributes.

## Account attributes

---

The following table lists the account attributes:

Attribute name	Description
GRANTEE	Database user name.
GRANTEETYPE	Database user Type. U=grantee is an individual user.
GRANTOR	Grantor of the authority.
GRANTORTYPE	S=Grantor is system U=Grantor is an individual
BINDADDAUTH	Authority to create packages.

## Schema Attributes

Attribute name	Description
CONNECTAUTH	Authority to connect to the database.  N=Not held Y=Held
CREATETABAUTH	Authority to create tables.  N=Not held Y=Held
DBADMAUTH	DBADM authority.  N=Not held Y=Held
EXTERNALROUTINEAUTH	Authority to create external routines.  N=Not held Y=Held
IMPLSCHEMAAUTH	Authority to implicitly create schemas by creating objects in non-existent schemas.  N=Not held Y=Held
LOADAUTH	Authority to use the DB2 load utility.  N=Not held Y=Held
NOFENCEAUTH	Authority to create non-fenced user-defined functions.  N=Not held Y=Held
QUIESCECONNECTAUTH	Authority to access the database when it is quiesced.  N=Not held Y=Held
SECURITYADMAUTH	Authority to monitor and tune SQL statements.  N=Not held Y=Held
roles	Roles connected to user.

## Roles attributes

---

The following table lists the Roles attributes:

Attribute name	Description
ROLENAMES	Name of the role.
ROLEID	Identifier for the role.

Attribute name	Description
CREATE_TIME	Time when the role was created.
HierarchicalRoles	List of inherited roles.  <b>Note:</b> The hierarchical roles display the child roles of parent roles in the Group object properties of Entitlement Catalog. For existing applications which are getting upgraded, mark entitlement as true to display the child roles in Entitlement grid of Group object properties.
AUDITPOLICYID	Identifier for the audit policy.
AUDITPOLICYNAME	Name of the audit policy.
REMARKS	User-provided comments or null.

## Provisioning Policy attributes

---

The following table lists the provisioning policy attributes for Create Account:

Attribute name	Description
GRANTEE*	Database user name.
CONNECTAUTH	Authority to connect to the database.
BINDADDAUTH	Authority to create packages.
CREATETABAUTH	Authority to create tables.
NOFENCEAUTH	Authority to create non-fenced user-defined functions.
IMPLSCHEMAAUTH	Authority to implicitly create schemas by creating objects in non-existent schemas.
DBADMAUTH	DBADM authority.
LOADAUTH	Authority to use the DB2 load utility.
QUIESCECONNECTAUTH	Authority to access the database when it is quiesced.
EXTERNALROUTINEAUTH	Authority to create external routines.
SECURITYADMAUTH	Authority to monitor and tune SQL statements.

## Additional information

---

This section describes the additional information related to the DB2 Windows Connector.

## Troubleshooting

---

Following targets are supported:

- SCHEMA
- TABLE
- INDEX
- TABLE SPACE
- PACKAGE
- FUNCTION
- PROCEDURE
- METHOD

**Note:** **SCHEMA** is appended before Schema name. Similar appending is done for other objects.

## Create user

---

The DB2 server manages all access on DB2 server for Windows users and the Windows authentication is used to login to DB2 server. Hence a user already existing in windows box/domain is required.

To create user in DB2, perform the following steps:

1. Create identity and navigate to request access.
2. Select the identity.
3. Select the entitlement and checkout it.
4. Enter a User Name.
5. Select Y to any one authorizations listed by default in the create template policy.

**Note:** **User creation fails if any one authorization is not selected as 'Y'.**

## Delete user

---

In order to perform the delete operation, set the value of the **DeleteDatabaseUserBYdefault** parameter to Y as follows:

```
<entry key="DeleteDatabaseUserBYdefault" value="Y" />
```

## Delete Role

---

In order to perform the delete role operation, set the value of the **DeleteRoleBYdefault** parameter to Y as follows:

```
<entry key="DeleteRoleBYdefault" value="Y" />
```

# Troubleshooting

---

## 1 - Revoke permission is not working

If db2jcc4.jar file is used, then the revoke permission will not work.

**Resolution:** If revoke permission is not working then copy or download the correct version of **db2jcc.jar** file.



## **Troubleshooting**

# Chapter 7: SailPoint Delimited File Connector

---

The following topics are discussed in this chapter:

Overview.....	81
Configuration parameters.....	81
Schema attributes .....	83

## Overview

---

The SailPoint Delimited File Connector is a *read only* and rule driven connector. This connector has rules that can be customized to handle the complexity of the data that is being extracted.

This connector can be configured to enable the automatic discovery of schema attributes. See “Schema attributes” on page 83.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Delimited File connector uses the following account and application object type “group” connection attributes. The group attributes are optional and the settings default to settings from the Account if they are not specifically defined.

**Table 1—Delimited File Connector - Account Tab Descriptions**

Parameters	Descriptions
<b>File:</b>	
Parsing Type	Enter which type of parsing technique should be used when parsing the contents of the data file.  <b>Note: The parsing type is only applicable for Account Attributes.</b>
File Path	Enter the path and name of the data file that should be parsed.
File Encoding	Specify the encoding that was used when saving the data file. If this is left blank the application's server default encoding will be used when parsing the file.
Delimiter	Enter the character that should be used as a delimiter. If the delimiter is a unicode character use the \\u format. For example, \\u0009 is used to specify for the tab character.
File has column header on first line	Only available for Delimited Parsing Type. Select this option if the data file has a header defined on the first line of the file.

## Configuration parameters

**Table 1—Delimited File Connector - Account Tab Descriptions**

Parameters	Descriptions
Fail on column length mismatch	Only available for Delimited Parsing Type. Select this option if you want the connector to fail if all of the columns are not part of each line. Sometimes the last token is left out of the data. If this is the case in your file, select this option.
Regular Expression	Only available for Regular Expression Parsing Type. Enter the regular expression using regular expression groups that can be used to break the data into tokens.
Columns	Enter the names of the columns that will be used while parsing the file. If you are using the Regular Expression Parsing Type, field is required. If you are using the Delimited Parsing Type, you only have to configure this field if there is not a header defined or you want to rename of the columns that will be used in the buildMap rule.
<b>Transport:</b>	
<b>Note: Transport attributes only apply to Accounts.</b>	
File Transport	Specify how the file will be transferred. If the file resides locally on the application server, select <b>Local</b> .
Host	Specify the host name where the file is located
User	Specify the username that will be used during the file transfer.
Password	Specify the password for the user that will be used during the file transfer.
<b>Filtering:</b>	
Number of lines to skip	Enter the number of lines to skip from the top of the data file before parsing begins.
Filter Empty	Select this option if you want to filter out any objects that parse but have no attributes.
Comment Character	Enter a comment character used in the data file. Any line starting with this character will be skipped.
Filter String	Enter the string representation of an filter object. Any object matching the filter will be filtered out of the dataset and will not be returned. For example, a filter that will filter out all objects from the Manufacturing department is written as follows: <code>department == "Manufacturing"</code>
<b>Merging:</b>	
Data needs to be merged	Select this option if the data for a single object spans multiple lines.
Index Column	Enter the name of the index column that will be used when finding like objects in the dataset.

**Table 1—Delimited File Connector - Account Tab Descriptions**

Parameters	Descriptions
Data sorted by the indexColumn(s)?	Select this option if the data is sorted by the index columns. If the data is not sorted, an in-memory representation of the data is built and used.
Which Columns should be merged?	Enter the names of the columns from the file from which values should be merged.
<b>Connector Rules:</b>	
<b>Note:</b> Click the “...” icon to launch the Rule Editor to make changes to your rules if needed.	
Build Map Rule	A rule that is called for each row in the data file. This rule is used to convert the string tokens from the data file into a <code>java.util.Map</code> object. If a rule is not specified the connector builds a map with the contents keyed by the column name.
Preliterate Rule	A rule that is called before the iteration process begins and provides a hook for things like checking the file, building an alternate feed, or returning a stream object.
PostIterate Rule	A rule that is called after the iteration process has completed.
Map To ResourceObject Rule	A rule that is called for each unique <code>java.util.Map</code> created from the data file. This rule's job is used to convert a <code>java.util.Map</code> object, built from the data file, into a <code>ResourceObject</code> . If a rule is not specified the connector builds a <code>ResourceObject</code> using the schema.
MergeMaps Rule	A rule that is called during merging for each row that has a matching index column. The rule will receive the existing map along with the newly parsed map that has to be merged. If a rule is not specified the connector builds a combined <code>java.util.Map</code> using the original object and merges the attributes specified in the <code>mergeColumns</code> configuration option.

## Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. In version 7.1, this connector now supports multiple types of objects, account and any number of group application object types. Account objects are used when building identities Link objects. The Additional Schema definitions can be used when building AccountGroup objects which are used to hold entitlements shared across identities.

For delimited file connectors the schema is usually dictated by the data in the file. If this connector is configured to use the automatic discovery function and you've specified column names (`columnNames`, `group.columnNames`, `account.columnNames`), those names are used to populate the schema. If there is a header in the file and the `hasHeader` option is enabled the columns are pulled directly from the file and populate the schema. All automatically generated schema attributes are marked as type String.

## **Schema attributes**

# Chapter 8: SailPoint JDBC Connector

---

The following topics are discussed in this chapter:

Overview.....	85
Supported features .....	85
Supported Managed Systems .....	86
Pre-requisites .....	86
Administrator permissions .....	86
Configuration parameters.....	86
Additional configuration parameters .....	88
Schema Attributes .....	89
Troubleshooting.....	89

## Overview

---

The SailPoint JDBC Connector is used for Read/Write operations on the data of JDBC- enabled database engines. This connector supports flat table data. To handle complex, multi-table data, you need to define a rule and a more complex SQL statement.

This connector can be configured to enable the automatic discovery of schema attributes. See “Schema Attributes” on page 89.

## Supported features

---

SailPoint JDBC Connector supports the following features:

- Account Management
  - Manage JDBC Users as Account
  - Aggregate, Delta Aggregation, Partitioning Aggregation, Refresh Accounts, Discover Schema
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements
- Group Management
  - Manage JDBC Groups
  - Aggregate, Delta Aggregation, Refresh Groups
  - Create, Update, Delete

**Note:** For Group schema, multiple-group objects are supported.

## Configuration parameters

SailPoint supports the following additional JDBC Connector features:

- Ability to provide the SQL statement or stored procedure during application configuration for automatic discovery of group schema attributes from same or different database used for the account schema.
- Ability to define provisioning rule(s) called for each row in the data file to provision account and group attributes.
- Ability to define separate provisioning rule for specific operation called for each row in the data file to provision account and group attributes. Operation that include are Enable, Disable, Unlock, Delete, Create, and Modify.

**Note:** An example of a provisioning rule is located in `examplerules.xml` file.

### References

- Appendix A: Delta Aggregation
- Appendix C: Partitioning Aggregation

## Supported Managed Systems

---

SailPoint JDBC Connector supports the following Managed System:

- Any database having JDBC Driver. For example, MySQL, Oracle, DB2, SQL Server and Sybase

## Pre-requisites

---

An appropriate JDBC driver for the database.

## Administrator permissions

---

The JDBC application will use the database user context to support the aggregation and other provisioning operations. The database user mentioned in the application configuration must have appropriate rights to fetch and set data related to the entities and attributes mentioned in the JDBC SQL query that is, Account, Group, Entitlements/Direct Permission and so on.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The JDBC connector uses the following connection attributes under different tabs (Settings, Merging, Iteration Partitioning and Delta Aggregation):

Attributes	Description
<b>Settings</b>	
<b>JDBC Connection Settings</b>	
Connection User*	The user with which to connect to the host.
Connection Password	The password associated with the specified user.

Attributes	Description
Database URL*	The URL with which to connect to the database.
JDBC Driver*	The JDBC driver class path.
<b>Query Settings</b>	
Test Connection SQL	The SQL Statement for Test Connection.
SQL Statement*	The SQL Statement attribute can be used to customize the selected statement that is generated when iterating over objects. You can specify the exact SQL Statement that is executed if you want to filter out objects or only want to select a few objects from a table. Additionally, if you need to perform joins between more than one table, it's impossible to describe with the schema alone.
getObjectSQL	The object SQL statement.
useExecuteQuery	Use Statement.executeQuery() instead of the default Statement.execute()
Direct Permission Execute Query	Enable this option to execute the query for direct permission.
Get Direct Perm Object SQL	<p>Direct Permission Execute Query is used to pull the direct permission data from permission table. Permission table must contain at least Identity attribute column. The permission data is pulled by referring the identity attribute in the column at the time of aggregation via main SQL query in which the identity attribute is mentioned.</p> <p><b>Note:</b> Query must be written in such a way that <b>ResultSet</b> data must contain first column as <b>Target</b>, second column as <b>Permission</b> and third column as <b>annotation (optional)</b>.</p> <p>For example, SELECT column4 AS TARGET,column5 AS PERMISSION FROM Permission p WHERE CONCAT(TRIM(CONCAT(p.column1,' ')), TRIM(p.column2)) = '\${identity}';</p> <p>Here table name is <b>Permission</b> and <b>\$(identity)</b> is Identity attribute.</p>
usePrepareCall	<p>This parameter must be set from the application debug page when a stored procedure is called from Query Settings in JDBC connector.</p> <p>The entry key must be in the following format:</p> <pre>&lt;entry key="usePrepareCall"&gt;     &lt;value&gt;         &lt;Boolean&gt;true&lt;/Boolean&gt;     &lt;/value&gt; &lt;/entry&gt;</pre>

## Configuration parameters

Attributes	Description
<b>Merging</b>	
Data needs to be merged	Select this option if the data for a single object spans multiple lines.  This option enables the connector to verify the order of the data returned from the database when merging to prevent data loss. When merging, it is very important to have the ORDER BY clause in your SQL statement to prevent out of order errors.
Index Column	Name of the index column that will be used when finding like objects in the dataset.
Which columns should be merged?	Names of the columns from the file from which values must be merged.
<b>Note:</b> User must discover the schema to get the suggested column values in index and merge columns for selection. Discover schema populates the values in multi-suggest attribute drop-down of index and merge columns which have the auto complete facility.	
<b>Iteration Partitioning</b>	
Partitioning Enabled	Select this checkbox to configure and enable partitioning.
Partitioning Statements	Enter the list of sql/stored procedure statements that must be executed when partitioning. The statements must include all of the rows and each line/statement so it can be proceeded in separate threads and/or multiple hosts.  For more information, see Appendix C: Partitioning Aggregation.
<b>Delta Aggregation</b>	
Delta Aggregation Enabled	To use Delta Aggregation feature, the <b>Delta Aggregation Enabled</b> field must be selected.  The <b>Database Table Containing Delta Changes</b> attribute must be provided with the value of table name in which delta changes are captured. This table must have read and write permissions. This field is not mandatory.  The <b>Delta Aggregation SQL</b> can be used when the alias is used in main SQL.  For more information, see Appendix A: Delta Aggregation.

## Additional configuration parameters

---

The following parameters are not displayed on UI but are used to tuning jdbc pooling:

Parameters	Description
pool.maxWait	Maximum time to wait for connection to become available in millisecond.  Default: <entry key="pool.maxWait" value="60000" />
pool.evictRuns	Wait time between closing idle connections in milliseconds.  Default: <entry key="pool.evictRuns" value="300000" />

Parameters	Description
pool.maxActive	Maximum number of connections. Default: <entry key="pool.maxActive" value="10" />
pool.setMaxIdle	Maximum number of idle connections Default: <entry key="pool.setMaxIdle" value="10" />
pool.minEvictIdle	Minimum idle time to close connection in milliseconds. Default: <entry key="pool.minEvictIdle" value="10" />

## Schema Attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface and supports multiple types of objects, account and any number of group application object types. Account objects are used when building identities Link objects. Additional schema definitions can be used when building AccountGroup objects which are used to hold entitlements shared across identities.

The JDBC connector's most important attribute is the SQL Statement. In many cases this is a stored procedure. (call mystoredProcedure). In other cases it is select from a table with any number of joins included. If this connector is configured to use the automatic discovery function, it connects to the database and executes the statement provided and then uses the meta-data returned from the result to build the column names.

## Troubleshooting

---

### 1 - Connection getting locked up

By default the JDBC Connector works in pooling connection mode. The connection gets locked up for already configured application account when password of the account changes dynamically from managed system without displaying any warning message.

**Resolution:** Add the following xml in the application.xml file:

```
<entry key="pool.disablePooling">
  <value>
    <Boolean>true</Boolean>
  </value>
</entry>
```

### 2 - JDBC Connector fails to discover schema with correct alias names in case of MYSQL Database

JDBC Connector fails to discover schema with correct alias names in case of MYSQL Database.

**Resolution:** Add **useOldAliasMetadataBehavior** parameter at the end of the database URL (after the SID (Database Name)) and set it to true.

For example, if the database url in the UI is URL = "jdbc:mysql://localhost:3306/mydb" then, prefix a ? to the **useOldAliasMetadataBehavior** parameter and add this parameter at the end after setting it to true.

## **Troubleshooting**

The URL would be as follows:

```
URL = "jdbc:mysql://localhost:3306/mydb?useOldAliasMetadataBehavior=true"
```

# Chapter 9: SailPoint Jive Connector

---

The following topics are discussed in this chapter:

Overview .....	91
Supported features .....	91
Pre-requisites .....	92
Administrator permission .....	92
Configuration parameters .....	92
Schema attributes .....	92
Account attributes .....	92
Group attributes .....	94
Provisioning Policy attributes .....	94
Create account attributes .....	94
Create group attributes .....	95
Additional information .....	95
Troubleshooting .....	95

## Overview

---

SailPoint Jive Connector was developed to manage Jive User Accounts and Security Groups (Security Groups may have permissions such as Full Access, Manage Community, Manage System, Moderate Content, Manage Users, Manage Groups).

## Supported features

---

SailPoint Jive Connector supports the following features:

- Account Management
  - Manage Jive Person as Account
  - Aggregate, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable
  - Add/Remove Entitlements
- Account - Group Management
  - Manage Jive SecurityGroups as Account - Groups
  - Aggregate, Refresh Groups
  - Create, Update, Delete

## Pre-requisites

---

1. Jive connector requires the following libraries in class path:
  - httpcore-4.2.1.jar
  - httpclient-4.2.1.jar
  - httpclient-cache-4.2.1.jar
  - commons-logging-1.1.1.jar
  - commons-codec-1.6.jar
  - gson-2.1.jar
2. Jive software should be up and running.
3. Administrator should be configured to have proper access rights for modifying Jive users.

## Administrator permission

---

Administrator should be configured to have proper access rights for modifying Jive users.

# Configuration parameters

---

The following table lists the configuration parameters of Jive Connector:

Parameters	Description
Jive URL	URL for accessing Jive (For example, <a href="http://jive1.jivedev.com">http://jive1.jivedev.com</a> )
User Name	User Name used for logging into Jive with sufficient rights (administrator credentials.)
Password	User's password.
Page Size	The maximum size of each data set when querying over large number of objects. Default and max value is 100.

## Schema attributes

---

This section describes the different schema attributes.

### Account attributes

---

The following table lists the account attributes:

Attributes	Description
Id	Identifier (unique within an object type and Jive instance) of the object.
username	The login username for the person. This field is required for person creation, but cannot be changed on an update.
name	Name components (familyName, givenName and formatted) for the person.

Attributes	Description
location	Geographic location of the person.
type	The object type of the object (person).
displayName	Formatted full name of this person, suitable for use in UI presentation. If the person has privacy settings that do not allow you to see his or her name, this will be the Jive username instead.
published	Date and time when the person was originally created. Visible only to Jive administrators or on your person object.
thumbnailurl	URL of the thumbnail (avatar) image for the person.
followingCount	Number of people the person is following.
followerCount	Number of people following the object.
jive	Jive extensions to OpenSocial person object (it is a complex JSON object).
phoneNumbers	Phone numbers belonging to the person, with standard types: fax, home, mobile, other, pager, work.
updated	Date and time the person was most recently updated.
addresses	Postal addresses belonging to the person, with standard types home, other, pobox, work and value type of address.
work_country	Name of the country where person is working.
work_locality	Name of the city where person is working.
work_street_1	Street 1 name where person is working.
work_street_2	Street 2 name where person is working.
work_postalCode	Zip/Postal code where person is working.
work_region	Name of State or Province where person is working.
home_country	Name of the country where person lives.
home_locality	Name of the city where person lives.
home_postalCode	Zip/Postal code where person lives.
home_region	Name of state or province where person lives.
home_street_1	Name of street 1 where person lives.
home_street_2	Name of street 2 where person lives.
givenName	First name of the person.
familyName	Last name of the person.
Department	The person's department name.
Company	The person's company name.
Title	The person's job title name.
Biography	The person's biography.
Expertise	The person's expertise.
Email	This attribute will be used for updating primary email address of a person.

## Provisioning Policy attributes

Attributes	Description
emails	Email addresses belonging to this person, with standard types home, other, work and value type of string.

## Group attributes

The following table lists the group attributes:

Attributes	Description
id	Identifier (unique within an object type and Jive instance) of the object.
name	Name of the security group.
type	Object type of this object ("securityGroup").
description	Description of this security group.
administratorCount	Number of administrative members in the security group.
memberCount	Number of regular members of the security group.
published	Date and time the security group was initially created.
updated	Date and time the security group was last updated.
federated	Flag indicating that the membership of the group is federated with an external directory service.
Admins	List of administrative members in the security group.
Members	List of regular members of the security group.

## Provisioning Policy attributes

This section lists the different policy attributes of Jive Connector.

**Note:** All the attributes marked with \* sign are the mandatory attributes.

## Create account attributes

The following table lists the provisioning policy attributes for Create Account:

Attributes	Description
username*	The login username for this person. This field is required for person creation, but cannot be changed on an update.
givenName*	First name of the person.
familyName*	Last name of the person.
Email*	Primary email address of a person.
password*	Password of a person.

## Create group attributes

---

The following table lists the provisioning policy attributes for Create Group:

Attributes	Description
name*	Name of the group.
description	Text describing the user.

## Additional information

---

This section describes the additional information related to the Jive Connector.

**Note: To enable logging, specify the logging**

log4j.logger.openconnector.connector.JiveConnector **in the log4j.properties file. For example, logging**  
log4j.logger.openconnector.connector.JiveConnector=debug

## Configuration settings

---

The **TimedCache** attribute is used to set the time period to hold the Account-Group memberships information in a cache. This is a configurable field and it has default value of 60 minutes.

To change the TimedCache timeout the user must add the following entry into the application by navigating to the debug page:

```
<entry key="timedCacheTimeout" value="90" />
```

In the above entry **timedCacheTimeout** is **key** and value is the **value** of that key in minutes. For example, 90 minutes.

## Troubleshooting

---

### 1 - Inconsistent UI and API results.

The Jive software has a system property called `jive.pageCached.enabled` which is used to enable/disable the page caching on Jive. By default the value of this system property is **True**. This issue appears when user is performing add/remove entitlements operation.

**Resolution:** Set the `jive.pageCached.enabled` system property value to **False**.

## **Troubleshooting**

# Chapter 10: SailPoint LDAP Connector

---

The following topics are discussed in this chapter:

Overview .....	97
Supported features .....	97
Supported Managed Systems .....	98
Pre-requisites .....	100
Administrator permissions .....	100
Configuration parameters .....	100
Additional configuration parameter .....	101
Configuring Account Search Scope .....	101
Configuring Group Search Scope .....	103
Schema attributes .....	103
Account attributes .....	103
Group attributes .....	107
posixgroup and nisNetgroup Attributes .....	108
Group Membership attribute .....	109
Group Entitlement attribute .....	109
Provisioning Policy attributes .....	110
Additional information .....	111
Adding additional group types .....	111
Managing Revoke-Restore for SunOne .....	113
Using Novell eDirectory as a Pass-through Authentication Source .....	113
Troubleshooting .....	113

## Overview

---

This connector was developed using the LDAP RFC. The LDAP Connector must plug into almost any LDAP server with no customization. The LDAP Connector now supports provisioning of users and entitlements along with the retrieval of LDAP account and group object classes.

## Supported features

---

SailPoint LDAP Connector supports the following features:

- Account Management
  - Manage LDAP Users as Account
  - Delta Aggregation (SunOne -Direct, IBM Tivoli DS- Direct, ADAM- Direct)
  - Aggregate, Refresh Accounts, Partitioning Aggregation, Pass Through Authentication
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements

## Overview

**Note:** The Enable and Disable feature is supported only for ADAM-Direct, Novell eDirectory, Oracle Internet Directory and SunOne - Direct managed systems.

**Note:** The Unlock feature is supported only for Novell eDirectory, Oracle Internet Directory and SunOne - Direct managed systems.

- Account - Group Management
  - Manage LDAP Groups as Account - Groups
  - Delta Aggregation (SunOne -Direct, IBM Tivoli DS - Direct, ADAM- Direct)
  - Aggregate, Refresh Group
  - Create, Update, Delete

### References

- “Appendix A: Delta Aggregation”
- “Appendix C: Partitioning Aggregation”

## Supported Managed Systems

---

SailPoint LDAP Connector supports the following Managed Systems:

- Microsoft ADAM 2012 R2, 2012
- OpenLDAP version 2.4, 2.3
- ODSEE 11g
- IBM Tivoli Directory Server version 6.4, 6.3
- Novell eDirectory version 9.0, 8.8
- Oracle Internet Directory version 11gR2

For each of the supported managed systems we have application types already existing in the connector registry as mentioned in the following section.

## LDAP Connector Application Types

In order to speed up the process of building an application quickly, this connector comes with the following default LDAP application types to manage the corresponding directory server:

Application Types	Description
ADAM - Direct	Manages ADAM directory server
SunOne - Direct	Manages SunOne directory server
IBM Tivoli DS - Direct	Manages Tivoli directory server
Novell eDirectory - Direct	Manages Novell eDirectory server
Oracle Internet Directory - Direct	Manages Oracle Internet directory server
OpenLDAP - Direct	Manages OpenLDAP directory server

## Object Types Managed

Each of the application types has been pre-configured to manage commonly used object classes and their attributes. For instance, the application schema of ADAM directory server has been configured for user and group

object classes. Table 1 and Table 2 displays the mapping for various application types. Custom object classes may be mapped by modifying the corresponding application schema.

The following table displays the object classes mapped for each of the LDAP application types:

**Table 1—Object classes mapped for each of the LDAP application types**

Application types	Objectclass mapped for accounts	Objectclass mapped for groups	Group membership attributes	Group entitlement attribute
ADAM - Direct	user	group	member	groups
SunOne - Direct	inetOrgPerson	groupofUniqueNames	uniqueMember	groups
IBM Tivoli DS - Direct	inetOrgPerson	groupofUniqueNames	uniqueMember	groups
Novell eDirectory - Direct	inetOrgPerson	groupofUniqueNames	uniqueMember	groups
Oracle Internet Directory - Direct	inetOrgPerson	groupofUniqueNames	uniqueMember	groups
OpenLDAP - Direct	inetOrgPerson	groupofUniqueNames	uniqueMember	groups

The following table displays the object classes mapped for nisNetgroup and posixgroup for SunOne, Tivoli DS and OpenLDAP application types:

**Table 2—Object classes mapped for nisNetgroup and posixgroup for LDAP application types**

Application types	Objectclass mapped for groups	Group membership attributes	Group entitlement attribute
Sun One - Direct <i>OR</i> IBM Tivoli DS - Direct <i>OR</i> OpenLDAP - Direct	nisNetgroup	nisNetgroupTriple	nisNetgroups
	posixgroup	memberUid	posixgroups

#### *Create TLS communication between IdentityIQ and LDAP Server*

If you want secure TLS connection for LDAP, TLS communication needs to be enabled between IdentityIQ and LDAP Server. For a Java client to connect using TLS and self-signed certificates, you have to install the certificate into the JVM keystore.

To create TLS communication between IdentityIQ and LDAP Server, perform the following:

1. Export server certificate and copy the exported .cer file to the Java client computer (IdentityIQ computer).
2. At the client computer execute the following command from the bin directory of JDK:  

```
keytool -importcerts -trustcacert -alias aliasName -file <absolute path of certificate> -keystore <JAVA_HOME>/jre/lib/security/cacerts
```

In the preceding command line, *aliasName* is the name of the alias.
3. Login to IdentityIQ.
4. Create the application for LDAP application type and provide all the required values after selecting the **Use TLS** checkbox.
5. Click on **Test Connection** and save the application.

## Pre-requisites

---

SailPoint LDAP Connector requires that the directory server has the administrator credentials.

Ensure that the following pre-requisites are satisfied for the respective directory servers:

- (*For SunOne Directory Server*) Global Password Policy must have the **Require Password Change at First Login and After Reset** option selected.

**Note:** **For self change password, add the CURRENT\_PASSWORD attribute to the featureString in the application schema.**

- (*For IBM Tivoli*)

- Password Policy must be enabled and assigned to the user.
- The **User can change Password** option must be selected.

- The **User must change Password after reset** option must be selected.

**Note:** - Add the following attribute to the system configuration schema and set it to true:

```
<entry key="requireOldPasswordAtChange" value="true" />
```

- **For self change password, add the CURRENT\_PASSWORD attribute to the featureString in the application schema.**

- (*For Novell eDirectory*)

- Universal Password must be configured.
- Enable Universal Password in Password Policy.
- The **Allow users to initiate password change** option must be selected.
- The **Do not expire the user's password when the administrator sets the password** option must not be selected.
- Grace logins can be applied according to password policy added to the application schema.

- (*For Oracle Internet Directory*) Assign password policy to user.

**Note:** - Add the following attribute to the system configuration schema:

```
<entry key="requireOldPasswordAtChange" value="true" />
```

- **For self change password, add the CURRENT\_PASSWORD attribute to the featureString in the application schema.**

## Administrator permissions

---

SailPoint LDAP Connector must have the read /write privilege's over the directory information tree in order to manage the LDAP data.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The LDAP connector uses the following connection attributes:

Parameters	Description
useSSL	Specifies if the connection is over TLS.
authorizationType	The authorization type to use when connecting to the server.
user*	The user to connect as a DN string such as Administrator.
password	Password for the administrator account.
port*	Port the server is listening through.
host*	Host of the LDAP server.
pageSize	Number of objects to get, per page, when iterating over large numbers of objects. Default is 500.
authenticationSearchAttributes	Attributes used when authenticating the application using pass - through authentication.

## Additional configuration parameter

---

Attributes	Description
expiredPasswordErrorMessages	List of possible error strings (full or partial) returned by the LDAP server that indicates the password expiration for the user.

## Configuring Account Search Scope

---

The searchDNs define list of distinguished names of the containers along with other relevant attributes which defines scope for this application. Each of these searchDNs is considered as a partition for parallel aggregation. Accounts and Groups can have different set of searchDNs to define different scope for each of them. In case the scope is not defined for Groups, it follows Accounts scope. Defining one search DN to the minimum is required to successfully configure application.

Attributes to be defined for searchDNs are as follows:

Attributes	Description
searchDN*	Distinguished Name of the container.
iterateSearchFilter	LDAP filter that defines scope for accounts/groups from this container.

## Configuration parameters

Attributes	Description
groupMembershipSearchScope	<p>(Applicable only for account search scope) List of map which represents scope and filter for group membership of each different group-objectType (for example, posixgroup, nisNetgroup).</p> <p>If group membership search scope is not defined then search DN will be considered as scope for fetching group membership for that specific group-object type</p> <ul style="list-style-type: none"> <li>• <b>objectType*</b>: Group ObjectType name <b>Note:</b> This object type Name must match with objectType name of the respective group schema.</li> <li>• <b>groupMembershipSearchDN</b>: (Optional) Multivalued attribute to define scope for group memberships.</li> <li>• <b>groupMemberFilterString</b>: (Optional) LDAP filter for groups membership.</li> </ul>

**Note:** The LDAP connector uses the **groupMembershipSearchDN** attribute as the starting point in the directory to start searching for ALL group memberships. LDAP does not store a user's group references on the user so the LDAP connector must always do a separate query to return a list of all of the user's groups.

### Following is an example for the attributes of Account Search Scope with Group Membership Search Scope

```

<entry key="searchDNs">
    <value>
        <List>
            <Map>
                <entry key="groupMembershipSearchScope">
                    <value>
                        <List>
                            <Map>
                                <entry key="groupMemberFilterString"
value="(&objectClass=posixgroup)(cn=*)"/>
                                    <entry key="groupMembershipSearchDN">
                                        <value>
                                            <List>
                                                <String>ou=Sales,dc=org,dc=com</String>
                                                <String>ou=HR,dc=org,dc=com</String>
                                            </List>
                                        </value>
                                    </entry>
                                    <entry key="objectType" value="posixgroup"/>
                                </Map>
                            </List>
                        </value>
                    </entry>
                </Map>
            </List>
        </value>
    </entry>
    <entry key="iterateSearchFilter"
value="(&objectClass=inetOrgPerson)(cn=a*)"/>
        <entry key="searchDN" value="ou=Canada,dc=org,dc=com"/>
        <entry key="searchScope" value="SUBTREE"/>
    </Map>
    <Map>
        <entry key="iterateSearchFilter"
value="(&objectClass=inetOrgPerson)(cn=b*)"/>
            <entry key="searchDN" value="ou=America,dc=org,dc=com"/>
            <entry key="searchScope" value="SUBTREE"/>
    </Map>

```

```

        </Map>
    </List>
</value>
</entry>

```

## Configuring Group Search Scope

---

The `<objectType>.searchDNs` define list of distinguished names of the containers along with other relevant attributes which define scope and filter for group aggregation.

Attributes	Description
Search DN*	Defines scope of group aggregation for mentioned objectType.
Iterate Search Filter	LDAP filter for group aggregation.
Filter String	Filter which can filter object after they have been returned from the underlying directory.

### Following is an example for attributes of Group Search Scope

```

<entry key="posixgroup.searchDNs" >
    <value>
        <List>
            <Map>
                <entry key="iterateSearchFilter" value="(&amp;(objectclass=posixgroup))" />
                <entry key="searchDN" value="ou=HR,dc=org,dc=com" />
                <entry key="searchScope" value="SUBTREE" />
            </Map>
            <Map>
                <entry key="iterateSearchFilter" value="(&amp;(objectclass=posixgroup))" />
                <entry key="searchDN" value="ou=Sales,dc=org,dc=com" />
                <entry key="searchScope" value="SUBTREE" />
            </Map>
        </List>
    </value>
</entry>

```

**Note:** The name of the `objectType` must be same as the name of the `objectType` of respective group schema for which the scope is defined.

## Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports two types of objects, account and group.

### Account attributes

---

Account objects are used when building identities Link objects.

## Schema attributes

**Table 3—LDAP Connector - Account Attributes**

Name	Description
businessCategory	Types of business performed by an organization. Each type is one value of this multi-valued attribute. Examples: “engineering”, “finance”, and “sales”.
carLicense	License plate or vehicle registration number associated with the user.
cn	Names of an object. Each name is one value of this multi-valued attribute. If the object corresponds to a person, it is typically the person’s full name. Examples: “Martin K Smith”, “Marty Smith” and “printer12”.
dn	Distinguished name by which the user is known.
departmentNumber	Numerical designation for a department within your enterprise.
description	Human-readable descriptive phrases about the object. Each description is one value of this multi-valued attribute. Examples: “Updates are done every Saturday, at 1am.”, and “distribution list for sales”.
destinationIndicator	Country and city strings associated with the object (the addressee) required to provide the Public Telegram Service. The strings are composed in accordance with CCITT Recommendations F.1 [F.1] and F.31 [F.31]. Each string is one value of this multi-valued attribute. Examples: “AASD” as a destination indicator for Sydney, Australia. “GBLD” as a destination indicator for London, United Kingdom.  <b>Note: The directory will not ensure that values of this attribute conform to the F.1 and F.31 CCITT Recommendations. It is the application’s responsibility to ensure destination indicators that it stores in this attribute are appropriately constructed.</b>
displayName	Preferred name to be used for this person throughout the application.
employeeNumber	Numerical identification key for this person within you enterprise.
employeeType	Descriptive type for this user, for example, contractor, full time, or part time.
facsimileTelephoneNumber	Telephone numbers and any required parameters for facsimile terminals. Each telephone number is one value of this multi-valued attribute.
givenName	Name strings that are the part of a person’s name that is not their surname. Each string is one value of this multi-valued attribute. Examples: “John”, “Sue”, and “David”.
groups	List of groups of which this person is a member. Example: “Sales” or “Engineering”
posixgroups	(Applicable only for IBM Tivoli, OpenLDAP Direct, and SunOne Direct)  List of posixgroups of which this person is a member. Example: “Sales” or “Engineering”  For more information, see “ posixgroup and nisNetgroup Attributes” on page 108.

**Table 3—LDAP Connector - Account Attributes (Continued)**

Name	Description
nisNetgroups	(Applicable only for IBM Tivoli, OpenLDAP Direct, and SunOne Direct)  List of nisNetgroup of which this person is a member. Example: "Sales" or "Engineering"  For more information, see " posixgroup and nisNetgroup Attributes" on page 108.
homePhone	Employees home phone number.
homePostalAddress	Employees mailing address.
initials	Strings of initials of some or all of an individual's names, except the surname(s). Each string is one value of this multi-valued attribute. Examples: "J. A." and "J".
internationalISDNNumber	Integrated Services Digital Network (ISDN) addresses, as defined in the International Telecommunication Union (ITU) Recommendation E.164 [E.164]. Each address is one value of this multi-valued attribute. Example: "0198 444 444".
l	Names of a locality or place, such as a city, county, or other geographic region. Each name is one value of this multi-valued attribute. Examples: "Austin", "Chicago", and "Brisbane".
mail	RFC822 mailbox for the user.
manager	Distinguished name of the manager to whom this person reports.
mobile	Mobile telephone number of this person.
o	Names of an organization. Each name is one value of this multi-valued attribute. Examples: "xyz", "xyz Technologies, Inc.", and "xyz, Incorporated.".
ou	Names of an organizational unit. Each name is one value of this multi-valued attribute. Examples: "Sales", "Human Resources", and "Information Technologies".
pager	Telephone number of this persons pager.
physicalDeliveryOfficeName	Names that a Postal Service uses to identify a specific post office. Examples: "Austin, Downtown Austin" and "Chicago, Finance Station E".
postOfficeBox	Postal box identifiers use by a postal service to locate a box on the premises of the Postal Service rather than a physical street address. Each postal box identifier is a single value of this multi-valued attribute. Example: "Box 27".
postalAddress	Addresses used by a Postal Service to perform services for the object. Each address is one value of this multi-valued attribute. Example: "1111 Elm St.\$Austin\$Texas\$USA".
postalCode	Codes used by a Postal Service to identify postal service zones. Each code is one value of this multi-valued attribute. Example: "78664", to identify Pflugerville, TX, in the USA.

## Schema attributes

**Table 3—LDAP Connector - Account Attributes (Continued)**

Name	Description
preferredDeliveryMethod	Indication of the preferred method of getting a message to the object. Example: If the mhs-delivery Delivery Method is preferred over telephone-delivery, which is preferred over all other methods, the value would be: "mhs \$ telephone".
preferredLanguage	Preferred written or spoken language of this person.
pwdReset*	(Applicable only for IBM Tivoli, OpenLDAP, and Oracle Internet Directory) Specifies whether the password has been reset by administrator.  <b>Note: Must be added manually to support password reset.</b>
pwdLastSet*	(Applicable only for ADAM) User password last set time.  <b>Note: Must be added manually to support password reset.</b>
passwordExpirationTime*	(Applicable only for Novell eDirectory and SunOne managed system) Users password expiration time.  <b>Note: Must be added manually to support password reset for Novell eDirectory.</b>
registeredAddress	Postal addresses to be used for deliveries that must be signed for or require a physical recipient. Each address is one value of this multi-valued attribute. Example: "Receptionist\$xyz Technologies\$6034 Courtyard Dr. \$Austin, TX\$USA"
roomNumber	Room or office number or this persons normal work location.
secretary	Distinguished name of this persons secretary.
seeAlso	Distinguished names of objects that are related to the subject object. Each related object name is one value of this multi-valued attribute. <b>Example:</b> The person object "cn=Elvis Presley,ou=employee,o=xyz\, Inc." is related to the role objects "cn=Bowling Team Captain,ou=sponsored activities,o=xyz\, Inc." and "cn=Dart Team,ou=sponsored activities,o=xyz\, Inc.". Since the role objects are related to the person object, the 'seeAlso' attribute will contain the distinguished name of each role object as separate values.
sn	Name strings for surnames, or family names. Each string is one value of this multi-valued attribute. <b>Example:</b> "Smith"
st	Full names of states or provinces. Each name is one value of this multi-valued attribute. <b>Example:</b> "Texas"
street	Site information from a postal address (i.e., the street name, place, avenue, and the house number). Each street is one value of this multi-valued attribute. <b>Example:</b> "15 Main St."
telephoneNumber	Telephone numbers that comply with the ITU Recommendation E.123 [E.123]. Each number is one value of this multi-valued attribute.

**Table 3—LDAP Connector - Account Attributes (Continued)**

Name	Description
teletexTerminalIdentifier	The withdrawal of recommendation F.200 has resulted in the withdrawal of this attribute.
telexNumber	Sets of strings that are a telex number, country code, and answerback code of a telex terminal. Each set is one value of this multi-valued attribute
title	Persons job title. Each title is one value of this multi-valued attribute. <b>Examples:</b> “Vice President”, “Software Engineer”, and “CEO”
uid	Computer system login names associated with the object. Each name is one value of this multi-valued attribute. <b>Examples:</b> “s9709015”, “admin”, and “Administrator”
objectClass	The values of the objectClass attribute describe the kind of object which an entry represents. The objectClass attribute is present in every entry, with at least two values. One of the values is either “top” or “alias”.

## Group attributes

---

The group schema is used when building AccountGroup objects which are used to hold entitlements shared across identities.

**Table 4—LDAP Connector - Group Attributes**

Name	Description
cn	Names of object. Each name is one value of this multi-valued attribute. If the object corresponds to a person, it is typically the person's full name. <b>Examples:</b> “Martin K Smith”, “Marty Smith” and “printer12”
uniqueMember	Groups to which this person is a unique member.
dn	Directory path to the object.
o	Names of organization. Each name is one value of this multi-valued attribute. <b>Examples:</b> “xyz”, “xyz Technologies, Inc.”, and “xyz, Incorporated.”
ou	Names of organizational unit. Each name is one value of this multi-valued attribute. <b>Examples:</b> “Sales”, “Human Resources”, and “Information Technologies”
owner	Distinguished names of objects that have ownership responsibility for the object that is owned. Each owner's name is one value of this multi-valued attribute. <b>Example:</b> The mailing list object, whose DN is “cn=All Employees, ou=Mailing List,o=xyz, Inc.”, is owned by the Human Resources Director. Therefore, the value of the 'owner' attribute within the mailing list object, would be the DN of the director (role): “cn=Human Resources Director,ou=employee,o=xyz, Inc.”.
description	Human-readable descriptive phrases about the object. Each description is one value of this multi-valued attribute. <b>Examples:</b> “Updates are done every Saturday, at 1am.”, and “distribution list for sales”

## posixgroup and nisNetgroup Attributes

---

Names	Description
<b>nisNetgroup Attributes</b>	
cn	Names of objects. Each name is one value of this multi-valued attribute. If the object corresponds to a person, it is typically the person's full name.  <b>Examples:</b> Martin K Smith, Marty Smith and printer12
nisNetgroupTriple	Unique member of a nisNetgroup.
dn*	Directory path to the object.
description	Human-readable descriptive phrases about the object. Each description is one value of this multi-valued attribute.
<b>posixgroup Attributes</b>	
cn	Names of objects. Each name is one value of this multi-valued attribute. If the object corresponds to a person, it is typically the person's full name.  <b>Examples:</b> Martin K Smith, Marty Smith and printer12
memberUid	Unique member of a posixgroup.
gidNumber*	Integer value that uniquely identifies a group in an administrative domain.
dn*	Directory path to the object.
description	Human-readable descriptive phrases about the object. Each description is one value of this multi-valued attribute.

### Additional group schema configuration attributes

Entry Key	Description
*groupMemberAttribute	Name of Group Membership attribute of account.  <b>Example:</b>  <b>posixgroup:</b> memberUid  <b>nisNetgroup:</b> nisNetgroupTriple  <b>groupOfUniqueNames:</b> uniqueMember

Entry Key	Description
memberAttribute	<p>Attribute name or distinguished name which is stored as value of the groupMemberAttribute. Values: <b>cn</b>, <b>uid</b> or <b>dn</b></p> <p><b>Example:</b> groupMemberAttribute for posixgroup is <b>memberUid</b> against which the values of account can be <b>cn</b> or <b>uid</b>, so for posixgroup groupMemberAttribute name is <b>memberUid</b> and <b>memberAttribute</b> names are <b>cn</b> or <b>uid</b> or both.</p> <p>Similarly for groupOfUniqueNames, groupMemberAttribute is <b>uniqueMember</b> and <b>memberAttribute</b> name is <b>dn</b>.</p> <p><b>Note:</b> One or more than one memberAttribute can be configured as given in the above example of sample schema for sudoRole under “ Adding additional group types” on page 111.</p>
memberPrefix	<p>Value for this field is required if we have any prefix string before the memberAttribute value.</p> <p>For example: (,user 1,)</p> <p>Here prefix is “(,”</p> <p><b>Note:</b> The memberPrefix attribute is not required if the value of the member attribute is ‘dn’.</p>
memberSuffix	<p>Value for this field is required if we have any suffix string after the memberAttribute value.</p> <p>For example: (,user 1,)</p> <p>Here suffix is “,)”</p> <p><b>Note:</b> The memberSuffix attribute is not required if the value of the member attribute is ‘dn’.</p>

**Note:** Value mentioned against “displayAttribute” and “identityAttribute” must be present in Group Schema attributes.

## Group Membership attribute

---

The group membership attribute has been implicitly mapped for the various application types. This attribute and its value can be seen in the application page. Refer to Table 1 and Table 2 for the group membership attribute mapped for each application type. This attribute can be changed from the default to a membership attribute specific to the custom object class mapped. For instance, if the groupOfUniqueNames is the default object class that has been mapped in the application schema for managing groups, then the default group membership attribute can be changed from uniquemember to member if groupOfNames is mapped in the application schema to manage groups.

## Group Entitlement attribute

---

By default, all application types have the groups attribute mapped as the default entitlement attribute. This attribute is simply a placeholder to contain user/group memberships. While creating a new group aggregation task for the application, you would need to specify the value groups in the group account attribute text box in the group aggregation task.

# Provisioning Policy attributes

---

The following table lists the provisioning policy attributes:

Attribute	Description
<b>Account creation</b>	
dn	Distinguished name of the user to be created.
password	Password of the user to be created.
CN	Full name of the user to be created. For example, <b>Martin K Smith, Marty Smith</b> and <b>printer12</b> .
givenName	First Name of the user to be created.
SN	Last name of the user to be created.
During creation of account ensure value of "CN" should be same as mentioned in "dn".	
<b>Example :</b> If value of "dn" is CN=Jeff Smith,OU=Sales,DC=Fabrikam,DC=COM then value of "CN" should be "Jeff Smith"	
<b>Create group (groupOfUniqueNames)</b>	
dn	Distinguished name of the user to be created.
uniqueMember	The distinguished name for the member of a group.
description	Description of the group to be created.
<b>Update group (groupOfUniqueNames)</b>	
description	Description of the group to be created.
<b>Create posixgroup</b>	
dn	Distinguished name of the user to be created.
gidNumber	Contains an integer value that uniquely identifies a group in an administrative domain.
description	Description of the posixgroup to be created.
<b>Update posixgroup</b>	
description	Description of the posixgroup to be created.
<b>Create nisNetgroup</b>	
dn	Distinguished name of the user to be created.
description	Description of the nisNetgroup to be created.
<b>Update nisNetgroup</b>	
description	Description of the nisNetgroup to be created.

## Configuring group provisioning policy for new group

---

1. Ensure that we have added new group schema in application configuration.
2. Identify required attributes for provisioning operation of newly added group.
3. If required add appropriate provisioning policy for create and update operation of that group.
4. Perform Provisioning operations.

## Additional information

---

This section describes the additional information related to the LDAP Connector.

### Adding additional group types

---

Following application types of LDAP support additional group types:

- SunOne - Direct
- IBM Tivoli DS - Direct
- OpenLDAP - Direct

Perform the following steps to configure additional group type:

1. After upgrading IdentityIQ to version 7.1, from UI navigate to account schema of the application and save the application.
2. Add required group schema.

Following is a sample schema for sudoRole:

```
<Schema aggregationType="group" created="" displayAttribute="cn"
featuresString="PROVISIONING" id="" identityAttribute="dn" instanceAttribute=""
modified="" nativeObjectType="sudoRole" objectType="sudoRole">
    <AttributeDefinition name="cn" type="string">
        <Description>common name(s) for which the entity is known by</Description>
    </AttributeDefinition>
    <AttributeDefinition name="dn" type="string">
        <Description>Directory Path</Description>
    </AttributeDefinition>
    <AttributeDefinition name="ou" type="string">
        <Description>organizational unit this object belongs to</Description>
    </AttributeDefinition>
    <AttributeDefinition name="description" type="string">
        <Description>descriptive information</Description>
    </AttributeDefinition>
    <AttributeDefinition multi="true" name="sudoUser" type="string">
        <Description>unique member of a sudoRole </Description>
    </AttributeDefinition>
    <Attributes>
        <Map>
            <entry key="groupMemberAttribute" value="sudoUser" />
            <entry key="memberAttribute" />
        </Map>
    </Attributes>

```

## Additional information

```
<value>
  <List>
    <String>cn</String>
    <String>uid</String>
  </List>
</value>
</entry>
</Map>
</Attributes>
</Schema>
```

### 3. Add entitlement attribute to account schema

- For newly added group schema add entitlement attribute in account schema from UI.
  - a. Ensure that the following steps are performed after creating entitlement attribute:
    - Change the type of entitlement attribute from String to newly added group schema object Type
    - Mark Entitlement attribute as Managed, Entitlement and Multivalued
  - (*Applicable only if nisnet and posix groups are configured as entitlements*) To manage **nisnet** and **posix** groups on upgraded application as group, perform the following:
    - Add nisnet and posix group schema from Debug and Save application.

**Note:** **Name of schema objectType must match with** `objectType="nisNetgroup"` **or** `objectType="posixgroup"` **respectively.**

- From UI navigate to account schema of the application and change the type of entitlement attribute from **String** to **posixgroup** or **nisNetgroup** accordingly and save the application.

Note the following:

- Multigroup application supports static groups as follows:
  - `groupofUniqueNames`
  - `groupOfNames`
  - `nisNetgroup`
  - `posixgroup`
  - `sudoRole`
- For nisNetgroup memberships in **nisNetgroupTriple** attribute all types of braces are supported. By default curly braces '{' are supported.  
**For example,** `{host1,user1,}`    `(host1,user1,)`,    `[host1,user1,]`, **is a supported format.** Whereas, `<host1,user1,>` are unsupported formats. For instance, if user1 has a nisnetgroup memberships which is in the format any other than the curly braces, round braces and square braces then this entitlement would not be retrieved.

**Note:** **For instance, if user wants to use "<" angular braces the format should be like - "&lt;"**

- The nisNetgroup entitlement is added only with the user portion of the **nisNetgroupTriple** attribute value. The domain and host counterpart are not incorporated.  
For example, on SunOne, Tivoli and Open LDAP `{user1,}` is the value of the nisNetgroupTriple attribute after adding an entitlement for user1 on a nisNetgroup.

## Managing Revoke-Restore for SunOne

---

SunOne directory server requires the complete DN of the nsmanagedDisabledRole object to manage revoke-restore functionality.

By default, in the application schema for SunOne - Direct, nsmanagedDisabledRole attribute has been mapped as follows to manage restore and revoke respectively:

```
<entry key="restoreVal" value="cn=nsManagedDisabledRole,dc=Naming Context"/>
```

and

```
<entry key="revokeVal" value="cn=nsManagedDisabledRole,dc=Naming Context"/>
```

You need to modify this attribute to contain the complete DN of the nsManagedDisabledRole object. For instance, if the DN of the nsManagedDisabledRole is cn=nsManagedDisabledRole, dc=sailpoint, dc=com, the restore entry would be modified to as follows:

```
<entry key="restoreVal" value="cn=nsManagedDisabledRole,dc=example,dc=com"/>
```

Similarly, you would need to modify the revoke xml entry as follows:

```
<entry key="revokeVal" value="cn=nsManagedDisabledRole,dc=example,dc=com"/>
```

## Using Novell eDirectory as a Pass-through Authentication Source

---

If using the Novell eDirectory - Direct application type as a pass-through authentication source, remove the dn entry from the Authentication Search Attributes. Using the DN is currently not supported.

## Troubleshooting

---

### 1 - During account aggregation IdentityIQ displays two entries

During account aggregation, if distinguished names are having space difference, IdentityIQ displays two entries for account.

**Resolution:** Ensure that there are no space differences in distinguished names.

### 2 - Group aggregation displays multiple entries for a group with multiple object class

If any group is having multiple object class associated with it then after group aggregation multiple entries are displayed for that respective group.

### 3 - During account aggregation, nisNetgroup membership is not displayed

Latest version of OpenLDAP server on windows (2.4.26) is unable to perform an equality search on the **nisNetgroupTriple** attribute of **nisNetgroup** objectclass. As a result, the nisNetgroup membership is not displayed during account aggregation.

**Resolution:** Open **nis.schema** file of the OpenLDAP server installation and verify if the **nisNetgroupTriple** schema attribute definition is same as the following attribute type:

```
(1.3.6.1.1.1.14 NAME 'nisNetgroupTriple'  
DESC 'Netgroup triple'  
EQUALITY caseExactIA5Match
```

## Troubleshooting

```
SUBSTR caseExactIA5SubstringsMatch  
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26)
```

If the attribute type is not same as the above, then take a back up of the existing nis.schema file and replace the existing nisNetgroupTriple definition with the above. Save the file and restart the OpenLDAP server. After performing aggregation, nisNetgroup membership must get fetched.

### 4 - After upgrading to IdentityIQ version 7.1, the nisNetgroup and posixgroup features are not working in the application configuration page

The version 6.0 patch 5 incorporated the support of **nisNetgroup** and **posixgroup** feature. According to which, **nisNetgroup** and **posixgroup** were mapped as native object types in the application schema. After upgrading to version 7.1, the functionality would not work for the existing applications.

**Resolution:** To ensure that the functionality works in the application configuration page, select the Enable Posix Groups check-box and enter appropriate value for the Map To Member Attribute field.

For instance, if the configuration attribute appeared as

- <entry key="groupMembershipAttributeType" value="uid"/>  
then it must be changed to  
<entry key="posixgroup\_Member\_Attribute" value="uid"/>
- <entry key="groupMembershipAttributeType" value="uid"/>  
then it must be changed to  
<entry key="nisNetGroupTriple\_Member\_Attribute" value="uid"/>

# Chapter 11: SailPoint LDIF Connector

---

The following topics are discussed in this chapter:

Overview .....	115
Configuration parameters .....	115
Schema Attributes .....	116

## Overview

---

The SailPoint LDIF Connector is a *read only* connector used to extract data from LDIF files. To help when the membership is not part of the account data there is an option that can be configured named **groupMembershipAttribute**. This configuration setting holds the name of the attribute from the group file which contains the list of its members. Add this attribute to account schema and mark it multi-valued.

The **groupMembershipAttribute** along with a group file must be configured for this feature to work. During account iteration the connector will read in the groups file to get the group => use mapping and adorn each account with their assigned groups as they are aggregated.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The LDIF connector uses the following configuration parameters:

Parameters	Description
filetransport	local, ftp, scp
host	The host of the server to which you are connecting.
transportUser	The user to use with ftp and scp. Not valid with local.
transportUserPassword	The password to use with of ftp and scp. Not valid with local.
file	The fully qualified path to the file.
fileEncoding	Specify the file encoding to be used by the connector. Valid values for this attribute can be found at: <b><a href="http://www.iana.org/assignments/character-sets">http://www.iana.org/assignments/character-sets</a></b>  If this field is empty, the default encoding (the value of <code>file.encoding</code> specified by the jvm) is used.

## Schema Attributes

Parameters	Description
mapToResourceObjectRule	Rule that is called to override the transformation of the data from the Map<String, String> form into a ResourceObject.
filterString	Filter lines that match this string.
filterEmptyRecords	If activated, records that have no data are filtered.
preIterativeRule	The pre-iterate rule will check for a specially named Configuration object that will hold the last run statistics that can be compared against the current values.  This rule is called after the file has been transferred, but before iteration over the objects in the file is started.  For validation this rule can use the existing statistics stored by the postIterationRule during the last aggregation. The rule can compare the stored values with the new values to check for problems
postIterativeRule	The post-iterate rule can store away the configuration object and rename/delete the file if desired.  This rule is called after aggregation has completed and ALL objects have been iterated.
groupMembershipAttribute	Holds the name of the attribute from the group file which contains the list of its members.

## Schema Attributes

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports two types of objects, account and group.

### Account attributes

Account objects are used when building identities Link objects.

**Table 1—LDIF Connector - Account Attributes**

Name	Description
businessCategory	The types of business performed by an organization. Each type is one value of this multi-valued attribute. Examples: “engineering”, “finance”, and “sales”.
carLicense	License plate or vehicle registration number associated with the user.
cn	Names of object. Each name is one value of this multi-valued attribute. If the object corresponds to a person, it is typically the person’s full name. Examples: “Martin K Smith”, “Marty Smith” and “printer12”.
dn	Distinguished name by which the user is known.

**Table 1—LDIF Connector - Account Attributes (Continued)**

Name	Description
departmentNumber	Numerical designation for a department within your enterprise.
description	Human-readable descriptive phrases about the object. Each description is one value of this multi-valued attribute. Examples: "Updates are done every Saturday, at 1am.", and "distribution list for sales".
destinationIndicator	Country and city strings associated with the object (the addressee) needed to provide the Public Telegram Service. The strings are composed in accordance with CCITT Recommendations F.1 [F.1] and F.31 [F.31]. Each string is one value of this multi-valued attribute. Examples: "AASD" as a destination indicator for Sydney, Australia. "GBLD" as a destination indicator for London, United Kingdom. <b>Note: The directory will not ensure that values of this attribute conform to the F.1 and F.31 CCITT Recommendations. It is the application's responsibility to ensure destination indicators that it stores in this attribute are appropriately constructed.</b>
displayName	Preferred name to be used for this person throughout the application.
employeeNumber	Numerical identification key for this person within you enterprise.
employeeType	Descriptive type for this user, for example, contractor, full time, or part time.
facsimileTelephoneNumber	Telephone numbers and any required parameters for facsimile terminals. Each telephone number is one value of this multi-valued attribute.
givenName	Name strings that are the part of a person's name that is not their surname. Each string is one value of this multi-valued attribute. Examples: "John", "Sue", and "David".
groups	List of groups of which this person is a member. Example: "Sales" or "Engineering"
homePhone	Employees home phone number.
homePostalAddress	Employees mailing address.
initials	Strings of initials of some or all of an individual's names, except the surname(s). Each string is one value of this multi-valued attribute. Examples: "J. A." and "J".
internationalISDNNumber	Integrated Services Digital Network (ISDN) addresses, as defined in the International Telecommunication Union (ITU) Recommendation E.164 [E.164]. Each address is one value of this multi-valued attribute. Example: "0198 444 444".
l	Names of a locality or place, such as a city, county, or other geographic region. Each name is one value of this multi-valued attribute. Examples: "Austin", "Chicago", and "Brisbane".
mail	The RFC822 mailbox for the user.
manager	Distinguished name of the manager to whom this person reports.
mobile	Mobile telephone number of this person.

## Schema Attributes

**Table 1—LDIF Connector - Account Attributes (Continued)**

Name	Description
o	Names of an organization. Each name is one value of this multi-valued attribute. Examples: "xyz", "xyz Technologies, Inc.", and "xyz, Incorporated.".
ou	Names of an organizational unit. Each name is one value of this multi-valued attribute. Examples: "Sales", "Human Resources", and "Information Technologies".
pager	Telephone number of this persons pager.
physicalDeliveryOfficeName	Names that a Postal Service uses to identify a specific post office. Examples: "Austin, Downtown Austin" and "Chicago, Finance Station E".
postOfficeBox	Postal box identifiers use by a postal service to locate a box on the premises of the Postal Service rather than a physical street address. Each postal box identifier is a single value of this multi-valued attribute. Example: "Box 27".
postalAddress	Addresses used by a Postal Service to perform services for the object. Each address is one value of this multi-valued attribute. Example: "1111 Elm St.\$Austin\$Texas\$USA".
postalCode	Codes used by a Postal Service to identify postal service zones. Each code is one value of this multi-valued attribute. Example: "78664", to identify Pflugerville, TX, in the USA.
preferredDeliveryMethod	An indication of the preferred method of getting a message to the object. Example: If the mhs-delivery Delivery Method is preferred over telephone-delivery, which is preferred over all other methods, the value would be: "mhs \$ telephone".
preferredLanguage	Preferred written or spoken language of this person.
registeredAddress	Postal addresses to be used for deliveries that must be signed for or require a physical recipient. Each address is one value of this multi-valued attribute. Example: "Receptionist\$xyz Technologies\$6034 Courtyard Dr. \$Austin, TX\$USA".
roomNumber	Room, office number or this persons normal work location.
secretary	Distinguished name of this persons secretary.
seeAlso	Distinguished names of objects that are related to the subject object. Each related object name is one value of this multi-valued attribute. Example: The person object "cn=Elvis Presley,ou=employee,o=xyz\, Inc." is related to the role objects "cn=Bowling Team Captain,ou=sponsored activities,o=xyz\, Inc." and "cn=Dart Team,ou=sponsored activities,o=xyz\, Inc.". Since the role objects are related to the person object, the 'seeAlso' attribute will contain the distinguished name of each role object as separate values.
sn	Name strings for surnames or family names. Each string is one value of this multi-valued attribute. Example: "Smith".

**Table 1—LDIF Connector - Account Attributes (Continued)**

Name	Description
st	Full names of states or provinces. Each name is one value of this multi-valued attribute. Example: “Texas”.
street	Site information from a postal address (that is, the street name, place, avenue, and the house number). Each street is one value of this multi-valued attribute. Example: “15 Main St.”.
telephoneNumber	Telephone numbers that comply with the ITU Recommendation E.123 [E.123]. Each number is one value of this multi-valued attribute.
teletexTerminalIdentifier	The withdrawal of recommendation F.200 has resulted in the withdrawal of this attribute.
telexNumber	Sets of strings that are a telex number, country code, and answerback code of a telex terminal. Each set is one value of this multi-valued attribute
title	Persons job title. Each title is one value of this multi-valued attribute. Examples: “Vice President”, “Software Engineer”, and “CEO”.
uid	Computer system login names associated with the object. Each name is one value of this multi-valued attribute. Examples: “s9709015”, “admin”, and “Administrator”.
objectClass	The values of the objectClass attribute describe the kind of object which an entry represents. The objectClass attribute is present in every entry, with at least two values. One of the values is either “top” or “alias”.

## Group attributes

---

The group schema is used when building AccountGroup objects which are used to hold entitlements shared across identities.

**Table 2—LDIF Connector - Group Attributes**

Name	Description
cn	Names of object. Each name is one value of this multi-valued attribute. If the object corresponds to a person, it is typically the person's full name. Examples: “Martin K Smith”, “Marty Smith” and “printer12”.
uniqueMember	Groups to which this person is a unique member.
dn	Directory path to the object.
o	Names of an organization. Each name is one value of this multi-valued attribute. Examples: “xyz”, “xyz Technologies, Inc.”, and “xyz, Incorporated.”.
ou	Names of an organizational unit. Each name is one value of this multi-valued attribute. Examples: “Sales”, “Human Resources”, and “Information Technologies”.

## Schema Attributes

**Table 2—LDIF Connector - Group Attributes**

Name	Description
owner	Distinguished names of objects that have ownership responsibility for the object that is owned. Each owner's name is one value of this multi-valued attribute. Example: The mailing list object, whose DN is “cn=All Employees, ou=Mailing List,o=xyz, Inc.”, is owned by the Human Resources Director. Therefore, the value of the ‘owner’ attribute within the mailing list object, would be the DN of the director (role): “cn=Human Resources Director,ou=employee,o=xyz, Inc.”.
description	Human-readable descriptive phrases about the object. Each description is one value of this multi-valued attribute. Examples: “Updates are done every Saturday, at 1am.”, and “distribution list for sales”.

# Chapter 12: SailPoint Logical Connector

---

The following topics are discussed in this chapter:

Overview .....	121
Configuration parameters .....	121
Schema attributes .....	121
Additional information .....	122
Logical Connector - Tiers Tab .....	122
Defining Logical Connectors .....	124
Logical Application Filtering.....	124

## Overview

---

The SailPoint Logical Connector is a *read only* connector developed to create objects that function like applications, but that are actually formed based on the detection of accounts from other, or tier, applications in existing identity cubes.

For example, you might have one logical application that represents three other accounts on tier applications, an Oracle database, an LDAP authorization application, and a custom application for internal authentication. The logical application scans identities and creates an account on the logical application each time it detects the three required accounts on a single identity.

You can then use the single, representative account instead of the three separate accounts from which it is comprised for certification, reporting, and monitoring.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Logical applications do not have connection attributes, by default. If you have defined custom logical connectors there might be connection attributes on this tab.

Use this tab to test your logical application connection.

## Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports two types of objects, account and group. Account objects are used when building identities Link objects. The group schema is used when building AccountGroup objects which are used to hold entitlements shared across identities.

Logical applications enable you to pull schema attribute information from the tier applications from which it is compiled. When you use this feature the schema attribute information is automatically added to the attributes table and you can edit it as needed.

## Additional information

Click **New Tier Attribute** to display the Select Source Attribute dialog and select the tier application and attribute to pull into the logical application.

# Additional information

---

This section describes the additional information related to the Logical Connector.

## Logical Connector - Tiers Tab

---

This section contains the information that this connector uses to build the relationships between the tier applications that make up a logical application. For an identity to have an account on a logical application they must have the required, matching accounts on all tier applications. For example an identity, Lori Ferguson, might be represented by the attribute `dbid` on one tier and `username` on another. You must correlate those attributes, either manually or with a correlation rule, to create accounts on the logical application.

### Add Tiers to a Logical Application

You must define the tier applications that are contained within the logical application and identify the application to be used as the primary tier application.

To add a tier application, select the application from the Select an Application drop-down list and click **Add Tier**. Click the arrow to right of the field to display all applications configured to work with IdentityIQ or type the first few letters of an application name to display a list of applications with names containing that letter string. You can add as many applications as required.

Specify the primary tier application by selecting it in the Primary Tier column. The primary tier application is the application containing all of the attributes to which the attributes on the other tiers will correlate. Every account on the logical application must have an account on the primary tier application. In some instances this might be a human resources application containing all of the identities. A logical application can only have one primary tier application.

To remove tier applications, select the application using the selection boxes in the left-most column and click **Remove Selected**.

### Correlate Tier Application Attributes

Use the logical application tier attribute mapping, or correlation, panel to either manually map attributes for correlation or assign an existing correlation rule. For an identity to have an account on a logical application they must have the required, matching, accounts on all tier applications. Map the attributes on each application that should have matching values.

To manually map attributes on the tier applications do the following:

1. Select a non-primary tier application in the application list. The selected application is highlighted and any mapped correlation attributes are displayed in the attribute correlation panel.  
If you select the primary tier application a note is displayed stating that no correlation is required on the primary tier.
2. Click **Add Attribute** to display a row in which to add the new attribute.
3. Click on the row to activate either the **Tier Attribute** or **Primary Tier Attribute** field.
4. Select an attribute from the drop-down list in both columns.

5. Click **Save Changes** or continue mapping attributes for the applications.

To use an existing correlation rule, open the Use Correlation Rule panel and select a rule from the **Correlation Rule** drop-down list. The rule should contain all of the attribute mapping required for this logical application.

The Tiers tab contains the following information:

**Table 1—Logical Connector - Tier Applications**

Attribute	Description
Account Rule	Select an existing account rule from the drop-down list.  The logical application rule defines the requirements that must be met before an identity is assigned an account on this logical application. <b>Note: Click the “...” icon to launch the Rule Editor to make changes to your rules if needed.</b>
Provisioning Rule	Select an existing provisioning rule from the drop-down list.  The logical provisioning rule defines how provision requests for the logical application account or any of the accounts with which it is comprised are handled. <b>Note: Click the “...” icon to launch the Rule Editor to make changes to your rules if needed.</b>
Application	The tier applications that make up the logical application.
Primary Tier	Designate one tier application as the primary tier application. The primary tier application is the application containing all of the attributes to which the attributes on the other tiers will correlate. Every account on the logical application must have an account on the primary tier application. In some instances this might be a human resources application containing all of the identities in IdentityIQ.  <b>Note: A logical application can only have one primary tier application.</b>
Tier Attribute	Attributes from the selected tier application whose values must match the values of the associated attributes from the primary tier application.
Primary Tier Attribute	Attributes on the primary tier application to which the attribute values from the tier applications must match.

**Table 1—Logical Connector - Tier Applications**

Attribute	Description
Account Matching	<p>Use account matching to select attributes and permissions from existing application tiers as the parameters for your logical application. This panel contains the following:</p> <p><b>Application Items</b> — Click <b>Add Attribute</b> to include application attributes in your account matching parameters. Click <b>Add Permission</b> to include application permissions in your account matching parameters.</p> <p><b>Operation</b> — choose the AND / OR operator to include multiple attributes / permissions</p> <p><b>Type</b> — indicates either Attribute or Permission</p> <p><b>Application</b> — indicates the application from which the attribute or permission is being matched</p> <p><b>Name</b> — select an attribute from the drop-down list or input the permission name into the field</p> <p><b>Value</b> — input the value of the attribute or permission</p> <p><b>Group/Ungroup/Delete Selected</b> — use the check box to select line items on which to perform the respective action</p>

## Defining Logical Connectors

---

Use the following procedure to define a logical connector.

1. Define all tier applications.
2. Perform the following tasks on each tiered application:
  - a. Run aggregation task.
  - b. Run entitlement correlation task.
  - c. Scan for missing entitlements or define new managed entitlements.
3. Define the logical application
  - a. Define application tiers
  - b. Discover schema attributes from selected tier applications for editing.
  - c. Scan for missing entitlements using the filters from the selected tiered applications for editing.
4. Run aggregation task on your newly defined logical application.
5. (*Optional*) Run Account-Group Aggregation task on the newly defined logical application.  
This will update the logical application entitlements to have the configured display value for the respective groups. The tier application information used to update the entitlement is based upon the logical applications configured “Group Attribute” from its Account Schema.

## Logical Application Filtering

---

Logical applications use the **Find missing entitlement** scan on the Managed Entitlements tab as filtering action using the Account Matching criteria provided on the Tiers tab. This gives a more focused starting point instead of using all of the entitlement values from the selected application tiers.

For example, a new logical application uses the “memberOf” attribute in Active Directory. There are likely thousands of values that are assigned in an enterprise. With specific criteria defined in Account Matching, only the values you are interested in appear for easier editing.



## **Additional information**

# Chapter 13: SailPoint Lotus Domino Connector

---

The following topics are discussed in this chapter:

Overview .....	127
Supported features .....	127
Supported Managed Systems .....	128
Pre-requisites .....	128
Administrator permissions .....	129
Configuration parameters .....	129
Schema attributes .....	130
Account attributes .....	130
Group attributes .....	132
Provisioning policy attributes .....	133
Create account attributes .....	133
Create group attributes .....	136
Update policies .....	136
Additional information .....	138
ID Vault functionalities .....	138
Password management .....	138
Troubleshooting .....	139

## Overview

---

SailPoint Lotus Domino Connector manages the accounts and groups contained in a Notes database.

### Supported features

---

SailPoint Lotus Domino Connector supports the following features:

- Account Management
  - Manage Lotus Domino Users as Account
  - Aggregate, Delta Aggregation, Refresh Accounts, Pass Through Authentication (uses HTTP password)
  - Create, Update, Delete (Update attributes, Rename, Re-certify, Move user to a different certifier)
  - Enable, Disable, Unlock, Change Password (HTTP (default) and ID file)
 

**Note:** For Self Service HTTP Password Change without specifying the current password, user has to remove the CURRENT\_PASSWORD feature string from the application xml file. For ID File Password Change, without specifying the CURRENT\_PASSWORD feature string add the IDFileCurrentPassword attribute in change password provisioning.

**Note:** Lotus Domino Connector supports attaching ID file to mail file at the time of change password. For more information, see “Attach ID File to Mail File” on page 137.
  - ID Vault functionalities: Reset Password, Extract ID from vault, Upload ID to vault, Sync ID file

## Overview

- Account - Group Management
  - Manage Lotus Domino Groups as Account-Groups
  - Aggregate, Refresh Groups, Delta Aggregation
  - Create, Update, Delete

## References

- “Password management” on page 138
- “Appendix A: Delta Aggregation”
- “Appendix E: IQService”

## Supported Managed Systems

---

- Domino Server 9.0.x (for “ ID Vault functionalities”)
- Domino Server 8.5.x (for “ ID Vault functionalities”)

## Pre-requisites

---

- The computer must have the `NCSO.jar` file in the classpath.
- Ensure that Domino server `notes.ini` file contains the following line:  
`ServerTasks=<any other tasks>, DIIOP, HTTP`  
HTTP task is required to be mentioned only if the DIIOP port is not a part of the HostName in the Application attributes.
- In the Domino server, select **Server => Full Access Administrators** should have the name of the user which is being used to open a session with the server.
- Domino Server should be reachable from the host computer.
- The IQService is a native Windows service that enables this connector to participate in a Windows environment and access information only available through Windows APIs.

IQService must be installed before performing the following operations:

- Sync ID file
- Delta Aggregation
- Upload ID file to vault
- Get ID file from vault
- Reset password of an ID file stored in an ID Vault
- ID File password change through self-service
- Helpdesk HTTP (Internet) Password change.

**Note:** For more information on IQService, see “Appendix E: IQService”.

**Note:** Lotus Domino Connector requires Microsoft Visual Studio C++ 2015 Redistributable 14.0 (32-bit) installed on the computer where IQService is installed.

## Administrator permissions

---

The Administrator user should have Manager Access to the following databases on Domino Server:

- Public Address Book (PAB) Database (default name is `names.nsf`)
- Administration Requests Database (default name is `admin4.nsf`)
- Certification Log Database (default name is `certlog.nsf`)

## Configuration parameters

---

**Note:** All paths are with respect to the Domino Server computer. For example, ID file path, mail file path and so on. These paths must be accessible from the Domino Server computer.

The following table lists the configuration parameters of Lotus Domino Connector:

Parameters	Description
IQService Host	Host Name of the computer on which IQService is installed.
IQService Port	IQService port number.
Admin ID File Path	Administrator ID file path required by IQService.
Host Name	Fully qualified host name of the Domino Server. The DIIOP port should be a part of HostName if the HTTP task is not mentioned in the Server Tasks (refer Pre-requisites section). In this case, the HostName should be <code>fullyQualifiedHostName:DIIOPPortNumber</code> . For example, <code>sailpoint.server.com:63148</code>
Admin Name	Name of the Database Administrator which must be in the format <code>Administrator/CertifierName</code> .
Admin Password	HTTP password of administrator account.
Database Name	Name of the database to be managed. For example, <code>names.nsf</code>
Server Name	Name of the server to be managed. For example, <code>Lotus/IBM</code>
Search formula - Accounts	<p>Search formula to be used during Account Aggregation.</p> <p>The Search formula follows the following format:</p> <p><code>Domino @Formula language</code></p> <p>For example,</p> <ul style="list-style-type: none"> <li>• For non-indexed database search: <code>SELECT @UpperCase(Type) = "PERSON"</code></li> <li>• For indexed database search: <code>[Type] = "Person"</code></li> </ul>
Search formula - Groups	Search formula to be used during Group Aggregation.

## Schema attributes

Parameters	Description
Indexed database	<p>Specifies if the database is indexed or not.</p> <ul style="list-style-type: none"><li>• Y - The database is indexed</li><li>• N - The database is not indexed</li></ul> <p><b>Note:</b> A maximum of 5,000 documents will be returned by default. The <b>FT_MAX_SEARCH_RESULTS</b> variable in <b>Notes.ini</b> file overrides this limit for indexed databases or databases that are not indexed but that are running an agent on the client. For a database that is not indexed and is running in an agent on the server, set the <b>TEMP_INDEX_MAX_DOC</b> variable in the <b>Notes.ini</b> file. The absolute maximum value is 2147483647.</p>
Indexed search interval in seconds	<p>(Applicable only when <b>Indexed Database</b> attribute is set to <b>Y</b>) In case of an exception for indexed search, the number of seconds to wait until an indexed database search query should be fired again.</p> <p>Add the attribute to application xml with key as <b>idxInterval</b>. For example, <code>&lt;entry key="idxInterval" value="15"/&gt;</code></p>

## Additional configuration parameters

Parameters	Description
groupMembersSize	<p>Specifies maximum size in bytes of the group membership attribute value for any group. The connector returns error if the size of membership attribute value is exceeding the limit at the time of adding user to a group.</p> <p>For example, if groupMembersSize is 23000 and the size of group g1's 'Members' attribute is 23400 bytes, then an exception will be thrown when an add user to group g1 will be performed.</p>

## Schema attributes

This section describes the different schema attributes.

**Note:** For an attribute to be multivalued on managed system side, change the attribute type to multivalued in schema.

## Account attributes

The following table lists the account attributes:

Attributes	Description
NOTEID	NOTEID of the user.
UserName	The user full name.
Type	The type of the document.

Attributes	Description
Owner	The owner of the document.
MailSystem	The type of mail system.
InternetAddress	Mail internet address.
JobTitle	Job title of the user.
CompanyName	Company name of the user.
Department	Department of the user.
EmployeeID	EmployeeID of the user.
Location	Location of the user.
Manager	Manager of the user.
OfficePhoneNumber	Office phone number of the user.
OfficeFAXPhoneNumber	Office fax phone number of the user.
CellPhoneNumber	Cell phone number of the user
PhoneNumber_6	Phone number_6 of the user.
Assistant	Assistant of the user.
OfficeStreetAddress	Office street address of the user.
OfficeCity	Office city of the user.
OfficeState	Office state of the user.
OfficeZIP	Office ZIP/Postal of the user.
OfficeCountry	Office country of the user.
OfficeNumber	Office number of the user.
StreetAddress	Street address of the user.
City	City of the user.
State	State of the user.
Zip	Zip/Postal code of the user.
Country	Country of the user.
PhoneNumber	Phone number of the user.
HomeFAXPhoneNumber	Home fax phone number of the user.
Spouse	Spouse of the user.
Children	Children of the user.
PersonalID	PersonalID of the user.
Comment	Office number of the user.
WebSite	Address of the user Web Page.
PhotoURL	Photo URL of the user.
LocalAdmin	Local Admin of the user.

## Schema attributes

Attributes	Description
CheckPassword	Check password of the user.
PasswordChangeInterval	Password change interval of the user.
PasswordGracePeriod	Password grace period of the user.
PasswordDigest	Password digest of the user.
Policy	Policy of the user.
Profiles	Profiles of the user.
ClientType	Type of the client.
PostalAddress	Postal address of the user.
HomePostalAddress	Home postal address of the user.
Street	Street of the user.
BusinessCategory	Business category of the user.
CarLicense	Car license of the user.
DepartmentNumber	Department number of the user.
EmployeeNumber	Employee number of the user.
EmployeeType	Employee type of the user.
FirstName	First name of the user.
MiddleInitial	Middle initials of the user.
LastName	Last name of the user.
FullName	Full name of the user.
ShortName	Short name of the user.
MailDomain	Mail domain of the user.
MailServer	Mail server of the user.
MailFile	Mail file of the user.
PasswordChangeDate	Password change date of the user.
HTTPPasswordChangeDate	HTTP password change date of the user.
SametimeServer	Home sametime server of the user.
\$UpdatedBy	Name of the user who last updated the user document.
Groups	A list of groups of which the user is a member of.

**Note:** If the FullName/UserName attributes are to be updated, the attribute name to be used in the policy must be 'UserName'.

## Group attributes

---

The following table lists the group attributes:

Attributes	Description
GroupType	Type of the group.
ListDescription	Description of the group.
MailDomain	Mail domain of the group.
InternetAddress	Internet address of the group.
Comments	Comments about the group.
ListOwner	Owner of the group.
LocalAdmin	Local administrator of the group.
ListName	Name of the group.
\$UpdatedBy	Name of the user who last updated the group document.

## Provisioning policy attributes

---

This section lists the different policy attributes of Lotus Domino Connector.

**Note:** In this section all the attributes marked with the \* sign indicate that the attributes are mandatory.

**Note:** All paths are with respect to the Domino Server computer. For example, ID file path, mail file path and so on. These paths must be accessible from the Domino Server computer.

## Create account attributes

---

The following table lists the provisioning policy attributes for Create Accounts:

Attributes	Description
ServerName*	The name of the server on which the account should be created.
CertifierIDfile	The ID file path of the certifier. For example, c:\id\cert.id
CertifierPassword	The password of the certifier ID file.
FirstName	First name of the user.
MiddleInitial	Middle name initial of the user.
LastName*	Last name of the user.
FullName*	Full name of the user. The format is as follows: FirstName LastName/CertifierName.
IDFilePath*	ID file path of the user. For example, c:\id\user.id
UserIDFilePassword*	ID file password of the user.

## Provisioning policy attributes

Attributes	Description
IDType*	Type of the ID file. Following are the permissible values for the keyword: <ul style="list-style-type: none"> <li>• FLAT</li> <li>• HIERARCHICAL</li> <li>• CERTIFIER</li> </ul>
MinimumPasswordLength	Minimum length of the ID file password.
IDFileIsNorthAmerican	Indicates whether the id file is North American or not. Following are the permissible values for the keyword: <ul style="list-style-type: none"> <li>• Y: indicates that the ID file is North American</li> <li>• N: indicates that the ID file is not North American</li> </ul>
StoreIDInAddressbook	Indicates whether the ID file should be stored in the address book or not. Following are the permissible values for the keyword: <ul style="list-style-type: none"> <li>• Y: indicates that the ID file should be stored in the address book</li> <li>• N: indicates that the ID file should not be stored in the address book.</li> </ul>
MailServer	Server on which the mail file should be created.
MailSystem	Specifies the type of the mail system. Following are the permissible values for the keyword: <ul style="list-style-type: none"> <li>• NOTES</li> <li>• POP</li> <li>• IMAP</li> <li>• INOTES</li> <li>• INTERNET</li> <li>• OTHER</li> <li>• NONE</li> </ul>
MailInternetAddress	Internet address for the mail.
MailTemplateName	Name of the mail template.
MailForwardingAddress	Forwarding address for the mail.
MailFileName	Name of the mail file. For example, mail/mailfilename.nsf
MailReplicaServer	The names of the servers on which the mail file replicas should be created. Applies only to clustered servers. Should be multi-valued.
MailACLManager	A name that is assigned to the Manager for accessing the mail database ACL. The format should be as, FirstName LastName/CertifierName.
MailOwnerAccess	The mail database ACL setting for the owner. Allowed values are MANAGER, EDITOR and DESIGNER
MailQuotaSizeLimit	The maximum size of the user's mail database, in megabytes.
MailQuotaWarningThreshold	The size in megabytes, at which the user's mail database issues a warning that it is getting too large.
ShortName	The short name of the user.

Attributes	Description
CreateMailDatabase	<p>Indicates whether the mail database should be created or not. Following are the permissible values for the keyword:</p> <ul style="list-style-type: none"> <li>• <b>Y:</b> indicates that a mail database should be created for the user</li> <li>• <b>N:</b> indicates that a mail database should not be created for the user; it will be created during setup.</li> </ul>
StoreIDInMailFile	<p>Indicates whether the ID should be stored in mail file or not. Following are the permissible values for the keyword:</p> <ul style="list-style-type: none"> <li>• <b>Y:</b> indicates that the ID file should be stored in mail file.</li> <li>• <b>N:</b> indicates that the ID file should not be stored in mail file.</li> </ul>
SynchInternetPassword	<p>Indicates whether the ID password and internet password should be in synchronization. Following are the permissible values for the keyword:</p> <ul style="list-style-type: none"> <li>• <b>Y:</b> indicates that the ID file password and internet password should be in synchronization</li> <li>• <b>N:</b> indicates that the ID file password and internet password should not be in synchronization</li> </ul>
ExpirationPeriod	<p>The expiration period in years. For example, if 20 is specified and the current year is 2013, the expiration period will be 2033.</p>
RegistrationLog	<p>No logging occurs if this parameter is null. If this parameter has a value other than null, logging goes to the <b>certlog.nsf</b> file in the Domino data directory on the registration server.</p>
EnforceUniqueShortName	<p>Indicates whether a unique short name should be used. Following are the permissible values for the keyword:</p> <ul style="list-style-type: none"> <li>• <b>Y:</b> indicates that the short name should be unique</li> <li>• <b>N:</b> indicates that the short name may or may not be unique</li> </ul>
PolicyName	<p>Name of the explicit policy.</p>
RoamingUser	<p>Indicates whether a user is roaming or not. Following are the permissible values for the keyword:</p> <ul style="list-style-type: none"> <li>• <b>Y:</b> indicates that the user is roaming</li> <li>• <b>N:</b> indicates that the user is not roaming</li> </ul>
UseCAProcess	<p>Specifies if CA process should be used at the time of user creation. Values: Y, N</p>
CertifierName	<p>(Required if UseCAProcess attribute is enabled) Name of the certifier of format /ABC/rootCert.</p>
RoamingCleanupSetting	<p>Indicates the clean-up process for data on Notes clients set up for roaming users. The values are as follows:</p> <ul style="list-style-type: none"> <li>• NEVER CLEANUP</li> <li>• CLEANUP EVERY N DAYS</li> <li>• CLEANUP AT SHUTDOWN</li> <li>• CLEANUP PROMPT</li> </ul>

## Provisioning policy attributes

Attributes	Description
RoamingCleanupPeriod	(The RoamingCleanupSetting attribute must be CLEANUP EVERY N DAYS). The interval in days for cleaning up data on Notes clients set up for roaming users.
RoamingServer	The server on which the user's roaming data is stored.
RoamingSubdir	The subdirectory that contains the user's roaming data. For example, roaming\TestUser

**Note:** All attributes should be of type 'String'.

## Create group attributes

The following table lists the provisioning policy attributes for Create Group:

Attributes	Description
ListName	Name of the group to be created.

**Note:** Only the name of the group is required at the time of group creation. Even if the other attributes are specified they will not be set. After creation, the group will be of type 'Multi-Purpose'.

## Update policies

**Note:** In update policies for account/group, the attribute names must be the same as their corresponding names in the document properties of account/group on Lotus Notes. For updating the FullName of an account, the name of the attribute should be UserName.

The following table lists the attributes for different update policies:

Attributes	Description
<b>Rename a user</b>	
AC_Operation*	The value of this attribute should be <b>Rename User</b> .
AC_certifierFilePath*	The location of the Certifier ID file of the user. For example, c:\id\cert.id
AC_certifierPassword*	The Certifier ID file password.
AC_lastName	New last name of the user.
AC_middleInitial	New middle name initial of the user
AC(firstName	New first name of the user.
AC_orgUnit	New organizational unit of the user.
AC_altOrgUnit	New alternate organizational unit of the user.
AC_altLanguage	New alternate language of the user.
AC_renameNotesUser*	If you want to rename Notes User or not. Values: True or False.
<b>Recertify a user</b>	
AC_Operation*	The value of this attribute should be <b>Recertify User</b> .

Attributes	Description
AC_certifierFilePath*	The location of the Certifier ID file of the user. For example, c:\id\cert.id
AC_certifierPassword*	The Certifier ID file password.
<b>Move a user</b>	
AC_Operation*	The value of this attribute should be <b>Move User</b> .
AC_currentCertifierFilePath*	The location of the Certifier ID file of the user. For example, c:\id\cert.id
AC_currentCertifierPassword*	The Certifier ID file password.
AC_targetCertifierFilePath*	The location of the Certifier ID file of the user. For example, c:\id\target.id
AC_targetCertifierPassword*	The Certifier ID file password.
AC_targetCertifierName*	The name of the target certifier.
<b>Change/Reset password of a user</b>	
HTTP_PASSWORD_CHANGE	Should be set to <b>Yes</b> to change the HTTP (Internet) Password of a user. Values: <b>Yes</b> (Default) and <b>No</b> .
IDFileCurrentPassword	(Used only when CURRENT_PASSWORD feature string is not present) The current password of the ID file.
IDFilePath	The location where the user ID file is stored. For example, c:\id\user.id should be provided to change the ID file password of a user.
RESET_PASSWORD	Should be set to <b>Yes</b> to reset the password of an ID file stored in the vault. Values: <b>Yes</b> and <b>No</b> (Default).
<b>Sync ID File</b>	
Operation*	The value of this attribute should be <b>Sync ID File</b> .
IDFilePath*	The location of the User ID file of the user. For example, c:\id\user.id
IDFilePassword*	The User ID file password.
<b>Get ID File</b>	
Operation*	The value of this attribute should be <b>Get ID File</b> .
IDFilePath*	The location where the User ID file should be stored. For example, c:\id\user.id
IDFilePassword*	The User ID file password.
<b>Upload ID File</b>	
Operation*	The value of this attribute should be <b>Upload ID File</b> .
IDFilePath*	The location where the User ID file is stored. For example, c:\id\user.id
IDFilePassword*	The User ID file password.
<b>Attach ID File to Mail File</b>	

## Additional information

Attributes	Description
ATTACH_ID_TO_MAIL	Specifies if ID File should be attached to Mail File or not. Values: <b>Yes</b> and <b>No</b> (Default).
MailServer	Mail server of the users mail file.
MailFile	Mail File path. For example, <code>mail\userMail.nsf</code>

**Note:** No default value needs to be assigned for optional attributes if those need not to be set.

# Additional information

---

This section describes the additional information related to the Lotus Domino Connector.

## ID Vault functionalities

---

ID Vault is a feature introduced by IBM in Domino version 8.5. The following functionalities are a part of ID Vault which are supported through IQService:

- **Reset Password:** Allows the Help Desk Personnel to reset the password of the ID file for a vaulted user. This requires application Administrator to have password reset authority.
- **Extract ID file from vault:** A vault administrator assigned to the Auditor role in the vault database ACL can extract an ID from a vault to gain access to a user's encrypted data. A copy of the ID remains in the vault after extraction.
- **Upload ID file to vault:** Upload an ID file that has not yet been uploaded to the vault.
- **Sync ID file:** Synchronizes the local ID file with the copy in ID vault.

Check the provisioning policy for each of the above transactions.

## Password management

---

Administrator password reset (Change password for others):

- HTTP Password
- Password of the vaulted ID file (Reset password)

Self-service password change:

- HTTP Password
- ID File Password
- Password of the ID file which is vaulted (Reset password)

**Note:** Ensure that self-service change password with the IBM Lotus Domino connector is successful after adding the 'ValidateHttpPassword' attribute, even if the current passwords for HTTP Password and ID File are different.

# Troubleshooting

---

## 1 - Could not get IOR from Domino Server

**Resolution:** Perform the following:

1. Check if the Domino Server is accessible from a computer which is using the Fully Qualified Internet Host Name. The ping should be successful using the Fully Qualified Internet Host Name of the Domino Server.
2. Check if DIIOP is present in the ServerTasks of notes.ini file.
3. If HTTP is not added to **notes.ini** file ServerTasks, the HostName in the Application Parameter should include the port number of the DIIOP Server in the following format:  
fullyQualifiedInternetHostName : DIIOPPortNumber  
For example, LOTUS-AME.SAILPOINT.COM:63148
4. Check if NCSO.jar file is present in the CLASSPATH environment variable.

## 2 - Could not open the ID file

**Resolution:** Perform the following:

1. All paths in the connector are with respect to the Domino Server. Verify if the ID file path you have provided is accessible from the Domino Server computer.
2. The ID files will be read from and created on a path with respect to the Domino Server.

## 3 - Add Account gives Object does not exist exception

**Resolution:** Perform the following:

1. If the name of the user you created is Derek Stevens and the name of the certifier under which the user was created is /USA then the following attributes should be populated:
  - FirstName: Derek
  - LastName: Stevens
  - FullName: Derek Stevens/USA
 Add account searches a user based on the FullName of the user, hence it is important that it is provided correctly.

## 4 - IQService - Unable to load DLL 'SPLotusNotesWrapper.dll': The specified module could not be found or the IQService stops responding

**Resolution:** Perform the following:

1. Verify if the PATH system variable contains the Notes data folder. For example, C:\Program Files\IBM\Notes must be present in the PATH system variable.
2. Verify if you have restarted the computer after modifying the PATH system variable.
3. Close the Notes Administrator/Client and restart the IQService.
4. Copy the IQService installation files in the Notes folder of IBM Lotus Notes Client. For example: C:\Program Files\IBM\Notes

## Troubleshooting

### 5 - Attempt to load a program with an incorrect format

An attempt to load a program with an incorrect format, displays the following error message:

Unable to create iterator: sailpoint.connector.ConnectorException: Errors returned from IQService.

**Resolution:** Ensure that all the pre-requisites mentioned in the “Pre-requisite for Lotus Domino Connector” on page 578 are performed.

### 6 - ID file could not be opened

When creating an account from IdentityIQ, the following error message appears:

sailpoint.connector.ConnectorException: Notes error: Could not open the ID file

**Resolution:** The cert.id file must be present on the Server computer.

### 7 - During Delta Aggregation an exception error is displayed

While performing Delta Aggregation, the following error message is displayed:

Exception during aggregation. Reason: sailpoint.connector.ConnectorException: Errors returned from IQService. Object reference not set to an instance of an object.

**Resolution:**

1. While performing Delta aggregation or resetting the password, ensure that the Domino Client is closed.
2. If Domino Client is open and you perform Delta Aggregation, an exception error message is displayed.  
In such scenario, if you close the Domino Client and perform the same operation, the transaction appears to be in pending or idle state.
3. Restart the Domino Server and terminate / cancel such request in IdentityIQ.

### 8 - Test connection fails with Invalid user name/password

Lotus Notes test connection fails with invalid user name/password.

**Resolution:** Ensure that the administrator password is a HTTP password and not ID file password.

# Chapter 14: SailPoint Linux Connector

---

The following topics are discussed in this chapter:

Overview .....	141
Supported features .....	141
Supported Managed Systems .....	142
Pre-requisites .....	142
Administrator permissions .....	142
Configuration parameters .....	143
Additional configuration parameters for SSH configuration .....	143
Public key authentication configuration .....	144
Schema attributes .....	144
Account attributes .....	144
Group attributes .....	145
Provisioning policy attributes .....	145
Account attributes .....	146
Group attributes .....	146
Additional information .....	147
Unstructured Target Collector .....	147
Troubleshooting .....	148

## Overview

---

The SailPoint Linux Connector was developed to enable user managing their Linux Account, Groups and resources. The Linux system data will be aggregated and user would be able to edit entities and their attributes.

### Supported features

---

The SailPoint Linux Connector supports the following features:

- Account Management
  - Manage Linux Users as Account
  - Aggregate, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements

## Overview

- Account Group Management
  - Manage Linux Groups as Account-Groups
  - Aggregate, Refresh Groups
  - Create, Update, Delete
- Permissions Management
  - Application can be configured to read permissions directly assigned to accounts and groups using Unstructured Target Collector.
  - The connector supports automated revocation of the aggregated permissions for accounts and groups.

**Note:** Linux connector supports MD5, SHA-1, and SHA-2 cryptographic hash functions.

## References

- “Appendix D: Before and After Provisioning Action”
- “Additional information” on page 147

## Supported Managed Systems

---

The Linux connector supports the following versions of the operating system:

- Red Hat Enterprise Linux versions 7.2, 7.1, 7.0, 6.8, 6.7, 6.6, 6.5, 6.3, 6.2, 6.1, 6.0
- SUSE Linux Enterprise Server 12 and 11

**Note:** For any issues related to SUSE Linux, see “Troubleshooting” on page 148 section.

## Pre-requisites

---

SSH should be installed on Linux computer.

## Administrator permissions

---

- You can use root user for managing your applications.
- If you want to use sudo user to perform the provisioning operations, the sudo user must be configured with the following rights and permissions:

### Rights to execute the following commands with root privilege:

```
/bin/chmod, /usr/sbin/useradd, /usr/sbin/usermod, /usr/sbin/userdel,  
/usr/sbin/groupadd, /usr/sbin/groupmod, /usr/sbin/groupdel, /usr/bin/passwd,  
/usr/bin/faillog, /usr/bin/groups, /bin/rm, /bin/echo, /usr/bin/chage,  
/usr/bin/find, /bin/cat /etc/shadow, /bin/cat /etc/passwd, /bin/cat /etc/group,  
/bin/cat /etc/pam.d/system-auth, /usr/bin/getent, /bin/grep, /usr/bin/awk,  
/usr/bin/id, /usr/bin/lastlog, /usr/sbin/pam_tally2, /sbin/pam_tally2, /bin/cat  
/etc/pam.d/password-auth, /bin/cat /etc/pam.d/common-account, /bin/cat  
/etc/pam.d/common-auth, /usr/bin/printf
```

### An entry in /etc/sudoers file should look similar to the following:

```
username ALL = (root) PASSWD: /bin/chmod, /usr/sbin/useradd, /usr/sbin/usermod,  
/usr/sbin/userdel, /usr/sbin/groupadd, /usr/sbin/groupmod,
```

```
/usr/sbin/groupdel, /usr/bin/passwd, /usr/bin/faillog, /usr/bin/groups, /bin/rm,
/bin/echo, /usr/bin/chage, /usr/bin/find, /bin/cat /etc/shadow, /bin/cat
/etc/passwd, /bin/cat /etc/group, /bin/cat /etc/pam.d/system-auth,
/usr/bin/getent, /bin/grep, /usr/bin/awk, /usr/bin/id, /usr/bin/lastlog,
/usr/sbin/pam_tally2, /sbin/pam_tally2, /bin/cat /etc/pam.d/password-auth,
/bin/cat /etc/pam.d/common-account, /bin/cat /etc/pam.d/common-auth,
/usr/bin/printf
```

**Note:** All commands mentioned above are for default configuration. If any of the command is modified in application definition, then the respective changes in /etc/sudoers file entry should also be performed. Verify command paths on Linux computers as they might differ from the values mentioned here.

**Note:** If you want to use sudo user to perform the provisioning operations ensure to configure home directory with proper write access for this sudo user. In case sudo user is using Guest home directory then ensure it has proper write access over this directory.

## Configuration parameters

---

The following table lists the configuration parameters of Linux Connector:

Parameters	Description
Unix Server Host	Host Name/IP address of the computer.  <b>Note: For IdentityIQ version 6.4 Patch 4 and above, the format of the application XML has been changed from</b> <code>&lt;entry key="UnixServerHost" value="&lt;hostname&gt;" /&gt;</code> <b>to</b> <code>&lt;entry key="host" value="&lt;hostname&gt;" /&gt;</code>
SSH Port	SSH port configured. Default value: 22
User Name	User ID on the computer that you want to use for connector operations.
User Password	Password of the target system user account that you want to use for connector operations.
Not a 'root' user	If User ID specified is not root, check this parameter.
Private Key File Path	Path to Private Key File. Private/Public key authentication will have precedence over password authentication.
Passphrase For Private Key	Passphrase provided for creating Private Key.

## Additional configuration parameters for SSH configuration

---

The following procedure provides the steps for adding the additional configuration parameters for SSH configuration in Application or Target Source debug page.

**Note:** These additional configuration parameters must be added in the Application/Target Source debug page.

- Following is the default command for setting shell prompt on UNIX computer:

```
<entry key="SetPrompt" value="PS1='SAILPOINT>' />
```

## Schema attributes

In the above command, “SetPrompt” is the application/target source attribute and PS1='SAILPOINT' is the value of the application/target source attribute.

If the command for setting shell prompt is different than the default command, change the value of the application/target source attribute. If the application/target source attribute is not found, add the following new entry in the application/target source debug page:

For example: For tcsh shell, the entry value would be:

```
<entry key="SetPrompt" value="set prompt='SAILPOINT' " />
```

2. For executing the commands, verify that the default shell is present on your system.

If the default shell present on your UNIX system is different, modify the value of the application/target source attribute. If the application/target source attribute is not found, add the following new entry in the application/target source debug page:

```
<entry key="DEFAULT_SSH_SHELL" value="tcsh" />
```

## Public key authentication configuration

---

This is an alternative security method to using passwords. To use public key authentication, you must generate a public and a private key (that is, a key pair). The public key is stored on the remote hosts on which you have accounts. The private key is saved on the computer you use to connect to those remote hosts. This method allows you to log into those remote hosts, and transfer files to them, without using your account passwords.

Perform the following configuration steps to make the UNIX computer as the server and IdentityIQ computer as client:

1. Generate Private and Public key's. For more information of the standard steps, see “8 - Test connection fails for SUSE Linux” on page 151.
2. Append contents of public key file to `~/.ssh/authorized_keys` as shown below.  
`cat <public key file> >> ~/.ssh/authorized_keys`
3. The `~/.ssh/authorized_keys` file must have the read, write, and execute permissions in the `-rw-r--r--` format. Enter the following command to achieve the `-rw-r--r--` format permissions:  
`chmod 0644 ~/.ssh/authorized_keys`
4. Copy private key file to a location which is accessible by the server.
5. Provide path of private key file in application configuration.

**Note:** When generating public keys, if permission related issue occurs use the following command from user home directory (this overrides selinux policies):

```
chcon -t ssh_home_t .ssh
```

## Schema attributes

---

This section describes the different schema attributes.

### Account attributes

---

The following table lists the account attributes:

Attributes	Description
username	It is used when user logs in.
uid	Each user must be assigned a user ID (UID). UID 0 (zero) is reserved for root and UIDs 1-99 are reserved for other predefined accounts. Further UID 100-999 are reserved by system for administrative and system accounts/groups.
home	The absolute path to the directory the user will be in when they log in. If this directory does not exists then users directory becomes /
pwdlastchg	Days since Jan 1, 1970 that password was last changed.
pwdmin	The minimum number of days required between password changes that is, the number of days left before the user is allowed to change his/her password.
pwdmax	The maximum number of days the password is valid (after that user is forced to change his/her password).
pwdwarn	The number of days before password is to expire that user is warned that his/her password must be changed.
comment	Description
expiration	Days since Jan 1, 1970 that account is disabled that is, an absolute date specifying when the login may no longer be used.
inactive	The number of days after password expires that account is disabled.
lastlogin	Last login date and time of the Account.
primgrp	Name of primary group of the user.
groups	Secondary groups of user.
shell	User's shell.

## Group attributes

---

The following table lists the group attributes:

Attributes	Description
groupid	GID. Each user must be assigned a group ID. You can see this number in your /etc/group file.
name	It is the name of group. If you run ls -l command, you will see this name printed in the group field.

## Provisioning policy attributes

---

This section lists the different policy attributes of Linux Connector.

## **Account attributes**

---

The following table lists the provisioning policy attributes for Create Account:

<b>Attributes</b>	<b>Description</b>
User Name	It is used when user logs in. It should be between 1 and 32 characters in length.
User ID	Each user must be assigned a user ID (UID). UID 0 (zero) is reserved for root and UIDs 1-99 are reserved for other predefined accounts. Further UID 100-999 are reserved by system for administrative and system accounts/groups.
Home Directory	The absolute path to the directory the user will be in when they log in. If this directory does not exists then users directory becomes /
Min password change days	The minimum number of days required between password changes i.e. the number of days left before the user is allowed to change his/her password.
Max password validity	The maximum number of days the password is valid (after that user is forced to change his/her password).
Password change warning time	The number of days before password is to expire that user is warned that his/her password must be changed.
Comment	Description
Account expire duration	Days since Jan 1, 1970 that account is disabled that is, an absolute date specifying when the login may no longer be used.
Account inactivity time	The number of days after password expires that account is disabled.
Do not add to last login	Whether to add to last login log file.
Shell	User's shell.
Allow duplicate UID	Allow creation of account with a duplicate (non-unique) UID.
Create Home Directory	Whether to create home directory for new user.
Primary Group name	Specify primary group name.
Password	Initial password for newly created user account.
Force password change on next login	Specify if user has to be forced to change password on next logon.

## **Group attributes**

---

The following table lists the provisioning policy attributes for Create Group:

Attributes	Description
Group ID	GID. Each user must be assigned a group ID. You can see this number in your /etc/passwd file.
Group Name	It is the name of group. If you run ls -l command, you will see this name printed in the group field.
Allow duplicate GID	Duplicate GID of Group.

## Additional information

---

This section describes the additional information related to the Linux Connector.

### Unstructured Target Collector

---

Linux uses a data structure which requires the configuration in the **Unstructured Targets** tab to collect targeted data and correlate it with account **identityAttribute** for Accounts and group **identityAttribute** for Account Groups. For more information on the **Unstructured Targets** tab, see “Unstructured Targets Tab” section of the *SailPoint User’s Guide*.

For Linux target permission, the Unstructured Targets functionality will be enabled if **UNSTRUCTURED\_TARGETS** feature string is present in the application.

Multiple target sources can be specified and configured for an application which supports unstructured targets. This will be useful for applications which want to fetch resource information from multiple target sources.

Linux Target Collector support aggregation of file/directories under specified file system path(s). Only direct access permissions will be correlated to the Users and Groups. For UNIX platforms direct access means ownership of file or directory.

Attributes	Description	Possible values
Unix File System Path(s)*	Absolute path(s) which are to be scanned for resources.	Multiple paths can be mentioned with comma separated values. For example, /etc,/tmp
Application Name*	Name of the application with which Unstructured Target will be correlated.	

**Note:** Attributes marked with \* sign are the mandatory attributes.

**Note:** If Unstructured Configuration is configured before upgrading to version 7.1 from version 6.0 Patch 5 or 6.0 Patch 6, then update the configuration and specify the Connector Application Name.

### Rule configuration parameters

The rule configuration parameters are used to transform and correlate the targets.

## Troubleshooting

**Correlation Rule:** The rule used to determine how to correlate account and group information from the application with identity cubes in IdentityIQ.

**Note:** For version 6.2 onwards, the default schema does not have correlation keys defined. Update correlation rule in Unstructured Target Configuration accordingly.

### Provisioning related parameters

Select the settings for provisioning to the box.

- **Override Default Provisioning:** Overrides the default provisioning action for the collector.
- **Provisioning Action:** The overriding provisioning action for the collector.

## Troubleshooting

---

### 1 - Test connection failed on SUSE computer with an error message

Test connection failed on SUSE computer with the following error message:

```
Unexpected output captured from host: xxx.xx.xx.xxx. Expected: TestConnection.  
Captured: sword sudo: pam_authenticate: Module is unknown SAILPOINT> Password  
Sh: Password: command not found. Command exit code: sword sudo:  
pam_authenticate: Module is unknown SAILPOINT> Password sh: Password: command not  
found
```

**Resolution:** When the test connection fails on SUSE computer, the following setting must be changed in /etc/ssh/sshd\_config file:

```
PasswordAuthentication yes
```

Enter the following command to restart the sshd after updating the sshd\_config file:

```
/etc/init.d/sshd restart
```

### 2 - Password command failed with an error message

Password command fails if password prompts are not matching.

**Resolution:** Verify the password command on Linux computer for password prompts and if the required prompts are present in your application.

For example, passwd Person2

Changing password for Person2.

**New Password:** New Password is the prompt, so if this prompt is not present in your application, add/update it as follows:

For example,

```
<entry key="PasswdPrompts">  
  <value>  
    <Map>  
      <entry key="0">  
        <value>  
          <Map>
```

## 3 - Aggregation/test connection fails with timeout error

Aggregation/test connection fails with the following timeout error:

Exception during aggregation. Reason: sailpoint.connector.ConnectorException:  
Account aggregation failed. Timeout occurred.

**Resolution:** Change the value of the **SSHLoginTimeout (in millisecond)** application attribute as per your requirement in the debug page of the application:

```
<entry key="SSHLoginTimeout" value="1000" />
```

#### **4 - After target aggregation resources are not getting correlated with Account Groups**

After target aggregation the resources are not getting correlated with Account Groups.

**Resolution:** Ensure that your correlation rule populates "Correlator.RULE\_RETURN\_GROUP\_ATTRIBUTE" as follows:

```
....  
if ( isGroup ) {  
    returnMap.put(Correlator.RULE_RETURN_GROUP_ATTRIBUTE, "nativeIdentity");  
    returnMap.put(Correlator.RULE_RETURN_GROUP_ATTRIBUTE_VALUE, nativeId);  
}  
....
```

### 5 - Test connection fails for key based authentication with an error

Test connection fails for key based authentication with the following error.

Login failed. Error while connecting to host:<hostname>. Cannot read key file.

**Resolution:** Perform the following steps to generate/convert private/public keys in format which is supported by UNIX direct connectors.

- Generate keys using openssl. This method can be used for any version of SSH.
  - a. Create private key using the following command:

```
openssl <gendsa/genrsa> -des3 -out <private_key> 1024
```
  - b. Change the permission on the <private\_key> file as follows:

```
chmod 0600 <private_key>
```
  - c. Create public key from private\_key

```
ssh-keygen -y -f <private_key> > <public_key>
```
  - d. Use the <private\_key> and <public\_key> files for authentication.
- Generate keys using ssh-keygen. (OpenSSH 5.8 or above)
  - a. Create private and public key using the following command

```
ssh-keygen -t <dsa/rsa> -b 1024
```

By default files with name `id_dsa`/`id_rsa` and `id_dsa.pub`/`id_rsa.pub` will be created.
  - b. Convert <private key> to have DES-EDE3-CBC encryption algorithm by using the following command:

```
openssl <dsa/rsa> -in <private_key> -out <new_private_key> -des3
```
  - c. Change the permission on the <new\_private\_key> file as follows:

```
chmod 0600 <new_private_key>
```
  - d. Create public key file using the new private key as follows:

```
ssh-keygen -y -f <new_private_key> > <new_public_key>
```
  - e. Use the <new\_private\_key> and <new\_public\_key> files for authentication.

### 6 - Test connection fails with one of the following error when sudo user is configured for public key authentication

- Test connection fails with the following error when sudo user is configured for public key authentication:

Test SSH communication failed over host: xxxxxxxx. Error while executing command: sudo -p %SAILPOINTSUDO echo TestConnection over host: xxxxxxxx. Invalid sudo user password.

**Resolution:** On managed system,

- if Sudoers file is having Sudo user with **PASSWD** attribute assigned, then the sudo user's password specified in application configuration, password must be correct for certificate based authentication.
- if Sudoers file is having Sudo user with **NOPASSWD** attribute assigned, then the sudo user's password specified in application configuration, password can be incorrect/or any value. Certificate based authentication must still work.

**Note:** Password is mandatory field on application UI.

- Login failed. Failed to authenticate the ssh credentials for user: shraddha to host: xxx.xx.xx.xxx

**Resolution:** Verify `pam_tally2` counter and reset it to 0 (zero) and perform the operations again.

## 7 - Enable user failed with an error

Enable user failed with the following error:

```
Failed to enable account for user: user1. Error code: 254. Error: Unlocking password
for user user1.passwd: Unsafe operation (use -f to force)
```

**Resolution:** Update the following entry in connector registry/debug application configuration as:

```
<entry key="enable.account" value="passwd -u -f" />
```

## 8 - Test connection fails for SUSE Linux

Test connection fails on SUSE Linux as Password Authentication was not enabled.

**Resolution:** Perform the following steps to enable Password Authentication:

1. Change the value of Password Authentication from No to Yes in /etc/ssh/sshd\_config file as follows:  

```
PasswordAuthentication no
To
PasswordAuthentication yes
```
2. Restart the server using the following command:  

```
/etc/init.d/sshd restart
```

## 9 - Account Aggregation and Account provisioning displays an error for Lock/Unlock status

**Resolution:** Perform the following:

### For RHEL 6.x and above

1. Specify the maximum allowed failed login attempts before the account is locked by the system. Edit the configuration file pointed by registry key:  

```
<entry key="get.loginsyslimit" value="cat /etc/pam.d/system-auth"/>
```

**Default value:** /etc/pam.d/system-auth or /etc/pam.d/password-auth  
 Specify maximum allowed failed login using "deny=".  
 For example, add the following lines in /etc/pam.d/system-auth or /etc/pam.d/password-auth:  

```
auth required pam_tally2.so onerr=fail deny=5
account required pam_tally2.so
```
2. Ensure that pam\_tally2 command, as required in the following registry key works correctly:  

```
<entry key="aggregation.lockstatus" value="pam_tally2 | awk '{print $1} {print $2}' '/>
```
3. Ensure that the following command to get failed login works on the system:  

```
<entry key="get.userfailedlogin" value="pam_tally2" />
```
4. Verify if unlock command specified in the registry correctly resets the failed login counter:  

**Default settings:** <entry key="unlock.account" value="pam\_tally2 -u" />

**Note:** For RHEL version's below 6.0 where pam\_tally2 module not installed, replace pam\_tally2 with faillog in above commands.

## **Troubleshooting**

# Chapter 15: SailPoint Mainframe Connector

---

The following topics are discussed in this chapter:

Overview .....	153
Configuration parameters .....	153
Schema attributes .....	154

## Overview

---

The SailPoint Mainframe Connector is a *read only* connector which uses a technique called screen scraping and each deployment must write Rules to drive the login/logout/fetch accounts. The connector parses the screens and emulates the user during the interaction. On some legacy systems screen scraping is the only way to get to the data needed by IdentityIQ. Each Mainframe connector requires a lot of hands on configuration, because the Rules that drive this connector are very specific to the application on which the connector is running.

The Mainframe connector is designed for TN3270 applications and built on the IBM Host Access API libraries. You must have the IBM Host Access API libraries before working with this connector. You can purchase these libraries from IBM.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Mainframe connector uses the following connection attributes:

**Table 1—Mainframe Connector - Configuration parameters**

Parameters	Descriptions
host	The host of the server to which you are connecting.
port	The port the server is listening through.
user	The valid user name with which to connect to the server.
password	The password associated with the connection user.
logonRule	The rule used to log on to the application
logoffRule	The rule called to log off of the application
userIterateRegularExpression	The regular expression that should be used when fetching/iterating accounts. This expression breaks the screens into records that can be manipulated by the script.
userTransformRule	The rule called for each record delineated by the regular expression. This rule takes the text from the screens and converts it to a ResourceObject.

## Schema attributes

**Table 1—Mainframe Connector - Configuration parameters**

Parameters	Descriptions
userIterateCommand	The command used to natively iterate over all users
defaultTimeout	The length of time scripts should wait for data to be returned during command execution.
defaultIdleTimeout	The length of time the screen should be idle before timing out.
morePrompt	The prompt scripts should expect to receive to indicate there is more data on the screen
readyPrompt	The prompt scripts should expect to receive to indicate the mainframe is ready

## Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports two types of objects, account and group.

## Account attributes

---

Account objects are used when building identities Link objects.

**Table 2—Mainframe Connector - Account Attributes**

Name	Description
USER	The user ID or login ID of the user.
NAME	The user's name.
DEFAULT-GROUP	The default group to which the owner of the attribute belongs.
OWNER	The owner of the profile, or object.
SECURITY-LABEL	The security label assigned to the data being collected as defined by the Open Systems Interconnection Security Architecture.
ATTRIBUTES	The attributes assigned to the user.
GROUP	Group ID for the owner group.

# Chapter 16: SailPoint Microsoft SQL Server

---

The following topics are discussed in this chapter:

Overview .....	155
Supported features .....	156
Supported Managed Systems .....	156
Pre-requisites .....	156
Administrator permissions .....	156
Configuration parameters .....	159
Schema attributes .....	159
Account attributes .....	159
Group attributes .....	160
Provisioning Policy attributes .....	161
Additional information .....	161
Delete login .....	161
Direct permission .....	162
Identity and Entitlement representation .....	162
Troubleshooting .....	163

## Overview

---

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database, it is a software product whose primary function is to store and retrieve data as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the Internet).

SailPoint Microsoft SQL Server Connector manages the following entities on Microsoft SQL Server:

- User
  - Login User
  - Database User
- Role
  - Application Role
  - Database Role

## Supported features

---

SailPoint Microsoft SQL Server Connector supports the following features:

- Account Management
  - Manage Microsoft SQL Server Login Users and Database Users as Accounts
  - Aggregate, Refresh Accounts
  - Create, Update, Delete (Server Roles can be granted to SQL login, Database role can be granted to Database user. Application role grant is not supported)
  - Enable, Disable, Change Password
  - Add/Remove Entitlements
- Account - Group Management
  - Manage Microsoft SQL Server Application and Database Roles as Account-Groups
  - Aggregate, Refresh Groups
  - Create, Delete
- Permissions Management
  - Permissions directly assigned to accounts and groups as direct permissions during accounts and groups aggregations respectively.
  - Automated revocation of the aggregated permissions for accounts and groups.

### *References*

- “Direct permission” on page 162

## Supported Managed Systems

---

Following versions of Microsoft SQL Server is supported by the SailPoint Microsoft SQL Server Connector:

- Microsoft SQL Server 2016
- Microsoft SQL Server 2014
- Microsoft SQL Server 2012

## Pre-requisites

---

The compatible JDBC drivers must be used in the classpath of IdentityIQ for connecting to Microsoft SQL Server. For example, `sqljdbc4.jar`.

## Administrator permissions

---

Login using administrator credentials and create a new user on managed system using the following command:

```
CREATE LOGIN <USER> WITH PASSWORD = '<PASSWORD>'
```

Following are the minimum permissions required for Microsoft SQL Server user based on the operations:

Operation	Permissions
Test Connection	<pre>grant Connect SQL to [user]</pre> <p><b>Note:</b> In order to access databases, service account must have user mapping on the databases with public role defined.</p>
Aggregation	<pre>use [master] GO grant Connect SQL to [user] grant view any database to [user] grant view any definition to [user]</pre> <ul style="list-style-type: none"> <li>• <b>For Microsoft SQL Server version 2014 and above:</b> <pre>grant connect any database to [user]</pre> </li> <li>• <b>For versions below Microsoft SQL Server 2014:</b> <pre>use [database name] GO exec sp_addrolemember 'db_datareader', '[User]' GO</pre> </li> </ul>
Enable/Disable User	<pre>grant alter any login to [user]</pre>
Change Password	<pre>grant alter any login to [user]</pre>

## Overview

Operation	Permissions	
	Account	Role
Create	<p>grant alter any login to [user]</p> <p>To perform any operation on a database, the service account must have database user on the specific database.</p> <pre>use [database name] create user [username] for login [server login name] exec sp_addrolemember db_owner, [username]</pre> <p><b>Note:</b> User must have proper server role assignments to assign the same role to another user.</p> <p>For example, if administrator has granted Role1, Role2, Role3 roles to user A then, user A can grant only Role1, Role2, Role3 to any other user. User A cannot assign other roles apart from the roles assigned to it.</p> <p>Above mentioned permissions are required for adding and removing entitlements.</p>	use [database name] create user [username] for login [server login name] grant alter any application role to [username] grant alter any role to [username]
Delete	<p>grant alter any login to [user]</p>	use [database name] create user [username] for login [server login name] grant alter any application role to [username] grant alter any role to [username]

# Configuration parameters

---

The following table lists the configuration parameters of SailPoint Microsoft SQL Server Connector:

Parameters	Description
URL*	A valid URL of Microsoft SQL Server with the following format:  <i>jdbc:sqlserver:// [serverName [ \instanceName ] [ :portNumber ] ]</i> <ul style="list-style-type: none"> <li>• <b>jdbc:sqlserver://</b>: (<i>Required</i>) is known as the sub-protocol and is constant</li> <li>• <b>serverName</b>: is the address of the server to connect to. This could be a DNS, IP address, localhost, or 127.0.0.1 for the local computer.</li> <li>• <b>instanceName</b>: is the instance to connect to <i>serverName</i>.</li> <li>• <b>portNumber</b>: is the port to connect to <i>serverName</i>. The default is 1433.</li> </ul>
User*	Administrative Account to connect to Microsoft SQL Server.
Password*	Administrative Account password.
Driver*	The name of the Driver class supported by JDBC com.microsoft.sqlserver.jdbc.SQLServerDriver
Included Databases	List of comma separated databases names to be included in the aggregation operation.
Excluded Databases	List of comma separated databases name to be excluded in the aggregation operation.  <b>Note:</b> If the Include Database parameter is populated, the exclude database parameter would be ignored.

**Note:** All the parameters in the above table marked with the \* sign are mandatory parameters.

# Schema attributes

---

This section describes the different schema attributes.

## Account attributes

---

The following table lists the account attributes:

Attribute name	Description
native_identity	Native identity represented by default as <i>loginName@serverName</i> or <i>loginName@databaseName</i> .
database_name	Database name of the login.
database_id	Database Id.
server_login	Server login associated to account.

## Schema attributes

Attribute name	Description
name	Account name.
principal_id	ID of database principal.
type	Type of the login.
type_desc	Description type of the login.
default_schema_name	Name to be used when SQL name does not specify schema.
create_date	Creation date of the login.
modify_date	Last modification date of the login.
sid	SID of the login.
server_name	Server name.
is_fixed_role	If the value is 1, then this row represents an entry for one of the fixed database roles.
owning_principal_id	ID of the principal that owns this database principal.
roles	Roles assigned to the login.
DBUser	Database users which are associated to the login.

**Note:** It is recommended that, on Managed System the server name and database name must not be the same.

## Group attributes

---

The following table lists the group attributes:

Attribute name	Description
native_identity	Native identity represented by default as <i>groupName@serverName</i> or <i>groupName@databaseName</i> .
name	Group name.
database_name	Database name in which group exists.
database_id	Database ID in which group exists.
principal_id	ID of database principal.
roles	Roles assigned to the group.
server_name	Server name of the group.
type_desc	Description type of the group.

## Provisioning Policy attributes

---

The following table lists the provisioning policy attributes for Create Account and Create Group:

Attribute name	Description
<b>Create Account</b>	
Login name*	Server login name.
password	Password of server login name.
Account Type*	Type of the server login name. <b>Note:</b> Either one of the attributes (Windows Login or SQL Login) is <b>Yes</b> .
User Mapping	User mapping format (Username@databasename).
<b>Create Group</b>	
Group name*	Group name.
Group Type*	Type of the Group. <b>Note:</b> Either one of the attributes (Application Role or Database Role) is <b>Yes</b> .
Password	Password for Application type role.

## Additional information

---

This section describes the additional information related to the Microsoft SQL Server Connector.

### Delete login

---

To delete the login users, set the value of the **DeleteLoginBYdefault** configuration parameter to **Y** by adding the following entry key:

```
<entry key="DeleteLoginBYdefault" value="Y" />
```

## Direct permission

---

Following targets are supported:

- DATABASE
- SERVER
- PROCEDURE
- ASYMMETRIC\_KEY
- SYMMETRIC\_KEYS
- USER
- CERTIFICATE\_MAPPED\_USER
- DATABASE\_ROLE
- APPLICATION\_ROLE

For example, GRANT CONNECT on DATABASE\_DatabaseName

GRANT CONNECT on SERVER\_ServerName

{ [GRANT/REVOKE] [SELECT/ INSERT/ DELETE/UPDATE] ON Table TO Account}

**Note:** **DATABASE\_** is appended before DatabaseName. Similar appending is done for other objects such as **SERVER\_** for ServerName, **PROCEDURE\_** for Procedures.

## Identity and Entitlement representation

---

This section describes the Identity and Entitlement representation for SailPoint Microsoft SQL Server Connector.

### Identity representation

**Account:** The Account in Microsoft SQL Server Connector is represented as follows:

<Account name>@<Container name>

- For Database login it is represented as <Account name>@<Database name>  
For example, **username@master**
- For Server Login it is represented as <Account name>@<Server name>  
For example, **loginname@MSSERVER**

### Entitlement representation

**Groups:** The Groups in Microsoft SQL Server Connector are represented as follows:

<Group name>@<Container name>

- For Database Role and Application Role it is represented as <Group name>@<Database name>  
For example, **userDatabaseRole@master**
- For Server roles it is represented as <Group name>@<Server name>  
For example, **userGroup@MSSERVER**

# Troubleshooting

---

## 1 - Aggregation fails

When a login user is created in Microsoft SQL Server and is granted permission only on some of the Databases present on the server and if aggregation task is run for that application, Aggregation fails as the user is not able to access other databases.

**Resolution:** In application Configuration page under the “Include Databases” section, provide the complete list of databases (comma separated list) for which the login user have accesses.

This completes the aggregation successfully, and only details of the users present in the list of included database will be fetched.

## **Troubleshooting**

# Chapter 17: SailPoint Oracle Connector

---

The following topics are discussed in this chapter:

Overview .....	165
Supported features .....	165
Supported Managed Systems .....	166
Pre-requisites .....	166
Administrator permissions .....	166
Configuration parameters .....	167
Additional configuration parameter .....	168
Schema attributes .....	168
Account attributes .....	168
Group attributes .....	169
Provisioning policy attributes .....	169
Troubleshooting .....	170

## Overview

---

The Oracle Database (commonly referred to as Oracle RDBMS or simply as Oracle) is an object-relational database management system (ORDBMS).

SailPoint Oracle Server Connector is a connector to Oracle database server that allows full user administration with provisioning and password management capabilities of Oracle server. Oracle Server Connector manages the following entities of Oracle server:

- User
- Role

## Supported features

---

SailPoint Oracle Connector supports the following features:

- Account Management
  - Manages Oracle users
  - Aggregation, Refresh Accounts, Pass Through Authentication, Discover Schema
  - Create, Update, Delete
  - Enable, Disable, Change Password
  - Add/Remove Entitlements

**Note:** The Oracle Connector respects the case sensitivity of the oracle user name. Users having mixed case character must enclose the name in double quotes for login into the system.

## Overview

- Account - Group Management
  - Manages Oracle groups as ROLE
  - Aggregation, Refresh Groups
  - Create, Update, Delete
- Permission Management
  - Permissions directly assigned to accounts and groups as direct permissions during account and group aggregation.
  - Automated revocation of the aggregated permissions.

## Supported Managed Systems

---

SailPoint Oracle Connector supports the following version of Oracle Server:

- Oracle Database 12c

**Note:** The connector manages users and groups of an Oracle 12c pluggable database (PDB). The users and groups of an Oracle 12c container database (CDB) are not supported.

## Pre-requisites

---

The compatible JDBC drivers must be used in the classpath of IdentityIQ for connecting to Oracle Server. For example, ojdbc6.jar.

## Administrator permissions

---

The Oracle administrator must have all the permissions mentioned below for performing the provisioning operations.

Login with administrator credentials and execute the following command to create a new user:

```
CREATE USER ${UserName} IDENTIFIED BY ${Password};
```

Operation	SELECT permission
Test Connection	GRANT create session TO \${UserName};
Aggregation	
USER	GRANT SELECT ON dba_users TO \${UserName}; <b>(By this command discovering schema is possible)</b>
	GRANT SELECT ON dba_sys_privs TO \${UserName};
GROUP	GRANT SELECT ON dba_roles TO \${UserName}; <b>(By this command discovering schema is possible)</b>
	GRANT SELECT ON dba_role_privs TO \${UserName};

Operation	SELECT permission					
PERMISSIONS	GRANT SELECT ON dba_tab_privs TO \${UserName};					
	GRANT SELECT ON dba_col_privs TO \${UserName};					
	GRANT SELECT ON dba_sys_privs TO \${UserName};					
	GRANT SELECT ON system_privilege_map TO \${UserName};					
	GRANT SELECT ON V_\$version TO \${UserName};					
	GRANT SELECT on V_\$PWFILE_USERS to \${UserName};					
<b>Note: To view Sysdba privileges:</b> SELECT * FROM V\$PWFILE_USERS						
Provisioning						
Entity/Operation	Account	Role	Profile			
UPDATE/MODIFY	GRANT ALTER USER TO \${UserName};	GRANT ALTER ANY ROLE TO \${UserName};	GRANT ALTER PROFILE TO \${UserName};			
DELETE/DROP	GRANT DROP USER TO \${UserName};	GRANT DROP ANY ROLE TO \${UserName};				
CREATE	<ul style="list-style-type: none"> <li>• GRANT CONNECT TO \${UserName};</li> <li>• GRANT CREATE USER TO \${UserName};</li> <li>• GRANT GRANT ANY ROLE TO \${UserName};</li> <li>• GRANT GRANT ANY PRIVILEGE TO \${UserName};</li> </ul>	GRANT CREATE ROLE TO \${UserName};	GRANT CREATE PROFILE TO \${UserName};			

## Configuration parameters

---

The following table lists the configuration parameters of SailPoint Oracle Connector:

Parameters	Description
Administrative Username*	Name of the administrative account which has all the privileges to perform the CRUD (Create, Read, Update, and Delete) operations. The default administrator of Oracle Server is <b>system</b> .
Password*	The password of Administrative account.
Driver class*	Name of the type4 driver to use when making connection with oracle server. By default this connector uses <code>oracle.jdbc.driver.OracleDriver</code>

## Schema attributes

Parameters	Description
Url*	<p>The url to connect to the database. The format is <code>jdbc:oracle:thin:@&lt;HOST&gt;:&lt;PORT&gt;:&lt;SID&gt;</code></p> <p>For example <code>jdbc:oracle:thin:@xxx.xx.xx.xx:1521:ORCL</code> url consist of</p> <ul style="list-style-type: none"><li>• <b>jdbc:oracle:thin:@</b>: This is common part which states that the connection is made using thin driver.</li><li>• <b>xxx.xx.xx.xx</b>: server Name or IP of the oracle server</li><li>• <b>1521</b>: The port number of the oracle server. This port number should be known by the oracle server administrator.</li><li>• <b>ORCL</b>: The SID of the oracle server.</li></ul> <p><b>Note:</b> To connect to PDB of Oracle Database 12c, use the service name instead of SID in the URL as follows:</p> <pre>jdbc:oracle:thin:@&lt;HOST&gt;:&lt;PORT&gt;/&lt;SERVICE_NAME&gt;</pre> <p>For example, <code>jdbc:oracle:thin:@xxx.xx.xx.xxx:1522/orcl.16.23.200</code></p>

**Note:** All the parameters marked with the \* sign in the above table are the mandatory parameters.

## Additional configuration parameter

Parameter	Description
deleteUserOnCascade	An Oracle database user can have associated objects. Specify <code>deleteUserOnCascade</code> with the value set to <code>true</code> as a configuration attribute to drop all objects in the user's schema before dropping the user: <pre>&lt;entry key="deleteUserOnCascade"&gt;   &lt;value&gt;     &lt;Boolean&gt;true&lt;/Boolean&gt;   &lt;/value&gt; &lt;/entry&gt;</pre>

## Schema attributes

This section describes the different schema attributes.

## Account attributes

The following table lists the account attributes:

Attribute name	Description
USERNAME	User name.
USER_ID	User ID.
ACCOUNT_STATUS	Account status.
DEFAULT_TABLESPACE	Default tablespace.
ROLES	Roles assigned to the user.
PROFILES	Profiles assigned to the user.
TEMP_TABLESPACE	Temporary tablespace.
SYSTEM_PRIVILEGES	System privileges assigned to the user.  <b>Note:</b> SYSDBA and SYSOPER permissions are visible under SYSTEM_PRIVILEGES.
AUTHENTICATION_TYPE	Authentication type.

## Group attributes

---

The following table lists the group attributes:

Attribute name	Description
ROLE	Role name.
AUTHENTICATION_TYPE	Authentication type.
SYSTEM_PRIVILEGES	System privileges assigned to the role.
ROLES	Roles assigned to the role.
PASSWORD_REQUIRED	Password is required or not.

## Provisioning policy attributes

---

The following table lists the provisioning policy attributes for Create Account and Create Group:

Attribute name	Description
<b>Create Account</b>	
Username*	Username
password*	Password of the user.
DEFAULT_TABLESPACE	Default tablespace.
TEMP_TABLESPACE	Temporary tablespace.
PROFILE	Profile
AUTHENTICATION_METHOD	Authentication method.

## Troubleshooting

Attribute name	Description
<b>Create Group</b>	
Role*	Role name.
Granted Role	Role to assign for new role.
System Privileges	System privileges to assign for new role.

**Note:** All the parameters marked with the \* sign in the above table are the mandatory parameters.

## Troubleshooting

---

### 1 - Provision of SYSDBA and SYSOPER fails

When provisioning SYSDBA and SYSOPER, the following error message is displayed:

"User entitlements not modified sailpoint.connector.ConnectorException: ORA-01031: insufficient privileges"

**Resolution:** If above error message is displayed, the user must connect with the user as "<username> as sysdba" and the <username> must have Sysdba privileges.

# Chapter 18: SailPoint PeopleSoft Connector

---

The following topics are discussed in this chapter:

Overview .....	171
Supported features .....	171
Supported Managed Systems .....	172
Pre-requisites .....	172
Administrator permission .....	172
Configuration parameters .....	172
Schema attributes .....	173
Account attributes .....	173
Group attributes .....	175
Additional information .....	175
Creating the Component Interfaces .....	175
Creating the Component interface jar file .....	175
Configuring the Component Interface Security .....	176
Troubleshooting .....	177

## Overview

---

The SailPoint PeopleSoft Connector manages the administrative entities of PeopleSoft server (User Profiles and Roles). The PeopleSoft Connector communicates to the PeopleSoft server through component interfaces.

### Supported features

---

SailPoint PeopleSoft Connector supports the following features:

- Account Management
  - Manages PeopleSoft users as Accounts
  - Aggregation, Partitioning Aggregation, Refresh Accounts, Discover Schema
  - Create, Update, Delete
  - Enable, Disable, Change Password
  - Add/Remove Entitlements
- Account - Group Management
  - Manages PeopleSoft roles as Account-Groups
  - Aggregation, Refresh Groups

### References

- Appendix C: Partitioning Aggregation

## Supported Managed Systems

---

SailPoint PeopleSoft Connector supports the following managed systems:

- PeopleTools version 8.55, 8.54, 8.53
- PeopleSoft Server version 9.2, 9.1

## Pre-requisites

---

To use the PeopleSoft Connector, you must first configure the component interfaces on PeopleSoft. This requires the following steps:

1. Creating the Component Interfaces
2. Creating the Component interface jar file
3. Configuring the Component Interface Security

The following files must be present on the computer where the Connector is installed:

- `psjobj.jar` (found on PeopleSoft server at `%PS_HOME%\classes` where `%PS_HOME%` is the location where PeopleSoft is installed)
- `iiqPeopleSoftCompInt.jar` (See Creating the Component interface jar file)

## Administrator permission

---

The PeopleSoft user who must act as an administrator for proper functioning of the connector must have access to the related Component Interfaces. For more information, see Configuring the Component Interface Security.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The PeopleSoft connector uses the following connection attributes:

Attribute	Description
Host*	The hostname of the PeopleSoft server.
Port*	The Jolt port (Jolt Server Listener Port) on which the PeopleSoft server is listening. Default: 9000
User*	The user name used to login to PeopleSoft.
Password*	The password to use to login to PeopleSoft.
User Component Interface*	The name of the PeopleSoft component interface to use to read PeopleSoft User Profile.

Attribute	Description
Group Component Interface*	The name of the PeopleSoft component interface to use to read PeopleSoft Roles.  For more information, see “Creating component interface for Peoplesoft financials” on page 551.
Jar location	If there are more than one PeopleSoft application of different PeopleTools versions running under the same instance of JVM, the location specified would be added in the classpath. (The <code>psjoa.jar</code> and <code>iiqPeopleSoftCompInt.jar</code> files). For more information, see Creating the Component interface jar file).  <b>Note:</b> For single PeopleSoft application, the PeopleSoft jars can be located in <code>WEB-INF\lib</code> directory.
Partition Enabled	Check box to determine if partition aggregation is required.
Partition Statements	Criteria to specify the range of users to be downloaded. For example, if the range is specified as A-M, then this specifies that all the Users whose User ID's are between <b>A</b> and <b>M</b> (including A and M) would be treated as one partition and downloaded.  To specify more than one partition the entries should be separated using a newline character. For more information, see Appendix C: Partitioning Aggregation
Domain Connection Password Enabled	Determines if Domain connection Password is configured.
Domain Connection Password*	Password is required if <b>Domain Connection Password Enabled</b> attribute is selected.

**Note:** All the parameters marked with the \* sign in the above table are the mandatory parameters.

**Note:** While deleting a User, add Component Interface in debug as `deleteComponentInterface`.  
**For example,** `<entry key="deleteComponentInterface" value="IIQ_DEL_USER"/>`

## Schema attributes

---

This section describes the different schema attributes.

### Account attributes

---

The following table lists the account attributes:

Attributes	Description
UserID	The PeopleSoft User ID.
AccountLocked	Status of Account if it is locked or not.
AlternateUserID	User ID Alias.

## Schema attributes

Attributes	Description
CurrencyCode	Currency code of the user.
DefaultMobilePage	Default mobile page.
EffectiveDateFrom	Workflow attribute - from date.
EffectiveDateTo	Workflow attribute - to date.
EmailAddresses	Email address of the user.
EmailUser	Routing preferences - email user. It is a multivalued attribute.
ExpertEntry	Enable expert entry.
FailedLogins	Number of failed logins.
IDTypes	User ID types and values.
LanguageCode	Language code.
LastUpdateDateTime	Last update date/time.
LastUpdateUserID	Last update user ID.
MultiLanguageEnabled	Multi-language enabled.
NavigatorHomePermissionList	Default navigator home page permission list.
Opertype	Use external authentication.
PasswordExpired	Is password expired.
PrimaryEmailAddress	Primary email address.
PrimaryPermissionList	Primary permission list.
ProcessProfilePermissionList	Process profile permission list.
roleNames	Roles assigned to the user are treated as entitlements.
RowSecurityPermissionList	Row security permission list.
SymbolicID	Used to map the User Id to Access ID.
UserDescription	Description of the user.
Roles	Roles consists of permissions and can be assigned to user profile.
Encrypted	Encrypted
ReassignWork	Reassign work to alternate user.
ReassignUserID	Reassigned user's UserID.
RowSecurityPermissionList	Row Security Permissions.
SupervisingUserID	Supervisor's User Id.
UserIDAlias	Alias of the user.
WorkListEntriesCount	Count of worklist entries.
WorklistUser	Displays user workflow.

## Group attributes

---

The following table lists the group attributes:

Attributes	Description
ROLENAME	Name of the role.
ROLETYP	Type of the role.
RolePermissionLists	Permission List for the role.
DESCR	Description of the role.
DESCRLONG	Long description.
ALLOWNOTIFY	Workflow routing - allow notifications.
ALLOWLOOKUP	Workflow routing - allow recipient lookup.
LASTUPDDTTM	Last update date/time.
LASTUPDOPERID	Last update user ID.
Roles that can be granted	Roles that can be granted by this role.
Roles that can grant	Roles that can grant this role.

## Additional information

---

This section describes the additional information related to the PeopleSoft Connector.

### Creating the Component Interfaces

---

For creating the component interfaces, see Appendix B: Component Interface.

### Creating the Component interface jar file

---

The `iiqPeopleSoftCompInt.jar` file contains the PeopleSoft Component Interface java classes. It must be generated from the respective PeopleSoft resource and then copied into the IdentityIQ classpath.

Perform the following steps to create the `iiqPeopleSoftCompInt.jar` file from the Component interface java files.

1. Logon to PeopleSoft Application Designer in two tier mode.
2. Open the Component Interface project and open all the component interfaces by double clicking each component interface. For example, `IIQ_USERS`
3. From the menu select **Build ==> PeopleSoft APIs**.  
The **Build PeopleSoft API Bindings** window appears.
4. From the **Build PeopleSoft API Bindings** window, select the **Build** check box in the java Classes frame and clear the COM Type Library and C Header Files Build check boxes.

## Additional information

In the **Select APIs to Build** drop down menu, select the following options:

- ComplIntfc.ComplIntfcPropertyInfo
- ComplIntfc.ComplIntfcPropertyInfoCollection
- PeopleSoft.\* (all Component Interfaces that begin with the prefix PeopleSoft)
- ComplIntfc.IIQ\_\* (all Component Interfaces that begin with the prefix ComplIntfc.IIQ\_)

**Note:** If you need to generate Component Interface Java files for the entire group of Component Interfaces click ALL.

Create a directory to deploy the Java files. For example, if you specify C:\CI as the file path, then the Component Interface Java files are generated in C:\CI\PeopleSoft\Generated\CompIntfc.

6. Compile the JAVA files by performing the following steps:

- a. Open the command prompt and change directories to the folder where the generated JAVA files are located. For example, C:\CI.
- b. Navigate to the PeopleSoft\Generated\CompIntfc\ directory.
- c. Run the following command:

```
javac -classpath %PS_HOME%\class\psjoa.jar *.java
```

Where %PS\_HOME% is the location that PeopleSoft is installed.

Important: Ensure that the JAVA compiler used for compiling the generated JAVA files is compatible with the JAVA provided with the PeopleSoft installation that needs to be managed.

- d. (Optional) You can delete all the generated java files from the existing directory, however, do not delete the .class files.
7. Perform the following steps to package the compiled files as the iiqPeopleSoftCompInt.jar file:
  - a. Open the Command prompt and navigate to the newly created directory. For example, C:\CI
  - b. Run the command: jar -cvf iiqPeopleSoftCompInt.jar \*
8. Copy the generated iiqPeopleSoftCompInt.jar and %PS\_HOME%\class\psjoa.jar files to the computer where IdentityIQ is running.

## Configuring the Component Interface Security

---

Before using the connector, you must allow the PeopleSoft user, for whom the connector is configured, to access the generated component interfaces.

To set security for the PeopleTools project, perform the following:

1. Log into the PeopleSoft web interface.  
Default: http://<server-name>:<port-number>/psp/ps
2. Navigate to **PeopleTools ==> Security ==> Permissions & Roles ==> Permission Lists**.
3. Click **Add a New Value** to create a new permission list. Type **New\_Name** as the name of the permission list, then click **Add**.
4. Click the Component Interfaces tab and add the created component interface.  
For more information, see “Creating component interface for Peoplesoft financials” on page 551.

For example,

- IIQ\_DEL\_ROLE
  - IIQ\_DEL\_USER
  - IIQ\_ROLES
  - IIQ\_USERS
5. For each added component interface, click **Edit ==> Full Access (All)**, then click **OK**.
  6. Click **Save** to save the new permission list.
  7. Navigate to **PeopleTools ==> Security ==> Permissions & Roles ==> Roles**.
  8. Click **Add a New Value** to create a new role. Type **New\_Name** as the name and then click **Add**. For example, **IIQ\_ROLE**.
  9. Enter the description as **IdentityIQ Role**.
  10. Click the **Permission Lists** tab and add the permission list created in Step 3. Click **Save** to save the role.
  11. Navigate to **PeopleTools ==> Security ==> User Profiles**, and select the user (for whom the permissions must be provided) that is being used in the connector.
  12. Click the **Roles** tab and add the role created in Step 10. Click **Save** to add the role to the user.

## Troubleshooting

---

### 1 - When the supported platform version is Java 1.6 an error message appears

When the supported platform version is Java version 1.6, the following error message appears:

```
java.lang.UnsupportedClassVersionError: psft/pt8/joa/API : Unsupported major.minor
version 51.0 (unable to load class psft.pt8.joa.API)
```

**Resolution:** Ensure that the supported platform version is Java 1.7.

### 2 - (Only for PeopleTools version 8.54) Connection to Server not established

When testing the CI (Component Interface) Java APIs after upgrading to PeopleTools version 8.54, the connection to the application server fails with the following error message appears:

```
openconnector.ConnectorException: Connection to server not established
```

**Resolution:** Navigate to the location where PeopleSoft is installed. For example,  
C:\PS\_CFG\_HOME\webserv\peoplesoft\applications\peoplesoft\PORTAL.war\WEB-INF\classes.

Copy all the files from this directory into the WEB\_INF\classes directory of IdentityIQ.

Now you will be able to successfully connect to the server. This solution is documented in the following knowledge base article on the Oracle support site:

**E-CI: Java API Connection Fails With "java.lang.NoClassDefFoundError:  
com/peoplesoft/pt/management/runtime/pia/JoltSessionMXBean" Error(1947124.1)**

## **Troubleshooting**

# Chapter 19: SailPoint RACF Connector

---

The following topics are discussed in this chapter:

Overview .....	179
Supported features .....	179
Configuration parameters.....	179
Schema Attributes .....	181
Account attributes .....	181
Group attributes.....	185

## Overview

---

The SailPoint RACF Connector is a *read only* connector to read the file produced by the RACF unload utility.

**Note:** The RACF Full Connector supports the provisioning operations. For more information, see *SailPoint IdentityIQ Connector for RACF Administration Guide*.

## Supported features

---

SailPoint RACF Connector supports the following features:

- Account Management
  - Manages RACF users as Accounts
  - Aggregation, Discover Schema
- Account - Group Management
  - Manages RACF groups as Account-Groups
  - Aggregation
- Permission Management
  - Application reads permissions directly assigned to accounts and groups as direct permissions during account and group aggregation.
  - The connector does not support automated revocation of the aggregated permissions and creates work item for such requests

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

## Configuration parameters

The RACF connector uses the following configuration parameters:

**Table 1—RACF Connector - Configuration parameters**

Parameters	Description
filetransport	local, ftp, scp
host	The host of the server to which you are connecting.
transportUser	The user to use with ftp and scp. Not valid with local.
transportUserPassword	The password to use with of ftp and scp. Not valid with local.
file	The fully qualified path to the file.
fileEncoding	Specify the file encoding to be used by the connector. Valid values for this attribute can be found at: <a href="http://www.iana.org/assignments/character-sets">http://www.iana.org/assignments/character-sets</a>  If this field is empty, the default encoding (the value of <code>file.encoding</code> specified by the jvm) is used.
mapToResourceObjectRule	Rule that is called to override the transformation of the data from the <code>Map&lt;String, String&gt;</code> form into a <code>ResourceObject</code> .
filterString	Filter lines that match this string.
filterEmptyRecords	If activated, records that have no data are filtered.
preIterativeRule	The pre-iterate rule will check for a specially named Configuration object that will hold the last run statistics that can be compared against the current values.  This rule is called after the file has been transferred, but before iteration over the objects in the file is started.  For validation this rule can use the existing statistics stored by the <code>postIterationRule</code> during the last aggregation. The rule can compare the stored values with the new values to check for problems
postIterativeRule	The post-iterate rule can store away the configuration object and rename/delete the file if desired.  This rule is called after aggregation has completed and ALL objects have been iterated.
RACF Attribute Customization Rule	The rule used to extend the parsing capabilities to customer records or redefine existing record configurations. The RACF attribute customization rule creates a map of <code>LineRecord</code> objects that hold the record ID and other field definitions.

# Schema Attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports two types of objects, account and group.

## Account attributes

---

Account objects are used when building identities Link objects.

**Table 2—RACF Connector - Account Attributes**

Attribute	Description
CLASSES	
CATEGORIES	Defines the categories associated with a general resource. There is one record per general resource/category combination.
KERB_NAME	RACF user name as taken from the profile.
KERB_MAXLIFE	Maximum ticket life.
KERB_KEY_VER	Current key version.
KERB_ENCRYPT_DES	Is key encryption using DES enabled?
KERB_ENCRYPT_DES3	Is key encryption using DES3 enabled?
KERB_ENCRYPT_DESD	Is key encryption using DES with derivation enabled?
KERB_ENCRYPT_A128	Is key encryption using AES128 enabled?
KERB_ENCRYPT_A256	Is key encryption using AES256 enabled?
KERB_KEY_FROM	Key source. Valid values are PASSWORD or PHRASE.
NAME	User ID as taken from the profile name.
CREATE_DATE	The date that the profile was created.
OWNER_ID	The user ID or group name that owns the profile.
ADSP	Does the user have the ADSP attribute?
SPECIAL	Does the user have the SPECIAL attribute?
OPER	Does the user have the OPERATIONS attribute?
REVOKE	Is the user REVOKEd?
GRPACC	Does the user have the GRPACC attribute?
PWD_INTERVAL	The number of days that the user's password can be used.
PWD_DATE	The date that the password was last changed.
PROGRAMMER	The name associated with the user ID.
DEFGRP_ID	The default group associated with the user.
LASTJOB_TIME	The time that the user last entered the system.
LASTJOB_DATE	The date that the user last entered the system.

## Schema Attributes

**Table 2—RACF Connector - Account Attributes (Continued)**

Attribute	Description
INSTALL_DATA	Installation-defined data.
UAUDIT	Do all RACHECK and RACDEF SVCS cause logging?
AUDITOR	Specifies if the user has the auditor attribute.
NOPWD	YES - indicates that this user ID can logon without a password using OID card. NO - indicates that this user must specify a password. PRO - indicates a protected user ID. PHR - indicates that the user has a password phrase.
OIDCARD	Specifies if this user has the OIDCARD data.
PWD_GEN	The current password generation number.
REVOKE_CNT	The number of unsuccessful logon attempts.
MODEL	The data set model profile name.
SECLEVEL	The user's security level.
REVOKE_DATE	The date that the user will be revoked.
RESUME_DATE	The date that the user will be resumed.
ACCESS_SUN	Can the user access the system on Sunday?
ACCESS_MON	Can the user access the system on Monday?
ACCESS_TUE	Can the user access the system on Tuesday?
ACCESS_WED	Can the user access the system on Wednesday?
ACCESS_THU	Can the user access the system on Thursday?
ACCESS_FRI	Can the user access the system on Friday?
ACCESS_SAT	Can the user access the system on Saturday?
START_TIME	After what time can the user logon?
END_TIME	After what time can the user not logon?
SEC_LABELS	The user's default security label.
ATTRIBS	Other user attributes (RSTD for users with RESTRICTED attribute).
PWDENV_EXISTS	Has a PKCS#7 envelope been created for the user's current password?
PWD_ASIS	Should the password be evaluated in the case entered?
PHR_DATE	The date the password phrase was last changed.
PHR_GEN	The current password phrase generation number.
CERT_SEQN	Sequence number that is incremented whenever a certificate for the user is added, deleted, or altered.
PPHENV_EXISTS	Has the user's current password phrase been PKCS#7 enveloped for possible retrieval?

**Table 2—RACF Connector - Account Attributes (Continued)**

Attribute	Description
ASSOCIATED_MAPPING	Defines the certificate name filter in the DIGTNMAP class associated with this user ID.
CSDATA_CUSTOM	Record type of the User CICS Data record
LNOTES_SHORTNAME	User ID as taken from the profile name.
CICS_OP_CLASSES	The class associated with the CICS operator.
GROUPS	
OVM_UID	User identifier (UID) associated with the user name from the profile.
OVM_HOME_PATH	Home path associated with the user identifier (UID).
OVM_PROGRAM	Default program associated with the user identifier (UID).
OVM_FSROOT	File system root for this user.
PRIMARY_LANGUAGE	The primary language for the user.
SECONDARY_LANGUAGE	The secondary language for the user.
CICS_RSL_KEY	Defines the resource security level (RSL) keys associated with a CICS user. There is one record per combination of user and CICS RSL key.
LDAP_HOST	LDAP server URL.
LDAP_BIND_DN	LDAP BIND distinguished name.
NETVIEW_IC	Command list processed at logon.
NETVIEW_CONSOLE_NAME	Default console name.
NETVIEW_CTL	CTL value: GENERAL, GLOBAL, or SPECIFIC.
NETVIEW_MSGRECVR	Eligible to receive unsolicited messages?
NETVIEW_NGMFADMN	Authorized to NetView graphic monitoring facility?
NETVIEW_NGMFVSPN	Value of view span options.
NDS_UNAME	NDS user name associated with the user ID.
CICS_OPIDENT	The CICS operator identifier.
CICS_OPPRTY	The CICS operator priority.
CICS_NOFORCE	Is the extended recovery facility (XRF) NOFORCE option in effect?
CICS_TIMEOUT	The terminal time-out value. Expressed in hh:mm.
DCE_UUID	DCE UUID associated with the user name from the profile.
DCE_NAME	DCE principal name associated with this user.
DCE_HOMECELL	Home cell name.
DCE_HOMEUUID	Home cell UUID.
DCE_AUTOLOGIN	Is this user eligible for an automatic DCE login?

## Schema Attributes

**Table 2—RACF Connector - Account Attributes (Continued)**

Attribute	Description
CERTIFICATE	Defines the names of the certificate profiles in the DIGTCERT class that are associated with this user ID.
CICS_TSL_KEY	Defines the transaction security level (TSL) keys for a CICS user. There is one record per combination of user and CICS TSL key.
TSO_ACCOUNT_NAME	User ID as taken from the profile name.
TSO_COMMAND	The command issued at LOGON.
TSO_DEST	The default destination identifier.
TSO_HOLD_CLASS	The default hold class.
TSO_JOB_CLASS	The default job class.
TSO_LOGIN_PROC	The default logon procedure.
TSO_LOGIN_SIZE	The default logon region size.
TSO_MSG_CLASS	The default message class.
TSO_LOGON_MAX	The maximum logon region size.
TSO_PERF_GROUP	The performance group associated with the user.
TSO_SYSOUT_CLASS	The default sysout class.
TSO_USER_DATA	The TSO user data, in hexadecimal in the form X<cccc>.
TSO_UNIT_NAME	The default SYSDA device.
TSO_SECLABEL	The default logon security label.
DFP_DATA_RECORDS	Defines the information required by the System Managed Storage facility of the Data Facility Product (DFP).
AREA_NAME	Area for delivery for the user.
BUILDING	Building for delivery.
DEPARTMENT	Department for delivery.
ROOM	Room for delivery.
ADDRESS1	Address line 1
ADDRESS2	Address line 2
ADDRESS3	Address line 3
ADDRESS4	Address line 4
ACCOUNT_NUMBER	User account number for delivery.
MVS_UID	z/OS UNIX user identifier (UID) associated with the user name from the profile.
MVS_HOME_PATH	HOME PATH associated with the z/OS UNIX user identifier (UID).
MVS_PROGRAM	Default Program associated with the z/OS UNIX user identifier (UID).
MVS_MAX_CPUTIME	Maximum CPU time associated with the UID.

**Table 2—RACF Connector - Account Attributes (Continued)**

Attribute	Description
MVS_MAX_ASSSIZE	Maximum address space size associated with the UID.
MVS_MAX_FILEPROC	Maximum active or open files associated with the UID.
MVS_MAX_PROC	Maximum number of processes associated with the UID.
MVS_MAX_THREADS	Maximum number of threads associated with the UID.
MVS_MAX_MAP_STORAGE	Maximum mappable storage amount associated with the UID.
MVS_MEM_LIMIT	Maximum size of non-shared memory.
MVS_SHMEM_LIMIT	Maximum size of shared memory.
NETVIEW_OPCLASS	OPCLASS value from 1 to 2040.
EIM_LDAPPROFILE	EIM LDAPBIND profile name.

## Group attributes

---

The group schema is used when building AccountGroup objects which are used to hold entitlements shared across identities.

**Table 3—RACF Connector - Group Attributes**

Attribute	Description
SUBGROUPNAME	The name of a subgroup within the group.
MVS_GID	OMVS z/OS UNIX group identifier (GID) associated with the group name from the profile.
CSDATA_CUSTOM	Defines the custom fields associated with a group. There is one record per combination of group and CSDATA custom fields.
MEMBERS	A user ID within the group.
NAME	Group name as taken from the profile name.
SUPERIOR_GROUP	Name of the superior group to this group.
CREATE_DATE	Date that the group was defined.
OWNER_ID	The user ID or group name which owns the profile.
UACC	The default universal access. Valid values are NONE for all groups other than the IBM-defined VSAMDSET group which has CREATE.
NOTERMUACC	Indicates if the group must be specifically authorized to use a particular terminal through the use of the PERMIT command.
INSTALL_DATA	Installation-defined data.
GROUP_MODEL	Data set profile that is used as a model for this group.
UNIVERSAL	Indicates if the group has the UNIVERSAL attribute.
OVM_GID	OMVS z/OS UNIX group identifier (GID) associated with the group name from the profile.
TME_ROLE	Role profile name.

## **Schema Attributes**

# Chapter 20: SailPoint RACF LDAP Connector

---

The following topics are discussed in this chapter:

Overview .....	187
Supported features .....	187
Supported Managed Systems .....	188
Pre-requisites .....	188
Administrator permissions .....	189
Configuration parameters .....	189
Schema Attributes .....	190
Account attributes .....	190
Group attributes .....	192
Provisioning Policy Attributes .....	193
Account attributes .....	190
Additional information .....	194
Support for PassPhrase .....	194
Support for Connection Attributes .....	194
Implementing Secured Communication to RACF LDAP Server .....	194
Troubleshooting .....	197

## Overview

---

The SailPoint RACF LDAP Connector mainly uses the LDAP interfaces to communicate with z/OS LDAP server. The RACF LDAP Connector supports reading and provisioning of RACF LDAP users and entitlements.

## Supported features

---

SailPoint RACF LDAP Connector supports the following features:

- Account Management
  - Manages RACF LDAP Users as Account
  - Aggregate, Refresh Accounts, Partitioning Aggregation
  - Create, Update, Delete
  - Enable, Disable, Change Password
  - Add/Remove Entitlements
- Group Management
  - Aggregation

## **Supported Managed Systems**

---

SailPoint RACF LDAP Connector supports the following managed systems:

- IBM Tivoli Directory Server for z/OS 2.2 with SDBM LDAP back end
- IBM Tivoli Directory Server for z/OS 2.1 with SDBM LDAP back end

### **TLS communication between IdentityIQ and RACF LDAP Server**

If you want secure TLS connection for RACF LDAP, TLS communication must be enabled between IdentityIQ and RACF LDAP Server. For a Java client to connect using TLS and self-signed certificates, install the certificate into the JVM keystore.

#### *System requirements*

- The following respective components for z/OS versions must be installed for TLS communication:

<b>z/OS version</b>	<b>Cryptographic Services</b>	<b>z/OS Security Level 3</b>
z/OS 2.1	System SSL Base: FMID HCPT410	System SSL Security Level: FMID: JCPT411
z/OS 2.2	System SSL Base: FMID HCPT420	System SSL Security Level: FMID JCPT421

- The CSF started task must be active.

#### *Creating TLS communication between IdentityIQ and RACF LDAP Server*

To create TLS communication between IdentityIQ and RACF LDAP Server, perform the following:

1. Implement z/OS Secured Communication to RACF LDAP Server.  
For more information on implementing the secured communication to RACF LDAP Server, see “Implementing Secured Communication to RACF LDAP Server” on page 194.
2. Export server CA certificate and copy the exported .cer file to the Java client computer (IdentityIQ computer).
3. At the client computer execute the following command from the bin directory of JDK:  
`keytool -importcerts -trustcacert -alias aliasName -file <absolute path of certificate> -keystore <JAVA_HOME>/jre/lib/security/cacerts`  
In the preceding command line, `aliasName` is the name of the alias.
4. Login to IdentityIQ.
5. Create the application for RACF LDAP, use TLS and provide all the required values.
6. Click on **Test Connection** and save the application.

## **Pre-requisites**

---

Ensure that the following pre-requisites are satisfied for the directory servers:

- **Set the value of the LDAP\_COMPAT\_FLAGS environment variable to 1**

The SDBM attributes which are in DN format are by default returned in Uppercase format. This causes duplicate entry of entitlement in IdentityIQ due to the difference in the cases of group DN fetched while aggregation and group DN fetched while group membership provisioning operation.

To avoid the mentioned issue, the `LDAP_COMPAT_FLAGS` environment variable is set to 1 which would return the values for the mentioned attributes in mixed case format that is in the same format as of group DN returned during aggregation.

The `LDAP_COMPAT_FLAGS` environment variable value can be specified in LDAP server environment variables file. By default, the file name is `/etc/ldap/ds.envvars`.

- **RACF restriction on amount of output**

When processing certain LDAP search requests, SDBM uses the RACF `R_admin` run command interface to issue RACF search commands. The `R_admin` run command interface limits the number of records in its output to 4096. This means that the RACF search command output might be incomplete if you have many users, groups, connections, or resources.

To avoid the mentioned search limit issue, Partition must be defined to retrieve all requested objects. Partitions must be created in such a way that each Partition must not exceed the default or specified search limit. For more information on defining Partitions, see Appendix C: Partitioning Aggregation.

## Administrator permissions

---

The service account configured for SailPoint RACF LDAP Connector must have the read/write privileges over the RACF directory information tree in order to manage the RACF data, that is, the administrator user must have SPECIAL attribute to be able to manage all RACF entries. In order to limit the scope of service account, group-SPECIAL user can be created as per the requirement. Administrator user must not be a PROTECTED user that is, administrator user must have password.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The RACF LDAP connector uses the following configuration parameters:

Parameters	Description
<b>RACF LDAP Configuration Parameters</b>	
useSSL	Specifies if the connection is over TLS.
authorizationType	The authorization type to use when connecting to the server.
user*	User to connect as a DN string such as Administrator.
password	Password for the administrator account.
port*	Port number through which the server is listening.
host*	Host of the LDAP server.
racfConnectProfileDN *	Connect Profile type DN used during group membership provisioning.
setGroupAsConnectio nOwner	Sets the RACF Group as the owner of the RACF connection when connecting a RACF User to a RACF Group.
provisionPropertiesTo AllConnections	Sets the RACF connection properties defined in Provisioning Policy to all the RACF connections when multiple RACF Groups are requested in single operation.

## Schema Attributes

Parameters	Description
<b>Account Settings</b>	
searchScope	<p>Depth to search the LDAP tree.</p> <ul style="list-style-type: none"> <li>• <b>OBJECT_SCOPE</b>: Limits the search to the base object or named object.</li> <li>• <b>ONELEVEL_SCOPE</b>: Search is restricted to the immediate children of a base object, but excludes the base object itself.</li> <li>• <b>SUBTREE_SCOPE</b>: A subtree search (or a deep search) includes all child objects as well as the base object. When referrals are followed (by default, connector follow referrals) then the scope will also include child domains of the base object (when it is a parent domain) in a forest.</li> </ul>
searchDN*	Distinguished name of the container.
iterateSearchFilter	LDAP filter that defines scope for accounts/groups from this container.
filterString	Used to filter object as they are returned for an underlying application. Derived attributes can also be included in the filter.

Note: Attributes marked with \* sign are the mandatory attributes.

## Schema Attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports two types of objects, account and group.

### Account attributes

---

Account objects are used when building identities Link objects.

Attribute	Description
dn	Distinguished name by which the user is known.
racfid	ID for an user on RACF.
objectClass	Describes the kind of object which an entry represents. This attribute is present in every entry, with at least two values. One of the value is <b>top</b> or <b>alias</b> .
racfAttributes	Multi-valued attribute which list keywords that describes more about the user account. For example, racfAttributes can be used to add a RACF user entry with <b>ADSP GRPACC NOPASSWORD</b> or modify a RACF user entry with <b>NOGRPACC SPECIAL NOEXPIRED RESUME NOOMVS</b> .
racfClassName	Multi-valued attribute used to specify the classes in which the new user is allowed to define profiles to RACF for protection. Classes that can be specified are USER, and any resource classes defined in the class descriptor table.

Attribute	Description
racfDefaultGroup	Represents the default group associated with the user.
racfConnectGroupName	List of groups of which this person is a member. Example: "Sales" or "Engineering"
racfLastAccess	Information about last date-time user logged in to system.
racfProgrammerName	Users name associated with the user ID.
racfPasswordChangeDate	Last date the user changed his password.
racfPasswordInterval	Number of days during which a user's password and password phrase (if set) remain valid.
racfHavePasswordEnvelope	Information whether users password is enveloped.
racfPassPhraseChangeDate	Last date the user changed his password phrase.
racfHavePassPhraseEnvelope	Information whether users password phrase is enveloped.
racfResumeDate	Starting date when user will be allowed to access the system again.
racfRevokeDate	Starting date when user will be disallowed to access the system.
racfSecurityLabel	Users default security label.
racfSecurityLevel	Users default security level.
racfSecurityCategoryList	Multi-valued attribute contains one or more names of installation-defined security categories.
racfLogonDays	A multi-valued attribute which specifies the days of the week when the user is allowed to access the system from a terminal.
racfLogonTime	Hours in the day when the user is allowed to access the system from a terminal.
racfAuthorizationDate	Date when user was defined to RACF system.
racfInstallationData	Installation data associated the user.
racfDatasetModel	Discrete data set profile name that is used as a model when new data set profiles are created that have userid as the high-level qualifier.
racfOwner	Distinguished name of the owner of the user.
racfOperatorClass	Multi-valued attribute contains classes assigned to this operator to which BMS (basic mapping support) messages are to be routed - CICS segment.
racfOperatorIdentification	Operator ID for use by BMS - CICS segment.
racfOperatorPriority	Number from 0 - 255 that represents the priority of the operator - CICS segment.
racfTerminaltimeout	Time, in hours and minutes, that the operator is allowed to be idle before being signed off - CICS segment.
racfOperatorReSignon	Specifies whether the user is signed off by CICS when an XRF takeover occurs - CICS segment.

## Schema Attributes

Attribute	Description
SAFAccountNumber	Users default TSO account number when logging on through the TSO/E logon panel - TSO segment.
SAFDefaultCommand	Specifies the command run during TSO logon - TSO segment.
SAFDestination	Specifies the default destination to which the system routes dynamically-allocated SYSOUT data sets - TSO segment.
SAFHoldClass	Specifies the users default hold class. The specified value must be 1 alphanumeric character, excluding national characters - TSO segment.
SAFJobClass	Specifies the users default job class. The specified value must be 1 alphanumeric character, excluding national characters - TSO segment.
SAFMessageClass	Specifies the users default message class. The specified value must be 1 alphanumeric character, excluding national characters - TSO segment.
SAFTsoSecurityLabel	Specifies the users Security label entered or used during TSO LOGON - TSO segment.
SAFDefaultSysoutClass	Specifies the users default SYSOUT class - TSO segment.
SAFDefaultUnit	Specifies the default name of a device or group of devices that a procedure uses for allocations - TSO segment.
SAFDefaultLoginProc	Specifies the name of the users default logon procedure when logging on through the TSO/E logon panel - TSO segment.
SAFLogonSize	Specifies the default or requested region size during TSO logon - TSO segment.
SAFMaximumRegionSize	Specifies the maximum region size the user can request at logon - TSO segment.
SAFUserData	Specifies the optional installation data defined for the user. The specified value must be 4 EBCDIC characters. Valid characters are 0 - 9 and A - F - TSO segment

## Group attributes

The group schema is used when building AccountGroup objects which are used to hold entitlements shared across identities.

Attribute	Description
dn	Distinguished name by which the Group is known.
racfid	ID for group on RACF.
objectClass	The values of the objectClass attribute describe the kind of object which an entry represents. The objectClass attribute is present in every entry, with at least two values. One of the values is either “top” or “alias”.
racfAuthorizationDate	Date when group was defined to RACF system.
racfInstallationData	Installation data associated the group.

Attribute	Description
racfOwner	Distinguished names of objects that have ownership responsibility for the object that is owned.
racfGroupNoTermUAC	Specifies that during terminal authorization checking, RACF is to allow the use of the universal access authority for a terminal when it checks whether a user in the group is authorized to access a terminal.
racfSuperiorGroup	Distinguished name of the superior group of the associated group.
racfSubGroupName	Distinguished name of the groups to which the associated group is superior group.
racfGroupUniversal	Specifies that this is a universal group that allows an effectively unlimited number of users to be connected to it for the purpose of resource access.
racfGroupUserids	Distinguished names of the users which are member of the group.
racfDatasetModel	Discrete data set profile name that is used as a model when new data set profiles are created that have group name as the high-level qualifier.

## Provisioning Policy Attributes

---

The following table lists the provisioning policy attributes for create and update Account:

Attributes	Description
<b>Create Account</b>	
dn*	Distinguished name of the user to be created.
password*	Password of the user to be created.
racfDefaultGroup	Default group of the user to be created. Value for this field will be the DN of the group.
racfOwner	The owner of the user to be created. Value for this field will be the DN of the group or user.
connection_racfconnectowner	Distinguished name of the connection owner.
connection_racfConnectRevokeDate	Connection Revoke Date. For example, mm/dd/yy
<b>Update Account</b>	
connection_racfconnectowner	Distinguished name of the connection owner.
connection_racfConnectRevokeDate	Connection Revoke Date. For example, mm/dd/yy

**Note:** The attributes marked with \* sign are required attributes.

## Additional information

---

This section describes the additional information related to the RACF LDAP Connector.

### Support for PassPhrase

---

SailPoint RACF LDAP connector supports PassPhrase feature as follows:

For password change operation on RACF managed system, `racfPassword` or `racfPassPhrase` is supported. If the length of password provided is less than or equal to 8 characters then password attribute used would be `racfPassword` and if the length of password provided is greater than 8 characters then password attribute used would be `racfPassPhrase`.

### Support for Connection Attributes

---

SailPoint RACF LDAP Connector supports provisioning of `racfConnectionOwner` and `racfConnectRevokeDate` while provisioning entitlements. For a single entitlement request along with connection attribute values, the values of the attributes are assigned to the connection.

- **Provision Properties to All Connections:** Select to provision same set of connection attributes values to all requested entitlements.
- **Set Group As Connection Owner:** Select to set requested entitlement as connection owner.  
When **Set Group As Connection Owner** is not selected connection owner value can be specified through provisioning policy. To specify the connection owner value as `racfConnectOwner` must be included in update/create provisioning policy.

## Implementing Secured Communication to RACF LDAP Server

---

Secured communication to RACF LDAP Server must be implemented using one of the following methods:

- **LDAP SSL:** Communication must be implemented on a port defined to LDAP as secured (`ldaps`).  
For more information, see “[Implementing LDAP TLS](#)”.
- **AT-TLS policy:** Communication must be implemented on a port defined to LDAP as non-secured (`ldap`).  
The TLS processing is done by TCPIP and is transparent to RACF LDAP Server.  
For more information, see “[Implementing AT-TLS policy for RACF LDAP communication](#)”.

The secured communication is implemented using server authentication.

### Common implementation procedure

1. A valid server certificate with its associated server private key must be defined. This certificate must be signed by a trusted Certificate Authority's (CA).
2. The server certificate and the CA certificate must be connected to a key ring.
3. The CA certificate must be exported to a file, transferred (using FTP with ASCII mode) to the client and installed there to be used for certificate verification by the TLS handshake process.

**Note:** For testing purposes, a local CA can be defined for signing the server certificate.

## Implementing LDAP TLS

For detailed information about implementing LDAP TLS, see “Setting up for SSL/TLS” chapter of *z/OS IBM Tivoli Directory Server Administration and Use for z/OS IBM* manual.

**Note:** RACF LDAP server must be granted with permission to access the key ring containing the RACF LDAP server certificate and the CA certificate.

## Implementing AT-TLS policy for RACF LDAP communication

For detailed information about implementing AT-TLS policy, see “Application Transparent Transport Layer Security data protection” chapter of *z/OS Communications Server IP Configuration Guide*.

The required policy attributes for AT-TLS policy are:

- Local Port Range – ports defined in LDAP as non-secured
- Direction = Inbound
- TLS Enabled = On
- TLS v1.1 = On
- TLS v1.2 = On
- Handshake Role = Server
- Client Authorization Type = PassThru
- Application Controlled = Off
- Secondary Map = Off
- The name of the certificate created for the secured communication and the name of the key ring to which the server certificate and the CA certificate are connected, should be specified.

**Note:** TCPIP must be granted permission to access the key ring to which the RACF LDAP server certificate and the CA certificate are connected.

### Sample file for AT-TLS policy

```
# RULE for LDAP GLDSRV
#####
TTLSSRule LDAP
{
    LocalAddr ALL
    RemoteAddr ALL
    LocalPortRange 389
    Direction Inbound
    Priority 255 # highest priority rule
    Userid GLDSRV
    TTLSSGroupActionRef GrpAct_LDAP
    TTLSEnvironmentActionRef GrpEnv_LDAP
    TTLSConnectionActionRef GrpCon_LDAP
}

TTLSSGroupAction GrpAct_LDAP
{
    TTLSEnabled On
    Trace 7
}

TTLSEnvironmentAction GrpEnv_LDAP
{
```

## Additional information

```
Trace 7
HandshakeRole Server
EnvironmentUserInstance 0
TTLSKeyringParmsRef PrmKeyRing_LDAP
TTLSEnvironmentAdvancedParmsRef PrmEnvAdv_LDAP
}

TTLSEnvironmentAdvancedParms PrmEnvAdv_LDAP
{
    TLSv1.1 On
    TLSv1.2 On
    ClientAuthType PassThru
}

TTLSConnectionAction GrpCon_LDAP
{
    HandshakeRole Server
    TTLSCipherParmsRef PrmCipher_LDAP
    TTLSConnectionAdvancedParmsRef PrmConAdv_LDAP
    CtraceClearText Off
    Trace 7
}
TTLSEnvironmentAdvancedParms PrmConAdv_LDAP
{
    ApplicationControlled Off
    CertificateLabel GLDSRV
    SecondaryMap Off
}
TTLSCipherParms PrmCipher_LDAP
{
# supported cipher suites - we used a wide list, that should be decreased according
# to specific needs
V3CipherSuites      TLS_DH_DSS_WITH_DES_CBC_SHA
V3CipherSuites      TLS_DH_RSA_WITH_DES_CBC_SHA
V3CipherSuites      TLS_NULL_WITH_NULL_NULL
V3CipherSuites      TLS_RSA_WITH_NULL_MD5
V3CipherSuites      TLS_RSA_WITH_NULL_SHA
V3CipherSuites      TLS_RSA_EXPORT_WITH_RC4_40_MD5
V3CipherSuites      TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5
V3CipherSuites      TLS_RSA_WITH_DES_CBC_SHA
V3CipherSuites      TLS_DHE_DSS_WITH_DES_CBC_SHA
V3CipherSuites      TLS_DHE_RSA_WITH_DES_CBC_SHA
V3CipherSuites      TLS_RSA_WITH_AES_256_CBC_SHA256
V3CipherSuites      TLS_RSA_WITH_AES_256_CBC_SHA
V3CipherSuites      TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256
V3CipherSuites      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
V3CipherSuites      TLS_RSA_WITH_AES_128_CBC_SHA256
V3CipherSuites      TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256
V3CipherSuites      TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256
V3CipherSuites      TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
V3CipherSuites      TLS_DHE_DSS_WITH_AES_128_CBC_SHA256
V3CipherSuites      TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
V3CipherSuites      TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
V3CipherSuites      TLS_RSA_WITH_AES_128_CBC_SHA
V3CipherSuites      TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA
V3CipherSuites      TLS_ECDH_RSA_WITH_AES_128_CBC_SHA
V3CipherSuites      TLS_DHE_RSA_WITH_AES_128_CBC_SHA
```

```

V3CipherSuites      TLS_DHE_DSS_WITH_AES_128_CBC_SHA
V3CipherSuites      TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
V3CipherSuites      TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
V3CipherSuites      TLS_RSA_WITH_AES_128_GCM_SHA256
V3CipherSuites      TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256
V3CipherSuites      TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256
V3CipherSuites      TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
V3CipherSuites      TLS_DHE_DSS_WITH_AES_128_GCM_SHA256
V3CipherSuites      TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA
V3CipherSuites      TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
V3CipherSuites      TLS_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA
V3CipherSuites      TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA
}
TTLSKeyringParms  PrmKeyRing_LDAP
{
    Keyring GLDRING
}

```

## Troubleshooting

---

### 1 - When setting password/passphrase with 9 - 13 characters an error message is displayed

When setting password/passphrase with 9 - 13 characters, the following error message is displayed:

Invalid Password

**Resolution:** Passphrase can be 9 - 100 characters if KDFAES or ICHPWX11 encryption algorithm is present on the server. If KDFAES or ICHPWX11 encryption algorithm is not present on the server then the allowed number of characters for passphrase are 14 - 100.

### 2 - Change Password operation fails with an error

When performing a self change password operation for an account and if any one of the connection is revoked, the following error message is displayed:

```
[LDAP:error code 1 - R000208 Unexpected racroute error safRC=8 racfRC=36
racfReason=0(srv_authenticate_native_password:3567)]
```

**Resolution:** For change password operation, connections of the accounts must not be revoked.

## **Troubleshooting**

# Chapter 21: SailPoint Salesforce Connector

---

The following topics are discussed in this chapter:

Overview .....	195
Supported features .....	195
Administrator permissions .....	196
Configuration parameters .....	196
Additional configuration parameters .....	197
Schema attributes .....	198
Account attributes .....	198
Profile attributes .....	200
Provisioning Policy attributes .....	201
Troubleshooting .....	202

## Overview

---

The SailPoint Salesforce Connector supports reading and provisioning of Salesforce accounts, profiles as account groups and implement the **sailpoint.connector** interface.

This connector is written using the `partner.wsdl` and underlying soap interface. The connector uses SOAP stub generated from a wsdl that was available at the time of development. The stubs are generated using axis 1.2. We do not have to generate the stubs once already done. Partner API is easy to use and we can add custom attributes in the schema without generating the stubs. Partner API is generic and have the same Java implementation for Salesforce connector.

The API is fairly rich for SOAP based API and has the concept of login which requires us to login just once for each operation. It has formal models around the user and profile objects and they are generated as part of the stubs. Earlier Salesforce Connector internally used Enterprise WSDL which was complex to use (regenerating STUB classes by customer) and not much flexible on custom attributes. With Partner WSDL approach, those limitations will be removed.

## Supported features

---

SailPoint Salesforce Connector supports the following features:

- Account Management
  - Manages Salesforce users as Accounts
  - Aggregation, Refresh Accounts
  - Create, Update
  - Enable, Disable, Change Password
  - Add entitlement (Account-Groups and User Roles)
  - Add and Remove entitlements (PermissionSet)

## Configuration parameters

**Note:** Administrator Reset Password operation does not set password provided for the user account. Salesforce sends Email Notification with temporary password to the user for these operations.

- Account - Group Management
  - Manages Salesforce Profiles as Account-Groups
  - Aggregation, Refresh Groups
- Permission Management
  - Application reads permissions directly assigned to groups as direct permissions during group aggregation.
  - The connector does not support automated revocation of the aggregated permissions and creates work item for such requests

## Administrator permissions

---

For user provisioning, it is required that the administrator must have the appropriate rights on the Salesforce Account.

The System Administrator Profile can configure and customize the application.

- Has access to all functionality that does not require an additional license. Can create, edit, and delete custom profiles. Can reset password of multiple user accounts.
- Can add multiple user accounts.
- Has access to all User Accounts, Profile Permissions
- Enable /Disable User Accounts

## Profile Access to User Accounts

A user Profile determines what a user can do in the system. By default, the System Administrator Profile can do the most; the Read Only Profile can do the least. For most users, the Standard User Profile is a good choice: it lets people create and edit most records, as well as access and run reports.

(The following assumes that you are a System Administrator for your organization's instance of Salesforce.com.)

Users, Roles and Profiles are all configured within the Setup area. To access these settings when logged in to Salesforce, click on your name in the upper right corner, then choose Setup from the drop-down menu. The Users, Roles and Profiles settings are all available under Manage Users in the lower left Administration Setup menu.

## User Licenses create access

Most of your users will need a standard Salesforce user license. This license gives the user full access to Salesforce's CRM features and applications, including Chatter. Other user license options limit user access.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Salesforce connector uses the following connection parameters:

**Table 1—Configuration parameters**

Parameters	Description
Salesforce URL*	<p>Enter the fully qualified url to the root of the salesforce server. For example, <code>http://login.salesforce.com/services/Soap/u/26.0/</code></p> <p><b>Note:</b> To figure out the url of your site, login to salesforce.com. Click Develop under the Application heading toward bottom. Next, click API &gt; Generate Partner WSDL, and click Generate. The URL is located under the SalesforceService service name.</p>
Username*	<p>Display name attribute. It's typically in an email in email type format.</p> <p>For example, <code>denise.hunt@demoexample.com</code></p>
Password*	Defines the password which is used for logging in the managed System.
Manage Active Accounts	<p>Retrieves the active accounts during account aggregation. Otherwise it retrieves all the accounts which are enabled/disabled while account aggregation.</p>
Search Query for User/Profile	<p>Helps to scope the User/Profile that are retrieved during Account or Account-Group aggregation.</p> <p>For instance, specifying the following search query, retrieves only Active Users during Account Aggregation:</p> <pre>select Id from User where IsActive = true</pre> <p>Users or Profiles retrieved during aggregation can be scoped using custom attributes in the where clause as follows:</p> <pre>select Id from User where EMP_DEPARTMENT__c= 'tester'</pre> <p><b>Note:</b> Only the where clause of the search query can be modified as per the customers requirement.</p> <p>While configuring Salesforce application if where clause in <b>Search Query For User/Profile</b> field contains apostrophe(') then use backslash(\) prefix to apostrophe. For example,</p> <pre>Find Id,username from user where lastname is buru'4</pre> <p>Expected Query: Select Id,username from user WHERE lastname = 'buru\'4'</p>

**Note:** In the above table all the attributes marked with \* sign are mandatory attributes.

## Additional configuration parameters

- To enable the session and getObject feature, update the following parameters in the application debug page:
  - **sessionEnabled:** Set this parameter to **true** to persist the session information. Default: **false**
  - **disableGetObject:** Set this parameter to **true** to perform the 'getObject' operation after creating an account on the managed system. Default: **false**

## Schema attributes

- To enable the delete operation, set the value of the **deleteToDisable** parameter to **true** as follows in the application debug page:

```
<ProvisioningConfig deleteToDisable="true" />
```

# Schema attributes

---

This section provides the different attributes of the Account attributes and Profile attributes for Salesforce connector.

**Note:** The Identity Attribute for account and group has been changed to 'Id'. For applications created in prior to version 6.4, IdentityIQ would display the Identity Attribute for account as 'Username' and for groups the Identity Attribute would be 'Name'.

## Account attributes

---

The Salesforce connector returns several attributes falling into two categories. The first are general attributes: name, city, state, and so on. Additionally, there are entitlement attributes that specifies user level access granted to Salesforce:

Attributes	Description
UserName	By default, this attribute is the connectors default nativIdentity AND display name attributes. It's typically in an email type format. For example, denise.hunt@demoexample.com
Id	This attribute is the connector default nativIdentity and internal salesforce id like "005A00000014ySylXX".
Name	Users fullname.
FirstName	Users firstname.
LastName	Users lastname.
Alias	Users assigned alias.
City	Users city.
CommunityNickname	Display Names for user's online communities
CallCenterId	Users call center.
CompanyName	Users company name.
Country	Users country.
Department	Users department.
Email	Users Email address.
Division	Users division.
EmployeeNumber	Users employee number.
Extension	Users telephone extension.
Street	Name of the street.
Fax	Users fax number

Attributes	Description
FederationIdentifier	A Federation ID is an identifier that is unique within a salesforce Organization.
IsActive	Flag that indicates if the user is active in sf. False would indicate disabled.
EmailEncodingKey	Encoding that should be used during email communications
ProfileId	ID of the profile assigned to a user. Profiles contain settings and permissions, which control what users can do. The available profiles depend on which user license is selected.
ProfileName	Name of the profile assigned to a user. Profiles contain settings and permissions, which control what users can do. The available profiles depend on which user license is selected.
UserRoleId	User Role's Id.
UserRoleName	User Role's name.
PublicGroups	Public groups are the entitlements for user.
UserPermissionsMarketingUser	Maps to the Marketing User Flag.
UserPermissionsMobileUser	Maps to the Mobile User Flag.
UserPermissionsOfflineUser	Maps to the Offline user Flag.
Phone	Users phone number.
ReceivesAdminInfoEmails	Receive the salesforce.com administrator newsletter.
UserType	Type of the user.
UserPermissionsSFContentUser	Maps to Sales Anywhere User.
ReceivesInfoEmails	Receive the salesforce.com newsletter.
State	Users state.
Title	Users title.
TimeZoneSidKey	Defaults to America/Los_Angeles. The timezone of the user, it uses a display name defined by sales force. Only a few timezones are defined in the policy drop down and this will need to be customized for each customer.
LocaleSidKey	Defaults to UTF-8. This is the user's locale.
Email EncodingKey	Defaults to UTF-8 and there are several selections to choose from in from the web interface. They can be customized by the customer.
LanguageLocaleKey	Defaults to en_US. There are several selections to choose from in from the web interface. They can be customized by the customer.

## Schema attributes

**Note:** The ProfileId and UserRoleID fields are required in the schema to fetch the ProfileName and userRoleName respectively. If the ProfileId or UserRoleID is removed then profile name and user role name will not be fetched.

## Additional account attributes for Salesforce connector

In addition to the above account attributes, following are the additional custom attributes which are required for configuration to connect to Salesforce connector:

**Note:** The attributes in the following table must be added manually when upgrading IdentityIQ from any version below 6.4 to IdentityIQ version 7.1.

Attributes	Description
PermissionSet	PermissionSet assigned to a user. PermissionSet contain settings and permissions which control users action. The available Permission depends on which user license is selected. User can have multiple permission sets.
UserLicense	User's license.

Support for custom attributes which are not present in Salesforce system but are required internally in IdentityIQ (may be for a process in a correlation rule):

SailPoint Salesforce Connector respects the attribute only if it starts with \_#. For example  
\_#Emp\_company\_history.

## Profile attributes

---

Profiles are aggregated during account group aggregation, below are the attributes returned by the group aggregation process.

Attributes	Description
Id	The internal id for this group. For example, 00eA000000OoP6IAK.
Name	The friendly name assigned to the profile. For example, Force.com - Free User, it also has to be unique so that it can be used as the identity and display attribute by default.
UserType	This is the type of profile even though the attribute name would indicate a user.
Description	Description for the profiles.
UserLicense	User's license. <b>Note:</b> This attribute must be added manually when upgrading from any version below 6.4 to version 7.1.

**DirectPermissions:** The connector reads the permissions assigned to a profile using the salesforce.com api. To get the permissions, the connector queries the service to describe the profile object. In the returned attribute all of the permissions contained by a group are prefixed with **Permissions**, and camel cases the permission such that right and target are separated by camel case convention. For example, **PermissionsEditTask** or **PermissionsTransferAnyEntity**. We break these down into a Permission attribute per prefixed-attribute.

## Provisioning Policy attributes

---

IdentityIQ has a default Provisioning Policy defined which allows for the creation of accounts. The provisioning policy can be edited to fit specific customer environments.

Most of the fields on the Salesforce connector default provisioning policy are generated and all fields are marked review required. The provisioning policy attributes must be customized based on specific customer requirements.

Attributes	Description
<b>Create User Policy</b>	
Alias	8 character alias, which is required. By default, it generates a value based on lastname and firstname in the field's inline script. It takes first 7 chars from last name and prefixes it with the first character of the first name.
IsActive	Defaults to true.
Username	Defaults to the identity's user name.
Email	Defaults to the identity's email address.
FirstName	Defaults to the identity's first name.
Lastname	Defaults to the identity's last name
CommunityNickname	Defaults to identity's full name.
TimeZoneSidKey	Defaults to America/Los_Angeles. The timezone of the user, it uses a display name defined by sales force. Only a few timezones are defined in the policy drop down and this will need to be customized for each customer.
LocaleSidKey	Defaults to UTF-8. This is the user's locale.
Email EncodingKey	Defaults to UTF-8 and there are several selections to choose from in from the web interface. They can be customized by the customer.
LanguageLocaleKey	Defaults to en_US. There are several selections to choose from in from the web interface. They can be customized by the customer.
Federation Identifier	A Federation ID is an identifier that is unique within a salesforce Organization.

# Troubleshooting

---

- ProfileName is required to select in **Entitlement** section as it is mandatory to Salesforce, else it will not create account in Salesforce system.
- If Duplicate User error is displayed, it means that the email Id of that user is already utilized in Salesforce system. You have to create new account with different email id. Email id is the username of the user which is mandatory.
- If any of the create User policy attributes are not filed up, it will display an error and it may happen in rare case that the email id is used by Salesforce system and you have to create new account with different email Id's.
- The Community nickname must be unique.
- Do not add profileId/userRoleId attribute in create /update user policy as the code will automatically handle when the customer is selecting profile name and userrolename from **Entitlement** section.
- If the total active account limit on Salesforce exceeds, the connector fails to create new account by displaying an error. You can increase the account limit on Salesforce or disable the existing active accounts in Salesforce system before initiating create request.
- In Salesforce, Email notification is not sent to the user if the `disableUserCreationEmail` flag is set to true as follows:

```
<entry key="disableUserCreationEmail">
    <value>
        <Boolean>true</Boolean>
    </value>
</entry>
```

- If the Salesforce url has version mismatch with the existing stubs version, and if any of the Aggregation task fails, then generate new stubs of that specific version.

For more information, see “ Configuration of Stubs generation” section below.

## *Configuration of Stubs generation*

Salesforce.com allows each customer to extend the schemas for objects. Customers add new attributes specific to the data kept by a company as the connector uses SOAP. Each time a customization is made to the salesforce.com data model by the customer a new wsdl file is retrieved and integrated into the IdentityIQ environment. To integrate changes from salesforce.com to IdentityIQ perform the following:

1. Download the new version of the partner wsdl from Salesforce.com

After your changes to the salesforce.com model you must download the newest wsdl file from salesforce.com which will include the customized attributes as part of the SOAP stub model. This involves logging into Salesforce and requesting the new wsdl file, the wsdl file is generated using the customer's data model defined in salesforce when the wsdl is generated.

In salesforce, the navigation is: **Setup => App Setup => Develop => API => Generate Partner WSDL**.

2. Generate stubs using the downloaded wsdl file and AXIS.

After generating and downloading the new version, run them through AXIS classes that will generate java stubs. Use the `iq` command to configure the class path to AXIS and then point to the new wsdl file.

For example, `$iq org.apache.axis.wsdl.WSDL2Java -p sailpoint.connector.salesforce.webservices.partner partner.wsdl`

**Note:** The `-p` flag tells the generator which package to place the stubs under. In this case the class files will be placed into a directory named sailpoint

3. Compile generated stubs using JAVA.

The stubs that are generated by axis are java class files and reference only the axis libraries. Building the java files is very simple and below are some instructions on how this can be done.

The stubs must be compiled using axis 1.4 (the old version of axis not axis2) along with any supported jdk.

The compile procedure appears as the following example:

- a. Create a directory called salesforcestubs/src.
- b. Copy files from the sailpoint directory generated from the wsdl file into salesforcestubs/src directory.
- c. In the salesforcestubs directory create a file called build.xml and copy the sample build.xml (listed in the last section) into that file.

Directory structure of salesforcestubs should look like this:

```
build.xml
src/sailpoint/connector/salesforce/webservices/partner
```

(this is where all the .java files will be)

- d. Edit the pathToAxis1\_4 property to point at the axis 1.4 distribution (where axis.jar exists). If this is being done on a local box where IdentityIQ is expanded, you can alternately just specify the *iqHome* and it will find the jar file relative to that directory.
- e. Run ant command. This will compile all of the Java files into .class files which are placed into a directory called build which has the typical web-application directory structure. (For example, WEB-INF/classes/).
4. Copy newly compiled stubs into WEB-INF/classes/sailpoint/connector/salesforce/webservices/partner directory.

Copy the new class files manually by copying the contents produced in the build directory to the installation directory.

If you are doing this locally on the same machine AND using the example build.xml alternatively, you can use the 'copy' target (ant copy). This target will copy the new salesforce.com stub class files from the build area into to the *iqHome* directory specified at the top of the build.xml file.

Once the files have been copied, the java work is done then configure IdentityIQ to fetch and write the custom attributes

5. Restart the application server.

After the new class files have successfully been copied over to the IdentityIQ directory you must reboot the application server and close any open consoles.

In order for the salesforce connector to read the new version, change the old version to new version in salesforce URL. Additionally, if you are provisioning salesforce accounts, define new attributes in your provisioning policies.

Sample build.xml file

Using the following build.xml file along with ant makes building these stubs simple.

```
<!-- (c) Copyright 2008 SailPoint Technologies, Inc., All Rights Reserved. -->
<project name="salesforceStubs" default="build" basedir=".">
  <description>
    Build file for to help compiling stubs for sales force.
  </description>
  <!-- **** Global properties **** -->
  <!-- Global properties -->
  <!-- **** Global properties **** -->

  <property name="src" location="src" />
```

## Troubleshooting

```
<!-- Only necessary if you want to copy the class files using ant -->
<property name="iiqHome" location="c:/home/SystemID/work/trunk/build/" />

<!-- This needs to point at a directory with the axis 1.4 distribution. -->
<!-- By default relative to the iiq instalation, but doesn't have to be -->
<!-- if on a different machine -->
<property name="pathToAxis1_4"
          location="${iiqHome}/WEB-INF/lib/" />

<property name="build" location="build" />

<!-- **** -->
<!-- Target: build -->
<!-- **** -->

<target name="build" >
    <mkdir dir="${build}/WEB-INF/classes" />

    <!-- build the salesforce axis stubs -->
    <javac destdir="${build}/WEB-INF/classes"
           debug="true" fork="true" memoryMaximumSize="512m"
           includeantruntime="false">
        <src path="${src}"/>
        <classpath>
            <fileset dir="${pathToAxis1_4}">
                <include name="**/*.jar" />
            </fileset>
        </classpath>
    </javac>

</target>

<!-- **** -->
<!-- Target: clean -->
<!-- -->
<!-- Clean build area -->
<!-- -->
<!-- **** -->

<target name="clean">
    <delete dir="${build}" />
</target>

<!-- **** -->
<!-- Target: copy -->
<!-- -->
<!-- Copy stub class files into the IIQ installation -->
<!-- Target: copy -->
<!-- **** -->
```

```
<target name="copy" depends="build">
  <copy todir="${iiqHome}">
    <fileset dir="${build}">
      <exclude name="**/*.java" />
    </fileset>
  </copy>
</target>

</project>
```

## **Troubleshooting**

# Chapter 22: SailPoint SAP Portal-User Management Web Service Connector

---

The following topics are discussed in this chapter:

Overview .....	207
Supported features .....	207
Supported Managed Systems .....	208
Pre-requisite .....	208
Administrator permission .....	208
Configuration parameters .....	208
Schema attributes .....	209
Account attributes .....	209
Group attributes .....	210
Provisioning Policy attributes .....	211
Additional information .....	212

## Overview

---

SAP Enterprise Portal integrates information and applications across the enterprise to provide an integrated single point of access to information, enterprise applications, and services both inside and outside an organization. SAP Enterprise Portal Connector uses the UME service to perform user management. The User Management Engine (UME) provides a centralized user management for all Java applications and can be configured to work with user management data from multiple data sources.

The UME can be configured to read and write user-related data from and to multiple data sources, such as Lightweight Directory Access Protocol (LDAP) directories, the system database of the AS Java, and user management of an AS ABAP.

SailPoint SAP Portal-User Management Web Service Connector manages the following entities of SAP User Management Engine (UME):

- User
- Role (UME and Portal)

## Supported features

---

SailPoint SAP Portal-User Management Web Service Connector supports the following features:

- Account Management
  - Manages SAP Portal users as Accounts
  - Aggregation, Refresh Accounts, Pass Through Authentication
  - Create, Update, Delete
  - Enable, Disable, Change Password
  - Add/Remove Entitlements

## Configuration parameters

- Account - Group Management
  - Create, Update, Delete
  - Manages SAP Roles as Account-Groups
  - Aggregation

## Supported Managed Systems

---

Following versions of SAP NetWeaver versions are supported by the SAP Portal-User Management Web Service Connector:

- SAP NetWeaver 7.5, 7.4, 7.3, 7.2 and 7.1

**Note:** **SailPoint SAP Portal-User Management Web Service Connector manages SAP User Management Engine users.** For more information, see "Supported features" on page 207.

## Pre-requisite

---

The sailpoint\_ume.sda file must be deployed on the SAP Enterprise Portal server which must be provisioned.

Perform the following steps to deploy the sailpoint\_ume.sda file:

1. Copy the SDA file from (\$build)/integration/sap/dist directory to a temporary directory on the SAP server.
2. Navigate to the home directory of SAP Enterprise Portal server  
..\\usr\\sap\\(ep\_instance\_name)\\J02\\j2ee\\console on SAP server and execute textconsole.bat.
3. Run the following command:  
>DEPLOY tmpDir\\sailpoint\_ume.sda (location of the file sailpoint\_ume.sda)  
where tmpDir is the temporary directory where the SDA file is extracted.

For undeploying the .sda file, see "Undeploy .sda file" on page 212.

## Administrator permission

---

The administrative account must have either one of the following permissions mentioned below for performing provisioning operation:

- pcd:portal\_content/administrator/user\_admin/user\_admin\_role
- pcd:portal\_content/administrator/system\_admin/system\_admin\_role
- pcd:portal\_content/administrator/super\_admin/super\_admin\_role
- SAP\_J2EE\_ADMIN

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The SAP Portal UMWebService connector uses the following connection attributes:

**Table 1—SAP Portal UMWebservice Connector - Primary Attributes**

Attribute	Description
UMWebService URL*	<p>The url for the UMWebService. For example:</p> <pre>http://HOST:PORT</pre> <p>In the above url, <i>HOST</i> refers to the instance where SAP Portal-User Management WebService is installed and <i>PORT</i> is the listening port of the server.</p> <p>This url can use either http or https.</p> <p><b>Note: When using https, the portal server's keystore and the application server's keystore must be configured.</b></p>
Username*	The SAP Portal user name used when connecting to the web service.
password*	Password for the user account specified in Username.
Account Filter	<p>Enter the string representation of an object filter. Any account object matching the filter is filtered out of the dataset. The following is an example of a filterString that filters out all objects where the uniqueId starts with USER.R3_DATASOURCE:</p> <pre>uniqueId.startsWith(&amp;quot;USER.R3_DATASOURCE.&amp;quot;)</pre> <p>If this property is non-empty, filtering happens on the IdentityIQ server side and does not filter on the SAP portal side.</p>
Group Filter	<p>Enter the string representation of an object filter. Any roles object matching the filter is filtered out of the dataset. The following is an example of a filterString that filters out all objects from the that have a displayName starting with com.sap.pct:</p> <pre>displayName.startsWith(&amp;quot;com.sap.pct&amp;quot;)</pre> <p>When this property is non-empty filtering happens on the IdentityIQ server side and does not filter on the SAP portal side</p>

## Schema attributes

---

This section describes the different schema attributes.

**Note:** The attributes marked with \* sign are the required attributes.

### Account attributes

---

The following table lists the account attributes:

Attributes	Description
uniqueId	Users unique identification.

## Schema attributes

Attributes	Description
firstName	Users first name.
lastName	Users last name.
displayName	Users display name.
company	Users company name.
title	Users title.
uniqueName (Identity Name+ Display Name)	Users unique name.
city	Users city.
postalCode	Users postal address.
email	Users email address.
street	Users street.
state	Users state.
country	Users country.
zip	Users postal zip code.
fax	Users fax.
telephone	Users telephone number.
cellPhone	Users cell phone number.
department	Users department assigned.
salutation	Users salutation.
jobTitle	Users job title.
timeZone	Timezone of the user.
language	Language of the user.
securityType	Users's security type.
lockStatus	User is locked or open.
roles	Role assigned to the user.
groups	Groups assigned to the user.
validFrom	Valid from date.
validTo	Valid to date.

## Group attributes

The following table lists the group attributes:

Attributes	Description
displayName is	Display name of the role.

Attributes	Description
uniqueName identity Attribute	Unique name of the role.
uniqueld	Unique ID of the role.
description	Description of the role.
userMembers	Users associated to the role.
groupMembers	Groups associated to the role.

## Provisioning Policy attributes

---

This section lists the different policy attributes of SAP Portal-User Management WebService Connector.

**Note:** The attributes marked with \* sign are the required attributes.

### Create account attributes

---

The following table lists the provisioning policy attributes for Create Account:

Attributes	Description
uniqueld	Users unique identification.
First Name	Users first name.
Last Name*	Users last name.
Display Name	Users display name.
company	Users company name.
Department	Users department assigned.
Unique Name*	Users unique name.
Password*	Users password.
City	Users city.
Street	Users street.
Email	Users email address.
State	Users state.
Country	Users country.
Zip	Users postal zip code.
Fax	Users fax.
Tele Phone	Users telephone number.
Cell Phone	Users cell phone number.
Salutation	Users salutation.
JobTitle	Users job title.

## Additional information

Attributes	Description
Language	Language of the user.
Security Type	Users's security type.
Lock Status	User is locked or open.
Password Change Required	To create a new account in SAP Portal Server with productive password.  Values are as follows: <ul style="list-style-type: none"><li>• True: Does not sets the password as productive</li><li>• False: Sets the password as productive.</li></ul> <b>Note:</b> User must add “changePasswordRequired” attribute in schema and create provisioning policy and set the required display name (for example, “Password Change Required”).

## Create Group attributes

The following table lists the provisioning policy attributes for Update Account:

Attributes	Description
Role Name*	Display name of the role.
Description	Description of the role.
User Members	Users associated to the role.
Group Members	Groups associated to the role.

## Additional information

This section describes the additional information related to the SAP Portal-User Management Web Service Connector.

## Undeploy .sda file

Perform the following steps to undeploy the .sda file:

1. From the command prompt browse the following location:  
..\\usr\\sap\\(SAPEP instance)\\J02\\j2ee\\console
2. Run the following file:  
textconsole.bat
3. At the query prompt enter the following command:  
>UNDEPLOY name=SailpointSapEPArchive vendor=sailpoint.com

# Chapter 23: SailPoint SAP HR/HCM Connector

---

The following topics are discussed in this chapter:

Overview .....	213
Supported features .....	213
Supported Managed Systems .....	214
Pre-requisites .....	214
Administrator permissions .....	214
Configuration parameters.....	217
Schema Attributes .....	219
Account attributes .....	219
Additional information .....	225
Troubleshooting.....	226

## Overview

---

The SailPoint SAP HR/HCM Connector aggregates and provisions the employee information from the SAP HR/HCM system.

SAP HR/HCM Connector supports the following SAP Info Types:

- Action (0000)
- Organizational Assignment (0001)
- Personal Data (0002)
- Addresses (0006)
- Communication (0105)

In addition to aggregate any specific Info Type the build map rule can be used.

## Supported features

---

SailPoint SAP HR/HCM Connector supports the following features:

- Account Management
  - Manages SAP HR/HCM employees as Accounts (Active, Terminated and Future Hires)
  - Aggregation, Partitioning Aggregation, Delta Aggregation, Refresh Accounts

For more information on Delta and Partitioning Aggregation, see “Delta Aggregation” on page 224 and “Partitioning” on page 224.
- Provisioning (with the help of rule) support added for IdentityIQ version 7.0 and above
  - Ability to define separate provisioning rule for specific operation (operations that include are Enable, Disable, Unlock, Delete, Create, and Modify).

An example of modify provisioning rule is located in WEB-INF/config/examplerules.xml file. For more information, see “Customization Rule” on page 225 section.

## Supported Managed Systems

---

Following versions of SAP HR/HCM system are supported by the SAP HR/HCM connector:

- SAP ECC 6.0 on SAP NetWeaver 7.5, 7.4, 7.3, 7.2, 7.1 and 7.0

## Pre-requisites

---

SAP JCO version 3.0.x libraries, along with `sapjco3.dll` (on Microsoft Windows) or `libsapjco3.so` (on UNIX), must be present in the `java.library.path` directory on the IdentityIQ host. The JCO libraries (JCO Release 3.0.x) must be downloaded from the SAP website by navigating to the customer service marketplace.

## Administrator permissions

---

The following table lists the required permissions for the specific operations mentioned below in this section:

**Table 1— Operation specific required permissions**

Operation	Required permissions
Test Connection	Test Connection
Account Aggregation	Test Connection and Account Aggregation
Delta Aggregation	Test Connection, Account Aggregation and Delta Aggregation
Provisioning Rule	Test Connection, Account Aggregation and Provisioning Rule

The role assigned to the SAP Administrative user must have the following Authorization Objects as mentioned in the tables below.

### Test Connection

Authorization Objects	Field name	Field description	Field value
S_RFC	ACTVT	Activity	16 -Execute
	RFC_NAME	Name of RFC object	RFCPING
	RFC_TYPE	Type of RFC object	FUGR, FUNC

## Account Aggregation

Authorization Objects	Field name	Field description	Field value
S_RFC	RFC_NAME	Name of RFC object	OPBAPI0105,BAPI_ADDRESSEMPGE TDETAILEDLIST, BAPI_EMPLCOMM_GETDETAILEDLI ST, BAPI_EMPLOYEE_GETDATA, BAPI_EMPLOYEE_GETLIST, BAPI_PERSDATA_GETLIST, MSS_GET_SY_DATE_TIME,  BAPI_PERSDATA_GETDETAIL, SDTX, SMSSDATA1, PERS,PADR,RFC_GET_FUNCTION_I NTERFACE, DDIF_FIELDINFO_GET
S_TABU_NAM	ACTVT	Activity	03 Display
	TABLE Name	TABLE	HRP1001, HRP1000, PA0000, PA0001, PA0002  <b>Note: T530T, T529T (Add these tables name only if you want to populate Infotype0000 account schema attribute)</b>  <b>Note: PA0006, PA0105 (Add these tables if Delta Aggregation Enabled checkbox is selected)</b>

Authorization Objects	Field name	Field description	Field value
P_Orgin	AUTHC	Authorization Level	R
	INFETY	INFOTYPE	0001, 0002, 0003, 0006, 0032, 0105
	PERSA	Personal area	(Depending on the organizational area you have assigned to user while creating)
	PERSG	Employee group	(Depending on the organizational area you have assigned to user while creating)
	SUBTYPE	SUBTY	<ol style="list-style-type: none"> <li>1. '*' - If you want to get data for all the subtypes</li> <li>2. If you want to get data for specific info types then add specific subtypes as per data in your environment)</li> </ol> <p>For example: For Addresses (Infotype 0006) add subtype - 1,2,3,4,5,6</p> <p>For Communication (Infotype 0105) add subtype - 0001,0010, 0020, 0030</p>
	PERSK	Employee sub group	(Depending on the organizational area you have assigned to user while creating)
	VDSK1	Organization Key	(Depending on the organizational area you have assigned to user while creating)

## Provisioning Rule

The administrator permissions mentioned in the following table are applicable only to the provisioning operations specified in the **Example SAP HRMS Modify Rule** (E-mail, Phone number and System user name) rule. This rule is specified in the `examplerules.xml` file located in `WEB-INF/config`/directory.

Authorization Objects	Field name	Field description	Field value
S_RFC	RFC_NAME	Name of RFC object	RFC1, 1065, BAPI_EMPLOYEE_ENQUEUE, SYSU, SYSTEM_RESET_RFC_SERVER, SDIFRUNTIME
P_Orgin	AUTHC	Authorization Level	E,S,W
	INFETY	INFOTYPE	0007

# Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The SAP HR/HCM connector uses the following connection attributes:

**Table 2—SAP HM/HCM Connector - Configuration parameters**

Parameters	Description
<b>SAP JCO Connection Settings</b>	
SAP Host*	Host on which the SAP Server is running
System Number*	2-digit SAP system number. Default: 00
ClientNumber*	3-digit SAP client number. Default: 001
ClientLanguage*	2-letter SAP client language. Default: EN
Username*	SAP service account which has permissions mentioned in “Administrator permissions” on page 214.
Password*	SAP service account password.
Action Type(s)	Enter the comma separated value of Action Type(s) to be aggregated for each SAP HR person. For example: 01, 20, 21  In the above example, 01 stands for Hiring, 20 for Termination and 21 for Re-Hire.
Aggregate Inactive Employees	Select this checkbox to aggregate inactive employees. Inactive employees refers to the employees having the following values of STAT2 as follows: <ul style="list-style-type: none"> <li>• STAT2 = 0 refers to Action Type Termination</li> <li>• STAT2 = 1 refers to Action Type Leave of absence</li> <li>• STAT2 = 1 refers to Action Type Retirement</li> </ul>
Inactive Employees Offset	Enter the number of past days to aggregate inactive employees. The Inactive Employees Offset can have the following values: <ul style="list-style-type: none"> <li>• <b>0</b>: aggregates only the active employees</li> <li>• <b>Blank</b>: aggregates all the inactive employees</li> <li>• <b>Any positive value</b>: indicates the number of days in past since when the terminations must be aggregated</li> </ul> Default value is 30.

## Configuration parameters

**Table 2—SAP HM/HCM Connector - Configuration parameters**

Parameters	Description
Future Dated Hires Offset	<p>Indicates the number of days to aggregate the future hires. The Future Dated Hires Offset can have the following values:</p> <ul style="list-style-type: none"> <li>• <b>0</b>: aggregates no future hires.</li> <li>• <b>Blank</b>: aggregates all future hires until 9999-12-31,</li> <li>• <b>A positive value</b>: aggregates Future Hires within the specified number of days.</li> </ul> <p>Default value is 30.</p>
JCO RFC Trace	<p>To enable JCO RFC client trace. If enabled, the JCO traces are written to one or multiple .trc files.</p> <p><b>Note:</b> .trc files are generated on the application server where IdentityIQ is running.</p>
BuildMap Rule	<p>A rule is used to modify the default object or add custom attributes to each object returned from SAP HR server. (A rule that is called for each object returned from SAP.)</p>
<b>Manager Configuration</b>	
Manager Relationship Model	<p>Select one of the following model to determine the manager of an employee:</p> <ul style="list-style-type: none"> <li>• <b>Organization Chief Manager Model (O-O-S-P)</b>: OOSP (Organization(O)-Organization(O)-Position(S)-Person(P)) model deals with the chief of organization model in which employee's organization is detected with the relationship code 012. between position of employee and organization unit.</li> <li>• <b>Supervisory Model (S-S)</b>: SS (position(S)-position(S)) model deals with the position-position manager relationship model in which the two positions are connected with the relationship code 002.</li> <li>• <b>Custom</b>: Custom rule to determine the manager of the SAP HR employee. An example of custom rule is located in WEB-INF/config/examplerules.xml file. For more information on <b>Manager Rule</b>, see “Customization Rule” on page 225.</li> </ul>
<b>SNC Configuration</b>	
SNC Mode	<p>Represents Secure Network Connection which also internally signifies jco.client.snc_mode in SAP. SNC will be enabled if the mode is selected as ON whose value is 1. If SNC is off, the value will be 0.</p>
SNC Level of Security	<p>Represents the quality of protection level (QOP) which is defined from 1 to 9. In SAP, it relates to jco.client.snc_qop.</p> <p>Default: 1</p>

**Table 2—SAP HM/HCM Connector - Configuration parameters**

Parameters	Description
SNC Partner Name	Represents SNC partner. For example, provide input as follows in SAP: p:CN=R3, O=XYZ-INC, C=EN If SNC is configured, it relates to <code>jco.client.snc_partnername</code>
SNC Name	Represent SNC name which internally signifies <code>jco.client.snc_myname</code> . It overrides default SNC Partner Name.
SNC Library	Path to library which provides SNC service. It internally signifies <code>jco.client.snc_lib</code> . For example, the value to be passed: <ul style="list-style-type: none"> <li>• on Microsoft Windows: <code>C:/sapcryptolib/sapcrypto.dll</code> (the location of the cryptographic library)</li> <li>• on UNIX: <code>/opt/sailpoint/lib/custom/libsapcrypto.so</code> (the location of the cryptographic library)</li> </ul>

Note: Attributes marked with \* sign are the mandatory attributes.

## Schema Attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. Account objects are used when building identities Link objects.

### Account attributes

---

**Table 3—SAP HR/HCM Connector - Account Attributes**

Name	Description
Academic Grade	Academic grade attained by the person
Address	Address of the employee
Address Type	Address type of employee; home, work
Address Type Code	Address type code of employee
Admin Group	Administrative group to which the employee belongs
Aristocratic Title	Aristocratic title that apply to this person
Birth Date	Date of birth of employee
Birth Name	Name given to the employee at time of birth
Birth Place	Name or location of birth place of employee

## Schema Attributes

**Table 3—SAP HR/HCM Connector - Account Attributes (Continued)**

Name	Description
Business Area	Business area
City	City in which the employee is located
Co Area	Corporate area
Comp Code	Company code
Company Name	Name of the company by which they are employed
Contract	Work contract: yes, no
Cost Center	Cost center with which they are associated
Country	Country in which employee is located
Country Code	Country code
Country of Birth	Country in which they were born
Country of Birth Code	Country code for country in which they were born
District	District in which they are located or report
E Group	E-mail groups to which they belong
Email	E-mail address
Employee Number	Employee number
FirstName	First name
Form of Address	Form of address; Miss, Mrs., Sir
FullName	Full, legal name
Fund	Fund name.
Funds Center	Funds center name.
Effective Dates	Stores the Effective Dates for manager change and last name change in the following format:  {<lastname schema attr =lastname>#<effective date =yyyy-MM-dd>, <supervisor-id schema attr = supervisor-id>#<effective date= yyyy-MM-dd >}  For example: LastName=Jones#effectiveDate=1999-12-04, Supervisor=00000067#effectiveDate=2016-05-23
Gender	Gender
Gender Code	Gender code
Id Number	Identification number
Initials	Initials

**Table 3—SAP HR/HCM Connector - Account Attributes (Continued)**

Name	Description
Infotype0000	<p>Information about Actions Info Type.</p> <p>The <b>Infotype0000</b> attribute contains action data in the following format:</p> <p>ActionTypeCode=Code of Action to Use(a numeric value),  ActionTypeName=Name of action (a text),  ReasonCode=code of Reason to use(a numeric value),  ReasonTypeText=Name of reason(a text),  ActionStartDate=yyyy-MM-dd, ActionEndDate=yyyy-MM-DD</p> <p><b>Note: Applicable only when upgrading IdentityIQ from version 7.0 Patch 2 to IdentityIQ version 7.1. With this release of IdentityIQ version 7.1, SailPoint recommends the use of newly added 'Infotype0000JSON' schema attribute.</b></p>
Job	Title
Job Description	Description of their function
Known As	Nickname or preferred name
Language	Primary language
Language Code	Language code
Language ISO	Primary language ISO code
LastName	Surname
LegPerson	Legal Person
Marital Status Code	Code associated with the marital status of this person
Marital Status Since	Time period since the last change in marital status
MaritalStatus	Marital status
MiddleName	Middle name
Name	Full name
Name Format Indicator	The formats used for name formatting
Nationality	Nationality
Name State of Birth	Name of the state of birth
Name Third Nationality	Name of the third nationality
Nationality Code	Nationality code
Number of Children	Number of children
Org Key	Organizational key
Org Unit	Organizational unit
Organization Description	Organization description
P subArea	Personal sub area.

## Schema Attributes

**Table 3—SAP HR/HCM Connector - Account Attributes (Continued)**

Name	Description
Payarea	The area from which their pay is received
Payroll Admin	The payroll administrator associated with this person
Personal Admin	The personal administrator associated with this person
Personal Area	The personal area to which employee report
Personal Number	Their personal number
Position	Title
Position Description	Description of job function
Reason Code	Name of the reason code.
Religion	Religion
Religion Code	Religion code
Second Academic Grade	Secondary academic grade associated with this person
Second Address Line	Second line of address
Second Name Prefix	Secondary name prefix
Second Nationality	Secondary nationality
Second Nationality Code	Secondary nationality code
SecondName	Second name
State	State in which they are located
State Abbreviation	State abbreviation of the state in which they are located
State of Birth	State of birth
STAT2_Current	Current employment status
STAT2_Next	Future employment status
STAT2_Next_Start_Date	Effective start date of future employment status
Sub E Group	Sub E group name.
Supervisor	Supervisor area
Surname Prefix	Last name prefix
System user name (SY-UNAME)	User ID
Telephone	Telephone number and dialing code
Third Nationality	Third nationality
Time Admin	Time Administrator name
Title	Function
Zip Code	Zip code for this person

**Table 3—SAP HR/HCM Connector - Account Attributes (Continued)**

Name	Description
Infotype0000JSON	<p>Information about Actions InfoType.</p> <p>The Infotype0000 attribute contains action data in JSON format:</p> <pre>Infotype0000JSON : { "Actions": [ { "ActionTypeCode" : "&lt;Value&gt;","ActionTypeName" : "&lt;Value&gt;","ReasonCode" : "&lt;Value&gt;","ReasonTypeText" : "&lt;Value&gt;","ActionStartDate" : "&lt;Value&gt;","ActionEndDate" : "&lt;Value&gt;" } ] }</pre> <p>The attributes in the above format represent the following:</p> <ul style="list-style-type: none"> <li>• ActionTypeCode = Code of action to use (a numeric value)</li> <li>• ActionTypeName = Name of action (a text),</li> <li>• ReasonCode = Code of reason to use (a numeric value)</li> <li>• ReasonTypeText = Name of reason (a text)</li> <li>• ActionStartDate = YYYY-MM-DD</li> <li>• ActionEndDate = YYYY-MM-DD</li> </ul> <p><b>Note:</b></p> <ul style="list-style-type: none"> <li>- In case there are no values in infotype0000JSON, then the attribute is displayed as Infotype0000JSON: { "Actions": [ ] }</li> <li>- To fetch all action data in JSON format for an user after upgrading IdentityIQ to version 7.1, add the Infotype0000JSON attribute to the application with:</li> </ul> <p><b>Property:</b> Multi-Valued  <b>Data Type:</b> string</p>

## Support for Future Data

SAP HR/HCM connector supports aggregation for the future data for all the accounts.

### *Schema changes*

Following new parameters are added in the IdentityIQ version 7.1 for the future data:

Attribute	Description	Value
STAT2_Current	Current employment status	<b>Inactive:</b> For terminated employee  <b>Active:</b> For active employee
STAT2_Next	Future employment status	<b>Inactive:</b> For terminated employee  <b>Active:</b> For active employee
STAT2_Next_Start_Date	Effective start date of future employment status	Date with format yyyy-MM-dd

## Schema Attributes

Attribute	Description	Value
Effective Dates	Stores the Effective Dates for manager change and last name change in the following format:  {<lastname schema attr =lastname>#<effective date =yyyy-MM-dd>, <supervisor-id schema attr = supervisor-id>#<effective date= supervisor-id>#<effective date= yyyy-MM-dd >}	{<lastname schema attr =lastname>#<effective date =yyyy-MM-dd>, <supervisor-id schema attr = supervisor-id>#<effective date= yyyy-MM-dd >}

**Note:** By default the above attributes will be a part of the schema attributes for the new application. If required for the upgraded application the above attributes must be added manually. If attribute is not present, then no valid event exists for that attribute.

## Delta Aggregation

In IdentityIQ version 7.1, delta aggregation approach is updated. With this approach no additional configuration is required for SAP HR/HCM system.

To enable delta aggregation, user must select the **Enable Delta Aggregation** flag in Account aggregation task.

The supported features for delta aggregation are:

- Any future hires that have been added since the date of last full aggregation.

**Note:** These are the Future Hires within the 'offset'.

- All changes in the **Effective Dates** attribute of the employee are already aggregated in IdentityIQ.
- Any changes for the employment status (STAT2) of the employee already aggregated in IdentityIQ.
- Any changes in following employee attributes:
  - Organization data
  - Communication details
  - Personal data
  - Address information data
  - Supervisor Changes

**Note:** To improve the performance, it is recommended to select the 'Enable Delta Aggregation' flag while running the account aggregation task.

## Upgrade Consideration

Users upgrading to IdentityIQ version 7.1 must update the service account with permissions specified in the "Administrator permissions" section.

## Partitioning

To perform partitioning aggregation, you must enable the **Partition Enabled** option in account aggregation task definition user interface page. The SAP HR/HCM connector itself will determine the number of optimal partitions to be made based on the total number of account and IdentityIQ system wide settings.

**Note:** If you have enabled ‘Partition Enabled’ option, there is a second configuration option presented in the task definition UI -- Objects per partition -- which supports setting the number of accounts to include in each partition. This parameter will not be considered in case of SAP HR/HCM.

## Customization Rule

- **Modify Rule:** The rule name is defined as Example “SAP HRMS Modify Rule”. This is a sample rule to update the existing (non blank values) E-mail, Phone number and System user name.
- **Manager Rule:** The rule name is defined as Example “SAP HR Custom Manager Model Rule”. This example rule can be used as reference to populate the supervisor of the employee.

# Additional information

---

This section describes the additional information related to the SAP HR/HCM Connector.

## Upgraded Application

---

**Note:** For any operation other than Test Connection, the attributes of old application will be updated to new one (upgraded to 7.1). Test Connection would still display the same attributes present in old application.

For the application upgrading to IdentityIQ version 7.1, following changes would be performed:

- Include Terminated Employees Flag and Termination offset

Include Terminated Employees	Termination Offset	Aggregate Inactive Employees	Inactive Employees Offset
Checked		Checked	Blank
Checked	-1	Checked	Blank
Checked	0	Checked	Blank
Not checked	-1	Not checked	Not Applicable

- SearchAdditionalField and SearchAdditionalValue

searchAdditionalField	searchAdditionalStrings	Future Dated Hires Offset
DATE or SEARCH_DATE	+n (any positive integer)	n
DATE or SEARCH_DATE	Any date (for example, 31-12-999)	120
Any value other than DATE or SEARCH_DATE	Not considered	

**Note:** For the upgraded application (IdentityIQ 7.1) partitioning Aggregation will not respect the partitioning string which was present in old application, for more information, see “Partitioning” on page 224.

# Troubleshooting

---

## 1 - Unable to load class files

The following error appears, when unable to load class files:

```
java.lang.UnsatisfiedLinkError: no sapjco3 in java.library.path*
at java.lang.ClassLoader.loadLibrary(ClassLoader.java:1682)
```

### Resolution:

Install the missing DLLs by performing the following steps:

1. Navigate to the following link:  
<http://www.microsoft.com/technet/security/bulletin/MS09-035.mspx>
2. Under the selected security update, scroll to the **Affected Software** section on the right-hand side and click on the **Microsoft Visual C++ 2005 Service Pack 1 Redistributable Package (KB973544)** link.
3. Select the required language and download the platform-specific files.
4. Execute the downloaded file and follow the installation instructions.

For more information, see <https://service.sap.com/sap/support/notes/684106>

**Note:** The URL requires SAP support (SAP username and password).

## 2 - When the supported platform version is Java 1.7 or 1.8 an error message appears

When the supported platform version is Java version 1.7 or 1.8, the following error message appears:

```
getting the version of the native layer: java.lang.UnsatisfiedLinkError: no sapjco3
in java.library.path
```

**Resolution:** Download the latest SAPJCO.jar and SAPJCO.dll files from SAP Marketplace and use that SAPJCO.jar file with the latest downloaded SAPJCO.dll file.

## 3 - Provisioning rules not working for upgraded application

Provisioning rules not working for upgraded application.

**Resolution:** For applications upgraded from IdentityIQ version 6.4 patch 3 and below to IdentityIQ version 7.0 Patch 3, add the following attributes in feature string:

```
PROVISIONING and SYNC_PROVISIONING
```

## 4 - For SAP HR/HCM upgraded application, action data for an employee is not fetched in IdentityIQ

Action data for an user is not getting fetched in IdentityIQ for SAP HR upgraded application.

**Resolution:** To fetch all action data for an user after upgrading IdentityIQ to version 7.0 Patch 2 or above, add the **Infotype0000** attribute to the application with:

- Property: **Multi-Valued**
- Data Type: **string**.

## 5 - While performing Test Connection through SAP router strings, an error message is displayed

While performing Test Connection through SAP router strings, the following error message is displayed:

```
ERROR SPSAPROUTER: route permission denied (xxx.xx.X.X to xxx.xxx.Y.Y, 3300.)
```

**Resolution:** Depending on the error message, (for the above error) add the following entry in the **saprouttab** of your SAProuter:

```
P xxx.xx.X.X xxx.xxx.Y.Y 3300
```

Activate the new saprouttab with the following command or restart the SAProuter:

```
saprouter -n
```

## **Troubleshooting**

# Chapter 24: SailPoint SAP HANA Connector

---

The following topics are discussed in this chapter:

Overview .....	229
Supported features .....	229
Supported Managed Systems .....	230
Pre-requisites .....	230
Administrator permissions .....	230
Configuration parameters .....	231
Schema Attributes .....	232
Provisioning Policy attributes .....	233
Additional information .....	234
Enabling SSL connection to SAP HANA database through IdentityIQ .....	234
Troubleshooting .....	234

## Overview

---

SailPoint SAP HANA Connector manages the users, roles and privileges for the SAP HANA database system.

### Supported features

---

SailPoint SAP HANA Connector supports the following features:

- Account Management
  - Manages SAP HANA database users as Accounts
  - Aggregation, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements
  - Manages System Privileges and Application privileges of user
- Account - Group Management
 

Supports multiple group functionality

  - Manage SAP HANA Catalog Roles as Account group CATALOG\_ROLE
    - Aggregation
  - Manage SAP HANA Repository Roles as Account group REPOSITORY\_ROLE
    - Aggregation

## Overview

**Note:** SAP HANA connector supports single and multi tenant SAP HANA system. User must create separate applications based on the type of Database to be configured. It can be a System database or a Tenant database. For more information, see “Configuration parameters” on page 231.

## Supported Managed Systems

---

SailPoint SAP HANA Connector supports the following SAP System:

- SAP HANA SPS12
- SAP HANA SPS11

## Pre-requisites

---

SAP HANA JDBC driver is required for proper functioning of SailPoint SAP HANA connector. The `ngdbc.jar` file must be copied in the `..\identityiq\WEB-INF\lib` directory of IdentityIQ installation.

## Administrator permissions

---

Following are the minimum required permissions for SAP HANA Administrative account for the listed operation:

Operation	Minimum required permissions
Test Connection	CATALOG role as public
Account Aggregation	<b>System Privileges</b> <ul style="list-style-type: none"><li>• CATALOG Read</li><li>• ROLE ADMIN</li><li>• USER ADMIN</li></ul>
Group Aggregation	<ul style="list-style-type: none"><li>• CATALOG Read</li><li>• ROLE ADMIN</li></ul>
Removing the role assigned to the user	Revoking the role can be achieved by the same user who has granted it.

Operation	Minimum required permissions		
	CATALOG role and Application privilege	Repository roles	System Privileges
Creating a user by assigning the respective privileges and roles	<p>System privileges</p> <ul style="list-style-type: none"> <li>• CATALOG Read</li> <li>• ROLE ADMIN</li> </ul>	<p><b>System Privileges</b></p> <ul style="list-style-type: none"> <li>• CATALOG Read</li> <li>• ROLE ADMIN</li> <li>• USER ADMIN</li> </ul> <p><b>Object Privileges</b></p> <p>Execute Privilege on the following objects:</p> <ul style="list-style-type: none"> <li>• _SYS_REPO.GRANT_ACTIVATED_ROLE</li> <li>• _SYS_REPO.REVOKE_ACTIVATED_ROLE</li> <li>• _SYS_REPO.GRANT_APPLICATION_PRIVILEGE</li> <li>• _SYS_REPO.REVOKE_APPLICATION_PRIVILEGE</li> </ul>	<p>Service account must have the system privilege assigned to it with Grantable to other user and groups flag as checked.</p>

Note: For Enable/Disable/Lock/Unlock Account/Delete Account/change password only System privileges USER ADMIN is required.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The SAP HANA Connector uses the following connection attributes:

Attributes	Description
User Name*	Enter name of the user.
Password*	Enter the password to connect to SAP HANA Database.

## Schema Attributes

Attributes	Description
Database URL*	<p>Enter the URL of the database. The specified default URL is provided for multi-tenant HANA Database system. For a single tenant HANA Database system, attribute database name is not applicable.</p> <p><b>Default:</b> <code>jdbc:sap://&lt;HOST&gt;:&lt;PORT&gt;/?databaseName=&lt;dbName&gt;</code></p> <p>where:</p> <ul style="list-style-type: none"><li>• &lt;HOST&gt;: server name where SAP HANA database is installed</li><li>• &lt;PORT&gt;: port on which database is configured</li><li>• &lt;dbName&gt;: name of the database to connect to</li></ul> <p><b>Note:</b> Enter URL in the following format to connect to the SAP HANA SSL enabled database: <code>jdbc:sap://&lt;HOST&gt;:&lt;PORT&gt;/?databaseName=&lt;dbName&gt;&amp;encrypt=true</code></p> <p>For more information on enabling SSL Connection to SAP HANA database through Identity IQ, see “Enabling SSL connection to SAP HANA database through IdentityIQ” on page 234.</p>
Driver Class*	The driver class used for connecting to SAP HANA Database. <b>Default:</b> com.sap.db.jdbc.Driver
Page Size	The number of objects to fetch in a single page when iterating over large data sets. <b>Default:</b> 1000

## Schema Attributes

The application schema is used to configure the objects returned from a connector. When application is configured, the schema is supplied to the methods on the connector interface and supports multiple types of objects, account and any number of group application object types. Account objects are used when building identities Link objects. Additional schema definitions can be used when building AccountGroup objects which are used to hold entitlements shared across identities.

## Account attributes

The following table lists the account attributes:

Attributes	Description
USER_ID	ID of the user
USER_NAME	Name of the user
USER_MODE	Mode of the user: <ul style="list-style-type: none"><li>• LOCAL</li><li>• GLOBAL</li><li>• EXTERNAL</li></ul>
CREATOR	Creator of the user

Attributes	Description
VALID_FROM	Specify a date time from which the user account is valid
VALID_UNTIL	Specify a date time until which the user account is valid
IS_RESTRICTED	Specifies if the user is a restricted user
IS_CLIENT_CONNECT_ENABLED	Specifies if the user is able to connect to client
HAS_REMOTE_USERS	Specifies if there is a database user in another tenant database as the remote identity of the database user.
SESSION_CLIENT	Specifies the client whose data can be accessed by user
EMAIL_ADDRESS	Email address of the user.
TIME_ZONE	Time zone of the user.
AUTHENTICATION_TYPE	Different Authentication methods supported by user.
SYSTEM_PRIVILEGES	System Privileges assigned to the user.
APPLICATION_PRIVILEGES	Application Privileges assigned to the user.
CATALOG_ROLES	Catalog Roles assigned to the user.
REPOSITORY_ROLES	Repository Roles assigned to the user.
FORCE_PWD_CHG_ON_NEXT_LOGON	Specifies whether user must change the password on next logon in the SAP HANA database.

## Group attributes

---

The following table lists the group attributes for REPOSITORY\_ROLE and CATALOG\_ROLE:

Attributes	Description
ROLE_NAME	Role name
ROLE_ID	Role ID
ROLE_MODE	Mode of the role: LOCAL
GLOBAL_IDENTITY	Identity specified for role with ROLE_MODE GLOBAL
CREATOR	Name of the user who created the role
GRANTED_ROLES	The roles which are assigned to the current role

## Provisioning Policy attributes

---

This section lists the different policy attributes of SAP HANA Connector.

**Note:** The attributes marked with \* sign are the required attributes.

## Additional information

### Create account attributes

---

The following table lists the provisioning policy attributes for Create Account:

Attributes	Description
User Name	User name of SAP HANA database.
Password	Password of the user.
Force Password Change On Next Logon	Set the value to <b>Yes</b> , if user must be requested to change his password on next login.  <b>Note:</b> If 'Force Password Change On Next Logon' flag is not specified in the provisioning plan, then password mode (permanent or temporarily) would be depended on SAP HANA security policy.
Restricted	Select <b>true</b> if user must be created in Restricted mode.

## Additional information

---

This section describes the additional information related to the SAP HANA Connector.

### Enabling SSL connection to SAP HANA database through IdentityIQ

---

Perform the following steps to enable SSL Connection to SAP HANA database through IdentityIQ:

1. Import the SSL certificates from the SAP HANA Server to IdentityIQ system.
2. Add the certificates to the IdentityIQ keystore present in the <JAVA\_HOME>/jre/lib/security/cacerts directory.
3. Restart the WEB Server.
4. Login to IdentityIQ.
5. For database, use the following URL:  
jdbc:sap://<HOST>:<PORT>/?databaseName=<dbName>&encrypt=true

## Troubleshooting

---

### 1 - An error message was displayed while removing the role assigned to the user

Following error message was displayed while removing the role assigned to the user:

Service account does not have permission to revoke <Role>

**Resolution:** Revoking the role can be achieved by the same user who has granted it.

## 2 - Test connection fails with an error message

Test connection fails with the following error message:

Please enter valid Driver Class

**Resolution:** Copy the latest ngdbc.jar file in \identityiq\WEB-INF\lib directory.

## **Troubleshooting**

# Chapter 25: SailPoint System for Cross-Domain Identity Management Connector 2.0

---

The following topics are discussed in this chapter:

Overview .....	237
Supported features .....	237
Administrator permissions .....	238
Supported Managed System .....	238
Configuration parameters .....	238
Schema attributes .....	238
Provisioning Policy attributes .....	239
Create account attributes .....	239
Provisioning of extended attributes .....	239
Troubleshooting .....	240

## Overview

---

The SCIM (System for Cross-Domain Identity Management) standard defines a schema and API to create, read, update, and delete identity and identity-related information on other systems. This standard creates a common language, by which a client system can communicate with many different servers in the same way. SaaS providers (such as Salesforce) and other software vendors are beginning to adopt this standard, and are exposing their identity management interfaces through SCIM.

## Supported features

---

SailPoint SCIM 2 Connector supports the following features:

- Account Management
  - Aggregation, Discover Schema
  - Create, Update, Delete
  - Enable, Disable
  - Change Password
  - Add/Remove Entitlements

**Note:** **Add/Remove Entitlement would work if entitlements, groups and roles are a part of SCIM 2 core schema. For more information on SCIM 2 core schema, see <https://tools.ietf.org/html/rfc7643>.**
- Account - Group Management
  - Aggregation, Discover Schema

## Administrator permissions

---

The required administrator permissions depends on which SCIM 2 server is being connected to. For example, IdentityIQ as SCIM 2 server has SCIM 2 executor capabilities which has minimal permission required for operations to be executed through SCIM 2 Connector.

## Supported Managed System

---

SailPoint SCIM2 Connector supports SCIM servers which are compliant to SCIM 2 protocol.

# Configuration parameters

---

The following table lists the configuration parameters of SCIM Connector:

Parameters	Description
Base URL*	The base URL to connect to the SCIM 2 server.
Authentication Type*	Select one of the following method of authentication to the SCIM 2 server: <ul style="list-style-type: none"><li>• Basic (username and password)</li><li>• OAuth2</li></ul>
Username*	Username for the SCIM 2 Server. <b>Note: Required if the 'Authentication Type' is selected as 'Basic'.</b>
Password*	Password for the SCIM 2 Server. <b>Note: Required if the 'Authentication Type' is selected as 'Basic'.</b>
OAuth2 Token*	The OAuth2 bearer token to be used for authorization. <b>Note: Required if the 'Authentication Type' is selected as 'OAuth2'.</b>
Explicit Attribute Request	Select to create request for fetching required attributes in request. <b>Note: If selected, Connector will request for attributes present in schema else will request for all attributes.</b>

**Note:** Attributes marked with \* sign are the mandatory attributes.

## Schema attributes

---

The application schema is used to configure the objects returned from connector. The following types of objects (account, group, entitlements, and roles) are supported:

- Account object type is mapped to SCIM 2 server User resource.
- Group object type is mapped to SCIM 2 server Group resource.
- Entitlements object type is mapped to SCIM 2 server Entitlement resource.
- Roles object type is mapped to SCIM 2 server Role resource.

**Note:** Discover schema populates schema attribute values for supported object type. The newly added extended schema attributes on SCIM 2 Server can be obtained into IdentityIQ schema by clicking on the Discover Schema option of the respective objectType. For example, to obtain the ‘test’ extended attribute on SCIM 2 Server into IdentityIQ schema, click on the account object type Discover Schema option.

## Provisioning Policy attributes

---

This section lists the different policy attributes of SCIM 2 Connector.

**Note:** The attributes marked with \* sign are the required attributes.

### Create account attributes

---

The following table lists the provisioning policy attributes for Create Account:

Attributes	Description
User Name	A service provider's unique identifier for the user, used by the user to directly authenticate to the service provider.
Formatted Name	The full name, including all middle names, titles and suffixes as appropriate, formatted for display.
Family Name	The family name of the user or last name.
Given Name	The given name of the user or first name
Display Name	The name of the user suitable for display to end-users.
Email	Email addresses for the user.

### Update account attributes

---

Update account provisioning policy varies according to SCIM 2 managed system. To update any attribute, add account schema attributes name as it is in update provisioning policy.

#### Pre-requisite:

Execute **Discover Schema** to obtain attributes present on managed system after creating application for accounts in IdentityIQ.

### Provisioning of extended attributes

---

For provisioning of extended attributes write a Before Provisioning rule to modify the Provisioning Plan and prepare a AttributeRequest that includes only the right value (not the full JSON).

Following is an example of before provisioning rule for updating extended complex schema attributes when managed system is supporting HTTP PATCH method:

```
import com.google.gson.Gson;
import com.google.gson.JsonObject;
import com.google.gson.JsonParser;
```

## Troubleshooting

```
import com.google.gson.reflect.TypeToken;
import org.codehaus.jackson.map.ObjectMapper;
import sailpoint.object.Application;
import sailpoint.object.ProvisioningPlan;
import sailpoint.object.ProvisioningPlan.AttributeRequest;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

try {
    AttributeRequest attrReq =
plan.getAccountRequest(application.getName()).getAttributeRequest("manager");
    String valueList = attrReq.getValue().toString();
    // Here you get a value in JSON format, fetch your interesting value from the JSON
    // Parser Logic goes here.
    ObjectMapper mapper = new ObjectMapper();
    Map map = mapper.readValue(valueList, Map.class);
    String value = (String) map.get("value");
    // Prepare the right AttributeRequest
    attrReq.setName("manager.value");
    attrReq.setValue(value);
} catch (Exception e) {
    e.printStackTrace();
}
```

# Troubleshooting

---

## 1 - Create account status is pending

During create account operation, the create account task action status remains in pending state.

**Resolution:** Perform the following:

1. Add **active** attribute in Create Account Provisioning Policy.
2. Perform create account operation.
3. Run **Perform Identity Request Maintenance** task.

## 2 - While aggregation an error message is displayed

During aggregation the following error message is displayed:

ResourceObject is returned with null identity error

**Resolution:** Verify if the identity attribute is present in the schema attribute else add the appropriate schema attribute name as identity attribute.

# Chapter 26: SailPoint ServiceNow Connector

---

The following topics are discussed in this chapter:

Overview .....	241
Supported features .....	241
Supported Managed System .....	242
Pre-requisites .....	242
User permissions .....	243
Configuration parameters .....	243
Schema attributes .....	244
Account attributes .....	245
Group attributes .....	246
Role attributes .....	247
Provisioning Policy attributes .....	248
Additional information .....	249
Upgrade .....	249
Session management .....	249
Troubleshooting .....	250

## Overview

---

The SailPoint ServiceNow Connector manages ServiceNow accounts, groups, and roles. It supports read and write for ServiceNow accounts and groups.

### Supported features

---

SailPoint ServiceNow Connector supports the following features:

- Account Management
  - Manages ServiceNow Users as Accounts
  - Aggregation, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements (ServiceNow Groups and ServiceNow Roles)
- Account - Group Management
  - Manages ServiceNow Groups and Roles as Account-Groups
  - Aggregation, Refresh Groups
  - Create, Update, Delete (applicable for groups only)

## Overview

- ServiceNow Connector supports configuration of multiple applications of different ServiceNow versions on same IdentityIQ.

## Supported Managed System

---

SailPoint ServiceNow Connector supports the following ServiceNow versions:

- Istanbul
- Helsinki
- Geneva
- Fuji

ServiceNow Rest API supports Basic and OAuth2 methods of authentication.

Each client must perform the OAuth setup to participate in OAuth authorization. To configure OAuth in ServiceNow Connector, a Client ID, Client Secret and Refresh Token are required. The Client ID, Client Secret and Refresh Token are specific to the ServiceNow instance and configured while enabling the OAuth in ServiceNow instance. Contact your ServiceNow Administrator to obtain the Client ID, Client Secret and Refresh Token.

Refer to the following link for token generation:

[http://wiki.servicenow.com/index.php?title=Generating\\_OAuth\\_Tokens](http://wiki.servicenow.com/index.php?title=Generating_OAuth_Tokens)

## Pre-requisites

---

- ServiceNow should be up and running.
- Apply the ServiceNow Connector update set as follows:
  1. Copy the relevant update set from `identityiq-releaseVersion.zip\integration\servicenow\iiqIntegration-Servicenow.zip\ConnectorUpdateSet`  
In the above directory, *releaseVersion* is the version of the current IdentityIQ release.
  2. Based on the required version of ServiceNow, copy the relevant update set from the following respective files:

ServiceNow version	Update Sets
Geneva or later	<code>SailPointServiceNowConnector.v1.1.xml</code>
Fuji	<code>SailPointServiceNowConnector.v1.1_fuji.xml</code>

3. Import relevant update set in ServiceNow instance. For more information and guidelines on usage of the update set, refer to the following wiki link:  
[http://wiki.servicenow.com/index.php?title=Saving\\_Customizations\\_in\\_a\\_Single\\_XML\\_File#gsc.tab=0](http://wiki.servicenow.com/index.php?title=Saving_Customizations_in_a_Single_XML_File#gsc.tab=0)  
**For Service Now Fuji or later instance**
  - a. Set system (target) table application access after import, preview and commit of update set.

For the connector to work smoothly, ensure that all the access (read, create, update, delete and allow access to the following tables via web services) has been provided.

- **User [sys\_user]**
- **Group [sys\_user\_group]**
- **Group Member [sys\_user\_grmember]**
- **User Role [sys\_user\_has\_role]**
- **Group Role [sys\_group\_has\_role]**

Perform the following to provide application access:

- Ensure that Global scope is selected in ServiceNow.
  - Navigate to **System Definition => Tables**.
  - Search for the table using label or name.
  - Click on table and scroll down to **Application Access**.
  - Select Can read, Can create, Can update, Can delete and Allow Access to this table via web services.
  - Update or Save the table.
- b. To support unlock operation in ServiceNow Fuji or later, create the following ACL in global scope and assign it to the `x_sapo_iiq_connect.admin` Role:

ACL	Type	Operation	Name	Attribute
<code>sys_user.locked_out</code>	record	read	User [sys_user]	Locked out

For more information on procedure for creating the ACL, see the following link:

[http://wiki.servicenow.com/index.php?title=Using\\_Access\\_Control\\_Rules#Creating\\_ACL\\_Rules](http://wiki.servicenow.com/index.php?title=Using_Access_Control_Rules#Creating_ACL_Rules)

- To configure any custom field the import set and transform map must be updated with the custom field.

## User permissions

---

Assign the `x_sapo_iiq_connect.admin` role to the user when using ServiceNow instance.

## Configuration parameters

---

This section contains the information that is used to connect and interact with the application. Each application type requires different information to create and maintain a connection.

## Schema attributes

The ServiceNow Connector uses the connection attributes listed in the following table:

Parameters	Description
Authentication Type	<ul style="list-style-type: none"><li><b>Basic:</b> In case of Basic, Username/Password will be used for authentication.</li><li><b>OAuth2:</b> Select this option when ServiceNow is configured to support OAuth2 authentication.</li></ul> <p><b>Note:</b> Each time the Authentication Type for ServiceNow Connector is changed, ensure that you perform Test Connection operation.</p>
UserName	Name of the user having the privileges mentioned in the “User permissions” section above.
Password	Password of the user having minimum privileges.
Client ID	The Client ID for OAuth2 authentication.
Client Secret	The Client Secret for OAuth2 authentication.
Refresh Token	The Refresh Token for OAuth2 authentication.
Page Size	The Page size specifies the maximum size of each data set when querying large number of objects. Its default value is set to 1000 and maximum value be 10000.
Filter Condition for Account	(Optional) To filter accounts during aggregation. For example, active=true^locked_out=false
Filter Condition for Group	(Optional) To filter groups during aggregation. For example, active=true
Filter Condition for Role	(Optional) To filter roles during aggregation. For example, sys_scope=Global

## Schema attributes

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. IdentityIQ currently supports the following types of objects:

- Account:** Account objects are used when building identities Link objects.
- Group:** The group schema is used when building Account-Group objects that are used to hold entitlements shared across identities.
- Role:** The role schema is used when building roles as Account-Group objects.

**Note:** - For account aggregation, ServiceNow Role aggregation is supported only for roles having the following property:  
inherited=false  
- For Account-Group aggregation, ServiceNow Role aggregation is supported only for direct roles connected to the group.

## Account attributes

---

The following table lists the account attributes ([Table 1—Account attributes](#)):

**Table 1—Account attributes**

Attributes	Description
first_name	First name of the user.
last_name	Last name of the user.
email	Email ID of the user.
user_name	Name of the user.
department	The user's department name.
title	Title (designation) of the user.
sys_id	Unique ID generated by system for user.
phone	Phone number of user.
calendar_integration	Determines whether change requests assigned to that user are sent to their Outlook calendar.
sys_class_name	Class name of the user.
company	The user's company.
cost_center	Cost center of the user.
sys_created_on	Date this user is created in ServiceNow.
sys_created_by	Administrator who created the user in ServiceNow.
groups	List of groups the user is part of.
roles	List of roles the user is part of.
active	Determines whether the user account has been staged for use.
building	The building of the user.
city	The city of the user.
country	The country of the user.
location	The location of the user.
manager	The manager of the user.
middle_name	Middle name of the user.
name	Name of the user.
password_needs_reset	Determines should the user be prompted to change password at next login.
default_perspective	Default perspective for the user.
sys_domain	Domain of the user.
employee_number	Employee number of the user.
failed_attempts	Number of login failed attempts.

## Schema attributes

**Table 1—Account attributes (Continued)**

Attributes	Description
gender	Gender of the user.
home_phone	Home phone number of the user.
ldap_server	LDAP server the user has an account. Identifies which LDAP server authenticates the user when there are multiple LDAP servers.
preferred_language	Language spoken by the user.
last_login	Last login date of the user.
last_login_time	Time of the last login time for the user.
locked_out	Determines if user account is locked.
mobile_phone	Mobile number of the user.
notification	Determines if the user should be notified for any changes made on his account.
schedule	Schedule of the user.
state	The state for the user.
source	Identifies whether LDAP is used to validate a user. If the Source field starts with <b>ldap</b> , then the user is validated via LDAP. If the Source field does not start with <b>ldap</b> , then the password on the user record is used to validate the user upon login.
street	The street for the user.
time_format	Time format selected for user to display time fields.
time_zone	The timezone for the user.
sys_updated_on	Last updated time for the user.
sys_updated_by	The last update for the user occurred from.
sys_mod_count	Number of updates for the user.
vip	Determines if the user is treated as VIP.
zip	Zip for the user.

## Group attributes

---

The following table lists the group attributes ([Table 2—Group attributes](#)):

**Table 2—Group attributes**

Attributes	Description
active	Determines whether the user account has been staged for use.
cost_center	Cost center of the user group.
sys_created_on	Date the user group is created in ServiceNow.
sys_created_by	Administrator who created the user in ServiceNow.

**Table 2—Group attributes**

Attributes	Description
default_assignee	Defaults assignee for the user group.
description	Description of the user group
exclude_manager	Determines if the manager should be excluded for the user group.
name	Name of the user group.
parent	Parent group of this user group.
roles	Roles the user group is having.
source	Source of the user group.
sys_id	Unique ID generated by system for user group.
type	Type of the user group
sys_updated_on	Last updated time for the user group.
sys_updated_by	The last update for the user group occurred from.
sys_mod_count	Number of updates for the user group.

## Role attributes

---

The following table lists the role attributes ([Table 3—Role attributes](#)):

**Table 3—Role attributes**

Attributes	Description
sys_name	System name of the Role.
sys_updated_on	Last updated.
sys_id	Unique ID generated by system for role.
grantable	Can be granted independently.
sys_created_on	Created date and time.
suffix	Application scope.
sys_created_by	Created by.
can_delegate	Can be delegated.
sys_policy	Determines how application files are protected when downloaded or installed.
sys_updated_by	Updated by.
sys_tags	Tags
sys_package	Application name.
description	Description of the role.
name	Name of the role.
sys_class_name	Class name of the role.

## Provisioning Policy attributes

**Table 3—Role attributes**

Attributes	Description
sys_update_name	System updated name.
elevated_privilege	This role is an elevated privilege.
sys_mod_count	Number of updates for the role.
sys_customer_update	Added or modified by customer.
sys_scope	Scope name.
includes_roles	Includes roles.
contains_roles	Contained roles.

## Provisioning Policy attributes

---

This following table lists the provisioning policy attributes for create ([Table 4—Provisioning Policy attributes](#)):

**Table 4—Provisioning Policy attributes**

Attributes	Description
UserID	User ID for the user.
FirstName	First name of user.
LastName	Last name of the user.
Department	The user's department name.
Title	Title (designation) of the user
Password	Password for the user.
Password need reset	Determines if the user must be prompted to change the password at next login.
Locked Out	Determines if user account is locked.
Active	Determines whether the user account has been staged for use.
Notifications	Determines if the user should be notified for any changes made on his account.
Calender integration	Determines whether change requests assigned to that user are sent to their Outlook calendar.
Time Zone	The time zone for the user.
Email	Email of the user.
Mobile Phone	Mobile number of the user.
Business Phone	Official phone of the user.

## Additional information

---

This section describes the additional information related to the ServiceNow Connector.

**Note:** **To enable logging, specify the logging**

```
log4j.logger.openconnector.connector.servicenow.ServiceNowConnector in the
log4j.properties file. For example,
log4j.logger.openconnector.connector.servicenow.ServiceNowConnector=debu
g.
```

- ServiceNow Connector uses the following Transform Maps for write operations:
  - SailpointSysUserHasRole\_D
  - SailpointSysGroupHasRole\_D
  - SailPointSysUserHasRole
  - SailPointSysGroupHasRole
  - SailpointSysUserGrmember\_D
  - SailpointSysUserGroup\_D
  - SailPointSysUserGrmember
  - SailpointSysUser\_D
  - SailPointSysUsers
  - SailPointSysUserGroup

**Note:** “\_D” suffix entities used for delete operation only.

- The user can configure the connector to use any of the attributes of ServiceNow User/Group/Role which are supported by ServiceNow Rest APIs.

## Upgrade

---

For upgrading ServiceNow Connector from version 7.0 to version 7.1, ServiceNow Connector update set must be applied on ServiceNow instance.

**Note:** If the following error message is displayed while previewing the update set, then accept the remote update (that is, overwrite with the change in the update set) followed by committing it:

Preview problems for IdentityIQ ServiceNow Connector: 463 Errors | 0 Warnings. To commit this update set you must address all problems

For more information, see “Pre-requisites” on page 242.

## Session management

---

A REST session is a Glide session established with a ServiceNow instance by any external REST client like SailPoint ServiceNow Connector. It was observed that for every request the connector would open one session which resulted in opening number of sessions on ServiceNow. With this release of SailPoint ServiceNow Connector, the

## Troubleshooting

connector would maintain a pool of sessions (using the following parameters) which would be reused for subsequent operations:

- **sessionPoolSize**: defines how many maximum sessions can be opened on the ServiceNow. This parameter can be set in application template using the debug page. Default: 10.
- **sessionRetryCounter**: defines how many times IdentityIQ should try to get free session from sessionPoolSize. This parameter can be set in application template using the debug page. Default: 10.

### Behavioral change

Following are the behavioral changes observed with increase or decrease of the **sessionPoolSize** or **sessionRetryCounter** parameters:

- If the value of the **sessionPoolSize** or **sessionRetryCounter** parameter has been increased, to reflect the changes user has to perform test connection. The following examples would set the **sessionPoolSize** or **sessionRetryCounter** to 15.

```
<entry key="sessionPoolSize" value="15" />
<entry key="sessionRetryCounter" value="15" />
```

**Note:** If the value of **sessionPoolSize** parameter is increased ServiceNow Connector will open new set of sessions. The sessions which was opened in ServiceNow would be closed after timeout in ServiceNow.

- If the value of **sessionPoolSize** parameter is decreased (for example from 10 to 5) the value will be effective only after restarting the Server.
- If the value of the **sessionRetryCounter** parameter has been decreased, to reflect the changes user has to perform test connection.

**Note:** ServiceNow Connector creates a new set of sessions on Test Connection.

**Note:** (*Factors to be considered while defining the session pool size*) The number of session pool size required to handle the ServiceNow Connector requests depends on the number of clients requests at a time, load on IdentityIQ and ability of ServiceNow instance to work with those sessions.

## Troubleshooting

---

### 1 - If a record is not created or updated on ServiceNow and there are no errors displayed

**Resolution:** Verify the record in relevant import set table listed in “Additional information” on page 249.

### 2 - Account entity has ‘groups and roles’ field which displays all the groups and roles the user is a part of

The Account entity has groups and roles fields which display all the groups and roles the user is a part of; the Account aggregation only displays the sys id of those groups and roles. After aggregation the account details display some alpha numeric strings in the **groups** and **roles** field.

**Workaround:** In order to get the group name, the account-group aggregation must be executed, which will replace the sys id from the corresponding group name in the **groups** field and role name in the **roles** field.

### 3 - Test Connection failed with message: "null". Please check the connection details

- The error is displayed when trying to create ServiceNow application for ServiceNow Calgary or Dublin instance.  
**Resolution:** Test connection fails as supported versions of ServiceNow are Helsinki, Geneva, and Fuji.
- When application created for ServiceNow on IdentityIQ version 7.0 contains url in the following format and after upgrading IdentityIQ to version 7.1 the above error message appears while performing test connection:  
<https://demo.service-now.com/navpage.do> and after upgrading IdentityIQ to 7.1, above error occurs while performing test connection  
**Resolution:** Change the url format to <https://demo.service-now.com> and perform test connection.

### 4 - Error message is displayed in log

- The error means that ServiceNow connector retried to get free session from the session pool but does not get it after number of retries mentioned by *sessionRetryCounter*. The following error message is displayed in the log when ServiceNow Connector tries to get free sessions from the session pool as mentioned in the *sessionRetryCounter*:  
No session information available. Please increase session pool size.  
**Resolution:** Due to high load the sessions are too busy and occupied hence increase the session pool size by using the following entry so that more concurrent operations can be performed.  

```
<entry key="sessionPoolSize" value="15" />
```
- Old sessions still valid and reused  
For example:  
Suppose S1 and S2 are the sessions opened by ServiceNow admin1 defined in ServiceNow Connector as administrator. User changes the username from admin1 to admin2. Now, the sessions S1 and S2 are still valid and can be used till it gets timeout on ServiceNow regardless of the user information is changed in IdentityIQ. In this case for ServiceNow administrator admin1 is still performing operations from IdentityIQ whereas customer would expect admin2 to perform operations from IdentityIQ. The activities will be logged against admin1 whereas, IdentityIQ should use admin2 to perform the operations.  
**Resolution:** Any kind of changes in application should be followed by the test connection operation, otherwise old sessions are still valid and will be reused.
- Reducing value of **sessionPoolSize** entry not been honored until the server is restarted.  
**Resolution:** If the values are increased then user must save application and perform test connection so that the changes are effective.

**Note:** The recommended sessionPoolSize is 10 so that IdentityIQ can process 10 requests at a time.

```
<entry key="sessionPoolSize" value="10" />
```

### 5 - Test Connection failed with an error message

When user does not have appropriate permissions to perform the test connection, the following error message appears:

"HTTP/1.1 403 Forbidden". Please check the connection details

**Resolution:** Verify if the user has permissions to perform test connection. For more information, see "User permissions" on page 243.

### 6 - openconnector.ConnectorException: Disable failed. HTTP/1.1 400 Bad Request

When update set is not present on ServiceNow instance, the following error message is displayed:

## Troubleshooting

openconnector.ConnectorException: Disable failed. HTTP/1.1 400 Bad Request

**Resolution:** Perform the following:

1. Upload the update set on ServiceNow instance and assign the appropriate role to the user.  
For more information, see “User permissions” on page 243.
2. Ensure that proper version is selected on Application Configuration Settings Page.

### 7 - Unable to fetch next block of account. Exception occurred during account aggregation. Transaction canceled: maximum execution time exceeded

ServiceNow prevents inbound REST request running for longer than 60 seconds starting with Fuji release.

**Resolution:** Increase the value of **Maximum Duration** field on Transaction Quota Rule - REST request timeout using the following steps:

1. Navigate to **System Definition => Quota Rules**.
2. Select REST request timeout and increase value of **Maximum Duration** (seconds) field as per requirement.

### 8 - Error message appears in IdentityIQ when deleting the ServiceNow entitlement from Identity

The following error message appears in IdentityIQ when deleting the ServiceNow entitlement from Identity:

```
Remove role 'xxx' failed for user 'abc' with message: Exception in  
removeItem() with error message HTTP/1.1 403 Forbidden
```

**Resolution:** Perform the following:

1. Clear import set table records.

**For example,**

- If the error message is displayed during the addition of the entitlements for identity, then clear records from the **SailPointSysUserHasRole** import set table.
  - If the error message is displayed during the deletion of the entitlements for identity, then clear records from the **SailPointSysUserHasRole\_D** import set table.
2. If a performance degradation is observed in provisioning operation, then use the following ServiceNow link to troubleshoot and improve the performance of the import set jobs:  
[http://wiki.servicenow.com/index.php?title=Troubleshooting\\_Import\\_Set\\_Performance](http://wiki.servicenow.com/index.php?title=Troubleshooting_Import_Set_Performance)

# Chapter 27: SailPoint Siebel Connector

---

The following topics are discussed in this chapter:

Overview .....	253
Supported features .....	253
Supported Managed Systems .....	254
Pre-requisites .....	254
Administrator permission .....	254
Configuration parameters .....	254
Schema attributes .....	255
Account attributes .....	256
Account Group attributes .....	256
Adding new custom attributes in schema .....	257
Provisioning policy attributes .....	257
Troubleshooting .....	258

## Overview

---

The SailPoint Siebel Connector manages entities in Oracle's Siebel CRM. Here **Employee** is managed as Accounts and **Position** as Account Groups. By default, the Siebel Connector uses the Employee Siebel business component of the Employee Siebel business object for account provisioning. For Account Group provisioning Position business component of Position business object is used by Connector. However, the Connector can be configured to manage other Siebel Business Object/Component in the Account/Account Group provisioning. The Connector manages both single and multi-valued attributes of Siebel system. The Connector schema can be modified to manage attributes other than Schema that comes by default with Connector.

## Supported features

---

SailPoint Siebel Connector provides support for the following features:

- Account Management
  - Manages Employee as Accounts
  - Aggregation, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable, Change Password
  - Add/Remove Entitlements

**Note:** **Enable Account** operation sets the Employment Status attribute to Active while it is set to Terminated for Disable Account operation.

## Configuration parameters

- Account - Group Management
  - Manages Position as Account-Groups
  - Aggregation, Refresh Groups
  - Create, Update, Delete

## Supported Managed Systems

---

SailPoint Siebel connector supports Siebel CRM version 8.2 Managed System.

## Pre-requisites

---

Following Siebel JAR files are required in the WEB-INF/lib directory:

- **Siebel 8.2:** Siebel.jar and SiebelJI\_<<Language>>.jar

For example, for Siebel CRM 8.2 with English language: Siebel.jar, SiebelJI\_enu.jar

The Siebel JAR files are available in the SIEBEL\_INSTALLATION\_DIRECTORY/siebsrvr/CLASSES directory.

**Note:** **Do not copy JAR files for multiple versions of Siebel into the WEB-INF/lib directory; it may create conflicts at runtime.**

**Note:** **Siebel Connector requires JRE 1.6 to manage Siebel CRM 8.2.**

## Administrator permission

---

The Siebel Connector requires Siebel administrator credentials to accomplish provisioning tasks. The administrator user name and password configured for the Siebel connector must be assigned sufficient privileges within Siebel to create new records and to update existing records for the specified business component.

For example, SADMIN user which is created during Siebel server installation is one of the example of administrator.

**Note:** **A responsibility named “Siebel Administrator” assigned to this user gives access to all views.**

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Siebel Connector uses the connection parameters listed in the following table:

**Table 1—Configuration parameters**

Parameter	Description
Transport Protocol	Transport protocol while communicating with Siebel server. Select TCPIP or NONE. Default: TCPIP
Encryption	Data Encryption method. Select RSA or NONE. Default: NONE
Compression	Data Compression technique. Select ZLIB or NONE. Default: ZLIB

**Table 1—Configuration parameters**

Parameter	Description
Siebel Server Host	Host Name where Siebel server is installed.
SCB Port	Listening port number for the Siebel Connection Broker (alias SCBroker). Sample value: 2321
Siebel Enterprise Name	Name of Siebel Enterprise. Sample value: SBA_82
Siebel Object Manager	Name of Siebel Application Object Manager. Sample value: SCCObjMgr
Admin User Name	User ID of the target system user account that you want to use for connector operations. Sample value: SADMIN
Password	Password of the target system user account that you want to use for connector operations. Sample value: sadmin
Language	Language in which the text on the UI is displayed. Specify any one of the following values: <ul style="list-style-type: none"> <li>• For English: ENU</li> <li>• For Brazilian Portuguese: PTB</li> <li>• For French: FRA</li> <li>• For German: DEU</li> <li>• For Italian: ITA</li> <li>• For Japanese: JPN</li> <li>• For Korean: KOR</li> <li>• For Simplified Chinese: CHS</li> <li>• For Spanish: ESP</li> <li>• For Traditional Chinese: CHT</li> </ul>
Account Business Object	Business Object for Account. Default value: Employee
Account Business Component	Business Component for Account. Default value: Employee
Entitlement Business Object	Business Object for Entitlement. Default value: Position
Entitlement Business Component	Business Component for Entitlement. Default value: Position
Siebel URL	Siebel server connection string. The server is connected using connection string. Specific parameters defined in the form are ignored. For example: <code>siebel.transport.encryption.compression:/host:port/EnterpriseServer/AppObjMgr_lang" lang="lang_code"</code>

## Schema attributes

---

By default the following mentioned set of attributes are managed:

## Schema attributes

### Account attributes

---

The following table lists the account attributes (Siebel **Employee** attributes):

Attributes	Description
Login Name	Employee's login name.
First Name	Employee's first name.
Last Name	Employee's Last name.
Position	Multi-value attribute that contains a list of all positions assigned to employee.
Primary Position	Employee's primary position.
Responsibility	Multi-value attribute that contains a list of all responsibilities of employee.
Primary Responsibility Id	Employee's Primary responsibility ID.
Division	Division
Employment Status	Employment Status
Street Address	Street Address
Job Title	Job Title
Phone Number	Phone Number
Fax Number	Fax Number
Hire Date	Hire date
Alias	Alias
State	State
Availability Status	Availability status of employee.
ManagerLogin	Employee's Manager login.

### Account Group attributes

---

The following table lists the Account Group attributes (Siebel **Position** attributes):

Attributes	Description
Id	Unique Id for Position Entity.
Name	Name of Position.
Last Name	Last Name of Employees having this Position.
Division	Division of Position.
Role	Role
Start Date	Start date for allocation of Position to Employee referred by Last Name.
Position Type	Position Type.
Parent Position Name	Parent Position's name.

**Note:** The search is made on *identityAttribute* while finding records. By default, "Login Name" for Account and "Id" for Account Group is set in the *identityAttribute*.

## Adding new custom attributes in schema

---

Currently Siebel Connector schema provides basic minimum attributes required to manage Employee and position. If you want to enhance schema, you can add more attributes to the existing schema. You can use Siebel Tools to get the details about attributes to be managed using schema. If you add any new multi value attribute, configure the following attribute in Application using the debug page:

```
<entry key="customMVGAttr">
  <value>
    <List>
      <!-- Format is <<Multi value attribute Name>>:<<MVG Business component>>:<<Business Object for field>>:<<Business component for field>>:<<Search key for multi value field>> -- >
      <String>Position:Position:Position:Position:Id</String>

      <String>Responsibility:Responsibility:Responsibility:Responsibility:Name</String>
      </List>
    </value>
  </entry>
```

**Note:** As position and responsibility are main multi value field in Employee, if you do not configure it, Siebel Connector will assume the default business components and objects. But for other Multi value attribute to work, you need to configure this attribute in Application.

## Provisioning policy attributes

---

The following table lists the provisioning policy attributes for Create and Update of Accounts and Group:

Attributes	Description
<b>Create Account</b>	
Login Name	Employee's login name.
First Name	Employee's first name.
Last Name	Employee's last name.
Position	Multi-value attribute that contains a list of all positions assigned to employee.
Primary Position Id	Employee's primary position Id.
Responsibility	Multi-value attribute that contains a list of all responsibilities of employee.
Password	Employee account password.
Verify Password	Employee account password.
Job Title	Job title.
Employee Type	Employee type.
<b>Update Account</b>	
First Name	Employee's first name.

## Troubleshooting

Attributes	Description
Last Name	Employee's last name.
Responsibility	Multi-value attribute that contains a list of all responsibilities of employee.
Primary Position Id	Employee's primary position Id.
<b>Create Group</b>	
Position	Name of position.
Division	Division of position.
Position Type	Position type.
Parent Position Id	Parent position's Id.
<b>Update Group</b>	
Position Type	Position type.
Parent Position Id	Parent position's Id.

## Troubleshooting

---

### 1 - When Siebel JAR files are not copied correctly in the WEB-INF/lib directory error messages appear

When Siebel JAR files are not copied correctly in the WEB-INF/lib directory, the following errors are obtained:

- Test connection fails with the following error:



- During add new entitlement the following error message is displayed:



**Resolution:** Copy the correct Siebel JAR files. For more information, see the “Troubleshooting” on page 258.

# Chapter 28: SailPoint Solaris Connector

---

The following topics are discussed in this chapter:

Overview .....	259
Supported features .....	259
Supported Managed Systems .....	260
Pre-requisites .....	260
Administrator permissions .....	260
Configuration parameters .....	261
Additional configuration parameters for SSH configuration .....	261
Public key authentication configuration .....	262
Schema attributes .....	262
Account attributes .....	262
Group attributes .....	264
Provisioning policy attributes .....	264
Account attributes .....	264
Group attributes .....	265
Additional information .....	266
Unstructured Target Collector .....	266
Troubleshooting .....	267

## Overview

---

In Solaris Connector, users on Solaris computer are used for account provisioning. For group provisioning, groups are used. You can configure the Connector to use any of the attributes of user/group which are supported by Solaris commands.

## Supported features

---

SailPoint Solaris Connector supports the following features:

- Account Management
  - Manages Solaris Users as Accounts
  - Aggregation, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements
- Account - Group Management
  - Manages Solaris groups as Account-Groups
  - Aggregation, Refresh Groups
  - Create, Update, Delete

## Overview

- Permission Management
  - Solaris application can be configured to read file permissions directly assigned to accounts and groups using Unstructured Target Collector.
  - The connector also supports automated revocation of the aggregated permissions for accounts and groups.

**Note:** Solaris connector supports MD5, SHA-1, and SHA-2 cryptographic hash functions.

## References

- “Unstructured Target Collector” on page 266
- Appendix D: Before and After Provisioning Action

## Supported Managed Systems

---

The Solaris connector supports the following versions of the operating system:

- Solaris 11.3 SPARC x86
- Solaris 11.2 SPARC x86
- Solaris 11 SPARC x86
- Solaris 10 SPARC x86

**Note:** For any issues related to Solaris, see “Troubleshooting” on page 267 section.

## Pre-requisites

---

SSH should be installed on Solaris computer.

## Administrator permissions

---

- You can use root user for managing your applications.
- If you want to use sudo user to perform the provisioning operations, the sudo user must be configured with the following rights and permissions:

**Rights to execute the following commands with root privilege:**

```
/bin/chmod, /usr/sbin/useradd, /usr/sbin/usermod, /usr/sbin/userdel,  
/usr/sbin/groupadd, /usr/sbin/groupmod, /usr/sbin/groupdel, /usr/bin/passwd,  
/usr/bin/groups, /usr/bin/date, /bin/rm, /bin/echo, /usr/bin/find, /bin/cat  
/etc/shadow, /bin/cat /etc/passwd, /bin/cat /etc/group, /bin/cat /etc/user_attr,  
/usr/bin/getent, /bin/grep -i * /etc/default/login, /bin/grep -i *  
/etc/security/policy.conf
```

**An entry in /etc/sudoers file should look similar to the following:**

```
username ALL = (root) PASSWD : /bin/chmod, /usr/sbin/useradd,  
/usr/sbin/usermod, /usr/sbin/userdel, /usr/sbin/groupadd, /usr/sbin/groupmod,  
/usr/sbin/groupdel, /usr/bin/passwd, /usr/bin/groups, /usr/bin/date, /bin/rm,  
/bin/echo, /usr/bin/find, /bin/cat /etc/shadow, /bin/cat /etc/passwd, /bin/cat  
/etc/group, /bin/cat /etc/user_attr, /usr/bin/getent, /bin/grep -i *  
/etc/default/login, /bin/grep -i * /etc/security/policy.conf
```

- Note:** All commands mentioned above are for default configuration. If any of the command is modified in application definition, then the respective changes in /etc/sudoers file entry should also be performed. Verify command paths on Solaris computers as they might differ from the values mentioned here.
- Note:** If you want to use sudo user to perform the provisioning operations ensure to configure home directory with proper write access for this sudo user. In case sudo user is using Guest home directory then ensure it has proper write access over this directory.

## Configuration parameters

---

The following table lists the configuration parameters of Solaris Connector:

Parameters	Description
UNIX Server Host	Host Name/IP address of Solaris computer.  <b>Note:</b> For IdentityIQ version 6.4 Patch 4 and above, the format of the application XML has been changed from <entry key="UnixServerHost" value=" <hostname>" /&gt; to &lt;entry key="host" value="<hostname>" /&gt;</hostname></hostname>
SSH Port	SSH port configured. Default value: 22
Not a 'root' user	If User ID specified is not root, check this parameter.
User Name	User ID on Solaris computer that you want to use for connector operations.
User Password	Password of the target system user account that you want to use for connector operations. Default value: <b>sadmin</b>
Private Key File Path	Path to Private Key File. Private/Public key authentication will have precedence over password authentication.
Passphrase For Private Key	Passphrase provided for creating Private Key.

## Additional configuration parameters for SSH configuration

---

The following procedure provides the steps for adding the additional configuration parameters for SSH configuration in Application or Target Source debug page.

**Note:** These additional configuration parameters must be added in the Application/Target Source debug page.

- Following is the default command for setting shell prompt on UNIX computer:  

```
<entry key="SetPrompt" value="PS1='SAILPOINT' "/>
```

In the above command, "SetPrompt" is the application/target source attribute and PS1='SAILPOINT' is the value of the application/target source attribute.  
If the command for setting shell prompt is different than the default command, change the value of the application/target source attribute. If the application/target source attribute is not found, add the following new entry in the application/target source debug page:  
For example: For tcsh shell, the entry value would be:

## Schema attributes

```
<entry key="SetPrompt" value="set prompt='SAILPOINT>' "/>
```

2. For executing the commands, verify that the default shell is present on your system. If the default shell present on your UNIX system is different, modify the value of the application/target source attribute. If the application/target source attribute is not found, add the following new entry in the application/target source debug page:

```
<entry key="DEFAULT_SSH_SHELL" value="tcsh"/>
```

## Public key authentication configuration

---

This is an alternative security method to using passwords. To use public key authentication, you must generate a public and a private key (that is, a key pair). The public key is stored on the remote hosts on which you have accounts. The private key is saved on the computer you use to connect to those remote hosts. This method allows you to log into those remote hosts, and transfer files to them, without using your account passwords.

Perform the following configuration steps to make the UNIX computer as the server and IdentityIQ computer as client:

1. Generate Private and Public key's. For more information of the standard steps, see "7 - Test connection fails for key based authentication with an error." on page 268.
2. Append contents of public key file to `~/.ssh/authorized_keys` as shown below.  
`cat <public key file> >> ~/.ssh/authorized_keys`
3. Copy private key file to a location which is accessible by the server.
4. Provide path of private key file in application configuration.

## Schema attributes

---

This section describes the different schema attributes.

### Account attributes

---

The following table lists the account attributes:

Attributes	Description
username	Name of user.
uid	Numeric ID for user.
primgrp	An existing group integer ID or character-string name. Without the <b>-D</b> option, it defines the new user primary group membership and defaults to the default group. You can reset this default value by invoking useradd -D -g group. GIDs 0-99 are reserved for allocation by the Solaris Operating System.
groups	Secondary groups of user. List of groups assigned to user.
roles	Contains the list of roles for each user.
home	Home directory of user.
shell	Default shell of user.

Attributes	Description
comment	Any text string. It is generally a short description of the login, and is currently used as the field for the user's full name. This information is stored in the user's /etc/passwd entry.
authorization	One or more comma separated authorizations defined in auth_attr(4). Only a user or role who has grant rights to the authorization can assign it to an account.
skel_dir	A directory that contains skeleton information (such as.profile) that can be copied into a new user's home directory. This directory must already exist. The system provides the /etc/skel directory that can be used for this purpose
project	Name of the project with which the added user is associated. See the projname field as defined in project(4).
expire	<p>Specify the expiration date for a login. After this date, no user will be able to access this login. The expire option argument is a date entered using one of the date formats included in the template file /etc/datemsk. See getdate(3C).</p> <p>If the date format that you choose includes spaces, it must be quoted. For example, you can enter 10/6/90 or October 6, 1990. A null value (" ") defeats the status of the expired date. This option is useful for creating temporary logins.</p>
inactive	The maximum number of days allowed between uses of a login ID before that ID is declared invalid. Normal values are positive integers. A value of 0 defeats the status.
lock_after_retries	Specifies whether an account is locked after the count of failed logins for a user equals or exceeds the allowed number of retries as defined by RETRIES in /etc/default/login. Possible values are yes or no. The default is no. Account locking is applicable only to local accounts and accounts in the LDAP name service repository if configured with an enableShadowUpdate of true as specified in ldapclient(1M).
limitpriv	The maximum set of privileges a user or any process started by the user, whether through su(1M) or any other means, can obtain. The system administrator must take ensure that when deleting the privileges from the limit set. Deleting any basic privilege has the ability of crippling all applications; deleting any other privilege can cause many or all applications requiring privileges to malfunction.
defaultpriv	The default set of privileges assigned to a user's inheritable set upon login.
profiles	Contains an ordered, comma-separated list of profile names selected from prof_attr(4). Profiles are enforced by the profile shells, pfcsh, pfksh, and pfsh. See pfsh(1). A default profile is assigned in /etc/security/policy.conf (see policy.conf(4)). If no profiles are assigned, the profile shells do not allow the user to execute any commands.
failedretries	<p>Indicates if the user account is locked. Possible values include:</p> <ul style="list-style-type: none"> <li>• <b>true</b>: The user account is locked. The values yes, true, and always are equivalent. The user is denied access to the system.</li> <li>• <b>false</b>: The user account is not locked. The values no, false, and never are equivalent. The user is allowed access to the system. Default value.</li> </ul>
pwdminage	The minimum number of days required between password changes for user. MINWEEKS is found in /etc/default/passwd and is set to NULL.

## Provisioning policy attributes

Attributes	Description
pwdmaxage	The maximum number of days the password is valid for user. MAXWEEKS is found in /etc/default/passwd and is set to NULL.
pwdwarn	The number of days relative to max before the password expires and the name are warned.
pwdlastchg	The date password was last changed for name. All password aging dates are determined using Greenwich Mean Time (Universal Time) and therefore can differ by as much as a day in other time zones.
audit_flags	Specifies per-user Audit pre selection flags as colon-separated <b>always-audit-flags</b> and <b>never-audit-flags</b> . For example, audit_flags=always-audit-flags:never-audit-flags.

## Group attributes

The following table lists the group attributes:

Attributes	Description
groupname	Name of the account group
groupid	Numeric ID of account group

## Provisioning policy attributes

This section lists the different policy attributes of Solaris Connector.

### Account attributes

The following table lists the provisioning policy attributes for Create and Update Account:

Attributes	Description
Create Account	
username	Name of user.
uid	Numeric ID for user.
	Allow duplication of User ID
primgrp	An existing group integer ID or character-string name. Without the -D option, it defines the new user primary group membership and defaults to the default group. You can reset this default value by invoking useradd -D -g group. GIDs 0-99 are reserved for allocation by the Solaris Operating System.
home	Home directory of user.
shell	Default shell of user.

Attributes	Description
comment	Any text string. It is generally a short description of the login, and is currently used as the field for the user's full name. This information is stored in the user's /etc/passwd entry.
authorization	One or more comma separated authorizations defined in auth_attr(4). Only a user or role who has grant rights to the authorization can assign it to an account.
profiles	Contains an ordered, comma-separated list of profile names selected from prof_attr(4). Profiles are enforced by the profile shells, pfcsh, pfksh, and pfsh. See pfsh(1). A default profile is assigned in /etc/security/policy.conf (see policy.conf(4)). If no profiles are assigned, the profile shells do not allow the user to execute any commands.
project	Name of the project with which the added user is associated. See the projname field as defined in project(4).
expire	Specify the expiration date for a login. After this date, no user will be able to access this login. The expire option argument is a date entered using one of the date formats included in the template file /etc/datemsk. See getdate(3C).  If the date format that you choose includes spaces, it must be quoted. For example, you can enter 10/6/90 or October 6, 1990. A null value (" ") defeats the status of the expired date. This option is useful for creating temporary logins.
inactive	The maximum number of days allowed between uses of a login ID before that ID is declared invalid. Normal values are positive integers. A value of 0 defeats the status.
lock_after_retries	Specifies whether an account is locked after the count of failed logins for a user equals or exceeds the allowed number of retries as defined by RETRIES in /etc/default/login. Possible values are yes or no. The default is no. Account locking is applicable only to local accounts and accounts in the LDAP name service repository if configured with an enableShadowUpdate of true as specified in <b>ldapclient(1M)</b> .
pwdwarn	Warning period for user's password expiry.
pwdminage	Minimum period between user's password change.
forcepwdchange	If user has to be forced to change password on next logon.
pwdmaxage	Maximum period for which password is valid for user.
Password	Initial password for newly created user account.

## Group attributes

---

The following table lists the provisioning policy attributes for Create and Update Group:

Attributes	Description
groupname	Name of the account group
groupid	Numeric ID of account group
dupgid	Allow duplication of groupid.

## Additional information

Attributes	Description
<b>Update Group</b>	
groupid	Numeric ID of account group.
dupgid	Allow duplication of groupid.

# Additional information

This section describes the additional information related to the Solaris Connector.

## Unstructured Target Collector

Solaris uses a data structure which requires the configuration in the **Unstructured Targets** tab to collect targeted data and correlate it with account **identityAttribute** for Accounts and group identityAttribute for Account Groups. For more information on the **Unstructured Targets** tab, see “Unstructured Targets Tab” section of the *SailPoint User’s Guide*.

For Solaris target permission, the Unstructured Targets functionality will be enabled if **UNSTRUCTURED\_TARGETS** feature string is present in the application.

Multiple target sources can be specified and configured for an application which supports unstructured targets. This will be useful for applications which want to fetch resource information from multiple target sources.

Solaris Target Collector support aggregation of file/directories under specified file system path(s). Only direct access permissions will be correlated to the Users and Groups. For UNIX platforms direct access means ownership of file or directory.

Attributes	Description	Possible values
Unix File System Path(s)*	Absolute path(s) which are to be scanned for resources.	Multiple paths can be mentioned with comma separated values. For example, /etc, /tmp
Application Name*	Name of the application with which Unstructured Target will be correlated.	

**Note:** Attributes marked with \* sign are the mandatory attributes.

**Note:** If Unstructured Configuration is configured before upgrading to version 7.1 from version 6.0 Patch 5 or 6.0 Patch 6, then update the configuration and specify the Connector Application Name.

## Rule configuration parameters

The rule configuration parameters are used to transform and correlate the targets.

**Correlation Rule:** The rule used to determine how to correlate account and group information from the application with identity cubes in IdentityIQ.

**Note:** For version 6.2 onwards, the default schema does not have correlation keys defined. Update correlation rule in Unstructured Target Configuration accordingly.

## Provisioning related parameters

Select the settings for provisioning to the box.

- **Override Default Provisioning:** Overrides the default provisioning action for the collector.
- **Provisioning Action:** The overriding provisioning action for the collector.

# Troubleshooting

---

## 1 - Test connection fails with an error.

The following error message appears when test connection fails:

`java.io.IOException: Corrupt Mac on input`

OR

`Error: Login failed. Error while connecting to host: xxxxx. The message store has reached EOF`

**Resolution:** Add Cipher **3des-cbc** or **blowfish-cbc** to the list of Cipher's in `/etc/ssh/sshd_config` file and restart `sshd`.

- **For X86:** include **3des-cbc** or **blowfish-cbc** in Ciphers list

For example, Ciphers `aes128-ctr, aes192-ctr, aes256-ctr, aes128-cbc, aes192-cbc, aes256-cbc, arcfour, arcfour128, arcfour256, 3des-cbc, blowfish-cbc`

- **For SPARC:** include **3des-cbc** in Ciphers list

For example, Ciphers `aes128-ctr, aes192-ctr, aes256-ctr, aes128-cbc, aes192-cbc, aes256-cbc, arcfour, arcfour128, arcfour256, 3des-cbc`

## 2 - Test connection fails with an error

The following error message appears when test connection fails:

`Login failed. Failed to authenticate the ssh credentials for user: root to host: xxxxxxx`

**Resolution:** Update `/etc/ssh/sshd_config` file for the following entry and restart `sshd`:

`PasswordAuthentication yes`

## 3 - Aggregation fails with an error

The following error message appears when aggregation fails:

`Exception during aggregation. Reason: sailpoint.connector.ConnectorException: Failed to execute command: cat /etc/group | grep -v '^+' | grep -v '^-' Error:`

**Resolution:** Create home directory for sudo user and run aggregation again. Ensure that the sudo user is able to create files in its home directory.

## 4 - Test connection fails with an error

The following error message appears when aggregation fails:

`Fails with error Login failed. Failed to authenticate the ssh credentials for user: test to host: xxxxxxx`

## Troubleshooting

**Resolution:** The **ksh93** shell is the default shell `/usr/sbin/sh -> .../bin/i86/ksh93`. The **J2SSH** library does not work properly with this shell.

In default installation of Solaris 11, bash and tcsh are installed, use one of them for provisioning. Use application attribute **DEFAULT\_SSH\_SHELL**.

For more information on **DEFAULT\_SSH\_SHELL** parameter, see “Additional configuration parameters for SSH configuration” on page 261.

## 5 - Aggregation/test connection fails with timeout error

Aggregation/test connection fails with the following timeout error:

```
Exception during aggregation. Reason: sailpoint.connector.ConnectorException:  
Account aggregation failed. Timeout occurred.
```

**Resolution:** Change the value of the **SSHLoginTimeout (in millisecond)** application attribute as per your requirement in the debug page of the application:

```
<entry key="SSHLoginTimeout" value="1000" />
```

## 6 - After target aggregation resources are not getting correlated with Account Groups

After target aggregation the resources are not getting correlated with Account Groups.

**Resolution:** Ensure that your correlation rule populates "Correlator.RULE\_RETURN\_GROUP\_ATTRIBUTE" as follows:

```
....  
if (isGroup) {  
    returnMap.put(Correlator.RULE_RETURN_GROUP_ATTRIBUTE, "nativeIdentity");  
    returnMap.put(Correlator.RULE_RETURN_GROUP_ATTRIBUTE_VALUE, nativeId);  
}  
....
```

## 7 - Test connection fails for key based authentication with an error

Test connection fails for key based authentication with the following error.

```
Login failed. Error while connecting to host:<hostname>. Cannot read key file.
```

**Resolution:** Perform the following steps to generate/convert private/public keys in format which is supported by UNIX direct connectors.

- Generate keys using openssl. This method can be used for any version of SSH.
  - a. Create private key using the following command:  

```
openssl <gendsa/genrsa> -des3 -out <private_key> 1024
```
  - b. Change the permission on the <private\_key> file as follows:  

```
chmod 0600 <private_key>
```
  - c. Create public key from private\_key  

```
ssh-keygen -y -f <private_key> > <public_key>
```
  - d. Use the <private\_key> and <public\_key> files for authentication.
- Generate keys using ssh-keygen. (OpenSSH 5.8 or above)
  - a. Create private and public key using the following command  

```
ssh-keygen -t <dsa/rsa> -b 1024
```

- By default files with name **id\_dsa/id\_rsa** and **id\_dsa.pub/id\_rsa.pub** will be created.
- b. Convert <private key> to have DES-EDE3-CBC encryption algorithm by using the following command:  

```
openssl <dsa/rsa> -in <private_key> -out <new_private_key> -des3
```
  - c. Change the permission on the <new\_private\_key> file as follows:  

```
chmod 0600 <new_private_key>
```
  - d. Create public key file using the new private key as follows:  

```
ssh-keygen -y -f <new_private_key> > <new_public_key>
```
  - e. Use the <new\_private\_key> and <new\_public\_key> files for authentication.

## 8 - Test connection fails with an error when sudo user is configured for public key authentication

Test connection fails with the following error when sudo user is configured for public key authentication:

Test SSH communication failed over host: xxxxxxxx. Error while executing command: sudo -p %SAILPOINTSUDO echo TestConnection over host: xxxxxxxx. Invalid sudo user password.

**Resolution:** On managed system,

- if Sudoers file is having Sudo user with **PASSWD** attribute assigned, then the sudo user's password specified in application configuration, password must be correct for certificate based authentication.
- if Sudoers file is having Sudo user with **NOPASSWD** attribute assigned, then the sudo user's password specified in application configuration, password can be incorrect/or any value. Certificate based authentication must still work.

**Note:** Password is mandatory field on application UI.

## 9 - Aggregation fails with an error for Solaris

The following error message appears when aggregation fails:

Unable to create iterator: sailpoint.connector.ConnectorException: Exception occurred while getting system information:openconnector.ConnectorException: get Remote Date failed

**Resolution:** Add the following entry of application attribute in the debug page of the application:

```
<entry key="DEFAULT_SSH_SHELL" value="bash" />
```

## 10 - For any Solaris user at a time, the maximum allowed addition of groups are 16

Solaris has restriction to select 16 group at a time to allocate any user, while IdentityIQ supports to select more than 16 through Console. In this case only first 16 selected groups are being allocated to user.

**Note:** This issue does not display any type of error from IdentityIQ and access request will also be marked as committed.

**Resolution:** To add more number of groups (exceeding 16), user must add the groups in batch of 16 at a time.

## **Troubleshooting**

# Chapter 29: SailPoint SQL Loader Connector

---

The following topics are discussed in this chapter:

Overview .....	271
Supported features .....	271
Supported Managed Systems .....	272
Administrator permissions .....	272
Configuration parameters .....	272
Schema Attributes .....	274
Troubleshooting .....	274

## Overview

---

The SailPoint SQL Loader Connector supports Read/Write operations on flat file data like CSV, TEXT flat files. The data in these files are separated with delimiters. This connector can handle aggregation for multiple file by defining complex SQL query.

This connector can be configured to enable the automatic discovery of schema attributes. See “Schema Attributes” on page 274.

## Supported features

---

SailPoint SQL Loader Connector supports the following features:

- Account Management
  - Manages SQL Users as Accounts
  - Aggregation, Partitioning Aggregation, Refresh Accounts, Discover Schema
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements
- Account - (Application Object Type)
  - Manages SQL groups as Account-(Application Object Type)
  - Aggregation, Refresh (Application Object Type)
  - Create, Update, Delete
- Permission Management
  - Application reads permissions directly assigned to application object types as direct permissions during account and application object type aggregation.
  - The connector does not support automated revocation of the aggregated permissions and creates work item for such requests.

## Configuration parameters

SailPoint supports the following additional SQL Loader Connector features:

- Ability to provide the SQL statement or stored procedure during application configuration for automatic discovery of account-group (application object type) schema attributes from same or different files used for the account schema.
- Ability to define provisioning rule(s) called for each row in the data file to provision account and group (application object type) attributes.
- Ability to define separate provisioning rule for specific operation called for each row in the data file to provision account and group (application object type) attributes. Operation that include are Enable, Disable, Unlock, Delete, Create, and Modify.

**Note:** An example of a provisioning rule is located in `examplerules.xml` file.

### References

- Appendix C: Partitioning Aggregation

## Supported Managed Systems

---

SailPoint SQL Loader Connector supports flat file data which contains delimiter. The extension for the flat file can be `.txt` or `.csv`.

## Administrator permissions

---

Administrator must have the read and write permission on the files in the given directory path.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The SQL Loader Connector uses the following connection attributes under different tabs (Settings, Merging and Iteration Partitioning):

Attribute	Description
<b>Settings</b>	
<b>SQL Loader Connection Settings</b>	
Directory Path*	The directory path in which the Target files are stored.
Delimiter*	Delimiter separator type with which the entire row gets separated with different columns.
Treat First Record As Header	Indicates whether the first record of CSV files container is provided with column headers as first column then URL would be as follows:  <code>jdbc:csv:/c:/data?_CSV_Separator=  ;_CSV_Header=true</code>

Attribute	Description
Database URL*	<p>URL to connect to the database. This will be automatically created when the delimiter and directory attributes are filled up with appropriate data.</p> <p>For example,</p> <ul style="list-style-type: none"> <li>• (For Windows) <code>jdbc:csv:/c:/data?_CSV_Separator= </code></li> <li>• (For UNIX) <code>jdbc:csv:///home/sqlloader/acc.csv?_CSV_Separator= </code></li> </ul>
JDBC Driver*	Enter the JDBC driver class path.
<b>Query Settings</b>	
SQL Statement*	<p>The SQL attribute can be used to customize the select statement that is generated when iterating over objects. You can specify the exact SQL that is executed if you want to filter out objects or only want to select a few objects from a table. Additionally, if you want to perform joins between more than one table, it is impossible to describe with the schema alone.</p> <p>By default if the SQL option is null when the query string is built using the schema attributes and <b>nativeObjectType</b>.</p>
getObjectSQL	The object SQL statement.
useExecuteQuery	Use Statement.executeQuery() instead of the default Statement.execute()
Direct Permission Execute Query	Enable this option to execute the query for direct permission.
Get Direct Perm Object SQL	<p>Direct Permission Execute Query is used to retrieve the direct permission data from permission file. Permission file should contain at least Identity attribute column. The permission data is retrieved by referring the identity attribute in the column at the time of aggregation through main SQL query in which the identity attribute is mentioned.</p> <p><b>Note:</b> Query must be written in such a way that <b>ResultSet</b> data must contain first column as <b>Target</b>, second column as <b>Permission</b> and third column as <b>annotation (optional)</b>.</p> <p>For example, <code>SELECT column4 AS TARGET, column5 AS PERMISSION FROM Permission p WHERE CONCAT(TRIM(CONCAT(p.column1,' ')), TRIM(p.column2)) = '\${identity}';</code></p> <p>Here file name is <code>Permission.csv</code> and <b>\$(identity)</b> is Identity attribute.</p>
<b>Merging</b>	
Data needs to be merged	<p>Select this option if the data for a single object spans multiple lines.</p> <p>This option enables the connector to verify the order of the data returned from the database when merging to prevent data loss. When merging, it is very important to have the ORDER BY clause in your SQL statement to prevent out of order errors.</p>
Index Column	Name of the index column that will be used when finding like objects in the dataset.

## Schema Attributes

Attribute	Description
Which columns should be merged?	Names of the columns from the file from which values must be merged.
<b>Note:</b> User must discover the schema to get the suggested column values in index and merge columns for selection. Discover schema populates the values in multi-suggest attribute dropdown of index and merge columns which have the auto complete facility.	
Iteration Partitioning	
Partitioning Enabled	Select this checkbox to configure and enable partitioning.
Partitioning Statements	Enter the list of sql/stored procedure statements that must be executed when partitioning. The statements must include all of the rows and each line/statement so it can be proceeded in separate threads and/or multiple hosts.  For more information, see Appendix C: Partitioning Aggregation.

## Schema Attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. Version 7.1 supports multiple types of objects, account and any number of group application object types. Account objects are used when building identities Link objects. Additional schema definitions can be used when building AccountGroup objects which are used to hold entitlements shared across identities.

The SQL Loader Connector's most important attribute is the SQL statement. In many cases this is a stored procedure (`call mystoredProcedure`). In other cases it is select from a table with any number of joins included. If this connector is configured to use the automatic discovery function, it connects to the database and executes the statement provided and then uses the meta-data returned from the result to build the column names.

## Troubleshooting

---

### 1 - Database URL format requires change if application is on UNIX computer

The following error message is displayed:

```
Unable to discover the [account] schema for this
application.[sailpoint.connector.ConnectorException: Error while trying to discover
the columns. SQL[null]sailpoint.connector.ConnectorException: Failure
trying to get a pooled connection to
[jdbc:csv://home/manoj?_CSV_Separator=,]java.sql.SQLException: home/manoj doesn't
exist or can't be accessed. If you're using mapped drives to access database files,
you may need to check the security permissions.]
```

**Resolution:** Update the directory path field in the following format:

```
//DirectoryName/SubDirectoryName
```

The database URL should be of the following format for UNIX computers:

`jdbc:csv:///DirectoryName/SubDirectoryName`

## 2 - Issue of SQL Loader dropping records

**Resolution:** If the CSV data is provided with column headers as first column then URL would be as follows:

`jdbc:csv:/c:/data?_CSV_Separator=|;_CSV_Header=true`

In this case, perform the following:

1. Use `CSV_Header=true` in the URL. By adding this, you can directly use `header_name` of the csv file as column name in the SQL query.

For example, url would be as follows:

`jdbc:csv:/c:/data?_CSV_Separator=\u003B;_CSV_Header=true`

2. Instead of using `column1, column2...` in the SQL query use the header names.

For Example, earlier SQL query was as follows:

`Select emp.column1 as pimId, emp.column2 as employeeId .... from table abc;`

## **Troubleshooting**

# Chapter 30: SailPoint Sybase Connector

---

The following topics are discussed in this chapter:

Overview .....	277
Supported features .....	278
Supported Managed Systems .....	278
Pre-requisites .....	278
Administrator permissions .....	279
Configuration parameters .....	279
Schema attributes .....	280
Account attributes .....	280
Group attributes .....	281
Provisioning policy attributes .....	281
Additional information .....	282
Identity and Entitlement representation .....	282
Troubleshooting .....	282

## Overview

---

Sybase Adaptive Server Enterprise (ASE) is widely used database Server, mainly used to store data for different business modules like Sales, Production, Human Resource, Finance and Accounting. It requires the user to authenticate in order to connect to database to manipulate business data. It controls the users/roles logging in to Sybase ASE Managed System and performs other activities like processing transactions, writing logs, updating database files and so on.

A group is a means of organizing users, where as a role is usually a means of organizing rights. User roles are aggregated as Account Groups as it is widely used by the customers.

SailPoint Sybase Adaptive Server Enterprise Connector manages the following entities on Sybase Adaptive Server Enterprise:

- Login User
- Database User
- Roles
- Database Groups
- Aliases

## **Supported features**

---

SailPoint Sybase Connector provides support for the following features:

- Account Management
  - Manages Sybase Users as Accounts
  - Aggregation, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements  
(Aliases, database\_groups, roles)
- Account - Group Management
  - Supports multiple group functionality.
  - Manages Sybase server roles as group
    - Aggregation, Refresh Groups
    - Create, Update, Delete
  - Manages database groups as database\_group
    - Aggregation, Refresh Groups

**Note:** If user adds the database group to login user, the database user is created with the name of login user on the respective database.

For example: Login user: JamesSmith. If user adds database group (master.public) to JamesSmith then the database user (master.JamesSmith) is created on the 'master' database.

## **Supported Managed Systems**

---

Following versions of Sybase ASE are supported by the SailPoint Sybase ASE Connector:

- SAP ASE 16.0
- Sybase ASE 15.7
- Sybase ASE 15.5
- Sybase ASE 15.0

## **Pre-requisites**

---

Sybase JDBC Driver is required for proper functioning of SailPoint Sybase ASE Connector. For example, jconn4.jar. This JDBC driver must be copied in the ..\identityiq\WEB-INF\lib directory.

**Note:** It is recommended that, on Managed System the server name and database name must not be the same.

## Upgrade

Before upgrading IdentityIQ version from 6.4 to 7.1, user must execute the following:

- All the certifications must be executed.

After upgrade, user can view the certification history by right clicking on the specific entity and selecting the **View History** in the certification panel.

- The certifications related to 6.4 must be executed.
- Before aggregating the data, user must select the **Detect deleted accounts** option in task menu.

## Administrator permissions

---

The minimum Administrative Account permission required to be granted for Provisioning is **SSO\_ROLE (System Security Officer)** and **SA\_ROLE (System Administrator Role)**.

### Minimum permission required for Account Aggregation task

1. Login using administrator credentials and create a new user on managed system using the following command:  
**CREATE USER <>USER\_NAME>> IDENTIFIED BY <>PASSWORD>>**
2. Grant Read-only access to the newly created user using the following command:  
**GRANT select on master..sysloginroles to <>USER\_NAME>>**  
**GRANT select on master..syssrvroles to <>USER\_NAME>>**

**Note:** For accessing user defined databases first create an account on that database.  
User will not be able to access the database without having an account on that database.

#### Query to create Database user:

```
USE [DATABASE_NAME];
GO
SP_ADDUSER 'LOGIN_USER_NAME', 'DATABASE_USER_NAME'
GO
```

## Configuration parameters

---

The following table lists the configuration parameters of SailPoint Sybase ASE Connector:

Parameters	Description
url*	A valid URL of Sybase ASE Connector which directly interacts with the managed system.  In case of jconn2.jar, use the following url: jdbc: sybase: Tds: <host>[ :<port>]  For example, jdbc: sybase: Tds: ACHAUDHARI: 5000
user*	Administrative Account to connect to Sybase ASE.
password*	Administrative Account Password.

## Schema attributes

Parameters	Description
driverClass*	The name of the Driver class supported by JDBC Type 4.  For example, In case of <code>jconn2.jar</code> , use the following driverClass: <code>com.sybase.jdbc2.jdbc.SybDataSource</code>
Included Databases	List of comma separated database names to be included in the aggregation operation.
Excluded Databases	List of comma separated database names to be excluded in the aggregation operation.  <b>Note:</b> If the Include Database parameter is populated, the Exclude Database parameter would be ignored.
Force Delete Login User	Deletes the login user. Options are: <ul style="list-style-type: none"><li>• <b>Yes:</b> Deletes the login user</li><li>• (Default) <b>No:</b> Does not delete the login users which have the database users attached.</li></ul>

**Note:** All the parameters marked with the \* sign in the above table are the mandatory parameters.

## Schema attributes

This section describes the different schema attributes.

### Account attributes

The following table lists the account attributes:

Attribute name	Description
name	Login user name.
server_user_id	Server User ID.
default_database	Default database. For example: master
default_language	Default language. For example: us_english
full_name	Full name of login user.
create_date	Date on which login user is created.
password_chg_date	Date on which password got changed.
last_login_date	Last login date of the user.
native_identity	An attribute which acts like a primary key during aggregation.
status	Status of login user: enable/disable
roles	Roles associated with login user.
database_groups	Database groups.

Attribute name	Description
aliases	Aliases associated with login user.

## Group attributes

---

The following table lists the group attributes:

Attribute name	Description
<b>Group Object Type = Groups</b>	
server_role_id	ID of the server Role.
native_identity	An attribute which acts like a primary key during aggregation.
name	Name of the Role.
password_chg_date	Date on which password got changed.
member_roles	Roles which are present under the hierarchy of the main role.
<b>Group Object Type = Database Groups</b>	
Group_name	Database Group Name.
native_identity	An attribute which acts like a primary key during aggregation.
Group_id	Database Group ID.

## Provisioning policy attributes

---

This section lists the single provisioning policy attributes of SailPoint Sybase ASE Connector that allows to select the type of user, login, or group.

Attribute name	Description
<b>Creating Group (User Role)</b>	
Role name*	Name of the role created.
Role Password	
Member Roles	
<b>Creating User (Login User)</b>	
Name*	Name of the Login User.
password*	Password for LoginUser.
Default database	Default database for Login User.
Default language	Default language.
Full name	Full name of the Login User.

**Note:** All the parameters marked with the \* sign in the above table are the mandatory parameters.

## Additional information

---

This section describes the additional information related to the Sybase Connector.

### Identity and Entitlement representation

---

This section describes the Identity and Entitlement representation for SailPoint Sybase Adaptive Server Connector.

#### Identity representation

**Account:** The Account in Sybase ASE Connector is represented as follows:

- For Server Login it is represented as <Account name>  
For example, login\_name

#### Entitlement representation

**Groups:** The Groups in Sybase ASE Connector are represented as follows:

- For Application Role it is represented as <Group name>

**Database Groups:** The Database Groups in Sybase ASE Connector are represented as follows:

For database groups it is represented as <database\_name>.<Group name>

For example: **master.public**

**Aliases:** The Alias in Sybase ASE Connector are represented as follows:

- For Alias it is represented as <database\_name>.<Alias name>  
For example: **master dbo**

## Troubleshooting

---

### 1 - Aggregation fails

When a login user is created in Sybase and is granted permission only on some of the Databases present on the server and if aggregation task is run for that application, Aggregation fails as the user is not able to access other databases.

**Resolution:** In application Configuration page under the “Include Databases” section, provide the complete list of databases (comma separated list) for which the login user have accesses.

This completes the aggregation successfully, and only details of the users present in the list of included database will be fetched.

# Chapter 31: SailPoint Tivoli Access Manager Connector

---

The following topics are discussed in this chapter:

Overview .....	283
Supported features .....	283
Supported Managed System .....	284
Pre-requisites .....	284
Configuration parameters .....	286
Schema attributes .....	286
Account attributes .....	286
Group attributes .....	287
Provisioning Policy attributes .....	287
Create account attributes .....	287
Create group attributes .....	288
Additional information .....	288
Unstructured Target Collector .....	288
Troubleshooting .....	289

## Overview

---

SailPoint Tivoli Access Manager Connector manages Users and their Entitlements through groups present in Tivoli Access Manager system.

## Supported features

---

SailPoint Tivoli Access Manager Connector supports the following features:

- Account Management
  - Manages Tivoli Access Manager Users as Accounts
  - Aggregation, Partitioning Aggregation, Refresh Accounts, Pass Through Authentication
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements
- Account - Group Management
  - Manages Tivoli Access Manager Group as Account-Groups
  - Aggregation, Refresh Groups
  - Create, Update, Delete

## Overview

- Permission Management
  - Application can be configured to read file permissions directly assigned to accounts and groups using Unstructured Target Collector.
  - The connector also supports automated revocation of the aggregated permissions for accounts and groups.

## References

- “Unstructured Target Collector” on page 288.
- “Appendix C: Partitioning Aggregation”

## Supported Managed System

---

SailPoint Tivoli Access Manager Connector supports

- IBM Security Access Manager for Web version 7.0
- Tivoli Access Manager version 6.1.

## Pre-requisites

---

Following are the pre-requisites for IBM Security Access Manager version 7.0 and Tivoli Access manager version 6.1.

### IBM Security Access Manager version 7.0

1. Install JAVA and set the environment variable for JRE path.
2. Install the following softwares from IBM installer pack where IdentityIQ is running:
  - **PDLIC:** Install Security Access Manager license by applying PDLIC license on the server hosting IdentityIQ application.
  - **PDJRT:** Install PDJRT to configure IBM Security Access Manager Runtime for Java component to enable the java application to use Security Access Manager security.
3. After successful installation, run the pdjrtecfg command,. This adds additional .jar file in the jre\_home\lib\ext directory which is used by Tivoli Access Manager Connector.  
For example,  
`"C:\Program Files\Tivoli\Policy Director\sbin> pdjrtecfg.exe -action config -host host_name -port 7135 -java_home "C:\Program Files\java\jdk1.7\jre"`

Following additional jar files are generated in `jre\lib\ext` directory:

- PD.jar
- ibmjcefips.jar
- ibmjcefw.jar
- ibmjceprovider.jar
- ibmjsseprovider2.jar
- ibmpkcs.jar
- local\_policy.jar
- US\_export\_policy.jar

4. Generate the config file which is required to communicate with IBM Security Access Manager by running the `com.tivoli.pd.jcfg.SvrSslCfg` command. The file path must be configured in application configuration.

For example:

```
C:\Program Files\Tivoli\Policy Director\java\export\pdjrte>java
com.tivoli.pd.jcfg.SvrSslCfg -action config -admin_id sec_master -admin_pwd
<password> -appsvr_id server1 -host <host> -port <port_number> -mode remote
-policysvr <host:7135:1> -authzsvr <host:7136:2> -domain default -cfg_file <path
of config file to be generated> -key_file <Path of key file to generate> -cfg_action
create
```

## Tivoli Access Manager version 6.1

1. Install the IBM Tivoli Access Manager Java Runtime component on the server.
2. Install the following libraries in JRE's lib/ext directory:

- PD.jar
- Ibjcefips.jar
- Ibjcefw.jar
- Ibjceprovider.jar
- ibmjsseprovider2.jar
- ibmpkcs.jar
- local\_policy.jar
- US\_export\_policy.jar

**Note:** Among the above files, PD.jar file can be found on Tivoli Access Manager java runtime installation and others are part of IBM's Java 1.5.

3. Tivoli Access Manager Authorization APIs uses the Java Authentication and Authorization Service (JAAS), for this the following changes are required in `java.security` file:
  - a. Specify the login file location: Point to the login configuration file from the JA-VA\_HOME/jre/lib/security/java.security file. For example, a sample entry from the java.security file might look like  
`login.config.url.1=file:${java.home}/lib/security/login.pd`
  - b. Creating a login configuration file: Create `login.pd` on the mentioned location, if it does not exist add an entry as follows:  
`pd {  
com.tivoli.pd.jazn.PDLoginModule required;  
};`

## Configuration parameters

- c. Specify the policy file location: Point the policy file location from JAVA\_HOME/jre/lib/security/java.security file. It displays as follows:  
policy.url.1=file:\${java.home}/lib/security/java.policy  
Add a grant permission entry as follows:  
permission javax.security.auth.AuthPermission "createLoginContext";
4. Create configuration file needed by Tivoli Access Manager Connector with the help of com.tivoli.pd.jcfg.SvrSslCfg utility.  
For example,  
C:\Program Files\Tivoli\Policy Director\java\export\pdjrte>java  
com.tivoli.pd.jcfg.SvrSslCfg -action config -admin\_id sec\_master -admin\_pwd <password> -appsvr\_id server1 -host <host> -port <port\_number> -mode remote -polcysvr <host:7135:1> -authzsvr <host:7136:2> -domain default -cfg\_file <path of config file to be generated> -key\_file <Path of key file to generate> -cfg\_action create

# Configuration parameters

---

The following table lists the configuration parameters of Tivoli Access Manager Connector:

Parameters	Description
Admin Name*	Tivoli Access Manager administrator name.
Admin Password*	Password of the administrator.
Domain	The domain that is to be managed by the connector.
Configuration file URL*	A URL reference to configuration data file generated by com.tivoli.pd.jcfg.SvrSslCfg utility.

# Schema attributes

---

This section describes the different schema attributes.

**Note:** All the attributes marked with \* sign are the mandatory attributes.

## Account attributes

---

The following table lists the account attributes:

Attributes	Description
userid	User ID of the user.
first_name*	The user's first name.
last_name*	The user's last name.
registryUID*	The account name stored in the user registry.
description	Text describing the user.

Attributes	Description
groups	The Access Manager groups that the user is a member of.
noPwdPolicy	Indicates whether a password policy is enforced.
ssoUser	Indicates whether the user has single sign-on abilities.
passwordvalid	The valid password.
accountValid	Indicates whether the account is disabled.
gsoWedCreds	gsoWedCreds.
gsoGroupCreds	Shows the list of gso group credentials assigned to a user. Will be shown as <userid>:<gso group name>.
importFromRegistry	Indicates that the new user must be imported from registry server and not created in registry server. Note that the user must be present in the registry server.

## Group attributes

---

The following table lists the group attributes:

Attributes	Description
GroupName*	Name of the group.
registryUID*	The group name stored in the user registry.
description	Text describing the Group.

## Provisioning Policy attributes

---

This section lists the different policy attributes of Tivoli Access Manager Connector.

### Create account attributes

---

The following table lists the provisioning policy attributes for Create Account:

Attributes	Description
UserID	User ID for the user.
First Name	The user's first name.
Last Name	The user's last name.
registryUID	The account name stored in the user registry.
Description	Text describing user.
Password	Password for the user.
Password Valid	Indicates whether the password will be expired.

## Additional information

Attributes	Description
Account Valid	Indicates whether the account is disabled.
GSO User	Indicates whether the user has single sign-on abilities.
GSO Web Credentials	List of gso credentials to be given to the new user. It should be as follows: <code>&lt;gso name&gt;:&lt;userid&gt;:&lt;gso-password&gt;</code>
GSO Group Credentials	List of gso group credentials to be given to the new user. It should be as follows: <code>&lt;gso group name&gt;:&lt;userid&gt;:&lt;gso-password&gt;</code>
ImportFromRegistry	Indicates whether the user is to be imported from registry server.
No Password Policy	Indicates whether a password policy is enforced.

## Configuration settings

Field separator for **GSO Web Credentials** and **GSO Group Credentials** can be defined in application template using the debug page. Default delimiter is ':'.

For example, entry: `<entry key="gso_field_seperator" value="#" />`

The above example will set the field separator for mentioned attributes to '#'.

**Note:** The delimiter selected should not be a part of any of the subfields in the mentioned attribute. For above example character '#' should not be part of **gso name** or **userID** or **gso password** on **GSO Web Credentials** attribute.

## Create group attributes

---

The following table lists the provisioning policy attributes for Create Group:

Attributes	Description
Name	Name of the group.
registryUID	The group name stored in the user registry.
Last Name	The user's last name.
Description	Text describing the user.

## Additional information

---

This section describes the additional information related to the Tivoli Access Manager Connector.

## Unstructured Target Collector

---

Tivoli Access Manager uses a data structure which requires the configuration in the **Unstructured Targets** tab to collect targeted data and correlate it with **account identityAttribute** for Accounts and **group identityAttribute** for AccountGroups. For more information on the **Unstructured Targets** tab, see "Unstructured Targets Tab" section of the *SailPoint User's Guide*.

The Unstructured Targets functionality will be enabled for Tivoli Access Manger connector if **UNSTRUCTURED\_TARGETS** feature string is present in the application.

Tivoli Access Manger Target Collector supports aggregation of Access Control List (ACL). Access permissions on ACL will be correlated to Users and Groups.

Following is the configuration parameter in **Unstructured Targets** tab for Tivoli Access Manager Connector:

Attribute	Description
IBM Tivoli Access Manager Application Name	Name of the IBM Tivoli Access Manager application.

## Provisioning related parameters

Select the following settings for provisioning to Tivoli Access Manager:

- **Override Default Provisioning:** Overrides the default provisioning action for the collector.
- **Provisioning Action:** The overriding provisioning action for the collector.

# Troubleshooting

---

## 1 - Connector not aggregating all accounts

When you have the LDAP user registry setup for Tivoli Access Manager, the Connector might not aggregate all accounts from Tivoli Access Manager.

**Resolution:** The Maximum search results is controlled by the following parameters:

- The **max-search-size** stanza entry in the [ldap] stanza of the **ldap.conf** configuration file:  
To indicate that there is no limit, set the stanza entry `max_search_size` to 0.  
For example: `max-search-size = 0`  
**Note:** **Restart the Tivoli Access Manager servers for the required changes.**
- The **ibm-slapdSizeLimit** parameter in the Tivoli Directory Server server **slapd32.conf** or **ibmslapd.conf** configuration file:  
To indicate there is no limit, set the size limit to 0.  
For example: `ibm-slapdSizeLimit = 0`  
**Note:** **This parameter affects all LDAP searches.**

**Note:** **Ensure that both parameters are set to value greater than or equal to the total number of records in Tivoli Access Manager.**

## **Troubleshooting**

# Chapter 32: SailPoint Top Secret Connector

---

The following topics are discussed in this chapter:

Overview .....	291
Supported features .....	291
Configuration parameters.....	292
Schema Attributes .....	293

## Overview

---

The SailPoint Top Secret Connector is a *read only* connector developed to read the TSSCFILE export.

**Note:** The Top Secret Full Connector supports the provisioning operations. For more information, see *SailPoint IdentityIQ Connector for CA-Top Secret Administration Guide*.

## Supported features

---

SailPoint Top Secret Connector supports the following features:

- Account Management
  - Manages TOP SECRET Users as Accounts
  - Aggregation, Refresh Accounts, Discover Schema
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements
- Account - Group Management
  - Manages TOP SECRET Groups as Account-Groups
  - Aggregation, Refresh Groups
  - Create, Update, Delete
- Permission Management
  - Application reads permissions directly assigned to accounts and groups as direct permissions during account and group aggregation.
  - The connector does not support automated revocation of the aggregated permissions and creates work item for such requests.

# Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Top Secret connector uses the following connection attributes:

**Table 1—Top Secret Connector - Configuration parameters**

Parameters	Description
filetransport	local, ftp, scp
host	The host of the server to which you are connecting.
transportUser	The user to use with ftp and scp. Not valid with local.
transportUserPassword	The password to use with of ftp and scp. Not valid with local.
file	The fully qualified path to the file.
fileEncoding	Specify the file encoding to be used by the connector. Valid values for this attribute can be found at: <a href="http://www.iana.org/assignments/character-sets">http://www.iana.org/assignments/character-sets</a>  If this field is empty, the default encoding (the value of file.encoding specified by the jvm) is used.
mapToResourceObjectRule	Rule that is called to override the transformation of the data from the Map<String, String> form into a ResourceObject.
filterString	Filter lines that match this string.
filterEmptyRecords	If activated, records that have no data are filtered.
preIterativeRule	The pre-iterate rule will check for a specially named Configuration object that will hold the last run statistics that can be compared against the current values.  This rule is called after the file has been transferred, but before iteration over the objects in the file is started.  For validation this rule can use the existing statistics stored by the postIterationRule during the last aggregation. The rule can compare the stored values with the new values to check for problems
postIterativeRule	The post-iterate rule can store away the configuration object and rename/delete the file if desired.  This rule is called after aggregation has completed and ALL objects have been iterated.
accountTypes	The type of account use to connect to the server. The default value is USER, but additional values can be specified.

**Table 1—Top Secret Connector - Configuration parameters**

Parameters	Description
groupTypes	The group type of the connector. The default values are GROUP ACID and PROFILE ACID.
Top Secret Attribute Customization Rule	The rule used to extend the parsing capabilities to customer records or redefine existing record configurations. TopSecret records hold a record identifier and all of the fields that are part of that record. This rule use the TopSecretRecord and TopSecretField classes to work with that information.

## Schema Attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports two types of objects, account and group. Account objects are used when building identities Link objects. The group schema is used when building AccountGroup objects which are used to hold entitlements shared across identities.

**Table 2—Top Secret Connector - Account Attributes**

Attribute	Description
XAUTH	The authorized level at which the user can access the resource.
VMMDDISK	The VM minidisks owned by the user.
ACTION	Specifies which action(s) CA-Top Secret will take when access to a resource is attempted.
LOCK TIME(MINUTES)	The time interval before unattended or inactive terminals are locked.
LOCK TIME FACILITY	The lock time for all terminals connected to the specified facility.
LANGUAGE PREFERENCE	The language preference code the user.
VOLSER(OWNED)	The volumes to which the user has access.
NAME	Identifies the ACID name.  Names can be up to 32 characters in length, must be surrounded by single quotes if embedded with blanks, and can use letters, numbers, and special characters.
SITRAN	Specifies which CICS transaction CA-Top Secret automatically executes after an ACID successfully signs on to a facility.  <b>Note:</b> If a SITRAN is added to an ACID that already has a CICS transaction defined, the transaction is replaced.
HOME	Defines the initial directory pathname. This is the initial directory used when a user enters the OMVS command or enters the ISPF shell. The HOME keyword accepts from one to 1024 characters. Both uppercase and lowercase characters are allowed. If HOME isn't defined, OpenEdition MVS sets the initial directory for the user to the root directory. HOME is optional.
XA ACID	XAUTH Resource Class Name

## Schema Attributes

**Table 2—Top Secret Connector - Account Attributes (Continued)**

Attribute	Description
MULTIPW	Used to assign or remove multiple password attributes, which means ACIDs need a different password to access each facility.
NOADSP	Used to prevent data sets, created by an ACID, from being automatically secured by MVS by setting the RACF bit.  NOADSP is used to define an ACID that will be used to create data sets that cannot be automatically protected by CA-Top Secret.
AUDIT	Used to allow an audit of ACID activity.
NOPWCHG	To prevent ACIDs from changing passwords at either signon or initiation.
OIDCARD	Used to support the physical identification of users through operator identification cards.
TRACE	Used to activate a diagnostic trace on all ACID activity (initiations, resource access, violations, user's security mode, etc.)
SUSPEND	Used to prevent ACIDs from accessing the system when a violation occurs.
MRO	Used to support the use of the multi-region option.
CONSOLE	Used to grant or remove an ACID's ability to modify control options. For VM, options are modified via the TSS MODIFY command only. With VSE and OS/390, options are modified at the O/S console or via the TSS MODIFY command function.
GAP	Used to specify that a profile will become, or will cease to be, globally administrable.
DUFXTR	Used to add or remove the DUFXTR attribute to an ACID. DUFXTR enables an ACID to use a RACROUTE REQUEST=AUTH (RACHECK) macro or the CA-Top Secret Application Interface to extract installation data (INSTDATA) or field data from a Security Record. DUFXTR is a component of the CA-Top Secret Dynamic Update Facility (DUF).
DUFUPD	Used to add or remove the DUFUPD attribute to an ACID. DUFUPD enables an ACID to use the CA-Top Secret Application Interface to update the installation data (INSTDATA) or field data from a Security Record. DUFUPD is a component of the CA-Top Secret Dynamic Update Facility (DUF).
TSOMPW	Used to support multiple TSO UADS passwords, on a user-by-user basis.
NOATS	Used to prevent an ACID in CICS and CA-IDMS from signing on via ATS (Automatic Terminal Signon).
ACEDEFAU	
ASUSPEND	Used to remove the suspension of an ACID that was suspended for administrative reasons.
WHO HAS RESOURCE	
PROFILE ACID	Used to assign profiles to an ACID.
PASSWORD	Used to assign a password, along with values that control its use, to a previously defined ACID.

**Table 2—Top Secret Connector - Account Attributes (Continued)**

Attribute	Description
PASSWORD EXPIRES DATE	The date, in string format, that the password expires.
PASSWORD INTERVAL	The interval in which the password must be changed.
PASSWORD FACILITY	The facility name applied to ACIDs with the multipw attribute.
OPIDENT	Used to assign or remove a CICS operator identification value that is equal to the ACID's OPIDENT entry in the CICS SNT (Signon Table). The OPIDENT value is placed into the ACID's TCT at signon.
OPPRTY	Used to assign or remove a CICS operator priority from the associated ACID. The OPPRTY value is placed into the ACID's TCT (Terminal Control Table) at signon.
PROGRAM	Used to secure system programs and utilities.
WHOHAS ADMIN	Used to determine who has administrative authority on the application.
ACIDS2	
SOURCES	
TSOLPROC	Used to provide a default procedure to be used for TSO logon.  The one- to eight-character logon procedure name. Procedure names are also TSO-related resources and the user must be permitted to any procedure name with which he attempts to log on.
DIV ACID	Specifies the Division ACID to which the ACID is attached.
DIV NAME	The name assigned to the ACID within the zone.
SUSPENDED	The date, in string format, that the suspension ends.
WHOHAS XAUTH	A list of resources that may be accessed by the ACID shown in the command, the level at which the ACID may access the resource, and the owner of the resource.
TSOUNIT	The default unit name to be used for dynamic allocations under TSO.  The one- to eight-character unit (device) name for dynamically allocated data sets. The name must be a defined generic unit class name at the installation. This field is not alterable by the user at logon and is not required for successful logon.
PHYSKEY	PHYSKEY (physical security key) supports external authentication devices.
ACID WITHIN DEPT/DIV/ZONE	Used to specify department, division, and zone to include.
DATE CREATED	The date on which the ACID was created.
DATE LAST MODIFIED	The date on which the ACID was last modified.
TIME LAST MODIFIED	The time at which the ACID was last modified.
ROOM NUMBER	The room number assigned to the ACID.
MISC2	Used to give, or to remove, a CA-Top Secret administrator's authority to perform one or more administrative functions.
ACCESSLEVELS	

## Schema Attributes

**Table 2—Top Secret Connector - Account Attributes (Continued)**

Attribute	Description
MISC8	Used to give, or to remove, a CA-Top Secret administrator's authority to list the contents of the RDT, FDT or STC or to use the ASUSPEND administrative function.
XA MINIDISK	The minidisk authorization information for the ACID.
SCOPE	Used to give CA-Top Secret administrators the authority, or to remove their authority, to assign the SCOPE of an LSCA.
DIGITAL CERT NAME	The name of the digital certification.
DEPARTMENT	The Department ACID to which the ACID is attached.
DLFTGRP	The default group for the ACID.
WHO OWNS RESOURCE	The resources owned by the ACID.
TSOOPT	The default options that a TSO user may specify at logon.
WANAME	The person to whom SYSOUT information should be delivered for this ACID.
XA	
SYSID	The SYSID (which is actually the SMFID) that the authorizations for the ACID apply to.
BUILDING	The building in which the ACID is located.
TSOCOMMAND	Default commands issued upon login of the ACID.
DIGITAL CERT STARTS	The date, in string format, that the digital certification starts.
XAUTH LIBRARY	The libraries for which the ACID has authority.
WHOHAS FACILITY	Returns facility information for the ACID.
RESOURCE CLASS NAME	The resource class for which the ACID has authority.
FCT/PREFIX(OWNED)	
FACILITIES	The facilities to which the user has access.
TSOHCLASS	The default hold class for TSO generated JCL for TSO the user.
DIGITAL CERT EXPIRES	The date, in string format, when the digital certification expires.
ZONE ACID	The Zone ACID to which the ACID is attached.
ZONE NAME	The name assigned to the ACID within the zone.
ADDRESS1	Physical address for the ACID.
XAUTHDAYS	Days of the week the ACID is authorized on this application.
ACID TYPE	The ACID type, for example zone, division, or department.
ACID SIZE	The size of the ACID.
RESTRICT	
ADDRESS4	Alternative physical address for the ACID.
NODSNCHK	To specify that no data set name check will be performed. That is, CA-Top Secret will bypass all data set access security checks. All data set access will be audited.

**Table 2—Top Secret Connector - Account Attributes (Continued)**

Attribute	Description
NOVOLCHECK	
NOLCFCHK	Used to allow an ACID to execute any command or transaction for all facilities, regardless of LCF (Limited Command Facility) restrictions. No auditing is done.
NOSUBCHK	Used to allow an ACID to bypass alternate ACID usage as well as all job submission security checking. Thus, associated ACIDs may submit all jobs regardless of the (derived) ACID on the job card being submitted.
NORESCHK	Used to allow an ACID to bypass security checking, including auditing, for all owned resources except data sets and volumes.
NOVMDCHK	Used to allow an ACID to bypass all checking for minidisk links. All links will be audited.  NOVMDCHK is intended only to be applied to special products such as DASD space managers, which may link to many minidisks.
NOSUSPEND	Used to allow an ACID to bypass suspension due to violations.
TSODEST	The default destination identifier for TSO generated JCL for TSO users.
XA VOLUMN	
TSODEFPRFG	The default TSO performance group.
RESOURCE CLASS NAME2	An additional resource class for which the ACID has authority.
MISC1	A CA-Top Secret administrator's authority to perform one or more administrative functions.
GID	IDs of the groups to which the ACID belongs.
TSOUDATA	The site-defined data fields for a TSO user.
ACCESSLEVELS2	
DSN/PREFIX(OWNED)	
TSOMSIZE	The maximum region size (in kilobytes) that the TSO user can specify at logon.
EXPIRES	The date on which the ACID expires.
TSOSCLASS	The default SYSOUT class for TSO generated JCL for the TSO users.
XAUTH FAC	
DEPT ACID	The Department ACID to which the ACID is attached.
DEPT NAME	The name assigned to the ACID within the department.
DATE LAST USED	Date the ACID was last used.
TIME LAST USED	Time the ACID was last used.
CPU	Name of the CPU on which the ACID was used.
FAC	System facilities defined to CA-Top Secret: BATCH, STC, TSO, IMS, CICS, NCCF, CA-Roscoe, WYLBUR, or any installation-defined facility.
COUNT	

## Schema Attributes

**Table 2—Top Secret Connector - Account Attributes (Continued)**

Attribute	Description
SEGMENT	Used to allow TSS administrators to list data about fields in a specific segment.
RESOURCES	
TSOJCLASS	The default job class for TSO generated job cards from TSO users.
ADMIN BY	
XAUTH MODE	
TSOLACCT	The default account number used for TSO logon.
TSOLSIZE	The default region size (in kilobytes) for TSO.
LISTDATA	
OMVSPGM	The user's OpenEdition MVS shell program. This is the first program started when the OMVS command is entered, or when an OpenEdition MVS batch job is started using the BPXBATCH program.
SMSSTOR	The default storage keyword for the ACID.
UID	The unique user ID for the ACID.
ADDRESS3	Alternative physical address for the ACID.
XAUTH PRIVPGM	The program pathing, if privileged program is in use.
TIME ZONE	The time zone attached to the ACID.
MASTER FACILITY	
LCF FACILITY	
FACILITY NAME	
FACILITY UNTIL DATE	
INSTDATA	Used to record or remove information about an ACID. Up to 255 characters of information about an associated ACID may be used for convenient record keeping, or for interrogation by a user-written Installation Exit.
ADDRESS2	Alternative physical address for the ACID.
GROUP ACID	The Group ACID to which the ACID is attached.
TSOMCLASS	The default message class for TSO generated JCL for TSO users.
MISC9	To give, or to remove, a TSS administrator's authority to perform one or more high-level administrative functions.

**Table 3—Top Secret Connector - Group Attributes**

Attribute	Description
XAUTH	The authorized level at which the user can access the resource.
VMMMDISK	The VM minidisks owned by the user.
ACTION	Specifies which action(s) CA-Top Secret will take when access to a resource is attempted.

**Table 3—Top Secret Connector - Group Attributes (Continued)**

Attribute	Description
LOCK TIME(MINUTES)	The time interval before unattended or inactive terminals are locked.
LOCK TIME FACILITY	The lock time for all terminals connected to the specified facility.
LANGUAGE PREFERENCE	The language preference code the user.
VOLSER(OWNED)	The volumes to which the user has access.
NAME	Identifies the ACID name.  Names can be up to 32 characters in length, must be surrounded by single quotes if embedded with blanks, and can use letters, numbers, and special characters.
SITRAN	Specifies which CICS transaction CA-Top Secret automatically executes after an ACID successfully signs on to a facility.  <b>Note:</b> If a SITRAN is added to an ACID that already has a CICS transaction defined, the transaction is replaced.
HOME	Defines the initial directory pathname. This is the initial directory used when a user enters the OMVS command or enters the ISPF shell. The HOME keyword accepts from one to 1024 characters. Both uppercase and lowercase characters are allowed. If HOME isn't defined, OpenEdition MVS sets the initial directory for the user to the root directory. HOME is optional.
XA ACID	XAUTH Resource Class Name
MULTIPW	Used to assign or remove multiple password attributes, which means ACIDs need a different password to access each facility.
NOADSP	Used to prevent data sets, created by an ACID, from being automatically secured by MVS by setting the RACF bit.  NOADSP is used to define an ACID that will be used to create data sets that cannot be automatically protected by CA-Top Secret.
AUDIT	Used to allow an audit of ACID activity.
NOPWCHG	To prevent ACIDs from changing passwords at either signon or initiation.
OIDCARD	Used to support the physical identification of users through operator identification cards.
TRACE	Used to activate a diagnostic trace on all ACID activity (initiations, resource access, violations, user's security mode, etc.)
SUSPEND	Used to prevent ACIDs from accessing the system when a violation occurs.
MRO	Used to support the use of the multi-region option.
CONSOLE	Used to grant or remove an ACID's ability to modify control options. For VM, options are modified via the TSS MODIFY command only. With VSE and OS/390, options are modified at the O/S console or via the TSS MODIFY command function.
GAP	Used to specify that a profile will become, or will cease to be, globally administrable.

## Schema Attributes

**Table 3—Top Secret Connector - Group Attributes (Continued)**

Attribute	Description
DUFXTR	Used to add or remove the DUFXTR attribute to an ACID. DUFXTR enables an ACID to use a RACROUTE REQUEST=AUTH (RACHECK) macro or the CA-Top Secret Application Interface to extract installation data (INSTDATA) or field data from a Security Record. DUFXTR is a component of the CA-Top Secret Dynamic Update Facility (DUF).
DUFUPD	Used to add or remove the DUFUPD attribute to an ACID. DUFUPD enables an ACID to use the CA-Top Secret Application Interface to update the installation data (INSTDATA) or field data from a Security Record. DUFUPD is a component of the CA-Top Secret Dynamic Update Facility (DUF).
TSOMPW	Used to support multiple TSO UADS passwords, on a user-by-user basis.
NOATS	Used to prevent an ACID in CICS and CA-IDMS from signing on via ATS (Automatic Terminal Signon).
ACEDEFAU	
ASUSPEND	Used to remove the suspension of an ACID that was suspended for administrative reasons.
XA DATASET	
WHO HAS RESOURCE	
PROFILE ACID	Used to assign profiles to an ACID.
PASSWORD	Used to assign a password, along with values that control its use, to a previously defined ACID.
PASSWORD EXPIRES DATE	The date, in string format, that the password expires.
PASSWORD INTERVAL	The interval in which the password must be changed.
PASSWORD FACILITY	The facility name applied to ACIDs with the multipw attribute.
OPIDENT	Used to assign or remove a CICS operator identification value that is equal to the ACID's OPIDENT entry in the CICS SNT (Signon Table). The OPIDENT value is placed into the ACID's TCT at signon.
OPPRTY	Used to assign or remove a CICS operator priority from the associated ACID. The OPPRTY value is placed into the ACID's TCT (Terminal Control Table) at signon.
PROGRAM	Used to secure system programs and utilities.
WHOHAS ADMIN	Used to determine who has administrative authority on the application.
ACIDS2	
SOURCES	
TSOLPROC	Used to provide a default procedure to be used for TSO logon.  The one- to eight-character logon procedure name. Procedure names are also TSO-related resources and the user must be permitted to any procedure name with which he attempts to log on.
DIV ACID	Specifies the Division ACID to which the ACID is attached.

**Table 3—Top Secret Connector - Group Attributes (Continued)**

Attribute	Description
DIV NAME	The name assigned to the ACID within the zone.
SUSPENDED	The date, in string format, that the suspension ends.
WHOHAS XAUTH	A list of resources that may be accessed by the ACID shown in the command, the level at which the ACID may access the resource, and the owner of the resource.
TSOUNIT	The default unit name to be used for dynamic allocations under TSO.  The one- to eight-character unit (device) name for dynamically allocated data sets. The name must be a defined generic unit class name at the installation. This field is not alterable by the user at logon and is not required for successful logon.
PHYSKEY	PHYSKEY (physical security key) supports external authentication devices.
ACID WITHIN DEPT/DIV/ZONE	Used to specify department, division, and zone to include.
DATE CREATED	The date on which the ACID was created.
DATE LAST MODIFIED	The date on which the ACID was last modified.
TIME LAST MODIFIED	The time at which the ACID was last modified.
ROOM NUMBER	The room number assigned to the ACID.
MISC2	Used to give, or to remove, a CA-Top Secret administrator's authority to perform one or more administrative functions.
ACCESSLEVELS	
MISC8	Used to give, or to remove, a CA-Top Secret administrator's authority to list the contents of the RDT, FDT or STC or to use the ASUSPEND administrative function.
XA MINIDISK	The minidisk authorization information for the ACID.
SCOPE	Used to give CA-Top Secret administrators the authority, or to remove their authority, to assign the SCOPE of an LSCA.
DIGITAL CERT NAME	The name of the digital certification.
DEPARTMENT	The Department ACID to which the ACID is attached.
DLFTGRP	The default group for the ACID.
WHO OWNS RESOURCE	The resources owned by the ACID.
TSOOPT	The default options that a TSO user may specify at logon.
WANAME	The person to whom SYSOUT information should be delivered for this ACID.
XA	
SYSID	The SYSID (which is actually the SMFID) that the authorizations for the ACID apply to.
BUILDING	The building in which the ACID is located.
TSOCOMMAND	Default commands issued upon login of the ACID.
DIGITAL CERT STARTS	The date, in string format, that the digital certification starts.

## Schema Attributes

**Table 3—Top Secret Connector - Group Attributes (Continued)**

Attribute	Description
XAUTH LIBRARY	The libraries for which the ACID has authority.
WHOHAS FACILITY	Returns facility information for the ACID.
RESOURCE CLASS NAME	The resource class for which the ACID has authority.
FCT/PREFIX(OWNED)	
FACILITIES	The facilities to which the user has access.
TSOCLASS	The default hold class for TSO generated JCL for TSO the user.
DIGITAL CERT EXPIRES	The date, in string format, when the digital certification expires.
ZONE ACID	The Zone ACID to which the ACID is attached.
ZONE NAME	The name assigned to the ACID within the zone.
ADDRESS1	Physical address for the ACID.
XAUTHDAYS	Days of the week the ACID is authorized on this application.
ACID TYPE	The ACID type, for example zone, division, or department.
ACID SIZE	The size of the ACID.
RESTRICT	
ADDRESS4	Alternative physical address for the ACID.
NODSNCHK	To specify that no data set name check will be performed. That is, CA-Top Secret will bypass all data set access security checks. All data set access will be audited.
NOVOLCHECK	
NOLCFCHK	Used to allow an ACID to execute any command or transaction for all facilities, regardless of LCF (Limited Command Facility) restrictions. No auditing is done.
NOSUBCHK	Used to allow an ACID to bypass alternate ACID usage as well as all job submission security checking. Thus, associated ACIDs may submit all jobs regardless of the (derived) ACID on the job card being submitted.
NORESCHK	Used to allow an ACID to bypass security checking, including auditing, for all owned resources except data sets and volumes.
NOVMDCHK	Used to allow an ACID to bypass all checking for minidisk links. All links will be audited.  NOVMDCHK is intended only to be applied to special products such as DASD space managers, which may link to many minidisks.
NOSUSPEND	Used to allow an ACID to bypass suspension due to violations.
TSODEST	The default destination identifier for TSO generated JCL for TSO users.
XA VOLUMN	
TSODEFPRFG	The default TSO performance group.
RESOURCE CLASS NAME2	An additional resource class for which the ACID has authority.

**Table 3—Top Secret Connector - Group Attributes (Continued)**

Attribute	Description
MISC1	A CA-Top Secret administrator's authority to perform one or more administrative functions.
GID	IDs of the groups to which the ACID belongs.
TSOUDATA	The site-defined data fields for a TSO user.
ACCESSLEVELS2	
DSN/PREFIX(OWNED)	
TSOMSIZE	The maximum region size (in kilobytes) that the TSO user can specify at logon.
EXPIRES	The date on which the ACID expires.
TSOSCLASS	The default SYSOUT class for TSO generated JCL for the TSO users.
XAUTH FAC	
DEPT ACID	The Department ACID to which the ACID is attached.
DEPT NAME	The name assigned to the ACID within the department.
DATE LAST USED	Date the ACID was last used.
TIME LAST USED	Time the ACID was last used.
CPU	Name of the CPU on which the ACID was used.
FAC	System facilities defined to CA-Top Secret: BATCH, STC, TSO, IMS, CICS, NCCF, CA-Roscoe, WYLBUR, or any installation-defined facility.
COUNT	
SEGMENT	Used to allow TSS administrators to list data about fields in a specific segment.
RESOURCES	
TSOJCLASS	The default job class for TSO generated job cards from TSO users.
ADMIN BY	
XAUTH MODE	
TSOLACCT	The default account number used for TSO logon.
TSOLSIZE	The default region size (in kilobytes) for TSO.
LISTDATA	
OMVSPGM	The user's OpenEdition MVS shell program. This is the first program started when the OMVS command is entered, or when an OpenEdition MVS batch job is started using the BPXBATCH program.
SMSSTOR	The default storage keyword for the ACID.
UID	The unique user ID for the ACID.
ADDRESS3	Alternative physical address for the ACID.
XAUTH PRIVPGM	The program pathing, if privileged program is in use.
TIME ZONE	The time zone attached to the ACID.

## Schema Attributes

**Table 3—Top Secret Connector - Group Attributes (Continued)**

Attribute	Description
MASTER FACILITY	
LCF FACILITY	
FACILITY NAME	
FACILITY UNTIL DATE	
INSTDATA	Used to record or remove information about an ACID. Up to 255 characters of information about an associated ACID may be used for convenient record keeping, or for interrogation by a user-written Installation Exit.
ADDRESS2	Alternative physical address for the ACID.
GROUP ACID	The Group ACID to which the ACID is attached.
TSOMCLASS	The default message class for TSO generated JCL for TSO users.
MISC9	To give, or to remove, a TSS administrator's authority to perform one or more high-level administrative functions.

# Chapter 33: SailPoint UNIX Connector

---

The following topics are discussed in this chapter:

Overview .....	305
Supported features .....	305
Configuration parameters.....	305
Schema attributes .....	306

## Overview

---

The SailPoint UNIX Connector is a *read only* connector developed to read and parse the **passwd** and **group** file from UNIX servers to build identities and groups. Since this connector is file based, there is some synergy between the UNIX and Delimited File connector.

Depending on your application configuration, the SailPoint UNIX Connector determines login success by authenticating using the ftp or scp service with the provided login credentials. Therefore, the **passwdfile** attribute of the UNIX application must be the same password file used by the system for authentication. This password file is typically **/etc/passwd**, but might be different in an environment where NIS is used.

## Supported features

---

SailPoint UNIX Connector supports the following features:

- Account Management
  - Manages UNIX Users as Accounts
  - Aggregation
- Account - Group Management
  - Manages UNIX Groups as Account-Groups
  - Aggregation

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The UNIX Database connector uses the following connection attributes:

**Table 1—UNIX Connector - Configuration parameters**

Parameters	Description
host	The host of the server to which you are connecting.

## Schema attributes

**Table 1—UNIX Connector - Configuration parameters**

Parameters	Description
filetransport	local, ftp, scp
transportUser	The user to use with ftp and scp. Not valid with local.
transportUserPassword	The password to use with of ftp and scp. Not valid with local.
passwdfile	The fully qualified path to the <code>passwd</code> file.
groupfile	The fully qualified path to the <code>group</code> file.

## Schema attributes

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports two types of objects, account and group.

### Account attributes

Account objects are used when building identities Link objects.

**Table 2—UNIX Connector - Account Attributes**

Attribute	Description
homedir	The path to the user's home directory on the host system. The home directory is the directory in which the user keeps personal files such as initialization files and mail.
shell	The shell, or program, preferred by the user for accessing the command line interface.
info	The information pertaining to the user.
groups	The groups to which the user belongs.

### Group attributes

The group schema is used when building AccountGroup objects which are used to hold entitlements shared across identities.

**Table 3—UNIX Connector - Group Attributes**

Attribute	Description
groupname	A name associated with the group. The group names are listed in the first comma-delimited field of the groups text file.
groupid	A group id used to identify the group. The group ids are listed in the third comma-delimited field of the groups text file.

**Table 3—UNIX Connector - Group Attributes**

Attribute	Description
members	A comma-delimited list of users who are members of the group. Members are listed in the forth comma-delimited field of the groups text file.

## **Schema attributes**

# Chapter 34: SailPoint Web Services Connector

---

The following topics are discussed in this chapter:

Overview .....	309
Supported features .....	309
Supported Managed Systems .....	310
Pre-requisites .....	310
Administrator permissions .....	310
Configuration parameters .....	310
Schema attributes .....	312
Additional information .....	312
Pagination .....	312
Configuration for Response .....	315
Configuration for Multiple endpoints .....	316
Other Operations .....	317

## Overview

---

The Web Services Connector is developed with an idea where any RESTful Web Service supported end managed system can be configured. This connector will be able to perform read and write operation on the end managed system using the respective end managed system REST API.

**Note:** Web Services Connector only supports JSON for read and write.

## Supported features

---

SailPoint Web Services Connector supports the following features:

- Account Management
  - Aggregation, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable, Change Password
  - Add/Remove Entitlements
- Account - Group Management
  - Aggregation, Refresh Groups

## Additional supported feature

SailPoint Web Services Connector provides additional support for pagination.

For more information on embedding pagination support in Web Service Connector, see “Pagination” on page 312.

## Supported Managed Systems

---

SailPoint Web Services Connector supports RESTful Web Service with JSON response only.

### Pre-requisites

---

RESTful Web Services must be accessible.

### Administrator permissions

---

The user/administrator must have the required permissions to call the RESTful Web Services API of the end managed system.

## Configuration parameters

---

This section provides the following type of configuration parameters of SailPoint Web Services Connector:

- Basic configuration parameters
- Operation specific configuration parameters

### (General Settings) Basic configuration parameters

---

The following table lists the basic configuration parameters of SailPoint Web Services Connector:

Parameters	Description
Base URL*	The base URL to connect to the RESTful Web Service end managed system.
Authentication Type*	Authentication method that is supported by the end managed system <ul style="list-style-type: none"><li>• BasicLogin (username and password) <b>Or</b> • OAuthLogin (a bearer token using the “Bearer” header)</li></ul>
Username*	Username that holds permission to execute RESTful Web Service Server. <b>Note: Required if the ‘Authentication Type’ is selected as ‘Basic’.</b>
Password*	Password for the RESTful Web Service Server. <b>Note: Required if the ‘Authentication Type’ is selected as ‘Basic’.</b>
Access Token*	The OAuth bearer token to use for authorization. <b>Note: Required if the ‘Authentication Type’ is selected as ‘OAuthLogin’.</b>
Schema Attribute for Account Enable status	Attribute name and value required to be provided to check the Enable status. For example, status=Active
Request Timeout (In Seconds)	Request Timeout Value in seconds.

**Note:** Attributes marked with \* sign are the mandatory attributes.

## (Connector Operations) Operation specific configuration parameters

---

**Note:** No default provisioning template is provided. The template may vary from one RESTful end managed system to another.

Perform the following procedure to add and configure the specific operations:

1. Click **Add Operation**.
2. Select the operation from the drop down list of **Choose Operation**.
3. Provide a unique name to the operation. For example: Account Aggregation-1
4. Select the configure option (Pencil image) on the same row.  
Allows user to provide additional options.

The following table lists the operation specific configuration parameters of SailPoint Web Services Connector:

Parameters	Description
ContextURL	Context URL specific to the operation.  For example, <code>/api/core/v3/securityGroups?startIndex=0&amp;count=100&amp;fields=%40all&amp;sort=lastNameAsc</code>
Method	Select one of the following type of HTTP method supported by the respective operation: <ul style="list-style-type: none"> <li>• GET</li> <li>• PUT</li> <li>• POST</li> <li>• DELETE</li> </ul>
Header	Header information in the form of key and value.  For example, Content Type = Application/json, Authorization = Bearer <ACCESS-TOKEN>
Body	Standard http body used to post data with request. User can send data in either of the following format: <ul style="list-style-type: none"> <li>• <b>form-data:</b> Key value. User must set the data that has to pass in the key value</li> <li>• <b>raw:</b> Data to be sent in JSON format. For example,               <pre>{                 "limit": 10,                 "cursor": "abcd1234"               }</pre> </li> </ul>

## Schema attributes

Parameters	Description
Response	<p><b>Response Attribute Mapping:</b> Used to map the respective operations, JSON path in the response to the particular field present in the JSON response.</p> <ul style="list-style-type: none"><li>• Schema Attribute: Attributes/Fields expected in the response from the particular url. For example, <code>member_id</code></li><li>• JSON Path: JSON path of the particular attribute in the returned JSON response. For example, <code>members[*].profile.member_id</code> User must mention the JSON path after the Root Path.</li></ul>
	<p><b>Root Path:</b> Common path present in the JSON response. For example, <code>\$.members.profile</code></p> <p><b>Note:</b> It must be common for all the above attribute mentioned in the Response Attribute Mapping</p>
	<p><b>Successful Response Code:</b> Successful response code expected by the respective RESTful Web Service operation.</p> <p>This field accepts HTTP status code in csv format (For example, 200, 201, 203). If the list does not contain any value, the status code from 200 to 299 would be checked.</p> <p>There could be situation where successful status code may start with 2, in this situation user can provide 2**.</p>
Before Rule	<b>Before Operation Rule:</b> Rule that will be invoked before performing any operation (account aggregation, enable, disable account and so on).
After Rule	<b>After Operation Rule:</b> Rule that will be invoked after performing any operation (account aggregation, enable, disable account and so on)

**Note:** For more information on operation specific configurations, see “Additional information” on page 312.

## Schema attributes

Discover schema functionality is not available. Hence user must add the schema attributes manually for the respective RESTful Web Service based end managed system.

## Additional information

This section describes the additional information related to the Web Services Connector.

### Pagination

To embed pagination in Web Service Connector, manual processing is required in BEFORE and AFTER operation rules of Web Service Connector.

1. The Web Service Connector relies on a temporary information stored in the application object in form of a map which has the name as **transientValues**.

2. The administrator must write the Before Rule and AFTER Rule for account/group aggregation as follows:

- **Web Service Before Rule:** The Before Rule alters the URL/request parameters if the value of the **hasMore** parameter is set to **TRUE** and the request to fetch further accounts is triggered. If **hasMore** parameter is not set or is set to **FALSE** the pagination request would be terminated.

For example, see sample Before Rule for account aggregation request in Web Services Connector for Dropbox using Rest APIs V2 in **examplerules.xml** file by name **Example WSBeforeRI DropboxPaging** as follows:

```

import sailpoint.tools.Util;

Map obj = (Map) application.getAttributeValue("transientValues");
System.out.println("BEFORE RULE: Transient Values ==> " + obj);
if(null != obj) {
    String offset = obj.get("offset");
    System.out.println("BEFORE RULE: offset value ==> " + offset);
    String urlString = (String) requestEndPoint.getFullUrl();
    if(Util.isNotEmpty(offset)) {
        System.out.println("BEFORE RULE: requestEndpoint ==> " + requestEndPoint);
        System.out.println("BEFORE RULE: URL ==> " + urlString);
        URL tempUrl = new URL(urlString);
        String queryString = tempUrl.getQuery();
        System.out.println("BEFORE RULE: Query String ==> " + queryString);

        if(Util.isNotEmpty(queryString)) {
            StringBuffer queryParams = new StringBuffer();
            String[] params = tempUrl.getQuery().split("&");
            for (String param : params) {
                if(queryParams.length() > 0)
                    queryParams.append("&");
                if(param.startsWith("sysparm_offset=")) {
                    queryParams.append("sysparm_offset=");
                    queryParams.append(offset);
                } else {
                    queryParams.append(param);
                }
            }
            urlString = urlString.replace(tempUrl.getQuery(), queryParams.toString());
        }
    }
    System.out.println("BEFORE RULE: Updated Query String ==> " + urlString);
    requestEndPoint.setFullUrl(urlString);
}
System.out.println("BEFORE RULE: requestEndpoint Updated ==> " + requestEndPoint);
return requestEndPoint;

```

In case of Dropbox V2, the **cursor** returned from the previous team membership listing API would be stored in the **transientValues** map in the application by the Web Service AFTER Rule. The REST API url is modified to direct to the paging API and the cursor would be sent as a part of the form data. Ensure that the **hasMore** flag is set by the earlier requests AFTER RULE

- **Web Service After Rule:** The AFTER Rule deduces whether the managed system has more records which can be fetched and added as an entry in **transientValues** with **hasMore** key and value as TRUE/FALSE depending upon the condition deduced.

For example, see sample AFTER Rule for account aggregation request in Web Services Connector for Dropbox using Rest APIs V2 in **examplerules.xml** by name **Example WSAfterRI DropboxPaging** as follows:

## Additional information

```
Integer fetchedRecordsCount = 0;
if(null != processedResponseObject) {
    fetchedRecordsCount = ((List) processedResponseObject).size();
}

Integer expectedCount = null;
Integer offset = null;
URL url = new URL(requestEndPoint.getFullUrl());
System.out.println("AFTER RULE: Original Url ==> " + url);
String[] params = url.getQuery().split("&");
for (String param : params) {
    String name = param.split("=")[0];
    String value = param.split("=")[1];

    switch(name) {
        case "sysparm_limit":
            expectedCount = Integer.parseInt(value);
            break;

        case "sysparm_offset":
            offset = Integer.parseInt(value);
            break;

        default:
    }
}

System.out.println("AFTER RULE: Fetch Count ==> " + fetchedRecordsCount);
System.out.println("AFTER RULE: Limit Count ==> " + expectedCount);
System.out.println("AFTER RULE: Fetch Offset ==> " + offset);

boolean hasMore = (fetchedRecordsCount != 0 && null != expectedCount &&
fetchedRecordsCount.equals(expectedCount) && null != offset);
System.out.println("AFTER RULE: Has More? ==> " + hasMore);

Map transientValues = application.getAttributeValue("transientValues");
if(transientValues == null) {
    transientValues = new HashMap();
    application.setAttribute("transientValues", transientValues);
}
transientValues.put("hasMore", hasMore);
if (hasMore) {
    if(null != offset) {
        System.out.println("AFTER RULE: New Offset ==> " + (offset + expectedCount));
        transientValues.put("offset", String.valueOf(offset + expectedCount));
    }
}
```

In case of Dropbox, Dropbox V2 REST APIs for team membership response contain the following elements:

- **cursor**: is an encrypted token which represents the next page to be fetched, if any, and would form part of the subsequent API calls.
- **has\_more**: is a boolean value which explicitly indicates whether more records are available for fetching.

AFTER Rule stores the **cursor** and **has\_more** values from the response in the **transientValues** map in the Application object. This map stores the necessary information which would be used by the BEFORE RULE to manipulate the next API call. Ensure that the flag indicating whether the managed system contains more

records is stored by the key named **hasMore**. This field is mandatory as it is the deciding factor for aborting the pagination requests.

## Configuration for Response

---

When configuring the Web Services application, map the schema attribute with JSON as explained in the following example:

**Figure 1—Example for mapping the schema attributes with JSON**

```
"list": [
    {
        "id": "2124",
        "resources": {
            "securityGroups": {
                "ref": "https://mydomain.jive.com/api/core/v3/people/2124/securityGroups"
            }
        },
        "displayName": "Bill Jackson",
        "emails": [
            {
                "value": "bill.jackson@mydomain.com",
            }
            {
                "value": "admin@mydomain.com",
            }
        ],
        "jive": {
            "enabled": true,
            "level": {
                "name": "Level 0",
            },
            "username": "bill.jackson",
        },
    },
]
```

In the above JSON response, all the attributes can be mapped as follows considering **Root Path** as **\$.list**:

```
Id = id
displayName=displayName
username=jive.username
enabled =jive.enabled
emails=emails[*].value
```

## Additional information

Figure 2—Mapped schema attributes

Schema Attribute <span style="color: #0070C0;">?</span>	JSON Path <span style="color: #0070C0;">?</span>
id	id
enabled	jive.enabled
username	jive.username
emails	emails[*].value
displayName	displayName
Root Path <span style="color: #0070C0;">?</span>	
\$.list	
Successful Response Code <span style="color: #0070C0;">?</span>	
2**	

## Configuration for Multiple endpoints

Perform the following to obtain the properties of account/group from multiple endpoints:

1. The basic attribute is obtained from the first endpoint and is then used for fetching the data from rest of the endpoints.  
For example, during aggregation of Jive some attributes are obtained from first endpoint (“Figure 2—Mapped schema attributes”) using the following URL:  
**[https://myDomain.jive.com /api/core/v3/people](https://myDomain.jive.com/api/core/v3/people)**
2. To fetch additional attribute from another endpoint use the `id` attribute from the previous response. Add these attributes in Schema Attribute of Response Attribute Mapping and response as follows:
  - **Schema Attribute**

Response Attribute Mapping	
Schema Attribute <span style="color: #0070C0;">?</span>	JSON Path <span style="color: #0070C0;">?</span>
Email	profile.email
groups	profile.groups

- **Response:** The following context URL contains `id` which fetches all the groups connected to that account:

**[https://myDomain.jive.com/api/core/v3/people/\\$response.id\\$/securityGroups](https://myDomain.jive.com/api/core/v3/people/$response.id$/securityGroups)**

## Other Operations

For certain operations, the Body must be updated accordingly.

### Create Account

This section provides an example for updating the Body for create account in Dropbox. For fetching attribute through Provisioning Plan, the body must be updated in the following manner. This fetches the attribute detail through Provisioning Form and updates the endpoint.

```
Body

 form-data  raw

{
  "member_email" : "$plan.member_email$",
  "member_given_name" : "$plan.member_given_name$",
  "member_surname" : "$plan.member_surname$",
  "send_welcome_email" : "$plan.send_welcome_email$",
  "member_external_id" : "$plan.member_external_id$"
}
```

In the above Body,

- **\$plan** represents the Provisioning Plan that is passed to provision method
- **\$plan.member\_surname**: the connector checks for **member\_surname** in the attribute request and updates in the body after it is found

### Enable/Disable

Set the get object endpoint for enable/disable operation as in the POST method the complete object would be required to update and not single attribute. Hence first endpoint getObject would fetch the whole account and later the endpoint would update the payload with all the required attributes using the response of the first endpoint.

Perform the following steps to get object for Enable operation with PUT method

1. Configure the first endpoint to get object for Enable.

		Enable Account	▼	Enable ONE - getObject	Configured		
		Enable Account	▼	Enable TWO - enable	Configured		

2. Configuration for the first endpoint.

## Additional information

Context URL <a href="#">?</a>	Method <a href="#">?</a>	
/api/core/v3/people/\$getObject.nativeIdentity\$	GET	
<b>Header</b>	Response Attribute Mapping	
<b>Body</b>	Schema Attribute <a href="#">?</a>	JSON Path <a href="#">?</a>
<b>Response</b>	region	addresses[0].value.region
<b>Before Rule</b>	id	id
<b>After Rule</b>	Email	emails[0].value
	status	jive.enabled
	street	addresses[0].value.streetAddress
	name	displayName
	familyName	name.familyName
	givenName	name.givenName
	type	type
	displayName	displayName
	country	addresses[0].country

This endpoint retrieves getObject for account for which Provisioning Operation is performed.

3. Configuration for second endpoint for Enable endpoint as shown in the following figure:

Context URL <a href="#">?</a>	Method <a href="#">?</a>
/api/core/v3/people/\$plan.nativeIdentity\$	PUT
<b>Header</b>	Body
<b>Body</b>	<input checked="" type="radio"/> form-data <input type="radio"/> raw
<b>Response</b>	
<b>Before Rule</b>	
<b>After Rule</b>	

```
{
  "emails": [
    {
      "value": "$response.Email$",
      "jive_label": "Email"
    }
  ],
  "jive": {
    "enabled": "true",
    "federated": "false",
    "visible": "true",
    "username": "$response.Email$"
  },
  "name": {
    "formatted": "$response.givenName$",
    "familyName": "$response.familyName$",
    "givenName": "$response.givenName$"
  }
}
```

**Note:** It may be required to update few attribute for performing enable/disable operation

Similar steps are to be performed for Disable operation.

## Add/Remove Entitlement

Following is an example of the Body entry for Add Entitlement:

The screenshot shows a configuration interface with the following fields:

- Context URL:** /1/team/groups/members/add
- Method:** POST
- Header:** (This section is empty)
- Body:**
  - Content Type:** form-data (radio button selected)
  - Raw Content:**

```
{
  "group_id": "$plan.groups$",
  "members": [
    { "team_member_id": "$plan.nativeIdentity$",
      "access_type": "member" }
  ]
}
```
- Response:** (This section is empty)
- Before Rule:** (This section is empty)
- After Rule:** (This section is empty)

On similar basis as above example the Body entry must be updated for Remove Entitlement.

## Update Account

Following is an example of the Body entry for Update Account:

The screenshot shows a configuration interface with the following fields:

- Context URL:** /1/team/members/set\_profile
- Method:** POST
- Header:** (This section is empty)
- Body:**
  - Content Type:** form-data (radio button selected)
  - Raw Content:**

```
{
  "member_id": "$plan.nativeIdentity$",
  "new_given_name": "$plan.given_name$",
  "new_email": "$plan.email$"
}
```
- Response:** (This section is empty)
- Before Rule:** (This section is empty)
- After Rule:** (This section is empty)

## Delete Account

Following is an example of the Body entry for Delete Account:

## Additional information

Context URL [?](#) /1/team/members/remove Method [?](#) POST

Header	Body
Body	
Response	
Before Rule	<input checked="" type="radio"/> form-data <input type="radio"/> raw
After Rule	{ "member_id": "\$getobject.nativelidentity\$" }

## Change Password

Following is an example of the Body entry for Change Password:

Details Configuration Correlation Accounts Risk Activity Data Sources Rules Password Policy

Settings Schema Provisioning Policies

Back Connection Settings

Context URL [?](#) /api/now/v1/import/x\_sapo\_iq\_connect\_sysuser Method [?](#) POST

Header	Body
Body	
Response	
Before Rule	<input checked="" type="radio"/> form-data <input type="radio"/> raw
After Rule	{"user_sys_id": "\$plan.nativelidentity\$", "password": "\$plan.password\$", "password_needs_reset": "false"}

[Cancel](#) [Save](#)

# Chapter 35: SailPoint Windows Local Connector

---

The following topics are discussed in this chapter:

Overview .....	321
Supported features .....	321
Supported Managed Systems .....	322
Pre-requisites .....	322
Administrator permissions .....	322
Configuration parameters .....	322
Schema attributes .....	323
Account attributes .....	323
Group attributes .....	324
Provisioning Policy attributes .....	325
Install and register IQService .....	325
Additional information .....	326
Unstructured Target Collector .....	326
Troubleshooting .....	327

## Overview

---

The SailPoint Windows Local Connector manages User Accounts and Groups on Windows Operating System based computers through IQService. IQService uses WinNT ADSI service provider to connect to local users/groups for all versions of windows.

## Supported features

---

SailPoint Windows Local Connector supports the following features:

- Account Management
  - Manages Windows Local Users as Accounts
  - Aggregation, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements
- Account - Group Management
  - Manages Windows Local Groups as Account-Groups
  - Aggregation, Refresh Groups
  - Create, Update, Delete

## Configuration parameters

- Permission Management
  - Application can be configured for the following unstructured target collector to read permissions from respective end system:
    - Windows File Share - Read Windows File Share permissions directly assigned to accounts and groups.
    - The connector supports automated revocation of the aggregated permissions.
- Supports executing native before/after scripts for provisioning requests

### References

- “Unstructured Target Collector” on page 326
- “IQService Before/After Scripts” on page 580”

## Supported Managed Systems

---

Following versions of Microsoft Windows are supported by the SailPoint Windows Local Connector:

- Microsoft Windows Server 2012 R2
- Microsoft Windows Server 2012
- Microsoft Windows Server 2008 R2
- Microsoft Windows Server 2008

## Pre-requisites

---

- IQService must be installed on a Windows system. For more information, see “Appendix E: IQService”.
- Remote registry service must be started on the target system.
- Allow Exception for **File and Printer Sharing** in windows firewall.
- To **Turn off User Account Control** for Microsoft Windows Vista or later, perform the following steps:
  - a. For Microsoft Windows Vista and Microsoft Windows Server 2008, open the **Control panel ==> User Accounts ==> Turn User Account Control on or off**
  - b. For Microsoft Windows 7 onwards, open the **Control panel ==> User Accounts ==> User Accounts ==> Change User Account Control settings**

## Administrator permissions

---

User should be a member of **Administrators** group of Windows host computer which is to be managed.

## Configuration parameters

---

The following table lists the configuration parameters of SailPoint Windows Local Connector:

Parameters	Description
IQService Host*	Host name or IP address where IQservice is installed.
IQservice port*	The TCP/IP port where the IQService is listening for requests (Default: 5050).

Parameters	Description
UserName*	User name of the account with administrator rights on the managed system (Syntax: <i>computerName\userName</i> or <i>userName</i> ). For domain users it will be: <i>domainName\userName</i>
Password*	Password of user account mentioned in UserName field.
Server*	Host name or IP address of windows computer which is to be managed.
disableQualifyingLocalObjects	Flag to indicate whether aggregated objects must not be prefixed with server name. (Defaults to false. If set to true then aggregated object will not be prefixed with server name).
pageSize	Number of objects to fetch in a single request. Defaults to 1000

## Additional configuration parameters

---

The following table lists the additional configuration parameters of SailPoint Windows Local Connector:

Parameters	Description
disableNonLocalLookup	Set this parameter to false to read non-local (domain users/groups) group membership of local groups. Run IQService from any domain member server to read non-local group membership. The connector does not support provisioning of non-local group membership. Default: true

## Schema attributes

---

This section describes the different schema attributes.

### Account attributes

---

The following table lists the account attributes:

Attributes	Description
AutoUnlockInterval	Time interval for auto unlocking of locked user account.
Disabled	Flag to indicate if the user is disabled.
Description	User's description.
DirectoryPath	Fully qualified directory path <code>WinNt://...</code>
FullName	User's fullname.
groups	List of groups assigned to a user.
HomeDirectory	Location of the user's home directory.
Lockedout	Flag to indicate a user is locked out.
MaxStorage	The maximum amount of disk space the user can use.
MinPasswordLength	Minimum length of the user's password.

## Schema attributes

Attributes	Description
Name	Name of the account unqualified SAMAccountName.
objectSid	Windows SID.
PasswordAge	Time duration of the password in use. This property indicates the number of seconds that have elapsed since the password was last changed.
PasswordExpired	Indicates if the password is expired.
PasswordNotRequired	Flag to indicate if the user requires a password.
PasswordUnchangeable	Flag to indicate if the user password can be changed.
Profile	User's profile.
PrimaryGroupID	ID of the user's primary group.
sAMAccountName	Fully qualified version of the sAMAccountName.
UserFlags	User Flag defined in ADS_USER_FLAG_ENUM.
BadPasswordAttempts	Number of consecutive Bad Password Attempts made last time.
LoginScript	File path of Login script file.
HomeDirDrive	Home Directory Drive of the user.
PasswordNeverExpires	Flag to indicate if the password never expires.
MaxPasswordAge	Indicates the maximum time interval, in seconds, after which the password must be changed.
MinPasswordAge	Indicates the minimum time interval, in seconds, before the password can be changed.
LastLogin	Date and time when user logged in last time.

## Group attributes

---

The following table lists the group attributes:

Attributes	Description
Description	User's description.
DirectoryPath	Fully qualified directory path WinNt://... .
GroupMembers	List of groups assigned to a group.
GroupType	Windows SID.
Members	List of users assigned to a group.
objectSid	Windows SID.
sAMAccountName	Fully qualified version of the sAMAccountName.

## Provisioning Policy attributes

---

This section lists the provisioning policy attributes of SailPoint Windows Local Connector for create Account, create Group, and update Group.

Attributes	Description
<b>For create Account</b>	
sAMAccountName*	Name for user account to create. (Syntax: if <b>disableQualifyingLocalObjects</b> attribute in application configuration is unchecked then the format is sAMAccountName = <i>hostName\userName</i> . Otherwise sAMAccountName = <i>userName</i> .)
Password*	Password for new user account.
Description	Description of new user account.
Full Name	Full name of the user account.
Disable user account	Flag to create disabled user account.
User must change password on next logon	Flag to indicate whether user must change his password on next logon.
User cannot change Password	Flag to indicate whether user is allowed to change his password. If the value is <b>false</b> , user can change his password. Otherwise only system administrator can change his password.
Password never expires	Flag to indicate that user account password never expires until next password set.
<b>For create Group</b>	
sAMAccountName*	Name for group to create. (Syntax: if <b>disableQualifyingLocalObjects</b> attribute in application configuration is unchecked then sAMAccountName = <i>hostName\groupName</i> . Otherwise sAMAccountName= <i>groupName</i> .)
<b>For update Group</b>	
Description	Description of the group.
GroupType	Type of the group.
objectSid	Windows SID of group.
DirectoryPath	Fully qualified directory path <code>WinNT://...</code>

Note: Attributes marked with \* sign are the mandatory attributes.

## Install and register IQService

---

To install and register IQService, perform the following:

1. Create a directory in which you want to download the service. For example, `c:\iqservice`.
2. Extract the `IQService.zip` archive from the `IIQHOME\WEB-INF\bin\win` directory of the IdentityIQ installation into the created directory.

## Additional information

3. Run the following command to install a Windows service named IQService.  
`IQService.exe -i`
4. Start the service either from the Services Applet or from the command line by running the following command:  
`IQService.exe -s`

Other command line options with this service are:

- **-d**: run in the foreground in debug mode instead of in the background using the service control manager
- **-k**: stop the service
- **-r**: remove the service
- **-v**: display version information
- **-u**: Uninstall the service. Removes the service components and clears the registry entries.

Trace Parameters (require a restart of the IQService):

- **-l [level]**: Trace Level 0-3
  - 0: Off
  - 1: Information
  - 2: Error
  - 3: Debug
- **-f [fileName]**: Trace File Name (For example, "C:\IQService\IQServiceLog.log")

## Additional information

---

This section describes the additional information related to the Windows Local Connector.

### Unstructured Target Collector

---

Windows Local unstructured target collector supports aggregating direct access permissions on resources such as shared files and folders from target system and correlate it with aggregated user accounts and groups using objectSid as the correlation key.

#### Pre-requisites for target aggregation

IQservice needs to be installed on Target Windows computer.

#### Target aggregation configuration parameters

The following table lists the different target aggregation configuration parameters:

Attributes	Description
<b>IQService configuration parameters</b>	
IQservice Host*	The host on which the IQService resides.
IQservice Port*	The TCP/IP port where the IQService is listening for requests.

Attributes	Description
Number of targets per block	Number or targets (files) to include in each block of data returned.
<b>File share configuration parameters</b>	
Path*	Path of file or directory. You can target a specific file or a directory and its sub-directories containing multiple files from which to extract the required data. If you target a directory, use the Wildcard and Directory Depth fields to narrow the query if possible.
Directories Only	Use to instruct to the collector to ignore files and just report back directory permission information. Valid only if <b>Path</b> value is directory path.
Directory Depth	The sub-directory depth from which to extract data. The <b>Directory Depth</b> field enables you to extend your query up to ten (10) sub-directories below the one specified in the <b>Path</b> field.
Wildcard	Use wild cards to target a particular file type or naming scheme. For example, to search only exe, use *.exe or to search only files with names beginning with New_ and New_*.*
Administrator*	The administrator that has access to this share so you can collect permissions. This value can be domain\userName, computerName\userName, or userName.
Password*	The password associated with the specified administrator.
<b>Rule configuration parameters (used to transform and correlate the targets)</b>	
Creation Rule	The rule used to determine how the unstructured data extracted from data source is transformed into data that can be read by IdentityIQ.
Correlation Rule*	The rule used to determine how to correlate account information from the application with identity cubes in IdentityIQ.
<b>Provisioning related parameters</b>	
Override Default Provisioning	Overrides the default provisioning action for the collector.
Provisioning Action	The overriding provisioning action for the collector.

**Note:** Attributes marked with \* sign are the mandatory attributes.

## Troubleshooting

---

### 1 - Error returned from IQService: Unspecified Error

The following error message is displayed for any Windows Local application request:

```
Error returned from IQService:Unspecified Error
```

**Resolution:** Perform the following:

1. Ensure that the Target system is up and accessible from IQservice host.
2. Ensure that the Username and Password provided in application configuration are correct.

## Troubleshooting

3. If the target system is in a workgroup **Guest Only** option for **Sharing and security model for local accounts** in local policy will force all incoming network file sharing connections to authenticate as **Guest**.  
To resolve this problem perform the following steps:
  - a. On the Windows Start menu, click **Start ==> Control Panel ==> Administrative Tools ==> Local Security Settings**.
  - b. In the left pane, expand **Local Policies ==> Security** options.
  - c. In the right pane, double-click **Network access: Sharing and security model for local accounts**.
  - d. Select **Classic - local users authenticate as themselves** and click **OK**.
4. If the target system is Windows Server 2003 Service Pack 2 then some Windows updates are missing from the system. Turn on the Windows updates and install the latest updates.
5. Ensure that exception for **File and Printer Sharing** in windows firewall is enabled.
6. If the problem still persists try restarting IQservice.

### 2 - Error returned from IQService: The network path was not found

When Remote registry service is not started on Windows computer the following error message is displayed:

Error returned from IQService: The network path was not found

**Resolution:** Ensure that Windows Service named, **Remote Registry Service** is started on the Windows managed system.

### 3 - Unspecified Error

The following error message is displayed for any Windows Local Connector operation after upgrading to latest version from version 6.0 Patch 5 or below.

Unspecified Error

**Resolution:** Perform following:

1. Navigate to IdentityIQ debug page.
2. Select **Application** from the object browser.
3. Select and open your application from the list.
4. If a line exists with the following text as the starting text, then delete the line and save the application  
    "`<entry key="domain"`"

### 4 - Target aggregation failed as one of the path was not accessible

The target Aggregation failed as one of the path was not accessible.

**Resolution:** The **continueOnError** attribute must be set to true in the `targetSource` xml file to continue the target aggregation for other paths configured in unstructured target configuration.

# Chapter 36: SailPoint Workday Connector

---

The following topics are discussed in this chapter:

Overview .....	329
Supported features .....	329
Prerequisites .....	330
Administrator permissions .....	330
Configuration parameters .....	332
Configuring an Integration System in Workday .....	333
Workday Worker Custom fields .....	334
Schema attributes .....	336
Account attributes .....	336
Troubleshooting .....	343

## Overview

---

SailPoint Workday Connector is used to aggregate Worker records from Workday, and update email and phone.

### Supported features

---

SailPoint Workday Connector supports the following features:

- Account Management
  - Aggregation
  - Delta Aggregation
  - Future data: supported for Hire, Terminate and On Boarding events
  - Update of following attributes:
    - EMAIL\_ADDRESS\_HOME
    - EMAIL\_ADDRESS\_WORK
    - ADDITIONAL\_EMAIL\_ADDRESS\_HOME
    - ADDITIONAL\_EMAIL\_ADDRESS\_WORK
    - PHONE\_HOME
    - PHONE\_WORK
    - ADDITIONAL\_PHONE\_HOME
    - ADDITIONAL\_PHONE\_WORK

**Note:** When upgrading IdentityIQ to version 7.1, \*PROVISIONING\* must be added as a feature string in application debug page.

## Prerequisites

---

- A Workday administrator account and password for connecting to the given URL of the Workday tenant

**Note:** For more information on Workday administrator permissions, see “Administrator permissions”.

- User must create a Workday Integration System

For more information on creating Workday Integration System, “Configuring an Integration System in Workday” on page 333.

## Administrator permissions

---

Administrator login must have the following privileges:

- Create a user for accessing the Workday Integration System
- Create Integration System Security Group (Unconstrained)
- Perform the following for Integration System Security Group:
  - Add the user in the Integration System Security Group (Unconstrained)
  - Modify the Integration System Security Group to associate Maintain Contact information Domain  
**Note:** Verify if user has Get and Put permission for Maintain Contact information Domain.
  - Modify the Integration System Security Group to associate the following domain required by Workday Integration System:

Get Permission	Get and Put Permission
<ul style="list-style-type: none"><li>- Public Worker Reports</li><li>- General Staffing Information</li><li>- Current Staffing Information</li><li>- Time in Position</li><li>- Organizations</li><li>- Worker Data: Active and Terminated Workers - security policy to user group</li><li>- Manage: Organization Integration</li></ul>	<ul style="list-style-type: none"><li>- Worker Data: Home Contact Information</li><li>- Worker Data: Work Contact Information</li><li>- Worker Data: Work Email</li><li>- Worker Data: Home Email</li></ul>

**Note:** The Workday fields on Workday Managed System marked as private are not accessible through Workday API and the fields marked as Public are accessible.

- (For Delta Aggregation only) Integration System Security Group which has the respective administrator user associated with, must have **View Completed Only** permission for the related Business Process Security Policy.

**Note:** If schema attributes are not related to any of the following events, then permission for that Business Process Type is not required.

Following are the list of events and related business process supported in delta aggregation:

Events	Business Process Type
Hire Employee	Hire
Onboarding	Onboarding
Terminate Employee	Termination
Change Personal Information	Personal Information Change
Contact Information Event	Contact Change
Change Job	Change Job
Change Legal Name	Legal Change Name
Change Business Title	Title Change
Add Retiree Status	Add Retiree Status
Assign Organization Roles	Assign Roles
Change Owner	Assign Self-Assign Roles
Assign Superior	Assign Superior
EMERGENCY_CONTACT_EVENT	Change Emergency Contacts
Change Organization Assignments for Worker	Change Organization Assignments for Worker
Change Primary Address	Change Primary Address
Contract Contingent Worker	Contract Contingent Worker
Create Change Order from Contingent Worker Contract	Create Change Order from Contingent Worker Contract
Create Primary Address	Create Primary Address
Edit Worker Additional Data	Edit Worker Additional Data
Maintain Employee Contracts	Employee Contract
End Additional Job	End Additional Job
End Contingent Worker Contract	End Contingent Worker Contract
Change Marital Status	Marital Status Change
Move to New Manager	Move to New Manager
Assign Worker	Move Worker (By Organization)
Move Workers Staffing	Move Worker (Supervisory)
Assign Workers	Move Workers (By Organization)
New Hire Provisioning	New Hire Provisioning
Change Preferred Name	Preferred Name Change
Request Worker	Request Worker
Submit Resignation	Submit Resignation

## Configuration parameters

Events	Business Process Type
Transfer Contingent Worker	Transfer Contingent Worker
Transfer Employee	Transfer Employee

### (Optional) Additional administrator permissions

- To access custom or calculated fields the user must have access to all the fields referenced in the calculation.  
For more information on accessing the custom or calculated fields, see "Workday Worker Custom fields" on page 334.
- To access Additional Standard attribute user must have access to corresponding standard attributes.  
For more information on accessing the additional standard attribute, see "XPath to support additional standard attribute" on page 342.

## Configuration parameters

---

The following table lists the configuration parameters of Workday Connector:

If the configuration attribute Don't allow terminated Accounts is checked then the account aggregation in workday will aggregate the user using the cost center organization where it will initially fetch all the cost center organization and then fetch workday account from each organization in chunks.

This is applicable only for full aggregation. This does not apply to delta aggregation.

Following permission are needed for workday administrator:--

Get permission for Get Organizations Domain's Manage: Organization Integration Domain security policy.

**Note: Attributes marked with \* sign are the mandatory attributes.**

Parameters	Description
Workday URL*	This is valid URL to connect to the Human Resource module of workday.
Username*	The name of administrative user. <b>Note: Username must always be in the following format:</b> username@tenantname
Password*	Password of administrative user.
Effective Date Offset	Number of days to be offset for effective date. <b>Note: SailPoint recommends that the end user must perform full aggregation if the Effective Date Offset is changed.</b>
Chunk Size	The number of account to be fetched per page (Limit 1 to 999).
Integration System ID	Provide System ID of Integration System to fetch custom or calculated attributes.

Parameters	Description
Server Time Zone	Set this parameter, if termination related data must be fetched according to a particular time zone. By default the value is <b>UTC</b> . SailPoint recommends the use of Server Time Zone parameter during the Delta Aggregation operation. For example, if Workday server is in PST time zone then enter PST in this field.
Connection TimeOut	Provide the timeout value in minutes. Default value is 1 minute.
Don't Allow Terminated Accounts	Terminated or disabled accounts will not be aggregated if checked.

**Note:** *(Applicable only for full aggregation)* If the Don't Allow Terminated Accounts parameter is checked, the account aggregation in Workday application would aggregate the user using the cost center organization where it would initially fetch all the cost center organization and then fetch Workday account from each organization in chunks.

## Configuring an Integration System in Workday

---

Perform the following steps to configure **Workday Integration System**:

1. Search and click on **Create Integration Field Override Service** in Workday search available in the upper left of the interface to create new Integration Field Override Service.
2. Assign a name to the Integration Field Override Service and select the business object as Worker.
3. Add fields as per your requirement and provide a name for each field:
  - a. Click the **Plus** icon to add a new row.
  - b. Click **OK**.
  - c. Choose a name that will match the schema attribute in IdentityIQ.
  - d. Define the required settings for the new fields.
- Note:** For details related to LATEST\_WORKER\_RECORD and WORKER\_STATUS, see “Configuration for default custom fields” on page 334.
  - e. Click **Done**.
  - f. Repeat Step e through Step f for each new field.
4. Create a new Workday Integration System as follows:
  - a. On your Workday Home page, navigate to **Integration System ==> Create Integration System**.
  - b. Provide a name for the Integration System
  - c. Select **New Using Template** and provide the value custom integration template.
  - d. Click **OK**.
5. In your new Workday Integration System, add the field override service created in steps 1-2, as follows:
  - a. Navigate to **Custom Integration Services**.
  - b. Click on the **Plus** icon. Under **Custom Integration Services** add the field override service created in step 1-2.
  - c. Click **OK**.
6. Note the System ID for this Workday Integration System. It is required in for retrieving the custom or calculated attributes.
7. Search for **View Integration System**.
8. Enter the name of Workday Integration System you created in Step 4.
9. Click **OK**.
10. Map the new fields to the correct value:

## Configuration parameters

- a. Select **Integration System ==> Configure Integration Field Overrides**.
  - b. Select the calculated or custom field and map the correct value to it in **Override External Field**.
  - c. Click **OK**.
  - d. Click **Done**.
11. When applicable, grant the required permissions to the Service Account associated with the IdentityIQ application to aggregate the new custom or calculated fields.
  12. In IdentityIQ, add the following to the Workday application configuration:
    - Add the Workday Integration System ID from Step 6 to the Integration System ID field.
    - For each field in Step 3, add related attributes to the account schema.
- Note:** **Custom and calculated attributes added to the Workday schema must have " \_\_c" appended to the field names created in Workday. For example, if you added PREVIOUS\_EMPLOYER to Workday, you would add "PREVIOUS\_EMPLOYER\_\_c" in the application schema.**

## Workday Worker Custom fields

---

The Workday connector supports aggregating custom fields defined as part of Worker attributes in a Workday tenant. To aggregate the values of custom and calculated fields from Workday into IdentityIQ, perform the following:

- Configure an Integration System in Workday  
For more information on configuring an Integration System in Workday, see “Configuring an Integration System in Workday” on page 333.
- Fetch calculated field on Workday to get LATEST\_WORKER\_RECORD\_\_c
- Fetch calculated field on Workday to get WORKER\_STATUS\_\_c

## Configuration for default custom fields

The following default schema attributes require additional configuration before their values can be aggregated from a Workday application:

- LATEST\_WORKER\_RECORD\_\_c
- WORKER\_STATUS\_\_c

### *Fetching calculated field on Workday to get LATEST\_WORKER\_RECORD\_\_c*

In Workday, when any contractor is changed to employee or vice versa, Workday returns two records (current and previous) for the same worker.

To ensure that IdentityIQ aggregates only the latest record, a new calculated field called LATEST\_WORKER\_RECORD must be added in the Workday Managed System.

### Pre-requisites

Complete Step 1 - Step 2 and Step 4 through Step 6 of “Configuring an Integration System in Workday” on page 333.

### Perform the following steps:

1. After creating the Integration Field Override Service, add a calculated field and provide a name for it as follows:
  - a. Click the **Plus** icon to add a new row.
  - b. Type a name that will match the schema attribute in IdentityIQ
  - c. Click **OK**.

- d. Define the required settings for the new fields as follows:
- Field Name: **Any valid name**
  - Business Object: **Worker**
  - Function: **Lookup Value As Of Date**
  - Source Field: **Current Worker Record**
  - Select Effective Date: **Today**
2. Add another calculated field and provide a name for it:
- a. Click the **Plus** icon to add a new row.
  - b. Type a name that will match the schema attribute in IdentityIQ
  - c. Click **OK**.
  - d. Define the required settings for the new fields as follows including a return the value as Calculated Field created in Step 1:
  - e. Click **Done**.
- Field Name: **Any valid name**
  - Business Object: **Worker**
  - Function: **Lookup Related Value**
  - Select Lookup Field: **Worker**
  - Return Value: **The value of Field Name**
3. In the Search field in the top left corner, search for **View Integration System**.
4. Click **View Integration System**.
5. In the Integration System field, enter the name of integration system you created in Step 4 of "Configuring an Integration System in Workday" on page 333.
6. Select the item in the drop down and click **OK**.
7. Map the new fields to the correct value:
- a. To the right of the ID, click the Actions icon and select **Integration System ==> Configure Integration Field Overrides**.
  - b. In the Configuration page, find the row for LATEST\_WORKER\_RECORD
  - c. Click in **Override External Field** column and map it to the calculated field you created in Step 2
  - d. Click **OK**.
  - e. Click **Done**.
8. After you finish configuring the Workday system, sign in to IdentityIQ as an administrator and select one of the following options:
- **New Workday Application:** Add the Workday Integration System ID from Step 6 to the Integration System ID field.
  - **Existing Workday Application:** Add the following to the application configuration:
    - Add the Workday Integration System ID from Step 6 above to the Integration System ID field.
    - Add the LATEST\_WORKER\_RECORD\_\_c attribute to the account schema.

#### *Fetching calculated field on Workday to get WORKER\_STATUS\_c*

The WORKER\_STATUS attribute can be added as a custom calculated field on the Workday Managed System. This can return a status of Active, Terminated, Retired, and so on.

## Schema attributes

There is already a Worker Status field in Workday. For this to be aggregated in IdentityIQ, you must create a new field with a name IdentityIQ can recognize.

### Pre-requisites

Complete Step 1 - Step 2 and Step 4 through Step 6 of “Configuring an Integration System in Workday” on page 333.

### Perform the following steps:

1. After creating the Integration Field Override Service, add a calculated field and provide a name for it as follows:
  - a. Click the **Plus** icon to add a new row.
  - b. Type a name that will match the schema attribute in IdentityIQ
  - c. Click **OK**.
  - d. Click **Done**.
2. In the Search field in the top left corner, search for **View Integration System**.
3. Click **View Integration System**.
4. In the Integration System field, enter the name of integration system you created in Step 4 of “Configuring an Integration System in Workday” on page 333.
5. Select the item in the drop down and click **OK**.
6. Map the new fields to the correct value:
  - a. To the right of the ID, click the Actions icon and select **Integration System ==> Configure Integration Field Overrides**.
  - b. In the Configuration page, find the row for WORKER\_STATUS
  - c. Click in **Override External Field** column and select the Workday field **Worker Status**.
  - d. Click **OK**.
  - e. Click **Done**.
7. Add the **Active and Terminated Workers** security policy permission to the user group Worker Data.
8. After you finish configuring the Workday system, sign in to IdentityIQ as an administrator and select one of the following options:
  - **New Workday Application:** Add the Workday Integration System ID from Step 6 to the Integration System ID field.
  - **Existing Workday Application:** Add the following to the application configuration:
    - Add the Workday Integration System ID from Step 6 above to the Integration System ID field.
    - Add the WORKER\_STATUS\_\_c attribute to the account schema.

## Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports the following type of objects:

- Account: objects used when building identities Link objects.

## Account attributes

---

The following table lists the account attributes:

Attributes	Description	Source Data Element WD Object /Field Web Service Element File field
USERID (Identity Attribute)	Worker ID of the worker.	Worker_Data/User_ID
WORKER_DESCRIPTOR (Display attribute)	Descriptor of the worker. Full name.	Worker_Reference/@Descriptor
FILENUMBER	Employee ID of the worker.	Worker_Data/Worker_ID
MANAGER	Current manager of the worker.	Worker_Data/Management_Chain_Data/Worker_Supervisory_Management_Chain_Data/Management_Chain_Data[last()]/Manager_Reference/@Descriptor
JOBTITLE	Business title of the worker.	Worker_Data/Employment_Data/Position_Data/Business_Title
JOBCODE	Job profile of the worker.	Worker_Data/Employment_Data/Worker_Job_Data/Position_Data/Job_Profile_Summary_Data/Job_Profile_Reference/ID[@type='Job_Profile_ID']
EMPLOYEE_TYPE	Type of Employee	Worker_Data/Employment_Data/Position_Data/Worker_Type_Reference/ID[@type='Employee_Type_ID']
FIRST_NAME	Legal first name of the worker.	Worker_Data/Personal_Data/Name_Data/Legal_Name_Data/Name_Detail_Data/First_Name
LAST_NAME	Legal last name of the worker.	Worker_Data/Personal_Data/Name_Data/Legal_Name_Data/Name_Detail_Data/Last_Name
CLASS	Combination of Position, Time Type and Employment Type.	concat(Worker_Data/Employment_Data/Worker_Job_Data/Position_Data/Position_Title,Worker_Data/Employment_Data/Worker_Job_Data/Position_Data/Position_Time_Type_Reference/ID[@type='Position_Time_Type_ID'],Worker_Data/Employment_Data/Worker_Job_Data/Position_Data/Worker_Type_Reference/ID[@type='Employee_Type_ID'])
DEPARTMENT	Cost center.	Concatenation of Worker/Cost Center/Name and Worker/Cost Center/Code

## Schema attributes

Attributes	Description	Source Data Element WD Object /Field Web Service Element File field
LOCATION	Location of the worker.	Worker_Data/Employment_Data/Position_Data/Business_Site_Summary_Data/Name
TEAM	Team in the organization data of worker.	Worker_Data/Organization_Data/Worker_Organization_Data/Organization_Data[Organization_Type_Reference/ID]
DIVISION	Sales channel in the organization data of worker.	Worker/Sales Channel/Name
COST_CENTER_HIERARCHY	Cost center hierarchy of the worker.	Worker/Cost Center Hierarchy/Name
MANAGER_ID	Employee ID of the current manager of the worker.	Worker/Manager/Employee ID
HIREDATE	Hire date of the worker.	Worker_Data/Employment_Data/Worker_Status_Data/Hire_Date
TERMINATION_DATE	Termination date of the terminated employee.	Worker_Data/Employment_Data/Worker_Status_Data/Termination_Date
FULLPARTTIME	Type of employment full time or part time.	concat(Worker_Data/Employment_Data/Worker_Job_Data/Position_Data/Position_Title,Worker_Data/Employment_Data/Worker_Job_Data/Position_Data/Position_Time_Type_Reference/ID[@type='Position_Time_Type_ID'])
WORKER_NAME	Name of the worker	Worker_Data/Personal_Data/Name_Data/Preferred_Name_Data/Name_Detail_Data/@Formatted_Name
MIDDLE_NAME	Preferred Middle Name	Worker_Data/Personal_Data/Name_Data/Legal_Name_Data/Name_Detail_Data/Middle_Name
ON_LEAVE	Status of worker whether is he on leave	Worker_Data/Employment_Data/Worker_Status_Data/Leave_Status_Date[1]/@On_Leave
POSTAL_CODE	Postal Code of the city of a worker	Worker_Data/Personal_Data/Contact_Data/Address_Data[Usage_Data/Type_Data/Type_Reference/ID[@type='Communication_Usage_Type_ID']='WORK']/Postal_Code

Attributes	Description	Source Data Element WD Object /Field Web Service Element File field
COUNTRY	Country of a worker	Worker_Data/Personal_Data/Contact_Data/Address_Data[Usage_Data/Type_Data/Type_Reference/ID[@type='Communication_Usage_Type_ID']='WORK']/Country_Reference/ID[@type='ISO_3166-1_Alpha-3_Code']
CITY	City of a worker	Worker_Data/Personal_Data/Contact_Data/Address_Data[Usage_Data[@Public='true']/Type_Data[@Primary='true']/Type_Reference/ID[@type='Communication_Usage_Type_ID']='WORK']/Municipality
LEGAL_MIDDLE_NAME	Legal Middle Name of a worker	Worker/Legal Middle Name
FUTURE_DATE	Fetches the future date when the respective future action will be effective.	
FUTURE_ACTION	<p>Fetches the future action that will be performed on the respective worker.</p> <p>For example, Onboarding, Hire, Terminate</p>	
PHONE_WORK	Primary business phone number of worker	Worker_Data/Personal_Data/Contact_Data/Phone_Data[Usage_Data[@Public='true']/Type_Data[@Primary='true']/Type_Reference/ID[@type='Communication_Usage_Type_ID']='WORK']/@Formatted_Phone
PHONE_HOME	Primary home phone number of a worker	Worker_Data/Personal_Data/Contact_Data/Phone_Data[Usage_Data[@Public='true']/Type_Data[@Primary='true']/Type_Reference/ID[@type='Communication_Usage_Type_ID']='HOME']/@Formatted_Phone
COST_CENTER	Represents the organization name whose type is COST_CENTER	Worker_Data/Organization_Data/Worker_Organization_Data/Organization_Data[Organization_Type_Reference/ID[@type='Organization_Type_ID']='COST_CENTER']/Organization_Name
LATEST_WORKER_RECORD_C	Calculated field which will indicate whether its a latest record or not.	

## Schema attributes

Attributes	Description	Source Data Element WD Object /Field Web Service Element File field
WORKER_STATUS__c	Represent the current status of the worker, that is, active/terminated/retired	
<b>Optional attributes</b>		
<i>If user requires the image as type string add the following attributes</i>		
<b>Note:</b> After adding the image attribute in the schema, the time taken to fetch the response from workday would be more.		
IMAGE_NAME (type as String)	Image file name	Worker/Employee_Image/file name
IMAGE (type as String)	Image value as string.	Worker/Employee_Image/Image
<i>If required user must add the following attributes manually to Workday schema after upgrading to IdentityIQ version 7.1</i>		
EMAIL_ADDRESS_WORK	Email address of the worker.	Worker_Data/Personal_Data/Contact_Data/Email_Address_Data[Usage_Data[@Public='true']/Type_Data[@Primary='true']/Type_Reference/ID[@type='Communication_Usage_Type_ID']='WORK']/Email_Address
EMAIL_ADDRESS_HOME	Home email address of the worker.	Worker_Data/Personal_Data/Contact_Data/Email_Address_Data[Usage_Data[@Public='true']/Type_Data[@Primary='true']/Type_Reference/ID[@type='Communication_Usage_Type_ID']='HOME']/Email_Address
ADDRESS_HOME	Home address of the worker.	Worker_Data/Personal_Data/Contact_Data/Address_Data[Usage_Data/Type_Data/Type_Reference/ID[@type='Communication_Usage_Type_ID']='HOME']/@Formatted_Address
ADDRESS_WORK	Work address of the worker.	Worker_Data/Personal_Data/Contact_Data/Address_Data[Usage_Data/Type_Data/Type_Reference/ID[@type='Communication_Usage_Type_ID']='WORK']/@Formatted_Address

Attributes	Description	Source Data Element WD Object /Field Web Service Element File field
ADDITIONAL_PHONE_WORK	Additional business phone number of the worker.	Worker_Data/Personal_Data/Contact_Data/Phone_Data[Usage_Data[@Public='true']/Type_Data[@Primary='false']/Type_Reference/ID[@type='Communication_Usage_Type_ID']='WORK']/@Formatted_Phone
ADDITIONAL_PHONE_HOME	Additional home phone number of a worker.	Worker_Data/Personal_Data/Contact_Data/Phone_Data[Usage_Data[@Public='true']/Type_Data[@Primary='false']/Type_Reference/ID[@type='Communication_Usage_Type_ID']='HOME']/@Formatted_Phone
ADDITIONAL_EMAIL_ADDRESSES_WORK	Additional Email address of the worker.	Worker_Data/Personal_Data/Contact_Data/Email_Address_Data[Usage_Data[@Public='true']/Type_Data[@Primary='false']/Type_Reference/ID[@type='Communication_Usage_Type_ID']='WORK']/Email_Address
ADDITIONAL_EMAIL_ADDRESSES_HOME	Additional Home email address of the worker.	Worker_Data/Personal_Data/Contact_Data/Email_Address_Data[Usage_Data[@Public='true']/Type_Data[@Primary='false']/Type_Reference/ID[@type='Communication_Usage_Type_ID']='HOME']/Email_Address
ORGANIZATION_NAME	Represent organization name whose type is BUSINESS_UNIT	Worker_Data/Organization_Data/Worker_Organization_Data/Organization_Data[Organization_Type_Reference[@Descriptor ='Business Unit']]/Organization_Name
CONTRACT_END_DATE	Represent the contract end date of worker.	Worker_Data/Employment_Data/Worker_Contract_Data/Contract_End_Date
ADDRESS_LINE_1	Represent business site's Address_line_1 of worker.	Worker_Data/Employment_Data/Position_Data/Business_Site_Summary_Data/Address_Data/Address_Line_Data[@Type='ADDRESS_LINE_1']
STATE	Represent business site's country region.	Worker_Data/Employment_Data/Position_Data/Business_Site_Summary_Data/Address_Data/Country_Region_Reference/@Descriptor

## Schema attributes

Attributes	Description	Source Data Element WD Object /Field Web Service Element File field
LAST_DAY_OF_WORK	Represents the last day of work of the worker.	Worker_Data/Employment_Data/Worker_Status_Data/Termination_Last_Day_of_Work
COST_CENTER_REFERENCE_ID	Represent reference ID of organization type COST CENTER	Worker_Data/Organization_Data/Worker_Organization_Data/Organization_Data[Organization_Type_Reference[@Descriptor ='Cost Center']]//Organization_Reference_ID
PRIMARY_TERMINATION_REASON	Holds the reason for termination.	Worker_Data/Employment_Data/Worker_Status_Data/Primary_Termination_Reason_Reference/Descriptor
COMPANY_NAME	Represent organization name whose type is COMPANY	Worker_Data/Organization_Data/Worker_Organization_Data/Organization_Data[Organization_Type_Reference/ID[@type='Organization_Type_ID']='COMPANY']/Organization_Name

## XPATH to support additional standard attribute

With this release of IdentityIQ, Workday connector provides support to aggregate fields in addition to the fields present in the default account schema.

To fetch the additional standard Workday fields during account aggregation, perform the following:

1. Add the attributes to the account schema.
2. Provide the XPATH for those attributes.

For example, to fetch work email address, add an entry in the XpathAttributesMap as follows, where the value of the entry would be the XPATH required to fetch the required field:

```

<entry key="XpathAttributesMap">
    <value>
        <Map>
            <entry key="EMAIL_ADDRESS_WORK"
value="ns1:Worker_Data/ns1:Personal_Data/ns1>Contact_Data/ns1>Email_Address_Data
[ns1:Usage_Data[@ns1:Public='true']/ns1>Type_Data[@ns1:Primary='true']/ns1>Type_
Reference/ns1>ID[@ns1:type='Communication_Usage_Type_ID'
='WORK']/ns1>Email_Address"/>
        </Map>
    </value>
</entry>

```

3. Add above entry key in the application debug.

# Troubleshooting

---

## 1 - Test Connection is failing because of invalid workday host URL

**Resolutions:** Following are the possible exceptions and their solution:

- Test Connection Failed
- UnknownHostException
- FileNotFoundException
- HTTP response code: 500

Ensure that the Workday URL is correct and is case sensitive.

## 2 - Test connection fails with an error message

Test connection fails with the following error message due to Weblogic Server not using standard SUN HTTPS implementation provided by jdk:

```
openconnector.ConnectorException: javax.net.ssl.SSLKeyException: FATAL
Alert:BAD_CERTIFICATE - A corrupt or unuseable certificate was received
```

the Weblogic Server uses `weblogic.net.http.SOAPHttpsURLConnection` class instead of `javax.net.ssl.HttpsURLConnection` class.

**Resolution:** Start the Weblogic Server with the following argument so that it will use SUN HTTPS default handler:

```
-DUseSunHttpHandler=true jvm
```

Perform the following steps:

1. Open WebLogic Sever installed directory and navigate to `\user_projects\domains\base_domain\bin` directory.
2. Edit the `startManagedWebLogic.cmd` file and set `JAVA_OPTIONS=-DUseSunHttpHandler=true%JAVA_OPTIONS%`  
Save the `startManagedWebLogic.cmd` file.
3. Edit the `startManagedWebLogic.sh` file and set `JAVA_OPTIONS="-DUseSunHttpHandler=true ${JAVA_OPTIONS}"`  
Save the `startManagedWebLogic.sh` file.
4. Start the Managed Server.

## 3 - While provisioning of PHONE\_HOME, PHONE\_WORK, ADDITIONAL\_PHONE\_HOME, ADDITIONAL\_PHONE\_WORK attributes an error message appears

While provisioning of PHONE\_HOME, PHONE\_WORK, ADDITIONAL\_PHONE\_HOME and ADDITIONAL\_PHONE\_WORK attributes the following error message appears:

```
not Valid ID value for type="Phone_Device_Type_ID"
```

The above error message appears as the Workday Connector uses the default phone device reference id. Phone device reference id varies according to tenant. Workday application contains **ReferenceIDMap** that contains phone device type id map.

**Resolution:** To change the reference id of specific phone device type, modify **Phone\_Device\_Type\_ID** map as follows:

## Troubleshooting

```
<entry key="ReferenceIDMap">
  <value>
    <Map>
      <entry key="Phone_Device_Type_ID">
        <value>
          <Map>
            <entry key="Mobile" value="1063.1"/>
            <entry key="Telephone" value="1063.5"/>
            <entry key="Fax" value="1063.4"/>
            <entry key="Pager" value="1063.6"/>
          </Map>
        </value>
      </entry>
    </Map>
  </value>
</entry>
```

Perform the following to find **Phone\_Device\_Type\_Reference\_Id**:

1. Search **View Workday ID** report.
2. Enter **Phone Device Type** as class name in the report and click **OK**.
3. Click on **Related action** of phone device type ==> Integration IDs ==> View Reference IDs  
The Reference ID values of Phone Device type ID are listed.

### 4 - If EMPLOYEE\_TYPE value for any contingent worker is not correct

If EMPLOYEE\_TYPE value for any contingent worker is not correct, for example, 382.2 which is the ID of the EMPLOYEE\_TYPE.

**Resolution:** The application contains default map as follows for mapping the incorrect IDs with employee type:

```
<entry key="ReferenceIDMap">
  <value>
    <Map>
      <entry key="Employee_Type_ID">
        <value>
          <Map>
            <entry key="382.1" value="Consultant"/>
            <entry key="382.2" value="Contractor"/>
            <entry key="382.3" value="Vendor"/>
          </Map>
        </value>
      </entry>
    </Map>
  </value>
</entry>
```

```
</Map>  
</value>  
</entry>
```

The above default map, would map 382.2 key with **Contractor** string literal. For numeric value issues in EMPLOYEE\_TYPE account attribute in IdentityIQ, if the actual Contractor string literal is not obtained then copy the value of the appropriate reference id of that employee type from Workday managed system and add it in Workday application **Employee\_Type\_ID** map.

For example, in IdentityIQ for a value of 382.2 in EMPLOYEE\_TYPE perform the following steps:

1. In Workday managed system search for **Contingent Worker Type** and click on it.
2. For each row a related action option exists. Open **Related action ==> Integration ID ==> View IDs** and find to which type of contingent worker is 382.2 assigned.
3. After finding the correct value of the contingent worker type, add the value to the **Employee\_Type\_ID** map in IdentityIQ ==> Application debug page.

For the correct changes to be reflected perform full aggregation task.

## **Troubleshooting**

# Chapter 37: SailPoint XML Connector

---

The following topics are discussed in this chapter:

Overview .....	347
Supported features .....	347
Configuration parameters .....	347
Additional information .....	348
1 - Using XML Schema Definition (XSD) .....	348
2 - Using Document Type Definition (DTD) .....	349

## Overview

---

The SailPoint XML Connector is a *read only* connector used to extract data from XML files. The Document Type Definition (DTD) or XML Schema Definition (XSD) can be used for data validation and to discover the schema attributes.

### Supported features

---

SailPoint XML Connector supports the following features:

- Account Management
  - Aggregation, Discover Schema
- Account - Group Management
  - Aggregation

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The XML Connector uses the following connection attributes:

Attribute	Description
XML Data File Path	A semicolon separated list of XML files that contains account/group data.
XML Schema/DTD File Path	Enter the path and name of the XSD/DTD file that must be used for Discover Schema and Data Validation.  You can specify only single schema file.
XML Element for Account	Specify XML ELEMENT to map with the Account.
XML Element for Account Group	Specify XML ELEMENT to map with the Account Group.

## Additional information

Attribute	Description
File Transport	Specify how the file will be transferred. If the file resides locally on the application server select Local. If the file is on remote host, select FTP or SCP transport.
Host	In case of FTP/SCP transport, specify the hostname where the file is located.
User	In case of FTP/SCP transport, specify the username that will be used for authentication during the file transfer.
Password	In case of FTP/SCP transport, specify the password for the user that will be used for authentication during the file transfer.

## Additional information

This section describes the additional information related to the XML Connector.

Following are the examples for XML Element for Account and XML Element for Account Group attributes of Application Configuration:

- 1 - Using XML Schema Definition (XSD)
- 2 - Using Document Type Definition (DTD)

### 1 - Using XML Schema Definition (XSD)

#### Using XML Schema Definition (XSD)

- XML Element for Account - “orderperson”
- XML Element for Account Group - “item”

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    targetNamespace="urn:shiporder"
    xmlns:bks="urn:shiporder">

    <xsd:element name="shiporder">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="shipto" minOccurs="0" maxOccurs="unbounded">
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:element name="name" type="xsd:string"/>
                            <xsd:element name="address" type="xsd:string"/>
                        </xsd:sequence>
                    </xsd:complexType>
                </xsd:element>
                <xsd:element name="orderperson" minOccurs="0" maxOccurs="unbounded">
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:element name="name" type="xsd:string"/>
                            <xsd:element name="id" type="xsd:string" />
                            <xsd:element name="email" type="xsd:string" maxOccurs="5"/>
                        </xsd:sequence>
                        <xsd:attribute name="id" type="xsd:string"/>
                    </xsd:complexType>
                </xsd:element>
                <xsd:element name="item" minOccurs="0" maxOccurs="unbounded">
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:element name="title" type="xsd:string"/>
                            <xsd:element name="note" type="xsd:string"/>
                        </xsd:sequence>
                    </xsd:complexType>
                </xsd:element>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
</xsd:schema>
```

Figure 1—File-shiporder.xsd

```

<?xml version="1.0"?>
<x:shiporder xmlns:x="urn:shiporder">
  <orderperson id="ord1">
    <name>Smith</name>
    <id>001</id>
    <email>smith@test.com</email>
  </orderperson>
  <orderperson id="ord1">
    <name>Tom</name>
    <id>002</id>
    <email>tom@test.com</email>
  </orderperson>
  <item>
    <title>Box</title>
    <note>Type: Large</note>
  </item>
</x:shiporder>

```

Figure 2—File-shipping.xml

## 2 - Using Document Type Definition (DTD)

---

### Using Document Type Definition (DTD)

- XML Element for Account - “user”
- XML Element for Account Group - “group”

```

<!ELEMENT MyUnix (user*,group*)>
<!ELEMENT user (name,id,home,phone*)>
<!ELEMENT group (grpname,comment?)>
<!ELEMENT name      (#PCDATA)>
<!ELEMENT id       (#PCDATA)>
<!ELEMENT home     (#PCDATA)>
<!ELEMENT phone   (#PCDATA)>
<!ELEMENT grpname (#PCDATA)>
<!ELEMENT comment (#PCDATA)>

```

Figure 3—File-UNIX.dtd

## Additional information

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE MyUnix SYSTEM "file:///C:/XMLs/unix.dtd">
<MyUnix>
    <user>
        <name>Test User</name>
        <id>1004</id>
        <home></home>
        <phone>84245</phone>
        <phone>66666</phone>
    </user>
    <user>
        <name>System user</name>
        <id>1005</id>
        <home>/local/home/scorp</home>
    </user>
    <group>
        <grpname>Artist</grpname>
        <comment>Group of artists</comment>
    </group>
</MyUnix>
```

Figure 4—File-UNIX.xml

### Important notes

- Identifying XML Element for Account/Account Group mapping for XML Schema:
  - It must be an XML Element
  - The XML Element type must be **complexType**
  - Elements of this element should not have been defined as **complexType**
- Multi-valued attributes for DTD:
  - An asterisk or plus sign (\* or +) can be used to define a multi-valued attribute. For example,  

```
<attribute name>*
```

Or  

```
<attribute name>+
```

This is same as DTD syntax to allow 1 or more and 0 or more occurrences of an element.
- Multi-valued attributes for XML Schema:
  - Add Element's attribute **maxOccurs** with value greater than 1 or **unbounded** to make an attribute a multi-valued attribute. This is also an XML Schema Syntax to allow more than 1 occurrence.
- Turning off XML validation:
  - Set the application configuration attribute **xmlValidation** to **false**.

# Assisted Deployment Connectors

A minority of SailPoint customers have deployed the Connectors in this section. SailPoint will provide assistance during the deployment of these connectors. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided on Compass, SailPoint's Online customer portal. In some instances, SailPoint will guide the deployment team and actively participate in the design, configuration, and testing of the connectivity to the managed system.

For more specific information, refer to the Connector and Integration Deployment Center on Compass.

This section contains information on the following:

- "SailPoint Amazon Web Services Identity and Access Management Connector" on page 351
- "SailPoint Box Connector" on page 365
- "SailPoint CyberArk Connector" on page 371
- "SailPoint Duo Connector" on page 375
- "SailPoint Google Apps Connector" on page 387
- "SailPoint GoToMeeting Connector" on page 403
- "SailPoint IBM i Connector" on page 407
- "SailPoint Microsoft SharePoint Server Connector" on page 415
- "SailPoint Microsoft SharePoint Online Connector" on page 423
- "SailPoint Microsoft Project Server Connector" on page 429
- "SailPoint NetSuite Connector" on page 435
- "SailPoint Oracle HRMS Connector" on page 441
- "SailPoint Oracle E-Business Suite Connector" on page 447
- "SailPoint PeopleSoft HCM Database Connector" on page 457
- "SailPoint RSA Authentication Manager Connector" on page 465
- "SailPoint Remedyforce Connector" on page 473
- "SailPoint SAP Connector" on page 481
- "SailPoint System for Cross-Domain Identity Management Connector" on page 501
- "SailPoint WebEx Connector" on page 509
- "SailPoint Yammer Connector" on page 515



# Chapter 38: SailPoint Amazon Web Services Identity and Access Management Connector

---

The following topics are discussed in this chapter:

Overview .....	351
Supported features .....	352
Pre-requisites .....	352
Administrator permissions .....	353
Schema attributes .....	357
Account schema .....	357
Group schema .....	357
Provisioning Policy attributes .....	358
Account .....	358
Account-Group .....	358
Additional information .....	358
Amazon Web Services Identity and Access Management API's .....	358
Troubleshooting .....	360

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

Amazon Web Services (AWS) Identity and Access Management (IAM) helps you securely control access to Amazon Web Services and your account resources. With IAM, you can create multiple IAM users under your AWS account or enable temporary access through identity federation with your corporate directory. In some cases, you can also enable access to resources across AWS accounts. IAM offers greater security, flexibility, and control when using AWS.

Without IAM, however, you must either create multiple AWS accounts-each with its own billing and subscriptions to AWS products-or share the security credentials of a single AWS account. In addition, without IAM, you cannot control the tasks a particular user or system can do and what AWS resources they might use.

IAM enables identity federation between your corporate directory and AWS services. This enables you to use your existing corporate identities to grant secure and direct access to AWS resources, such as Amazon S3 buckets, without creating a new AWS identity for those users.

IAM is a web service that enables AWS customers to manage users and user permissions under their AWS account.

For more information about this product, see [AWS Identity and Access Management \(IAM\)](#).

The objective of this connector is to support reading and provisioning of AWS IAM accounts, account groups and account group assignment.

## Supported features

---

SailPoint Amazon Web Services Identity and Access Management Connector supports the following features:

- Account Management
  - Manages IAM Users under the AWS Account as Accounts
  - Aggregate, Refresh Accounts
  - Create, Update, Delete
  - Change Password
  - Add/Remove Entitlements
  - Enable: Activates only one existing Access Key and Signing Certificate
  - Disable: Deactivates and/or deletes ALL existing Security Credentials
- Account - Group Management
  - Manages IAM Groups under the AWS Account as Account-Groups
  - Aggregate, Refresh Group
  - Create, Update, Delete
- Permissions Management
  - Application reads permissions directly assigned to accounts and groups as direct permissions during account and group aggregation respectively.
  - The connector does not support automated revocation of the aggregated permissions and creates work item for such requests.

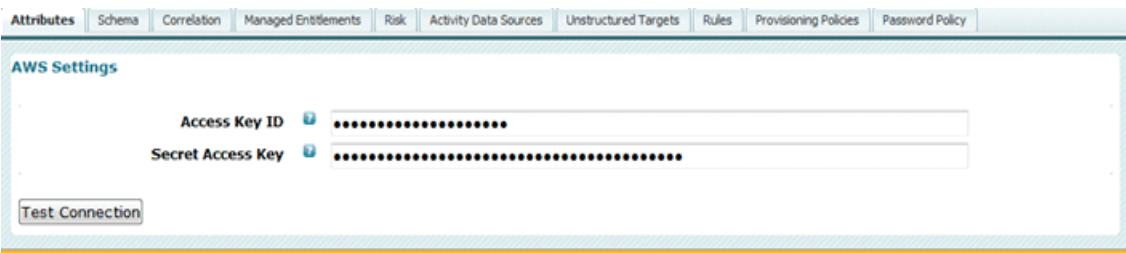
## Pre-requisites

---

**Note:** If AWS Identity and Access Management Connector is behind proxy server, see the “Special Java Considerations” section of the *SailPoint IdentityIQ Installation Guide*.

The connector requires the following Access Credentials to access the various IAM APIs:

- Access Key ID
- Secret Access Key



IAM is a feature of AWS account. If you are already signed up for a product that is integrated with IAM, you do not need to do anything else to sign up for IAM, and you will also not be charged extra for using it. You will be charged only for use of other AWS services by your users.

**Note:** IAM works only with AWS products that are integrated with IAM. For a list of such products, see [Integrating with Other AWS Products](#).

If you do not already have an AWS account, you need to create one to use IAM. You can create an AWS account when you sign up to use an AWS product for the first time. To sign up for AWS, perform the following:

1. Navigate to <http://aws.amazon.com>, and then click **Sign Up Now**.
2. Follow the on-screen instructions.  
Part of the sign-up procedure involves receiving a phone call and entering a PIN using the phone keypad.

An Access Key is automatically created upon creating an account. See the “Security Credentials” section of your account to obtain your Access Keys from the following link:

<http://aws-portal.amazon.com/gp/aws/developer/account/index.html?action=access-key>

To create SSL communication between IdentityIQ and AWS Server, perform the following:

1. Export server certificate and copy the exported .cer file to the Java client computer (IdentityIQ computer).
2. At the client computer execute the following command from the bin directory of JDK:  

```
keytool -importcerts -trustcacert -alias aliasName -file <absolute path of certificate> -keystore <JAVA_HOME>/jre/lib/security/cacerts
```

In the preceding command line, *aliasName* is the name of the alias.

## Administrator permissions

---

Custom policies must be created and these policies must be attached to the users.

### Creating Custom Policy

To create custom policy, perform the following steps:

1. Log in to AWS server with administrator privileges to create custom policy.
2. On the left hand side of the screen, click on **Policies**.
3. On the right hand side of the screen, click on the **Create Policy** button.
4. On **Step 1: Create Policy** page, select **Create Your Own Policy** section.
5. On **Step 2: Set Permissions** page, enter the following data respectively (based on the operation) and click on **Validate Policy**:

Policy Name	Description	Policy Document
<b>Test Connection</b>		
SPTestConnectionPolicy	SailPoint IdentityIQ policy for Test Connection	{         "Version": "2012-10-17",         "Statement": [             {                 "Effect": "Allow",                 "Action": [                     "iam>ListAccountAliases",                     "iam: ParseError"                 ],                 "Resource": "*"             }         ]     }

## Overview

Policy Name	Description	Policy Document
<b>Account Aggregation</b>		
SPAccountAggregationPolicy	SailPoint IdentityIQ policy for Account Aggregation	{         "Version": "2012-10-17",         "Statement": [             {                 "Effect": "Allow",                 "Action": [                     "iam>ListUsers",                     "iam GetUser",                     "iam GetLoginProfile",                     "iam>ListAccessKeys",                     "iam>ListSigningCertificates",                     "iam>ListMFADevices",                     "iam>ListPermissions",                     "iam>ListUserPolicies",                     "iam>ListGroupsForUser"                 ],                 "Resource": "*"             }         ]     }
<b>Group Aggregation</b>		
SPGroupAggregationPolicy	SailPoint IdentityIQ policy for Group Aggregation	{         "Version": "2012-10-17",         "Statement": [             {                 "Effect": "Allow",                 "Action": [                     "iam>ListGroups",                     "iam GetGroup",                     "iam>ListUsersInGroup",                     "iam>ListGroupPolicies"                 ],                 "Resource": "*"             }         ]     }
<b>Create user with assigned group</b>		
SPCreateUserPolicy	SailPoint IdentityIQ policy for creating user with assigned group	{         "Version": "2012-10-17",         "Statement": [             {                 "Effect": "Allow",                 "Action": [                     "iam>CreateUser",                     "iam>CreateLoginProfile",                     "iam>AddUserToGroup"                 ],                 "Resource": "*"             }         ]     }

Policy Name	Description	Policy Document
<b>Enable/Disable Account</b>		
<b>Note: Permission are already mentioned in Account Aggregation.</b>		
<b>Delete Account</b>		
SPDeleteAccountPolicy		<pre>{   "Version": "2012-10-17",   "Statement": [     {       "Effect": "Allow",       "Action": [         "iam:DeleteUser",         "iam:DeleteLoginProfile",         "iam:RemoveUserFromGroup"       ],       "Resource": "*"     }   ] }</pre>
<b>Update/Change Password User</b>		
SPChangePasswordPolicy		<pre>{   "Version": "2012-10-17",   "Statement": [     {       "Effect": "Allow",       "Action": [         "iam: UpdateUser",         "iam: UpdateLoginProfile"       ],       "Resource": "*"     }   ] }</pre>
<b>Create Group</b>		
SPCreateGroupPolicy		<pre>{   "Version": "2012-10-17",   "Statement": [     {       "Effect": "Allow",       "Action": [         "iam: CreateGroup"       ],       "Resource": "*"     }   ] }</pre>

## Overview

Policy Name	Description	Policy Document
<b>Update Group</b>		
SPUpdateGroupPolicy		{         "Version": "2012-10-17",         "Statement": [             {                 "Effect": "Allow",                 "Action": [                     "iam: UpdateGroup"                 ],                 "Resource": "*"             }         ]     }
<b>Delete Group</b>		
SPDeleteGroupPolicy		{         "Version": "2012-10-17",         "Statement": [             {                 "Effect": "Allow",                 "Action": [                     "iam: DeleteGroup"                 ],                 "Resource": "*"             }         ]     }

The **Review Policy** page is displayed.

- On **Step 3: Review Policy** page, review and validate the policy and click on **Create Policy**.

On successfully creating the policy the following message is displayed:

SPTestConnectionPolicy has been created.

Now you are ready to attach your policy to users, groups and roles.

## Attaching the Policy to users

To attach the policies to the users, perform the following:

1. Navigate to home page and click on **Users** on left hand side of the home page.
2. Click on the user for which the policy must be attached.
3. Under the **Permissions** tab click on **Attach Policy**.
4. On the Attach Policy page, search for **SP** policies (created in “Creating Custom Policy” section) under the **Policy Type** field.
5. Select the policy that must be attached to the user and click on **Attach Policy** button.

Perform the above steps to add more than one policy to the user.

# Schema attributes

---

The following schema attributes are defined:

- Account schema
- Group schema
- Schema extension and custom attributes

## Account schema

---

The following table lists the account schema:

Name	Type	Description
UserName	String	The name identifying the user.
UserId	String	The stable and unique string identifying the user.
Path	String	Path of the group.
Arn	String	The Amazon Resource name of the user.
CreateDate	String	User creation date.
Access Keys*	String	User Access keys.
Signing Certificates*	String	User signing certificates.
Has Password	String	Password status.
Multi-Factor Authentication Device*	String	Multi-Factor Authentication Device.
Groups	Group	Groups of the user.

**Note:** Attributes with the \* sign must be manually deleted only when upgrading IdentityIQ from version 7.0 or above to IdentityIQ version 7.1.

## Group schema

---

The following table lists the group schema:

Name	Type	Description
GroupName	String	Group name.
GroupId	String	Group ID.
Path	String	Group path.
Arn	String	Amazon Resource Name of the group.
CreateDate	String	Group creation date.
GroupPermissions	String	Group permissions.

## Provisioning Policy attributes

### Schema extension and custom attributes

The connector handles all the attributes currently retrieved or provisioned by the respective IAM APIs at the time of designing and developing the connector. In addition, AWS IAM has fixed schemas and does not support adding custom attributes to any of the schemas. Therefore, the connector does not provide support for extending the schema and defining custom attributes.

## Provisioning Policy attributes

---

The following default provisioning policies are defined for Account and Account-Group.

### Account

---

- **Create:** The following table lists the attributes that are required for creating an account.

Name	Type	Required	Description
User Name	String	Yes	Name of the user to create.
Password	Secret	Yes	The new password for the user name.

- **Update:** The following table lists the attributes that are required for updating an account.

Name	Type	Required	Description
New User Name	String	No	New name for the user.
New Path	String	No	New path for the user.

### Account-Group

---

- **Create:** The following table lists the attributes that are required for creating a group.

Name	Type	Required	Description
Group Name	String	Yes	Name of the group to create.

- **Update:** The following table lists the attributes that are required for updating a group.

Name	Type	Required	Description
Path	String	No	New path for the group.

## Additional information

---

This section describes the additional information related to the AWS Connector.

### Amazon Web Services Identity and Access Management API's

---

This section describes the API method used by the AWS IAM Connector.

## Interaction with the application

The connector makes use of the REST requests to call the functionality exposed by an Amazon Web Services (AWS) API. REST or Query requests are simple HTTP or HTTPS requests that use an HTTP verb (such as GET or POST) and the Action or Operation parameter that specifies the API you are calling.

Calling an API using a REST or Query request is the most direct way to access a web service, but requires that your application handles low-level details such as generating the hash to sign the request and error handling.

The benefit of using a REST or Query request is that you have access to the complete functionality of an API. The connector makes use of the REST requests and has the provision to handle the low-level details.

## APIs used

The following table lists the IdentityIQ operations along with the corresponding IAM APIs (Actions) used:

IdentityIQ Operation	IAM API (Action)
Test Connection	ListAccountAliases
Account-Group Aggregation <ul style="list-style-type: none"> <li>• Groups</li> <li>• Group Permissions</li> </ul>	ListGroups ListGroupPolicies
Account Create <ul style="list-style-type: none"> <li>• Set Password</li> </ul>	CreateUser CreateLoginProfile
Account Aggregation <ul style="list-style-type: none"> <li>• Summary/Attributes</li> <li>• Access Keys</li> <li>• Signing Certificates</li> <li>• Password</li> <li>• Multi-Factor Authentication (MFA) Device</li> <li>• Entitlements/Groups</li> <li>• Direct Permissions</li> </ul>	ListUsers ListAccessKeys ListSigningCertificates GetLoginProfile ListMFADevices ListGroupsForUser ListUserPolicies, ListGroupPolicies
Account Refresh <ul style="list-style-type: none"> <li>• Summary/Attributes</li> <li>• Access Keys</li> <li>• Signing Certificates</li> <li>• Password</li> <li>• MFA Device</li> <li>• Entitlements/Groups</li> <li>• Direct Permissions</li> </ul>	GetUser ListAccessKeys ListSigningCertificates GetLoginProfile ListMFADevices ListGroupsForUser ListUserPolicies, ListGroupPolicies
Account Update	UpdateUser

## Troubleshooting

Account Delete <ul style="list-style-type: none"> <li>• Read Entitlements/Groups</li> <li>• Remove Entitlements/Groups</li> <li>• Read Direct Permissions</li> <li>• Remove Direct Permissions</li> <li>• Read Security Credentials <ul style="list-style-type: none"> <li>- Access Keys</li> <li>- Signing Certificates</li> <li>- Password</li> <li>- MFA Device</li> </ul> </li> <li>• Remove Security Credentials <ul style="list-style-type: none"> <li>- Access Keys</li> <li>- Signing Certificates</li> <li>- Password</li> <li>- MFA Device</li> </ul> </li> </ul>	DeleteUser ListGroupsForUser RemoveUserFromGroup ListUserPolicies DeleteUserPolicy  ListAccessKeys ListSigningCertificates GetLoginProfile ListMFADevices  DeleteAccessKey DeleteSigningCertificate DeleteLoginProfile DeactivateMFADevice
Account-Group Create	CreateGroup
Account-Group Update	UpdateGroup
Account-Group Delete <ul style="list-style-type: none"> <li>• Read Accounts in the Group</li> <li>• Remove Accounts from the Group</li> <li>• Read Group Permissions</li> <li>• Remove Group Permissions</li> </ul>	DeleteGroup GetGroup RemoveUserFromGroup ListGroupPolicies DeleteGroupPolicy
Account Enable <ul style="list-style-type: none"> <li>• Activate Access Keys (One only)</li> <li>• Activate Signing Certificates (One only)</li> </ul>	UpdateAccessKey UpdateSigningCertificate
Account Disable <ul style="list-style-type: none"> <li>• Deactivate Access Keys (All)</li> <li>• Deactivate Signing Certificates (All)</li> <li>• Delete Password</li> <li>• Deactivate MFA Device</li> </ul>	UpdateAccessKey UpdateSigningCertificate DeleteLoginProfile DeactivateMFADevice
Reset Password	UpdateLoginProfile
Request Entitlement	AddUserToGroup
Remove Entitlement	RemoveUserFromGroup

## Troubleshooting

---

### 1 - Restore (Enable) security credentials

Restore security credentials for your IAM users.

**CreateLoginProfile:** Creates a password for the specified user, giving the user the ability to access AWS services through the AWS Management Console. IdentityIQ does not allow specifying the Password, which is a required parameter for this API, during **Account Enable** operation.

**Workaround:** The password must be set/created using Set/Reset Password operation to enable the account.

## 2 - Request timestamp is skewed

The test connection failed with the following error message:

```
openconnector.ConnectorException: Error Code 400 - RequestExpired - Request timestamp  
is too skewed. Timestamps must be within 900 seconds of server time. Timestamp date:  
2015-04-01T19:07:51.185Z
```

**Resolution:** Timestamp of server instance and IdentityIQ must be same.

## **Troubleshooting**

# Chapter 39: SailPoint Box Connector

---

The following topics are discussed in this chapter:

Overview .....	365
Supported features .....	365
Supported Managed Systems .....	366
Pre-requisites .....	366
Administrator permissions .....	366
Configuration parameter.....	366
Schema attributes .....	366
Account attributes .....	367
Group attributes.....	367
Provisioning Policy attributes .....	368
Troubleshooting .....	368

## Overview

---

**Note:** SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.

The Box Connector manages enterprise users and groups of Box server. Box Connector is a read/write connector that can retrieve enterprise users and groups of a particular network, activates/inactivates the users and assign enterprise user memberships to groups.

## Supported features

---

SailPoint Box Connector provides support for the following features:

- Account Management
  - Manage Box Users as Accounts
  - Aggregate, Refresh Account
  - Create, Update, Delete
  - Enable, Disable
  - Add/Remove Entitlements
- Account - Group Management
  - Manage Box Groups as Account Group
  - Aggregate, Refresh Group
  - Create, Update, Delete

## Supported Managed Systems

---

SailPoint Box Connector supports the following managed system:

- BOX API Version 2.0 supported by Box Server

## Pre-requisites

---

The Client Id, client secret, access and refresh token is required for the Box Connector. This has to be acquired by the Box Enterprise administrator from Box server using OAUTH2 mechanism to allow IdentityIQ to interact with Box server. The Box Connector uses the Client Id, client secret and the refresh token to automatically re-generate new authentication tokens and refresh tokens on expiry.

**Note:** If Box Connector is behind proxy server, see the “Special Java Considerations” section of the *SailPoint IdentityIQ Installation Guide*.

## Administrator permissions

---

The administrator should be able to create and retrieve enterprise users, enterprise user memberships, and Box groups.

The Box application created on box system must have **Manage an enterprise** scope selected as the permission.

**Note:** For a Box application existing prior to version 6.2 patch 2, the corresponding application must be deleted. A new application must be created by selecting the application type as “Box”. After the application is configured the aggregation must be performed.

## Configuration parameter

---

This section contains the information that this connector uses to connect and interact with the application.

The Box Connector uses the connection attributes listed in the following table:

Parameters	Description
Client Id	Box Server application Id.
Client secret	Box application secret key.
Access token and Refresh token	The refresh token which Box returns on completion of the OAUTH2 process. User can use clients like curl or POSTMAN to retrieve the first refresh token after granting access to the Box application.
Page Size	BoxNet page size.

## Schema attributes

---

The application schema is used to map Box server user or group objects to IdentityIQ account and group objects. The following Box managed user and group object attributes are mapped to Account and Group objects respectively.

## Account attributes

---

The following table lists the account attributes ([Table 1—Account attributes](#)):

**Table 1—Account attributes**

Attributes	Description
id	User ID as assigned by Box server.
name	Name of the enterprise user.
login	Email ID used to login.
role	User role as co-administrator/user.
memberof	Member of
space_amount	Space allocated to the user in GigaBytes.
max_upload_size	The maximum individual file size in bytes this user can have.
is_sync_enabled	Whether the user can use synchronization as Yes/No.
can_see_managed_users	Whether the user can see other users in the enterprise in their contact lists as Yes/No.
is_exempt_from_device_limits	Whether to exempt this user from Enterprise device limits as Yes/No.
is_exempt_from_login_verification	Whether or not this user must use two-factor authentication.
job_title	The user's job title displayed on their profile page.
phone	The user's phone number displayed on their profile page.
address	User address.
language	Two letter representation of user language. for example, en for English.
status	User is enabled or disabled.
enterprise	User enterprise.

## Group attributes

---

The following table lists the group attributes ([Table 2—Group attributes](#)):

**Table 2—Group attributes**

Attributes	Description
group_id	Group ID as assigned by Box server.
group_name	Group name.

**Box group membership and group access:** Whenever an entitlement is requested for a user, by default the Box connector adds a user into a group with **member** as the access value. This is true for all entitlement requests.

## Provisioning Policy attributes

**Note:** To disable the fetching of group members in account aggregation, set the value of the following attribute to true:

```
<entry key="disable_group_membership" value="true"/>
```

By default the value of this attribute is false.

**Note:** From IdentityIQ version 7.1 onwards, if user does not want to fetch Account - Group membership in account aggregation then delete 'memberof' (group membership) attribute from account schema.

## Provisioning Policy attributes

---

The following table lists the provisioning policy attributes ([Table 3—Provisioning Policy attributes](#)):

**Table 3—Provisioning Policy attributes**

Attributes	Description
name	Name for the enterprise user.
role	Box user role as co-administrator/user.
Login Id	login ID of the user.
space_amount	Space to be allocated to the user in GigaBytes.
Inactive account	Inactive the user account.
Unlimited Storage	Unlimited space amount.

## Troubleshooting

---

### 1 - Refresh token expire

In Box Connector for every transaction a refresh and an access token is required. Access token is valid for 60 minutes and refresh token is valid for 60 days.

In Box Connector after the access token has expired the connector implicitly creates new access and refresh tokens except for test connection transaction.

**Resolution:** If the refresh token has expired due to inactivity of connector for more than 60 days, the procedure for getting the new valid refresh token must be repeated as mentioned in the “Pre-requisites” on page 366 section.

### 2 - During test connection the following warning message appears

During test connection the following warning message appears:

Warning: New tokens have been generated. Exit application without save and retry again.

**Resolution:** This scenario appears when the access token has expired. Do not click the **Save** button for the application. Close the application and reopen it. Try doing the test connection now.

### 3 - During enable/disable of accounts the following error message appears

During enable/disable of accounts the following error message appears:

```
openconnector.ConnectorException: api.box.com
```

**Resolutions:** Verify the network connections.

### 4 - During provisioning operations the following error message appears

During provisioning operations the following error message appears:

```
Current tokens have expired. Failed to create new tokens.Error:Failed to retrieve box
tokens.Error:Box error code:400 Box error
message:{ "error": "invalid_grant", "error_description": "Refresh token has expired" }
```

**Resolution:** Create access and refresh token manually as mentioned in the “Pre-requisites” on page 366. and perform the test connection. Do not click the **Save** button for the application. Close the application and reopen it. Now perform the test connection.

## **Troubleshooting**

# Chapter 40: SailPoint CyberArk Connector

---

The following topics are discussed in this chapter:

Overview .....	371
Supported features .....	371
Pre-requisites .....	371
Configuration parameters .....	372
Schema attributes .....	373
Account attributes .....	373
Additional information .....	373
Direct permission .....	373

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

The SailPoint CyberArk Connector is a *read only* connector and a modified version of delimited file based connector which provides out-of box features needed for processing CyberArk Entitlement report. The CyberArk connector retrieves user and entitlement information from Entitlement Report generated by CyberArk system.

## Supported features

---

The CyberArk Connector supports the following features:

- Account Management
  - Manage CyberArk Users as Accounts
  - Aggregate (From CyberArk Entitlement report), Partitioning Aggregation
- Permissions Management
  - Application reads permissions directly assigned to accounts as direct permissions during account aggregation.
  - The connector does not support automated revocation of the aggregated permissions and creates work item for such requests.

### References

- “Appendix C: Partitioning Aggregation”

## Pre-requisites

---

User must have the Entitlement Report generated in csv format and sorted on “User” from the CyberArk system.

# Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application.

The CyberArk Connector uses the connection attributes listed in the following table:

Parameters	Description
<b>File:</b>	
File Path	Enter the path and name of the data file that should be parsed.
File Encoding	Specify the encoding that was used when saving the data file. If this parameter is blank the application's server default encoding will be used when parsing the file.
<b>Transport:</b>	
File Transport	Specify how the file will be transferred. If the file resides locally on the application server, select Local.
Host	Specify the host name where the file is located.
User	Specify the username that will be used during the file transfer.
Password	Specify the password for the user that will be used during the file transfer.
<b>Filtering:</b>	
Number of lines to skip	Enter the number of lines to skip from the top of the data file before parsing begins.
Filter Empty	Select this option if you want to filter out any objects that parse but have no attributes.
Comment Character	Enter a comment character used in the data file. Any line starting with this character will be skipped.
<b>Connector Rules:</b>	
Build Map Rule	A rule that is called for each row in the data file. This rule is used to convert the string tokens from the data file into a <b>java.util.Map</b> object. If a rule is not specified the connector builds a map with the contents keyed by the column name.
PreIterate Rule	A rule that is called before the iteration process begins and provides a bridge for operations such as checking the file, building an alternate feed, or returning a stream object.
PostIterate Rule	A rule that is called after the iteration process has completed.
Map To ResourceObject Rule	A rule that is called for each unique <b>java.util.Map</b> created from the data file. This rules job is used to convert a <b>java.util.Map</b> object, built from the data file, into a <b>ResourceObject</b> . If a rule is not specified the connector builds a <b>ResourceObject</b> using the schema.
MergeMaps Rule	A rule that is called during merging for each row that has a matching index column. The rule will receive the existing map along with the newly parsed map that has to be merged. If a rule is not specified the connector builds a combined <b>java.util.Map</b> using the original object and merges the attributes specified in the <b>mergeColumns</b> configuration option.

# Schema attributes

---

This section describes the different schema attributes.

## Account attributes

---

The following table lists the account attributes ([Table 1—Account attributes](#)):

**Table 1—Account attributes**

Attributes	Description
User	User name of the user.
Full name	Full name of the user.
Group membership	Indicated whether user has group membership.
Location	Location of the User.
Target Account	Target account for which user may have Retrieve, Use and change permissions.
Target System	Name of the system the Target Account belongs to.
Deleted	Indicates whether the user is deleted.
Retrieve	Indicates if the user has Retrieve permission for mentioned target account.
Use	Indicates if the user has Use permission for mentioned target account.
Change	Indicates if the user has change permission for mentioned target account.

# Additional information

---

This section describes the additional information related to the CyberArk Connector.

## Direct permission

---

Direct permission on users describe the permission (Use, Change and/or Retrieve) a User has on Target Account/Target System. The Use, Retrieve, and Change attributes are different type of permissions CyberArk Connector user can have on Target system which has target account.

**For example**, for a record given below the direct permissions for administrator user are :

- Use on test/Cark-test
- Retrieve on test/Cark-test
- Change on test/Cark-test

User	Target System	Target account	Retrieve	Use	Change
Administrator	Cark-test	Test	Yes	Yes	Yes

## **Additional information**

# Chapter 41: SailPoint Duo Connector

---

The following topics are discussed in this chapter:

Overview .....	375
Supported features .....	375
Pre-requisites .....	375
Administrator permissions .....	376
Configuration parameters .....	376
Schema Attributes .....	376
Account attributes .....	376
Group attributes .....	377
Provisioning Policy attributes .....	378
Account attributes .....	378
Behavioral changes .....	379

## Overview

---

**Note:** SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.

The SailPoint Duo Connector is a Cloud based Connector. The Duo authentication secures login's by two factor authentications and protects users, data and applications from credential theft and breaches with a focus on streamlined usability.

In Duo Connector the users who have an account on Duo managed system are used for Account provisioning. You can configure the Duo connector to use any of the attributes of user / group which are supported by Duo APIs.

## Supported features

---

SailPoint Duo Connector supports the following features:

- Account Management
  - Aggregation, Refresh Accounts
  - Create, Delete
  - Enable, Disable, Unlock
  - Add/Remove Entitlement
  - Add/Remove Phone attribute
- Account - Group Management
  - Aggregation

## Pre-requisites

---

JRE version 1.7 must be installed.

## Administrator permissions

---

1. Login as an administrator user and log into the **Duo Administrator Panel**.

**Note:** Ensure that the administrator creating the new Administrator API integration has the following permissions:

- Grant Administrators
- Grant Applications
- Grant Read Resource

2. Create a new Admin API integration.

This will generate an integration key and secret key for you to use as mentioned in the following “Configuration parameters” section.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Duo Connector uses the following connection attributes:

Attribute	Description
API hostname *	The API hostname is unique to account and shared with all integrations. Uses https, unsecured http is not supported.
Integration Key *	Identifies integration. Required to configure your system to work with Duo.
Secret Key *	The secret key is treated like a password. Identifies integration. Required to configure your system to work with Duo.

## Schema Attributes

---

This section describes the different schema attributes.

### Account attributes

---

The following table lists the account attributes:

Attributes	Description
username	Users name.

Attributes	Description
status	The users status: <ul style="list-style-type: none"> <li><b>Active:</b> User must complete secondary authentication</li> <li><b>Bypass:</b> User will bypass secondary authentication after completing primary authentication.</li> <li><b>Disabled:</b> User will not be able to login.</li> <li><b>Locked out:</b> User has been automatically locked out due to excessive authentication attempts.</li> </ul>
email	Users email address.
user_id	Users unique ID generated by Duo system.
realname	Users real name.
notes	Notes about the user. Seen in Duo administrative interface.
groups	List of groups to which user belongs. Contains description and name of the group.
phones	A list of phones the user can use. Contains phone_id, number, extension, name, postdelay, predelay, type, capabilities, platform, activated, sms_passcodes_sent.
last_login	The last time the user logged in, as a UNIX timestamp or <b>null</b> if the user has not logged in.
tokens	A list of tokens the user can use. Contains serial, token_id, type.
desktoktokens	A list of tokens the user can use.

## Group attributes

---

The following table lists the account attributes:

Attributes	Description
name	The groups name.
desc	The groups description
status	The groups authentication status. <ul style="list-style-type: none"> <li><b>Active:</b> User must complete secondary authentication</li> <li><b>Bypass:</b> User will bypass secondary authentication after completing primary authentication.</li> <li><b>Disabled:</b> User will not be able to login.</li> <li><b>Locked out:</b> User has been automatically locked out due to excessive authentication attempts.</li> </ul>
group_id	The groups ID.
voice_enabled	If true, users in the group would be able to authenticate with a voice callback. If false, users in the group would not be able to authenticate with a voice callback.  <b>Note:</b> This setting has no effect if voice callback is disabled globally.

## Provisioning Policy attributes

Attributes	Description
sms_enabled	If true, users in the group would be able to use SMS passcodes to authenticate. If false, users in the group would not be able to use SMS passcodes to authenticate.  <b>Note: This setting has no effect if SMS passcodes are disabled globally.</b>
mobile_otp_enabled	If true, users in the group would be able to use mobile otp password to authenticate. If false, users in the group would not be able to use mobile otp password to authenticate.  <b>Note: This setting has no effect if mobile otp passwords are disabled globally.</b>
push_enabled	If true, users in the group would be able to use Duo Push to authenticate. If false, users in the group would not be able to use Duo Push to authenticate.  <b>Note: This setting has no effect if Duo Push is disabled globally.</b>

## Provisioning Policy attributes

This section describes the various provisioning policy attributes for Account.

### Account attributes

The following table lists the provisioning policy attributes for Create, Enable and Unlock Account:

Attribute name	Description
<b>Create Account</b>	
User Name*	Name of user to be created.
Real Name	Real name of the user.
Email ID	Email address will be used for enrollment of the user.
Phone	Phone numbers for the user.  <b>Note: After upgrading IdentityIQ version 6.4 to version 7.1, add the Phone attribute manually.</b>
<b>Enable and Unlock Account</b>	
User Name*	Name of the user to be enabled/unlocked.
status	Users status: <ul style="list-style-type: none"><li>• <b>Active:</b> User must complete secondary authentication</li><li>• <b>Bypass:</b> User will bypass secondary authentication after completing primary authentication.</li></ul>

## Behavioral changes

---

- During Aggregation
  - If a users status is active and has one or more groups connected to the user and if one of the groups (for example, Firewall Admins) is disabled, then the users status would get disabled from Firewall Admins group. This would also disable the user from all the groups assigned to that user.  
In the above scenario, if Account is aggregated, the user status is still displayed as active.
  - On Duo system the Last login for some of the users is displayed as **Never authenticated**. In this case, during aggregation, the last Login is displayed as blank.
  - For the **Phone numbers of Account** attribute, the API returns the phone number as +19405429053 but on Duo Connector UI it is displayed as 9405429053.
- For provisioning
  - When performing provisioning on **AD\_SYNC user(s)** on Duo, it fails with the following error message:  
`openconnector.ConnectorException: disable failed. Duo error code (40010): User is synchronized with Active Directory. Some attributes may be read-only.`

## **Behavioral changes**

# Chapter 42: SailPoint Dropbox Connector

---

The following topics are discussed in this chapter:

Overview .....	381
Supported features .....	381
Supported Managed System .....	382
Pre-requisites .....	382
Administrator permissions .....	382
Group attributes .....	384
Configuration parameters .....	382
Schema attributes .....	382
Account attributes .....	383
Group attributes .....	383
Provisioning Policy attributes .....	383
Account attributes .....	383
Group attributes .....	384

## Overview

---

The SailPoint Dropbox Connector manages enterprise users and groups of Dropbox for Business. Dropbox Connector is a read/write connector that can retrieve enterprise users and groups of a particular business, update the permission of users and group, assign enterprise user memberships to groups.

## Supported features

---

SailPoint Dropbox Connector provides support for the following features:

- Account Management
  - Manage Dropbox Users as Accounts
  - Aggregate, Refresh Account
  - Create, Update, Delete
  - Add/Remove Entitlements
- Account - Group Management
  - Manage Dropbox Groups as Account - Group
  - Aggregate, Refresh Group
  - Create, Update, Delete

## Supported Managed System

---

SailPoint Dropbox Connector supports the following managed system:

- Dropbox Business API Version 1 supported by Dropbox Server

## Pre-requisites

---

User must generate the access token for business Dropbox application. This has to be acquired by the Dropbox administrator from Dropbox business server using OAUTH2 mechanism to allow IdentityIQ to interact with Dropbox server. The access tokens are valid until the user uninstalls the application, or explicitly revokes the grant via the Dropbox page.

**Note:** For more information, see the [OAuth Guide in Dropbox documentation](#).

## Administrator permissions

---

Application created in the Dropbox requires the following permission (This permission has both read and update rights):

- Team member management: Team information, and the ability to add, edit, and delete team members

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

**Note:** The attributes marked with \* sign are the required attributes.

**Table 1—Delimited File Connector - Account Tab Descriptions**

Parameters	Descriptions
Dropbox Url*	The location where Dropbox server is present. For example, <a href="https://api.dropbox.com">https://api.dropbox.com</a>
Dropbox Access Token*	Access token specific to business application.
Member Page size	Number of records per page. Default: 1000 (maximum)

The default value of Dropbox api version is 1. If version must be changed, add the following attribute in the application debug page:

```
<entry key="dpWsapiVersion" value="Version_name" />
```

```
For example, <entry key="dpWsapiVersion" value="V2" />
```

## Schema attributes

---

This section describes the different schema attributes.

## Account attributes

---

The following table lists the account attributes:

Attributes	Description
member_id	Member ID
status	Status of the member whether active or invited.
surname	Member's surname.
given_name	Member's first name.
email	Email address of the member.
permission	Permission of the Dropbox member on the Dropbox.
groups	List of groups connected to member.
external_id	External ID.

## Group attributes

---

The following table lists the group attributes:

Attributes	Description
group_id	Group ID
group_name	Name of the group.
num_members	Total count of the members connected to the group.
members	List of group members.
group owners	List of group owners.

## Provisioning Policy attributes

---

This section lists the different policy attributes of Dropbox Connector.

### Account attributes

---

The following table lists the provisioning policy attributes for Create Account:

Attributes	Description
Email	Member email.
First Name	Member given name.
Surname	Member surname.

## Provisioning Policy attributes

Attributes	Description
External Id	Member external Id.
Welcome Email	Welcome email sent to members.

## Group attributes

The following table lists the provisioning policy attributes for Create and Update Group:

Attributes	Description
Group Name	Name of the group.
<b>Update Group</b>	
Group Name	Name of the group.
Group Owners	Name of the group owners.
Group Id	Group Id.
Members Count	Number of members.
Group Members	Members of the group.

## Update Group Provisioning Policy

Updating Group provisioning policy requires the following Identity picker to add members and owners to the Group:

- Add Members to Group: helps end user to only add member to the group.
- Add Owners to Group: helps end user change the permission of the already added member to group owner. Add only those users who are already the members of the group.

**Note:** The Identity picker feature saves user from remembering and entering email ID of the target account. The email ID of account in Dropbox must be same as that of the identity in IdentityIQ.

## Delete Provisioning Policy

If required user can add the following delete provisioning policy for Dropbox application:

```
<Form name="account delete" objectType="account" type="Delete">
  <Attributes>
    <Map>
      <entry key="IIQTemplateOwnerDefinition">
        <value>
          <DynamicValue value=" "/>
        </value>
      </entry>
    </Map>
  </Attributes>
  <Description>account delete</Description>
  <Field displayName="email_id of member to transfer file" filterString=""
name="email_id of member to transfer error" reviewRequired="true"
type="string"/>
  <Field displayName="transfer_admin_email_id" filterString=""
```

## **Provisioning Policy attributes**

```
name="transfer_admin_email_id" reviewRequired="true" type="string"/> >  
</Form>
```

## **Provisioning Policy attributes**

# Chapter 43: SailPoint Google Apps Connector

---

The following topics are discussed in this chapter:

Overview .....	387
Supported features .....	388
Pre-requisites .....	388
Administrator permissions .....	389
Configuration parameters .....	389
Schema attributes .....	389
Account attributes .....	389
Group attributes .....	390
Provisioning Policy attributes .....	394
Troubleshooting .....	399

## Overview

---

**Note: SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

Google Apps is a service from Google which provides independent versions of several Google products under a custom domain name. It features several Web applications with similar functionality to traditional office suites, including Gmail, Google Groups, Google Calendar, Talk, Drive, Play, Docs, News, Wallet and Sites.

SailPoint Google Apps Connector manages accounts and groups of Google Apps for Business, Education, or ISP. The connector consists of a number of features like managing Gmail Delegates for accounts, moving a user from one Organizational Unit to another, managing large number of account and group attributes.

## Supported features

---

The SailPoint Google Apps Connector supports the following features:

- Account Management
  - Manage Google Apps Users as Accounts
  - Aggregate, Partitioning Aggregation, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable, Change Password
  - Add/Remove Entitlements
  - Manages Delegated Administrators and Alias on Accounts
  - Move user to other Organization Unit
- Account - Group Management
  - Manage Google Apps Groups as Account - Groups
  - Aggregate, Refresh Group
  - Create, Update, Delete

### References

- “Appendix C: Partitioning Aggregation”

## Pre-requisites

---

**Note:** If GoogleApps Connector is behind proxy server, see the “Special Java Considerations” section of the *SailPoint IdentityIQ Installation Guide*.

1. Enable API access from **Admin Console**.  
Select **Security => API Reference => API Access** and select **Enable API access**.
2. Create a project in **Google Developers Console**.
3. Select **APIs & auth** in the left side bar.  
In the list of displayed APIs, ensure that the Admin SDK and Groups Settings API status is set to ON.
4. Select **APIs & auth => Credentials** in the left side bar.  
Create a Client ID for Web Application and note the Client ID and Client Secret.
5. Acquire refresh token for the following scopes:

Scope	Purpose
<a href="https://www.googleapis.com/auth/admin.directory.group">https://www.googleapis.com/auth/admin.directory.group</a>	Group Provisioning
<a href="https://www.googleapis.com/auth/admin.directory.user">https://www.googleapis.com/auth/admin.directory.user</a>	User Provisioning
<a href="https://www.googleapis.com/auth/apps.groups.settings">https://www.googleapis.com/auth/apps.groups.settings</a>	Group Settings APIs
<a href="https://apps-apis.google.com/a/feeds/emailsettings/2.0/">https://apps-apis.google.com/a/feeds/emailsettings/2.0/</a>	Gmail Delegate Provisioning

6. Navigate to **Google Apps Admin Console => Dashboard => Google Apps => Gmail => User settings => Mail Delegation**.  
Verify if **Let users delegate access to their mailbox to other users in the domain** is selected.

For more information about the above listed pre-requisites, see the following respective links:

- To get a refresh token  
<https://developers.google.com/accounts/docs/OAuth2WebServer>
- Pre-requisites for using user and group APIs  
<https://developers.google.com/admin-sdk/directory/v1/guides/prerequisites>
- For more details about pre-requisites for using groups settings APIs  
<https://developers.google.com/admin-sdk/groups-settings/prerequisites>

## Administrator permissions

---

The application user should be configured to have the **Super Admin** role.

# Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Google Apps Connector uses the mandatory connection attributes listed in the following table:

Parameters	Description
Client ID	Client ID of user application.
Client Secret	Client Secret of user application.
Refresh Token	Refresh token value.
Domain Name	Name of the domain to be managed.
Read Group Details	<ul style="list-style-type: none"> <li>• <b>Y:</b> Reads all group attributes during group aggregation. This will take longer to complete the aggregation.</li> <li>• <b>N:</b> Reads minimum group attributes during group aggregation.</li> </ul>

# Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports the following types of objects:

- **Account:** Account objects are used when building identities Link objects.
- **Group:** The group schema is used when building AccountGroup objects that are used to hold entitlements shared across identities.

## Account attributes

---

The following table lists the account attributes:

## Schema attributes

Attributes	Description
objectID	Unique ID of the user
primaryEmail	Primary E-mail ID of user.  <b>Note: Domain of email Id must be same as that of application domain.</b>
name	Full name of the user.
isAdmin	Is user an administrator.
isDelegatedAdmin	Is user a delegated administrator.
lastLoginTime	Last login time of user.
suspended	Is user suspended.
suspensionReason	Reason for suspension.
changePasswordAtNextLogin	Indicates if the user is forced to change password at next login.
ipWhiteListed	Indicate if user's IP address is white listed.
ims	The user's Instant Messenger (IM) accounts.
emails	A list of the user's E-mail addresses.
externalIds	A list of external IDs for the user, such as an employee or network ID.
relations	A list of the user's relationships to other users.
addresses	A list of the user's addresses.
organizations	List of organizations the user belongs to
phones	A list of the user's phone numbers.
aliases	List of the user's alias E-mail addresses.
nonEditableAliases	List of the user's non-editable alias E-mail addresses.
customerId	The customer ID to retrieve all account users.
orgUnitPath	The full path of the parent organization associated with the user.
isMailboxSetup	Indicates if the user's Google mailbox is created.
includeInGlobalAddressList	Indicates if the user's profile is visible in Global Address List when the contact sharing feature is enabled for the domain.
thumbnailPhotoUrl	Photo Url of the user
delegatedAdmins	Delegated administrators of a user.
Groups	Groups connected to the user.

## Group attributes

---

The following table lists the group attributes:

Attribute	Description
objectID	ID of group
email	Group E-mail address.
name	Name of the group.
directMembersCount	Number of group members.
description	Description of the group.
nonEditableAliases	List of the group's non-editable alias E-mail addresses that are outside of the account's primary domain or sub domains. These are functioning E-mail addresses used by the group. This is a read-only property.
adminCreated	Whether it is created by administrator.
aliases	Group aliases.
members	Members of the group.
whoCanJoin	<p>Permissions to join the group.</p> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• ALL_IN_DOMAIN_CAN_JOIN: Anyone in the account can join.</li> <li>• ANYONE_CAN_JOIN: Anyone outside your domain can join.</li> <li>• CAN_REQUEST_TO_JOIN: Non group members can request an invitation to join.</li> <li>• INVITED_CAN_JOIN: Candidates for membership can be invited to join.</li> </ul>
whoCanViewMembership	<p>Permissions to view membership. Possible values -</p> <ul style="list-style-type: none"> <li>• ALL_IN_DOMAIN_CAN_VIEW: Anyone in the account can view the group members list.</li> <li>• ALL_MANAGERS_CAN_VIEW: The group managers can view group members list.</li> <li>• ALL_MEMBERS_CAN_VIEW: Group members can view the group members list</li> </ul>
whoCanViewGroup	<p>Permissions to view group.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• ALL_IN_DOMAIN_CAN_VIEW: Anyone in your account can view this group's messages.</li> <li>• ALL_MANAGERS_CAN_VIEW: Any group manager can view this group's messages.</li> <li>• ALL_MEMBERS_CAN_VIEW: All group members can view the group's messages.</li> <li>• ANYONE_CAN_VIEW: Any Google Apps user can view the group's messages.</li> </ul>

## Schema attributes

whoCanInvite	<p>Permissions to invite members.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• ALL_MANAGERS_CAN_INVITE: Only managers can invite a new member. this includes the group's owner.</li> <li>• ALL_MEMBERS_CAN_INVITE: Managers and members can invite a new member candidate.</li> </ul>
allowExternalMembers	A Boolean indicating if Google Apps users external to your account can view or become members of this group.
MANAGERS	Managers of a group.
OWNERS	Owners of a group.
whoCanPostMessage	<p>Permissions to post messages to the group.</p> <p>Possible values:</p> <ul style="list-style-type: none"> <li>• ALL_IN_DOMAIN_CAN_POST: Anyone in the account can post a message.</li> <li>• ALL_MANAGERS_CAN_POST: Managers, including group owners, can post messages.</li> <li>• ALL_MEMBERS_CAN_POST: Any group member can post a message.</li> <li>• ANYONE_CAN_POST: Any Google Apps user outside the account can access the Google Groups service and post a message.</li> <li>• NONE_CAN_POST: The group is disabled and archived. No one can post a message to this group.</li> </ul>
allowWebPosting	A Boolean indicating if any member allowed to post to the group web forum.
primaryLanguage	Language tag for a group's primary language.
maxMessageBytes	The maximum size of a message.
isArchived	A Boolean indicating if the contents of the group to be archived.
archiveOnly	A Boolean indicating if the group to be only archived.
messageModerationLevel	<p>Moderation level for messages.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>• MODERATE_ALL_MESSAGES: All messages are approved by owner before the message is sent to the group.</li> <li>• MODERATE_NEW_MEMBERS: All messages from new members are approved by owner before the message is sent to the group.</li> <li>• MODERATE_NONE: No moderator approval is required.</li> <li>• MODERATE_NON_MEMBERS: All messages from non-group members are approved by owner before the message is sent to the group.</li> </ul>

spamModerationLevel	Moderation levels for messages detected as spam.  Possible values are: <ul style="list-style-type: none"><li>• ALLOW: Post the message to the group.</li><li>• MODERATE: Send the message to the moderation queue.</li><li>• SILENTLY_MODERATE: Send the message to the moderation queue, but do not send notification to moderators.</li><li>• REJECT: Immediately reject the message.</li></ul>
replyTo	The default reply to a message is sent here.  Possible values are: <ul style="list-style-type: none"><li>• REPLY_TO_CUSTOM: For replies to messages, use the group's custom E-mail address.</li><li>• REPLY_TO_IGNORE: Group users individually decide where the message reply is sent.</li><li>• REPLY_TO_LIST: Reply message is sent to the group.</li><li>• REPLY_TO_MANAGERS: Reply message is sent to the group's managers.</li><li>• REPLY_TO_OWNER: Reply is sent to the owner(s) of the group.</li><li>• REPLY_TO_SENDER: Reply is sent to author of message.</li></ul>
customReplyTo	An E-mail address used when replying to a message.
sendMessageDenyNotification	A Boolean indicating if the members are notified if his message is denied by owner.
defaultMessageDenyNotificationText	Text sent to the message's author as part of rejection notification.
showInGroupDirectory	A Boolean indicating if group is listed in the Groups directory.
allowGoogleCommunication	A Boolean allowing Google to contact group administrators.
membersCanPostAsTheGroup	A Boolean indicating if members can post using the group E-mail address.
messageDisplayFont	Default message's display font.  Possible values are: <ul style="list-style-type: none"><li>• DEFAULT_FONT: Uses the account's default font.</li><li>• FIXED_WIDTH_FONT: Uses fixed width font.</li></ul>
includeInGlobalAddressList	A Boolean indicating if group is included in the <a href="#">Global Address List</a> .

# Provisioning Policy attributes

---

This section lists the provisioning policy attributes for the following:

- Create Account
- Update Account
- Create Group
- Update Group

**Note:** In this section all the attributes marked with the \* sign indicate that the attributes are mandatory.

## *Create Account*

The following table lists the provisioning policy attributes for Create Account:

Attribute	Description
familyName*	The user's last name.
givenName*	The user's first name.
password*	Password for the user account.
primaryEmail*	Primary E-mail address of the user
suspended	A Boolean indicating if the user is suspended.
changePasswordAtNextLogin	Indicates if the user is forced to change their password at next login.
hashFunction	Hash function name for password. Values: MD5, SHA-1 and crypt.
includeInGlobalAddressList	A Boolean indicating if the user's profile is visible in the Global Address List.
ipWhitelisted	A Boolean indicating if user's IP address is white listed.
organizationUnit	Full path of the parent organization of the user. Root organization is represented as forward slash (/).
<b>Address Attributes</b>	
country	Country
countryCode	Country code
extendedAddress	Extended addresses, such as an address that includes a sub-region.
locality	The town or city of the address.
poBox	The post office box, if present.
postalCode	The ZIP or postal code, if applicable.
primaryAddress	A Boolean indicating if this is primary address of the user
region	The abbreviated province or state.
sourceIsStructured	A Boolean indicating if the user-supplied address was formatted. Formatted addresses are not currently supported.

streetAddress	The street address. Whitespace within the string is ignored.
addressType	The address type. Values: custom, home, other, work
addressCustomType	Custom address type.
<b>Email Address Attributes</b>	
emailAddress	User's primary E-mail address or an alias.
ifPrimary	A Boolean indicating if this is the user's primary E-mail.
emailtype	The type of the E-mail account. Values: custom, home, other, work
emailCustomType	Custom E-mail type.
<b>External IDs Attributes</b>	
externalIdsType	The type of the ID. Values: account, custom, customer, network, organization
externalIdsCustomType	Custom external ID type.
externalIdsValue	The value of the ID.
<b>Messenger Attributes</b>	
IMType	IM Type. Values: custom, home, other, work
IMCustomType	Custom IM type.
IMID	The user's IM network ID.
primaryIM	A Boolean indicating if this is the user's primary IM.
IMProtocol	An IM protocol identifies the IM network. The value can be a custom network or the standard network. Values: <ul style="list-style-type: none"> <li>• custom_protocol: A custom IM network protocol</li> <li>• aim: AOL Instant Messenger protocol</li> <li>• gtalk: Google Talk protocol</li> <li>• icq: ICQ protocol</li> <li>• jabber: Jabber protocol</li> <li>• msn: MSN Messenger protocol</li> <li>• net_meeting: Net Meeting protocol</li> <li>• qq: QQ protocol</li> <li>• skype: Skype protocol</li> <li>• yahoo: Yahoo Messenger protocol</li> </ul>
<b>Organization Unit Details</b>	
organizationName	Name of the organization.
costCenter	The cost center of the user's organization.
department	Specifies the department within the organization
description	Description of the organization.
domain	Domain the organization belongs to.
organizationLocation	Physical location of the organization.

## Provisioning Policy attributes

primaryOrganization	A Boolean indicating if this is the user's primary organization.
organizationSymbol	Text string symbol of the organization.
organizationTitle	User's title within the organization
organizationType	Type of organization. Values: unknown, school, work, domain_only
organizationCustomType	Custom organization type.
<b>Phone Attributes</b>	
primaryPhone	A Boolean indicating if this is user's primary phone number.
phoneNumber	Phone number.
phoneType	The type of phone number. Values: custom, home, work, other, home_fax, work_fax, mobile, pager, other_fax, compain_main, assistant, car, radio, isdn, callback, telex, tty_tdd, work_mobile, work_pager, main, grand_central
phoneCustomType	Custom phone type.
<b>Relation Attributes</b>	
relation	The name of the person the user is related to.
relationType	The type of relation. Values: custom, spouse, child, mother, father, parent, brother, sister, friend, relative, domestic_partner, manager, assistant, referred_by, partner
relationCustomType	Custom relation type.

### Update Account

The following table lists the provisioning policy attributes for Update Account:

Attribute	Description
delegatedAdmins	The user's delegated administrators. Values are String or List of E-mail IDs.
name	<p>Full Name of the user. The value of the attribute must consist values for the following attributes:</p> <ul style="list-style-type: none"> <li>• givenName</li> <li>• familyName</li> </ul> <p>The value of the name attribute must be in the JSON format. For example, { "givenName" : "abc" , "familyName" : "xyz" }</p>
aliases	The user's aliases. Values are String or List of E-mail IDs.
organizationUnit	Full path of the organization unit the user should be moved to. For example, test.com/Marketing/Manager where Manager is the organization unit the user should be moved to.
hashFunction	Hash function name for password. Values: MD5, SHA-1 and crypt

includeInGlobalAddressList	A Boolean indicating if the user's profile is visible in the Global Address List.
ipWhitelisted	A Boolean indicating if user's IP address is white listed.
primaryEmail	The user's primary E-mail address.
addresses#	A list of the user's addresses.  Each address can consist of values for the following attributes: type, customType, sourceIsStructured, formatted, poBox, extendedAddress, streetAddress, locality, region, postalCode, country, primary, countryCode
emails#	A list of the user's E-mail addresses.  Each email can consist of values for the following attributes: address, type, customType, primary
externalIds#	A list of external IDs for the user, such as an employee or network ID. Each externalId can consist of values for the following attributes: value, type, customType
ims#	A list of user's Instant Messenger (IM) accounts. Each ims can consist of values for the following attributes: type, customType, protocol, customProtocol, im, primary
organizations#	A list of organizations the user belongs to. Each organization can consist of values for the following attributes: name, title, primary, type, customType, department, symbol, location, description, domain, costCenter
phones#	A list of the user's phone numbers. Each phone can consist of values for the following attributes: value, primary, type, customType
relations#	A list of the user's relationships to other users. Each relation can consist of values for the following attributes: value, type, customType

### Create Group

The following table lists the provisioning policy attributes for Create Group:

Attribute	Description
email*	The group's E-mail address.
name	The group's name.
description	Description of the group.

### Update Group

The following table lists the provisioning policy attributes for Update Group:

Attribute	Description
name	Name of the group.

## Provisioning Policy attributes

description	Description of the group.
MANAGERS	Managers of a group.
OWNERS	Owners of a group.
email	(Required when creating a group) The group's email address. If your account has multiple domains, select the appropriate domain for the email address. The email must be unique.
whoCanJoin	Permissions to join the group.  Values: ALL_IN_DOMAIN_CAN_JOIN, ANYONE_CAN_JOIN, CAN_REQUEST_TO_JOIN, INVITED_CAN_JOIN
whoCanViewMembership	Permissions to view membership.  Values: ALL_IN_DOMAIN_CAN_VIEW, ALL_MANAGERS_CAN_VIEW, ALL_MEMBERS_CAN_VIEW
whoCanViewGroup	Permissions to view group.  Values: ALL_IN_DOMAIN_CAN_VIEW, ANYONE_CAN_VIEW, ALL_MANAGERS_CAN_VIEW, ALL_MEMBERS_CAN_VIEW,
whoCanInvite	Permissions to invite members.  Values: ALL_MANAGERS_CAN_INVITE, ALL_MEMBERS_CAN_INVITE
allowExternalMembers	A Boolean indicating if Google Apps users external to your account can view or become members of this group
whoCanPostMessage	Permissions to post messages to the group.  Values: ALL_IN_DOMAIN_CAN_POST, ALL_MANAGERS_CAN_POST, ALL_MEMBERS_CAN_POST, ANYONE_CAN_POST, NONE_CAN_POST
allowWebPosting	A Boolean indicating if members are allowed posting to the group web forum.
primaryLanguage	Language tag of the primary language for the group
maxMessageBytes	Maximum size of a message, which, by default, is 1Mb.
isArchived	A Boolean indicating if the contents of the group are archived.
archiveOnly	A Boolean indicating if the group is only archived.
messageModerationLevel	Moderation level for messages.  Values: MODERATE_ALL_MESSAGES, MODERATE_NEW_MEMBERS, MODERATE_NONE, MODERATE_NON_MEMBERS.
spamModerationLevel	Sets moderation levels for messages detected as spam.  Values: ALLOW, MODERATE, SILENTLY_MODERATE, REJECT
replyTo	The default reply to a message is set here.  Values: REPLY_TO_CUSTOM, REPLY_TO_IGNORE, REPLY_TO_LIST, REPLY_TO_MANAGERS, REPLY_TO_OWNER, REPLY_TO_SENDER
customReplyTo	If ReplyTo is REPLY_TO_CUSTOM, the customReplyTo must hold a custom E-mail address

sendMessageDenyNotification	A Boolean indicating if the member is notified if his message is denied by owner.
defaultMessageDenyNotificationText	Text for the rejection notification sent to the message's author.
showInGroupDirectory	A Boolean indicating if the group should be listed in the Groups directory.
allowGoogleCommunication	A Boolean allowing Google to contact group administrators.
membersCanPostAsTheGroup	A Boolean indicating if group members can post messages using the group's email address instead of the member's own email address.
messageDisplayFont	Default message's display font. Values: DEFAULT_FONT, FIXED_WIDTH_FONT
includeInGlobalAddressList	A Boolean indicating if the group needs to be included in the <a href="#">Global Address List</a> .

\* Mandatory attributes.

# Values for these attributes need to be in JSON format. It should set values for all or some of the sub-attributes of respective attribute mentioned in the above table. For example, below request updates emails attribute of an account:

```
<AttributeRequest name="emails" op="Add">
  <Value>
    <List>
      <String>{"address": "abc1@test.com", "type": "work", "primary": "true"}</String>
      <String>{"address": "abc2@test.com", "type": "work"}</String>
    </List>
  </Value>
</AttributeRequest>
```

## Troubleshooting

---

### 1 - Account Aggregation fails with error

The account aggregation may fail due to insufficient permissions or in case required services are not turned on.

**Resolution:** To test a GET (read) user using Admin SDK Directory API from browser, use the following URL replacing the userEmail and accessToken values:

[https://www.googleapis.com/admin/directory/v1/users/userEmail?access\\_token=ya29.AHES6ZQRwqu6I9-EUNeSAQS8HoI2YKsLLriUKxAh5-UmIGsh-NEhgOA](https://www.googleapis.com/admin/directory/v1/users/userEmail?access_token=ya29.AHES6ZQRwqu6I9-EUNeSAQS8HoI2YKsLLriUKxAh5-UmIGsh-NEhgOA)

### 2 - Group aggregation fails with error

The group aggregation fails due to insufficient permissions or in case required services are not turned on.

**Resolution:** To test a GET (read) group using Admin SDK Directory API from browser, use the following URL replacing the groupEmail and accessToken values:

## Troubleshooting

[https://www.googleapis.com/admin/directory/v1/groups/groupEmail?access\\_token=ya29.1.AADtN\\_UOeNHQjnEDkjoMTkiswabM7fzSeWWpknYy\\_qOLSQevSAb0HEK2djECyy&token\\_type=Bearer](https://www.googleapis.com/admin/directory/v1/groups/groupEmail?access_token=ya29.1.AADtN_UOeNHQjnEDkjoMTkiswabM7fzSeWWpknYy_qOLSQevSAb0HEK2djECyy&token_type=Bearer)

To test a GET (read) group using Google Groups Settings API (Read Group Details – Y) from browser, use the following URL after replacing the groupEmail and accessToken values:

[https://www.googleapis.com/groups/v1/groups/groupEmail?access\\_token=ya29.1.AADtN\\_W2DdsN1YkvRc7meVgQ6XDKIOqgZsbA-Nt9O9zNWEcoF7TLVUGXKDwkwcnmzyY7H4&token\\_type=Bearer&alt=json](https://www.googleapis.com/groups/v1/groups/groupEmail?access_token=ya29.1.AADtN_W2DdsN1YkvRc7meVgQ6XDKIOqgZsbA-Nt9O9zNWEcoF7TLVUGXKDwkwcnmzyY7H4&token_type=Bearer&alt=json)

### 3 - Error message appears for some corrupt objects

During aggregation, error messages appear for some corrupt objects.

**Resolution:** Corrupt objects can be skipped at the time of aggregation by setting the `isContinueOnError` attribute to true. By default, the value of `isContinueOnError` attribute is false. This value can be set to true in the Application XML as follows:

```
<entry key="isContinueOnError" value="true" />
```

### 4 - Internal Server and Service Unavailable Error appears

The Internal Server and Service Unavailable error messages are sent by the Google Server.

**Resolution:** To retry the request, use `maxReadRetryCount` attribute. The retry count is set to 5 by default. Increase the retry count by adding the following entry to the Application XML and set the desired value:

```
<entry key="maxReadRetryCount" value="10" />
```

### 5 - SocketTimeoutException error

The `SocketTimeoutException` error message appears.

**Resolution:** Increase the timeout interval by adding the `maxReadTimeout` attribute to the application debug. By default the value of `maxReadTimeout` attribute is 180 seconds. To increase the timeout, add the following entry to the Application XML and set the desired value:

```
<entry key="maxReadTimeout" value="240" />
```

### 6 - Provisioning (Create Account) fails with an error message

Provisioning (Create Account) fails with the following error message:

Resource Not Found: domain

**Resolution:** Verify the domain name of primary email as the domain name of email id must be same as the GoogleApps-Direct application domain name.

### 7 - While updating the accounts from Integration Console, attributes having list values are not deleted from the account

While updating the accounts from Integration Console, attributes having list values are not deleted from the account.

**Resolution:** While deleting the attributes from Integration Console, it must be consider that those attributes are present in Google Apps managed system for that account.

If you are setting the "primary=false" in plan, then Google Apps does not consider that attribute. Hence while deleting any attribute (Organizations, phones and so on.) "primary" attribute type must not be present in the plan.

For example,

```
<AttributeRequest name="phones" op="Add">
  <Value>
    <List>
      <String>{"value": "345678", "customType": "", "type": "custom", "primary": "false"}</String>
    </Value>
  </AttributeRequest>

<AttributeRequest name="phones" op="Remove">
  <Value>
    <List>
      <String>{"value": "345678", "customType": "", "type": "custom"}</String>
    </Value>
  </AttributeRequest>
```

In some attributes like IMS, some required attribute appear as empty value if those attributes are not passed while adding those attributes. Hence add those attribute value as empty in the delete plan.

For example,

```
<AttributeRequest name="ims" op="Add">
  <Value>
    <List>
      <String>{"im": "test1@dev.sailpoint.com", "type": "work"}</String>
    </List>
  </Value>
  </AttributeRequest>

<AttributeRequest name="ims" op="Remove">
  <Value>
    <List>
      <String>{"im": "test1@dev.sailpoint.com", "type": "work", "customProtocol": ""}</String>
    </List>
  </Value>
  </AttributeRequest>
```

## **Troubleshooting**

# Chapter 44: SailPoint GoToMeeting Connector

---

The following topics are discussed in this chapter:

Overview .....	403
Supported features .....	403
Pre-requisites .....	403
Administrator permissions .....	404
Configuration parameter .....	404
Schema attributes .....	404
Account attributes .....	404
Group attributes .....	404
Provisioning Policy attributes .....	405

## Overview

---

**Note:** SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.

SailPoint GoToMeeting Connector manages GoToMeeting organizers. It supports read and write to GoToMeeting to create, retrieve, update, delete users, and retrieve groups.

## Supported features

---

SailPoint GoToMeeting supports the following features:

- Account Management
  - Manage GoToMeeting Users (excludes invited accounts) as Account
  - Aggregate, Refresh Accounts
  - Create, Delete
  - Enable, Disable
- Account - Group Management
  - Manage GoToMeeting Groups as Account - Groups
  - Aggregate, Refresh Groups

## Pre-requisites

---

**Note:** If GoToMeeting Connector is behind proxy server, see the “Special Java Considerations” section of the *SailPoint IdentityIQ Installation Guide*.

The user must go through the OAuth2 flow to generate the access token. The connector uses this Access Token to make calls to any GoToMeeting REST API.

## Administrator permissions

---

Role of the user must be an Administrator.

## Configuration parameter

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

**Access Token:** A valid Access Token for the user is required which enables your application to access the user's information and take actions on their behalf. The application and user are verified with each API call by passing an access token along with each request.

## Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports the following types of objects:

- **Account:** Account objects are used when building identities Link objects.
- **Group:** The group schema is used when building AccountGroup objects that are used to hold entitlements shared across identities.

### Account attributes

---

The following table lists the account attributes ([Table 1—Account attributes](#)):

**Table 1—Account attributes**

Attributes	Description
OrganizerKey	A unique key associated with each organizer.
FirstName	First Name of the organizer.
LastName	Last Name of the organizer.
Email	Email id of the organizer
Status	The status of the organizer (Active, Invited or Suspended).
GroupKey	A unique key associated with a group of which the organizer is a part of.
Groups	The entitlements of the organizer.
MaximumAttendeesAllowed	The maximum number of attendees allowed for the organizer.

### Group attributes

---

The following table lists the group attributes ([Table 2—Group attributes](#)):

**Table 2—Group attributes**

Attributes	Description
GroupName	The name of the group.
GroupKey	A unique key associated with the group.
ParentKey	The Parent Key of the group.
GroupStatus	The status of the group.
NumberOfOrganizers	The number of organizers in a group.

## Provisioning Policy attributes

---

This following table lists the provisioning policy attributes for create ([Table 3—Provisioning Policy attributes](#)):

**Table 3—Provisioning Policy attributes**

Attributes	Description
OrganizerEmail	A valid email id is required to whom a GoToMeeting invite should be sent.

**Note:** If multiple groups are specified at the time of account creation, the group to which the user is attached would be selected at random, as GoToMeeting supports only one group per user.

## **Provisioning Policy attributes**

# Chapter 45: SailPoint IBM i Connector

---

The following topics are discussed in this chapter:

Overview .....	407
Supported features .....	408
Supported Managed Systems .....	409
Pre-requisites .....	409
Administrator permissions .....	409
Configuration parameters .....	410
Schema Attributes .....	410
Provisioning policy attributes .....	412
Additional information .....	413
Direct Permissions .....	413
Upgrade Consideration .....	414

## Overview

---

**Note: SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

The SailPoint IBM i Connector is a Read/Write connector that uses the User Profiles on IBM i computer for account provisioning. For group provisioning, user profiles which have Group ID are used. The IBM i Connector can be configured to use any of the attributes of user/group which are supported by IBM i commands.

## Supported features

---

SailPoint IBM i Connector supports the following features:

- Account Management
  - Manage IBM i Users as Account
  - Aggregate, Partitioning Aggregation, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable, Change Password
  - Add /Remove Entitlement
- Account - Group Management
  - Manage IBM i Groups as Account - Groups
  - Aggregate, Refresh Groups
  - Create, Update, Delete
- Permissions Management
  - Application reads permissions directly assigned to accounts and groups as direct permissions during accounts and groups aggregations respectively.
  - The connector supports automated revocation of the aggregated permissions for accounts and groups.
- Authorization list
  - Application reads Authorization list assigned to accounts and groups as AUTL attribute during accounts and groups aggregations respectively.
  - The connector supports
    - Aggregating Authorization Lists associated with users
    - Aggregating Authorization Lists associated with groups
    - ADD/Remove Authorization Lists

**Note:** To aggregate direct permissions while aggregation, select the 'Include Permission' check box from account or group schema.

AUT(USER DEFINED) USER DEFINED is not a valid parameter

For example, if user tries to add an authorization list having user defined authority, following error message is displayed:

Failed to execute Command: ADDAUTLE USER(<User Name>) AUTL(<AUTL Name>)  
AUT(USER DEFINED) USER DEFINED is not a valid parameter

## References

- “Direct Permissions” on page 413
- Appendix C: Partitioning Aggregation
- Appendix D: Before and After Provisioning Action

## Supported Managed Systems

---

SailPoint IBM i Connector supports the following versions of IBM i:

- IBM i V7R3
- IBM i V7R2
- IBM i V7R1

## Pre-requisites

---

As we use IBM Toolbox for Java (JTOOpen) to manage IBM i system, in order to connect to the IBM i system and access its data and services, the IBM Toolbox for Java (JTOOpen) requires TC1 Licensed Program (TCP/IP Connectivity Utilities for IBM i, Host Server option of IBM i to be installed and configured on the system. It also utilizes following services running on the remote port:

Service Name	Port	SSL Port
as-signon	8476	9476
as-rmtcmd	8475	9475
as-svrmapping	449	NA

## Administrator permissions

---

The SailPoint IBM i Connector requires Security officer/User Profile with required permission to accomplish provisioning tasks. The administrator user that is, user configured for the IBM i Connector must be assigned sufficient privileges on IBM i Connector to create/update user profile and group profile.

For example, User profile can be configured as an administrator with any one of the following user classes and special authorities as mentioned in the following table:

**Note:** Depending on the System security level (QSECURITY) the special authorities assigned to the user differs. Ensure that the created user profiles have the following mentioned special authorities.

USER CLASS (USRCLS)	SPECIAL AUTHORITY (SPCAUT)
*USER (user)	*SECADM, *AUDIT, *ALLOBJ
*SECADM (security administrator)	*SECADM, *AUDIT, *ALLOBJ
*SECOFR (security officer)	*NONE
*SYSOPR (system operator)	*SECADM, *AUDIT, *ALLOBJ

## Configuration parameters

**Note:** For configuration of User profile as an Administrator, you can specify different options of user class and special authority in various ways as desired.

The above table displays few examples which mention the different ways in which you can configure administrator.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The SailPoint IBM i Connector uses the following connection attributes:

Attribute	Description
IBM i Host	Host Name/IP Address of IBM i Host.
User Profile	This user is used for get/set operations on IBM i Host.
Password	This field contains the password for user specified as User Profile in application.
Proxy Server	Proxy server host and port details to be used to connect to IBM i Host.
Use SSL	Select this option if using SSL certificates.
Partitioning Statements	Criteria to specify the range of users to be downloaded. For example, if the range is specified as A-M, then this specifies that all the Users who are between A and M (including A and M) would be treated as one partition and downloaded. For more information, see Appendix C: Partitioning Aggregation.

## Schema Attributes

---

This section describes the different schema attributes.

### Account and Account - Group attributes

---

The following table lists the account and account - group attributes:

Attributes	Description
USRPRF	User profile.
AUDLVL	Audit Level.
PWDEXP	Set password to expired.
STATUS	Status
USRCLS	User class
ASTLVL	Assistance level.

Attributes	Description
CURLIB	Current library.
INLPGM	Initial program to call.
INLMNU	Initial menu.
LMTCPB	Limit capabilities.
TEXT	Text description
SPCAUT	Special authority
SPCENV	Special environment
DSPSGNINF	Display sign-on information
PWDEXPITV	Password expiration interval
LCLPWDMGMT	Local password management
LMTDEVSSN	Limit device sessions
KBDBUF	Keyboard buffering
MAXSTG	Maximum allowed storage
PTYLMT	Highest schedule priority
JOBD	Job description
GRPPRF	Group profile
OWNER	Owner
GRPAUT	Group authority
GRPAUTTYP	Group authority type
SUPGRPPRF	Supplemental groups
ACGCDE	Accounting code
OBJAUD	Object Auditing
MSGQ	Message queue
DLVRY	Delivery
SEV	Severity code filter
PRTDEV	Print device
OUTQ	Output queue
ATNPGM	Attention program
SRTSEQ	Sort sequence
LANGID	Language ID
CNTRYID	Country or Region ID
CCSID	Coded character set ID
CHRIDCTL	Character Identifier control
SETOBJATTR	Local job attributes

## Provisioning policy attributes

Attributes	Description
LOCALE	Locale
USR OPT	User options
UID	User ID number
GID	Group ID number (Only for Groups)
HOMEDIR	Home directory.
AUTL	Authorization Lists
<i>If required user must add the following attributes manually to IBM i Connector schema after upgrading to IdentityIQ version 7.1</i>	
PWDLASTCHG	Date and time when the sign on password was changed
PREVSIGNON	Date and time of previous sign on
PWDEXPDATE	Date when password will expire
INVSIGNON	Number of unsuccessful sign on attempt
USREXPACT	User Expiration Action
USREXPDATE	User Expiration Date
USREXPITV	User Expiration Interval
LSTUSEDATE	Last Used Date
CHGDATE	Date and time of the last change to the user profile
CTRBYUSER	Created by user

## Provisioning policy attributes

---

This section lists the provisioning policy attributes of SailPoint IBM i Connector that allows to select the type of user, login, or group.

Attribute name	Description
<b>Create Account</b>	
USRPRF	User profile.
USRCLS	User class.
UID	User ID.
GRPPRF	Group profile.
PASSWORD	Password
PWDEXP	Set password to expired.
<b>Create Group</b>	
USRPRF	User profile.
USRCLS	User class.

Attribute name	Description
GID	Group ID.
PWDEXP	Set password to expired.
<b>Update Group</b>	
USRCLS	User class.
GID	Group ID.

## Additional information

---

This section describes the additional information related to the IBM i Connector.

### Direct Permissions

---

The SailPoint IBM i Connector supports reading direct permissions assigned to user profiles for different resources on IBM i system. The **Include Permissions** check box of account or group schema must be selected to get direct permissions during Aggregation.

The SailPoint IBM i Connector also supports revoking account direct permissions through certifications. The **directPermissionObjectType** configuration parameter contains list of various object types owned by the user. By default the following direct permission object types are configured:

- \*LIB
- \*MSGQ
- \*FILE
- \*PGM
- \*CMD
- \*MENU
- \*AUTL
- \*JOBQ

This above list can be modified as required for different object types. For example:

```
<entry key="directPermission" value="true" />
<entry key="directPermissionObjectType">
    <value>
        <List>
            <String>*LIB</String>
            <String>*MSGQ</String>
        </List>
    </value>
</entry>
```

Perform the following procedure to display direct permissions in IBM i system:

1. Execute DSPOBJAUT.
2. Enter the **Object** and **Object Type**.
3. Press **F11** twice.

## Upgrade Consideration

---

(Optional) To manage the authorization list, add the following attributes in the mentioned schema level with appropriate properties while upgrading to IdentityIQ version 7.1:

- **Account schema:** AUTL with type string and properties as Managed, Entitlement, Multi-Valued
- **Group schema:** AUTL with type string and properties as Entitlement, Multi-Valued

## Create SSL Communication between IdentityIQ and IBM i system

---

Perform the following to enable SSL communication between IdentityIQ and IBM i server, for securing SSL connection for IBM i system

**Note:** For a Java client to connect using SSL and self-signed certificates, install the certificate into the JVM keystore.

1. Export server certificate and copy the exported .cacrt file to the host running IdentityIQ.
2. At the client computer execute the following command from the /jre/lib/security path:  
`keytool -import -alias aliasName -keystore cacerts -trustcacerts -file <absolute path of certificate>`  
In the preceding command line, *aliasName* is the name of the alias.
3. Login to IdentityIQ.
4. Create the application for IBM i by selecting **Use SSL** option and provide all the required values.
5. Click on Test Connection and save the application.

# Chapter 46: SailPoint Microsoft SharePoint Server Connector

---

The following topics are discussed in this chapter:

Overview .....	415
Supported features .....	415
Supported Managed system .....	416
Pre-requisites .....	416
Application Account permissions .....	417
Configuration parameters .....	417
Schema attributes .....	418
Account attributes .....	418
Group attributes .....	418
Provisioning Policy attributes .....	419
Additional information .....	420
Certifications .....	420
Troubleshooting .....	420

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

SharePoint is central platform used for content management and provisioning of variety of business applications. SharePoint integrates intranet, content management, and document management. It is mostly used by midsize businesses and large departments.

The Microsoft SharePoint Server Connector is designed to manage SharePoint users and groups from all the SharePoint Site Collections present on the SharePoint Server via remote execution of the Microsoft SharePoint Server PowerShell commands.

## Supported features

---

The Microsoft SharePoint Server Connector supports the following features:

- Account Management
  - Manage SharePoint Users as Accounts.
  - Account Aggregation
  - Create
  - Add/Remove Entitlements

**Note:** **Active Directory Domain Groups are modeled as users in SharePoint. To avoid creating Identities for such groups, connector skips these domain groups during account aggregation. The membership of domain groups to SharePoint groups can be managed from the groups properties post group aggregation.**

## Overview

- Account - Group Management
  - Manage SharePoint Groups as Account-Group
  - Aggregate Groups
  - Add/remove Active Directory groups from SharePoint groups
- Support native Before/After scripts for provisioning requests.

## References

- “Troubleshooting” on page 420
- “IQService Before/After Scripts” on page 580”
- “Appendix E: IQService”

## Supported Managed system

---

Microsoft SharePoint Server Connector supports following Microsoft SharePoint servers:

- Microsoft SharePoint server 2016
- Microsoft SharePoint server 2013
- Microsoft SharePoint server 2010

## Pre-requisites

---

- Before you can use any of the features of the connector, the IQService must be installed on the computer having the same domain as that of SharePoint Server. For more information about installing IQService see, “Appendix E: IQService”.
- Install PowerShell version 2.0 or later on SharePoint Server.
- To enable the Connector to remotely communicate with SharePoint Server using PowerShell commands, perform the following on SharePoint Server computer:
  - a. Ensure that **WinRM** service is running on SharePoint Server and on IQService system.
  - b. To enable PowerShell Remoting, execute the following command on the system:

```
Enable-PSRemoting -Force
```

Configure Trust in SharePoint Server system and IQService system by running the following command on SharePoint Server:

```
Set-Item wsman:\localhost\client\trustedhosts "<IQService Host>"
```

- c. Set the authentication type as **CredSSP** for remote PowerShell session to work.

On SharePoint Server execute the following command:

```
Enable-WSManCredSSP -Role Server
```

On IQService system execute the following command:

```
Enable-WSManCredSSP -Role client -DelegateComputer "<SharePoint Server System Name>"
```

- d. On SharePoint Server and IQService system, restart the **WinRM** service for the new settings to take effect:

```
Restart-Service WinRM
```

## Hardware and Software requirements

- Windows Server 2012
- Windows Server 2012 R2

## Application Account permissions

---

To perform various operations on different SharePoint Site Collections, provide the following permissions to a SharePoint Server user that must be configured as Application User in IdentityIQ application:

- User must be a part of the following groups on SharePoint Server system:
  - Remote Desktop Users
  - WinRMRemoteWMIUsers
  - WSS\_ADMIN\_WPG

- User must have **SPShellAdmin** access role on all the content databases from the SharePoint Server that this connector must manage. This allows connector to execute SharePoint cmdlets. Execute the following command on SharePoint Server to give the **SPShellAdmin** role to the application user:

```
Add-SPShellAdmin -UserName <DOMAIN\UserName> -Database (Get-SPContentDatabase -Identity "WSS_Content")
```

To grant access to all content databases use the following command:

```
Get-SPDatabase | Add-SPShellAdmin DOMAIN\UserName
```

- On SharePoint Server and IQService host, the Application User must have **Read and Execute** permission for **Microsoft.PowerShell32**. Execute the following command on SharePoint Server and IQService host systems to allow that permission:

```
Set-PSSessionConfiguration -Name "Microsoft.PowerShell32"
-ShowSecurityDescriptorUI
```

- The Application User must have access to all SharePoint Web Applications that must be managed by the connector. Create PowerShell script as follows and execute on SharePoint Management Shell:

```
$webApp = Get-SPWebApplication -Identity "Web App Url"
$webApp.GrantAccessToProcessIdentity("Domain\ User Name ")
Add above lines for each web application.
```

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application.

The Microsoft SharePoint Server Connector uses the connection attributes listed in the following table:

Parameters	Description
SharePoint Server	SharePoint Server host name.
Username	Application account user with permissions as mentioned in “Application Account permissions” in Domain\User form.
Password	Password of the user.
IQService Host	Name of the host computer where IQService is deployed and running.
IQService Port	Port of communication of IQService.

## Schema attributes

Parameters	Description
Page Size	Number of objects fetched per page, when iterating over large numbers of objects. (Default: 500).

## Additional configuration parameter

Parameter	Description
setAttributeLevelResult	Enables setting results for every attribute request. <b>Note:</b> Enabling this parameter will result in delay as every attribute request is committed in Active Directory instead of bulk commit.
skipDomainGroups	Skips Active Directory Domain Groups during account aggregation. Default: True

# Schema attributes

This section describes the different schema attributes.

## Account attributes

The following table lists the account attributes ([Table 1—Account attributes](#)):

**Table 1—Account attributes**

Attributes	Description
AccountName	Account name of the user.
DisplayName	Display name of the user.
Email	E-mail address of the user.
Groups	Collection of groups of which the user is a member.
OwnedGroups	Groups that the user owns.
SiteCollections	SharePoint Site Collections associated with the user.
SID	Unique security ID for the network account of the user.
IsDomainGroup	If user is a domain group.
Notes	Notes in the user properties.
UserName	sAMAccountName of the user.

## Group attributes

The following table lists the group attributes ([Table 2—Group attributes](#)):

**Table 2—Group attributes**

Attributes	Description
GroupUrl	Identity attribute of group in <b>ParentWeb.Url\GroupName</b> format.
GroupName	Name of the group.
LoginName	Login name of the group.
ID	Identifier (ID) for the group.
ParentWeb	Parent web of the group.
Description	Description for the group.
AllowMembersEditMembership	Who can edit the membership.
OnlyAllowMembersViewMembership	Who can view the membership of the group.
AutoAcceptRequestToJoinLeave	Whether membership requests are automatically accepted.
AllowRequestToJoinLeave	Whether to allow users to request for membership of the group.
RequestToJoinLeaveEmailSetting	Membership requests to this e-mail address.
Owner	Name of the group owner.
SiteAdmin	Name of site collection administrator.
ADGroups	List of Active Directory groups which are member of this SharePoint Group.

## Provisioning Policy attributes

---

This following table lists the provisioning policy attributes ([Table 3—Provisioning Policy attributes](#)):

**Table 3—Provisioning Policy attributes**

Attributes	Description
<b>Provisioning policy attributes for Account creation</b>	
AccountName*	Login name of the user. The user must exist in the configured SharePoint user store (for example, Active Directory). For Windows Claim based authentication, the user name must be in encoding format. For example, i:0#w contoso\jim
sitecollectionurl	To create user without assigning entitlements, value of this attribute should be list of SharePoint Site Collections this user must be added to. This attribute is not applicable when entitlements are requested in create request.

## Additional information

**Table 3—Provisioning Policy attributes (Continued)**

Attributes	Description
<b>Provisioning policy attributes for Group update</b>	
ADGroups	List of domain group that must be added or removed from the SharePoint group.

Note: Attributes marked with \* sign are the mandatory attributes.

## Additional information

This section describes the additional information related to the Microsoft SharePoint Server Connector.

## Certifications

During group aggregation SharePoint Site Collection administrator can be promoted as owner of the entitlement. This facilitates in assigning certification for each SharePoint Site Collection to respective SharePoint Site Collection Administrator using **Entitlement Owner** certification. To configure this, import **Promote SiteAdmin as Entitlement Owner** rule from exemplerrules.xml file and set this rule as Group Aggregation Refresh Rule in task before running the Account-Group aggregation.

## Troubleshooting

### 1 - Exception while creating new PowerShell Session

The following error message appears in many scenarios:

Exception while creating new PowerShell Session

**Resolution:** Perform the following:

1. Ensure that IQService and SharePoint Server are in the same domain.
2. Verify if maximum number of PowerShell users allowed for Application User are not exceeding. Verify by executing the following command:  
`Get-Item WSMan:\localhost\Shell\MaxShellsPerUser`  
If number is not adequate then increase the number. For example,  
`Set-Item WSMan:\localhost\Shell\MaxShellsPerUser 50`
3. Application Account has enough privileges as described in “Application Account permissions”.
4. Allocate enough memory for PowerShell session. Verify memory space using the following command:  
`Get-Item WSMan:\localhost\Shell\MaxMemoryPerShellMB`  
If number is not adequate then increase the number. For example,  
`Get-Item WSMan:\localhost\Shell\MaxMemoryPerShellMB 256`

### 2 - The WS-Management service cannot process the request

The following error message appears when the user has exceeded the maximum number of concurrent shells:

The WS-Management service cannot process the request. This user is allowed a maximum number of 5 concurrent shells, which has been exceeded. Close the existing shells or raise the quota for this user.

**Resolution:** Verify if maximum number of PowerShell users allowed for Application User are not exceeding by executing the following command:

```
Get-Item WSMan:\localhost\Shell\MaxShellsPerUser
```

If number is not adequate increase it. For example,

```
Set-Item WSMan:\localhost\Shell\MaxShellsPerUser 50
```

### 3 - Access is denied

**Resolution:** Verify the following:

- if username and password are correctly entered.
- if this application user has enough access on SharePoint Server.

### 4 - Failed to connect to the SharePoint Server

**Resolution:** Verify if SharePoint Server is accessible through IdentityIQ server via IQService. Verify if IP or hostname are resolving correctly.

### 5 - Account aggregation fails

Account aggregation fails with the following error message in IQService logs:

Site Collection Administrator for Site collection: is =>

**Resolution:** Run the following command to get the user added as claim user:

```
$webapp=get-spwebapp -Identity "SITE_URL"  
$webapp.grantaccesstoprocessidentity(''DOMAIN\LOGON_USER'')
```

## **Troubleshooting**

# Chapter 47: SailPoint Microsoft SharePoint Online Connector

---

The following topics are discussed in this chapter:

Overview .....	423
Supported features .....	423
Prerequisites .....	424
Administrator permissions .....	425
Configuration parameters .....	425
Schema attributes .....	425
Account attributes .....	426
Group attributes .....	426
Provisioning Policy attributes .....	426
Additional information .....	427
Unstructured Target Collector .....	427

## Overview

---

**Note: SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

SailPoint Microsoft SharePoint Online Connector manages the users, SharePoint groups and their attributes present on the Microsoft SharePoint Online directory. It does not manage the attributes associated with other products in Microsoft Office 365 suite like Office 365 Online directory store, Exchange Online, Lync Online.

The SharePoint online connector uses Windows Identity Foundation for the authentication and web services for managing users and groups to implement its functionalities in IQService which needs to be running on Windows 7 or Windows Server 2008 R2 computer.

## Supported features

---

SailPoint Microsoft SharePoint Online Connector supports the following features:

- Account Management
  - Manage Microsoft SharePoint Online Users as Accounts
  - Aggregate, Refresh Accounts
  - Create, Update, Delete
  - Add/Remove Entitlements
- Account - Group Management
  - Manage Microsoft SharePoint Online Groups as Account-Groups
  - Aggregate, Refresh Groups
  - Create, Delete

## Overview

- Permissions Management
  - SharePoint Online application can be configured to read different permissions directly assigned to accounts and groups using Unstructured Target Collector.
  - Supports automated revocation of the aggregated permissions and creates work items for requests only when the default provisioning action is overridden and **Manual Work Item** is selected as the provisioning action.

## References

- “Unstructured Target Collector” on page 427
- “Appendix E: IQService”

## Prerequisites

---

- The IQService must be installed on any windows operating system on which installation of Office 365 Cmdlets/Windows Azure Active Directory Module is supported.  
For more information, see “Appendix E: IQService”.
  - Windows Identity Foundation must be installed on the computer where IQService is installed. To install Windows Identity Foundation, download it from the following site:  
<http://www.microsoft.com/en-us/download/details.aspx?id=17331>
  - While creating user, SharePoint license is also assigned and it requires the Office 365 cmdlets to be present. This is a prerequisite for Office 365 and Exchange Online Connector. To install it perform the following steps:
    - Before installing the Office 365 cmdlets, install the Microsoft Online Services Sign-in Assistant if not already present on the system. Download and install one of the following from the Microsoft Download Center:
      - [Microsoft Online Services Sign-In Assistant \(IDCRL7\) - 32 bit version](#)
      - [Microsoft Online Services Sign-In Assistant \(IDCRL7\) - 64 bit version](#)
    - Install the appropriate version of Windows Azure AD Module for Windows PowerShell for your operating system from the Microsoft Download Center:
      - [Windows Azure Active Directory Module for Windows PowerShell - \(32-bit version\)](#)
      - [Windows Azure Active Directory Module for Windows PowerShell - \(64-bit version\)](#)
- .NET Framework 3.5.1 must be installed on the computer where IQService is installed.
- For target aggregation, redistributable package of SharePoint Foundation 2010 Client Object Model should be installed on the computer where IQService is present. It can be downloaded from the following site:  
<http://www.microsoft.com/en-us/download/details.aspx?id=21786>

## Administrator permissions

---

- Administrator should be a part of the Global Administrator role in Office 365. The administrator role can be changed from **Users=>Setting=>Assign Role=>Global Administrator**.
- The administrator should also be a Site Collection owner. The site collection owner can be assigned to the administrator from **Site Collections=>Select site which is to be managed=>Owners=>Manage Administrators=>Add the user to Site Collection Administrators**.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Microsoft SharePoint Online Connector uses the configuration parameters listed in the following table ([Table 1—Configuration parameters](#)):

**Table 1—Configuration parameters**

Parameters	Description
IQService Host*	Host name of the system where IQService is installed.
IQService Port*	Port number on which IQService is listening. Default: 5050.
Site Collection URL*	URL of SharePoint Online site to manage.
Administrator User ID*	User ID or user principal name of the administrator.
Administrator Password*	Password of the administrator.
Page Size	The number of objects to fetch in a single page when iterating over large data sets. Default: 500.

\* Indicates the mandatory attributes to create the application.

## Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. IdentityIQ currently supports the following types of objects:

**Account:** Account objects are used when building identities Link objects.

**Group:** The group schema is used when building AccountGroup objects that are used to hold entitlements shared across identities.

## Provisioning Policy attributes

### Account attributes

---

The following table lists the account attributes ([Table 2—Account attributes](#)):

**Table 2—Account attributes**

Attributes	Description
LoginName	Login Name of the user.
UserName	User name of the user.
Name	Display Name of user.
Email	E-mail address of the user.
IsSiteAdmin	Specifies whether the user is a site collection administrator.
FirstName	First name of the user.
LastName	Last Name of the user.
PreferredName	Preferred name of the user.
ID	Member ID for the user.
WorkPhone	Office phone number of the user.
Groups	It specifies all the groups to which user belongs to.

### Group attributes

---

The following table lists the group attributes ([Table 3—Group attributes](#)):

**Table 3—Group attributes**

Attributes	Description
Name	Display name of the group.
Description	Description of the group.
OwnerId	Owner Id of the group.
ID	Identifier (ID) for the group.
OwnerIsUser	Owner type.

## Provisioning Policy attributes

---

[Table 4—Provisioning Policy attributes for create, delete, and create group](#) lists the provisioning policy attributes for create, delete, and Create Group respectively.

**Table 4—Provisioning Policy attributes for create, delete, and create group**

Attributes	Description	Required attribute
<b>Provisioning policy attributes for create</b>		
UserName	User Name of the User.	Yes

**Table 4—Provisioning Policy attributes for create, delete, and create group (Continued)**

Attributes	Description	Required attribute
Email	Email of the user.	No
AccountSkuld	Licensing plans that are available for SharePoint (AccountSkuld). These plans can be retrieved by executing the powershell cmdlet Get-MsolAccountSku and it will be displayed in domain:plan format under the AccountSkuld column.  <b>Note:</b> Before provisioning, replace/modify the default values of 'AccountSkuld' in provisioning policy with the values retrieved as mentioned above.	Required only if multiple licensing plans are available.
<b>Provisioning policy attributes for delete</b>		
RemoveSharePointLicense	Whether or not the SharePoint license should be removed after deleting the account.	No
AccountSkuld	Licensing plans that are available for SharePoint (AccountSkuld). These plans can be retrieved by executing the powershell cmdlet Get-MsolAccountSku and it will be displayed in domain:plan format under the AccountSkuld column.  <b>Note:</b> Before provisioning, replace/modify the default values of 'AccountSkuld' in provisioning policy with the values retrieved as mentioned above.	Required only if <b>RemoveSharePointLicense</b> attribute is true and there are multiple licensing plans available.
<b>Provisioning policy attributes for create group</b>		
Name	Name of the group.	
DefaultUser	Default user of the group.	
Permission	Group permission.	
Owner	Owner of the group.	
OwnerIsUser	Owner type.	
Description	Description of the group.	

## Additional information

---

This section describes the additional information related to the Microsoft SharePoint Online Connector.

### Unstructured Target Collector

---

SharePoint Online uses a data structure which requires the configuration of the **Unstructured Targets** tab to collect targeted data and correlates it with **LoginName** for Accounts and **Name** for groups. The **LoginName** for Accounts and **Name** for groups attributes should be marked as Correlation Key in respective schema. For more information on the Unstructured Targets Tab, see “Unstructured Targets Tab” section of the *SailPoint User’s Guide*.

SharePoint Online Target Collector supports aggregation for Sites, Lists, List Items, Folders and Files. The objects can be filtered based on various filters configured on the Unstructured Targets Tab.

## Additional information

Attribute	Description	Possible values
Site Collection URL	URL of Site or Site Collection for target aggregation.	URL. Cannot be blank
UserName	User to be used for aggregating the site targets.	admin@domain.onmicrosoft.com Cannot be blank.
Password	Password for UserName	
Target Types Filter	As mentioned above, the Target Collector supports aggregating Sites, Lists, List Items and Files. Using this filter, any of these target types can be selectively aggregated.	Any combination of following separated by comma:  Sites,Files,Lists,ListItems,Folders,Files  ListItem specific filtration - for example, Document,Discussion,Picture,Wiki Page and so on  If not specified, all target types would be aggregated.  Default – Not specified
Include inherited permissions	SharePoint has hierarchy structure for targets. The child target can inherit permissions from the parent. In that case, the permissions of child and parent would be same. For example, all files in a folder can inherit permissions from folder. Hence aggregating file permissions may not be of interest. This filter can include or exclude such targets.	True/False  If true, the target aggregation will fetch all targets including the one having inherited permissions.  Default: True
Site Filter Type	This is used in combination to the Site Filter. This tells whether the Site Filter define the inclusion filter or exclusion filter.	Include/Exclude  Default: Include
Site Filter	Targets with path containing Words / phrases mentioned here can be selectively included or excluded depending on the Site Filter Type	Words/phrases separated by comma.  If not specified all the targets would be aggregated.  Default: Not specified

# Chapter 48: SailPoint Microsoft Project Server Connector

---

The following topics are discussed in this chapter:

Overview .....	429
Supported features .....	429
Supported Managed system .....	430
Pre-requisites .....	430
Administrator permissions .....	430
Configuration parameters .....	430
Schema attributes .....	431
Account attributes .....	431
Group attributes .....	431
Provisioning Policy attributes .....	432
Troubleshooting .....	432

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

Microsoft Project Server provides flexible on-premises solution for project portfolio management (PPM) and everyday work.

Microsoft Project Server 2013 offers the following modes of security:

- **Microsoft SharePoint Mode:** Microsoft SharePoint Mode has same security architecture as SharePoint site collection which can be managed through Microsoft SharePoint Server Connector.
- **Microsoft Project Server Mode:** Using WCF Project Server Interfaces, the SailPoint's Microsoft Project Server Connector manages the Microsoft Project Server 2013 users and groups, and global and category permissions assigned to users and groups for a Project Server web instance running with Project Server Mode.

## Supported features

---

SailPoint Microsoft Project Server Connector supports the following features:

- Account Management
  - Manage Microsoft Project Server Users as Account
  - Aggregate, Refresh Accounts
  - Create, Update, Delete
  - Enable, Disable
  - Add/Remove Entitlements

## Configuration parameters

- Account - Group Management
  - Manage Microsoft Project Server Groups as Account-Groups
  - Aggregate
  - Create, Update, Delete
- Permissions Management
  - Application reads permissions directly assigned to accounts and groups as direct permissions during account and group aggregation.
  - The connector supports automated revocation of the aggregated account permissions. Work items are created for requests to revoke group permissions.

## Supported Managed system

---

Microsoft Project Server Connector supports Microsoft Project Server 2013.

## Pre-requisites

---

Before you can use any of the features of the connector, the IQService must be installed and registered on Microsoft Project Server. For more information about installing IQService see, “Appendix E: IQService”.

## Administrator permissions

---

Application user must be the member of Administrators group of the Project Server web app to be managed.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Microsoft Project Server Connector uses the connection attributes listed in the following table:

Parameters	Description
Project Server Web App url	Project Server Web application URL.
Administrator	Login name of the administrator. For Windows Claim based authentication, the user name should be in encoding format. For example, i:0#.w contoso\chris
Password	Password for the administrator account.
IQService Host	Host Name of the computer on which IQService is installed.
IQService Port	IQService port number.
Page Size	Page size

# Schema attributes

---

This section describes the different schema attributes.

## Account attributes

---

The following table lists the account attributes ([Table 1—Account attributes](#)):

**Table 1—Account attributes**

Attributes	Description
AccountName	Windows account name.
DisplayName	The name for the user account.
UserID	The unique ID of the user.
Initials	The user's initials.
Email	The email address for the user.
Notes	Notes associated with user account.
TerminationDate	Termination date.
ManagerUID	Manager's unique ID.
PhoneNumber	The user's contact number.
PersonalHyperlink	The url of the user's website.
HyperlinkName	The name of the user's website (for example, a team website).
HireDate	Hire date.
ExchangeSync	Synchronize with Exchange Server.
AssignmentOwner	The unique ID of assignment owner.
CostCenter	The pre-use cost of the resource.
CheckoutDate	Check out date.
CheckoutBy	Account name of user who checkout this resource.
ResourceCanLevel	Indication whether the resource can be levelled.
Categories	It specifies all the categories to which the user belongs to.
Groups	It specifies all the groups to which the user belongs to.

## Group attributes

---

The following table lists the group attributes ([Table 2—Group attributes](#)):

**Table 2—Group attributes**

Attributes	Description
GroupID	The unique ID of the group.

## Provisioning Policy attributes

**Table 2—Group attributes**

Attributes	Description
GroupName	The name of the group.
Description	Description of the group.
ADGroup	The name of the Active Directory group from which the group gets its membership.

# Provisioning Policy attributes

---

This following table lists the provisioning policy attributes ([Table 3—Provisioning Policy attributes](#)):

**Table 3—Provisioning Policy attributes**

Attributes	Description
<b>Provisioning policy attributes for Account creation</b>	
AccountName	Login name of the user. The user should exist in the configured user store (for example, Active Directory). For Windows Claim based authentication, the user name should be in encoding format. For example, i:0#.w contoso\chris
DisplayName	User display name.
Email	User's email address.
<b>Provisioning policy attributes for Group creation</b>	
GroupName	Group name.
<b>Provisioning policy attributes for Group update</b>	
Description	Group description.
ADGroup	The name of the Active Directory group from which the group gets its membership.

# Troubleshooting

---

## 1 - Test Connection failed an error message

Could not load file or assembly 'PS2013, Version=0.0.0.0, Culture=neutral, PublicKeyToken=null' or one of its dependencies. This assembly is built by a runtime newer than the currently loaded runtime and cannot be loaded

**Resolution:** Rename App.config file located at IQService home to IQservice.exe.config and restart IQService.

## 2 - An error message appears mentioning that the HTTP request is unauthorized

The following error message appears when an invalid Username/Password is entered or due to a failure of connection with Project Server:

The HTTP request is unauthorized with client authentication scheme 'Ntlm'. The authentication header received from the server was 'NTLM'.

**Resolution:** Perform the following:

1. Ensure that Username and password are correct.
2. Ensure that Project Server web Application URL is accessible from the IQService host.
3. Restart IIS application pool service of Project Server web application.

### **3 - An error message appears mentioning that the HTTP service is unavailable**

The following error message appears when an invalid Project Server Web Application URL is specified, Project Web Application URL is not accessible from IQService host, or the application pool service of Project Server web application is not running:

The HTTP service located at Url/\_vti\_bin/PSI/ProjectServer.svc is unavailable. This could be because the service is too busy or because no endpoint was found listening at the specified address. Please ensure that the address is correct and try accessing the service again later.

**Resolution:** Perform the following:

1. Ensure that Project Server Web Application URL is accessible from the IQService host.
2. Start the application pool service.

## **Troubleshooting**

# Chapter 49: SailPoint NetSuite Connector

---

The following topics are discussed in this chapter:

Overview .....	435
Supported features .....	436
Supported Managed Systems .....	436
Administrator permissions .....	436
Configuration parameters .....	437
Schema attributes .....	437
Account attributes .....	437
Group attributes .....	438
Schema extension and custom attributes .....	438
Provisioning Policy attributes .....	439
Additional information .....	440
NetSuite Application Program Interface (API) .....	440

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

NetSuite is cloud-based Software-as-a-Service integrated business management software. NetSuite's cloud business management system includes ERP/accounting, order management/inventory, CRM, Professional Services Automation (PSA) and E-commerce.

Enterprise Resource Planning (ERP) in NetSuite encompasses several areas of your business, including accounting, inventory, order management, project management, and employee management.

For more information, see <http://www.netsuite.com/portal/products/main.shtml>

NetSuite Connector will manage the employee data in the NetSuite ERP system. The connector is a write-capable connector which manage the following entities:

- Employee Account
- Employee Role
- Employee Entitlement

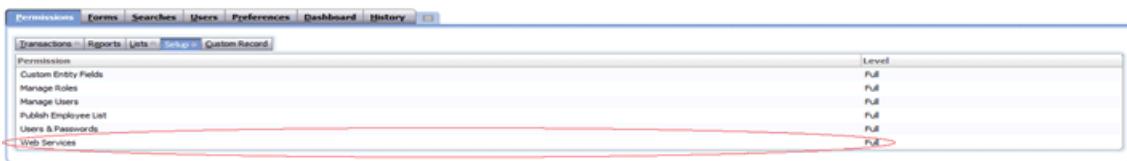
## Supported features

---

SailPoint NetSuite Connector supports the following features:

- Account Management
  - Manages NetSuite users as Accounts
  - Aggregation, Refresh Accounts, Pass Through Authentication
  - Create, Update, Delete
  - Enable, Disable, Change Password
  - Add/Remove Entitlements

**Note:** For Pass through Authentication, the account should have at least one role assigned with permissions required to perform the operation. Also this role needs to be Web Service enabled role as displayed in the following figure:



- Account - Group Management
  - Manages NetSuite groups as Account-Groups
  - Aggregation, Refresh Groups

## Supported Managed Systems

---

SailPoint NetSuite Connector supports the following managed system:

- NetSuite 2015.2
- Netsuite 2012\_1

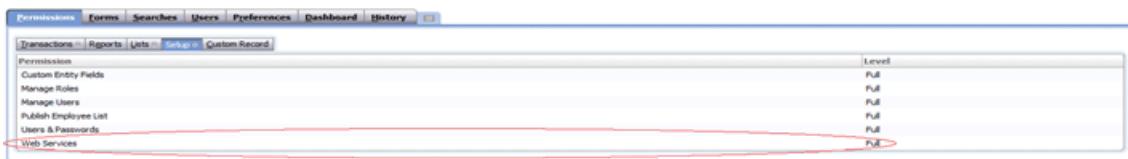
## Administrator permissions

---

The NetSuite Connector administrator must be able to perform the following operations on NetSuite employee data:

- Search
- Create
- Update
- Delete
- Access Custom Attributes

Hence a role is required which has the permissions to the above operations. We need to create a role in NetSuite.



## Configuration parameters

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The NetSuite connector uses the following connection parameters:

Parameters	Description
Account ID*	the account number assigned to an organization by NetSuite. This account number must be provided by each login request. This can be found by navigating to <b>Setup =&gt; Integration =&gt; Web Services Preferences</b> .
Role ID	When logging in using Web Services provide a role id along with your credentials. The role defined here must be a valid role contained in the Employee record of the given user. If no role id is provided, then the user's default role is used. If neither the request nor the Web Services default role is set, then the user's default UI role is used, provided it has the Web Services permission. For security reasons, it is recommended that you restrict permissions levels and access allowing only the most restricted permissions necessary to perform a given set of operations. For more information about the permissions, see "Administrator permissions" on page 436.
Administrator Email*	Email of the Account in Employee package having provisioning privileges.
Administrator Password*	Password of the employee Account.
Page Size	Limit to fetch number of accounts or groups per iteration through NetSuite Connector. If the value is not set then the default value is 50.

## Schema attributes

The following schema attributes are defined:

- Account schema
- Group schema
- Custom attributes

## Account attributes

The following table lists the account schema:

## Schema attributes

Attribute Name	Description
EmpID (Display Attribute)	Employee ID
InternalID (Identity Attribute)	Auto generated Internal ID of the employee
EmployeeStatus	The status of employee
Email	Email ID of employee
Initial	The initials of first name and last name
OfficePhoneNumber	Office phone number of employee
HomePhoneNumber	Home phone number of employee
MobilePhoneNumber	Mobile number of employee
Department	Department of employee
Class	Class of employee
BillingClass	Billing class of employee
Groups (Entitlements)	Groups associated to the employee
GlobalSubscriptionStatus	Subscription status of employee
SocialSecurityNumber	Security number of employee
Supervisor	Supervisor of employee
DateOfHiring	Date of hiring of employee
Type	Working type of employee
JobTitle	Job title of employee
DateOfBirth	Date of birth of employee
JobDescription	Description of job of employee
TimeApprover	Approver of time for the employee (some one like supervisor or manager)

## Group attributes

The following table lists the group schema:

Attribute Name	Description
GroupName (Display Attribute)	Name of the group
GroupInternalID (Identity Attribute)	Auto generated Internal id of the group

## Schema extension and custom attributes

NetSuite system allows the support for extending the schema through custom entity fields. Custom entity fields are fields that you can add to your entity records to gather information specific to your business needs. Entity custom fields can be added to existing and custom sub tabs on the entry forms you use to enter entity records in your NetSuite account.

NetSuite connector supports the read and write of custom attributes.

Following NetSuite Custom field type are supported in IdentityIQ

- Check Box
- Date
- Free-Form Text
- Email Address
- Phone Number
- HyperLink

### Supporting of custom attributes

Perform the following to support the custom attributes from IdentityIQ:

- Add the custom attribute name in the schema by clicking **Add attribute** button.
- Add the following lines in the application debug page:

```
<entry key = "customAttribute" >
<value>
<List>
<String>custom1</String>
<String>custom2</String>
</List>
</value>
</entry>
```

**Note:** No code change would be required while adding new custom attributes in schema. This is applicable only for custom attributes.

## Provisioning Policy attributes

---

The NetSuite connector is pre-configured with an account creation provisioning policy that includes the commonly-used attributes that need to be set when creating an account. This field list can be modified as required.

The attributes listed in the following table are required for creating an user.

Attribute Name	Description
EmpID (Entity ID)*	Employee name
*password*	Password for the employee
Email*	Email of the employee
OfficePhoneNumber	Office phone number for the employee
Fax	Fax for the employee

In the above table, **EmpID** is the minimum parameter which is required to create a user on NetSuite server. But in IdentityIQ a user can only be created after assigning a role to it.

In NetSuite when a role is assigned to a user, the user requires UserName, Email and password as mandatory parameter for accessing the NetSuite server.

## Additional information

**Note:** The field list can also be extended by adding custom attributes provided the attributes are defined in the application schema. For more information, see “Schema extension and custom attributes” on page 438.

# Additional information

---

This section describes the additional information related to the NetSuite Connector.

## NetSuite Application Program Interface (API)

---

SuiteTalk exposes NetSuite as a data source for programmatic access. The following operations supported in SuiteTalk would be used by NetSuite Connector

Operation/API	Summary
add	Use to add record into the system. The system returns a NetSuite identifier (internalId) that is unique for each record created within a record type.
changePasswordOrEmail	Use to change a user's email or password
get	Use to query the system for one record. You must provide either the internal or external ID and the record type for each query item.
getCustomizationId	Use to retrieve the internalIds, externalIds, and/or scriptIds of all custom objects of a specified type.
login	Use to login into NetSuite. This operation is similar to the NetSuite UI and requires you to provide a valid username, password, role, and account number.
logout	Use to logout from the system. The logout operation invalidates the current session.
search	Use to search for a set of records based on specific search criteria. This operation supports pagination, so that large result sets can be retrieved in smaller sets.
searchMore	Used to retrieve more records after an initial search operation is invoked.
update	Use to update existing record in the system by providing new values for the fields to be updated for each record. The records to be updated are identified by either the internal or external ID and the record type.
delete	Use to delete an existing record in the system by providing the internal id and record type.

**Note:** For more information, see *NetSuite SuiteTalk (Web Services) Platform Guide*.

# Chapter 50: SailPoint Oracle HRMS Connector

---

The following topics are discussed in this chapter:

Overview .....	441
Supported features .....	441
Supported Managed Systems .....	441
Pre-requisites .....	441
Administrator permissions .....	442
Configuration parameters .....	444
Schema attributes .....	444
Account attributes .....	444

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

The Oracle HRMS connector aggregates the HR Person details from Oracle HRMS system. Details would include the Personal and organization information.

## Supported features

---

SailPoint Oracle HRMS Connector supports the following features:

- Account Management
  - Manages Oracle HRMS Users as Accounts
  - Aggregation, Refresh Accounts
  - Update Email address and Work-Telephone

## Supported Managed Systems

---

Following versions of Oracle HRMS are supported by the connector:

- Oracle E-Business Suite 12.2
- Oracle E-Business Suite 12.1

## Pre-requisites

---

The compatible JDBC drivers must be used in the classpath of IdentityIQ for connecting to Oracle Server. For example, ojdbc6.jar.

## Administrator permissions

---

### Rights present on HR\_PERSON\_API package

Enter the following command to find the rights present on the **HR\_PERSON\_API**:

```
SELECT dbo.object_name,
       (DECODE(SIGN(bitand(options,16)),1,'INVOKER','DEFINER')) "authid"
  FROM dba_objects dbo, sys.PROCEDURE$ p
 WHERE p.obj# = dbo.object_id
   AND dbo.object_type = 'PACKAGE'
   AND dbo.object_name = 'HR_PERSON_API'
   AND dbo.owner = 'APPS'
```

Following list of minimum permissions can help to resolve the issue:

- DEFINER Rights
- INVOKER Rights

### DEFINER Rights

If you do not want to use the APPS account as the administrator, use the following procedure to create and set up an Oracle account for the administrator when the **DEFINER** rights are assigned to **HR\_PERSON\_API**.

1. Log in to the Oracle database as database administrator for creating the new administrator user account using SQL\*Plus as follows:

```
Create user ${new user} identified by ${password};
Grant create session to ${new user};
Grant create synonym to ${new user};
```

2. Grant permissions to the new user created in the above step (\${new user}):

```
GRANT SELECT ON HR_ALL_POSITIONS_F_TL TO ${new user};
GRANT SELECT ON HR_ALL_ORGANIZATION_UNITS_TL TO ${new user};
GRANT SELECT ON PER_ALL_PEOPLE_F TO ${new user};
GRANT SELECT ON PER_ALL_ASSIGNMENTS_F TO ${new user};
GRANT SELECT ON PER_ASSIGNMENT_STATUS_TYPES_TL TO ${new user};
GRANT SELECT ON PER_PERSON_TYPES_TL TO ${new user};
GRANT SELECT ON PER_JOB_GROUPS TO ${new user};
GRANT SELECT ON PER_JOBS TO ${new user};
GRANT SELECT ON PER_PHONES TO ${new user};
GRANT EXECUTE ON APPS.HR_PERSON_API TO ${new user};
GRANT EXECUTE ON APPS.HR_PHONE_API TO ${new user};
```

3. Login by new user name (\${new user}) and create the following synonyms:

```
CREATE OR REPLACE SYNONYM HR_ALL_ORGANIZATION_UNITS_TL FOR
HR.HR_ALL_ORGANIZATION_UNITS_TL;
CREATE OR REPLACE SYNONYM HR_ALL_POSITIONS_F_TL FOR HR.HR_ALL_POSITIONS_F_TL;
CREATE OR REPLACE SYNONYM PER_ALL_ASSIGNMENTS_F FOR HR.PER_ALL_ASSIGNMENTS_F;
CREATE OR REPLACE SYNONYM PER_ALL_PEOPLE_F FOR HR.PER_ALL_PEOPLE_F;
```

```

CREATE OR REPLACE SYNONYM PER_ASSIGNMENT_STATUS_TYPES_TL FOR
HR.PER_ASSIGNMENT_STATUS_TYPES_TL;
CREATE OR REPLACE SYNONYM PER_JOBS FOR HR.PER_JOBS;
CREATE OR REPLACE SYNONYM PER_JOB_GROUPS FOR HR.PER_JOB_GROUPS;
CREATE OR REPLACE SYNONYM PER_PERSON_TYPES_TL FOR HR.PER_PERSON_TYPES_TL;
CREATE OR REPLACE SYNONYM PER_PHONES FOR HR.PER_PHONES;

```

## INVOKER Rights

If you do not want to use the APPS account as the administrator, use the following procedure to create and set up an Oracle account for the administrator when the INVOKER rights are assigned to HR\_PERSON\_API package.

1. Log in to the Oracle database as database administrator for creating the new administrator user account using SQL\*Plus as follows:  

```

Create user ${new user} identified by ${password};
Grant create session to ${new user};
Grant create synonym to ${new user};

```
2. Copy the package scripts from *iiqBase\integration\OracleHRMS* directory to the *OracleHome\bin* directory and rename the type of scripts from \*.txt to \*.sql
3. Using SQL\*Plus, log in to the Oracle database as APPS and run the following:
  - Run the @IIQ\_UPDATE\_EMAIL\_API script using SQL\*Plus
  - Run the @IIQ\_UPDATE\_EMAIL\_API\_BODY script using SQL\*Plus
4. Login by administrator user and grant the following permissions:  

```

GRANT SELECT ON HR_ALL_POSITIONS_F_TL TO ${new user};
GRANT SELECT ON HR_ALL_ORGANIZATION_UNITS_TL TO ${new user};
GRANT SELECT ON PER_ALL_PEOPLE_F TO ${new user};
GRANT SELECT ON PER_ALL_ASSIGNMENTS_F TO ${new user};
GRANT SELECT ON PER_ASSIGNMENT_STATUS_TYPES_TL TO ${new user};
GRANT SELECT ON PER_PERSON_TYPES_TL TO ${new user};
GRANT SELECT ON PER_JOB_GROUPS TO ${new user};
GRANT SELECT ON PER_JOBS TO ${new user};
GRANT SELECT ON PER_PHONES TO ${new user};
GRANT EXECUTE ON APPS.IIQ_UPDATE_EMAIL_API TO ${new user};

```
5. Login by new user name (\${new user}) and create the following synonyms:  

```

CREATE OR REPLACE SYNONYM HR_ALL_ORGANIZATION_UNITS_TL FOR
HR.HR_ALL_ORGANIZATION_UNITS_TL;
CREATE OR REPLACE SYNONYM HR_ALL_POSITIONS_F_TL FOR HR.HR_ALL_POSITIONS_F_TL;
CREATE OR REPLACE SYNONYM PER_ALL_ASSIGNMENTS_F FOR HR.PER_ALL_ASSIGNMENTS_F;
CREATE OR REPLACE SYNONYM PER_ALL_PEOPLE_F FOR HR.PER_ALL_PEOPLE_F;
CREATE OR REPLACE SYNONYM PER_ASSIGNMENT_STATUS_TYPES_TL FOR
HR.PER_ASSIGNMENT_STATUS_TYPES_TL;
CREATE OR REPLACE SYNONYM PER_JOBS FOR HR.PER_JOBS;
CREATE OR REPLACE SYNONYM PER_JOB_GROUPS FOR HR.PER_JOB_GROUPS;
CREATE OR REPLACE SYNONYM PER_PERSON_TYPES_TL FOR HR.PER_PERSON_TYPES_TL;
CREATE OR REPLACE SYNONYM PER_PHONES FOR HR.PER_PHONES;
CREATE OR REPLACE SYNONYM HR_PERSON_API for APPS.IIQ_UPDATE_EMAIL_API;

```

# Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection. The following table lists the configuration parameters of Oracle HRMS Connector:

**Note:** Attributes marked with \* sign are the mandatory attributes.

Attributes	Type
URL	URL for server which directly interacts with the Managed system. For example, <code>jdbc:oracle:thin:@xxx.xx.xx.xxx:xxxx:VIS</code>
User	Oracle user minimum permissions mentioned in “Administrator permissions” section. For example, “apps”
Password	Authentication details of login.
Driver	Name of the Driver class supported by JDBC Type 4. For example, “oracle.jdbc.driver.OracleDriver”

**Note:** The Oracle HRMS Connector by default aggregates only organizational data with active persons and assignments. If there is a requirement to aggregate personal data, navigate to debug page to access the application created for Oracle HRMS Connector and modify the following entry by setting the value to false:

```
<entry key="aggregateOnlyOrganisationData" value="true" />
```

# Schema attributes

---

This section describes the different schema attributes.

## Account attributes

---

The following table lists the account attributes:

Attributes	Description
FULL_NAME	Full name of the employee.
GENDER	Gender of an employee. For example, M-male, F- female
EMAIL_ADDRESS	Email ID of the person. For updating email address the default mode is “UPDATE” and the effective date is considered as “Sysdate”.
MARITAL_STATUS	Marital status - M: Married - S: Single - D: Divorced

Attributes	Description
PERSON_ID	Unique ID from which details of person can be fetched.
SUPERVISOR_ID	Person ID of Supervisor.
BUSINESS_GROUP	Business group from which employee is.
DATE_OF_BIRTH	Date of Birth of Employee.
PERSON_TYPE	Person Type will give the current status. For example, Employee, applicant
SUPERVISOR_NAME	Name of supervisor/mentor/manager of an employee.
EMPLOYEE_NUMBER	If person is an employee then employee number will be displayed at this attribute.
JOB	Job details of an employee.
START_DATE	Start date of the person.
END_DATE	End date of the person.
ORGANIZATION	Organization name in which employee is working.
POSITION	Current position or job title of an employee.
WORK_TELEPHONE	<p>Work telephone of an employee.</p> <p>For updating phone number of type <b>Work</b>, the default mode is "Correction".</p> <p><b>Note:</b> When upgrading from any older version to version 7.1, ensure that you add WORK_TELEPHONE attribute manually.</p>

## **Schema attributes**

# Chapter 51: SailPoint Oracle E-Business Suite Connector

---

The following topics are discussed in this chapter:

Overview .....	447
Supported features .....	448
Supported Managed Systems .....	448
Pre-requisites .....	448
Administrator permissions .....	448
Configuration parameters .....	451
Additional configuration parameter .....	451
Schema attributes .....	452
Account attributes .....	452
Group attributes .....	453
Provisioning Policy attributes .....	454
Create account attributes .....	454
Delete account attributes .....	454
Create group attributes .....	454
Deleting Group (Responsibility) .....	455
Troubleshooting .....	455

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

The Oracle E-Business Suite is an integrated suite of development, runtime, and system management tools. It also includes Forms, JDeveloper, Single Sign-On, Oracle Internet Directory, Portal, Discoverer, Web Cache, Integration, Oracle BPEL Process Manager.

SailPoint Oracle E-Business Suite Connector controls the activities related to account/groups by signing in managed system. SailPoint Oracle E-Business Suite Connector will manage the following entities of Oracle E-Business Suite:

- User
- Group (Responsibility, Role)

## Supported features

---

SailPoint Oracle E-Business Suite Connector supports the following features:

- Account Management

- Manages Oracle E-Business Suite users
- Aggregation, Refresh Accounts, Discover Schema
- Create, Update
- Enable, Disable, Change Password
- Add/Remove Entitlements

- Account - Group Management

Supports multiple group functionality.

- Manages Oracle E-Business Suite groups as RESPONSIBILITY
  - Aggregation, Refresh Groups

**Note:** The following versions represents the respective responsibilities:

- 4: Oracle Applications
- W: Oracle Self-Service Web Applications
- M: Oracle Mobile Applications

Oracle E-Business Connector aggregates responsibilities of only type '4' and 'W'. Hence Account-Group aggregation fetches only responsibilities of type 'Oracle Applications' and 'Self-Service Web Applications'.

- Create, Update, Delete
- Manages Oracle E-Business Suite groups as ROLE
  - Aggregation, Refresh Groups

## Supported Managed Systems

---

Following versions of Oracle E-Business Suite are supported by the connector:

- Oracle E-Business Suite 12.2.x
- Oracle E-Business Suite 12.1.x

## Pre-requisites

---

- The compatible JDBC drivers must be used in the classpath of IdentityIQ for connecting to Oracle Server.  
For example, ojdbc6.jar.
- During Account Aggregation task, **Detect Deleted Accounts** must always be unchecked.

## Administrator permissions

---

**Note:** For Invoker rights, if the packages are already invoked, then no need to invoke the packages after upgrading IdentityIQ from any previous version to IdentityIQ version 7.1.

1. Rights present on Oracle packages:

Enter the following command to find the rights present on the Oracle packages:

```

SELECT dbo.object_name,
(DECODE(SIGN(bitand(options,16)),1,'INVOKER','DEFINER')) "authid"
FROM dba_objects dbo, sys.PROCEDURE$ p
WHERE p.obj# = dbo.object_id
AND dbo.object_type = 'PACKAGE'
AND dbo.object_name = 'xxx'
AND dbo.owner = 'APPS';
Where xxx package is FND_USER_PKG, FND_RESPONSIBILITY_PKG, WF_LOCAL_SYNCH, FND_WEB_SEC, or
FND_GLOBAL.

```

**Sample example:**

Enter the following command to find the rights present on the FND\_USER\_PKG:

```

SELECT dbo.object_name,
(DECODE(SIGN(bitand(options,16)),1,'INVOKER','DEFINER')) "authid"
FROM dba_objects dbo, sys.PROCEDURE$ p
WHERE p.obj# = dbo.object_id
AND dbo.object_type = 'PACKAGE'
AND dbo.object_name = 'FND_USER_PKG'
AND dbo.owner = 'APPS';

```

2. If **xxx** package has Invoker rights, perform the following:

Copy the package scripts from

`identityiq\integration\OracleEBS\iiqIntegration-OracleEBS.zip` directory to the  
OracleHome\bin directory and rename the type of scripts from \*.txt to \*.sql

Using SQL\*Plus, log in to the Oracle database as APPS and run the following:

Run the @SP\_xxx package script using SQL\*Plus

**Sample example:** If **FND\_USER\_PKG** has invoker rights, run the `@SP_FND_USER_PKG` script using SQL\*Plus

Perform this step for all **xxx** packages.

3. Log in to the Oracle database as database administrator for creating the new administrator user account using SQL\*Plus as follows:

```

create user ${new user} identified by ${password};
grant create session to ${new user};
grant create synonym to ${new user};

```

**Grant permissions to the new user created in the above step (\${new user}):**

```

grant select on APPS.FND_PRODUCT_GROUPS to ${new user};
grant select on APPS.FND_USER to ${new user};
grant select on SYS.DBA_USERS to ${new user};
grant select on APPS.FND_RESPONSIBILITY_VL to ${new user};
grant select on APPS.FND_APPLICATION_VL to ${new user};
grant select on APPS.FND_DATA_GROUPS to ${new user};
grant select on APPS.FND_USER_RESP_GROUPS_ALL to ${new user};
grant select on DUAL to ${new user};
grant select on APPS.PER_ALL_PEOPLE_F to ${new user};
grant select on APPS.RA_CUSTOMERS to ${new user};
grant select on APPS.FND_MENUS to ${new user};
grant select on APPS.FND_REQUEST_GROUPS to ${new user};
grant select on APPS.FND_APPLICATION to ${new user};
grant select on APPS.FND_DATA_GROUP_UNITS to ${new user};
grant select on APPS.FND_APPLICATION_TL to ${new user};

```

## Overview

```
grant select on APPS.FND_RESPONSIBILITY to ${new user};  
grant select on APPS.WF_ROLES to ${new user};  
grant select on APPS.WF_USER_ROLES to ${new user};  
grant select on APPS.WF_LOCAL_ROLES to ${new user};  
grant select on APPS.WF_ALL_ROLES_VL to ${new user};  
grant select on APPS.WF_ROLE_HIERARCHIES to ${new user};  
grant select on APPS.FND_REQUEST_GROUP_UNITS to ${new user};  


- If xxx package has Definer rights, perform the following:  
  
grant execute on APPS.xxx to ${new user};  
For example, grant execute on APPS.FND_USER_PKG to ${new user};
- If xxx package has Invoker rights, perform the following:  
  
grant execute on APPS.SP_XXX to ${new user};  
For example, grant execute on APPS.SP_FND_USER_PKG to ${new user};  
Where xxx package is FND_USER_PKG, FND_RESPONSIBILITY_PKG, WF_LOCAL_SYNCH, FND_WEB_SEC, or FND_GLOBAL.



4. Login by the new user name ${new user} and create the following synonym:



```
create synonym FND_PRODUCT_GROUPS for APPS.FND_PRODUCT_GROUPS;  
create synonym FND_USER for APPS.FND_USER;  
create synonym DBA_USERS for SYS.DBA_USERS;  
create synonym FND_RESPONSIBILITY_VL for APPS.FND_RESPONSIBILITY_VL;  
create synonym FND_APPLICATION_VL for APPS.FND_APPLICATION_VL;  
create synonym FND_DATA_GROUPS for APPS.FND_DATA_GROUPS;  
create synonym FND_USER_RESP_GROUPS_ALL for APPS.FND_USER_RESP_GROUPS_ALL;  
create synonym PER_ALL_PEOPLE_F for APPS.PER_ALL_PEOPLE_F;  
create synonym RA_CUSTOMERS for APPS.RA_CUSTOMERS;  
create synonym FND_MENUS for APPS.FND_MENUS;  
create synonym FND_REQUEST_GROUPS for APPS.FND_REQUEST_GROUPS;  
create synonym FND_APPLICATION for APPS.FND_APPLICATION;  
create synonym FND_RESPONSIBILITY for APPS.FND_RESPONSIBILITY;  
create synonym FND_APPLICATION_TL for APPS.FND_APPLICATION_TL;  
create or replace synonym FND_DATA_GROUP_UNITS for APPS.FND_DATA_GROUP_UNITS;  
create or replace synonym WF_USER_ROLES for APPS.WF_USER_ROLES;  
create or replace synonym WF_ROLES for APPS.WF_ROLES;  
create or replace synonym WF_LOCAL_ROLES for APPS.WF_LOCAL_ROLES;  
create or replace synonym WF_ROLE_HIERARCHIES for APPS.WF_ROLE_HIERARCHIES;  
create or replace synonym WF_ALL_ROLES_VL for APPS.WF_ALL_ROLES_VL;  
create synonym FND_REQUEST_GROUP_UNITS for APPS.FND_REQUEST_GROUP_UNITS;
```



- If xxx package has Definer rights, perform the following:  
  
create or replace synonym xxx for APPS.XXX;  
For example, create or replace synonym FND_USER_PKG for APPS.FND_USER_PKG;
- If xxx package has Invoker rights, perform the following:  
  
create or replace synonym xxx for APPS.SP_XXX;  
For example, create or replace synonym FND_USER_PKG for APPS.SP_FND_USER_PKG;  
Where xxx package is FND_USER_PKG, FND_RESPONSIBILITY_PKG, WF_LOCAL_SYNCH, FND_WEB_SEC, or FND_GLOBAL.

```

**Note:** If table ar\_customers exist instead of ra\_customer then provide the select permissions as follows:

```
grant select on APPS.AR_CUSTOMERS to ${new user};
```

**Also the synonym must be as follows:**

```
create synonym RA_CUSTOMERS for APPS.AR_CUSTOMERS;
```

## Configuration parameters

---

The following table lists the configuration parameters of Oracle E-Business Suite Connector:

**Note:** Attributes marked with \* sign are the mandatory attributes.

Attributes	Type
Connection User*	The Oracle EBS Login name through which we want to connect Oracle EBS. For example, <b>APPS</b>
Password*	The authentication details of login.
Database URL*	<p>The url to connect to the database. The format is <code>jdbc:oracle:thin:@&lt;HOST&gt;:&lt;PORT&gt;:&lt;SID&gt;</code></p> <p>For example <code>jdbc:oracle:thin:@xxx.xx.xx.xx:xxxx:ORCL</code> url consist of</p> <ul style="list-style-type: none"> <li>• <b>jdbc:oracle:thin:@</b>: This is common part which states that the connection is made using thin driver.</li> <li>• <b>xxx.xx.xx.xx</b>: server Name or IP of the oracle server</li> <li>• <b>1521</b>: The port number of the oracle server. This port number should be known by the oracle server administrator.</li> <li>• <b>ORCL</b>: The SID of the oracle server.</li> </ul>
JDBC Driver*	It is the name of the Driver class supported by JDBC Type 4. For example, <code>oracle.jdbc.driver.OracleDriver</code>

## Additional configuration parameter

---

Parameter	Description
aggregateActiveAccounts	<p>For aggregating disabled accounts, set the value of the aggregateActiveAccounts parameter to false in the application debug page as follows:</p> <pre>&lt;entry key="aggregateActiveAccounts"&gt;   &lt;value&gt;     &lt;Boolean&gt;false&lt;/Boolean&gt;   &lt;/value&gt; &lt;/entry&gt;</pre>

## Schema attributes

---

This section describes the different schema attributes.

### Account attributes

---

The following table lists the account attributes:

Attributes	Description
USER_NAME	Application username (what a user types in at the Oracle Applications sign-on screen).
USER_ID	Application user identifier.
START_DATE	The date the user name becomes active.
END_DATE	The date the user name becomes inactive.
DESCRIPTION	Description.
PASSWORD_DATE	The date the current password was set.
PASSWORD_EXPR	The number of accesses left for the password.
PASSWORD_NO_OF_DAYS	The number of accesses allowed for the password.
EMAIL_ADDRESS	The electronic mail address for the user.
FAX	The fax number for the user.
EMPLOYEE_ID	Identifier of employee to whom the application username is assigned.
EMPLOYEE_NUMBER	Unique number of the employee.
FULL_NAME	Full name of the user.
CUSTOMER_ID	Customer contact identifier. If the AOL user is a customer contact, this value is a foreign key to the corresponding customer contact.
CUSTOMER_NAME	Customer name.
RESPONSIBILITIES	Responsibilities assigned to a user.
ROLES	Roles assigned to a user.

### Custom attributes

Perform the following to support the custom attributes in Oracle E-Business Suite Connector:

- Add the custom attribute name in the account schema by clicking Add attribute button.
- Add the following lines in the application debug page:

```
<entry key = "customAttribute" >
  <value>
    <List>
      <String>custom1</String>
      <String>custom2</String>
    </List>
```

```
</value>
</entry>
```

## Group attributes

---

This section describes the different group attributes.

### Responsibility GroupObjectType attributes

The following table lists the Responsibility GroupObjectType attributes:

Attributes	Description
RESPONSIBILITY_ID	Responsibility identifier.
RESPONSIBILITY_NAME	Name of the responsibility.
RESPONSIBILITY_KEY	Internal developer name for responsibility.
START_DATE	The date the responsibility becomes active.
END_DATE	The date the responsibility expires.
DESCRIPTION	Description
STATUS	Shows status of the responsibility.
VERSION	Version
WEB_HOST_NAME	IP address or alias of the computer where the Webserver is running. Defaults to the last agent.
WEB_AGENT_NAME	Name of Oracle Web Agent. Defaults to the last agent.
DATA_GROUP_APPL_NAME	Name of the data group application.
REQUEST_GROUP_APPL_NAME	Request Group Application name.
DATA_GROUP_ID	Identifier of data group.
DATA_GROUP_NAME	Name of the Data Group.
MENU_NAME	Name of the menu.
REQUEST_GROUP_NAME	Request group name.

### Role GroupObjectType attributes

The following table lists the Role GroupObjectType attributes:

Attributes	Description
NAME	An internal name for the role.
DISPLAY_NAME	The display name of the role.
DESCRIPTION	Description
START_DATE	The date at which the role becomes valid.
EXPIRATION_DATE	The date at which the role is no longer valid in the directory service.

## Provisioning Policy attributes

Attributes	Description
APPLICATION_NAME	Application that owns the information for the role.
STATUS	The availability of the Role to participate in a workflow process.
SUBORDINATE_ROLES	Subordinate roles for a role.
SUBORDINATE_RESPONSIBILITIES	Subordinate responsibilities for a role.

# Provisioning Policy attributes

---

This section lists the single provisioning policy attributes of Oracle E-Business Suite Connector that allows you to select the type of user/group to create.

**Note:** Attributes marked with \* sign are the mandatory attributes.

## Create account attributes

---

The following table lists the provisioning policy attributes for Create Accounts:

Attributes	Description
Name*	Name of the login user.
Password*	Password of the login user.
Description	Description.
Start Date*	The date from which login user becomes active.
End Date	The date from which login user becomes inactive.
Password Expiration Type	Type of the password to expire.
Number of Days	Days after which user password will expire.
Permanent Mode	In permanent mode change of password on first login is not required.

## Delete account attributes

---

In delete operation, Oracle E-Business Suite sets the value of **END DATE** parameter of user to current system date, this operation is nothing but the **disable** operation.

When the delete operation is performed as a result Oracle E-Business Suite connector would display an error. In such case administrator should use the disable user operation.

## Create group attributes

---

The following table lists the provisioning policy attributes for Create Group (Responsibility):

Attributes	Description
Responsibility Name*	Name of the responsibility.

Attributes	Description
Application Name*	Name of the application.
Description	Description
Responsibility Key*	Internal developer name for responsibility.
Start Date*	The date the responsibility becomes active.
End Date	The date the responsibility expires.
Responsibility Version*	Responsibility version.
Data Group Name*	Name of the data group.
Data Group Application Name*	Name of the data group application.
Menu Name*	Name of the menu.
Request Group Name	Request group name.
Request Group Application Name	Request group application name.

## Deleting Group (Responsibility)

---

In delete operation, Oracle E-Business Suite sets the value of END DATE parameter of group to current system date. This operation is the **disable** operation.

## Troubleshooting

---

### 1 - RA\_Customers table not found when managing Oracle version 12c

If Customer is using Oracle E-Business Suite Connector to manage Oracle version 12c, the Connector installation expects a table named **RA\_Customers**. This table is renamed as **AR\_Customers** in Oracle version 12c.

**Resolution:** Assign the following synonym to the new user,

```
create synonym ra_customers for apps.ar_customers;
```

### 2 - User must be re-hired who is disabled in native system

When a user must be re-hired who is disabled in native system, the following error message appears on native system:

```
User already exists
```

**Resolution:** To re-hire user who is disabled in native system, refresh the accounts from IdentityIQ using manage accounts. This corrects the status of the user in IdentityIQ and can be enabled manually from IdentityIQ or native system.

### 3 - A new user with an existing user name on native system is in disabled state must be newly hired

When a new user must be hired with an old user name on native system is in disabled state, the following error message appears on the native system:

## Troubleshooting

User already exists

**Resolution:** IdentityIQ does not have the old user details which is disabled on the native system. The create user request would fail in IdentityIQ with the above error message. Therefore a new name must be entered for the new user.

# Chapter 52: SailPoint PeopleSoft HCM Database Connector

---

The following topics are discussed in this chapter:

Overview .....	457
Supported features .....	457
Supported Managed Systems .....	458
Pre-requisites .....	458
Administrator permission .....	458
Configuration parameters.....	459
Schema attributes .....	460
Account attributes .....	460
Additional information .....	462
Configuring Component Interface Security .....	462
Creating PeopleSoft HRMS Jar File .....	462
Troubleshooting.....	464

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

The SailPoint PeopleSoft HCM Database Connector aggregates and provisions the Personal and Job related data of Person records from PeopleSoft HCM Database system. The connector makes use of database connection for all aggregation related operations. The provisioning operations are handled using the PeopleSoft Component Interfaces.

## Supported features

---

SailPoint PeopleSoft HCM Database Connector supports the following features:

- Account Management
  - Manages PeopleSoft HRMS Person as Account
  - Aggregation, Refresh Accounts
  - Provisioning

IdentityIQ supports the following additional PeopleSoft HCM Database Connector provisioning features in version 7.0 and above:

- Ability to define Global Provisioning Rule
- Ability to define separate provisioning rule for specific operation (operations that include are Enable, Disable, Unlock, Delete, Create, and Modify).

For more information, see “Customization Rule” on page 462.

## Supported Managed Systems

---

SailPoint PeopleSoft HCM Database Connector supports the following PeopleSoft version running on Oracle database:

- Supported PeopleSoft versions
  - PeopleSoft HCM versions 9.2 and 9.1
  - PeopleTools version 8.55, 8.54 and 8.53

## Pre-requisites

---

- Using the PeopleSoft Application Designer verify if the following Component interfaces related to person's personal data and job data are present:
  - CI\_PERSONAL\_DATA
  - CI\_JOB\_DATA
- The following jar files must be present on the configured IdentityIQ Application Server:
  - psjoa.jar (found on PeopleSoft server at %PS\_HOME%\classes where %PS\_HOME% is the location of PeopleSoft installation server directory, referred to as PS\_HOME)
  - PeopleSoftHRMS.jar (For more information, see “Creating PeopleSoft HRMS Jar File” on page 462)

**Note:** The PeopleSoft jar files can be located in WEB-INF\lib directory.
- Visit the Oracle website and download an appropriate Oracle JDBC driver and compatible JDK version for IdentityIQ.

## Administrator permission

---

This section describes the Database and Component Interface related permissions.

### *Database related permissions*

The PeopleSoft Application Server will use the database user context to support the aggregation operations. The database user mentioned in the application configuration should have appropriate rights to fetch data related to the entities and attributes mentioned in the SQL query related to Person.

### Creating Administrator Account in Oracle database for PeopleSoft HCM

1. Log in to the Oracle database as database administrator for creating the new administrator user account using SQL\*Plus as follows:

```
create user ${new user} identified by password;
grant create session to ${new user};
grant create synonym to ${new user};
```

#### **Grant permissions to the new user created from the above step (\${new user}):**

```
grant select on ${Administrator}.PS_PERSONAL_DATA to ${new user}
grant select on ${Administrator}.PS_PERSONAL_PHONE to ${new user}
grant select on ${Administrator}.PS_EMAIL_ADDRESSES to ${new user}
grant select on ${Administrator}.PS_JOB to ${new user}
```

2. Login by the new user name (\${new user}) and create the following synonym:

```
create synonym PS_PERSONAL_DATA for ${Administrator}.PS_PERSONAL_DATA;
create synonym PS_PERSONAL_PHONE for ${Administrator}.PS_PERSONAL_PHONE;
create synonym PS_EMAIL_ADDRESSES for ${Administrator}.PS_EMAIL_ADDRESSES;
create synonym PS_JOB for Admin.PS_JOB;
```

#### *Component interface related permissions*

For provisioning operations, the PeopleSoft user who acts as an administrator must have proper access to the HRMS related Component Interfaces. For more information, see “Configuring Component Interface Security” on page 462.

## Configuration parameters

---

This section contains the information that IdentityIQ uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

**Note: Attributes marked with \* sign are the mandatory attributes.**

The PeopleSoft HCM Database connector uses the following connection attributes:

Attribute	Description
<b>JDBC Connection Settings</b>	
URL*	The URL with which to connect to the database.
User*	The user with which to connect to the host of the database.
Password*	The password associated with the specified user.
Driver class*	The Java JDBC class to use for the connection.
SQL Query*	<p>Is used to provide a SQL statement which provides a criteria to fetch record of person from PeopleSoft HRMS underlying tables. This query would be used in aggregation task.</p> <p>Default: Query is provided to fetch personal record which also includes email and phone from PeopleSoft HRMS system.</p>
getObjectSQL	The getObjectSQL query will be similar to SQL query, but will fetch details of only one Person record at a time. It is recommended that this query must be updated when the SQL Query is updated.
Person Sub Query	Used to provide a SQL statement which will give a criteria to fetch record of Job from PeopleSoft HRMS underlying tables.
BuildMap Rule	The rule called for each row returned by the database after the SQL has been executed. The rule uses ResultSet and builds a Map out of it to be consumed by IdentityIQ.
<b>PeopleSoft Connection Settings</b>	
Host	Peoplesoft Server host.

## Schema attributes

Attribute	Description
Port	JOLT Port.
User	Administrator User of PeopleSoft Server.
Password	Password for the administrator user.
Domain Connection Password Enabled	Determines if Domain connection Password is configured.
Domain Connection Password*	Password is required if <b>Domain Connection Password Enabled</b> attribute is selected.
Jar location	If there are more than one PeopleSoft application of different PeopleTools versions running under the same instance of JVM, the location specified would be added in the classpath. (The <code>psjoa.jar</code> and <code>PeopleSoftHRMS.jar</code> files). For more information, see Pre-requisites).  <b>Note:</b> For single PeopleSoft application, the peoplesoft jars can be located in <code>WEB-INF\lib</code> directory.

## Schema attributes

This section describes the different schema attributes.

### Account attributes

The following table lists the account attributes:

Attributes	Description
EMPLID	ID of the employee.
COUNTRY	Country of the employee.
CITY	City of the employee.
STATE	State of the employee.
PER_ORG	Organization of the employee.
ADDRESS1	Postal address1 of the employee.
ADDRESS2	Postal address2 of the employee.
ADDRESS3	Postal address3 of the employee.
BIRTHDATE	Birth date of the employee.
FIRST_NAME	First name of the employee.
HR_RESPONSIBLE_ID	HR Responsible ID of the employee.
LAST_NAME	Last name of the employee.
MIDDLE_NAME	Middle name of the employee.
NAME	Name of the employee.

Attributes	Description
NAME_PREFIX	Prefix of the employee.
NAME_SUFFIX	Suffix of the employee.
NAME_TITLE	Title of the employee.
POSTAL	Postal pin code of the employee.
PREF_FIRST_NAME	Preferred first name of the employee.
EMPL_RCD	Job employee record.
EFFDT	Effective date of the job.
DEPTID	Department ID of the job.
JOBCODE	Job code of the job.
POSITION_NBR	Position number of the job.
SUPERVISOR_ID	Supervisor ID of the job.
HR_STATUS	HR status of the job.
EMPL_STATUS	Employee status of the employee.
ACTION	Action code of the job.
ACTION_REASON	Action reason of the job.
LOCATION	Location of the job.
FULL_PART_TIME	Full part time of the job.
COMPANY	Company of the employee for current job.
EMPL_TYPE	Employee type of the job.
OFFICER_CD	Officer code of the job.
EMPL_CLASS	Employee class of the job.
ACCT_CD	Account code of the job.
BUSINESS_UNIT	Business unit of the job.
REPORTS_TO	Reporting to of the employee.
HIRE_DT	Hiring date of the employee.
TERMINATION_DT	Termination date of the employee.
EMAIL_ADDR	Email address of the employee.
PHONE	Personal phone of the employee.

## Additional information

### Customization Rule

- **Modify Rule:** The rule name is defined as **Example PeopleSoft HRMS Modify Rule**. This is a sample rule to update Collection Attribute (E-mail, Phone and samAccountName) and Non Collection Attribute (BirthPlace).
- **buildMap Rule:** The rule name is **Example PeopleSoft HRMS BuildMap Rule**. This rule is required to fetch additional attributes which are not defined in the account schema.

## Additional information

---

This section describes the additional information related to the PeopleSoft HCM Database Connector.

### Configuring Component Interface Security

---

Before using the connector, allow the PeopleSoft user that the connector is configured with to access the generated component interfaces.

Perform the following to set security for the PeopleTools project:

1. Log into the PeopleSoft web interface.
2. Navigate to **PeopleTools => Security => Permissions & Roles => Permission Lists**.
3. Click **Add a New Value** to create a new permission list. Enter **IIQ\_HRMS\_PERM** as the name of the permission list, and click **Add**.
4. Click the **Component Interfaces** tab, which will be used in provision rule. For example, **CI\_PERSONAL\_DATA**.
5. For each Component Interface used in provisioning rule provide an appropriate access based on operation performed.
6. Click **Save** to save the new permission list.
7. Navigate to **PeopleTools => Security => Permissions & Roles => Roles**.
8. Click **Add a New Value** to create a new role. Enter **IIQ\_HRMS\_ROLE** as the name of the Role, and click **Add**.
9. Enter **Allows access to the IIQ HRMS component interfaces** as the description.
10. Click the **Permission Lists** tab and add the **IIQ\_HRMS\_PERM** permission list. Click **Save** to save the role.
11. Navigate to **PeopleTools => Security => User Profiles**, and select the user that is being used in the connector.
12. Click the **Roles** tab and add the **IIQ\_HRMS\_ROLE** role. Click **Save** to add the role to the user.

### Creating PeopleSoft HRMS Jar File

---

Perform the following steps to create the **PeopleSoftHRMS.jar** file:

1. Login to PeopleSoft Application Designer in two tier mode.
2. Open the component interface which will be used in provisioning rule. For example, **CI\_PERSONAL\_DATA**
3. From the menu select **Build => PeopleSoft APIs**.
4. From the Build PeopleSoft API Bindings window, select the JAVA classes Build checkbox and deselect the **COM Type Library** and **C Header Files Build** check boxes.

5. In the JAVA Classes frame check Build and select the appropriate Component Interfaces from the drop down menu. Select the following options from the drop down menu:

- ComplIntfc.ComplIntfcPropertyInfo
- ComplIntfc.ComplIntfcPropertyInfoCollection
- PeopleSoft\*
- ComplIntfc.CI\_PERSONAL\_DATA\*
- ComplIntfc.CI\_JOB\_DATA\*

Specify the appropriate file path for the JAVA files. The Component Interface JAVA files are generated in the PeopleSoft\Generated\CompIntfc directory that is created in the specified location.

For example, if you specify C:\CI as the file path, then the Component Interface Java files are generated in C:\CI\PeopleSoft\Generated\CompIntfc.

6. Compile the JAVA files by performing the following steps:

- a. Open the command prompt and change directories to the folder where the generated JAVA files are located. For example, C:\CI.
- b. Navigate to PeopleSoft\Generated\CompIntfc\ directory.
- c. Run the following command:

```
javac -classpath %PS_HOME%\class\psjoa.jar *.java
```

Where %PS\_HOME% is the location that PeopleSoft is installed.

**Note:** Ensure that the JAVA compiler used for compiling the generated JAVA files is compatible with the JAVA provided with the PeopleSoft installation that needs to be managed.

- d. (Optional) You can delete all the generated java files from the existing directory, except the .class files.
7. Perform the following steps to package the compiled files as the PeopleSoftHRMS.jar file:
  - a. Open the Command prompt and change directories to the folder where the generated JAVA files are located. For example, if the java files are generated in C:\CI\PeopleSoft\Generated\CompIntfc folder, then run the command from cd C:\CI
  - b. Run the command: jar -cvf PeopleSoftHRMS.jar \*
8. Copy the generated PeopleSoftHRMS.jar and %PS\_HOME%\class\psjoa.jar files to the computer where IdentityIQ is running.
9. The location of the above jar file must be specified for the Jar location configuration attribute during connector configuration.

# Troubleshooting

---

## 1 - Cannot find Component Interface

- The Cannot find Component Interface {CI\_PERSONAL\_DATA} (91,2) ERROR (0,0) : Failed to execute PSSession request error message appears when the Component Interface is not present on the PeopleSoft HRMS Application server  
**Resolution:** Ensure that the Component Interface is present on the PeopleSoft HRMS Application Server.
- The jar file is created but inflated structure is not properly extracted.  
**Resolution:** Extract the jar file and ensure that the inflated structure is present in the PeopleSoft/Generated/CompIntfc directory. If inflated structure is not properly created, recreate it as mentioned in the “Creating PeopleSoft HRMS Jar File” on page 462 section.

## 2 - While performing test connection, when the supported platform version is Java 1.6 an error message appears

When the supported platform version is Java version 1.6, the following error message appears:

```
java.lang.UnsupportedClassVersionError: psft/pt8/joa/API: Unsupported major.minor version 51.0 (unable to load class psft.pt8.joa.API)
```

**Resolution:** Ensure that the supported platform version is Java 1.7.

## 3 - While performing the test connection or aggregation the SQL Query error may appear

The following error message appears if the user does not have minimum permission:

```
Error in SQL Query: User might not have permission to access the table mention in the query
```

**Resolution:** Ensure that the administrator permissions mentioned in the “Administrator permission” on page 458 section are provided.

## 4 - While performing provisioning, error message appears if proper access rights are not provided for Component Interface

The following error message appears if minimum permissions are not provided for Component Interface:

```
Operation failed. Please Check logs for more details. InvocationTargetException: Method save for Class: PeopleSoft.Generated.CompIntfc.CiPersonalData not able to invoke
```

**Resolution:** Ensure that the steps mentioned in “Configuring Component Interface Security” on page 462 are performed.

# Chapter 53: SailPoint RSA Authentication Manager Connector

---

The following topics are discussed in this chapter:

Overview .....	465
Supported features .....	466
Supported Managed Systems .....	466
Pre-requisites .....	466
Administrator permissions .....	467
Configuration parameters .....	468
Schema attributes .....	469
Account attributes .....	469
Group attributes .....	470
Provisioning Policy attributes .....	470
Additional information .....	471

## Overview

---

**Note: SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

SailPoint RSA Authentication Manager Connector manages the Users, Groups and Access Tokens in the RSA Authentication Manager. The connector manages the following entities:

- Users
- Groups
- Administrative Roles
- Secure ID tokens

The RSA groups are considered as the account-group with support for provisioning (Create, Update, and Delete) while Administrative Roles are additional entitlements which can only be assigned or removed.

## **Supported features**

---

SailPoint RSA Authentication Manager Connector supports the following features:

- Account Management
  - Manages RSA Users as Accounts
  - Aggregation, Refresh Accounts, Pass Through Authentication
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password, PIN Reset
  - Add/Remove entitlements
- Account - Group Management
  - Manages RSA User Groups as Account-Groups
  - Aggregation, Refresh Groups
  - Create, Update, Delete
- Permission Management
  - Application reads RSA tokens directly assigned to accounts as direct permissions during account aggregation.
  - The connector supports automated revocation of the aggregated permissions.

**Note:** To aggregate direct permissions while aggregation, select the ‘Include Permission’ check box from account or group schema.

### **References**

- “RSA Token PIN Reset” on page 467

## **Supported Managed Systems**

---

SailPoint RSA Authentication Manager Connector supports the following versions of RSA Authentication Manager:

- RSA Authentication Manager 8.1 SP1
- RSA Authentication Manager 8.1

## **Pre-requisites**

---

- **Configuring the Trust Store**

Server root certificate should be imported into the keystore for the remote API calls. Ensure to add the following Java option to the application server for SSL SOAP connections:

`-Djavax.net.ssl.trustStore = <Path of the imported root certificate>`

- **Importing the Server Root Certificate (Java)**

When RSA Authentication Manager is installed, the system creates a self-signed root certificate and stores it in `RSA_AM_HOME/server/security/server_name.jks` directory. This certificate must be exported from the server, and import it into the keystore for remote API clients. Use the Java keytool, as described in the following sections to export and import the certificate into Java clients.

- To export the server root certificate:

- a. Change directories to RSA\_AM\_HOME/appserver/ and enter the following:

```
jdk/jre/bin/keytool -export -keystore
RSA_AM_HOME/server/security/server_name.jks -file am_root.cer -alias
rsa_am_ca
```

- b. At the prompt for **keystore\_password**, press **Enter** without the password.

**Note:** **Ignore the warning message that appears as the server root certificate will still be exported.**

- c. The Java keytool outputs the certificate file to the RSA\_AM\_HOME/appserver/ directory.

- To import the server root certificate (Java):

**Note:** **Provide your cacerts keystore password to import the server root certificate. The Java default is "changeit".**

- a. Locate the server root certificate file that you exported from Authentication Manager, and copy it to the target host.
- b. Import the certificate to the local cacerts keystore. Change directories to JAVA\_HOME/jre/bin, and enter the following:

```
keytool -import -keystore SDK_HOME/lib/java/trust.jks -storepass
cacerts_keystore_password -file am_root.cer -alias rsa_am_ca -trustcacerts
```

- c. The Java keytool displays a confirmation that the certificate was added to the keystore.

## Administrator permissions

---

The RSA Authentication Manager Connector administrator must have enough rights to execute the requested operation.

To assign the rights to the administrator:

- Assign the default administrative roles present on the RSA. For most of the operations **Auth Mgr Realm Admin** administrative role must be assigned.

**Note:** **For RSA Authentication Manager version 8.1, the TrustedRealmAdminRole administrative role must be assigned.**

- Create new administrative roles with relevant permissions and assigning it a scope and then assign it to the administrator. The scope of an administrative role determines in what security domains an administrator may manage objects and from what identity sources an administrator may manage users. Below are the permissions which can be assigned to the administrative role.

- **All** grants an administrator permission to perform any administrative action on the object.
- **Delete** grants an administrator permission to delete an object.
- **Add** grants an administrator permission to add an object.
- **Edit** grants an administrator permission to view and edit an object, but not the ability to add or delete.
- **View** grants an administrator permission to view an object, but not to add, edit, or delete.

## RSA Token PIN Reset

---

The RSA Authentication Manager Connector supports updating the PIN of assigned RSA tokens to an identity.

## Configuration parameters

### *Configuring RSA Token PIN Reset feature*

1. Open IdentityIQ console and import `IIQHOME\WEB-INF\config\workflow_RSA_PIN_Reset.xml` file. The `workflow_RSA_PIN_Reset.xml` file creates the **Update My RSA Token PIN** quick link on the dashboard and adds the Update RSA Token PIN workflow.
2. The **Update My RSA Token PIN** quick link must be visible when logged into IdentityIQ.

## Configuration parameters

---

The following table lists the configuration parameters of RSA Authentication Manager Connector:

Parameters	Description
Host	The RSA Authentication Manager to connect to.
Port	The port to use to connect to RSA Authentication Manager. Default: 7002.
Administrator	The account that has permission to connect to the RSA Authentication Manager resource remotely. This account should have permission to manage this resource.
Password	Password of the Administrator account.
Command Client User*	<p>The command client user name. On installation of RSA Authentication Manager, the system creates a command client user name and password for secure connections to the command server. This user name and password are randomly generated on creation, and are unique to each deployment.</p> <p><b>Note:</b> For obtaining the command client user name, see “Obtaining the command client user name and password from Authentication Manager”.</p>
Command Client Password*	<p>Command Client Password corresponding to the Command Client User.</p> <p><b>Note:</b> For obtaining the command client password, see “Obtaining the command client user name and password from Authentication Manager”.</p>
Realm	Name of the Realm to manage.
Identity Source	Identity Source name linked to the Realm.
Security Domain	Name of the security domain to manage.
Search Subdomain	Whether or not to manage the subdomain, when the parent security domain is specified for <b>Security Domain</b> field.
Page Size	Limit to fetch number of accounts or groups per iteration through RSA Authentication Manager. Default: 500.

### **Obtaining the command client user name and password from Authentication Manager**

1. From a command prompt on Authentication Manager host, change directories to `RSA_AM_HOME/utils`.
2. Enter the following:  
`rsautil manage-secrets --action list`
3. When prompted, enter your master password.

The system displays the list of internal system passwords.

4. Locate the values for command client user name and password.

For example:

```
Command Client User Name.....: CmdClient_vKr9aLK9
Command Client User Password.....: e9SHbK0W4i
```

## Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports the following types of objects:

- **Account:** Account objects are used when building identities Link objects.
- **Group:** The group schema is used when building AccountGroup objects that are used to hold entitlements shared across identities.

## Account attributes

---

The following table lists the account attributes:

Attributes	Data type	Description
Guid	String	Guid of the entity (Native Identity).
userID	String	Unique name by which the entity is known by. (Display Attribute).
firstName	String	First name of the entity.
middleName	String	Middle name of the entity.
lastName	String	Last name for which the entity is known by.
Notes	String	Notes or description for the entity.
Email	String	Email of the entity.
certificateDN	String	Certificate DN of the entity.
securityDomain	String	Security Domain Name to which entity belongs.
identitySource	String	Identity Source Name to which entity belongs.
lastModifiedBy	String	Administrator or user who modified the entity last time.
Groups	Multivalued String	Groups Membership (marked as "groupAttribute").
Roles	Multivalued String	Administrative roles assigned to the entity.
lastModifiedOn	String	Last time when the entity was modified.
accountStartDate	String	Time when the entity was created.
accountExpireDate	String	Time when the entity will get expired.
lastLogin	String	Last time when the entity was logged in.

## Provisioning Policy attributes

Attributes	Data type	Description
forceChangePassword	Boolean	Whether or not user need to change the password during next logon.
Mobile Number	String	Mobile number of the entity.

## Group attributes

The following table lists the group attributes:

Attributes	Data type	Description
guid	String	Guid of the entity (Native identity).
groupName	String	Name by which the entity is known by (Display Attribute).
Notes	String	Notes or description for the entity.
securityDomain	String	Security Domain Name to which entity belongs.
identitySource	String	Identity Source Name to which entity belongs.

## Provisioning Policy attributes

The following table lists the provisioning policy attributes for Create Account of RSA Authentication Manager Connector.

**Note:** The attributes marked with \* sign are the required attributes.

Attributes	Description
<b>Provisioning policy attributes for Create Account</b>	
userID*	Unique name by which the entity is known by.
password*	Password for RSA user.
lastName*	Last Name of the user.
firstName	First name of the user.
email	Email of the user.
forceChangePassword	Whether or not user need to change the password during next logon.
nextAvailableToken	Select to assign the next available SecureID token to the user.
<b>Provisioning policy attributes for UpdateGroup</b>	
groupName	Name of the group.
notes	Notes or description of the group.
securityDomain	Security Domain Name to which group belongs (Read only attribute).

Attributes	Description
identitySource	Identity Source Name to which group belongs (Read only attribute).

## Additional information

---

This section describes the additional information related to the RSA Connector.

### Active Directory configured as an identity source

---

When configuring Active Directory as an identity source, it is recommended to verify the default LDAP policy on the active directory server and check for **MaxPageSize** that limits the number of objects that the server will return. The default value is 1000.

Perform the following steps to verify the quotas on the Active Directory server:

1. Open the ADSI Edit page.
2. In the Configuration partition window, navigate to **Services ==> Windows NT ==> Directory Service ==> Query Policies**.
3. In the left pane, click on the **Query Policies** container, then right-click on the **Default Query Policy** object in the right pane, and select **Properties**.
4. Double-click on the **IDAPAdminLimits** attribute and select the **MaxPageSize** attribute.
5. Click Remove and modify the value in the **Value to add** field to add the new value (for example, **MaxPageSize=2000**) and click **Add**.
6. Click **OK** twice.

**Note:** LDAP policy can also be modified using `Ntdsutil.exe`, follow instructions mentioned in <https://support.microsoft.com/en-us/kb/315071> to view and set LDAP policy on Active Directory server.

## **Additional information**

# Chapter 54: SailPoint Remedyforce Connector

---

The following topics are discussed in this chapter:

Overview .....	473
Supported features .....	473
Configuration parameters.....	474
Schema attributes .....	475
Account attributes .....	476
Additional account attributes for Remedyforce connector.....	477
Profile attributes .....	478
Provisioning Policy attributes .....	478
Troubleshooting.....	479

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

The SailPoint Remedyforce Connector supports reading and provisioning of Remedyforce accounts, profiles as account groups and implement the **sailpoint.connector** interface.

This connector is written using the `partner.wsdl` and underlying soap interface. The connector uses SOAP stub generated from a wsdl that was available at the time of development. The stubs are generated using axis 1.2. We do not have to generate the stubs once already done. Partner API is easy to use and we can add custom attributes in the schema without generating the stubs. Partner API is generic and have the same java implementation for RemedyForce connector.

The API is fairly rich for SOAP based API and has the concept of login which requires us to login just once for each operation. It has formal models around the user and profile objects and they are generated as part of the stubs. RemedyForce extends account schema of remedyforce to accommodate extra attributes related to BMC Remedy systems.

## Supported features

---

SailPoint Remedyforce Connector supports the following features:

- Account Management
  - Manages Remedyforce users as Accounts
  - Aggregation, Refresh Accounts
  - Create, Update,
  - Enable, Disable, Change Password
  - Add entitlement (Account-Groups and User Roles)
  - Add and Remove entitlements (PermissionSet)

## Configuration parameters

- Account - Group Management
  - Manages Remedyforce Profiles as Account-Groups
  - Aggregation, Refresh Groups
- Permission Management
  - Application reads permissions directly assigned to groups as direct permissions during group aggregation.
  - The connector does not support automated revocation of the aggregated permissions and creates work item for such requests

# Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Remedyforce connector uses the following connection parameters:

**Table 1—Configuration parameters**

Parameters	Description
Salesforce URL*	<p>Enter the fully qualified url to the root of the remedyforce server. For example, <code>http://login.salesforce.com/services/Soap/u/26.0/</code></p> <p><b>Note:</b> To figure out the url of your site, login to <a href="#">remedyforce.com</a>. Click Develop under the Application heading toward bottom. Next, click API &gt; Generate Partner WSDL, and click Generate. The URL is located under the SalesforceService service name.</p>
Username*	<p>Display name attribute. It's typically in an email in email type format.</p> <p>For example, <code>denise.hunt@demoexample.com</code></p>
Password*	Defines the password which is used for logging in the managed System.
Manage Active Accounts	Retrieves the active accounts during account aggregation. Otherwise it retrieves all the accounts which are enabled/disabled while account aggregation.
Search Query for User/Profile	<p>Helps to scope the User/Profile that are retrieved during Account or Account-Group aggregation.</p> <p>For instance, specifying the following search query, retrieves only Active Users during Account Aggregation:</p> <pre>select Id from User where IsActive = true</pre> <p>Users or Profiles retrieved during aggregation can be scoped using custom attributes in the where clause as follows:</p> <pre>select Id from User where EMP_DEPARTMENT__c= 'tester'</pre> <p><b>Note:</b> Only the where clause of the search query can be modified as per the customers requirement.</p> <p>While configuring Remedyforce application if where clause in <b>Search Query For User/Profile</b> field contains apostrophe(') then use backslash(\) prefix to apostrophe. For example,</p> <pre>Find Id,username from user where lastname is buru'4</pre> <p>Expected Query: Select Id,username from user WHERE lastname = 'buru\'4'</p>

**Note:** In the above table all the attributes marked with \* sign are mandatory attributes.

## Schema attributes

---

This section provides the different attributes of the Account attributes and Profile attributes for Remedyforce connector.

## Account attributes

---

The Remedyforce connector returns several attributes falling into two categories. The first are general attributes: name, city, state, and so on. Additionally, there are entitlement attributes that specifies user level access granted to Remedyforce:

Attributes	Description
UserName	By default, this attribute is the connectors default nativIdentity AND display name attributes. It's typically in an email type format. For example, denise.hunt@demoexample.com
Id	This attribute is the connector default nativIdentity and internal salesforce id like "005A00000014ySylXX".
Name	User's fullname
FirstName	User's firstname
LastName	User's lastname
Alias	User's assigned alias
City	User's city
CommunityNickname	DisplayNames for user's online communities
CallCenterId	User's call center
CompanyName	User's company name
Country	User's country
Department	User's department
Email	User's Email address
Division	User's division
EmployeeNumber	User's employee number
Extension	User's telephone extension
Fax	User's fax number
IsActive	Flag that indicates if the user is active in sf. False would indicate disabled.
EmailEncodingKey	Encoding that should be used during email communications
ProfileId	ID of the profile assigned to a user. Profiles contain settings and permissions, which control what users can do. The available profiles depend on which user license is selected.
ProfileName	Name of the profile assigned to a user. Profiles contain settings and permissions, which control what users can do. The available profiles depend on which user license is selected.
UserRoleId	User Role's Id.
UserRoleName	User Role's name.

Attributes	Description
PublicGroups	Public groups are the entitlements for user.
UserPermissionsMarketingUser	Maps to the Marketing User Flag.
UserPermissionsMobileUser	Maps to the Mobile User Flag.
UserPermissionsOfflineUser	Maps to the Offline user Flag.
Phone	User's phone number.
ReceivesAdminInfoEmails	Receive the remedyforce.com administrator newsletter.
UserType	Type of the user.
UserPermissionsSFContentUser	Maps to Sales Anywhere User.
ReceivesInfoEmails	Receive theremedyforce.com newsletter.
State	User's state.
Title	User's title.

**Note:** The ProfileId and UserRoleid fields are required in the schema to fetch the ProfileName and userRoleName respectively. If the ProfileId or UserRoleid is removed then profile name and user role name will not be fetched.

### Additional account attributes for Remedyforce connector

In addition to the above account attributes, following are the additional custom attributes which are required for configuration to connect to Remedyforce connector:

Attributes	Description
BMCServicedesk__IsStaffUser__c	Maps to BMC ServiceDesk Staff
BMCServicedesk__Remedyforce_Knowledge_User__c	Maps to Remedyforce Knowledge User
BMCServicedesk__Account_Name__c	Maps to Account Name
BMCServicedesk__remarks__c	Maps to Remarks
BMCServicedesk__IsOutOfOffice__c	Maps to Out of Office
BMCServicedesk__ContactId__c	Maps to Contact Id
BMCServicedesk__Account_ID__c	Maps to Account ID
BMCServicedesk__FPLoginID__c	Maps to FootPrints Login ID
BMCServicedesk__Room__c	Maps to Room attribute

In addition to the above account attributes, following is the additional schema attribute which is required for configuration to connect to Remedyforce connector:

**Note:** The attribute in the following table must be added manually when upgrading from any version below 7.0 to version 7.1.

## Provisioning Policy attributes

Attribute	Description
PermissionSet	Entitlement and Managed System property must be enabled. It is the PermissionSet assigned to a user. PermissionSet contains settings and permissions which control users action. The available Permission depends on which user license is selected. User can have multiple permission sets.

## Profile attributes

Profiles are aggregated during account group aggregation, below are the attributes returned by the group aggregation process.

Attributes	Description
Id	The internal id for this group. For example, 00eA000000OoP6IAK.
Name	The friendly name assigned to the profile. For example, Force.com - Free User, it also has to be unique so is used as the identity and display attribute by default.
UserType	This is the type of profile even though the attribute name would indicate a user.
Description	Description for the profiles.
UserLicense	User's license.

**DirectPermissions:** The connector reads the permissions assigned to a profile using the remedyforce.com api. To get the permissions, the connector queries the service to describe the profile object. In the returned attribute all of the permissions contained by a group are prefixed with **Permissions**, and camel cases the permission such that right and target are separated by camel case convention. For example, **PermissionsEditTask** or **PermissionsTransferAnyEntity**. We break these down into a Permission attribute per prefixed-attribute.

## Provisioning Policy attributes

IdentityIQ has a default Provisioning Policy defined which allows for the creation of accounts. The provisioning policy can be edited to fit specific customer environments.

Most of the fields on the Remedyforce connector default provisioning policy are generated and all fields are marked review required. The provisioning policy attributes must be customized based on specific customer requirements.

Attributes	Description
<b>Create User Policy</b>	
Alias	8 character alias, which is required. By default, it generates a value based on lastname and firstname in the field's inline script. It takes first 7 chars from last name and prefixes it with the first character of the first name.
IsActive	Defaults to true.
Username	Defaults to the identity's email address.
Email	Defaults to the identity's email address.

Attributes	Description
FirstName	Defaults to the identity's first name.
Lastname	Defaults to the identity's last name
CommunityNickname	Defaults to identity's full name.
TimeZoneSidKey	Defaults to America/Los_Angeles. The timezone of the user, it uses a display name defined by sales force. Only a few timezones are defined in the policy drop down and this will need to be customized for each customer.
LocaleSidKey	Defaults to UTF-8. This is the user's locale.
Email EncodingKey	Defaults to UTF-8 and there are several selections to choose from in from the web interface. They can be customized by the customer.
LanguageLocaleKey	Defaults to en_US. There are several selections to choose from in from the web interface. They can be customized by the customer.

## Troubleshooting

---

- The Community nickname must be unique.
- Do not add profileId/userRoleId attribute in create /update user policy as the code would automatically handle when the customer is selecting profile name and userrolename from **Entitlement** section.

## **Troubleshooting**

# Chapter 55: SailPoint SAP Connector

---

The following topics are discussed in this chapter:

Overview .....	481
Supported features .....	481
Supported Managed Systems .....	482
Pre-requisites .....	483
Administrator permissions .....	483
Configuration parameters .....	487
Schema attributes .....	489
Account attributes .....	489
Group attributes .....	493
Schema extension and custom attributes .....	494
Upgrade considerations .....	494
Provisioning Policy attributes .....	494
Create account attributes .....	494
Additional information .....	495
Entitlement validity period .....	495
CUA support .....	495
Entitlement Data .....	495
Password Change .....	495
Logon and Communication Language attributes .....	496
Troubleshooting .....	497

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

SAP Enterprise Resource Planning software solution is an integrated software solution that incorporates the key business functions of the organization.

The SAP Connector aggregates and provisions all the users along with their roles/profiles of the SAP system.

SailPoint SAP Connector supports provisioning to a standalone SAP system as well as SAP Central User Administration (CUA) system.

## Supported features

---

SailPoint SAP Connector supports the following features:

- Account Management
  - Manages SAP users as Accounts
  - Aggregation, Partitioning Aggregation, Delta Aggregation, Refresh Accounts, Pass Through Authentication

## Overview

**Note:** SAP Connector aggregates Generated Profile associated to Role as a part of Account-Group Aggregation.

- Create, Update, Delete
- Enable, Disable, Unlock
- Change Password

**Note:** For "Change password in Permanent Mode" ensure that the SNC is configured on SAP server. The log on session during which a productive password is set must be secured using Secure Network Communications (SNC).

**Note:** SAP recommends that setting of productive passwords is more risky than setting an initial one, therefore additional security checks must be applied as follows:

- The log on session during which a productive password is set must be secured using Secure Network Communications (SNC).
- The user needs an additional authorization to set a productive password (authorization object: S\_USER\_GRP, activity: 'PP' - Set Productive)

For more information, see SAP note <https://service.sap.com/sap/support/notes/1287410> (SAP Service marketplace login required)

- Add/Remove Entitlements

Entitlements are Roles (for user), Profiles (for user), UserGroup (User group of the user).

- Account - Group Management

- Manages SAP Roles as Account-Groups
- Manages SAP Profiles as Account-Groups

**Note:** Few system composite profiles might have child profiles which are not present in SAP system. For example, for each release composite profile SAP\_NEW contains a single profile SAP\_NEW\_<rel>, (for example, SAP\_NEW\_21D). This profiles holds its release status. Profiles like SAP\_NEW\_<rel> may not be aggregated.

- Aggregation, Refresh Groups

**Note:** In Account-Group aggregation for SAP CUA landscape, SAP Connector will not fetch child roles, child profiles of any composite role and profile, as CUA system does not maintain child level roles and profile details for child subsystems. Same way it will not fetch TCodes and Generated Profile for group object type.

## References

- "Appendix A: Delta Aggregation"
- "Appendix C: Partitioning Aggregation"

## Supported Managed Systems

---

Following versions of SAP NetWeaver system are supported by the SAP connector:

- SAP NetWeaver 7.5, 7.4, 7.31, 7.3, 7.2, 7.1 and 7.0

**Note:** SailPoint SAP Connector manages ABAP users. For more information, see "Supported features" on page 481.

## Pre-requisites

---

SAP JCO version 3.0.x libraries, along with `sapjco3.dll` (on Microsoft Windows) or `libsapjco3.so` (on UNIX), must be present in the `java.library.path` directory on the host. The JCO libraries (JCO Release 3.0.x) must be downloaded from the SAP website by navigating to the customer service marketplace and download the Java Connector.

## Administrator permissions

---

The following table lists the required permissions for the specific operations mentioned below in this section:

**Table 1— Operation specific required permissions**

Operation	Required permissions
Test Connection	Test Connection
Account Aggregation	Test Connection and Account Aggregation  <b>Note: For Account Aggregation of CUA systems, additional permissions must be executed as specified in the “ Account Aggregation” section.</b>
Group Aggregation	Test Connection and Group Aggregation  <b>Note: For Group Aggregation of CUA systems, additional permissions must be executed as specified in the “ Group Aggregation” section.</b>
Delta Aggregation	Test Connection, Account Aggregation and Delta Aggregation
Create Account	Test Connection, Account Aggregation and Create Account  <b>Note: For Create Account of CUA systems or SNC network, additional permissions must be executed as specified in the “ Create Account (Create user with assign role and profiles)” section.</b>
Enable/Disable/Unlock Account	Test Connection, Account Aggregation and Enable/Disable/Unlock Account
Delete Account	Test Connection, Account Aggregation and Delete Account
Add/Remove Entitlement	Test Connection, Account Aggregation and Add/Remove Entitlement
Change Password	Test Connection, Account Aggregation and Change Password  <b>Note: For Change Password of SNC network, additional permissions must be executed as specified in the “ Add/Remove Entitlements and Change Password” section.</b>

The role assigned to the SAP Administrative user must have the following Authorization Objects as mentioned in the tables below.

## Overview

### Test Connection

Authorization Objects	Field name	Field description	Field value
S_RFC	ACTVT	Activity	16 - Execute
	RFC_NAME	Name of RFC object	RFCPING
	RFC_TYPE	Type of RFC object	FUGR, FUNC

### Account Aggregation

Authorization Objects	Field name	Field description	Field value
S_RFC	RFC_NAME	Name of RFC object	BAPI_USER_GETLIST, BAPI_USER_GET_DETAIL, DDIF_FIELDINFO_GET, MSS_GET_SY_DATE_TIME, RFC_GET_FUNCTION_INTERFACE, SDTX, SMSSDATA1, SU_USER
S_TABU_NAM	ACTVT	Activity	03 - Display
	TABLE Name	TABLE	USR06, USR02, TUTYP
S_USER_GRP	ACTVT	Activity	03 - Display
	CLASS	User group in user master maintenance	* or specify the Group you want to assign for the user.  For example, SUPER

- Additional permissions for CUA systems

Authorization Objects	Field name	Field description	Field value
S_RFC	RFC_NAME	Name of RFC object	BAPI_USER_LOCACTGROUPS_READ, BAPI_USER_LOCPROFILES_READ

## Group Aggregation

Authorization Objects	Field name	Field description	Field value
S_RFC	ACTVT	Activity	03 - Display
	RFC_NAME	Name of RFC object	BAPI_HELPVALUES_GET, PRGN_ACTIVITY_GROUPS_LOAD_RFC, PRGN_EXCHANGE, COLL_ACTGROUPS_GET_ACTGROUPS, DDIF_FIELDINFO_GET, MSS_GET_SY_DATE_TIME, PRGN_COLLECTIVE_ACTGROUPS, RFC_GET_FUNCTION_INTERFACE, SDTX, SMSSDATA1
S_TABU_NAM	TABLE Name	TABLE	AGR_FLAGS, AGR_PROF, AGR_TCODES, AGR_TEXTS (Roles), USR11, UST10C (Profiles)

- Additional permissions for CUA systems

Authorization Objects	Field name	Field description	Field value
S_TABU_NAM	TABLE Name	TABLE	<ul style="list-style-type: none"> <li>(Profiles) USRSYSPRF, USRSYSPRFT</li> <li>(Roles) USRSYSACTT, USRSYSACT</li> </ul>

## Delta Aggregation

Authorization Objects	Field name	Field description	Field value
S_RFC	RFC_NAME	Name of RFC object	/SAILPOIN/USR_CHANGE_DOC_USERS , /SAILPOIN/IDENTITYIQ_FUGR, /SAILPOIN/USR_CHANGE_DOC_ROLES
S_TABU_NAM	TABLE Name	TABLE	USBAPILINK
S_USER_GRP	ACTVT	Activity	08 - Display change document

## Overview

### Create Account (Create user with assign role and profiles)

Authorization Objects	Field name	Field description	Field value
S_USER_GRP	ACTVT	Activity	01 - Create or generate
S_RFC	RFC_NAME	Name of RFC object	SDIFRUNTIME
S_USER_SAS	ACTVT	Activity	22 - Enter, Include, Assign, 01 - Create
	ACT_GROUP	Role name	* or you can specify role name for which you have assigned
	CLASS	User group in user master maintenance	* or specify the Group you want to assign for the user. For example, SUPER
	PROFILE	Auth. profile in user master maintenance	* or you can specify Profile for which you have assigned
	SUBSYSTEM	Receiving system for central user administration	* or specify the system you are targeting.

- For SNC (Secure Network Communication)

Authorization Objects	Field name	Field description	Field value
S_USER_GRP	ACTVT	Activity	PP – Set Productive

### Enable/Disable/Unlock Account

Authorization Objects	Field name	Field description	Field value
S_USER_GRP	ACTVT	Activity	05 - Lock

### Delete Account

Authorization Objects	Field name	Field description	Field value
S_USER_GRP	ACTVT	Activity	06 - Delete

## Add/Remove Entitlements and Change Password

Authorization Objects	Field name	Field description	Field value
S_USER_GRP	ACTVT	Activity	02 - Change, 05 - Lock
S_RFC	RFC_NAME	Name of RFC object	SDIFRUNTIME
S_USER_SAS	ACTVT	Activity	22 - Enter, Include, Assign
	ACT_GROUP	Role name	* or you can specify role name for which you have assigned
	CLASS	User group in user master maintenance	* or specify the Group you want to assign for the user. For example, SUPER
	PROFILE	Auth. profile in user master maintenance	* or you can specify Profile for which you have assigned
	SUBSYSTEM	Receiving system for central user administration	* or specify the system you are targeting.

- **(For Change Password only) For SNC (Secure Network Communication)**

Authorization Objects	Field name	Field description	Field value
S_USER_GRP	ACTVT	Activity	PP – Set Productive

## Configuration parameters

The following table lists the configuration parameters of SAP Connector:

Parameters	Description
SAP Host*	Host on which the SAP Server is running
System Number*	2-digit SAP system number (Default: 00)
Client Number*	3-digit SAP client number (Default: 001)
Client Language*	2-letter SAP client language (Default: EN)
Username*	SAP Administrator user
Password*	SAP Administrator user password
CUA system	For CUA system detection
Unlock on Password Change	If checked, the account would be unlocked while changing password. <b>Note: Account will be unlocked at the time of set password only if the account is locked by incorrect password attempts.</b>
Partition Enabled	Check box to determine if partition aggregation is required.

## Configuration parameters

Parameters	Description
Partition Statements	<p>Criteria to specify the range of users to be downloaded.</p> <p>For example, If the range is specified as <b>A-M</b>, then this specifies that all the users whose User ID's are between A and M (including A and M) would be treated as one partition and downloaded.</p> <p>To specify more than one partition the entries should be separated using a new line character. For more information, see Appendix C: Partitioning Aggregation.</p>
<b>SNC Configuration parameters</b>	
SNC Mode	Represents Secure Network Connection which also internally signifies <code>jco.client.snc_mode</code> in SAP. SNC will be enabled if the mode is selected as ON whose value is 1. If SNC is off, the value will be 0.
SNC Level of Security	<p>Represents the quality of protection level (QOP) which is defined as follows:</p> <ul style="list-style-type: none"> <li>1 — Apply authentication only</li> <li>2 — Apply integrity protection (authentication)</li> <li>3 — Apply privacy protection (integrity and authentication)</li> <li>8 — Apply the default protection</li> <li>9 — Apply the maximum protection</li> </ul> <p>In SAP, it relates to <code>jco.client.snc_qop</code>. Default: 1</p>
SNC Partner Name	<p>Represents SNC partner.</p> <p>For example, provide input as p:CN=R3, O=XYZ-INC, C=EN in SAP. If SNC is configured, it relates to <code>jco.client.snc_partnername</code>.</p>
SNC Name	Represents SNC name which internally signifies <code>jco.client.snc_myname</code> . It overrides default SNC partner.
SNC Library	<p>Path to library which provides SNC service. It internally signifies <code>jco.client.snc_lib</code>.</p> <p>For example, the value to be passed:</p> <ul style="list-style-type: none"> <li>• on Microsoft Windows: <code>C:/sapcrypto/lib/sapcrypto.dll</code> (the location of the cryptographic library)</li> <li>• on UNIX: <code>/opt/sailpoint/lib/custom/libsapcrypto.so</code> (the location of the cryptographic library)</li> </ul>
<b>SAP GRC Settings parameters</b>	
Enable SAP GRC	Enables the application for SAP GRC policy violation checks.
SAP GRC Connector Name	SAP GRC Connector name which is configured on GRC server for this application.
<b>Note:</b> For more information on SAP GRC configuration, see <i>SailPoint IdentityIQ Integration Guide</i> .	

**Note:** Attributes marked with \* sign are the mandatory attributes.

# Schema attributes

---

This section describes the different schema attributes.

## Account attributes

---

The following table lists the account attributes:

Attributes	Description
Academic Title (Address)	Academic title of the user.
Academic Title 2 (Address)	2nd Academic title of the user.
Addr Number (Address)	Address number of the user.
Alias (Logon Data)	Alias name.
Birth Name (Address)	Name at birth.
Building (Address)	Name of the building.
Building 2 (Address)	Name 2 of the building.
Building Long (Address)	Long name of the building.
Care of (Address)	Care of name.
Check Status (Address)	Check status for the user.
City (Address)	Name of the city.
City Number (Address)	Number of the city.
Code (Address)	Signature initials
Communication Language (Address)	Communication language of the user.  <b>Note:</b> The different values to be set for this attribute are mentioned in “Logon and Communication Language attributes” on page 496.
Communication type (Address)	Communication method for the user.
Company (Address)	Name of the company.
Company Address (Address)	Address of the company.
Company Address 2 (Address)	Address 2 of the company.
Company Address 3 (Address)	Address 3 of the company.
Company Address 4 (Address)	Address 4 of the company.
Country (Address)	Name of the country.
Country ISO (Address)	ISO name of the country.
Delivery District (Address)	Delivery district name.
Department (Address)	Department name.
District (Address)	District name.

## Schema attributes

Attributes	Description
District Number (Address)	District number for the user.
E-Mail (Address)	E-mail address.
E-Mail List (Address)	E-mail address list.
Employee Number (Address)	Employee number of the user.
Fax (Address)	Fax number.
Fax Extension (Address)	Fax extension number
Fax List (Address)	Fax number list
First name (Address)	First name of the user
Floor (Address)	Floor number
Floor 2 (Address)	Floor 2 number
Format (Address)	Format name
Full Name (Address)	Full name of the user
Full Name 2 (Address)	Full name 2 of the user
Function (Address)	Function of the user
GUI Flag	Unsecured communication permitted.
House Number 2 (Address)	House number 2 of the user
House Number (Address)	House number of the user
House Number 3 (Address)	House number 3 of the user
Inhouse ML (Address)	Inhouse mail of the user
Initials (Address)	Initials of the user
Language CR P (Address)	CR P language of the user
Language ISO (Address)	ISO language of the user
Language UCP ISO (Address)	CP ISO language of the user
Language UP ISO (Address)	P ISO language of the user
Last Name (Address)	Last name of the user
Location (Address)	Location name
Logon Language (Defaults)	Logon language for the user.  <b>Note:</b> The different values to be set for this attribute are mentioned in “Logon and Communication Language attributes” on page 496.
Middle Name (Address)	Middle name of the user
Name Country (Address)	Name of the country
Nickname (Address)	Nickname of the user
Notes (Address)	Notes for the user

Attributes	Description
Other City (Address)	Name of the other city
Other City Number (Address)	Number of the other city
Pager/SMS List (Address)	Pager or SMS number list in the format pager_type#pager_number
Parameter List (Parameters)	Parameter list in the format parameter_ID=parameter_value
Pboxcity Number (Address)	Pbox number of the city
PCODE 1 Ext (Address)	Postal code 1 extension
PCODE 2 Ext (Address)	Postal code 2 extension
PCODE 3 Ext (Address)	Postal code 3 extension
PO Box (Address)	PO box number
PO Box City (Address)	PO box number of the city
PO Box City ISO (Address)	PO box number of the ISO city
PO Box Country (Address)	PO box number of the country
PO Box Region (Address)	PO box number of the region
PO Box Without Number (Address)	PO box without number
Postal Code (Address)	Postal code of the user
Postal Code 2 (Address)	2nd postal code of the user
Postal Code 3 (Address)	3rd postal code of the user
Prefix 1 (Address)	1st prefix
Prefix 2 (Address)	2nd prefix
Print Immediately (Defaults)	Print immediately flag for the user
Printer List (Address)	Print destination list
Region (Address)	Name of the region
Region Group (Address)	Group name of the region
Remote Communication List (Address)	Communication notes list
Remote Function Call List (Address)	Remote function call destination list
Remote Mail List (Address)	Remote mail list of the user
Room Number (Address)	Room number of the user
Room Number 2 (Address)	2nd room number of the user
Reference User	Reference user name.
Search Term 2 P (Address)	2nd search term P for the user
Search Term P (Address)	Search term P for the user
Search Term 1 (Address)	1st search term for the user
Search Term 2 (Address)	2nd search term for the user

## Schema attributes

Attributes	Description
Second Name (Address)	Second name of the user
Start Menu (Defaults)	Start menu for the user
Street Abbreviation (Address)	Street abbreviation for the user
Street Address (Address)	Street address of the user
Street Address 2 (Address)	Street address 2 of the user
Street Address 3 (Address)	Street address 3 of the user
Street Address 4 (Address)	Street address 4 of the user
Street Number (Address)	Street number of the user
SNC Name	SNC name.
Tax Jurisdiction Code (Address)	Tax jurisdiction code of the user
Telephone (Address)	Telephone number
Telephone Extension (Address)	Telephone extension number
Telephone List (Address)	Telephone number list
Teletex List (Address)	Teletex number list
Telex List (Address)	Telex number list
Time Format (Defaults)	Time format of the user
Time Zone (Address)	System time zone.
Title (Address)	Title of the user
Title SPPL (Address)	Title SPPL of the user
Transportation Zone (Address)	Transportation zone of the user
TZone (Defaults)	Personal time zone.
URL (Homepage) List (Address)	URL (Homepage) address list in the format URI_type#URI_name
User Last Logon Time	User last log in time.
User Last Logon Date	User last log in date.
Productive Password	User password set in permanent mode.
User Name	User Name.
User Title (Address)	Title of the user
User Type (Logon Data)	Type of the user
User Valid From (Logon Data)	Valid from date for the user
User Valid To (Logon Data)	Valid to date for the user.
User Group (Groups)	User group of the user
X.400 List (Address)	Organization name list

Attributes	Description
Roles	Roles for user.  <b>Note:</b> The Account Aggregation fetches the active roles (composite /simple) assigned directly to the user.
Profiles	Profiles for user.

## Group attributes

---

The following table lists the different group attributes:

Attributes	Description
<b>Group Object Type = Role</b>	
Name	Role name.
Type	Role type.
Description	Role description.
Child Roles	Sub Role list.  <b>Note:</b> The child roles will display the child roles of composite roles in the Group object properties of Entitlement Catalog. For existing applications which are getting upgraded, mark entitlement as true to display the child roles in Entitlement grid of Group object properties.
Long Description	Role long description.
Subsystem	System name for CUA System Aggregation.
Generated Profile	System generated profile associated to Role which has authorizations.
TCodes	Transaction code list.
<b>Group Object Type = Profile</b>	
ID	Profile name along with the description.
Name	Profile name.
Type	Profile type.
Description	Profile description.
Subsystem	System name for CUA System Aggregation.
Child Profiles	Sub profile list.

## Schema extension and custom attributes

---

The schema can be extended up to the extent of the fields within the structures provisioned by the SAP standard BAPI. The fields in the following structures will be provisioned:

- ADDRESS
- ALIAS
- COMPANY
- DEFAULTS
- LOGONDATA
- PASSWORD

**Note:** No custom attributes will be supported during provisioning.

## Upgrade considerations

---

While upgrading to IdentityIQ version 7.1, perform the following changes at schema level:

- Ensure that in **Role** schema, following attributes are added with appropriate properties:
  - Generated Profile
  - TCodes (Entitlement, Multi-Valued)
- In order to achieve the profile aggregation functionality for an existing application in previous releases it is recommended to perform the following procedure:
  - Add **Profile** schema under the Settings tab in the application page
  - In Account Schema the **schemaObjectType** attribute of Profiles must be changed to **profile**.
- To skip the inactive roles assignment during aggregation, add the following line in the application debug page:  
`<entry key="skipInactiveRoles" value="true" />`

**Note:** When upgrading IdentityIQ from version 6.x to 7.1, ensure that the 'Include Permissions' check box in Role schema is not selected.

# Provisioning Policy attributes

---

This section lists the different policy attributes of SAP Connector.

**Note:** The attributes marked with \* sign are the required attributes.

## Create account attributes

---

The following table lists the provisioning policy attributes for Create Account:

Attributes	Description
User Name*	Name of the user to create.
password	Password for the user.

Attributes	Description
Last Name*	Last name of the user.

## Additional information

---

This section describes the additional information related to the SAP Connector.

### Entitlement validity period

---

The user can be assigned a SAP Role with Start Date and an End Date. The ability to select or specify the same, while requesting an entitlement for an account, is available in IdentityIQ by creating custom Provisioning Plan.

### CUA support

---

By default the SAP Connector would not download data from CUA configured SAP System. In order to override this behavior, the **CUASystem** configuration parameter must be checked in configuration parameter list.

### Entitlement Data

---

The aggregated entitlement data consists of the following:

- SAP Roles (Simple and Composite)
- SAP Profiles (Simple and Composite)

### Password Change

---

The following change password policy must be added to set password as productive using administrative change password request.

```
<Form name="con_prov_policy_user_create_username" objectType="account"
type="ChangePassword">
    <Attributes>
        <Map>
            <entry key="IIQTemplateOwnerDefinition">
                <value>
                    <DynamicValue value=" "/>
                </value>
            </entry>
        </Map>
    </Attributes>
    <Field displayName="Productive Password" filterString=""
helpKey="ProductivePasswordFlag" name="Productive Password" required="true"
type="string" value="true">
        <AllowedValuesDefinition>
            <Value>
                <List>
                    <String>true</String>
                    <String>false</String>
                </List>
            </Value>
        </AllowedValuesDefinition>
    </Field>
</Form>
```

## Additional information

```
</Value>
</AllowedValuesDefinition>
</Field>
</Form>
```

## Logon and Communication Language attributes

---

The different values for Logon and Communication Language fields are as follows:

- Serbian=0
- Chinese=1
- Thai=2
- Korean=3
- Romanian=4
- Slovenian=5
- Croatian=6
- Malay=7
- Ukrainian=8
- Estonian=9
- Afrikaans=a
- Icelandic=b
- Catalan=c
- Serbian (Latin)=d
- Indonesian=i
- Arabic=A
- Hebrew=B
- Czech=C
- German=D
- English=E
- French=F
- Greek=G
- Hungarian=H
- Italian=l
- Japanese=J
- Danish=K
- Polish=L
- Chinese traditional=M
- Dutch=N
- Norwegian=O
- Portuguese=P
- Slovak=Q
- Russian=R
- Spanish=S
- Turkish=T
- Finnish=U
- Swedish=V
- Bulgarian=W
- Lithuanian=X
- Latvian=Y
- Customer reserve=Z

# Troubleshooting

---

## 1 - Distribution of a user to SAP CUA Subsystem

In a SAP CUA landscape, a SAP role or profile requires a SUBSYSTEM to distribute the user to. The facility to select or specify the same, while requesting an entitlement for an account, is absent in IdentityIQ.

**Workaround:** The subsystem name is prepended to the Account-Group while aggregating account-groups from a SAP CUA system. As a result, only a limited subset of subsystem and account-group combinations will be available while requesting entitlements, and thus distributing users, in a SAP CUA landscape.

## 2 - Removed Entitlements are present in Current access page

Even after the execution of **Refresh Entitlement Correlation** the entitlements are not getting deleted from the current access page.

**Workaround:** Execute the **Perform Identity Request Maintenance** task to remove those entitlements. Ensure that the **Verify provisioning for requests** option is selected for this task.

## 3 - Password not set in permanent mode

After upgrade to the existing application, the password is not set in permanent mode, even when the user is created with the **Password in permanent mode** attribute selected.

This behavior occurs since the attribute name has changed from **Password in permanent mode** to **Productive Password**.

**Workaround:** In the debug page rename **Password in permanent mode** to **Productive Password** in schema and provisioning plan.

## 4 - Few attributes are not working after upgrading to version 7.1

Few attributes are not working after upgrading from version 6.0 patch 7 and version 6.1 to version 7.1.

**Resolution:** Open the application debug page of version 6.4 and use the following corresponding parameters:

Parameters used in version 6.0 patch 7/6.1	Parameters to be used in version 7.1
Password in permanent mode	Productive Password
Deactivate	Password Deactivated
LASTNAME	Last name
Reference User Name	Reference User
User Last Login	User Last Logon Time

## 5 - Login fails for non aggregated accounts when passthrough is enabled

Login fails for non aggregated accounts when passthrough is enabled.

In SAP-Direct connector the SAPJCO libraries are used, which need permission to make connection with SAP Server. The user who does not have these permissions will not be able to log in and will not be a valid member of the authentication process.

## Troubleshooting

**Resolution:** Perform the following to add the administrator permissions:

1. Run the **PFCG** transaction (Profile generator, maintain your roles, authorizations, and profiles) and enter the role name.
2. Click on **Single** and save the Role created.
3. Click on **Authorization Tab => Display Authorization Data**.  
Template will appear, cancel the template.
4. Click on **Manual** tab and add the following:
  - S\_RFC (All Activities)
  - S\_USER\_AGR (Activities: 02, 03, 22, 36, 78)
  - S\_USER\_GRP (Activities: 01, 02, 03, 05, 06, 22, 78)
  - S\_USER\_PRO (Activities: 01, 02, 03, 06, 07, 22)
  - S\_USER\_AUT (Activities : 03, 08)
  - S\_USER\_SAS (Activities : 01, 06, 22)
  - S\_TABU\_DIS (Activities: All Activities)  
(Additionally for SAP CUA System) S\_USER\_SYS (Activities: 03, 59, 68, 78)
5. Click on the **Generate (Shift+F5)** icon.
  - Click on the **Save (Ctrl+S)** icon.
  - Click on **Back (F3)** icon.
6. Click on the **Generate (Shift+F5)** icon and assign the above created role to a SAP user who must be an administrator.
7. Run the **PFCG** transaction.
8. Provide the role name which the customer has created.
8. Click on **USER tab => User Comparison**.

### 6 - When performing Delta Aggregation after upgrade, an error message appears

When performing Delta Aggregation after upgrade, the following error message appears:

Aggregation date needs to be set in configuration.

**Resolution:** Open the SAP-Direct application debug page and set the following parameters:

```
<entry key="lastAggregationDate" value="2014-06-21"/>
<entry key="lastAggregationTime" value="20:54:34"/>
```

In the above parameters the format of Date and Time are as follows:

- **Date:** yyyy-MM-dd (the date should be the current date of the SAP server)
- **Time:** HH:mm:ss (the time should be the current time of the SAP server)

### 7 - Change password feature is not working with SNC, when PRODUCITVE\_PWD attribute is X

Change password feature is not working with SNC, when PRODUCITVE\_PWD attribute is X.

**Resolution:** Define the **productivePasswordValue** attribute in debug pages as follows:

```
<entry key="productivePasswordValue" value="1">
```

By default the code would consider the value as x.

## 8 - Aggregation fails with error 'NOT AUTHORIZATION'

Aggregation fails with the following error due to not having proper authorization of authorization object 'S\_TABU\_DIS (Activities: All Activities)'.

**Resolution:** Provide the authorization of authorization object 'S\_TABU\_DIS (Activities: All Activities)'

Activities-All

Table Authorization Group-\* (means all)

Or skip aggregation of license data of the user by adding the following entry key in debug pages of the application:

```
<entry key="skipLicenseData">
  <value>
    <Boolean>true</Boolean>
  </value>
</entry>
```

## 9 - Test connection fails with an error message

Test connection fails with the following error message:

com.sap.conn.rfc.driver.CpicDirver

**Resolution:** Download the latest SAPJCO.jar and SAPJCO.dll files from SAP Marketplace and then use that SAPJCO Jar file with the latest downloaded SAPJCO dll file.

## 10 - Role and Profile description in a language other than English

**Resolution:** In Account-Group Aggregation, if the Role and Profile Description is required in a language other than English language, add the **descriptionLanguage** parameter with the correct value.

For example, `<entry key="descriptionLanguage" value="D" />`

In the above example, the value 'D' is the language code for Dutch language supported by SAP.

If the **descriptionLanguage** parameter is not provided, the descriptions displayed are in English language.

## 11 - Login to IdentityIQ fails for username and password with utf8 characters

The following error message appears when login to IdentityIQ for username and password with utf8 characters:

```
ERROR http-8080-1 sailpoint.server.Authenticator:323 -
sailpoint.connector.AuthenticationFailedExcept
com.sap.conn.jco.JCoException: (109) RFC_ERROR_CANCELLED: Handle close pending
```

**Resolution:** Add the following entry in the application debug page:

```
<entry key="jco.client.codepage" value="4110" />
```

## 12 - Test connection / aggregation fails with an error message

Test connection / aggregation fails with the following error message:

## Troubleshooting

Bad username or password. com.sap.conn.jco.JCoException: (109)

RFC\_ERROR\_CANCELLED: Handle close pending

**Resolution:** Ensure that the administrator user specified in application has sufficient rights on the SAP systems as mentioned in the “Administrator permissions” on page 483 section.

# Chapter 56: SailPoint System for Cross-Domain Identity Management Connector

---

The following topics are discussed in this chapter:

Overview .....	501
Supported features .....	501
Administrator permissions .....	502
Configuration parameters .....	502
Schema attributes .....	504
Account attributes .....	504
Group attributes .....	506
Provisioning Policy attributes .....	506
Create account attributes .....	507
Update group attributes .....	507
Troubleshooting .....	507

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

The SCIM (System for Cross-Domain Identity Management) standard defines a schema and API to create, read, update, and delete identity and identity-related information on other systems. This standard creates a common language, by which a client system can communicate with many different servers in the same way. SaaS providers (such as Salesforce) and other software vendors are beginning to adopt this standard, and are exposing their identity management interfaces through SCIM.

## Supported features

---

SailPoint SCIM Connector supports the following features:

- Account Management
  - Manage SCIM Users as Account
  - Aggregation, Refresh Accounts, Discover Schema
  - Create, Update, Delete
  - Enable, Disable
  - Change Password
  - Add/Remove Entitlements

## Configuration parameters

- Account - Group Management
  - Aggregation, Discover Schema
  - Create, Update, Delete
  - Enable, Disable
  - Change Password

## Administrator permissions

---

The required administrator permissions depends on which SCIM server is being connected to.

# Configuration parameters

---

The following table lists the configuration parameters of SCIM Connector:

Parameters	Description
Base URL*	The base URL to connect to the SCIM server.
Authentication Type*	Select one of the following method of authentication to the SCIM 2 server: <ul style="list-style-type: none"><li>• Basic (username and password)</li><li>• OAuth 2.0 (bearer token)</li><li>• OAuth 1.0 Token (a bearer token using the “Bearer” header)</li></ul>
Username*	Username for the SCIM Server. <b>Note: Required if the ‘Authentication Type’ is selected as ‘Basic’.</b>
Password*	Password for the SCIM Server. <b>Note: Required if the ‘Authentication Type’ is selected as ‘Basic’.</b>
OAuth2	The OAuth bearer token to use for authorization. <b>Note: Required if the ‘Authentication Type’ is selected as ‘OAuth’.</b>
Content Type*	Either XML or JSON (default).
ETag Required*	Whether ETags are required by the SCIM server for versioning resources.

**Note:** Attributes marked with \* sign are the mandatory attributes.

## Additional configuration parameters

---

Add the following parameters in the application debug page:

- **queryAccountSchemaAttributes** and **queryGroupSchemaAttributes**: Retrieves specified query attributes during account and group aggregation.

For example:

```
<entry key="queryAccountSchemaAttributes">
  <value>
```

```

<List>
  <String>userName</String>
  <String>urn:scim:schemas:extension:enterprise:1.0:manager</String>
</List>
</value>
</entry>

```

**Note:** Specify the fully qualified extended attributes with the associated Resource URN.

For example, the 'manager' attribute defined in

urn:scim:schemas:extension:enterprise is fully encoded as  
urn:scim:schemas:extension:enterprise:manager.

- **skipSchemaAttributes:** During update operation, SCIM connector skips the specified attributes.

For example:

```

<entry key="skipSchemaAttributes">
  <value>
    <List>
      <String>alias</String>
      <String>groups</String>
    </List>
  </value>
</entry>

```

- **extensionSchemaAttributes:** Provides the list of fully qualified extended attributes with the associated Resource URN.

For example:

```

<entry key="extensionSchemaAttributes">
  <value>
    <List>
      <String>urn:scim:schemas:extension:enterprise:manager</String>
    </List>
  </value>
</entry>

```

- **scimAttrMapping:** Defines mapping of connector schema attributes with SCIM schema attributes.

For example:

```

<entry key="scimAttrMapping">
  <value>
    <Map>
      <entry key="name">
        <value>
          <Map>
            <entry key="familyName" value="familyName"/>
            <entry key="formattedName" value="formatted"/>
            <entry key="givenName" value="givenName"/>
          </Map>
        </value>
      </entry>
    </Map>
  </value>
</entry>

```

## Schema attributes

In the above example, `<entry key="name">` is the SCIM complex attribute created with subAttributes `familyName`, `formatted` and `givenName` which are mapped to `familyName`, `formattedName` and `givenName` Connector schema attributes respectively.

# Schema attributes

---

This section describes the different schema attributes.

## Account attributes

---

The following table lists the account attributes:

Attributes	Description
<code>id</code>	A unique identifier for a SCIM resource.
<code>userName</code>	Name of the user.
<code>externalId</code>	A String that is an identifier for the resource.
<code>displayName</code>	The name of the user.
<code>nickName</code>	The casual way to address the user in real life.
<code>profileUrl</code>	A fully qualified URL to a page representing the Users Online profile.
<code>title</code>	The user's title, such as <b>Vice President</b> .
<code>userType</code>	Used to identify the organization to user relationship. Values can be: <b>Contractor, Employee, Intern, Temp, External, and Unknown</b>
<code>preferredLanguage</code>	Indicates the User's preferred written or spoken language.
<code>locale</code>	Used to indicate the User's default location for purposes of localizing items such as currency, date time format, numerical representations, and so on.
<code>timezone</code>	The User's time zone in the <b>Olson</b> timezone database format.
<code>formattedName</code>	The full name, including all middle names, titles, and suffixes as appropriate, formatted for display.
<code>familyName</code>	The family name of the User, or <b>Last Name</b> in most Western languages.
<code>givenName</code>	The given name of the User, or <b>First Name</b> in most Western languages.
<code>middleName</code>	The middle name(s) of the User.
<code>honorificPrefix</code>	The honorific prefix(es) of the User, or <b>Title</b> in most Western languages. For example, Ms. given the full name Ms. Barbara Jane Jensen, III.
<code>honorificSuffix</code>	The honorific suffix(es) of the User, or <b>Suffix</b> in most Western languages. For example, III. given the full name Ms. Barbara Jane Jensen, III.
<code>employeeNumber</code>	Numeric or alphanumeric identifier assigned to a person, typically based on order of hire or association with an organization.
<code>costCenter</code>	Identifies the name of a cost center.
<code>organization</code>	Identifies the name of an organization.

Attributes	Description
division	Identifies the name of a division.
department	Identifies the name of a department.
managerId	The id of the SCIM resource representing the Users manager.
managerName	Name of the manager.
emails	E-mail addresses for the User. The value must be canonicalized by the Service Provider. For example, <code>bjensen@example.com</code> instead of <code>bjensen@EXAMPLE.COM</code> .
emails_objects	A list of all of the users email addresses, including their type (for example, home, work) and whether it is their primary address.
emails_primary	The users primary email address.
phoneNumbers	Phone numbers for the User. The value must be Canonicalized by the Service Provider.
phoneNumbers_objects	A list of all of the users phone numbers, including their type (for example, home, work) and whether it is their primary phone number.
phoneNumbers_primary	The users primary phone number.
ims	Instant messaging address for the User.
ims_objects	A list of all of the instant messaging usernames, including their type (for example, aim, gtalk) and whether it is their primary messaging username.
ims_primary	The users primary instant messaging username.
photos	URL of a photo of the User.
photos_objects	A list of URLs of all of the users photos, including the photo type (for example, photo, thumbnail) and whether it is their primary photo.
photos_primary	The URL of the users primary photo.
addresses	A physical mailing address for this User.
addresses_objects	A list of all of the users physical mailing addresses, including their type (for example, home, work) and whether it is their primary address.
addresses_primary	The users primary physical mailing address.
groups	A list of groups that the user belongs to, either thorough direct membership, nested groups, or dynamically calculated.
groups_objects	A list of all of the users group memberships, including how they are assigned to the group (for example, <b>direct</b> : if assigned directly or <b>indirect</b> : if assigned indirectly through another group) and whether it is their primary group.
groups_primary	The users primary group membership.
entitlements	A list of entitlements for the User that represent a thing the User has.
entitlements_objects	A list of all of the users entitlements, including whether it is their primary entitlement.

## Provisioning Policy attributes

Attributes	Description
entitlements_primary	The users primary entitlement.
roles	A list of roles for the User that collectively represent who the User is. For example, Student, Faculty.
roles_objects	A list of all of the user's roles, including whether it is their primary role.
roles_primary	The users primary role.
created	The DateTime the Resource was added to the Service Provider.
lastModified	The most recent DateTime the details of this Resource were updated at the Service Provider.
location	The URI of the Resource being returned. This value MUST be the same as the Location HTTP response header.
version	The version of the Resource being returned. This value must be the same as the ETag HTTP response header.

## Group attributes

The following table lists the group attributes:

Attributes	Description
id	A unique identifier for a SCIM resource.
externalId	A String that is an identifier for the resource.
displayName	The name for the Group.
members	A list of members of the Group.
memberGroups	A list of the sub-groups of this group.
created	The DateTime the Resource was added to the Service Provider.
lastModified	The most recent DateTime the details of this Resource were updated at the Service Provider.
location	The URI of the Resource being returned. This value MUST be the same as the Location HTTP response header.
version	The version of the Resource being returned. This value must be the same as the ETag HTTP response header.

## Provisioning Policy attributes

This section lists the different policy attributes of SCIM Connector.

**Note:** The attributes marked with \* sign are the required attributes.

## Create account attributes

---

The following table lists the provisioning policy attributes for Create Account:

Attributes	Description
User Name	Name of the user to create.
FirstName	First name of the user.
Full Name	Full name of the user.
Last Name	Last name of the user.
Password	Password of the user.
Email	Email ID of the user.
Email Type	Email type of the user.
Email Primary	Determine if the email is primary.

## Update group attributes

---

The following table lists the provisioning policy attributes for Update Group:

Attributes	Description
Display Name	The display name of the group.
External ID	The external identifier of the group, which can natively identify the group on the resource.

# Troubleshooting

---

## 1 - Mapping of connector schema attributes with SCIM schema attributes fails with an error message

If **scimAttrMapping** attribute is missing in the application debug page, the following error message is displayed:

Unable to add entry 'uid=john,ou=people,dc=example,dc=com' because it violates the provided schema: The entry is missing required attribute cn

**Resolution:** Add the **scimAttrMapping** attribute to the application debug page.

For more information, see “ Additional configuration parameters” section.

## 2 - Provisioning failed with an error message

Provisioning fails with the following error message:

Resource 'User' is malformed: Attribute urn:scim:schemas:core:1.0:alias is not defined for resource User

**Resolution:** Add alias attribute in **skipSchemaAttributes** list, which is not supported by SCIM server.

## Troubleshooting

### 3 - Test Connection fails with an error message

The following error message appears during Test Connection for SCIM servers where **ServiceProviderConfigs** endpoint is not available (for example, WSO2):

```
com.unboundid.scim.sdk.ResourceNotFoundException: Not Found
```

**Resolution:** Ignore this error message and continue as Test connection fails but other operations can be performed successfully.

### 4 - Create account fails for Salesforce SCIM server with an error message

Create account fails for Salesforce SCIM server with the following error message:

```
REQUIRED_FIELD_MISSING:user_must_have_one_entitlement_which_must_be_a_profileid;
```

**Resolution:** Add the **Entitlement** field which is mandatory for Salesforce SCIM server from **Application ==> Provisioning Policies ==> Create Account Policy ==> Add New Field**. Enter the setting for **Entitlement** field as Required.

# Chapter 57: SailPoint WebEx Connector

---

The following topics are discussed in this chapter:

Overview .....	509
Supported features .....	509
Pre-requisites .....	510
Administrator permissions .....	510
Configuration parameters .....	510
Schema attributes .....	510
Account attributes .....	510
Group attributes .....	512
Provisioning Policy attributes .....	513

## Overview

---

**Note:** SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.

The SailPoint WebEx Connector manages WebEx accounts and groups (Meeting Types). It supports read and write for WebEx accounts. The WebEx connector supports creation, deletion, retrieval, authentication and unlock for users and retrieval for groups.

**Note:** In the WebEx connector, Meeting Types are treated as Groups.

## Supported features

---

SailPoint WebEx Connector supports the following features:

- Account Management
  - Manages Webex Users as Accounts
  - Aggregation, Refresh Accounts
  - Create, Update, Delete
 

**Note:** When performing Delete operation, the account does not get deleted but gets disabled.
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements
- Account - Group Management
  - Manages Webex meeting types as Account-Groups
  - Aggregation, Refresh Groups

## Pre-requisites

---

**Note:** If WebEx Connector is behind proxy server, see the “Special Java Considerations” section of the *SailPoint IdentityIQ Installation Guide*.

## Administrator permissions

---

The user must be a **Site Administrator**.

# Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The WebEx Connector uses the connection attributes listed in the following table:

Parameters	Description
webExID	WebEx user ID for the meeting host.
password	The password for the user with a webExID.
siteID	The WebEx-assigned identification number that uniquely identifies your website.
siteName	The first string in your WebEx site URL, provided by WebEx. For example, if <b>acme</b> is the siteName for the <a href="https://acme.webex.com">https://acme.webex.com</a> site.
partnerID	(Optional) A reference to the WebEx partner, provided by WebEx.
xmIURL	XML URL of the site. For example, WBXService/XMLService
Manage Disabled Accounts	If set to yes, the disabled accounts will be a part of the Aggregation.

## Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports the following types of objects:

**Account:** Account objects are used when building identities Link objects.

**Group:** The group schema is used when building AccountGroup objects that are used to hold entitlements shared across identities.

## Account attributes

---

The following table lists the account attributes:

Attributes	Description
WebexID	WebEx ID of the user.
FirstName	First Name of the user.
LastName	Last Name of the user.
Email	Email id of the user.
RegistrationDate	The creation date of the user
Active	<p>Determines whether the user account has been staged for use. Default: ACTIVATED</p> <p><b>Note:</b> If you set the <b>user's active</b> parameter to <b>ACTIVATED</b> such that the WebEx sites host limit is exceeded, the <b>CreateUser</b> or <b>SetUser</b> command displays the following error: <code>exceededSiteHostLimit</code></p>
TimezoneID	Determines the time zone for the geographic location of the meeting.
Company	The user's company name.
Description	A description of the user's virtual office.
CategoryID	A reference to the office category for the user's office.
AddressType	Determines whether the meeting participant is a personal contact of the meeting host or is a site-wide (global) contact.
Country	The country for the user.
Phone	The user's Office Profile phone number.
MobilePhone	The attendee's mobile phone number.
Fax	Indicates the fax number for the user.
Pager	The user's Office Profile pager number.
PersonalURL	The user's website.
ExpirationDate	A WebEx-maintained date and time at which the user's account expires.
Prod/ServiceAnnouncement	Indicates product or service announcements.
TrainingInfo	Indicates training information.
ElectronicInfo	Indicates electronic information.
Promos	Indicates promotions and special offers.
PressRelease	Indicates press releases.
UserEmail	The email address as stored in the user profile.
UserPhone	Indicates the phone number for the user.
MailInfo	Indicates the mail information for the user.
TimeZone	Determines the time zone for the geographic location of the user or user's office.

## Schema attributes

Attributes	Description
TimeZoneWithDST	A timezone description which is adjusted by DST. For example, GMT-7:00, Pacific (San Francisco)
Service	The type of service that the user has.
Host	Indicates whether the user is the host for the meeting.
TelephoneConferenceCallOut	Indicates whether conference calling out of meetings is supported for the meeting.
TelephoneConferenceCallOutInternational	Indicates whether international calling out of meetings is supported for the meeting.
TelephoneConferenceCallIn	Indicates whether conference calling into meetings is supported for the meeting.
TelephoneConferenceTollFreeCallIn	Indicates whether toll-free calling into meetings is supported for the user.
SiteAdmin	Indicates whether the user has administrative privilege for the meeting.
VOIP	Specifies whether Voice Over IP telephony is enabled.
SiteAdminwithViewOnly	Indicates whether the current user is a site administrator with view only privilege.
LabAdmin	If TRUE, then user has access to the Hands-on Lab administration pages.
OtherTeleConferencing	Specifies whether a user account has the privilege to schedule a session with the <b>other teleconferencing</b> feature enabled. Default value depends on the configurations on the user's website.
TeleConferenceCallInInternational	Allows a user to access WebEx teleconferencing through international local call-in telephone numbers.
AttendeeOnly	If the value is TRUE, indicates that the user's role is attendee only. If the value is set to TRUE, then the <b>host</b> , <b>siteAdmin</b> , <b>labAdmin</b> and <b>roSiteAdmin</b> elements should be FALSE.
RecordingEditor	Indicates whether a user has the privilege to download WebEx Recording Editor from My WebEx > Support.
MeetingAssist	Enables Meeting Assist.
MeetingType	The meeting types of which the account is a part of.

## Group attributes

---

The following table lists the group attributes:

Attributes	Description
ProductCodePrefix	Indicates the product label for the type of meeting.
Active	Indicates whether the type of meeting represented by an object of this type is enabled or disabled.

Attributes	Description
DisplayName	The display name for the meeting type.
PrimaryTollCallInNumber	The telephone number for a toll call-in teleconference.
PrimaryTollFreeCallInNumber	The telephone number for a toll free call-in teleconference.
GroupName	The name of the group or meeting type
MeetingTypeID	Specifies IDs for the meeting types whose detailed information you want to get.
ServiceType	The type of meeting being returned.

## Provisioning Policy attributes

---

The following table lists the provisioning policy attributes for create:

Attributes	Description
AccountType	<b>Host:</b> Indicates whether the user is the host for the meeting.
	<b>SiteAdmin:</b> Indicates whether the user has administrative privilege for the meeting.
	<b>SiteAdminWithViewOnly:</b> Indicates whether the current user is a site administrator with view only privilege.
WebexID	WebEx ID of the user.
FirstName	First Name of the user.
LastName	Last Name of the user.
Email	Email ID of the user.
TelephoneConferenceCallOut	Indicates whether conference calling out of meetings is supported for the meeting.
TelephoneConferenceCallOutInternational	Indicates whether international calling out of meetings is supported for the meeting.
TelephoneConferenceCallIn	Indicates whether conference calling into meetings is supported for the meeting.
TelephoneConferenceTollFreeCallIn	Indicates whether toll-free calling into meetings is supported for the user.
VOIP	Specifies whether Voice Over IP telephony is enabled.
LabAdmin	If TRUE, then user has access to the Hands-on Lab administration pages.
OtherTeleConferencing	Specifies whether a user account has the privilege to schedule a session with <b>other teleconferencing</b> feature enabled. Default value depends on the configurations on the user's website.
TeleConferenceCallInInternational	Allows a user to access WebEx teleconferencing via international local call-in telephone numbers.

## Provisioning Policy attributes

Attributes	Description
RecordingEditor	Indicates whether a user has the privilege to download WebEx Recording Editor from My WebEx > Support.
WelcomeMessage	Holds a welcome message for when people enter the meeting room.

# Chapter 58: SailPoint Yammer Connector

---

The following topics are discussed in this chapter:

Overview .....	515
Supported features .....	515
Pre-requisites .....	515
Administrator permissions .....	516
Configuration parameter .....	516
Schema attributes .....	516
Account attributes .....	516
Group attributes .....	517

## Overview

---

**Note:** **SailPoint will provide assistance during the deployment of this Connector. Additional troubleshooting, diagnostic, and best practice information beyond what is contained in this document will be provided in the Connector and Integration Deployment Center on Compass.**

The SailPoint Yammer Connector is a *read only* connector which aggregates accounts and groups from one or more networks on Yammer (Enterprise Social Network).

## Supported features

---

SailPoint Yammer Connector supports the following features:

- Account Management
  - Manages Yammer Users as Accounts
  - Aggregation, Partitioning Aggregation, Refresh Accounts
- Account - Group Management
  - Manages Yammer Groups as Account-Groups
  - Aggregation, Refresh Groups

## Pre-requisites

---

**Note:** **If Yammer Connector is behind proxy server, see the “Special Java Considerations” section of the *SailPoint IdentityIQ Installation Guide*.**

The user will be walked through the **OAuth2** flow to generate the access token using the Cloud Commander and then pass it down to the Yammer connector. The connector will use this Access Token to make calls to any Yammer REST API.

## Administrator permissions

---

The Administrator should be configured to have proper access rights for reading people information in the social network within the organization.

## Configuration parameter

---

**Access Token:** A valid Access Token for the user is required which enables your application to access the user's information and take actions on their behalf. The application and user are verified with each API call by passing an access token along with each request.

## Schema attributes

---

The application schema is used to configure the objects returned from a connector. When a connector is called, the schema is supplied to the methods on the connector interface. This connector currently supports the following types of objects:

- **Account:** objects used when building identities Link objects.
- **Group:** schema used when building AccountGroup objects that are used to hold entitlements shared across identities.

## Account attributes

---

The following table lists the account attributes ([Table 1—Account attributes](#)):

**Table 1—Account attributes**

Attributes	Description
Admin	The user is an administrator in a specified network or not.
Department	The department of the user in the company.
Email	Email of the user .
EmailType	Type of Email (primary or secondary).
FullName	Full name of the user.
Groups	Groups to which user is a member of.
JobTitle	The job title of the user.
NetworkDomain	The Domain of the network of which the user is a member of.
NetworkID	The ID of the network of which user is a member of.
NetworkName	The name of the network of which user is a member of.
UserID	The ID of the user.
UserName	The user name internally stored by Yammer for each user.
UserType	The retrieved identity is the user.

**Table 1—Account attributes (Continued)**

Attributes	Description
UserURL	The url which stores the property of user.
UserWebURL	The url for the web page of the user on Yammer.
Location	The location of the user.
Summary	The summary of the user.

## Group attributes

---

The following table lists the group attributes ([Table 2—Group attributes](#)):

**Table 2—Group attributes**

Attributes	Description
GroupState	The group is active or not.
GroupType	The retrieved identity is group.
GroupWebURL	The URL of the web page for that group on Yammer.
GroupPrivacy	The group is private or public.
GroupURL	The URL stores the property of the group.
GroupDescription	The description given for the creation of the group.
GroupFullName	The full name of the group.
GroupName	The name of the group.
GroupMembers	Contains all the members of the group.
GroupID	The ID of the group.

## **Schema attributes**

# Collaborative Deployment Connectors

A minority of SailPoint customers have deployed the Connectors in this section. SailPoint will provide direct collaboration and oversight during the deployment of these connectors. Additional design, troubleshooting, diagnostic and best practice information beyond what is contained in this document will be provided on Compass, SailPoint's online customer portal. Due to the nature of deploying the connectors in this section, SailPoint will directly engage with the deployment team and actively participate in the design, configuration, and testing of the connectivity to the managed system.

For more specific information for these connectors, refer to the Connector and Integration Deployment Center on Compass.

This section contains information on the following:

- "SailPoint Cerner Connector" on page 521
- "SailPoint Epic Connector" on page 527
- "SailPoint GE Centricity Connector" on page 535



# Chapter 59: SailPoint Cerner Connector

---

The following topics are discussed in this chapter:

Overview .....	521
Supported features .....	521
Pre-requisites .....	522
Configuration parameters.....	522
Schema attributes .....	523
Account attributes .....	523
Group attributes.....	524
Provisioning Policy attributes .....	524
Troubleshooting.....	524

## Overview

---

**Note: SailPoint will provide direct collaboration and oversight during the deployment of this Connector.**

Cerner Corporation is a global supplier of health care information technology (HCIT) solutions, services, devices and hardware. Cerner solutions optimize processes for health care organizations. The SailPoint Cerner Connector is designed to provide automated way of provisioning through SailPoint IdentityIQ solution.

## Supported features

---

SailPoint Cerner Connector supports the following features:

- Account Management
  - Aggregation, Refresh Account
  - Create, Update, Delete
 

**Note: Delete operation will Disable the account on Cerner.**  
**The account must be enabled on Cerner system to perform any update operation.**
  - Enable, Disable, Change Password
  - Add/Remove Entitlements (position, organizations, and organization groups)
 

**Note: During Add/Remove Entitlements from Request Access, the update provisioning policy will be displayed with Confidentiality Level attribute when selecting any of the entitlements (Position, Organization, Organization Groups) if configured.**
  - Direct Permission of accounts- privilege will be treated as direct permissions of Cerner account
- Account - Group Management
  - Aggregation

### References

- “Appendix C: Partitioning Aggregation”

## Pre-requisites

---

- The Cerner Enterprise Provisioning Service exposes the provisioning mechanism to external requests and responses using the SPML (Service Provisioning Markup Language) standard Cerner Millennium provisioning language. The Service allows external provisioning solutions to create and maintain users. The Cerner connector accesses the enterprise provisioning service to perform all the requests. For accessing enterprise provisioning service, the target ID of the millennium and millennium domain is required.
- For accessing provisioning adaptor, the target ID of the millennium and millennium domain is required.
- Permissions given to TargetID in Cerner millennium:

The Cerner provisioning adaptor requires one Millennium account having Manage Accounts privilege, which modifies the users within Millennium. The service account is mapped to TargetID which is required in order to make calls to the provisioning adapter. The IdentityIQ requires targetID to connect to the Cerner provisioning adapter (through Cerner API's access). This is necessary, since Cerner has no way of defining users to have the authority to send requests.

## Configuration parameters

---

Parameters	Description
Cerner URL	URL to connect to Cerner Server. For example, <a href="http://&lt;hostName&gt;/security-provisioning/ProvisioningServlet">http://&lt;hostName&gt;/security-provisioning/ProvisioningServlet</a>
Target ID	ID with required permission to get the data and to perform provisioning on Cerner . Enter Valid Target ID.  For example, "millennium_xxxxxx"
Manage Active Accounts	The Manage Active accounts check box, if checked will aggregate only active accounts otherwise all accounts.
Partitioning Enabled	Determines if partition aggregation is required.
Partitioning Mode	Selection of search either by FirstName or lastName.
Partitioning Statements	Criteria to specify the range of users to be downloaded. For example, if the range is specified as A-M, then this specifies that all the Users whose firstName/lastName are between A and M (including A and M) would be treated as one partition and downloaded. For more information, see Appendix C: Partitioning Aggregation.

**Note:** - Time out setting is required if the response is getting delayed from the Cerner system. By default, timeout is set to 1 minute. The timeout settings can be configured from the application debug page as follows:

```
<entry key="timeout" value="1" />
```

- Cerner API's require version while executing the operations. Currently version 1.0 is supported. Version can be configured from the application debug page as follows:

```
<entry key="version" value="1.0" />
```

# Schema attributes

---

## Account attributes

---

The following table lists the account attributes ([Table 1—Account attributes](#)):

**Table 1—Account attributes**

Attributes	Description
ID	Identifies an object that exists on a target that is exposed by a provider
username	The user name associated with the account. The value of the user name field must be unique within the target Cerner Millennium domain. Any value between 1 and 48 characters
directoryIndicator	<ul style="list-style-type: none"> <li>• True (LDAP user)</li> <li>• False (non-LDAP user)</li> </ul> <p>Contains an indicator if the user is an LDAP directory user or not.</p>
birthdate	Birthdate of the personnel.
firstname	First name for the personnel.
lastname	Surname (last name) for the personnel.
middleName	Middle name of the personnel.
displayName	Display name for the personnel.
suffix	Suffix of the personnel.
privilege	Privileges assigned to the Cerner account.
gender	A coded value representing the gender of the personnel.
restriction	A restriction to be assigned to or unassigned from the account.
title	Title (or list of titles) for the personnel. For example, Dr. Mr. and so on.
physicianInd	An indicator if the personnel is a physician or not.
position	A coded value representing the position assigned to the personnel which is treated as Group entity.
beginEffectiveDateTime	Date/time at which the personnel becomes/became effective.
endEffectiveDateTime	Date/time at which the personnel ceases/ceased to be effective.
organization Group	When a personnel record is unassigned from an organization group, all organizations in the group will also be unassigned from the personnel record, unless they are associated to another organization group that is still assigned to the personnel. It will be read-only field and data will be displayed during account aggregation.
confidentialityLevel	A coded value representing the confidentiality code that applies to the relationship.

## Provisioning Policy attributes

**Table 1—Account attributes (Continued)**

Attributes	Description
personnelAlias	Personnel alias information.
credential	Personnel credential information. Only one active credential can exist for the personnel with the same credential name, type, and state.
address	Personnel addresses information. Only one active address of a given address type can be assigned to a personnel.
organization	The list of organizations assigned to the user.
personnelGroup	The personnel groups information assigned to user.

## Group attributes

The following table lists the group attributes ([Table 1—Account attributes](#)):

**Table 2—Group attributes**

Attributes	Description
Id	The Id of the group.
Display	Display name of the group.

## Provisioning Policy attributes

The following table lists the provisioning policy attributes for Create Account:

Attributes	Description
username	The user name associated with the account. The value of the user name field must be unique within target Cerner Millennium domain [1- 48 characters]
password	The password for the user account. Any value, assuming that value meets all criteria defined in the Cerner Millennium password policy maintained in AuthView. The password is only required when the user being provisioned is a non-LDAP user (when the user will authenticate against the Cerner Millennium user directory).
first name	Given (first) name for the personnel.
lastname	Surname (last name) for the personnel.
confidentialityLevel	The confidentiality level set for organization or organization groups.

## Troubleshooting

### 1 - An error message is displayed while performing the operations

The following error message is displayed while performing the operations:

`xml.soap.SOAPException: Read timed out" OR "call: Connection Refused: connect`

**Resolution:** Ensure that the Cerner server is up and running.

## 2 - Aggregation task fails with an error message

The Aggregation task fails with the following error message even when the test connection is successful:

An error has occurred retrieving user: XXXXXXXX

**Resolution:** Verify the read and Write privileges for the respective account.

## 3 - Insufficient privileges displayed in the Managed System

If insufficient privileges are displayed in the Managed System for a particular account and the domain server is not available, then the permissions of the account are disabled.

This issue is related to the Authorize server not running in the domain. This is caused by an issue with the server controller service.

**Resolution:** Perform the following to cycle the Millennium domain and resolve the issue:

- Run an **mbt –ctrl**, to verify if there were no orphaned processes
- Run an **mbs –ctrl** to restart

## **Troubleshooting**

# Chapter 60: SailPoint Epic Connector

---

The following topics are discussed in this chapter:

Overview .....	527
Supported features .....	527
Supported Managed System .....	528
Pre-requisites .....	528
Administrator permissions .....	528
Configuration parameters .....	528
Schema Attributes .....	529
Account attributes .....	529
Group attributes .....	530
Provisioning Policy attributes .....	531
Troubleshooting .....	532

## Overview

---

**Note: SailPoint will provide direct collaboration and oversight during the deployment of this Connector.**

Epic is a privately held health care software company. Epic offers an integrated suite of health care software centered on a MUMPS database. Their applications support functions related to patient care such as follows:

- including registration and scheduling
- clinical systems for doctors, nurses, emergency personnel, and other care providers
- systems for lab technicians, pharmacists, and radiologists
- billing systems for insurers

SailPoint Epic Connector supports managing Epic accounts (EMP records), linked templates and linked sub-templates.

## Supported features

---

SailPoint Epic Connector supports the following features:

- Account Management
  - Manage Epic EMP records as Accounts
  - Aggregation, Refresh Account
  - Create, Update, Delete
  - Enable, Disable, Unlock, Change Password
  - Add/Remove Entitlements (Epic Linked Template and Linked subtemplates)
- Account - Group Management
  - Manage Epic Linked Template as Account - Groups
  - Manage Epic Linked subtemplates as Account - Groups
  - Aggregation

## Supported Managed System

---

SailPoint Epic Connector supports Epic version 2015 and 2014.

### Pre-requisites

---

- **Epic Web Services:** Epic API's have provided SOAP based service calls supported by the Epic Web services. All communication with the Epic Interconnect server should be done via this service interface. For Epic connector to work, following web services must be enabled on Interconnect server and authentication and encryption settings must be configured appropriately:
  - **Core:** The Core WCF service fetches all the records matching specified filters. The connector uses this service to read all records with INI type as EMP.
  - **Personnel Management:** The personnel management is a web service that implements all the provisioning related API's used by the connector. In addition, it provides interface to read details about each of the EMP record that the Core service returns.

The **Core** and the **Personnel Management** Module of the Epic Web Services must be enabled for access. A debugging interface available on the Epic Web Services server, displays the enabled and disabled status of various Epic Web Services. This debugging interface must be used to view and verify that the required Web Services are enabled when integrating with IdentityIQ. The format of the URL for the diagnostic service is as follows:

**`http://[epic-webservices-server-name]/[epic-instancename]/StatusPage/Main.aspx`**

For example, **`http://example-epic-websrvr.acme.com/Interconnect-TST_POC2014/StatusPage/Main.aspx`**

- **Configuring the trust store:** For configuring the trust store, server root certificate should be imported into the keystore for the remote API calls. Ensure that the following java system property is set to the path of the imported root certificate for SSL SOAP connections:

**`Djavax.net.ssl.trustStore2 = <Path of the imported root certificate>`**

### Administrator permissions

---

To manage SailPoint Epic Connector, ensure that Web Services mentioned in the “ Pre-requisites” section must be enabled on Interconnect server.

## Configuration parameters

---

This section contains the information that this connector uses to connect and interact with the application. Each application type requires different information to create and maintain a connection.

The Epic Connector uses the following connection attributes:

Attribute	Description
URL	Specifies the host and port of Epic instance you are connecting.
Username	Specifies the administrator or the unique ID of the user which has Administrative level privileges to perform aggregation and provisioning operation on Epic system.
Password	Defines the password of the username.

Attribute	Description
Manage Active Accounts Only	(Applicable to Account aggregation only) By default this is selected and will aggregate only active accounts.
Page Size	The page size limit to fetch number of accounts or groups per iteration through Epic. Default: 500

## Schema Attributes

---

This section describes the different schema attributes.

### Account attributes

---

The following table lists the account attributes:

Attribute name	Description
UserID	Unique ID of the Epic user.
Name	The Epic user's name, in LastName, FirstName MI format.
SystemLoginID	The user's operating system login. The name must be unique.
UserAlias	Another name by which this user is known. Typically used for maiden names or other name changes. In Last, First format.
StartDate	The date the user started at the organization.
IsPasswordChangeRequired	Password change required Flag
BlockStatus	Indicates whether the newly created user must be blocked from logging into Epic.
EndDate	The date the user was terminated or left the organization.
DefaultLoginDepartmentID	By default, when the user logs into Epic, he is presented with this department.  Departments are customer-specific, hence it is required to work with your customers to determine the appropriate values.
LinkedProviderID	Every user linked to each provider.
LinkedSubtemplateIDs	User is assigned to Subtemplates which will be treated as Groups.  Sub-templates are used to provide modular configuration for many users and are highly configurable. You need to work with your customer to determine the appropriate values. Sub-templates with a lower index have priority.
LinkedTemplateID	User assigned to linked templates which will be treated as Groups.  Templates are used to provide modular configuration for many users and are highly configurable. You need to work with your customer to determine the appropriate values.

## Schema Attributes

Attribute name	Description
AuthenticationConfigurationID	A non-native authentication method (for example, LDAP) used to authenticate when user logs into Epic.
UserRoles	User Roles will not be treated as Entitlement.
ExternalIdentifiers	Manage user identity in multiple systems
CustomUserDictionaries	Maintain their own spell check corrections.
InBasketClassifications	Determines the messages the user receives in Epic.
Notes	Text notes about the user.
ContactComment	Comment associated with the creation of this user.
ContactDate	The date user was created. Defaults to the current date if not provided.
UserDictionaryPath	The file path at which the custom user dictionary files can be found.
LDAPOverrideID	A string that can be provided to identify the user to the LDAP server in place of the SystemLogin.

## Group attributes

---

The following table lists the Group attributes:

Attribute name	Description
<b>Linked template attributes</b>	
LinkedTemplateID	User assigned to linked templates which will be treated as Group.  Templates are used to provide modular configuration for many users and are highly configurable. You need to work with your customer to determine the appropriate values.
LinkedTemplateName	Name of the linked template.
<b>Linked Subtemplates attributes</b>	
LinkedSubtemplateIDs	ID of the linked subtemplate.  Sub-templates are used to provide modular configuration for many users and are highly configurable. You need to work with your customer to determine the appropriate values. Sub-templates with a lower index have priority.
LinkedSubTemplateName	Name of the linked subtemplate.

# Provisioning Policy attributes

---

The following table lists the provisioning policy attributes for Create Account:

Attribute name	Description
Name	The Epic user's name is in LastName, FirstName, MI format.
UserID	User ID for the newly created user. If provided, it will create user with specified ID else Epic will assign the ID automatically.
Password	The user's password.
Disabled	Defaults to false.
DefaultLoginDepartmentID	Represents the department of the user. For example, <b>INITIAL DEPARTMENT</b>
StartDate	Defaults to the initial start date.
EndDate	End date of the user account.
SystemLoginID	Unique name of the users operating system login. The maximum length is 254 characters.
Notes	Free text notes about the user.
ContactComment	A comment associated with the creation of the user.
LDAPOverrideID	A string that can be provided to identify the user to the LDAP server in place of the SystemLogin.
UserDictionaryPath	The file path at which custom user dictionary files can be found.
AuthenticationConfigurationID	If a non-native authentication method is used authenticate user when he logs into Epic.
InBasketClassifications	In Basket Classifications help determine the messages the user receives in Epic.
CustomUserDictionary_index_0	A number that indicates the priority of the value. Smaller numbers override greater numbers.
CustomUserDictionary_value_0	The string being stored at the indexed position.
CustomUserDictionary_index_1	A number that indicates the priority of the Value. Smaller numbers override greater numbers.
CustomUserDictionary_value_1	The string being stored at the indexed position.
ExternalIdentifier_id_0	The external ID to be set for this user.
ExternalIdentifier_type_0	The type of this ID - for the type of system it is valid.
ExternalIdentifier_isActive_0	True if this ID must be marked as active, that is, if the user can use it in the external system; else false.
ExternalIdentifier_id_1	The external ID to be set for this user.
ExternalIdentifier_type_1	The type of this ID - that is, for what kind of system it is valid.

## Troubleshooting

Attribute name	Description
ExternalIdentifier_isActive_1	True if this ID must be marked as active, that is, if the User can use it in the external system; else false.

**Note:** If required user can add multiple values for CustomUserDictionary and ExternalIdentifier fields in Provisioning Policy Plan, that is for additional values of Custom User Dictionary, user can add 4 attributes as follows (applicable for ExternalIdentifier attribute also):

- CustomUserDictionary\_index\_2
- CustomUserDictionary\_value\_2
- CustomUserDictionary\_index\_3
- CustomUserDictionary\_value\_3

## Troubleshooting

---

### 1 - If you create user policy from IdentityIQ, it fails with an error message

If create user policy from IdentityIQ fails, the following error message is displayed:

The [action] cannot be processed at the receiver.

**Resolution:** Ensure that the format of the Name is as follows:

LastName,FirstName

### 2 - While executing any operations in IdentityIQ, either of the following error messages are displayed

While executing any operations in IdentityIQ, either of the following error messages are displayed:

`java.security.InvalidAlgorithmParameterException`: the trustAnchors parameter must be non-empty

OR

`sun.security.validator.ValidatorException`: PKIX path building failed:

`sun.security.provider.certpath.SunCertPathBuilderException`: unable to find valid certification path to requested target

**Resolution:** Configure the certificates correctly.

### 3 - When the Aggregation fails an error message appears

Axis2 uses `temp` directory when the tomcat starts and if it is not accessible to axis, the following error message appears:

`Exception extracting jars into temporary directory: java.io.IOException: Permission denied: switching to alternate class loading mechanism`

**Resolution:** Ensure that the `temp` directory in tomcat home directory is not deleted. If the `temp` directory exists and if the error messages are still displayed, ensure that you have the access permissions.

### 4 - Create user request fails with an exception

Create user request fails if Linked Template and User Role are requested in the same request.

**Resolution:** Requesting Linked Template and User Role in the same request are not supported for some versions of Epic. To avoid this failure, request Linked Template or User Role in update request.

## 5 - Remove entitlement fails with an exception

Removing last entitlement from any entitlement attribute for a user has limitation in Epic version 2010.

**Resolution:** This issue can be resolved by requesting access for other entitlement (for same entitlement attribute) in the same request that requests removing the last entitlement. In such case Epic considers it as a replace operation instead of remove and replaces the entitlement to be removed with the entitlement to be added.

## 6 - An error message appears if the core service is not enabled

If the Core service is not enabled the following error message appears in the interface or log file:

```
ApplicationFault:<Type>FacadeServiceDisabled</Type>
```

The requested business service is disabled.

**Resolution:** Enable the Core web services on the Epic web services server.

## 7 - For JBoss EAP server, test connection fails with an error message

The following error message appears when the test connection fails for JBoss EAP Server:

```
Exception while connecting to Personnel service
```

**Resolution:** Copy the addressing-1.6.1.mar file from \\WEB-INF\\lib\\ directory to deployment directory of JBoss (for example, jboss-eap-6.2\\standalone\\deployments) in order to work with certificate based authentication on JBoss.

Provide the path to MAR files as a parameter while starting JBOSS EAP server (for example, standalone.bat -Daxis2.repo=\\jboss-eap6.2\\standalone\\deployments\\addressing-1.6.1.mar)

## 8 - Not able to generate SOAP Envelop logging in Epic connector

When performing any operation, not able to generate SOAP Envelop logging in Epic Connector.

**Resolution:** To enable advanced SOAP Envelop logging in Epic connector configure the following attribute in xml application schema:

```
<entry key="logSOAPEnvelop" value ="true" />
```

**Note:** Download the sailpoint\_epic\_connector\_axis2.xml file from IdentityIQ.zip/integration directory and copy it into identityiq\\WEB-INF\\classes directory in order to generate SOAP logs.

## 9 - Account Aggregation Task enters into an endless loop

Account Aggregation Task enters into an endless loop when GetRecords API enters into endless loop.

**Resolution:** To avoid the **GetRecords API** call getting into an endless loop, a **GetRecordsCallsThreshold** parameter is used. The default value of **GetRecordsCallsThreshold** is 5000. To increase the count of **GetRecordsCallsThreshold**, enter the following key in Epic application xml:

```
<entry key="getRecordsCallsThreshold" value="value" />
```

## **Troubleshooting**

# Chapter 61: SailPoint GE Centricity Connector

---

The following topics are discussed in this chapter:

Overview .....	535
Supported features .....	536
Prerequisites .....	536
Administrator permissions .....	536
Configuration parameters .....	536
Additional configuration parameters .....	537
Schema attributes .....	537
Account attributes .....	537
Group attributes .....	538
Provisioning Policy attributes .....	538
Troubleshooting .....	539

## Overview

---

**Note: SailPoint will provide direct collaboration and oversight during the deployment of this Connector.**

Centricity is a brand of 27 healthcare IT software solutions from GE Healthcare, a division of General Electric. It includes software for independent physician practices, academic medical centres, hospitals and large integrated delivery networks. The various modules perform practice management, revenue cycle management, electronic medical records, medical imaging, and other functions.

GE Healthcare acquired IDX Systems in 2006 and re-released its products under the Centricity brand. The GE-branded solution formerly produced by IDX acts as an interface engine for receiving and exchanging HL7 and other types of records.

SailPoint GE Centricity Connector manages GE Centricity accounts, roles/rights, applications and user system. The connector manages GE centricity Web portal interface. Changes to the character cell interface must be done manually.

## Supported features

---

SailPoint GE Centricity Connector supports the following features:

- Account Management
  - Manage GE Centricity accounts as Accounts
  - Aggregate, Refresh Account, Pass Through Authentication
  - Create, Update
  - Enable, Disable
  - Change Password - Supported for change/reset password in IDX Web Portal. Need to manually synchronize password on IDX Backend Console,
  - Add/Remove Entitlements - While additional entitlements can be requested for UserSystems, UserApplications, and Roles, remove Entitlement is supported only for Roles.
- Account Group Management
  - Manage GE Centricity Roles as Account- Groups
  - Aggregate
- Permissions Management
  - Application reads permissions directly assigned to accounts and groups as direct permissions during account and group aggregation.
  - The connector supports automated revocation of the aggregated account permissions. Work items are created for requests to revoke group permissions.

## Prerequisites

---

The DataURL must be configured in the GE client application. The URL (IP Address and Port) with Username and password is required for using the connector.

No third party jars are required.

## Administrator permissions

---

The GE IDX Connector administrator should have **Administrator - View all, edit users** role.

## Configuration parameters

---

The following table lists the configuration parameters of GE Centricity Connector:

Parameters	Description
GECentricity URL	Specifies the data url containing host and port of GE Centricity instance you are using.

Parameters	Description
Username	Specifies the administrator or the name of the user which has Admin level privileges to perform aggregation and provisioning operation on GE Centricity system.
Password	Defines the password of the username.
Manage Active Accounts Only	If selected will aggregate only active accounts.

## Additional configuration parameters

---

Attribute	Description
columnDelimiter	Column separator for multi column attribute value. For example, UserApplications. Default: #.
requireForceChangePswd Flag	Indicates whether force change password should be turned on for helpdesk password reset. Default: Y.
skipAppUserPswdChange	Defined when change of password for all user system applications is not required. Use the following format to configure in the application:  <code>&lt;entry key="skipAppUserPswdChange" value="true" /&gt;</code>
skipSetPswdForAppUserStartsWith	Lists the system application usernames which should not override the original password.  For example, the usernames like <b>msi</b> , <b>msitrain</b> which need to be skipped. If the attribute is not added in the configuration, it will update all the user applications with new password. The following format is used to be configure in the application:  <code>&lt;entry key="skipSetPswdForAppUserStartsWith" value="msitest,msitrain" /&gt;</code>

## Schema attributes

---

This section describes the different schema attributes.

### Account attributes

---

The following table lists the account attributes:

Attributes	Description
Key	By default, this attribute is the connectors default nativeIdentity and display name attributes. It's the user ID. For example, SailPoint
Name	User's full name
Email	User's department

## Provisioning Policy attributes

Attributes	Description
Password	User's password
Roles	List of Roles assigned to user.
UserSystems	User's system connections. It is multivalued attribute with SystemName, SystemID, and MenuKey.
UserApplications	User's application connections. It is multivalued attribute. It will consist of SystemId, SystemName, ApplicationID, ApplicationName and Username.
PasswordExpired	Flag that indicates if the user's password expired
PasswordFailures	Integer value indicates of user's failure attempts
PasswordLastChanged	Date timestamp of user's last password changed.
PasswordNeverExpires	Flag indicates the user's neverexpires password.
LastLogin	Date timestamp of user's last login.
Authentication	Authentication of which CF have been used.
DefaultRoleKey	Default Role of user's in CF application.
DefaultSystemID	Default System of user's in CF application.
LastFailedLogin	Date timestamp of user's last failed login.
directPermission	List of Rights assigned to User.

## Group attributes

Roles are aggregated during account group aggregation, following are the attributes returned by the group aggregation process. Right API brings rights as well as roles. Roles have TypeID as **H** while for Rights,TypeID is **L**.

Attributes	Description
key	By default, this attribute is the connectors default Group Attribute It's the role Id.
Name	Name of the Role in CF application.
ProductID	Name of the Product mapped to Role in CF application.
ApplicationID	Name of the Application mapped to Role in CF application.
VTBMenuKey	Menukey which is mapped to each Role.
ID	Represents the unique Role of the IDX.
Description	Shows detail description of Roles.
directPermissions	The rights will be treated as directPermission for Role (Group) entity.

## Provisioning Policy attributes

The SailPoint GE Centricity Connector has a default Provisioning Policy defined which allows creation of accounts. The provisioning policy can be edited to fit specific customer environments.

Attributes	Description
<b>Create user policy</b>	
Key	The userid of the GE centricity application
password	The user's password.
Disabled	<b>False</b> (Default): Will show active user <b>True</b> : Will show inactive user
Name	Defaults to the identity's Lastname/FullName/FirstName.
Email	Defaults to the identity's department.
Force Password Change	Indicates the last changed password of identity.
Password Never Expires	Defaults to identity's password NeverExpire Flag.
Default System	Defaults to UserSystems .
Default Role	Defaults to Role/rights of identity.

## Troubleshooting

---

### 1 - If Authentication field is added on create/update user policy, the API displays an error message

If Authentication field is added on create/update user policy, the API displays the following error message:

Error updating into table "Users"

**Resolution:** Remove Authentication field from policies.

### 2 - Change/Reset password works from IdentityIQ, but not able to login in GE IDX

Change/Reset password works if the password is changed from GE Web console and IDX Backend Console (character Cell Interface). The IdentityIQ change/reset password calls the WebAPI, hence the password is changed for GE Web console. By the time the password is synchronized with IDX Backend Console, IdentityIQ stops working.

**Resolution:** After the password is changed and the user is able to login, synchronize the password in specific User System applications.

### 3 - When creating user from IdentityIQ/Web console, user is unable to login through Web portal of GE

When creating user from IdentityIQ/Web console, user is unable to login through Web portal of GE.

**Resolution:** Create user in IDX character cell interface.

## **Troubleshooting**

# Appendix

This section contains information on the following:

- "A: Delta Aggregation" on page 543
- "B: Component Interface" on page 551
- "C: Partitioning Aggregation" on page 563
- "D: Before and After Provisioning Action" on page 571
- "E: IQService" on page 577



# Appendix A: Delta Aggregation

---

This appendix describes the following information.

Overview.....	543
Delta aggregation for Microsoft Active Directory, ADAM, SunOne and Tivoli .....	543
Configuring server for Delta Aggregation.....	545
Testing Delta Aggregation .....	545
Delta aggregation for JDBC .....	546
Delta aggregation for Lotus Domino .....	547
Delta aggregation for SAP .....	547

**Note:** For Delta Aggregation of SAP HR/HCM Connector, see ‘Delta Aggregation’ section of “Chapter 23: SailPoint SAP HR/HCM Connector”.

## Overview

---

Delta aggregation can be requested by checking a box on the task definition that is then passed through to the connector. If the connector does not support delta aggregation, then it ignores this flag and performs normal aggregation. The connectors supporting delta aggregation uses various mechanisms depending on the managed system to read the changes that have taken place after certain benchmark. It can be `lastModData`, `usnChanged`, or so on, else that indicates the last aggregation benchmark. This marker is stored on the application. Hence to take advantage of delta aggregation at least one full aggregation is required which will allow the connector to store the starting point for next delta aggregation.

If the volume of changes are more than 40% of the total data on the server, a normal aggregation run is recommended before over delta aggregation. The delta aggregation run can be scheduled at suitable interval considering the amount of changes happening on the managed system.

## Delta aggregation for Microsoft Active Directory, ADAM, SunOne and Tivoli

---

1. Delta aggregation is supported for the following directory server types:

- Active Directory - Direct
- ADAM - Direct
- SunOne - Direct
- IBM Tivoli DS - Direct

**Note:** This includes changes such as user/group has been added/updated/deleted on the managed system. This version now supports aggregation of delta changes for Move and Rename operations.

2. Prerequisite for delta aggregation:

After creating a fresh version of IdentityIQ application of type ADAM - Direct, Active Directory - Direct, SunOne - Direct and IBM Tivoli DS - Direct or after an upgrade to latest version of IdentityIQ, open the application configuration file in debug mode and ensure that the GROUPS\_HAVE\_MEMBERS feature string

## **Delta aggregation for Microsoft Active Directory, ADAM, SunOne and Tivoli**

have been added in respective Group schema. Ensure that the following entries exist for respective application types:

- (*For Active Directory - Direct*): Ensure that the following attribute is added under Group schema:  
`<entry key='groupMemberAttribute' value='member' />`

For **DirSync** delta aggregator of Active Directory, user must be a member of Domain administrators group or must have **Read and Replicating directory changes** permissions along with read permissions on **Deleted objects container**.

To provide **Replicating directory changes** permissions to the user, perform the following actions:

- In the Active Directory Users and Computers browser menu, select the **View** option, right-click and ensure that **Advanced features** check box is enabled.
- Right-click the domain node and select **property** option and open the the Security tab.
- Add user to the list of Security Principals.
- Select the user and select **Allow** checkbox for Replicating Directory Changes permission.

To provide **Read** permissions on **Deleted Objects Container** to user, perform the following actions:

- Log on to any domain controller in the target domain with a user account that is a member of the Domain Administrators group.
- Open a command prompt: navigate to Start, enter cmd and click **Enter**. Enter the following command and press **Enter**:

```
dsacls <Deleted objects container DN> /<takeownership>
```

In the above command line, Deleted objects container DN is the distinguished name of the deleted objects container.

- For example, `dsacls "CN=Deleted Objects,DC=SailPoint,DC=Com" /takeownership`
- To grant **Read** permission to the objects in the **Deleted Objects container** to a user type, enter the following command and press **Enter**:

```
dsacls < Deleted objects container DN > /G <domainName\userName >: LCRP
```

In the above command line, LCRP stands for the list object and read properties permission.

For example, `dsacls "CN=Deleted Objects, DC=SailPoint,DC=Com" /G  
Sailpoint\John:LCRP`

(*For ADAM - Direct*): `<entry key='groupMemberAttribute' value='member' />`

For existing Active Directory - Direct or ADAM - Direct applications, add the following key into the applications configuration file. For new applications, modify the following key:

```
<entry key="deletedObjectsContainer" value="CN=Deleted Objects,DOMAIN"/>
```

Where DOMAIN is a place holder for the naming context where the account and accountgroup objects reside. Replace DOMAIN with the corresponding naming context.

For example,

```
<entry key="deletedObjectsContainer" value="CN=Deleted  
Objects,dc=sailpoint,dc=com"/>
```

- (*For SunOne - Direct and IBM Tivoli Directory Server - Direct*): `<entry key='groupMemberAttribute' value='uniqueMember' />`

## Configuring server for Delta Aggregation

---

The mechanism used under the hood for Delta Aggregation are:

- (*For Active Directory*) The following delta aggregation modes are supported for Active Directory:

- **uSNChanged**: Based on uSNChanged attribute of Active Directory.
- **DirSync**: Based on DirSync feature of Active Directory.

Select one of the above mentioned types of delta aggregators from the **Delta Aggregation Mode** application configuration attribute.

(*For ADAM*) The following delta aggregation mode is supported for ADAM:

- **uSNChanged**: Based on uSNChanged attribute of ADAM.

**Note:** A full aggregation is required after selecting the ‘Delta Aggregation Mode’ application configuration attribute.

- (*For SunOne/Tivoli*) changeLog

Following sections describe how to configure SunOne and Tivoli directory servers for delta Aggregation.

**Note:** After enabling the changelog on directory server, run Account and Account-Group full aggregation task before running delta aggregation.

## Configuring SunOne directory server for Delta Aggregation

1. Locate the `dsconf` command of the SunOne directory server installation.
2. Using the command prompt execute the following command:  
`> dsconf set-server-prop --unsecured -h <host> -p <non ssl port>  
retro-cl-enabled:on retro-cl-deleted-entry-attr:nsUniqueId`
3. Enter the password for the directory server administrator.
4. Restart the server.

## Configuring IBM Tivoli directory server for Delta Aggregation

1. Stop the Tivoli Directory server instance.
2. Locate the `idscfgchglg` for your Tivoli Directory Server installation.
3. To configure a change log for directory server instance, run the following command:  
`idscfgchglg -I <Tivoli instance> -m 0`
4. Start the directory server instance.

**Note:** Confirm the server has been enabled for changelog, open a ldap browser and bind it to the ldap server instance and view the cn=changelog naming context. You should be able to see this naming context and the relevant change objects. Ensure this before you proceed with delta aggregation for SunOne and Tivoli directory servers.

## Testing Delta Aggregation

---

For delta aggregation to work properly, it needs a start point from where it would detect changes. To retrieve changes from the last iteration, it needs to first perform a full aggregation during which it maintains its reference point. Once the full aggregation completes, you may create a separate delta aggregation task to retrieve delta changes that occurred post the full aggregation.

## Delta aggregation for JDBC

Perform the following steps to test delta Aggregation:

1. Execute Account and Account - Group Aggregation task.
2. Create a task with delta aggregation flag set for Account and Account - Group Aggregation.
3. Perform Create/Update/Delete/Revoke operations for Accounts/Groups on the directory server.
4. Execute the respective delta aggregation task.
5. Confirm the changes have been retrieved into IdentityIQ.

**Note:** For SunOne-Direct and Tivoli-Direct applications, the delta aggregation task would fail even though the full aggregation is successful in case if the server has not been configured for changelog. Hence, before performing full aggregation ensure the changelog has been configured for the directory server.

## Delta aggregation for JDBC

---

1. Create two tables to capture the identities whose data is modified in the master tables:
  - One table for capturing the account whose attributes, entitlements, or direct permissions are modified in the master table for account, master table for its entitlements, or master table for its direct permissions respectively
  - Another table for capturing the account-group whose attributes are modified in the master table for account-group
2. Each of the two tables must contain two columns such that
  - the first column will store the identity attribute defined in IdentityIQ and
  - the second column will store the action. Values of action can be Insert, Update, or Delete.

(For Oracle database) For example, the SQL to create such a table

```
CREATE TABLE USER_DELTA(USER_ID VARCHAR2(20), ACTION VARCHAR2(10));
```
3. Assign the privileges to read and delete the records from the tables created in step 1. above to the connection user defined in the application configuration.
4. Create the triggers on the following master tables:
  - Account table
  - Entitlements table for account
  - Permissions table for account
  - Account-Group table

The triggers on the Account table, the Entitlements table for the Account, and the Permissions table for the Account would write the Account Identity attribute, in the first table created in step 1., whose attributes, entitlements, or permissions have changed in the master tables. Similarly, the trigger on the Account-Group table would write the Account-Group Identity attribute, in the second table created in step 1., whose attributes have changed.

(For Oracle database) For example, the following trigger writes the user IDs in the first table, that have undergone some modifications, along with the respective action:

```
CREATE OR REPLACE TRIGGER T1
AFTER DELETE OR INSERT OR UPDATE ON USER_MASTER
FOR EACH ROW
BEGIN
```

```

IF INSERTING THEN
    INSERT INTO USER_DELTA (USER_ID, ACTION)
        VALUES (:NEW.USER_ID, 'Insert');
END IF;
IF UPDATING THEN
    INSERT INTO USER_DELTA (USER_ID, ACTION)
        VALUES (:NEW.USER_ID, 'Update');
END IF;
IF DELETING THEN
    INSERT INTO USER_DELTA (USER_ID, ACTION)
        VALUES (:OLD.USER_ID, 'Delete');
END IF;
END;
/

```

5. Once the above-mentioned steps are performed, the tables would start capturing the user or group IDs that have undergone changes in their master tables.
6. After the delta aggregation is finished, the tables would be reset/re-initialized and would start capturing the delta afresh.

## Delta aggregation for Lotus Domino

---

Delta Aggregation is supported for SailPoint Lotus Domino Connector. On Full Aggregation, the respective time and date values of account and group aggregation are stored in the Application object which are used by Delta Aggregation to retrieve the changed data into IdentityIQ.

### Pre-requisites

---

IQService is required for Delta Aggregation.

## Delta aggregation for SAP

---

This section describes the procedure for configuring the SAP Connector for Delta Aggregation.

## Supported attributes

---

The SAP Direct Connector supports Delta Aggregation for the following attributes:

- User created
- User deleted
- Password
- User Type
- Administrator lock set // IdentityIQ Disabled
- Administrator lock released // IdentityIQ Enabled
- Incorrect logon lock set // IdentityIQ Locked
- Incorrect logon lock released // IdentityIQ Unlocked
- Validity Period
- Account Number
- User Group
- SAP Profile(s) Assigned
- SAP Profile(s) Deleted
- Security Policy

## Importing the BAPI's

---

This section describes how to import the transport request that contains the non - certified function modules used by SAP Connector to achieve the delta aggregation functionality.

Function modules are imported using one of the following methods:

- Using the Transport Control program manually
- Using the menu-driven administration of Transport Requests via SAP GUI

### Pre-requisites

Copy the API Transport files to SAP. Use the following procedure to unpack and import the transport files function modules:

1. Copy the transport files.

Transport request is contained in the `ImportsAPIDirect.TAR` compressed file.

The compressed file for each release contains the following files:

- `RrequestNumber.sapId`
- `KrequestNumber.sapId`

Using WinZip or a similar utility, uncompress and copy each file from the appropriate compressed file to the subdirectory of the local transport directory of the target SAP system as follows:

- Copy the `RrequestNumber.sapId` file to the  
`sapHomeDir\trans\data\RrequestNumber.sapId` directory.
- Copy the `KrequestNumber.sapId` file to the  
`sapHomeDir\trans\cofiles\KrequestNumber.sapId` directory.

**Note:** The values of `requestNumber` and `sapId`. These values are required later.

## Using the Transport Control Program manually

- At the command prompt (for both Microsoft Windows or UNIX systems), enter the following command to register the transport with the buffer:

```
os prompt> tp addtobuffer sapIdKrequestNumber yourSid
```

In the preceding command line, *sapId* and *requestNumber* were determined in step 1 of “ Pre-requisites”.

- Enter the following command to import the above transport request in to your SAP Solutions system:

```
os prompt>tp import sapIdKrequestNumber yourSid client=yourClient U126
```

After executing the `tp import` command, the system issues a return code, indicating the status of the import. The most common return codes are described in the table below:

**Table 1—Transport Request Import - Common Return Codes**

Return Code	Description
0	Successful import. The Transport imported successfully.
4	Warning status. Minor version differences were detected. The Transport Imported successfully. (No action is required.)

## Importing the Transport via SAP GUI

If you are running SAP GUI, use the following procedure for importing the function modules:

- Log in to SAP GUI with administrator permissions.
- Perform one of the following:
  - From the Command field, run transaction STMS

OR

  - From the menu, select **Tools => Administration => Transports => Transport management system**.
- In the Transport Management System window, press **F5**.
- In the Import Overview window, double-click your system queue.  
The requests list for the system is displayed.
- From the menu bar in the Import Queue window, select **Extras => Other requests => Add**.
- Enter the request number and click **Yes**.
- In the Attach to import queue message box, click **Yes**.
- In the Import Queue window, click on the new line, and then press **Ctrl + F11** to import the request.
- In the Import Transport Request dialog box, perform the following:
  - In the Target client field, enter the name of the SAP client to which you want to import the transport.
  - On the Date tab, under Start Date, set the values that you require.
  - On the Execution tab, under Import, ensure that **Synchronous** is selected.
  - On the Options tab, under Import options, ensure that all of the check boxes are selected.
- In the Start Import dialog box, click **Yes**.

## **Verification**

---

All non-validated function modules are transported as a function group whose name starts with SailPoint's unique namespace ("SAILPOIN")

To verify that the required function modules have been imported in to the SAP system, perform the following:

1. In the SAP system, execute **SE37** transaction.
2. Enter: /SAILPOIN/\* in the Function Module field.
3. Press the **F4** key.

This displays the Repository Info System window that lists all the imported functional modules currently available on the SAP system as follows:

- SAILPOIN/USR\_CHANGE\_DOC\_ROLES
- SAILPOIN/USR\_CHANGE\_DOC\_USERS

# Appendix B: Component Interface

---

This appendix describes the following information.

Creating component interface for Peoplesoft financials .....	551
Basic structure of Custom Component (CI) from USERMAINT component for Users .....	551
Basic structure of Custom Component (CI) from ROLEMAINT component for Roles.....	556
Basic structure of Custom Component (CI) from PURGE_USR_PROFILE component for Delete User ..	558
Basic structure of Component Interface (CI) from PURGE_ROLEDEFN component for Delete Role ..	559
Deleting the component interface.....	560

## Creating component interface for Peoplesoft financials

---

This section describes the procedure for creating the basic structure of a new Component Interface (CI) for Peoplesoft financial from USERMAINT and ROLEMAINT components.

### Basic structure of Custom Component (CI) from USERMAINT component for Users

---

This section describes the creation of basic structure of CI from USERMAINT component, changing GETKEYS, FINDKEYS and CREATEKEYS for the newly created CI, and verification of the newly created CI's.

#### Creating CI

1. Log on to Application Designer and click on **File => New**.  
A new dialog box named New Definition is displayed.
2. Select **Component Interface => OK**.  
A new dialog box named Select source Component for Component Interface is opened.
3. Enter the name as **USERMAINT** under the **Selection Criteria** tab and click **Select**.  
A dialog box appears with the following message:  
Do you want to default the properties based on the underlying component definition
4. Click **Yes**.

## Creating component interface for Peoplesoft financials

Following screen shot appears:

Name	Record	Field	Re...	Co...
PSWDEXPIRED	PSUSRPRFL...	PSWDEXPIRED		
ROLEUSER_ASSIGN...	PSUSRPRFL...	ROLEUSER...		
ROLEUSER_REASSI...	PSUSRPRFL...	ROLEUSER...		
TOTAL_ROW_COUNT	PSUSRPRFL...	TOTAL_ROW...	Y	
OPRALIAS	PSOPRALIAS...	OPRALIAS...		
OPRALIAS_TYPE	PSOPRALIAS...	OPRALIAS...		
ATTRNAME	PSOPRALIAS...	ATTRNAME	Y	
ATTRVALUE	PSOPRALIAS...	ATTRVALUE		
DESCR	PSOPRALIAS...	DESCR	Y	
POPTIONS	PSOPTIONS	POPTIONS		
FROM_DIR_RULE	ROLEDYNUSR...	FROM_DIR_R...		
FROM_PCODE...	ROLEDYNUSR...	FROM_PCODE...	Y	
FROM_QUERY...	ROLEDYNUSR...	FROM_QUER...		
ROLENAME	ROLEDYNUSR...	ROLENAME	Y	
PSROLEUSER_VW	PSROLEUSER...	PSROLEUSER...		
DYNAMIC_SW	PSROLEUSER...	DYNAMIC_SW	Y	
ROLENAME_1	PSROLEUSER...	ROLENAME		
RUN_CNTLLDAP	RUN_CNTLLDAP	RUN_CNTLLDAP		
LDAPMAPNAME	RUN_CNTLLDAP	LDAPMAPNA...		
PROCESS_INST...	RUN_CNTLLDAP	PROCESS_IN...		
RUN_CNTL_ID	RUN_CNTLLDAP	RUN_CNTL_ID		
PSUSEROTHER_VW	PSUSEROTH...	PSUSEROTH...		
DESCR100	PSUSEROTH...	DESCR100	Y	
METHODS				
Cancel				
Create				
Find				
< >				

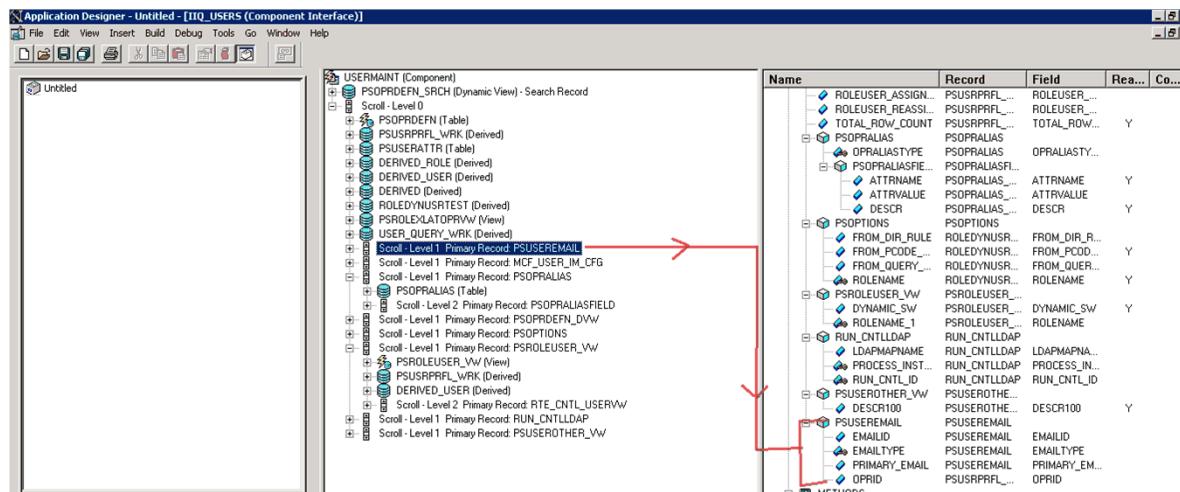
- Click on File => Save As.

A new dialog box appears requesting for the name of the CI as follows:

Name	Record	Field	Re...	Co...
IIQ_USERS	PSOPRDEFN_S...	OPRID		
GETKEYS	PSOPRDEFN_S...	OPRDEFNDE...		
FINDKEYS	PSOPRDEFN_S...	OPRDEFNDE...		
CREATEKEYS	PSOPRDEFN_S...	OPRID		
PROPERTIES				
SERVERNAME	DERIVED_ROLE	SERVERNAME		
ACCTLOCK	PSOPRDEFN	ACCTLOCK		
CURRENCY_CD	PSOPRDEFN	CURRENCY...		
DEFAULTNAVHP	PSOPRDEFN	DEFAULTNAV...		
EXPERT	PSOPRDEFN	EXPERT		
LANGUAGE_CD	PSOPRDEFN	LANGUAGE_CD		
LASTUPDDTM	PSOPRDEFN	LASTUPDT...	Y	
LASTUPDOPRIN	PSOPRDEFN	LASTUPDOP...	Y	
OPERPWD	PSOPRDEFN	OPERPWD		
OPRCCLASS	PSOPRDEFN	OPRCCLASS		
OPRDEFNDESC	PSOPRDEFN	OPRDEFNDE...	Y	
OPRID	PSOPRDEFN	OPRID		
PRCSPPRFCLS	PSOPRDEFN	PRCSPPRFCLS		
POLLOWSWITCHU...	PSOPRDEFN	POLLOWSW...		
ROSESECCLASS	PSOPRDEFN	ROSESECCLA...		
SYMBOLID	PSOPRDEFN	SYMBOLID		
USERALIAS	PSOPRDEFN	USERALIAS		
EFFDT_FROM	PSROLELATO...	EFFDT_FROM		
EFFDT_TO	PSROLELATO...	EFFDT_TO		

- Enter the name of the CI as {NEW\_Name}. For example, IIQ\_USERS.
- Drag the **Scroll-Level1 Primary Record: PSUSERMAIL** from source component (USERMAINT) to the properties of the newly created CI {NEW\_Name}. For example, IIQ\_USERS.

After dragging and dropping the Scroll-Level1 Primary Record: PSUSERMAIL attribute, a new property is listed in the PROPERTIES of the newly created CI.



## Changing GETKEYS, FINDKEYS and CREATEKEYS for the newly created CI

1. Expand **FINDKEYS** and click on **OPRID**. Right click on **OPRID** and select **Edit Name** to change the attribute name to **UserID**.

Name	Record	Field	Rea...	Co...
IIO_USERS				
<b>FINDKEYS</b>				
OPRID	PSOPRDEFN_S...	OPRID		
OPRDEFNDESC	PSOPRDEFN_S...	OPRDEFNDE...		
<b>CREATEKEYS</b>				
OPRID	PSOPRDEFN_S...	OPRID		
<b>GETKEYS</b>				
OPRID	PSOPRDEFN_S...	OPRID		
<b>PROPERTIES</b>				
OPRID	PSOPRDEFN	OPRID	Y	
OPRDEFNDESC	PSOPRDEFN	OPRDEFNDE...	Y	
OPRCLASS	PSOPRDEFN	OPRCLASS		
ROWSECLASS	PSOPRDEFN	ROWSECLASS...		
OPERPSSWD	PSOPRDEFN	OPERPSSWD		
SYMBOLOID	PSOPRDEFN	SYMBOLOID		
LANGUAGE_CD	PSOPRDEFN	LANGUAGE_CD		
CURRENCY_CD	PSOPRDEFN	CURRENCY_CD		
ACCTLOCK	PSOPRDEFN	ACCTLOCK		
PRCSPRFCLCS	PSOPRDEFN	PRCSPRFCLCS		
DEFAULTNAVHP	PSOPRDEFN	DEFAULTNAV...		
EXPERT	PSOPRDEFN	EXPERT		
USERIDALIAS	PSOPRDEFN	USERIDALIAS		
LASTUPDDTTM	PSOPRDEFN	LASTUPDDT...	Y	
LASTUPDOPRID	PSOPRDEFN	LASTUPDOP...	Y	
PTALLOWSWITCHU...	PSOPRDEFN	PTALLOWSW...		
OPERSWDCONF	PSUSRPRFL...	OPERSWDC...		
TOTAL_ROW_COUNT	PSUSRPRFL...	TOTAL_ROW...	Y	
ROLEUSER_REASSI...	PSUSRPRFL...	ROLEUSER...		

Similarly change the name of **OPRDEFNDESC** attribute to **UserDescription**.

2. Expand **GETKEYS** and change the name of **OPRID** to **UserID**.
3. Expand **CREATEKEYS** and change the name of **OPRID** to **UserID**.
4. After changing the keys for **GETKEYS**, **FINDKEYS** and **CREATEKEYS**, change the **PROPERTIES**.
  - Changing Single attribute
    - a. Expand **PROPERTIES**.
    - b. Select the attribute and right click on **Edit Name** to change the name of the attribute. Provide the names mentioned in the following table for the respective attributes:

Original attribute name	Changed attribute name
OPRID	UserID
OPRDEFNDESC	UserDescription
OPRCLASS	PrimaryPermissionList
ROWSECCLASS	RowSecurityPermissionList
OPERPSWD	Password
SYMBOLICID	SymbolicID
LANGUAGE_CD	LanguageCode
CURRENCY_CD	CurrencyCode
ACCTLOCK	AccountLocked
PRCSPRFLCLS	ProcessProfilePermissionList
DEFAULTNAVHP	NavigatorHomePermissionList
EXPENT	ExpertEntry
USERIDALIAS	UserIDAlias
LASTUPDDTTM	LastUpdateDateTime
PTALLOWSWITCHUSER	AllowSwitchUser
OPERPSWDCONF	ConfirmPassword
TOTAL_ROW_COUNT	WorkListEntriesCount
ROLEUSER_REASSIGN	ReassignUserID
ROLEUSER_ASSIGN_SW	ReassignWork
NO_SYMBOLID_WARN	NoSymbolicIDWarning
PSWDEXPIRED	PasswordExpired
MPDEFAULMP	DefaultMobilePage
WORKLIST_USER_SW	WorkListUser
EMAIL_USER_SW	EmailUser
LASTUPDOPRID	LastUpdateUserID
ROLEUSER_ALT	AlternateUserID
ROLEUSER_SUPR	SupervisingUserID
EFFDT_FROM	EffectiveDateFrom
EFFDT_TO	EffectiveDateTo
CHANGE_PWD_BTN	ChangePassword
	<b>Note: This attribute is applicable only for PeopleTools version 8.55 and above.</b>

- c. Delete the **SERVERNAME** property attribute.
- Changing collection attribute
  - a. Some attributes when expanded, have other attributes under them. Such attributes are called as collection attributes.

PSUSEREMAIL	PSUSEREMAIL	EMAILID
EMAILID	PSUSEREMAIL	EMAILTYPE
EMAILTYPE	PSUSEREMAIL	PRIMARY_EM...
PRIMARY_EMAIL	PSUSEREMAIL	OPRID
OPRID	PSUSRPRFL...	OPRID

- b. Select the collection attribute name => right click on the attribute => click on **Edit Name** and change the name of that attribute.
- c. Expand the Attribute. Change the internal Attribute names also in similar manner. The following table mentions the attribute names to be modified:

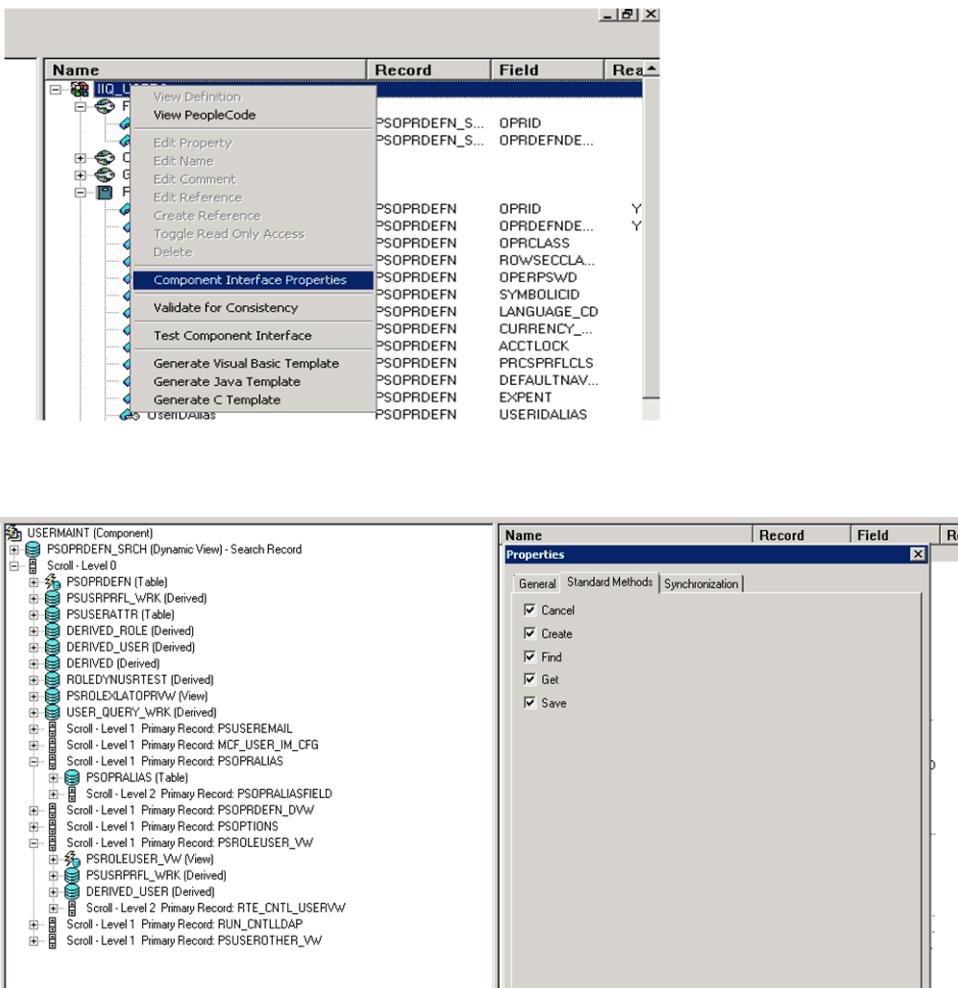
Original collection attribute name	Changed collection attribute name	Original child attribute name	Changed child attribute name
PSUSEREMAIL	EmailAddresses	EMAILID	EmailAddress
		EMAILTYPE	EmailType
		PRIMARY_EMAIL	PrimaryEmail
		OPRID	OPRID
PSOPRALIAS	IDTypes	OPRALIASTYPE	IDType
		PSOPRALIASFIELD	Attributes
		ATTRNAME	AttributeName
		ATTRVALUE	AttributeValue
		DESCR	DESCR
PSROLEUSER_VW	Roles	DYNAMIC_SW	Dynamic
		ROLENAMES_1	RoleName_1

- d. After renaming property attributes as mentioned in the above table, delete the following attributes:
  - PSOPTIONS
  - RUN\_CNTLLDAP
  - PSUSEROTHER\_VW

### Verification of the standard methods for the newly created CI

1. Right click on the name of the CI => click on **Component Interface Properties** => Click on **Standard Methods**.  
where CI is the Component Interface created as mentioned in “Creating component interface for Peoplesoft financials” on page 551.
2. Verify all properties (**cancel, create, find, get, save**) are selected.

## Creating component interface for Peoplesoft financials



The new CI is ready to be used. For example, **IIQ\_USERS**

## Basic structure of Custom Component (CI) from ROLEMAINT component for Roles

This section describes the creation of basic structure of CI from ROLEMAINT component.

### Creating CI

1. Log on to Application Designer and click on **File => New**.  
A new dialog box named New Definition is displayed.
2. Select **Component Interface => OK**.  
A new dialog box named Select Source Component for Component Interface is opened.
3. Enter the name as **ROLEMAINT** under the **Selection Criteria** tab and click **Select**.  
A dialog box appears with the following message:  
Do you want to default the properties based on the underlying component definition
4. Click **Yes**.

Following screen shot appears:

The screenshot shows the Oracle Database SQL Developer interface. On the left, there is a tree view of the schema structure under 'PSROLEDEFN [Component]'. The 'PSROLEDEFN\_SRCH (View) - Search Record' node is expanded, showing various tables and derived tables. On the right, a table titled 'Name' lists the columns of the 'PSROLEDEFN\_SRCH' view. The columns include 'NAME', 'Record', 'Field', 'Rea...', and 'Co...'. The table contains numerous entries, many of which have a small yellow 'Y' icon in the 'Co...' column.

Name	Record	Field	Rea...	Co...
GETKEYS				
ROLENAME	PSROLEDEFN...	ROLENAME		
FINDKEYS				
DESCR	PSROLEDEFN...	DESCR		
ROLENAME	PSROLEDEFN...	ROLENAME		
CREATEKEYS				
ROLENAME	PSROLEDEFN...	ROLENAME		
PROPERTIES				
USERID	DERIVED_DYN...	USERID		
DESCR	DERIVED_ROLE	DESCR		Y
SERVERNAME	DERIVED_ROLE	SERVERNAME		
ALLOWLOOKUP	PSROLEDEFN	ALLOWLOOK...		
ALLOWNOTIFY	PSROLEDEFN	ALLOWNOTIFY		
DESCR_0	PSROLEDEFN	DESCR		
FIELDNAME	PSROLEDEFN	FIELDNAME		
LASTUPDDTM	PSROLEDEFN	LASTUPD...		Y
LASTUPDOPRID	PSROLEDEFN	LASTUPDOP...		Y
LDAP_RULE_ON	PSROLEDEFN	LDAP_RULE...		
PC_EVENT_TYPE	PSROLEDEFN	PC_EVENT_T...		
PC_FUNCTION_NAME	PSROLEDEFN	PC_FUNCTION...		
ROLE_PCODE_RUL...	PSROLEDEFN	ROLE_PCODE...		
ROLE_QUERY_RUL...	PSROLEDEFN	ROLE_QUER...		
LDAP_RULE_ON	PSROLEDEFN	LDAP_RULE...		
ALLOWNOTIFY	PSROLEDEFN	ALLOWNOTIFY		
ALLOWLOOKUP	PSROLEDEFN	ALLOWLOOK...		
LASTUPDDTM	PSROLEDEFN	LASTUPD...		Y
LASTUPDOPRID	PSROLEDEFN	LASTUPDOP...		Y
SERVERNAME	DERIVED_ROLE	SERVERNAME		
DESCR	DERIVED_ROLE	DESCR		Y
ROLEUSER	PSUSRPRFL...	ROLEUSER		
USERID	DERIVED_DYN...	USERID		
PSROLECLASS	PSROLECLASS			
PSOPTIONS	POPTIONS			
RUN_CNTLLDAP	RUN_CNTLLDAP			
PSROLECANGRANT	PSROLECANG...			
PSROLEGRANTORVW	PSROLEGRAN...			
PSROLEOTHER_VW	PSROLEOTHE...			

- Click on **File => Save As**.

A new dialog box appears requesting for the name of the CI.

- Enter the name of the CI as **{NEW\_NAME}**. For example, **IIQ\_ROLES**.

- Delete the following collective attributes:

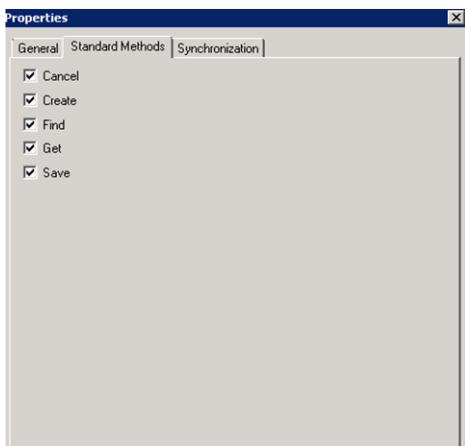
- **POPTIONS**
- **RUN\_CNTLLDAP**

The screenshot shows the Oracle Database SQL Developer interface. It displays a list of collective attributes. The 'RUN\_CNTLLDAP' attribute is highlighted with a red rectangular box.

PC_EVENT_TYPE	PSROLEDEFN	PC_EVENT_T...
QRYNAME_SEC	PSROLEDEFN	QRYNAME_S...
PC_FUNCTION_NAME	PSROLEDEFN	PC_FUNCTION...
ROLE_PCODE_RUL...	PSROLEDEFN	ROLE_PCODE...
ROLE_QUERY_RUL...	PSROLEDEFN	ROLE_QUER...
LDAP_RULE_ON	PSROLEDEFN	LDAP_RULE...
ALLOWNOTIFY	PSROLEDEFN	ALLOWNOTIFY
ALLOWLOOKUP	PSROLEDEFN	ALLOWLOOK...
LASTUPDDTM	PSROLEDEFN	LASTUPD...
LASTUPDOPRID	PSROLEDEFN	LASTUPDOP...
SERVERNAME	DERIVED_ROLE	SERVERNAME
DESCR	DERIVED_ROLE	DESCR
ROLEUSER	PSUSRPRFL...	ROLEUSER
USERID	DERIVED_DYN...	USERID
PSROLECLASS	PSROLECLASS	
POPTIONS	POPTIONS	
RUN_CNTLLDAP	RUN_CNTLLDAP	
PSROLECANGRANT	PSROLECANG...	
PSROLEGRANTORVW	PSROLEGRAN...	
PSROLEOTHER_VW	PSROLEOTHE...	

- Verification of the standard methods are selected for this newly created component Interface. For example, **IIQ\_ROLES**.

## Creating component interface for Peoplesoft financials



The newly created component interface is ready to be used. For example, **IIQ\_ROLES**.

## Basic structure of Custom Component (CI) from PURGEUSR\_PROFILE component for Delete User

This section describes the creation of basic structure of CI from Delete User component. For example, **IIQ\_DEL\_USER**

### Creating CI

1. Log on to Application Designer and click on **File => New**.  
A new dialog box named New Definition is displayed.
2. Select **Component Interface => OK**.  
A new dialog box named Select Source Component for Component Interface is opened.
3. Enter the name as **PURGEUSR\_PROFILE** under the **Selection Criteria** tab and click **Select**.  
A dialog box appears with the following message:  
*Do you want to default the properties based on the underlying component definition*
4. Click **Yes**.  
Following screen shot appears:

Name	Record	Field
OPRID_VW2	OPRID_VW2	OPRID
OPRDEFNDESC	OPRID_VW2	OPRDEFNDESC
OPRID	OPRID_VW2	OPRID
PRGUSR_PROFILE	PRGUSR_PROFILE	EMPLID
PRGUSR_PROFILE	PRGUSR_PROFILE	OPRDEFNDESC
PRGUSR_PROFILE	PRGUSR_PROFILE	OPRID
TBLSELECTION_VW	TBLSELECTION_VW	
RECDescr	TBLSELECTION_VW	RECDescr
RECNAME	TBLSELECTION_VW	RECNAME
ROLEUSR_TBLS_VW	ROLEUSR_TBLS_VW	RECDescr
RECDescr_1	ROLEUSR_TBLS_VW	RECNAME
RECNAME_1	ROLEUSR_TBLS_VW	RECNAME
CANCEL		
CREATE		
DELETE		
GET		
SAVE		

5. Click on **File => Save As**.  
A new dialog box appears requesting for the name of the CI.
6. Enter the name of the CI as **{NEW\_NAME}**. For example, **IIQ\_DEL\_USER**.  
Delete the following collective attributes:
  - TBLSELECTION\_VW
  - ROLEUSR\_TBLS\_VW

### Changing GETKEYS, FINDKEYS and CREATEKEYS for the newly created CI

1. Expand **FINDKEYS** and click on **OPRID**. Right click on **OPRID** and select **Edit Name** to change the attribute name to **UserID**.  
Similarly change the name of **OPRDEFNDESC** attribute to **UserDescription**.
2. Expand **GETKEYS** and change the name of **OPRID** to **UserID**.
3. After changing the keys for **GETKEYS** and **FINDKEYS** change the **PROPERTIES**.
  - Changing Single attribute
    - a. Expand **PROPERTIES**.
    - b. Select the attribute and right click on **Edit Name** to change the name of the attribute. Provide the names mentioned in the following table for the respective attributes:

Original attribute name	Changed attribute name
OPRID	UserID
OPRDEFNDESC	UserDescription

After the changes the CI would appear as follows:

Name	Record	Field	Read...
IIQ_DEL_USER			
FINDKEYS			
UserID	OPRID_VW2	OPRID	
UserDescription	OPRID_VW2	OPRDEFNDESC	
GETKEYS			
UserID	OPRID_VW2	OPRID	
PROPERTIES			
UserID	PRG_USR_PROFILE	OPRID	Y
UserDescription	PRG_USR_PROFILE	OPRDEFNDESC	Y
EMPLID	PRG_USR_PROFILE	EMPLID	Y
METHODS			
Cancel			
Find			
Get			
Save			

### Basic structure of Component Interface (CI) from PURGE\_ROLEDEFN component for Delete Role

This section describes the creation of basic structure of CI from Delete Role component. For example, **IIQ\_DEL\_ROLE**

## Deleting the component interface

### Creating CI

1. Log on to Application Designer and click on **File => New**.  
A new dialog box named New Definition is displayed.
2. Select **Component Interface => OK**.  
A new dialog box named Select Source Component for Component Interface is opened.
3. Enter the name as **PURGE\_ROLEDEFN** under the **Selection Criteria** tab and click **Select**.  
A dialog box appears with the following message:  
Do you want to default the properties based on the underlying component definition
4. Click **Yes**.

Following screen shot appears:

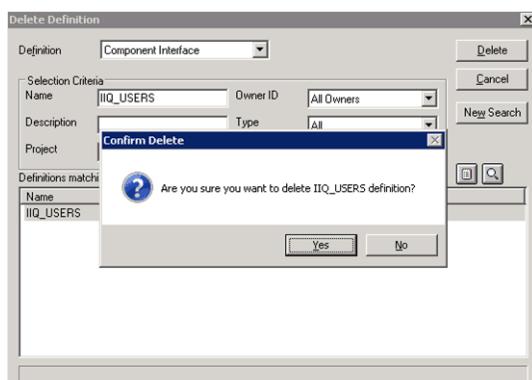
Name	Record	Field	Read ...
GETKEYS	PSROLEDEFN_VW	ROLENAME	
FINKEYS	PSROLEDEFN_VW	DESCR	
ROLENAME	PSROLEDEFN_VW	ROLENAME	
ROLESTATUS	PSROLEDEFN_VW	ROLESTATUS	
PROPERTIES			
ROLENAME	PURGE_ROLEDEFN	ROLENAME	Y
ROLE_TABLES_VW	ROLE_TABLES_VW	RECDescr	Y
RECDescr	ROLE_TABLES_VW	RECDescr	Y
RECNAMe	ROLE_TABLES_VW	RECNAMe	Y
METHODS			
Cancel			
Find			
Get			
Save			

5. Click on **File => Save As**.  
A new dialog box appears requesting for the name of the CI.
6. Enter the name of the CI as **{NEW\_NAME}**. For example, **IIQ\_DEL\_ROLE**.  
Delete the **ROLE\_TABLES\_VW** collective attribute.
7. Click **Save**.

## Deleting the component interface

Perform the following procedure to delete the Component interface:

1. Open **Application Developer => Files => Delete**.  
The Delete Definition window appears. as follows:



2. Select **Definition** as the name of the CI you want to delete and click on **Delete**.  
The required Component Interface is deleted.

## **Deleting the component interface**

# Appendix C: Partitioning Aggregation

---

This appendix describes the following information.

Overview.....	563
Partitioning Aggregation for ERP Connectors (SAP and PeopleSoft).....	564
Partitioning Aggregation for JDBC Connector .....	564
Partitioning Aggregation for Active Directory and LDAP Connectors .....	565
Partitioning Aggregation for Delimited and CyberArk Connectors .....	566
Partitioning Aggregation for IBM i Connector.....	566
Partitioning Aggregation for GoogleApps .....	566
Partitioning Aggregation for Cerner.....	567
Partitioning Aggregation for Tivoli Access Manager.....	568
Partitioning Aggregation for Azure Active Directory Connector .....	568
Partitioning Aggregation for RACF LDAP Connector .....	569

**Note:** For Partitioning Aggregation of SAP HR/HCM Connector, see ‘Partitioning’ section of “Chapter 23: SailPoint SAP HR/HCM Connector”.

## Overview

---

Partitioning aggregation processes the connector data in parallel, across multiple threads and multiple hosts to help increase the performance of aggregation tasks.

- Partitioning Aggregation can be requested by clicking the **Enable Partitioning** check box on the aggregation task definition. When the partitioning is enabled during aggregation, the aggregation task builds separate request for each partition. For more information, see “Connector specific Partitioning”.
- Some connectors do not support “Partitioning Aggregation” feature. For such connectors Default Partitioning is applied. For more information, see “Default Partitioning”.

## Default Partitioning

Default Partitioning processes the connector data in parallel, across multiple threads and multiple hosts to help increase the performance of Account Aggregation tasks.

Default Partitioning can be requested by clicking the **Enable Partitioning** check box on the Account Aggregation task definition and by specifying objects-per-partition field (default:1000).

To use the Default Partitioning feature for any connectors, perform the following:

- Select the **Enable Partitioning** check box in Account Aggregation Task.
- In **Objects per partition** textbox specify how many object should be used in one partition.
- Save and execute the task.

**Note:** The 'Objects per partition' field is invalid if the connector already supports partitioning. Connector specific partitioning is applied and gets preference if configured. In case, when connector supports partitioning but it is not configured then Default Partitioning is applied. If object per partition field is blank then default value 1000 is considered.

## Partitioning Aggregation for ERP Connectors (SAP and PeopleSoft)

### Connector specific Partitioning

When Connector Partitioning is used, then the partitioning criteria must be provided by the connector. Each partition is handled independently and configured at the application level. While forming the partitioning criteria ensure that all the objects on the server have been processed and nothing is skipped.

## Partitioning Aggregation for ERP Connectors (SAP and PeopleSoft)

---

To use the partitioning aggregation feature in ERP Connectors, perform the following:

1. Select the **Partition Enabled** check box.
2. Specify the criteria for partitioning in the **Partition Statements** textbox of the configuration parameter. For example, the statement A-M would be treated as one partition and in case of:
  - *SAP Direct Connector*, download all the SAP administrative users from A to M (including A and M).
  - *PeopleSoft Direct Connector*, download all the PeopleSoft User Profiles from A to M (including A and M).

To specify more than one partition the entries must be separated using a newline character.

## Partitioning Aggregation for JDBC Connector

---

JDBC Connector supports the manual partitioning through configured SQL statements.

The **Partitioning Enabled** configuration parameter must be selected and the list of SQL statements/parameterized stored procedures must be specified in the **Partitioning Statements** textbox.

For example, if there is an employee data-set that has 100,000 rows with a sequential **employeeId** field, the partitioning statements that can be used are as follows:

```
select x,y,z from a where employeeId <= 10000;  
select x,y,z from a where employeeId > 10000 AND employeeId <= 20000;  
select x,y,z from a where employeeId > 20000 AND employeeId < =30000;  
...  
select x,y,z from a where employeeId > 90000;
```

The above example would have 10 partitions, handling approximately 10,000 accounts and the last sentence (with `employeeId > 90000`) handling larger number of accounts depending on the total number of employees in the system.

**Note:** An additional aggregation option “`noAttributePromotion`” has been added. If this attribute is set to true, the attribute promotion would be skipped during aggregation.

# Partitioning Aggregation for Active Directory and LDAP Connectors

---

Active Directory and LDAP Connectors support the Partitioning Aggregation feature to enable faster retrieval of Active Directory and LDAP Directory data.

In LDAP, objects can be retrieved by means of a **searchDN**, **searchFilter** and **searchScope**. In Active Directory Connector, data can be partitioned by specifying a **searchDN** and/or a **iterateSearchFilter** as a partition entry. Active Directory Connector partition entries are the application configuration searchDNs list with each entry of the list treated as a single partition.

Typically, for a container based partitioning of data, define the searchDNs or partition list as follows:

```
<entry key="searchDNs">
  <value>
    <List>
      <Map>
        <entry key="searchDN" value="ou=test1,DC=test,DC=sailpoint,DC=com"/>
        <entry key="iterateSearchFilter" value="(&(objectclass=user) )"/>
        <entry key="searchScope" value="SUBTREE"/>
      </Map>
      <Map>
        <entry key="searchDN" value="ou=test2,DC=test,DC=sailpoint,DC=com"/>
        <entry key="iterateSearchFilter" value="(&(objectclass=user) )"/>
        <entry key="searchScope" value="SUBTREE"/>
      </Map>
    </List>
  </entry>
```

And for filter based partition, define the searchDNs list or partition list as follows:

```
<entry key="searchDNs">
  <value>
    <List>
      <Map>
        <entry key="searchDN" value="DC=test,DC=sailpoint,DC=com"/>
        <entry key="iterateSearchFilter" value="(&(objectclass=user)(sn=a*))"/>
        <entry key="searchScope" value="SUBTREE"/>
      </Map>
      <Map>
        <entry key="searchDN" value="DC=test,DC=sailpoint,DC=com"/>
        <entry key="iterateSearchFilter" value="(&(objectclass=user) (sn=b*))"/>
        <entry key="searchScope" value="SUBTREE"/>
      </Map>
    </List>
  </entry>
```

As seen above, in the first example, the OUs on which the search is performed are different although the **searchFilter** is the same. Whereas, in the second partitions entry, the OUs are same, but the **iterateSearchFilter** values are different. Since the required key values are similar, we could have both the above examples coupled together into the application configuration of a single Active Directory Connector application. Active Directory Connector combines the **searchDN** value and the **iterateSearchFilter** value and considers it as the partition context, avoiding any additional required configurations.

## Partitioning Aggregation for Delimited and CyberArk Connectors

- Note:** Each of the partitions specified has to be unique by way of the searchDN value or the iterateSearchFilter value. If not, the first partition would get aggregated skipping the subsequent duplicate ones.
- When there is no partitions list defined, the aggregation would execute over the baseDN and the iterateSearchFilter only, even though the task definition has partitioning enabled.
- Similarly, with partition list defined and partitioning is not enabled on the task definition, IdentityIQ would retrieve data from each searchDN entry in a sequential manner.

## Partitioning Aggregation for Delimited and CyberArk Connectors

---

Delimited and CyberArk Connectors support the following types of partitioning modes:

- **Auto:** Auto mode will automatically calculate the number of partitions and objects per partition based on the hints provided by the aggregator.
- **Manual:** Manual mode allows to specify the number of objects per partition and would be split equally as possible in partitions. By default partitioning uses the **Auto** mode.

In manual mode, user has to click on Manually Defined radio button and provide the value for Number of objects per partition field.

For example, if the total number of entities in the file is 1050 and the number of objects specified is 100 per partition, then there would be 11 partition task created. The first 10 partitions would fetch 100 objects per partition. The last would fetch 50 objects.

## Partitioning Aggregation for IBM i Connector

---

To use the partitioning aggregation feature in IBM i Connectors, perform the following:

1. Enable partitioning on the aggregation task definition page by selecting the **Enable Partitioning** check box.
2. On the Define application page in the iteration partitioning section, enter the list of statements for partitioning. Separate each statement by new line. Each statement can be an expression as mentioned below:

A\*  
AB\*  
C\*  
GA\*

The user profiles matching with that specific name will be retrieved. For example, if A\*, D\* or AB\* is mentioned then it will return all user profiles starting with A, D or AB respectively.

## Partitioning Aggregation for GoogleApps

---

Google Apps Rewrite Connector supports Partitioning Aggregation by using email, givenName (First Name) and familyName (Last Name) as filters. An asterisk is required in the value.

The filter names for the attributes are as mentioned below:

Attribute Name	Filter Name
email	partitionEmail
givenName	partitionGivenName
familyName	partitionFamilyName

The filters are to be added as entries into the application xml file.

For example:

- One partition which brings all users with email ID's starting with an 'a' and another partition which brings all users whose first name starts with an 's' appear as follows:

```
<entry key="partitionEmail">
    <value>
        <List>
            <String>a*</String>
        </List>
    </value>
</entry>
<entry key="partitionGivenName">
    <value>
        <List>
            <String>s*</String>
        </List>
    </value>
</entry>
```

- Partition of users with last name starting with an F appear as follows:

```
<entry key="partitionFamilyName">
    <value>
        <List>
            <String>F*</String>
        </List>
    </value>
</entry>
```

## Partitioning Aggregation for Cerner

---

Cerner Connector supports Partitioning Aggregation for the following attributes:

- firstName
- lastName

To use the partitioning aggregation feature in Cerner Connectors, perform the following:

1. Enable Partitioning on the aggregation task definition page by selecting the **Enable Partitioning** checkbox.
2. Select the **Partition Enable** checkbox from the application definition page.
3. For performing the search on basis of selection, select PartitionMode (FirstName or LastName).
4. Enter the partitioning statement as follows:
  - a. **Define Range:** For example, enter "A-C".  
This performs aggregation of all the accounts whose first character is A, B, C.

Or

- b. **startsWith:** For example, enter only "A" character.  
This performs aggregation of all the accounts whose first character is A.

## Partitioning Aggregation for Tivoli Access Manager

---

To use the partitioning aggregation feature in Tivoli Access Manager Connectors, perform the following:

1. Select the **Partition Enabled** check box.
2. Specify the criteria for partitioning in the **Partition Statements** text-box of the configuration parameter.  
**For example**, the statement A-M would be treated as one partition:
  - Tivoli Access Manager Connector would aggregate accounts, whose names start with the character between A and M, with A and M inclusive.
  - A new partition can be mentioned in the new line.

## Partitioning Aggregation for Azure Active Directory Connector

---

Azure Active Directory Connector supports partitioning aggregation based on search filters. To use partitioning feature perform the following:

1. Enable Partitioning on the aggregation task definition page by selecting the **Enable Partitioning** check box.
2. Add the following application configuration attribute:

```
<entry key="userPartitions">
```

The **userPartitions** configuration attribute is a multi-valued attribute. Its value consists of different search filters for the attributes which are filterable like accountEnabled, city, displayName, mail, usageLocation and so on.

For example,

```
<entry key="userPartitions">
  <value>
    <List>
      <String>startswith(displayName, 'J')</String>
      <String>startswith(givenName, 'Smith')</String>
      <String>accountEnabled eq true</String>
      <String>userPrincipalName eq 'Paul@contoso.onmicrosoft.com'</String>
    </List>
  </value>
</entry>
```

*Supported operators are*

- Logical operators: **and**, **or**
- Comparison operators: '**eq**'(equal to), '**ge**' (greater than or equal to) and '**le**'(Less than or equal to)
- **startswith**
- **any** is supported while querying multi valued properties

For example,

- proxyAddresses/any(c:c eq 'smtp:Mary@contoso.com')
- proxyAddresses/any(c:startswith(c,'smtp:Mary@contoso.com'))

## Partitioning Aggregation for RACF LDAP Connector

---

RACF LDAP Connector supports Partitioning Aggregation feature to enable faster retrieval of RACF data.

In RACF LDAP Connector, objects can be retrieved by means of a **searchDN**, **searchFilter** and **searchScope**. RACF LDAP Connector partition entries are the application configuration searchDNs list with each entry of the list treated as a single partition.

Typically, the partitions can be defined as the searchDNs list as follows:

```
<entry key="searchDNs">
  <value>
    <List>
      <Map>
        <entry key="iterateSearchFilter" value="(racfid=a*)"/>
        <entry key="searchDN" value="profiletype=USER,cn=SDBM "/>
        <entry key="searchScope" value="SUBTREE"/>
      </Map>
      <Map>
        <entry key="iterateSearchFilter" value="(racfid=b*)"/>
        <entry key="searchDN" value="profiletype=USER,cn= SDBM "/>
        <entry key="searchScope" value="SUBTREE"/>
      </Map>
      <Map>
        <entry key="iterateSearchFilter" value="(racfid=c*)"/>
        <entry key="searchDN" value="profiletype=USER,cn= SDBM "/>
        <entry key="searchScope" value="ONELEVEL_SCOPE"/>
      </Map>
      <Map>
        <entry key="iterateSearchFilter" value="(racfid=d*)"/>
        <entry key="searchDN" value="profiletype=USER,cn= SDBM "/>
        <entry key="searchScope" value="SUBTREE"/>
      </Map>
      .....
      .....
      .....
      .....
      <Map>
        <entry key="iterateSearchFilter" value="(racfid=z*)"/>
        <entry key="searchDN" value="profiletype=USER,cn= SDBM "/>
        <entry key="searchScope" value="SUBTREE"/>
      </Map>
    </List>
  </value>
```

## Partitioning Aggregation for RACF LDAP Connector

```
</entry>
```

**Note:** Each specified partition has to be unique by way of the iterateSearchFilter value. If not, the first partition would get aggregated skipping the subsequent duplicate ones.  
Partitions must be created in such a way that each partition must not exceed the default or specified search limit.

# Appendix D: Before and After Provisioning Action

---

This appendix describes the following information.

Overview.....	571
Before and After Provisioning Action for AIX/Linux/Solaris Connectors.....	571
Before and After Provisioning Action for IBM i Connector.....	573

## Overview

---

While managing account and group on any managed system, you may need to perform some custom action which is not available out of the box on the Managed System. This can be achieved through before and After action. Some of the connectors support Before and after action using Before and After rules configuration in Application. This appendix describes the same. The java code executed in rule would be specific to Connector and can perform any custom action.

## Before and After Provisioning Action for AIX/Linux/Solaris Connectors

---

For AIX/Linux/Solaris Connectors, you can configure before and after provisioning rule to support Before/After Actions. In the Before/After provision rule we can carry out any operation before/after the provisioning operation. This document describes the different steps required to perform the same.

### Pre-requisite

---

AIX/Linux/Solaris Connector application must be configured in IdentityIQ.

### Creating Before and After Provisioning Action

---

Perform the following procedure to use the Before and After Action functionality for UNIX Connectors:

1. Navigate to where UNIX application is configured.  
Open UNIX application Rules tab. Select the following option as required:
  - Before Provisioning Rule
  - After Provisioning Rule
2. Write java code in Rule Editor section. Specify the Rule Name and Save it.  
Select the rule name you saved in earlier step by using Select Rule option.  
Perform any provisioning task and check if before/after provisioning rule gets executed.  
For example, java code for After provisioning action which creates directory for user after Unix account is created:

## Before and After Provisioning Action for AIX/Linux/Solaris Connectors

```
import java.io.InputStreamReader;
import ch.ethz.ssh2.Connection;
import ch.ethz.ssh2.Session;
import ch.ethz.ssh2.StreamGobbler;
import java.util.*;
import sailpoint.object.ProvisioningPlan.ObjectOperation;
import sailpoint.object.ProvisioningPlan.ObjectRequest;
import sailpoint.object.ProvisioningPlan.AttributeRequest;
import sailpoint.object.ProvisioningPlan.AccountRequest;
import sailpoint.object.ProvisioningPlan.GenericRequest;
import sailpoint.api.*;
// Here I have hardcoded hostname, user, password,
// we can take this from Application config
String hostname = "127.0.0.1";
String username = "joe";
String password = "joespass";
try
{
String userId = null;
boolean operationCreate = false;
// Get the reuest
List accountRequests = plan.getAccountRequests();
if(accountRequests != null){
for (AccountRequest acctReq : accountRequests) {
// Get the opertion
AccountRequest.Operation op = acctReq.getOperation();
if (op == AccountRequest.Operation.Create){
userId = acctReq.getNativeIdentity();
operationCreate = true;
}
}
}
if (operationCreate)
{
// Create a connection instance
Connection conn = new Connection(hostname);
// Now connect
conn.connect();
// Authenticate. Here we have used password authentication,
// you can use public key authentication as well.
boolean isAuthenticated = conn.authenticateWithPassword(username,
password);
if (isAuthenticated == false)
throw new IOException("Authentication failed.");
// Create a session
Session sess = conn.openSession();
// To customize implementation,
// you can execute any command/shell script here
if(userId != null){
String command="mkdir /tmp/" + userId ;
sess.execCommand(command);
}
// Show exit status, if available (otherwise "null")
System.out.println("ExitCode: " + sess.getExitStatus());
// Close this session
sess.close();
// Close the connection
```

```

        conn.close();
    }
}
catch ( IOException e )
{
e.printStackTrace( System.out );
}

```

**Note:** This is an example of After Provisioning Rule for Create operation. User can configure rule for Create/Delete/Update operation as required. The java code which is executed in Rule should be modified accordingly.

## Before and After Provisioning Action for IBM i Connector

---

For IBM i Connector, Customer can configure before and after provisioning action to support Before/After Actions. JTOpen library provides API to execute Command or CL scripts on IBM i host which are used to perform pre/post provisioning actions.

### Pre-requisites

---

- IBM i application configured
- CL scripts configured on IBM i host.

### Creating CL scripts

---

Perform the following procedure to create the CL scripts on IBM i computer:

1. Create a library as follows:  
CRTLIB library-name  
For example, CRTLIB MTEST
2. Make the library as current library CHGCURLIB library-name  
For example, CHGCURLIB MTEST
3. Create source physical file in that library CRTSRCPF QCLSRC  
(QCLSRC is the standard naming convention in the IBM i for CLP source members).
4. Add member to **ADDPFM** file and enter the following details:  
Physical File – QCLSRC  
Library – MTEST  
Member – Test123  
Text – \*BLANK  
Press <ENTER>
5. Enter the following command:  
STRPDM  
Select option “3 – work” with members and enter the following details:  
File – QCLSRC  
Library – MTEST  
Press <ENTER>
6. To create members press **F6** option and enter the following details:

## Before and After Provisioning Action for IBM i Connector

```
Source member - TEST123
Source Type   - CLP
Press <ENTER>
```

7. List of members will appear. To write CL script or edit member file:

```
Opt      - 2 (edit)
```

```
Member - Test123
```

```
Type    - CLP
```

```
The member file opens in seu editor.
```



```
Columns . . . : 1 71          Edit          YASIRU/QCLSRC
SEU==> _____ SRCMBR
FMT ** ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
***** Beginning of data *****
***** End of data *****
```

Type **I** (for insert) on the first line as shown in the following figure and press **Enter**.



```
Columns . . . : 1 71          Edit          YASIRU/QCLSRC
SEU==> _____ SRCMBR
FMT ** ...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
I ***** Beginning of data *****
***** End of data *****
```

For more information of seu editor, see the following link:

<http://as400iseries.wordpress.com/2013/03/13/using-the-seu-editor/>

8. Write your CL script program in the seu editor and press **F3**.

Options for saving the file are displayed.

Enter **Y (Yes)** and press **Enter**. The file is saved.

9. To compile, use option 14 in front of member file name. Enter **Y** in the following field:

Delete existing object ..... Y Y=Yes, N=No

10. Navigate to where IBM i application is configured. Select IBM i application.

Open IBM i application Rules tab. Select the following option as required:

- Before Provisioning Rule
- After Provisioning Rule

11. Write java code in Rule Editor section. Specify the Rule Name and Save it.

Select the rule name you saved in earlier step by using Select Rule option.

Perform any provisioning task and check if before/after provisioning action gets executed.

For example, java code to run CL-script

```
import java.io.IOException;
import com.ibm.as400.access.AS400;
import com.ibm.as400.access.AS400Exception;
import com.ibm.as400.access.AS400Message;
import com.ibm.as400.access.AS400SecurityException;
```

```

import com.ibm.as400.access.CommandCall;
import com.ibm.as400.access.ErrorCompletingRequestException;
import com.ibm.as400.access.ObjectDoesNotExistException;
AS400 system = null;
String host = "NDS400.isr.bmc.com";
String user = "UMITTAL";
String password = "UMITTAL4";
system = new AS400(host, user, password);
System.out.println ("Connected sucessfully!!!!");
CommandCall cmd = new CommandCall(system);

try{
if(cmd.run("CALL PGM(MTEST/Test123)" ) != true){
// Test123 is member file in library MTEST which gets called here.
//similarly we can use - command.run("CRTLIB FOREST");
}
else {
AS400Message[] messagelist = cmd.getMessageList();
for (int i = 0; i < messagelist.length; ++i){
System.out.println(messagelist[i].getText());
System.out.println("Command Success!!!");
}
}
}catch(Exception e){
System.out.println("error" + e.getMessage());
}
System.out.println("ending program");

```

**Note:** In this example, hostname and related parameters have been hardcoded, you can access these parameter from IdentityIQ objects ( Application, AccountRequest, Items).

## **Before and After Provisioning Action for IBM i Connector**

# Appendix E: IQService

---

This appendix describes the following information.

Install and register the IQService for Windows.....	577
IQService Before/After Scripts .....	580

## Install and register the IQService for Windows

---

The IQService is a native Windows service that enables IdentityIQ to participate in a Windows environment and access information only available through Windows APIs. Following are connectors for which IQService must be installed and registered on windows host computer from where the respective connectors are accessible.

Connector	Location	Other Libraries
Active Directory	Remote or Local	<ul style="list-style-type: none"> <li>• .NET Framework version 4.5.2</li> <li>• For Exchange Server <ul style="list-style-type: none"> <li>- Windows PowerShell version 3.0</li> </ul> </li> <li>• For Lync/Skype Server <ul style="list-style-type: none"> <li>- (<i>For Microsoft Lync server 2013</i>) Microsoft Lync Server Administrative Tools 2013</li> <li>- (<i>For Microsoft Skype for Business Server 2015</i>) Microsoft Skype Server Administrative Tools 2015</li> <li>- Windows PowerShell version 3.0</li> </ul> </li> </ul> <p><b>Note: IQService host must be in the same domain as that of Microsoft Lync\Skype for Business Server.</b></p>
Lotus Domino	Remote or Local	<ul style="list-style-type: none"> <li>• .NET Framework version 4.5.2 on Windows 32-bit or 64-bit</li> </ul> <p><b>Note: See “ Pre-requisite for Lotus Domino Connector”.</b></p> <ul style="list-style-type: none"> <li>• Lotus Notes client</li> <li>• The PATH environment variable must contain the Notes data folder. For example, C:\Program Files\IBM\Notes</li> <li>• Visual Studio C++ 2005 SP1 Redistributable Package</li> </ul>
Microsoft Project Server	Remote or Local	.NET Framework version 4.5.2 *

## Install and register the IQService for Windows

Connector	Location	Other Libraries
Microsoft SharePoint Server	Must be installed on the computer having the same domain as that of SharePoint Server	<ul style="list-style-type: none"> <li>• .NET framework version 4.5.2</li> <li>• Windows PowerShell version 3.0</li> </ul>
Microsoft SharePoint Online	Remote	<ul style="list-style-type: none"> <li>• .NET Framework version 4.5.2</li> <li>• Windows Azure Active Directory Module</li> <li>• Microsoft On-line Services Assistant</li> <li>• Windows Identity Foundation</li> <li>• SharePoint Foundation 2010 Client Object Model</li> <li>• Windows PowerShell version 3.0</li> </ul>
Microsoft Windows Local - Direct	Remote or Local	.NET Framework version 4.5.2  Must to be installed locally to support revoking permissions.
Forefront Identity Manager Provisioning Integration Module	Remote	.NET Framework version 4.5.2

**Note:** The \* sign represents that Framework version 4.0 is required. Rename app.config file present in the IQService directory to IQService.exe.config file and restart IQService.

### Pre-requisite for Lotus Domino Connector

IQService must be running as a 32-bit process in order to interact with 32-bit Lotus Domino Client. If IQService must be installed on a 64-bit Windows system perform the following before installing IQService:

1. Download Microsoft Windows SDK or .NET Framework SDK.

2. Run the following command from command prompt:

```
<SDK Bin>\CorFlags.exe <IQServiceHome>\IQService.exe /32BIT+
```

For example, C:\Program Files\Microsoft SDKs\Windows\v7.1\Bin\x64>CorFlags.exe

```
C:\IQService\IQService.exe /32BIT+
```

This command converts the IQService.exe to a 32-bit application.

## Installing and registering IQService

**Note:** If IQService is installed, the IQService version must match the IdentityIQ server version, including the patch version. If you upgrade one you must upgrade the other, ensuring that the IQSERVICE patch version matches the IdentityIQ Application server.

**Note:** For upgrading IQService, SailPoint recommends the following:

- uninstall IQService and then install the new version of IQService. This clears the registry of settings, in the event that they change as part of the upgrade.
- back up the current installation before uninstalling to aid with troubleshooting the new version.

To install and register the IQService, perform the following:

1. Create a directory in which you want to download the service. For example, `c:\iqservice`.
2. Extract the `IQService.zip` archive from the `\IQHOME\WEB-INF\bin\win` directory of the IdentityIQ installation into the created directory.
3. Run the following command to install a Windows service named IQService.  
`IQService.exe -i`

It registers the service with the new registry path,

`HKEY_LOCAL_MACHINE\SOFTWARE\SailPoint\IQService` with the following keys:

- **port**: port to listen
- **tracefile**: path to the tracefile
- **tracelevel**: 0 (off)  
3 (verbose)
- **maxTraceFiles**: maximum number of Trace log files that must get created before overwriting the older files.
- **traceFileSize**: maximum file size of a trace file in bytes

A new file is created when the current file exceeds this limit.

4. Start the service either from the Services Applet or from the command line by running the following command:

`IQService.exe -s`

Other command line options with this service are:

- **-d**: run in the foreground in debug mode instead of in the background using the service control manager
- **-k**: stop the service
- **-r**: remove the service
- **-v**: display version information
- **-u**: Uninstall the service. Removes the service components and clears the registry entries.

Trace Parameters (require a restart of the IQService):

- **-l [level]**: Trace Level 0-3
  - 0: Off
  - 1: Information
  - 2: Error
  - 3: Debug
- **-f [fileName]**: Trace File Name (For example, "`C:\IQService\IQServiceLog.log`")

## IQService Public Key Exchange Task

---

By Default, the IQService uses a shared key encryption technique where IdentityIQ server and the IQService encrypts and decrypts data using a common key.

Optionally, communication between IdentityIQ and IQService can be configured to use dynamic keys. When configured, it uses the public/private pair key approach, where each side uses a public key to encrypt data sent to each side. The receiving end then decrypts the data using the local private key.

## **IQService Before/After Scripts**

This approach requires the key exchange to be performed in IdentityIQ and IQService as part of securing a communication channel by public/private key.

To achieve this, **IQService Public Key Exchange Task** must be run which takes in a list of applications as input. For each application, the key is updated and used in the transmission.

- Note:**
- Once secured using the new dynamic key, only one IdentityIQ server (or cluster using the same database) can talk to a single IQService.
  - Applications using the same IQService host will be using the same public/private key.
  - Ensure to pair the IQService version with the IdentityIQ server version deployed in the production environment.

## **IQService Before/After Scripts**

---

IdentityIQ provides most of the provisioning functionality for many systems through its connectors. Some systems provide better integration interface from Windows platform compared to other platforms. Hence connectors for such systems require IQService deployed on a Windows system. The IQService implementation performs the provisioning functions (such as Add User, Connect User to a Group) that are supported by the respective System. The IQService functions are called by the IdentityIQ connector implementation.

In addition to the basic action, some organizations may require supplementary actions performed by each function from Windows system. The IQService supports customization of the functions by allowing integrating before / after scripts implemented in any language. Following are some features of the IQService Before/After script:

- Centralized configurations (in IdentityIQ) for setting up Before/After scripts
- Supports Object Oriented scripting
- Script refers SailPoint library to get the request, result classes
- Can be executed with specific context
- Script can modify request/result

A script is a group of statements that perform one or more actions and manipulate request / result attributes. Scripts can be used to automate any required actions that are currently performed manually. Scripts called before processing the request are referred to as native before scripts and scripts called after processing the request are referred to as native after scripts.

The scripts needs to be defined in a Rule and then configured for an Application in IdentityIQ. Based on the rule type, the connector would send the scripts to IQService that needs to be executed before / after processing the request. The IQService supports executing before and after Rules for Create, Modify, and Delete request operations.

## **Writing a script**

---

IQService divides scripts in the following categories:

- Scripts with Object Oriented support
- Scripts without Object Oriented support

## Scripts with Object Oriented support

Scripting languages with Object Oriented capabilities (for example, PowerShell) are popular because of their simplistic nature and easy to use. These scripts can form objects of any type by referring any library/assembly implemented in any language and call its methods.

Native scripts implemented in these languages have easier and more powerful access to request and result objects. IQService comes with a class library named `Utils.dll` which bundles all required classes to access the request and result objects. The inputs provided to the script would be in the form of process environment variables. The following table describes the environment variables created by IQService:

Name	Type	Before Script	After Script
Application	System.Collections.Hashtable	Read Only	Read Only
Request	SailPoint.Utils.objects.AccountRequest	Read/Write	Read Only
Result	SailPoint.Utils.objects.ServiceResult	Not Available	Read/Write

The data in the environment variables is in XML. The script creates respective objects using `Utils.dll`. Once the object is modified, the script should convert it to XML by calling `toxml()` method of the object and write the XML to a file at the path that is passed as the only argument to the script. The script returns non-zero value in case of error and writes the error message in the file at the path that is passed as the argument to the script. This failure is communicated to IdentityIQ as part of result.

### *Sample PowerShell before script*

Following is a sample PowerShell before script which modifies value of an attribute and add one new attribute to the request:

```
# Refer to SailPoint class library Requires PowerShell v2 installed on the system.
Add-type -path utils.dll

# Read the environment variables
$sReader = New-Object System.IO.StringReader([System.String]$env:Request);

# Form the xml reader object
$xmlReader =
[System.xml.XmlTextReader]([SailPoint.Utils.xml.XmlUtil]::getReader($sReader));

# Create SailPoint Request object
$requestObject = New-Object SailPoint.Utils.objects.AccountRequest($xmlReader);

# Loop through the attributes from the request
foreach ($attribute in $requestObject.AttributeRequests){
    if($attribute.Name -eq "description"){
        $attribute.value = "my description";#change value of the attribute
    }
}

# Add a new attribute to request
$attributeObject = New-Object SailPoint.Utils.objects.AttributeRequest;
$attributeObject.Name = "otherMobile";
$otherMobileValues = New-Object System.Collections.ArrayList;
$otherMobileValues.Add("222-292-2929");
$otherMobileValues.Add("333-292-2929");
$attributeObject.Value= $otherMobileValues;
$attributeObject.Operation = "Set";
$requestObject.AttributeRequests.Add($attributeObject);

# Write the request xml to file at the path passed as argument
$requestObject.toxml()|out-file $args[0];
```

## IQService Before/After Scripts

### *Sample PowerShell after script*

Following is a sample PowerShell after script which ensures that the request was processed successfully and creates home directory at the path specified in the request:

```
# Refer to SailPoint class library. Requires PowerShell v2 installed on the system.
Add-type -path E:\SVN\trunk\src\WinRPCGateway\IQService\bin\Debug\utils.dll

# Read the environment variables
$sr = New-Object System.IO.StringReader([System.String]$env:Request);
$sr = New-Object System.IO.StringReader([System.String]$env:Result);

# Form the xml reader objects
$xmlReader = [
System.xml.XmlTextReader]([sailpoint.utils.xml.XmlUtil]::getReader($sr));
$xmlReader_Result = [
System.xml.XmlTextReader]([sailpoint.utils.xml.XmlUtil]::getReader($sr));

# Create SailPoint objects
$requestObject = New-Object Sailpoint.Utils.objects.AccountRequest($xmlReader);
$resultObject = New-Object Sailpoint.Utils.objects.ServiceResult($xmlReader_Result);

#Check if the request was processed successfully
if($resultObject.Errors.Count -eq 0){

    #Get Home directory path
    foreach ($attribute in $requestObject.AttributeRequests){
        #Create Home directory
        if($attribute.Name -eq "TS_TerminalServicesHomeDirectory"){
            New-Item $attribute.Value -itemtype directory;
        }
    }
}
```

### Scripts without Object Oriented support

Non Object Oriented scripts do not support referring to the class library or a way of parsing XML. To have easy access to each attribute along with their operation and values, IQService creates process environment variables for each of the application and request attribute with name in the form **SP\_<OPERATION>\_<NAME>** for requests and **SP\_APP\_<NAME>** for application. For native identity, the environment variable would be **SP\_NativeIdentity**. These types of scripts have limited access to result and get only **SUCCESS** or **FAIL** in the **Result** environment variable. Hence the after scripts implemented using these scripting languages cannot modify any attribute/result. The before scripts can add, modify, or remove any attribute from the request. The script needs to write the newly added or modified attribute to the file at the path passed as an argument to the script in the form **SP\_<OPERATION>\_<NAME>=<VALUE>**. For removing the attribute from the request, write **/~<ATTRIBUTE\_NAME>** to the file. Value for the multivalued attribute is delimited by **/#**

Following is a sample batch after script which ensures that the request was processed successfully and creates home directory at the path specified in the request:

```
IF %Result% ==SUCCESS md %SP_Set_TS_TerminalServicesHomeDirectory%
```

## Creating a Rule

---

IdentityIQ (6.0) user interface does not have facility to create Native Rule applicable for IQService. Create a rule with any supported type from the user interface. Add the script to the Rule source and save the Rule. Navigate to the debug page, open the newly created Rule and perform following steps:

1. Change the rule type to one of the following types as appropriate:

Type name	Description
ConnectorBeforeCreate	Before script for create operation.
ConnectorAfterCreate	After script for create operation.
ConnectorBeforeModify	Before script for modify operation includes enable/disable, unlock.
ConnectorAfterModify	After script for modify operation includes enable/disable, unlock.
ConnectorBeforeDelete	Before script for delete operation.
ConnectorAfterDelete	After script for delete operation.

2. Add the following attributes to the Rule in the form:

```
<Attributes>
  <Map>
    <entry key=<NAME> value=<VALUE>/>
  </Map>
</Attributes>
```

Name	Description	Default Value
ObjectOrientedScript	Whether the rule source uses object oriented scripting.	False
disabled	Set to true if the rule should not be executed on the IQService side.	False
extension	Extension of the script.	.bat
program	Program/application that can execute this type of script.  <b>Note: Ensure that this program is installed on the system where IQService is running and is properly configured to execute the scripts.</b>	cmd.exe or cmd
timeout	Time interval (in seconds) for which IQService should wait for script to return. After this interval, IQService abort the script.	10

## Configuring the Rules in Application

With this releases, IdentityIQ user interface does not have facility to configure Native Rule applicable for IQService in Application. Navigate to the debug page, open the application and add **<nativeRules>** under Attributes map with list of names of the Rules that must be configured for this application.

For example:

```
<entry key="nativeRules">
  <value>
    <List>
      <String>AfterCreate-Powershell</String>
      <String>BeforeCreate-Powershell</String>
      <String>BeforeModify-Batch</String>
    </List>
  </value>
</entry>
```

## **IQService Before/After Scripts**