NetIQ® Identity Manager

Setup Guide for Linux

October 2019

Legal Notice

For information about NetIQ legal notices, disclaimers, warranties, export and other use restrictions, U.S. Government restricted rights, patent policy, and FIPS compliance, see https://www.netiq.com/company/legal/.

Copyright (C) 2019 NetIQ Corporation. All rights reserved.

Contents

About this Book and the Library 11
About NetIQ Corporation 13

Part I Overview of Identity Manager Environment 15

1 Brief Introduction of Identity Manager Components 17

Identity Manager Server Components . . . 18
Identity Manager Server . . . 19
Remote Loader . . . 19
Fanout Agent . . . 20
iManager . . . 20
Identity Applications Components . . . 21
User Application . . . 21
Authentication Service . . . 21
Self-Service Password Reset . . . 21
Web Application Server . . . 22
Identity Applications Database . . . 22
Drivers for Identity Applications . . . 23
Identity Reporting Components . . . 23
Identity Reporting . . . 23
Authentication Service . . . 24
Self-Service Password Reset . . . 24
Identity Reporting Database . . . 24
Web Application Server . . . 24
Drivers for Identity Reporting . . . 25
Sentinel Log Management for Identity Governance and Administration . . . 25
Identity Manager Tools . . . 26
Designer for Identity Manager . . . 26
Analyzer for Identity Manager . . . 26
Functional Architecture . . . 27
Deployment Options for Identity Manager . . . 28
Sample Identity Manager Deployments . . . 29
Sample Advanced Edition Deployment . . . 31
Sample Standard Edition Deployment . . . 32

Part II Planning to Install Identity Manager 35

2 Planning Your Installation 37

Implementation Checklist . . . 37
Recommended Installation Scenarios and Server Setup . . . 38
Deciding When to Install SLM for IGA . . . 39
Considerations for Installing in a Distributed Setup . . . 39
Determine Hardware Requirements . . . 41
System Requirements Worksheet . . . 41

Installing Identity Manager on SLES Servers . . . 42
Installing Identity Manager on RHEL Servers . . . 43
Prerequisites . . . 43
Ensuring that the Server has Dependent Libraries . . . 44
Creating a Repository on RHEL 8.x for the Installation Media . . . 45
Creating a Repository on RHEL 7.x for the Installation Media . . . 47
Running a Prerequisite Check . . . 48

3 Considerations for Installing Identity Manager Components 49

Installation Order . . . 49
Understanding the Installation and Configuration Process . . . 49
Considerations for Installing Identity Manager Engine Components and Remote Loader . . . 51
Considerations for Installing Identity Applications Components . . . 52
Installation Considerations . . . 52
Database Considerations. . . . 53
Configuring the Database for Identity Applications . . . 54
Considerations for Installing Identity Reporting Components . . . 56
Prerequisites for Identity Reporting . . . 56
Identifying Audit Events for Identity Reporting. . . . 57
Considerations for Installing Designer. . . . 58
Considerations for Installing Analyzer . . . 58
Considerations for Installing SLM for IGA . . . 59
Reviewing the Ports Used by the Identity Manager Components . . . 60

Part III Installing and Configuring Identity Manager Components 63

4 Installing Identity Manager 65

Performing an Interactive Installation . . . 65
Performing a Silent Installation . . . 65
Installing Identity Manager Engine as a Non-root User . . . 66
Installing NICI . . . 66
Performing a Non-root Installation of Identity Vault . . . 67
Performing a Non-root Installation of Engine . . . 68
Installing SSPR . . . 69
Performing an Interactive Installation of SSPR . . . 69
Performing a Silent Installation of SSPR. . . . 70
Installing Remote Loader . . . 70
Interactive Installation. . . . 70
Silent Installation . . . 70
Installing Designer. . . . 71
Installing Analyzer . . . 71
Using the Wizard to Install Analyzer. . . . 71
Installing Analyzer Silently . . . 72
Adding XULRunner to Analyzer.ini . . . 72
Installing Sentinel Log Management for Identity Governance and Administration . . . 73
Installing Sentinel Log Management as a Root User. . . . 73
Installing Sentinel Log Management as a Non-root User . . . 74
Installing Java Remote Loader . . . 75
Understanding the Directory Structure. . . . 76

5 Configuring the Identity Manager Components 77

Using Non-Intuitive Passwords During Configuration . . . 77
Understanding the Configuration Parameters . . . 77
Creating and Configuring a Driver Set . . . 87
Configuring the Identity Manager Components. . . . 89
Performing an Interactive Configuration . . . 89
Performing a Silent Configuration . . . 90
Configuring SSPR . . . 90
Performing an Interactive Configuration . . . 90
Performing a Silent Configuration . . . 90
Modifying the Single Sign-on Access Settings on the OSP Server . . . 91

6 Final Steps for Completing the Installation 93

Configuring the Identity Vault . . . 93
Creating Value Indexes for Identity Vault. . . . 93
Manually Importing Identity Applications and Identity Reporting Certificates into Identity Vault . . . 94
Configuring a Non-Administrator User as an Identity Vault Administrator . . . 94
Configuring the Remote Loader and Drivers. . . . 94
Configuring a Connected System . . . 95
Creating and Configuring a Driver Set . . . 95
Creating a Driver . . . 97
Defining Policies. . . . 98
Configuring Forgotten Password Management . . . 98
Using Self Service Password Reset for Forgotten Password Management . . . 98
Using an External System for Forgotten Password Management . . . 100
Updating SSPR Links in the Dashboard for a Distributed or Clustered Environment . . . 102
Configuring Identity Applications . . . 102
Configuring the Settings for the Identity Applications . . . 103
Specifying a Location for the Permission Index. . . . 128
Deploying REST APIs for Identity Applications . . . 128
Configuring the Oracle Database With Identity Applications. . . . 129
Accessing the Oracle Database Using Oracle Service Name . . . 129
Manually Creating the Database Schema . . . 130
Configuring Single Sign-On Settings for the Identity Applications. . . . 131
Starting the Identity Applications. . . . 132
Configuration and Usage Considerations for the Identity Applications . . . 132
Configuring the Runtime Environment for Data Collection. . . . 132
Configuring the Data Collection Services Driver to Collect Data from the Identity Applications . . . 133
Migrating the Data Collection Service Driver . . . 134
Adding Support for Custom Attributes and Objects . . . 136
Adding Support for Multiple Driver Sets . . . 138
Configuring the Drivers to Run in Remote Mode with SSL. . . . 140
Configuring Identity Reporting. . . . 141
Configuring the Managed System Gateway Driver . . . 141
Manually Adding the DataSource in the Identity Data Collection Services Page . . . 142
Running Reports on an Oracle Database . . . 142
Manually Generating the Database Schema . . . 142
Deploying REST APIs for Identity Reporting. . . . 146
Connecting to a Remote PostgreSQL Database. . . . 146
Completing a Non-root Installation . . . 147

Creating a Container for Password Policies 147
Adding Support for Graphics in Email Notifications 147
Activating Identity Manager 148
Creating SSL Certificate for Distributed Deployment 148

Part IV Deploying Identity Manager Containers 151

7 Planning Your Container Deployment 153

System Requirements 153
Obtaining the Docker Images 153

8 Identity Manager Containers Deployment 155

Best Practices 155
Managing Container Volume Data 156
Creating the Silent Properties File 156
Modes of Container Deployment 158
Deploying Containers on a Single Server 158
Prerequisites 158
Deploying Identity Manager Engine Container 159
Deploying Remote Loader Container 160
Deploying Fanout Agent Container 160
Deploying iManager Container 161
Deploying OSP Container 162
Deploying PostgreSQL Container 162
Deploying Identity Applications Container 163
Deploying Form Renderer Container 164
Deploying ActiveMQ Container 165
Deploying Identity Reporting Container 165
Deploying SSPR Container 166
Deploying Containers on Distributed Servers 167
Prerequisites 168
Exposing the Ports to Be Accessed by Containers 168
Deploying Identity Manager Engine Container 170
Deploying Remote Loader Container 170
Deploying Fanout Agent Container 171
Deploying iManager Container 171
Generating Certificates With Identity Vault Certificate Authority 172
Deploying OSP Container 176
Deploying PostgreSQL Container 177
Deploying Identity Applications Container 178
Deploying Form Renderer Container 179
Deploying ActiveMQ Container 179
Deploying Identity Reporting Container 180
Deploying SSPR Container 181

Part V Upgrading Identity Manager 183

9 Preparing to Upgrade Identity Manager 185

Checklist for Upgrading Identity Manager 185
Understanding Upgrade Process 186

Supported Upgrade Paths 187
Upgrading from Identity Manager 4.7.x Versions 188
Upgrading from Identity Manager 4.6.x Versions 189
Backing Up the Current Configuration 191
Exporting the Designer Project 191
Exporting the Driver Configuration 192

10 Upgrading Identity Manager Components 195

Considerations for Upgrade 195
Upgrade Sequence 196
Upgrading Designer 196
Upgrading Identity Manager Engine 197
Upgrading the Identity Vault 197
Upgrading the Identity Manager Engine 197
Upgrading the Identity Manager Engine as a Non-root User 199
Upgrading the Remote Loader 200
Upgrading the Java Remote Loader 201
Upgrading iManager 202
Stopping and Starting Identity Manager Drivers 204
Stopping the Drivers 204
Starting the Drivers 205
Upgrading the Identity Manager Drivers 206
Creating a New Driver 206
Replacing Existing Content with Content from Packages 206
Keeping the Current Content and Adding New Content with Packages 207
Upgrading Identity Applications 207
Considerations for Upgrade 208
System Requirements 209
Understanding the Upgrade Program 209
Upgrading PostgreSQL 209
Upgrading the Identity Applications Components 210
Post-Upgrade Tasks for Identity Applications Components 214
Verifying the Version Numbers After Upgrade 217
Upgrading Identity Reporting 217
Considerations for Upgrade 218
Upgrading Sentinel Log Management for IGA 218
Upgrading Identity Reporting 219
Post-upgrade Steps for Reporting 219
Verifying the Upgrade for Identity Reporting 220
Upgrading Analyzer 221
Adding New Servers to the Driver Set 221
Removing the Old Server from the Driver Set 221
Restoring Custom Policies and Rules to the Driver 222
Using Designer to Restore Custom Policies and Rules to the Driver 223
Using iManager to Restore Custom Policies and Rules to the Driver 223

11 Switching from Advanced Edition to Standard Edition **225**

Part VI Migrating Identity Manager Data to a New Installation **227**

12 Preparing to Migrate Identity Manager **229**

Checklist for Performing a Migration. 229

13 Migrating Identity Manager to a New Server **231**

Prerequisites 231
Preparing Your Designer Project for Migration. 231
Migrating the Identity Manager Engine to a New Server 232
Copying Server-specific Information for the Driver Set 232
Copying the Server-specific Information in Designer 233
Changing the Server-specific Information in iManager 233
Changing the Server-specific Information for the User Application 234
Updating the User Application Drivers 234
Deploying the Drivers for Identity Applications 234
Migrating Identity Applications 235
Migrating the Database to the New Server 235
Installing Identity Applications On the New Server 236
Migrating Identity Reporting 237
Updating the Drivers for Identity Reporting 237
Deploying the Drivers for Identity Reporting 238
Migrating Your Existing Data to a New Database 238
Setting up the New Reporting Server. 242
Creating the Data Synchronization Policy 242

Part VII Deploying Identity Manager on AWS EC2 **243**

14 Planning and Implementation of Identity Manager on AWS EC2 **245**

Prerequisites 245
Deployment Procedure. 245
Preparing AWS Virtual Private Cloud 247
Creating and Deploying Instances 249
Preparing the EC2 Instances 250
Setting Up Identity Manager Components 252
Setting Up Database for Identity Applications and Identity Reporting 252
Setting Up Designer 254
Creating an AWS EC2 Load Balancer 254
(Optional) Creating Alias DNS with the Registered Hosted Zone. 259
Accessing Identity Manager Components 260
Security Considerations 260

15 Example Scenarios of Hybrid Identity Manager **263**

Using Remote Loader Connection 263
Using Multi-Server Driver Set Connection 264
Using eDirectory Driver Connection 266

Part VIII Deploying Identity Manager for High Availability 269

16 Preparing for Installing Identity Manager in a Cluster Environment 271

Prerequisites . . . 271
Identity Vault . . . 271
Identity Applications . . . 272
Database for Identity Applications . . . 272
Preparing a Cluster for the Identity Applications . . . 273
Understanding Cluster Groups in Tomcat Environments . . . 273
Setting System Properties for Workflow Engine IDs . . . 273

17 Sample Identity Manager Cluster Deployment Solution on SLES 12 SP3 or Later Versions 275

Prerequisites . . . 276
Installation Procedure. . . . 276
Configuring the iSCSI Server . . . 276
Configuring the iSCSI initiator on all Nodes. . . . 277
Partitioning the Shared Storage . . . 277
Installing the HA Extension . . . 278
Setting up Softdog Watchdog . . . 278
Configuring the HA Cluster . . . 279
Installing and Configuring Identity Vault and Identity Manager Engine on Cluster Nodes . . . 280
Configuring the Identity Vault Resource . . . 281
Primitives for eDirectory and Shared Storage Child Resources . . . 282
Changing the Location Constraint Score . . . 283

18 Sample Identity Applications Cluster Deployment Solution on Tomcat Application Server 285

Prerequisites . . . 286
Preparing a Cluster . . . 287
Understanding Cluster Groups in Tomcat Environments . . . 287
Setting System Properties for Workflow Engine IDs . . . 287
Installation Procedure. . . . 287
Enabling Permission Index for Clustering . . . 292
Configuring the User Application Driver for Clustering . . . 292
Configuring OSP and SSPR for Clustering . . . 293
Configuring SSPR to Support Clustering. . . . 293
Configuring Tasks on Cluster nodes . . . 293

19 Uninstalling Identity Manager Components 295

Removing Objects from the Identity Vault . . . 295
Uninstalling the Identity Manager Engine . . . 295
Uninstalling the Identity Applications . . . 296
Uninstalling the Identity Reporting Components. . . . 296
Deleting the Reporting Drivers . . . 297
Uninstalling Identity Reporting. . . . 297
Uninstalling Sentinel Log Management for IGA . . . 297
Uninstalling Designer . . . 297
Uninstalling Analyzer . . . 298

20 Troubleshooting **299**

Locating Log Files 299
Troubleshooting Identity Manager Engine 299
Troubleshooting the Identity Applications and Identity Reporting 301
Troubleshooting Login 305
Troubleshooting Installation and Uninstallation 308
Troubleshooting Upgrade 310

About this Book and the Library

The *Setup Guide* provides instructions for installing the NetIQ Identity Manager (Identity Manager) product. This guide describes the process for installing individual components in a distributed environment.

Intended Audience

This book provides information for identity architects and identity administrators responsible for installing the components necessary for building an identity management solution for their organization.

Other Information in the Library

For more information about the library for Identity Manager, see the Identity Manager documentation website.

About NetIQ Corporation

We are a global, enterprise software company, with a focus on the three persistent challenges in your environment: Change, complexity and risk—and how we can help you control them.

Our Viewpoint

Adapting to change and managing complexity and risk are nothing new

In fact, of all the challenges you face, these are perhaps the most prominent variables that deny you the control you need to securely measure, monitor, and manage your physical, virtual, and cloud computing environments.

Enabling critical business services, better and faster

We believe that providing as much control as possible to IT organizations is the only way to enable timelier and cost effective delivery of services. Persistent pressures like change and complexity will only continue to increase as organizations continue to change and the technologies needed to manage them become inherently more complex.

Our Philosophy

Selling intelligent solutions, not just software

In order to provide reliable control, we first make sure we understand the real-world scenarios in which IT organizations like yours operate—day in and day out. That's the only way we can develop practical, intelligent IT solutions that successfully yield proven, measurable results. And that's so much more rewarding than simply selling software.

Driving your success is our passion

We place your success at the heart of how we do business. From product inception to deployment, we understand that you need IT solutions that work well and integrate seamlessly with your existing investments; you need ongoing support and training post-deployment; and you need someone that is truly easy to work with—for a change. Ultimately, when you succeed, we all succeed.

Our Solutions

- Identity & Access Governance
- Access Management
- Security Management
- Systems & Application Management
- Workload Management
- Service Management

Contacting Sales Support

For questions about products, pricing, and capabilities, contact your local partner. If you cannot contact your partner, contact our Sales Support team.

Worldwide:	www.netiq.com/about_netiq/officelocations.asp
United States and Canada:	1-888-323-6768
Email:	info@netiq.com
Website:	www.netiq.com

Contacting Technical Support

For specific product issues, contact our Technical Support team.

Worldwide:	www.netiq.com/support/contactinfo.asp
North and South America:	1-713-418-5555
Europe, Middle East, and Africa:	+353 (0) 91-782 677
Email:	support@netiq.com
Website:	www.netiq.com/support

Contacting Documentation Support

Our goal is to provide documentation that meets your needs. The documentation for this product is available on the NetIQ website in HTML and PDF formats on a page that does not require you to log in. If you have suggestions for documentation improvements, click **comment on this topic** at the bottom of any page in the HTML version of the documentation posted at www.netiq.com/documentation. You can also email Documentation-Feedback@netiq.com. We value your input and look forward to hearing from you.

Contacting the Online User Community

NetIQ Communities, the NetIQ online community, is a collaborative network connecting you to your peers and NetIQ experts. By providing more immediate information, useful links to helpful resources, and access to NetIQ experts, NetIQ Communities helps ensure you are mastering the knowledge you need to realize the full potential of IT investments upon which you rely. For more information, visit https://www.netiq.com/communities/.

Overview of Identity Manager Environment

This guide focuses on the tasks that you must complete in order to install and configure Identity Manager.

If you are new to NetIQ Identity Manager, the information in the below sections will acquaint you with the solution and the components that it comprises. The components that you can download and install is determined by your Identity Manager Edition.

- Brief Introduction of Identity Manager Components
- Functional Architecture

1 Brief Introduction of Identity Manager Components

To cover the varying needs of customers, Identity Manager is available in Advanced and Standard Editions. Each edition comprises of a specific set of functionalities and each functionality is handled by multiple components. Therefore, your Identity Manager implementation can include one or all of the following components depending on your requirements:

- Identity Manager Server
- Identity Applications
- Identity Reporting
- Identity Manager Tools

Figure 1-1 lists the components deployed in an Identity Manager Advanced Edition environment.

Figure 1-1 *Components for Identity Manager Advanced Edition*

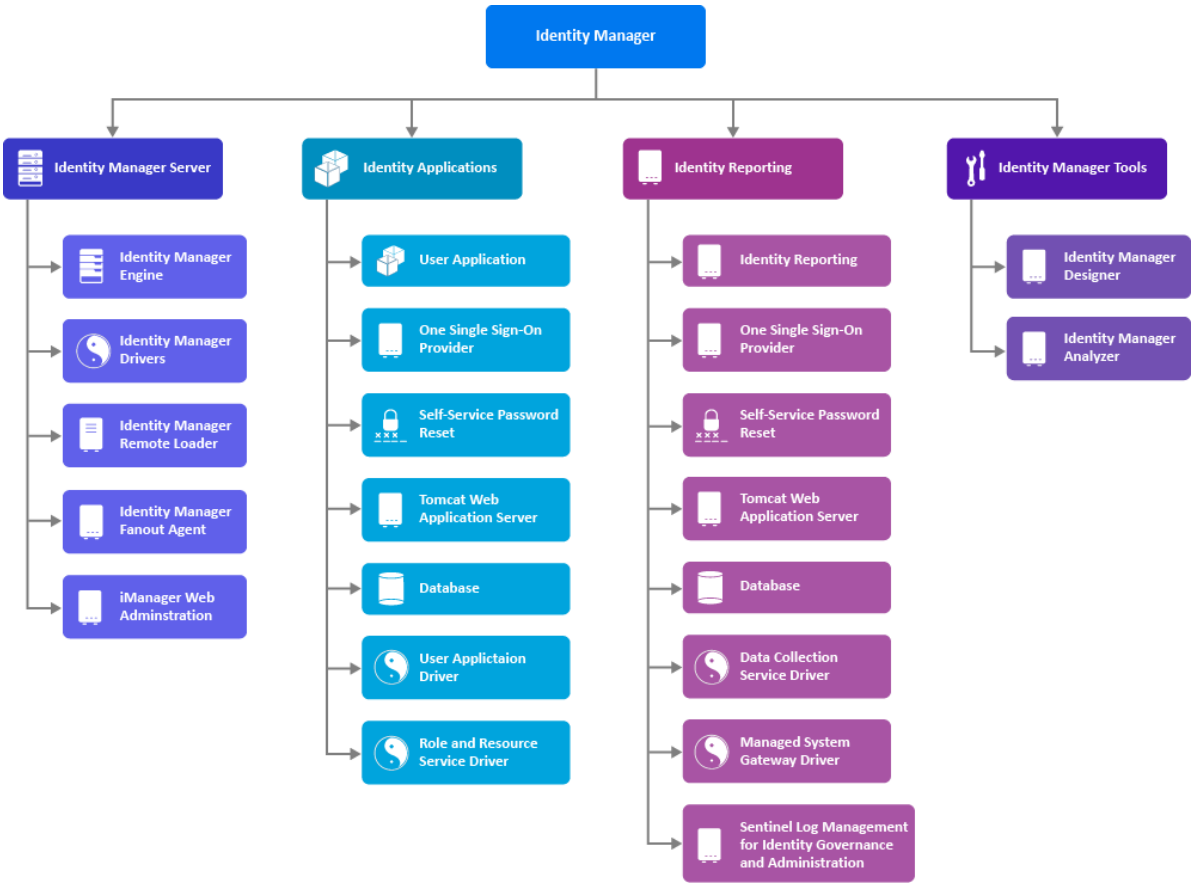
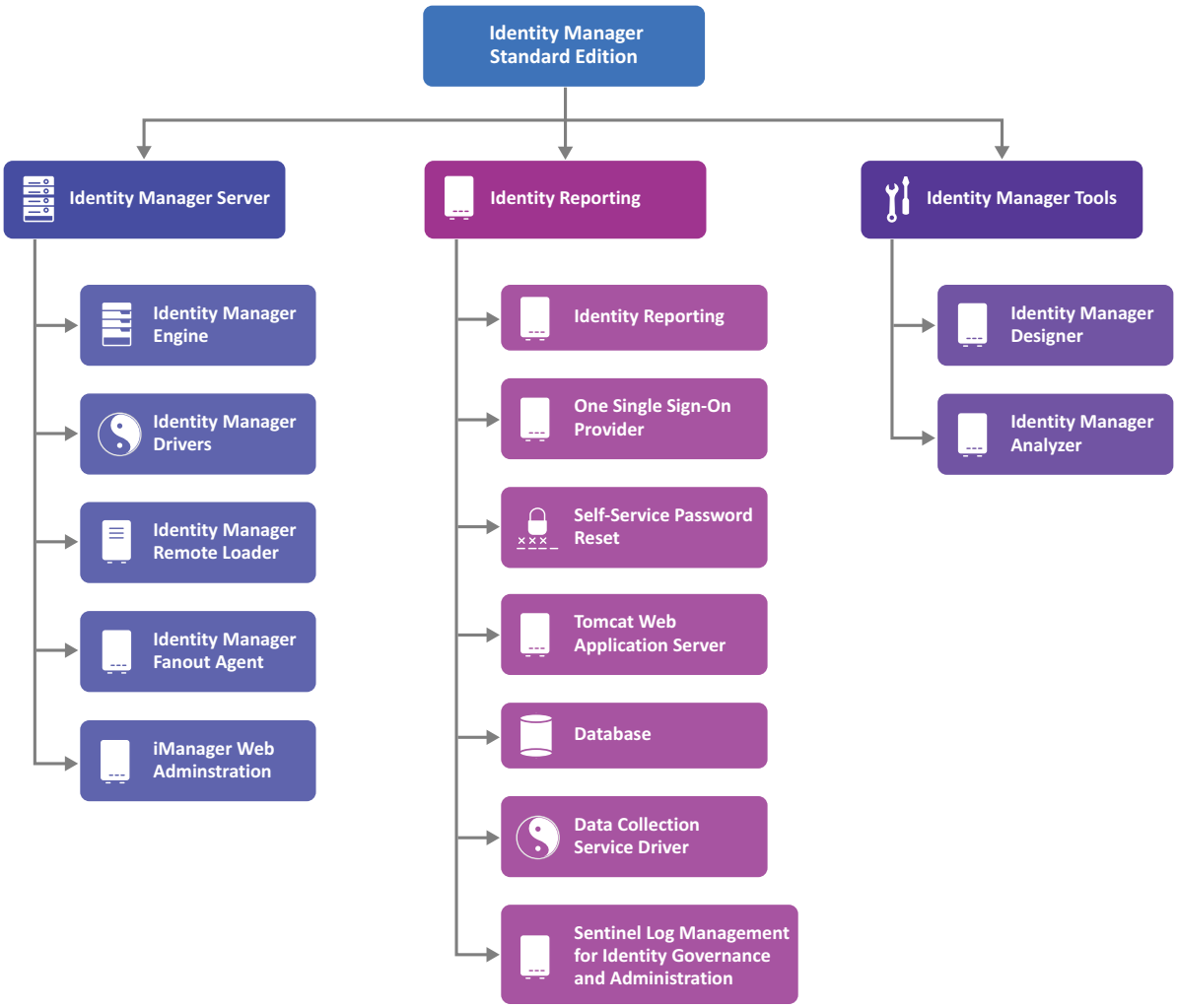


Figure 1-2 lists the components deployed in an Identity Manager Standard Edition environment.

Figure 1-2 *Components for Identity Manager Standard Edition*



Based on how the components interact with each other, some components are logically installed as a group of components. Some components are installed as standalone components to ease the installation experience. For information about how the components interact with each other, see the NetIQ Identity Manager Overview and Planning Guide.

Review the information from the subsequent sections to understand how the components are grouped and how each component or a group of components is installed.

Identity Manager Server Components

Required for all installations

An Identity Manager Server installation comprises of the following components.

Identity Manager Server

The Identity Manager Server executes tasks within Identity Manager. It comprises of Identity Vault, Identity Manager Engine, and Identity Manager drivers.

To support the Identity Manager Server operations, the installation program installs a supported version of Oracle Java Runtime Environment (JRE). To install the Identity Manager Server components, use the **Identity Manager Engine** installation option of the installation program.

Identity Vault

When you install Identity Manager Engine, the installation process creates and configures a connection to Identity Vault. Identity Manager uses Identity Vault as the default repository of all identity data. Identity data includes current state of managed identities, including user account and organizational data.

Identity Manager Engine

The Identity Manager engine processes all data changes that occur in the Identity Vault or a connected application. The server on which the Identity Manager engine runs is referred to as the Identity Manager server.

Identity Manager Drivers

The Identity Manager Server handles provisioning of users, and manages connected system accounts and groups through drivers. A driver is the software interface to a connected system.

Identity Manager Drivers run as part of the Identity Manager Server architecture. A driver acts as a gateway to a native endpoint type system technology. For example, computers running Active Directory Services can be managed only if the Active Directory driver is installed either on the Identity Manager server or the target application server with which the Identity Manager server can communicate. Drivers manage the objects that reside on the connected systems. Managed objects include accounts, groups, and optionally, endpoint-type specific objects.

A driver translates Identity Manager Engine actions into changes on the connected system, such as “Create a new email account on a Microsoft Exchange connected system.” Every driver that is configured in Identity Manager has an associated event cache file (TAO file). Events are cached in the cache file before a driver processes them. By default, the cache files are placed in Identity Vault’s DIB (Data Information Base) directory.

Identity Manager provides several in-built drivers (Java, native, .NET) to manage connections with different types of connected systems. Identity Manager also provides the ability to develop a custom driver to enable data synchronization to a variety of other systems such as a home-grown application or a repository that has no technology interface and cannot leverage out-of-box drivers.

Remote Loader

Drivers can be installed locally on the Identity Manager Server or with a Remote Loader. A Remote Loader loads drivers and communicates with the Identity Manager engine on behalf of drivers installed on remote servers. If the application runs on the same server as the Identity Manager engine, you can install the driver on that server. However, if the application does not run on the

same server as the Identity Manager engine, you must install the driver on the application's server. To help with the workload or configuration of your environment, you can install Remote Loader on a server separate from the servers that have Tomcat and the Identity Manager server. For more information about Remote Loader, see Determining When to Use the Remote Loader in the NetIQ Identity Manager Driver Administration Guide.

Use the **Identity Manager Remote Loader Server** installation option to install the Remote Loader service and the driver instances in the Remote Loader.

Fanout Agent

Identity Manager Fanout Agent is an installation component used by Java Database Connectivity (JDBC) Fanout driver to create multiple JDBC Fanout driver instances. The Fanout driver provisions users, groups, and password to multiple databases with minimal effort. This eliminates the need for the Identity Manager administrator to configure multiple JDBC drivers using the same policies to provision multiple databases of the same type. You can centrally manage user accounts and have them automatically created, configured, maintained, and removed when appropriate. For more information, see the *NetIQ Identity Manager Driver for JDBC Fanout Implementation Guide*.

To install Fanout Agent, use the **Identity Manager Fanout Agent** installation option of the installation program.

iManager

NetIQ iManager is a browser-based tool that provides a single point of administration for many Novell and NetIQ products, including Identity Manager. You can use iManager to perform administrative tasks such as managing Identity Manager Server options or driver attributes, which you cannot manage in Identity Manager Identity Applications. For more information about iManager, see the *NetIQ iManager Administration Guide*. After you install the Identity Manager plug- ins for iManager, you can manage Identity Manager and receive real-time health and status information about your Identity Manager system.

With iManager, you can perform similar tasks as performed with Designer and also monitor the health of your system. NetIQ recommends that you use iManager for administrative tasks. Use Designer for configuration tasks that require changes to packages, modeling, and testing prior to deployment.

Identity Manager requires the installation of Identity Manager plug-ins with iManager. Identity Manager provides a single installer to install the iManager client and Identity Manager plug-ins. You can install iManager on the Identity Manager server or on a separate computer.

To install iManager, use the iManager Web Administration installation option of the installation program.

TIP: After learning about the components, you must develop a good understanding of how they are installed and configured for use in a production environment.

Identity Applications Components

Required for Advanced Edition installation

Identity Applications are an interconnected set of browser-based Web applications. They enable your organization to manage the user accounts and permissions associated with the wide variety of roles and resources available to users. You can configure the identity applications to provide self-service support for your users, such as requesting roles or changing their passwords. You can also set up workflows to improve the efficiency in managing and assigning roles and resources. Identity Applications consists of Administration Console (for administration tasks), User Console (Dashboard), and REST services that help you perform these tasks.

NOTE: You must have the Identity Manager Engine installed before installing Identity Applications.

To install Identity Applications components, use the **Identity Applications** installation option of the installation program.

An Identity Applications installation comprises of the following components:

User Application

The User Application is a browser-based web application that gives users the ability to perform a variety of identity self-service and roles provisioning tasks. Some of the tasks that were performed by using the User Application interface in the previous versions of the product have been moved to the new user interface that includes an Administration Console and a User Console. The User Application continues to provide some of the functionality that does not yet exist in the new user interface. For more information, see the NetIQ Identity Manager - Administrator's Guide to the Identity Applications.

Authentication Service

The authentication service provides access to Identity Applications features. For more information about using Single Sign-on access in Identity Manager, see the NetIQ Identity Manager - Administrator's Guide to the Identity Applications.

The authentication service is provided by the NetIQ One Single Sign-On Provider (OSP) component. Identity Applications requires a local installation of OSP. OSP is automatically installed with Identity Applications.

Self-Service Password Reset

The self-service password management service provides access to self-service password management. Identity Applications include NetIQ Self Service Password Reset (SSPR) to help users who have access to the identity applications to reset their passwords without administrative intervention.

The Identity Applications installation process enables SSPR by default. However you can choose to install SSPR on a separate computer if your deployment needs it or if you are installing Standard Edition. When installing SSPR on a separate computer in Advanced Edition, you must define

password management settings in the Identity Applications configuration file (`ism-configuration.properties`) after completing the installation of both components, either manually or by using the ConfigUpdate utility.

Web Application Server

The application server provides the runtime framework in which the identity applications components execute. The identity applications are packaged as WAR (Web Application Resource or Web application ARchive) files. The installation process enables you to deploy the WAR files to the application server. The application server runs a Java™ virtual machine, providing the runtime environment for the application code. The following WAR files apply to the URL for a component of the identity applications:

- **IDMProv** for the Application Programming Interfaces (APIs) to the User Application
- **idmdash** for the Dashboard
- **idmadmin** for Identity Applications Administration interface

When a user interacts with `idmdash` or `idmadmin` applications, these applications query the underlying `IDMProv.war` file and fetch the information for the user. `IDMProv.war` exposes the REST and SOAP APIs where `idmdash` and `idmadmin` contain the information that provides the user interface.

The identity applications run on an Apache Tomcat application server, included in the installation kit. To support the Tomcat application server, the installation program installs supported versions of JRE and Apache ActiveMQ.

Identity Applications Database

The Identity Applications database maintains configuration data for the identity applications such as localized labels, entitlement values, and Email server configuration. It also stores workflow state data required by the Workflow Engine. The supported databases for Identity Applications are PostgreSQL, Oracle, and Microsoft SQL Server.

The Identity Applications installation program automatically installs a supported version of PostgreSQL database that acts as the default database for Identity Applications. If you do not want to use PostgreSQL as the database, you can configure a supported version of Oracle or MS SQL database with Identity Applications. Identity Applications require a Java Database Connectivity driver (JDBC type 4 driver) to communicate with the database. The installation program prompts for the location and name of the JDBC driver for the database. Therefore, you must obtain this JDBC driver from your database installation directory before starting the Identity Applications installation. The supported databases for Identity Reporting are PostgreSQL, Oracle, and MS SQL.

- For PostgreSQL database, the driver is bundled with the Identity Manager installation program.
- For Oracle database, you can download the driver from the Oracle web site.
- For Microsoft SQL Server database, download the driver from the Microsoft web site.

The database can reside locally on the Identity Applications server or a remote computer. When using a remote database, you must configure a connection to the database.

Drivers for Identity Applications

The Identity Applications components require the following drivers:

User Application Driver

Stores configuration information and notifies the Identity Applications whenever changes occur in the Identity Vault. You can configure the driver to allow events in the Identity Vault to trigger workflows. The driver can also report success or failure of a workflow's provisioning activity to the User Application so that users can view the final status of their requests.

Role and Resource Service Driver

Manages all role assignments, starts workflows for role assignment requests that require approval, and maintains indirect role assignments according to group and container memberships. The driver grants and revokes entitlements for users based on their role memberships, and it performs cleanup procedures for requests that have been completed. The driver also maintains resource requests in addition to role requests.

The **Identity Applications** installation option of the installation program deploys the User Application driver and the Role and Resource Service driver to the Identity Vault.

Identity Reporting Components

(Optional) Install this component only if you plan to implement the reporting functionality

Identity Reporting gives you a complete view of your users' entitlements, providing the knowledge you need to see the past and present state of authorizations and permissions granted to identities in your organization. Identity Manager provides predefined reports that you can use to monitor the status of an Identity Manager environment, including information collected from Identity Vaults and connected systems. To use the reports provided with Identity Manager, you install Identity Reporting, which is included with Identity Manager. Identity Reporting also includes a report packaging tool that facilitates the process of creating custom reports. The user interface for Identity Reporting makes it easy to schedule reports to run at off-peak times for optimized performance. For more information about Identity Reporting, see the Administrator Guide to NetIQ Identity Reporting.

NOTE: You must install Identity Applications before you install Identity Reporting in an Advanced Edition.

An Identity Reporting installation comprises of the following components:

Identity Reporting

Browser-based application that generates reports by making calls to the reporting service. The reporting service retrieves the data needed to generate reports from the Identity Reporting repository (Identity Information Warehouse), which contains all report management information (such as report definitions and schedules), database views, and configuration information required for reporting.

Authentication Service

The authentication service is provided by the OSP component. For more information, see "Authentication Service" on page 21.

NOTE: OSP is automatically installed with Identity Reporting. However, in an Advanced Edition installation, Identity Reporting can use the same authentication service that is installed with Identity Applications. When using the same authentication service, you must specify the authentication settings during the Identity Reporting configuration.

Self-Service Password Reset

The self-service password management service provides access to self-service password management. For more information, see "Self-Service Password Reset" on page 21.

Identity Reporting Database

The Identity Reporting database (Identity Information Warehouse) stores information about the actual and desired states of the Identity Vault and the connected systems within your organization. You can generate reports from this information to view the relationship between objects, such as users and roles. The database can reside locally on the Identity Reporting server or on a remote computer. Identity Manager uses data sources to connect to the database. Identity Reporting requires a Java Database Connectivity driver (JDBC type 4 driver) to communicate with the database. A JDBC driver enables an Identity Reporting server to communicate with the data source. The supported databases for Identity Reporting are PostgreSQL, Oracle, and Microsoft SQL.

- For PostgreSQL database, this driver is bundled with the Identity Manager installation program.
- For Oracle database, you can download the driver from the Oracle web site.
- For Microsoft SQL Server database, download the driver from the Microsoft web site.

NOTE: You must have the Identity Manager Server installed before installing the Identity Reporting components.

Web Application Server

The application server provides the runtime framework in which the identity reporting components execute. The following WAR files apply to the URL for a component of identity reporting:

- **IDMRPT** for the Identity Reporting application/interface
- **idmdcs** for Identity Manager Data Collection Service

When a user interacts with `IDMRPT` or `idmdcs` applications, these applications query the reporting service and fetch the information for the user. The reporting service exposes the REST APIs where `IDMRPT` and `idmdcs` contains the information that provides the user interface.

For more information on Web Application Server, see "Web Application Server" on page 22.

Drivers for Identity Reporting

The Identity Reporting components require the following drivers:

Managed System Gateway Driver

Queries the Identity Vault to collect the following type of information from managed systems:

- List of all managed systems
- List of all accounts for the managed systems
- Entitlement types, values, and assignments, and user account profiles for the managed systems

Data Collection Service Driver

The Data Collection Service uses the Data Collection Services driver to capture changes to objects stored in an Identity Vault, such as accounts, roles, resources, groups, and team memberships. The driver registers itself with the service and pushes change events (such as data synchronization, add, modify, and delete events) to the service.

The service includes three subservices:

- **Report Data Collector:** Uses a pull design model to retrieve data from one or more Identity Vault data sources. The collection runs on a periodic basis, as determined by a set of configuration parameters. To retrieve the data, the collector calls the Managed System Gateway driver.
- **Event-Driven Data Collector:** Uses a push design model to gather event data captured by the Data Collection Service driver.
- **Non-Managed Application Data Collector:** Retrieves data from one or more non-managed applications by calling a REST end point written specifically for each application. Non-managed applications are applications within your enterprise that are not connected to the Identity Vault.

The **Identity Reporting** installation option of the installation process deploys the Managed System Gateway driver and the Data Collection Service driver to the Identity Vault.

Sentinel Log Management for Identity Governance and Administration

Sentinel Log Management for Identity Governance and Administration (IGA) is a Security Information and Event Management (SIEM) system that receives information from many sources throughout an enterprise, standardizes it, prioritizes it and presents it to you to make threat, risk and policy related decisions. Sentinel Log Management for (IGA) captures log events associated with actions performed in several NetIQ products, including Identity Reporting, Identity Applications, and the Identity Vault. These events are stored in the public schema within the Identity Reporting repository (Identity Information Warehouse).

Identity Manager provides a separate installation program (`SentinelLogManagementForIGA8.2.2.0.tar.gz`) for Sentinel Log Management for IGA.

Identity Manager Tools

Required for all installations

Identity Manager includes a set of management tools to facilitate the implementation, customization and maintenance of the solution. Some of these tools are installed with Identity Manager and some must be installed separately.

Designer for Identity Manager

Designer for Identity Manager (Designer) helps you design, test, document, and deploy Identity Manager solutions in a network or test environment. You can configure your Identity Manager project in an off-line environment, and then deploy to your live system. From a design perspective, Designer helps do the following:

- Graphically view all of the components that comprise your Identity Manager solution and observe how they interact.
- Modify and test your Identity Manager environment to ensure it performs as expected before you deploy part or all of your test solution to your production environment.

Designer keeps track of your design and layout information. With a click of a button, you can print that information in a format of your choice. Designer also enables teams to share work on enterprise-level projects.

Identity Manager provides a separate installation program for Designer.

Analyzer for Identity Manager

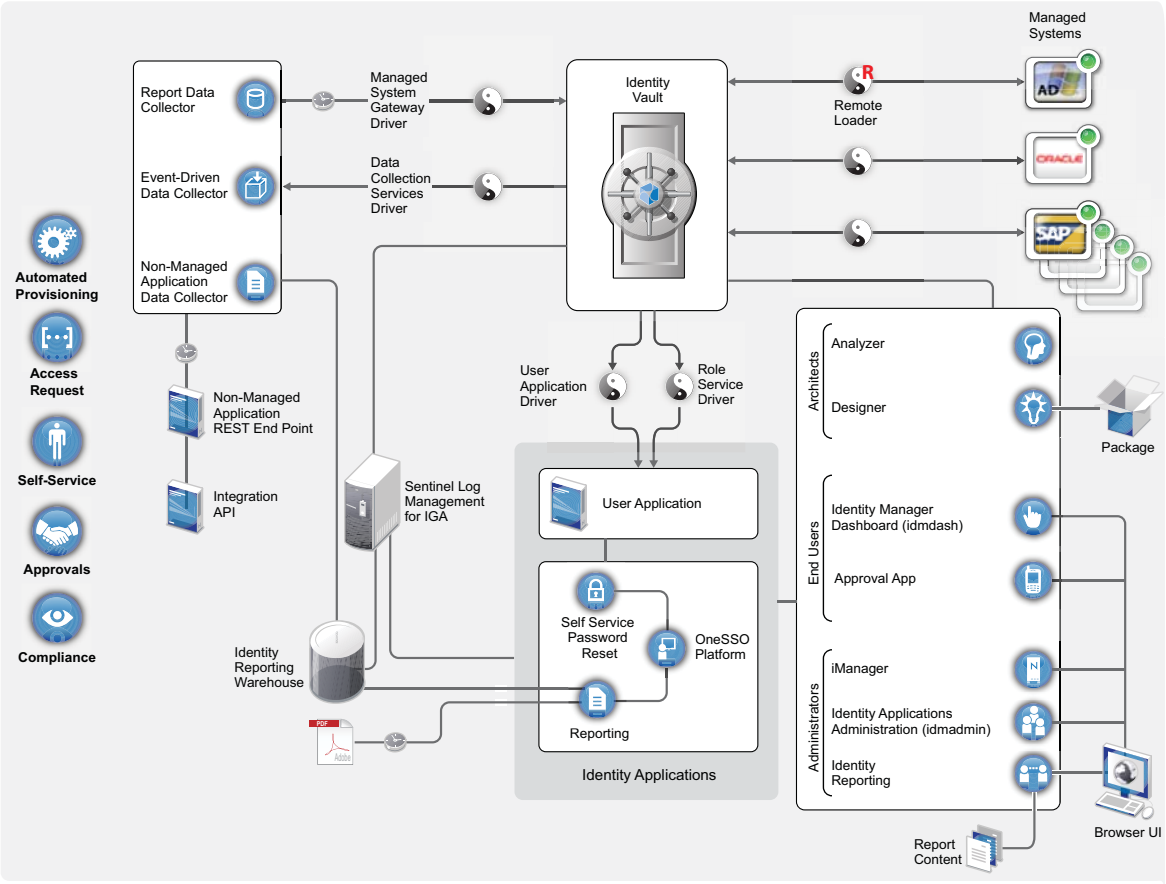
Analyzer for Identity Manager (Analyzer) provides data analysis, cleansing, reconciliation, and reporting to help you adhere to internal data quality policies. Analyzer lets you analyze, enhance, and control all data stores throughout the enterprise. Analyzer includes the following features:

- Analyzer's schema map associates an application's schema attributes to the corresponding schema attributes in Analyzer's base schema. This lets you ensure that your data analysis and cleaning operations properly associate similar values between the disparate systems. To accomplish this, Analyzer leverages the schema mapping features in Designer.
- The Analysis Profile editor lets you configure a profile for analyzing one or more data set instances. Each analysis profile contains one or more metrics against which you can evaluate attribute values to see how the data conforms to your defined data format standards.
- The Matching Profile editor lets you compare values in one or more data sets. You can check for duplicate values within a specified data set and check for matching values between two data sets.

Identity Manager provides a separate installation program for Analyzer.

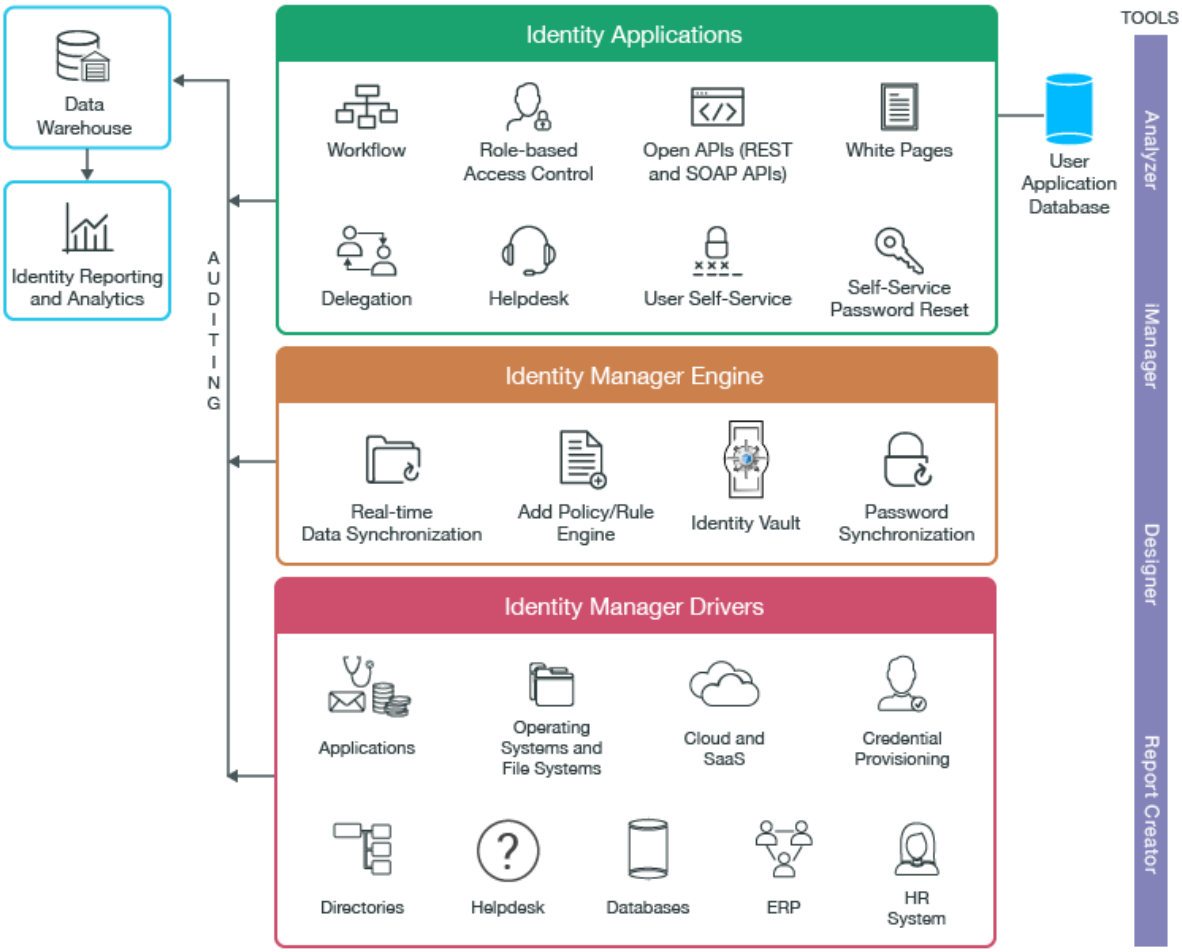
After understanding the purpose of different Identity Manager components and the way they are installed, see Figure 1-3 to understand how the components interact with each other.

Figure 1-3 Interaction of Identity Manager Components



Functional Architecture

The following illustration depicts the basic functional architecture for Identity Manager components. This illustration does not cover all possible integrations.



For information about the possible deployment scenarios, see Deployment Options for Identity Manager.

Deployment Options for Identity Manager

Consult the following table to plan the physical environment for your Identity Management solution. These deployment use cases provide an overview of the Identity Management physical architecture and how the component products are connected and communicate with each other and other products. For an introductory overview of the Identity Management functional architecture and the components, see "Functional Architecture" on page 27.

Deployment Option	Summary
Single-server configuration on one computer	The most basic deployment configuration includes Identity Manager server and other required applications on one computer. You must ensure that the computer has the required memory, speed, and available disk space to meet the workload. This is a basic deployment use case and mostly suited for Proof-of-Concept (POC) and demonstration purposes only. It might not be appropriate for a production environment.

Deployment Option	Summary
Distributed server configuration	This deployment has Identity Manager server on one computer and all other required applications on one or more additional computers. For example, components such as identity applications, iManager, OSP, and SSPR can run on a separate computer. You can include an additional computer to host the components for reporting service to suffice the system requirements for running the Sentinel Log Management for IGA component.
High-availability deployment	High availability is a redundancy operation that automatically switches to a standby server if the primary server fails or is temporarily shut down for maintenance. Identity Manager supports installing the following components in a high-availability environment: ◆ Identity Vault ◆ Identity Manager engine ◆ Remote Loader ◆ Identity applications, except Identity Reporting A typical cluster configuration contains Tomcat Application Server nodes hosting the Identity Applications for load balancing and fault tolerance. All the communication is routed through the load balancer. All nodes communicate to the same instance of the Identity Vault and the Identity Applications database. This configuration is scalable. You can easily increase the number of nodes to handle the load.

Sample Identity Manager Deployments

Identity Manager allows you to control user identities and their access to applications and accounts on connected systems. Based on the functionality you need, select which Identity Manager Edition to install, which in turn determines the components to install. The following table lists the features provided by Identity Manager Advanced Edition and Identity Manager Standard Edition.

Feature	Advanced Edition	Standard Edition	Components to Install
Rule-based automated user provisioning	✓	✓	Identity Manager Engine and Designer
Real-time identity synchronization	✓	✓	Identity Manager Engine and Designer
Password management and password self-service	✓	✓	Identity Manager Engine and SSPR
Uniform identity information tool (Analyzer)	✓	✓	Analyzer
REST APIs and single sign-on support	✓	✓ (limited support only)	Identity Manager Engine, OSP, and Identity Reporting

Feature	Advanced Edition	Standard Edition	Components to Install
Current state reporting	✓	✓	Identity Manager Engine and Identity Reporting
Role-based enterprise-level provisioning	✓	✗	Identity Manager Engine and Identity Applications
Automated approval workflows for business policy enforcement	✓	✗	Identity Manager Engine, Designer, and Identity Applications
Advanced self-service in the identity applications	✓	✗	Identity Manager Engine and Identity Applications
Resource model and catalog for easy resource provisioning	✓	✗	Identity Manager Engine and Identity Applications
Historical state reporting	✓	✗	Identity Manager Engine and Identity Reporting
Connected systems reporting	✓	✗	Identity Manager Engine and Identity Reporting
Role and resource administration	✓	✗	Identity Manager Engine and Identity Applications

NOTE: In all Identity Manager installations, Identity Manager Server is the central component. Depending on the Identity Manager edition, only Identity Reporting or both Identity Reporting and Identity Applications are installed on a Tomcat application server. Use the Identity Manager component-specific installer to install other components as needed. For example, install Designer, Analyzer, or Sentinel Log Management for Identity Governance and Administration.

In addition, review the goals for your implementation and pay attention to the physical topology options, such as high availability and scalability before installing Identity Manager. This helps you identify the configuration that matches your organization's requirements.

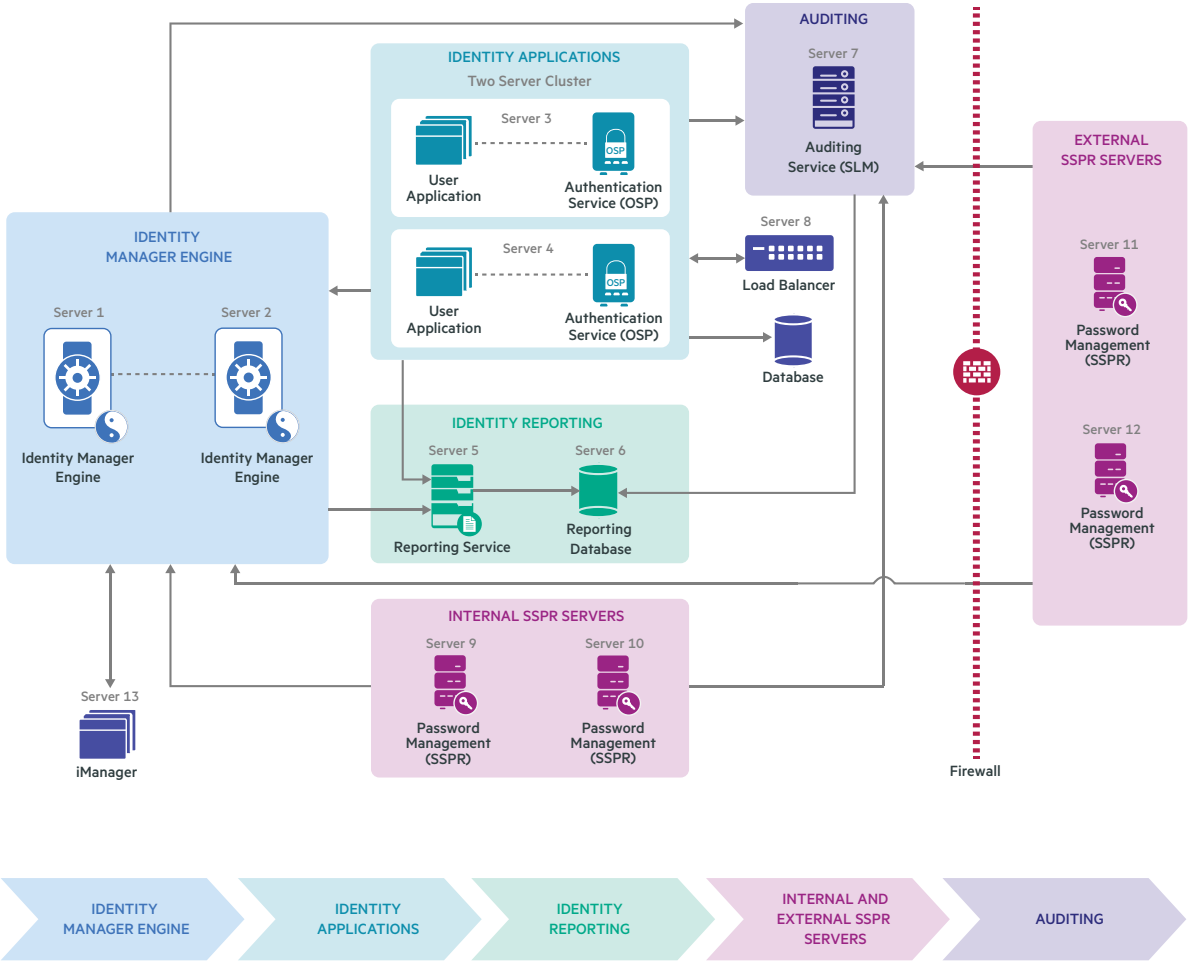
High availability ensures efficient manageability of critical network resources including data, applications, and services. You can implement high availability by reducing any single points-of-failure and by using redundant components. Similarly, connecting multiple instances of identity management components with a load balancer can provide a highly available environment.

This section describes two examples to illustrate Advanced Edition and Standard Edition implementations at a high level. You can use them as a reference to come up with a deployment diagram for your implementation.

Sample Advanced Edition Deployment

Figure 1-4 shows a high-level deployment topology of an Identity Manager Advanced Edition installation.

Figure 1-4 *Sample Advanced Edition Deployment*



- Identity Manager Server components and its underlying repository (Identity Vault) and Web-enabled components (Identity Applications and Identity Reporting) are installed in the intranet zone. The load balancer then routes the traffic to the Identity Applications components. This deployment provides enhanced security because these components are separated from Internet traffic by firewalls.
- The Identity Manager Server components are configured to use a two server (primary/secondary) configuration. A virtual logical IP address is active on the primary server, which acts as the primary (active) node and another server acts as the secondary node. If the primary server fails, the logical IP address is moved to the secondary server. All the processes are then started on the secondary server. The application processes accessing the secondary server may experience a temporary loss of service when the logical IP address is moved over, and all other processes are started. All the components use the same Identity Vault server at any point of time.

- SSPR services are available inside and outside the firewall to address the password management needs of local and mobile users of the organization. The services installed inside the firewall address the local password management needs. In case of forgotten password, the mobile workforce cannot access VPN which will prevent them from accessing the internally placed SSPR services. They can directly access the SSPR services placed outside the firewall to manage their passwords.
- User Application and authentication service (OSP) are deployed in a cluster to handle the load and support the failover process for Identity Applications. The cluster nodes are attached to the same Identity Applications database that is installed on a separate computer. This deployment provides increased scalability by allowing you to add more nodes to the cluster. The cluster configuration is immediately sent to the newly added nodes. The load balancer is typically part of the cluster. It understands the cluster configuration as well as failover policies. In this configuration, all the cluster nodes are active at any point of time. The load balancer distributes the load across the nodes to ensure that the nodes have roughly the same workload. If a node fails, it diverts the requests made to that node to the surviving nodes in the cluster. Because this installation is an intrasite, high availability solution, it provides protection from local hardware and software failures, using a two node hardware-based cluster to achieve high availability for Identity Applications components.

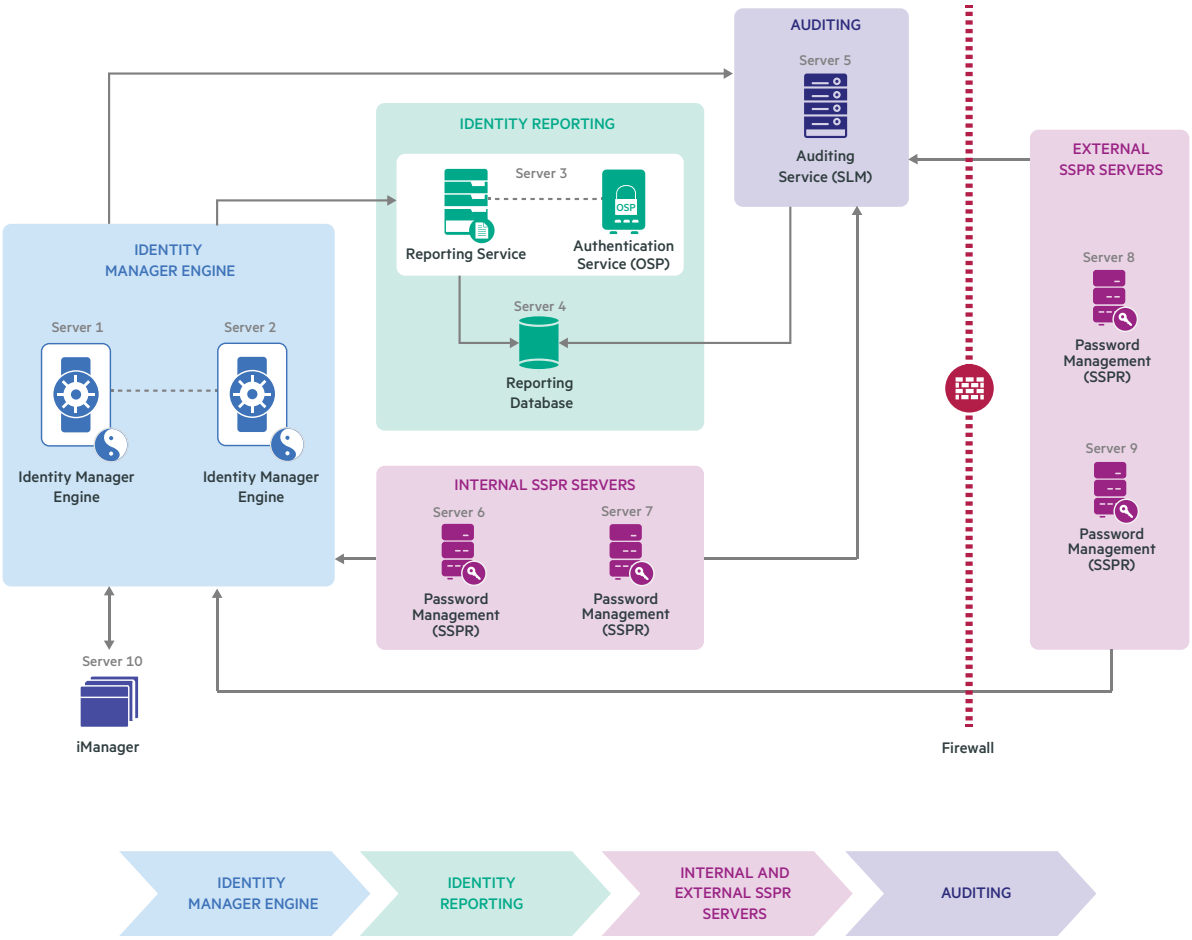
NetIQ has tested and recommends this configuration.

NOTE: Identity Manager does not support clustering the Identity Reporting components.

Sample Standard Edition Deployment

In production deployments, security policies might specify to not expose the authentication service that provides advanced authentication and protection for your environment to the public network. Figure 1-5 shows a high-level deployment topology of an Identity Manager Standard Edition installation .

Figure 1-5 Sample Standard Edition Deployment



- Identity Manager Server components and its underlying repository (Identity Vault) and Identity Reporting components are installed in the intranet zone. Internet Web traffic is routed to the Identity Reporting components through the Web servers that are installed behind the firewall for added protection. This deployment provides enhanced security because these components are separated from Internet traffic by firewalls.
- The Identity Manager Server components are configured to use a two-server (primary/secondary) configuration. A virtual logical IP address is active on the primary server, which acts as the active node while another server acts as the secondary node. If the primary server fails, the logical IP address is moved to the secondary server. All the processes are then started on the secondary server. The application processes accessing the secondary server may experience a temporary loss of service when the logical IP address is moved over, and all other processes are started. All the components use the same Identity Vault server at any point of time.
- SSPR services are available inside and outside the firewall to address the password management needs of local and mobile users of the organization. The services installed inside the firewall address the local password management needs. In case of forgotten password, the mobile workforce cannot access VPN which will prevent them from accessing the internally placed SSPR services. They can directly access the SSPR services placed outside the firewall to manage their passwords.

NetIQ has tested and recommends this configuration.

NOTE: Identity Manager does not support clustering the Identity Reporting components.

II Planning to Install Identity Manager

Planning your Identity Manager implementation depends on how you want Identity Manager to manage users and what functionality you need to accomplish your business goals. Consider the following points to help you make decisions:

- How do I manage identities.
- Do I need automated provisioning.
- Which business requirements should I implement using workflow.

The result of your decisions will determine the best way to implement Identity Manager for your requirements.

There are additional tasks that require planning before deploying Identity Manager in a large enterprise. For more information, refer to the Planning section of the NetIQ Identity Manager Overview and Planning Guide.

2 Planning Your Installation

The following table lists the components to install to support the functionality that you want to implement. For instructions on installing these components, see the Installation section.

Functionality	Component to Install
Manage user identities in a corporate directory	Identity Manager Server
Provision accounts in connected systems	Identity Manager Server
	Identity Applications
	User Application Driver
	Role and Resource Service Driver
	Designer
	NOTE: For instructions on installing Identity Manager drivers, see the driver implementation guide for the type of driver that you want to install on the Identity Manager Drivers Documentation Website.
Authentication	Identity Manager Server
	One Single Sign-On Provider
Password management	Identity Manager Server
	Self Service Password Management
Generate reports on Identity Manager activity	Identity Manager Server
	Identity Reporting
	One Single Sign-On Provider

Implementation Checklist

Use the following checklist to plan, install, and configure Identity Manager.

	Checklist Items
❑	1. Review the product architecture information to learn about Identity Manager components. For more information, see How Identity Manager Works in *NetIQ Identity Manager Overview and Planning Guide*.

	Checklist Items
❑	2. Review the Identity Manager licensing information to determine whether you need to use the evaluation license or the enterprise license of Identity Manager. For more information, see Understanding Licensing and Activation in *NetIQ Identity Manager Overview and Planning Guide*.
❑	3. Determine the type of deployment suitable for your environment based on the features you want to implement. For more information, see Identity Manager Deployment Configurations in *NetIQ Identity Manager Overview and Planning Guide*.
❑	4. Determine whether you can run the installation programs in your preferred language. For more information, see Understanding Identity Manager Localization in *NetIQ Identity Manager Overview and Planning Guide*.
❑	5. Locate the files for installation. For more information, see Where to Get Identity Manager in *NetIQ Identity Manager Overview and Planning Guide*.
❑	6. Install Identity Manager. For more information, see Part III, "Installing and Configuring Identity Manager Components," on page 63.
❑	7. Configure the installed components. For more information, see Chapter 5, "Configuring the Identity Manager Components," on page 77.
❑	8. Perform additional configuration steps for different components to be fully functional. For more information, see Chapter 6, "Final Steps for Completing the Installation," on page 93.

NOTE: For cluster and cloud deployments, ensure that you review the recommended configuration details and the requirements.

- "Deploying Identity Manager for High Availability" on page 269
- "Deploying Identity Manager on AWS EC2" on page 243

Recommended Installation Scenarios and Server Setup

This section helps you determine the installation order and server setup in a single-server or in a distributed environment.

- "Deciding When to Install SLM for IGA" on page 39
- "Considerations for Installing in a Distributed Setup" on page 39

Deciding When to Install SLM for IGA

Sentinel is the preferred audit event destination for Identity Manager. Identity Manager provides event forwarding capabilities to Sentinel by configuring Sentinel Link using Sentinel Event Source Management (ESM). If you are already using Sentinel for auditing or as an integration framework for tracking identities, you might choose to use your existing Sentinel for auditing events instead of installing SLM for IGA.

Regardless of whether you choose to reuse your existing Sentinel server or perform a new installation of SLM for IGA shipped with Identity Manager, you must configure the Sentinel server as a source of audit data. You do this by creating a data synchronization policy on the Sentinel server in the Identity Manager Data Collection Services page for auditing events. For more information, see About the Data Sync Policies tab in the *Administrator Guide to NetIQ Identity Reporting*.

Considerations for Installing in a Distributed Setup

Review the following considerations to help you plan your installation:

Component Stickiness

Component	Independent Installation	Notes
Identity Manager Engine	Yes	
Identity Applications	Yes	Must have its own OSP. Identity Applications and OSP must be installed on the same computer. **IMPORTANT:** Identity Manager does not support a remotely installed OSP. If you are upgrading to this version, you must use OSP that is installed with Identity Applications upgrade and then copy the OSP settings from your existing OSP server to the new server where OSP are installed. For more information, see "Post-Upgrade Tasks for Identity Applications Components" on page 214.
Identity Reporting	Yes	Can have its own OSP. The installer supports a locally or a remotely installed OSP for installing or upgrading Identity Reporting.
OSP	No	The installer does not support a remotely installed OSP for Identity Applications. You must install OSP and Identity Applications on the same computer. **IMPORTANT:** If you are upgrading to this version, you must use OSP that is installed with Identity Applications upgrade and then copy the OSP settings from your existing OSP server to the new server where OSP is installed. For more information, see "Post-Upgrade Tasks for Identity Applications Components" on page 214.

Component	Independent Installation	Notes
SSPR	Yes	The installer supports a standalone installation and an upgrade of SSPR. If you are installing or upgrading SSPR in a Standard Edition, NetIQ recommends that you install or upgrade SSPR on a standalone server. In other words, SSPR must not be installed on the same server as Identity Reporting. **IMPORTANT:** If you are upgrading to this version where Identity Applications and SSPR are deployed on different servers, and you want to restore the existing SSPR settings to the new server where SSPR is installed, ensure that you modify the SSPR settings on the new SSPR server by using the ConfigUpdate utility. For more information, see "Post-Upgrade Tasks for Identity Applications Components" on page 214.
Identity Applications Database	Yes	
Reporting Database	Yes	
Sentinel Log Management for Identity Governance and Administration (Sentinel Log Management)	Yes	

Server Setup

In a typical production environment, you might install Identity Manager on seven or more servers, as well as on client workstations. For example:

Computer setup	Component setup
All in One (Only recommended for demo / POC setup)	Install and configure all components on one computer (Identity Manager Engine, Identity Applications, Identity Reporting, OSP, SSPR, Identity Applications Database, and Reporting Database) and Sentinel Log Management on a separate computer.
Distributed setup	
Server 1	◆ Identity Vault ◆ Identity Manager Engine
Server 2	Identity Applications and OSP (can be clustered)
Server 3	Identity Reporting (OSP)
Server 4	SSPR

Computer setup	Component setup
Servers 5 and 6	Identity Manager databases for: ◆ Identity applications ◆ Identity Reporting
Server 7	Sentinel Log Management

NOTE: From the 4.7 release onward, installing Identity Manager on a server with multiple instances of Identity Vault is no longer supported.

Determine Hardware Requirements

The hardware that you need for your Identity Manager installation is governed by two factors:

- Functionality that you want to implement
- Size of your deployment

The following deployment types can help you estimate the size of the deployment.

Type Of Deployment	Hardware Requirements
Proof of Concept (demonstration)	A single server deployment for use in demonstrations or basic testing in a development environment.
Basic	A multi-server implementation that is suitable for small to medium size implementations. This type of implementation requires one server for running Identity Manager Server and its components and two additional servers for running the Identity Applications and Identity Reporting components.
Intermediate	A high availability implementation that is suitable for medium size implementations.
Large Enterprise	A high availability implementation that includes Identity Manager engine cluster to provide failover capabilities and another cluster of Identity Applications and authentication service to support single sign-on access (OSP on Windows) and load balancing and fault tolerance.

System Requirements Worksheet

For information about the recommended hardware, supported operating systems, and supported virtual environments, see the System Requirements for Identity Manager 4.8.

For information about system requirements for a specific release, see the Release Notes accompanying the release at the Identity Manager documentation website.

An Identity Manager implementation can vary based on the needs of your IT environment, so you should contact NetIQ Consulting Services or any of the NetIQ Identity Manager partners prior to finalizing the Identity Manager architecture for your environment.

Installing Identity Manager on SLES Servers

- Ensure to install the following dependent libraries or RPMs before installing Identity Manager:
  - `libgcc*.x86_64`
  - `gettext.x86_64`
  - `unzip`
  - `bc`
  - `lsof`
  - `net-tools`
  - `openssl`
  - `pam-modules`
  - `coreutils`
- Ensure that the `glibc-32bit-*.x86_64.rpm` and `libnsl-*.x86_64.rpm` is installed, where * denotes the latest version of the RPM.

  For example, to add `libnsl-<version>.x86_64.rpm` file you can enter the following command:

  ```
  zypper in libsnl-<version>.x86_64.rpm
  ```

- Ensure that the `libstdc++6-32bit` RPM is installed before installing iManager.
- (Conditional) If you are installing Identity Manager on SLES 15 or SLES 15 SPx (where x denotes all SLES 15 versions supported with Identity Manager), ensure that the `libncurses5*`, `libncurses6*` and `insserv-compat*` RPMs are installed. The * symbol denotes the latest version of the RPM.
- (Conditional) Before installing or upgrading Identity Manager to 4.8.x version, make sure that you download and execute the `idm_updated_os_linux.sh` script available at the location: TID KM000007635 (https://portal.microfocus.com/s/article/KM000007635?language=en_US). If you do not replace the script, the PostgreSQL database configuration may fail with the following error:

  ```
  symbol lookup error: /opt/netiq/idm/postgres/bin/../lib/
  libgssapi_krb5.so.2: undefined symbol: krb5_ser_context_init, version
  krb5_3_MIT
  ```

  NOTE: This issue occurs when the PostgreSQL database is installed on either Identity Applications or Identity Reporting on the same server.

  For more information, see TID KM000007635 (https://portal.microfocus.com/s/article/KM000007635?language=en_US).
- To install the required packages for Identity Manager on SLES, create an `install.sh` file and add the following contents to the file:

```
#!/bin/bash

zypper refresh

PKGS=" libstdc*.x86_64 libgcc*.x86_64 unzip bc bash coreutils
gettext.x86_64 glibc.x86_64 lsof net-tools openssl sed insserv ss* pam-
modules"

for PKG in $PKGS ;
do
    zypper -n install "$PKG"

done
```

Run the `install.sh` file.

NOTE: NetIQ recommends you to obtain the dependent packages from your operating system subscription service to ensure continued support from your operating system vendor. If you do not have a subscription service, you can find the recent packages from a website such as http://rpmfind.net/linux.

Installing Identity Manager on RHEL Servers

To install Identity Manager on a server running Red Hat Enterprise Linux or later operating systems, ensure that the server meets a specific set of prerequisites.

- "Prerequisites" on page 43
- "Ensuring that the Server has Dependent Libraries" on page 44
- "Creating a Repository on RHEL 8.x for the Installation Media" on page 45
- "Creating a Repository on RHEL 7.x for the Installation Media" on page 47
- "Running a Prerequisite Check" on page 48

Prerequisites

NetIQ recommends that you review the following prerequisites:

- If you have a loopback address alias to the hostname of the system in an `/etc/hosts` entry, it must be changed to the hostname or IP address. That is, if you have an entry similar to the one below in your `/etc/hosts` file, it needs to be changed to the correct entry given in second example below.

  The following example has problems when any utility tries to resolve to the ndsd server:

  ```
  <loopback IP address> test-system localhost.localdomain localhost
  ```

  The following is a correct example entry in `/etc/hosts`:

  ```
  <loopback IP address> localhost.localdomain localhost
  <IP address> test-system
  ```

If any third-party tool or utility resolves through localhost, it needs to be changed to resolve through a hostname or IP address and not through the localhost address.

- If you have configured Security-Enhanced Linux (SELinux) on RHEL 8.x, you must
  - set the value to **permissive** to install Identity Manager Engine. Otherwise, the Engine installation fails with the following error: `Identity Vault configuration failed with the exit code 10`

    For example, to set the value of SELinux to permissive, perform the following steps:

    1. Modify the `config` file located at the `/etc/selinux/` directory.
    2. In the **SELINUX** field, set the value to `permissive`.
    3. Save the changes and restart the system.
  - disable SELinux, if Identity Reporting is installed on a different server than Identity Manager Engine and Identity Applications. Otherwise, the Tomcat service will not come up and the Identity Reporting database configuration reports liquibase errors.
- Install the appropriate libraries on the server. For more information, see "Ensuring that the Server has Dependent Libraries" on page 44.
- Ensure that you set the Java path in either of two environment variables, $PATH or $JAVA_HOME on the server where you want to install Remote Loader. You must perform this action before running the `./RHEL-Prerequisite.sh script`. To set the Java path, run the following command:

  ```
  export PATH=<java installed location>/bin:$PATH
  ```

  For example, `export PATH=/opt/netiq/common/jre/bin/:$PATH`
- (Conditional) Before installing or upgrading Identity Manager to 4.8.x version, make sure that you download and execute the `idm_updated_os_linux.sh` script available at the location: TID KM000007635 (https://portal.microfocus.com/s/article/KM000007635?language=en_US). If you do not replace the script, the PostgreSQL database configuration may fail with the following error:

  ```
  symbol lookup error: /opt/netiq/idm/postgres/bin/../lib/
  libgssapi_krb5.so.2: undefined symbol: krb5_ser_context_init, version
  krb5_3_MIT
  ```

  NOTE: This issue occurs when the PostgreSQL database is installed on either Identity Applications or Identity Reporting on the same server.

  For more information, see TID KM000007635 (https://portal.microfocus.com/s/article/KM000007635?language=en_US).

Ensuring that the Server has Dependent Libraries

On a 64-bit platform, the required libraries for RHEL vary according to your chosen method of installation. Install the dependent libraries or RPMs in the following order.

NOTE: To add a `ksh` file, you can enter the following command:

```
yum -y install ksh
```

- `glibc-*.i686.rpm`

- libgcc-*.i686.rpm
- libXtst-*.i686.rpm
- libXrender-*.i686.rpm
- libXi-*.i686.rpm
- unzip
- bc
- lsof
- net-tools
- libncurses

NOTE: For Identity Manager, you can edit the ./RHEL-Prerequisite.sh script and remove all the occurrences of compat-libstdc++-33.x86_64.rpm and compat-libstdc++-33-*.i686.rpm. These packages are no longer necessary for Identity Manager installation.

Creating a Repository on RHEL 8.x for the Installation Media

If your RHEL 8.x server needs a repository for the installation media, you can manually create one.

NOTE: Your RHEL server must have the appropriate libraries installed. For more information, see "Ensuring that the Server has Dependent Libraries" on page 44.

To set up a repository for the installation:

1. Create a mount point on your local server.

   For example,

   ```
   mkdir -p /mnt/rhel8
   ```

2. Mount the RHEL 8 installation ISO:

   ```
   mount -o loop rhel-server-8.0-x86_64-dvd.iso mnt/rhel8
   ```

3. Copy the media.repo file from the mounted directory to /etc/yum.repos.d/ and set the required permissions.

   For example:

   ```
   cp /mnt/rhel8/media.repo /etc/yum.repos.d/rhel8.repo
   chmod 644 /etc/yum.repos.d/rhel8.repo
   ```

4. Modify the rhel8.repo file and add the following content:

```
[dvd-BaseOS]
name=DVD for RHEL8 - BaseOS
baseurl=file:///RHEL8/BaseOS
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release

[dvd-AppStream]
name=DVD for RHEL8 - AppStream
baseurl=file:///RHEL8/AppStream
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
```

5 If you want to install the 32-bit packages, change the value of `exactarch` parameter from 1 to 0 in the `/etc/yum.conf` file.

6 Run the following command:

```
yum clean all
```

7 (Conditional) If you want to retrieve the package list from the DVD repository, run the following command:

```
yum --noplugins list
```

8 Install the `yum-utils` package.

```
yum install createrepo yum-utils
```

9 To install the required packages for Identity Manager on RHEL8, create an `install.sh` file and add the following contents to the file:

NOTE: If you observe any warnings specific to duplicate RPMs, you must manually manage the warnings using the appropriate `yum` command.

```
#!/bin/bash
yum clean all
yum repolist
yum makecache

PKGS="ksh gettext.x86_64 libXrender.i686 libXau.i686 libxcb.i686
libX11.i686 libXext.i686 libXi.i686 libXtst.i686 glibc-*.i686.rpm
libstdc++.x86_64 libgcc-*.i686.rpm unzip bc lsof net-tools"

for PKG in $PKGS;
do
yum -y install "$PKG"
done
```

10 Install the following packages:

```
yum install libgcc*.i686 libnsl* libnsl*.i686 libncurses*
```

11 Run the `install.sh` file.

12 To confirm if the prerequisites are met, run the script as mentioned in "Running a Prerequisite Check" on page 48.

13 Install Identity Manager 4.8.

Creating a Repository on RHEL 7.x for the Installation Media

If your RHEL 7.x server needs a repository for the installation media, you can manually create one.

NOTE: Your RHEL server must have the appropriate libraries installed. For more information, see "Ensuring that the Server has Dependent Libraries" on page 44.

To set up a repository for the installation:

1. Create a mount point in your local server.

   Example: `/mnt/rhel (mkdir –p /mnt/rhel)`

2. If you use an installation media, you can mount using the following command:

   ```
   # mount -o loop /dev/sr0 /mnt/rhel
   ```

   OR

   Mount the RHEL 7 installation ISO to a directory like `/mnt/rhel`, using the following command:

   ```
   # mount -o loop RHEL7.x.iso /mnt/rhel
   ```

   Download RHEL 7.4 iso and mount the same.

   For example: `mount -o loop <path_to_downloaded rhel*.iso> /mnt/rhel`

3. Copy the `media.repo` file from the root of the mounted directory to `/etc/yum.repos.d/` and set the required permissions.

   For example:

   ```
   # cp /mnt/rhel/media.repo /etc/yum.repos.d/rhel7dvd.repo
   # chmod 644 /etc/yum.repos.d/rhel7dvd.repo
   ```

4. Edit the new repo file by changing the `gpgcheck=0` setting to `1` and add the following:

   ```
   enabled=1
   baseurl=file:///mnt/rhel/
   gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
   ```

   In the end, the new repo file would look like the following (though the mediaid would be different depending on the RHEL version):

   ```
   [InstallMedia]
   name=DVD for RHEL 7.x
   metadata_expire=-1
   gpgcheck=1
   cost=500
   enabled=1
   baseurl=file:///mnt/rhel
   gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-redhat-release
   ```

5. To install the 32-bit packages, change `"exactarch=1"` to `"exactarch=0"` in the `/etc/yum.conf` file.

6. To install the required packages for Identity Manager on RHEL7.x, create an `install.sh` file and add the following contents to the file:

NOTE: If you observe any warnings specific to duplicate RPMs, you must manually manage the warnings using the appropriate `yum` command.

```
#!/bin/bash
yum clean all
yum repolist
yum makecache

PKGS="ksh gettext.x86_64 libXrender.i686 libXau.i686 libxcb.i686
libX11.i686 libXext.i686 libXi.i686 libXtst.i686 glibc-*.i686.rpm
libstdc++.x86_64 libgcc-*.i686.rpm unzip bc lsof net-tools"

for PKG in $PKGS;
do
yum -y install "$PKG"
done
```

7 Run the `install.sh` file created in Step 6 depending on the RHEL version.

8 To confirm if the prerequisites are met, run the script as mentioned in "Running a Prerequisite Check" on page 48.

9 Install Identity Manager 4.8.

Running a Prerequisite Check

You can generate a report of the missing prerequisites for each Identity Manager component. Run the `./RHEL-Prerequisite.sh` script located in the mount directory of the installation kit.

3 Considerations for Installing Identity Manager Components

This section provides the prerequisites, considerations, and system setup needed to install the Identity Manager components.

- "Installation Order" on page 49
- "Understanding the Installation and Configuration Process" on page 49
- "Considerations for Installing Identity Manager Engine Components and Remote Loader" on page 51
- "Considerations for Installing Identity Applications Components" on page 52
- "Considerations for Installing Identity Reporting Components" on page 56
- "Considerations for Installing Designer" on page 58
- "Considerations for Installing Analyzer" on page 58
- "Considerations for Installing SLM for IGA" on page 59
- "Reviewing the Ports Used by the Identity Manager Components" on page 60

Installation Order

The components must be installed in the following order because the installation programs for some components require information about previously installed components:

- Sentinel Log Management for Identity Governance and Administration
- Identity Manager Engine components
- Identity Applications components (only for Advanced Edition)
- Identity Reporting components
- Designer for Identity Manager
- Analyzer for Identity Manager

You must review the installation prerequisites and considerations for each component before installing the component.

Understanding the Installation and Configuration Process

Interactive Installation: In an interactive installation, you must specify the inputs in an interactive mode. Identity Manager provides a scripted installation program for installing and configuring individual components or a group of components in two separate phases. The installation phase installs the components. The installation script, `install.sh`, is located in the root of the `.iso` image file of the Identity Manager installation package. For information about configuration, see Configuration.

Table 3-1 Installation Options

Installation Option	Components Installed
Identity Manager Engine	Installs the Identity Vault, Identity Manager engine, and Identity Manager drivers. The installation process also installs Azul Zulu JDK.
Identity Manager Remote Loader Server	Installs the Remote Loader service and the driver instances in the Remote Loader.
Identity Manager Fanout Agent	Installs the Fanout agent for the JDBC Fanout driver. For more information, see *NetIQ Identity Manager Driver for JDBC Fanout Implementation Guide*.
iManager Web Administration	Installs the iManager Web Administration console and iManager plug-ins.
Identity Applications	Installs several components that provide the underlying framework for the identity applications. ◆ User Application ◆ OSP ◆ SSPR ◆ Tomcat ◆ PostgreSQL database To support the Tomcat application server, the installation program installs supported versions of JRE and Apache ActiveMQ. The installation process also deploys the User Application driver and the Role and Resource Service driver to the Identity Vault.
Identity Reporting	Installs several components that provide the underlying framework for Identity Reporting. ◆ Identity Reporting ◆ Managed System Gateway driver (MSGW) ◆ Data Collection Service driver (DCS) ◆ OSP (when installed on a different server than Identity Applications) ◆ Tomcat (when installed on a different server than Identity Applications) ◆ PostgreSQL database (when installed on a different server than Identity Applications) To support the Tomcat application server, the installation program installs a supported version of JRE.

NOTE: Identity Manager provides separate installation programs for Designer, Analyzer, and Sentinel Log Management.

Silent Installation: The installer provides an option to create a silent properties file in an interactive mode. You can record the installation options in the properties file and then use the file to run the silent installation on different servers in your environment. The silent installation program reads the values from the file to perform the installation. For details on the component-wise configuration, see "Understanding the Configuration Parameters" on page 77.

Configuration: Identity Manager provides two modes of configuration:

- Typical configuration
- Custom configuration

A typical configuration assumes default settings for most of the configuration options. In a custom configuration, you can specify custom values according to your requirement. You can configure most of the settings using this option.

Considerations for Installing Identity Manager Engine Components and Remote Loader

- The Identity Manager Engine and iManager installation process requires the following minimum space for installation:

Path	Component	Minimum Safe Space Required
/opt	Identity Manager Engine	3 GB
/var	Identity Manager Engine	5 GB for dib of 100,000 object
/etc	Identity Manager Engine	5 MB
/opt	iManager	700 MB
/var	iManager	3 GB
/etc	iManager	10 MB

- Ensure that Identity Manager Engine is installed before installing the Remote Loader.
- You can install the Remote Loader on the same computer where you installed the Identity Manager engine. Ensure that the operating system supports both components.
- Install the Remote Loader on a server that can communicate with the managed systems. The driver for each managed system must be available with the relevant APIs.
- (Conditional) If you install the Identity Manager engine as a non-root user, the installation process does not install Linux Account Driver, or Remote Loader. You must install these components separately.
- NetIQ recommends you to install `changelog` module on a different server than Identity Manager Engine or Remote Loader.
- If you are planning to use a non-default deployment context, that is, anything other than `o=system`, NetIQ recommends that you create a custom LDIF file prior to the Identity Manager Engine installation. The location of the custom LDIF file will be prompted during the configuration process.

NOTE: If you have not created the OU structure prior to the installation, you can define the required OU structure with the hierarchy of all containers under which the driver set will be created. This structure and the hierarchy will be used when the new driver set is being configured through the LDIF file.

- If you install or upgrade to Identity Manager 4.8 on Open Enterprise Server 2018, you must manually install or update Identity Manager plug-ins from iManager. For more information, see Downloading and Installing Plug-in Modules in the NetIQ iManager Administration Guide.
- To upgrade Identity Manager 4.8.5 on Open Enterprise Server 2022, you must first install Identity Manager Engine 4.8 and upgrade to the 4.8.5 version. Do not configure the engine parameters until version 4.8.5 has been installed.
- Before you begin with the installation process, NetIQ recommends you to run the following command for all the languages:

  ```
  export LC_ALL=<language>
  ```

  For example,

  ```
  export LC_ALL=zh_TW
  ```

Considerations for Installing Identity Applications Components

NetIQ recommends that you review the prerequisites and computer requirements for the identity applications before you begin the installation process. For more information about configuring the identity applications environment after installing the application components, see *NetIQ Identity Manager - Administrator's Guide to the Identity Applications*.

- "Installation Considerations" on page 52
- "Database Considerations" on page 53
- "Configuring the Database for Identity Applications" on page 54

Installation Considerations

- The Identity Applications installation process requires the following minimum space for installing the components:
  - /opt - 5 GB
  - /var - 100 MB
- Identity Applications require a supported version of the following Identity Manager components:
  - Identity Manager engine
  - Remote Loader
- (Optional) NetIQ enables Secure Sockets Layer (SSL) protocol during the installation. To change the communication settings among the identity applications components in your environment, see Configuring Security in the Identity Applications in the *NetIQ Identity Manager - Administrator's Guide to the Identity Applications*.

- You cannot use the Role and Resource Service driver with the Remote Loader because the driver uses jClient.
- If you plan to install User Application in a non-default location, ensure that the new directory is writable by non-root users.
- Each User Application instance can service only one user container. For example, you can add users to, search, and query only the container associated with the instance. Also, a user container association with an application is meant to be permanent.
- In a distributed environment, you must have a certificate with CN as Identity Applications in the keystore (`idm.jks`) of the Identity Applications server. As part of enhanced Java security, now Identity Applications requires trusted certificate to communicate with OSP.

Database Considerations

The database stores the identity applications data and configuration information.

Before installing the database instance, review the following prerequisites:

- To configure a database for use with Tomcat, you must ensure that it contains the required JDBC jar file. The identity applications use standard JDBC calls to access and update the database. The identity applications use a JDBC data source file bound to the JNDI tree to open a connection to the database.
- You must have an existing data source file that points to the database. The installation program for the User Application creates a data source entry for Tomcat in `server.xml` and `context.xml` which points to the database.
- If you are using a supported version of Oracle or Microsoft SQL Server database, you must configure two database instances for Identity Applications to work correctly; Identity Applications (`idmuserappdb`) database and the Workflow (`igaworkflowdb`) database. Ensure that you configure the database instances on the same server.
- Ensure that you have the following information:
  - Host and port of the database server.
  - Name of the database to create. The default database for the identity applications is `idmuserappdb`.
  - Database username and password. The database username must represent an Administrator account or must have enough permissions to create tables in the Database Server. The default administrator for the User Application is `idmadmin`.
  - The driver `.jar` file provided by the database vendor for the database that you are using. NetIQ does not support driver JAR files provided by third-party vendors.
- The database instance can be on the local computer or a connected server.
- The database character set must use Unicode encoding. For example, UTF-8 is an example of a character set that uses Unicode encoding, but Latin1 does not use Unicode encoding. For more information about specifying the character set, see "Configuring the Character Set" on page 55 or "Configuring an Oracle Database" on page 54.
- If you are connecting to a remote database, ensure that you create the database before installing Identity Applications. For information on connecting to the remote PostgreSQL database, see "Connecting to a Remote PostgreSQL Database" on page 146.

- The case-sensitive collation for your database might cause a duplicate key error during migration. Check the collation and correct it, then re-install the identity applications.
- (Conditional) To use the same database instance both for auditing purposes and for the identity applications, NetIQ recommends installing the database on a separate dedicated server from the server that hosts Tomcat running the identity applications.
- (Conditional) If you are migrating to a new version of the identity applications, you must use the same database that you used for the previous installation.
- The only supported collation for MS SQL is SQL_Latin1_General_CP1_CI_AS.

Configuring the Database for Identity Applications

The database for the identity applications supports tasks such as storing configuration data and data for workflow activities. Before you can install the applications, the database must be installed and configured.

By default, the installation process installs PostgreSQL database for the identity applications and creates an administrative user called `idmadmin` to own the database. However, the installation does not create the schema in the database for the identity applications. Schema information is added when you install the identity applications.

Configuring an Oracle Database

This section provides configuration options for using an Oracle database for the User Application.

- "Checking Compatibility Level of Databases" on page 54
- "Configuring the Character Set" on page 55
- "Configuring the Admin User Account" on page 55

Checking Compatibility Level of Databases

Databases from different releases of Oracle are compatible if they support the same features and those features perform the same way. If they are not compatible, certain features or operations might not work as expected. For example, creation of schema fails that does not allow you to deploy the identity applications.

To check the compatibility level of your database, perform the following steps:

1. Connect to the Database Engine.
2. After connecting to the appropriate instance of the SQL Server Database Engine, in **Object Explorer**, click the server name.
3. Expand **Databases**, and, depending on the database, either select a user database or expand **System Databases** and select a system database.
4. Right-click the database, and then click **Properties**.

   The **Database Properties** dialog box opens.
5. In the **Select a page** pane, click **Options**.

   The current compatibility level is displayed in the **Compatibility level** list box.
6. To check the **Compatibility Level**, enter the following in the query window and click **Execute**.

```
SQL> SELECT name, value FROM v$parameter
WHERE name = 'compatible';
```

The expected output is 12.2.x.x, 18.x.x, or 19.x.x.

NOTE: Oracle 19c is supported from Identity Manager 4.8.1 onwards.

Configuring the Character Set

Your User Application database must use a Unicode-encoded character set. When creating the database, use AL32UTF8 to specify this character set.

To confirm that your supported Oracle database is set for UTF-8, issue the following command:

```
select * from nls_database_parameters;
```

If the database is not configured for UTF-8, the system responds with the following information:

```
NLS_CHARACTERSET
WE8MSWIN1252
```

Otherwise, the system responds with the following information that confirms the database is configured for UTF-8:

```
NLS_CHARACTERSET
AL32UTF8
```

For more information about configuring a character set, see "Choosing an Oracle Database Character Set".

Configuring the Admin User Account

The User Application requires that the Oracle database user account has specific privileges. In the SQL Plus utility, enter the following commands:

```
CREATE USER idmuser IDENTIFIED BY password;
GRANT CREATE SESSION TO idmuser;
GRANT CREATE CLUSTER TO idmuser;
GRANT CREATE PROCEDURE TO idmuser;
GRANT CREATE SEQUENCE TO idmuser;
GRANT CREATE TABLE TO idmuser;
GRANT CREATE TRIGGER TO idmuser;
ALTER USER idmuser quota 100M on USERS;
```

where *`idmuser`* represents the user account.

NOTE: It is recommended to use JDBC JAR version `ojdbc8.jar`.

Configuring a SQL Server Database

This section provides configuration options for using an SQL Server database for the User Application.

- "Configuring the Character Set" on page 56
- "Configuring the Admin User Account" on page 56

Configuring the Character Set

SQL Server does not allow you to specify the character set for databases. The User Application stores SQL Server character data in a NCHAR column type, which supports UTF-8.

Configuring the Admin User Account

After installing Microsoft SQL Server, create a database and database user using an application such as SQL Server Management Studio. The database user account must have the following privileges:

- `CREATE TABLE`
- `DELETE`
- `INSERT`
- `SELECT`
- `UPDATE`
- `REFERENCES`

NOTE: It is recommended to use JDBC JAR version `sqljdbc42.jar`.

Considerations for Installing Identity Reporting Components

This section provides guidance for preparing to install the components for Identity Reporting. You can use Sentinel to audit events.

NetIQ recommends that you review the following information before starting the installation process.

- "Prerequisites for Identity Reporting" on page 56
- "Identifying Audit Events for Identity Reporting" on page 57

Prerequisites for Identity Reporting

- The installation process requires the following minimum space requirements:
  - /opt - 2 GB
  - /var - 2 GB
  - /etc - 2 GB

- The installation process requires a supported and configured version of the following Identity Manager components:
  - Identity applications, including the User Application driver (applicable only for Advanced Edition)
  - Sentinel Log Management installed on a separate Linux computer.
- The installation process modifies `JAVA_OPTs` or `CATALINA_OPTS` entries for JRE mapping in the `setenv.sh` file for Tomcat.
- Do not install Identity Reporting on a server in a clustered environment.
- If you are connecting to a remote database, ensure that you create the database before installing Identity Reporting. For information on connecting to the remote PostgreSQL database, see "Connecting to a Remote PostgreSQL Database" on page 146.
- To run reports against an Oracle database, you must ensure that you have copied the `ojdbc8.jar`. For more information, see "Running Reports on an Oracle Database" on page 142.
- Assign the Report Administrator role to any users that you want to access reporting functionality
- Ensure that all servers in your Identity Manager environment are set to the same time. If you do not synchronize the time on your servers, some reports might be empty when executed. For example, this issue can affect data related to new users when the servers hosting the Identity Manager engine and the warehouse have different time stamps. If you create and then modify a user, the reports are populated with data.
- To configure Reporting, you must specify the hostname in lowercase. Identity Reporting 6.6.0 and its later versions no longer allows IP address to configure Reporting.
- If you are using a supported version of Remote PostgreSQL, Oracle, or Microsoft SQL Server databases, you must configure the database instance for Identity Reporting (idmrptdb) to work correctly. Ensure that you configure the database instances on the same server.

Identifying Audit Events for Identity Reporting

This section provides information on how to identify different audit events required for Identity Manager reports and custom reports. You can unzip all report sources and run the following script to identify the audit events:

```
find . -name *.jrxml -print0 |xargs -0 grep -H "'000[B3]" | perl -ne
'($file) = /^\.\/(.*?)\//;@a = /000[3B]..../g; foreach $a (@a) { print
"$file;$a\n"}' |sort -u
```

The following section provides information on how to identify and select various audit events for identity Manager reports and custom reports:

Event Name	Audit Flag
Authentication and Password Change	**Selecting Audit Flag using SSPR:** Launch **SSPR Configuration Editor** > **Audit Configuration** > Select from the following audit flags: ◆ Authenticate ◆ Change Password ◆ Unlock Password ◆ Recover Password ◆ Intruder Attempt ◆ Intruder Lock ◆ Intruder Lock User **Selecting Audit Flag using iManager:** Go to **iManager Roles and Tasks** > **eDirectory Auditing** > > **Audit Configuration** > **Novell Audit** > Select from the following audit flags: ◆ Change Password ◆ Verify Password ◆ Login ◆ Logout
All other reporting events	Go to **NetIQ Identity Manager UserApp** > **Administration** > **Logging** > **Enable audit service**

Considerations for Installing Designer

- On a computer running SLES or RHEL, install the GNU gettext utilities (`gettext`) before installing Designer. These utilities provide a framework for internationalized and multilingual messages.
- (Conditional) On RHEL 7.4, install `gtk2-2.24.31-1.el7.x86_64.rpm` before installing Designer. For example, you can download the package from the operating system vendor website.

NOTE: NetIQ recommends you to obtain the dependent packages from your operating system subscription service to ensure continued support from your operating system vendor website. If you do not have a subscription service, you can find the recent packages from a website such as http://rpmfind.net/linux.

Considerations for Installing Analyzer

- Before installing Analyzer on a computer running SLES 12 SP3 platform, ensure that the following libraries are installed:
  - `libswt3-gtk2`
  - `libxcomposite`

    - `libgdk_pixbuf`
    - `libgtk+-x11`
    - `gettext` (GNU gettext utilities)
- Before installing Analyzer on a computer running RHEL 7.5 or later platforms, ensure that the following libraries are installed:
    - `gtk2.i686.rpm`. For example, you can download the package from the operating system vendor website.
    - `gettext` (GNU gettext utilities)

NOTE: NetIQ recommends you to obtain the dependent packages from your operating system subscription service to ensure continued support from your operating system vendor. If you do not have a subscription service, you can find the recent packages from a website such as http://rpmfind.net/linux/.

- Ensure that the computer running Analyzer has a video resolution of 1024x768 (1280x1025 recommended).

Considerations for Installing SLM for IGA

The operating system for the SLM for IGA server must include at least the base server components of the SLES server or the RHEL server. Sentinel requires the 64-bit versions of the following RPMs:

- `bash`
- `bc`
- `coreutils`
- `gettext`
- `glibc`
- `grep`
- `libgcc`
- `libstdc`
- `lsof`
- `net-tools`
- `openssl`
- `python-libs`
- `sed`
- `zlib`

For more information, see the NetIQ Sentinel Technical Information website.

Reviewing the Ports Used by the Identity Manager Components

Identity Manager components use different ports for proper communication among the Identity Manager components.

NOTE: If a default port is already in use, ensure that you specify a different port for the Identity Manager component.

Port Number	Component	Port Use
389	Identity Vault	Used for LDAP communication in clear text with Identity Manager components
465	Identity Reporting	Used for communication with the SMTP mail server
524	Identity Vault	Used for NetWare Core Protocol (NCP) communication
636	Identity Vault	Used for LDAP with TLS/SSL communication with Identity Manager components
5432	Identity Applications	Used for communication with the identity applications database
7707	Identity Reporting	Used by the Managed System Gateway driver to communicate with the Identity Vault
8000	Remote Loader	Used by the driver instance for TCP/IP communication **NOTE:** Each instance of the Remote Loader should be assigned a unique port.
8005	Identity Applications	Used by Tomcat to listen for shutdown commands
8009	Identity Applications	Used by Tomcat for communication with a web connector using the AJP protocol instead of HTTP
8028	Identity Vault	Used for HTTP clear text communication with NCP communication
8030	Identity Vault	Used for HTTPS communication with NCP communication
8080	Identity Applications iManager	Used by Tomcat for HTTP clear text communication
8090	Remote Loader	Used by the Remote Loader to listen for TCP/IP connections from the remote interface shim **NOTE:** Each instance of the Remote Loader should be assigned a unique port.
8109	Identity Applications	Applies only when using the integrated installation process Used by Tomcat for communication with a web connector using the AJP protocol instead of HTTP

Port Number	Component	Port Use
8180	Identity Applications	Used for HTTP communications by the Tomcat application server on which the identity applications run
8443	Identity Applications iManager	Used by Tomcat for HTTPS (SSL) communication or redirecting requests for SSL communication
8543	Identity Applications	*Not listening, by default* Used by Tomcat to redirect requests that require SSL transport when you do not use TLS/SSL protocol
8600	Form Renderer	Used by the NGINX service
9009	iManager	Used by Tomcat for MOD_JK
5432	Identity Reporting	Used for the PostgreSQL database Sentinel
45654	User Application	Used by the server on which the database for the identity applications are installed to listen for communications, when running Tomcat with a cluster group

III Installing and Configuring Identity Manager Components

This section guides you through the process of installing and configuring Identity Manager components. For installation instructions, see Chapter 4, “Installing Identity Manager,” on page 65. For instructions on configuring the Identity Manager components, see Chapter 5, “Configuring the Identity Manager Components,” on page 77.

After Identity Manager components are installed and basic configuration has been completed, you must perform some additional configuration steps for the components to be fully functional. For more information, see Chapter 6, “Final Steps for Completing the Installation,” on page 93.

4 Installing Identity Manager

This section provides information about the various ways to install the Identity Manager components. You can install the Identity Manager components through the following ways:

- Interactive Installation
- Silent Installation

Performing an Interactive Installation

1. Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.
2. Mount the downloaded `.iso`.
3. From the root directory of the `.iso` file, run the following command:

   ```
   ./install.sh
   ```

4. Read through the license agreement.
5. Enter y to accept the license agreement.
6. Decide the Identity Manager server edition you want to install. Enter y for Advanced Edition and n for Standard Edition.
7. From the list of components available for installation, select the required components:
   - To install Engine, select **Identity Manager Engine**.
   - To install Remote Loader, select **Identity Manager Remote Loader**.
   - To install Fanout Agent, select **Identity Manager Fanout Agent**.
   - To install iManager, select **iManager Web Administration**.
   - To install Identity Applications, select **Identity Applications**.
   - To install Identity Reporting, select **Identity Reporting**.
8. (Conditional) Configure the installed components. For more information, see Chapter 5, “Configuring the Identity Manager Components,” on page 77.

Performing a Silent Installation

1. Download the `Identity_Manager_4.8_Linux.iso` from the download site.
2. Mount the downloaded `.iso`.
3. From the root directory of the `.iso`, run the `create_silent_props.sh` script.
4. Enter y to confirm the creation of the file.
5. Specify the path for the silent properties file.
6. Decide the Identity Manager server edition you want to install. Enter y for Advanced Edition and n for Standard Edition.
7. Decide if you want to configure the components in a typical or custom mode.

8 From the list of components available for installation, select the required components:

- To install Engine, select **Identity Manager Engine**.
- To install Remote Loader, select **Identity Manager Remote Loader**.
- To install Fanout Agent, select **Identity Manager Fanout Agent**.
- To install iManager, select **iManager Web Administration**.
- To install Identity Applications, select **Identity Applications**.
- To install Identity Reporting, select **Identity Reporting**.

For information about the configuration parameters, see "Understanding the Configuration Parameters" on page 77.

9 Run the following command to perform a silent installation:

```
./install.sh -s -f <location of the silent properties file>
```

For example,

`./install.sh -s -f /home/silent.properties`, where `/home/silent.properties` is the location where you stored the silent properties file.

Installing Identity Manager Engine as a Non-root User

You can install Identity Manager engine as a non-`root` user to enhance the security of your Linux server. You cannot install Identity Manager engine as a non-`root` user if you installed the Identity Vault as `root`. You need to perform the following steps if you want to install the engine as a non-`root` user:

1. Ensure that NICI is installed. For more information, see "Installing NICI" on page 66.
2. Perform a non-root installation of Identity Vault. For more information, see "Performing a Non-root Installation of Identity Vault" on page 67.
3. Perform a non-root installation of Identity Manager Engine. For more information, see "Performing a Non-root Installation of Engine" on page 68.

Installing NICI

You must install NICI before you proceed with the Identity Vault installation. Since the required NICI packages are used system-wide, you are recommended to use the root user to install the necessary packages. However, if necessary you can delegate access to a different account using `sudo` and use that account to install the NICI packages.

1 From the `iso` that you have mounted, navigate to the `/IDVault/setup/` directory.

2 Run the following command:

```
rpm -ivh nici64-3.1.0-1.00.x86_64.rpm
```

3 Verify that NICI is set to server mode. Enter the following command:

```
/var/opt/novell/nici/set_server_mode64
```

This is a mandatory step to ensure that the Identity Vault configuration process does not fail.

Performing a Non-root Installation of Identity Vault

This section describes how to use the tarball to install the Identity Vault. When you extract the file, the system creates the `etc`, `opt`, and `var` directories.

1. Log in as a `sudo` user with the appropriate rights to the computer where you want to install the Identity Vault.

   NOTE: You can also log in as a `root` user, when you want to specify a custom installation path.

2. From the `iso` that you have mounted, navigate to the `/IDVault/` directory.
3. Create a new directory and copy the `eDir_NonRoot.tar.gz` file to that directory. For example, `/home/user/install/eDirectory`.
4. Use the following command to extract the file:

   ```
   tar -zxvf eDir_NonRoot.tar.gz
   ```

5. To manually export the paths for environment variables, enter the following command:

   ```
   export LD_LIBRARY_PATH=custom_location/eDirectory/opt/novell/
   eDirectory/
   lib64:custom_location/eDirectory/opt/novell/eDirectory/lib64/
   ndsmodules:
   custom_location/eDirectory/opt/novell/lib64:$LD_LIBRARY_PATH

   export PATH=custom_location/eDirectory/opt/novell/eDirectory/
   bin:custom_location/eDirectory/opt/novell/eDirectory/sbin:/opt/novell/
   eDirectory/bin:$PATH

   export MANPATH=custom_location/eDirectory/opt/novell/
   man:custom_location/
   eDirectory/opt/novell/eDirectory/man:$MANPATH

   export TEXTDOMAINDIR=custom_location/eDirectory/opt/novell/eDirectory/
   share/locale:$TEXTDOMAINDIR
   ```

6. To use the ndspath script to export the paths for environment variables, you must prefix the ndspath script to the utility. Complete the following steps:

   6a From the `custom_location/eDirectory/opt directory`, run the utility with the following command:

   ```
   custom_location/eDirectory/opt/novell/eDirectory/bin/ndspath
   utility_name_with_parameters
   ```

   6b Export the paths in the current shell with the following command:

   ```
   . custom_location/eDirectory/opt/novell/eDirectory/bin/ndspath
   ```

   6c Run the utilities as normal.

   6d Add the instructions for exporting the path to the end of `/etc/profile`, `~/bashrc`, or similar scripts.

   This step allows you to start the utilities directly whenever you log in or open a new shell.

7. Configure Identity Vault by using one of the following methods:
   - Use the ndsconfig utility

```
ndsconfig new [-t <treename>] [-n <server_context>] [-a <admin_FDN>]
[-w
<admin password>] [-i] [-S <server_name>] [-d <path_for_dib>] [-m
<module>]
[e] [-L <ldap_port>] [-l <SSL_port>] [-o <http_port>] -O
<https_port>] [-p
<IP address:[port]>] [-c] [-b <port_to_bind>] [-B
<interface1@port1>,
<interface2@port2>,..] [-D <custom_location>] [--config-file
<configuration_file>] [--configure-eba-now <yes/no>]
```

where, -t denotes the tree name to which the server has to be added.

-n denotes the context of the server in which the server object is added.

-a fully distinguished name of the User object with Supervisor rights to the context in which the server object and Directory services are to be created.

-s denotes the server name

-d denotes the directory path where the database files are stored.

-m denotes the module name.

You must specify the same values that you specified during the configuration process.

For example:

```
ndsconfig new -t novell-tree -n novell -a admin.novell -S linux1 -d
/home/
mary/inst1/data -b 1025 -L 1026 -l 1027 -o 1028 -O 1029 -D /home/
inst1/var --config-file /home/inst1/nds.conf --configure-eba-now
yes
```

The port numbers you enter need to be in the range 1024 to 65535. Port numbers lesser than 1024 are normally reserved for the super-user and standard applications. Therefore, you cannot assume the default port 524 for any eDirectory applications.

This might cause the following applications to break:

- The applications that don't have an option to specify the target server port.
- The older applications that use NCP, and run as root for 524.

- Use the ndsmanage utility to configure a new instance. For more information, see the Creating an Instance through ndsmanage in the NetIQ eDirectory Installation Guide.

Performing a Non-root Installation of Engine

When you use this method, you cannot install the following components:

- **Remote Loader:** To install the Remote Loader as a non-root user, use the Java Remote Loader. For more information, see "Installing Java Remote Loader" on page 75.
- **Linux Account Driver:** Requires `root` privileges to function.

NOTE: When you install Identity Manager engine as a non-root user, the installation files are located under the non-root users directory. For example, `/home/user`; where user is non-root. The installation files are not required to run Identity Manager. You can delete the files after installation.

To install the Identity Manager engine as a non-root user:

1. Log in as the non-root user that you used to install the Identity Vault.

   The user account must have write access to the directories and files of the non-root Identity Vault installation.
2. Navigate to the location where you have mounted the `Identity_Manager_4.8_Linux.iso`.
3. From the mount location, navigate to the `/IDM` directory.
4. Execute the following command:

   ```
   ./idm-nonroot-install.sh
   ```
5. Use the following information to complete the installation:

   Base Directory for the non-root eDirectory Installation

   Specify the directory where the non-`root` eDirectory installation is. For example, `/home/user/install/eDirectory`.

   Extend eDirectory Schema

   If this is the first Identity Manager server installed in this instance of eDirectory, enter Y to extend the schema. If the schema is not extended, Identity Manager cannot function.

   You are prompted to extend the schema for each instance of eDirectory owned by the non-`root` user that is hosted by the non-`root` eDirectory installation.

   If you select to extend the schema, specify the full distinguished name (DN) of the eDirectory user who has rights to extend the schema. The user must have the Supervisor right to the entire tree to extend the schema. For more information about extending the schema as a non-`root` user, see the `schema.log` file that is placed in the `data` directory for each instance of eDirectory.

   Run the `/opt/novell/eDirectory/bin/idm-install-schema` program to extend the schema on additional eDirectory instances after the installation is complete.
6. To complete the installation process, continue to "Completing a Non-root Installation" on page 147.

Installing SSPR

The installer provides you an option to install SSPR separately. This is useful when you want to install Identity Applications and SSPR on separate computers.

NOTE: If you are installing Standard Edition, you must use the following procedure to install SSPR. By default, SSPR is not installed when you use standard edition.

Performing an Interactive Installation of SSPR

1. Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.
2. Mount the downloaded `.iso`.
3. From the root directory of the `.iso` file, navigate to the `sspr` directory.
4. To install SSPR, run the following command:

   ```
   ./install.sh
   ```

5 Read through the license agreement.
6 Enter y to accept the license agreement.
7 (Conditional) Configure SSPR. For more information, see "Configuring SSPR" on page 90.

Performing a Silent Installation of SSPR

1 Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.
2 Mount the downloaded `.iso`.
3 From the root directory of the `.iso` file, navigate to the `sspr` directory.
4 To perform a silent installation, run the following command:

```
./install.sh -s -f sspr_silentinstall.properties
```

5 (Conditional) Configure SSPR. For more information, see "Configuring SSPR" on page 90.

Installing Remote Loader

Identity Manager provides you an option to install Remote Loader on a standalone server. Use this option when you want to install Identity Manager Engine and Remote Loader on separate computers.

Interactive Installation

1 Download the `Identity_Manager_4.8_RL_Linux.iso` from the NetIQ Downloads website.
2 Mount the downloaded `.iso`.
3 From the mounted location, run the following command:

```
./install.sh
```

4 Read through the license agreement.
5 Enter y to accept the license agreement.

Silent Installation

1 Download the `Identity_Manager_4.8_RL_Linux.iso` from the NetIQ Downloads website.
2 Mount the downloaded `.iso`.
3 From the root directory of the `.iso`, run the following command:

```
./create_silent_props.sh
```

4 Enter y to confirm the creation of the file.
5 Specify the location for the file.
6 Specify *n* for the **Do you want to configure the silent properties file for Docker containers** parameter.
7 Run the following command to perform a silent installation:

```
./install.sh -s -f <location of the silent properties file>
```

For example,

`./install.sh -s -f /home/silent.properties`, where `/home/silent.properties` is the location where you stored the silent properties file.

Installing Designer

You can install Designer either in GUI or console mode.

NOTE: To install Designer on RHEL platform, the RHEL repositories should be created. For more information, see "Installing Identity Manager on RHEL Servers" on page 43.

---

1. Download the `Identity_Manager_4.8_Designer_Linux.tar.gz` from the NetIQ Downloads website.
2. Navigate to a directory where you want to extract the file.
3. Run the following command:

   ```
   tar -zxvf Identity_Manager_4.8_Designer_Linux.tar.gz
   ```

4. Run one of the following commands to install Designer.

   Console: `./install`

   GUI: `./install -i console`

5. Follow the prompts and proceed with the installation.

Installing Analyzer

This section provides information about the various ways to install Analyzer and configure your environment for Analyzer.

- Using the Wizard to Install Analyzer
- Installing Analyzer Silently
- Adding XULRunner to Analyzer.ini

Using the Wizard to Install Analyzer

The following procedure describes how to install Analyzer on a Linux or Windows platform using an installation wizard, either in the GUI format or from the console. To perform a silent, unattended installation, see "Installing Analyzer Silently" on page 72.

1. Log in as `root` or an administrator to the computer where you want to install Analyzer.
2. (Conditional) If you have the `.iso` image file for the Identity Manager installation package, navigate to the directory containing the Analyzer installation files, located by default in the `/Analyzer/packages` directory.
3. (Conditional) If you downloaded the Analyzer installation files, complete the following steps:

   3a Navigate to the `.tgz` or `win.zip` file for the downloaded image.

   3b Extract the contents of the file to a folder on the local computer.

4 Execute the installation program:

```
./install
```

5 Follow the instructions in the wizard until you finish installing Analyzer.

6 When the installation process completes, review the post-installation summary to verify the installation status and the location of the log file for Analyzer.

7 Click **Done**.

8 (Conditional) Complete the steps in "Adding XULRunner to Analyzer.ini" on page 72.

9 Activate Analyzer. For more information, see Activating Analyzer in *NetIQ Identity Manager Overview and Planning Guide*.

Installing Analyzer Silently

A silent (non-interactive) installation does not display a user interface or ask the user any questions. Instead, InstallAnywhere uses information from a default `analyzerInstaller.properties` file. You can run the silent installation with the default file or edit the file to customize the installation process.

By default, the installation program installs Analyzer in the `Program Files (x86)\NetIQ\Analyzer` directory.

1 Log in as `root` or an administrator to the computer where you want to install Analyzer.

2 If you downloaded the Analyzer installation files from the NetIQ Downloads website, complete the following steps:

   2a Navigate to the `.tgz` or `win.zip` file for the downloaded image.

   2b Extract the contents of the file to a folder on the local computer.

3 (Optional) To specify a non-default installation path, complete the following steps:

   3a Open the `analyzerInstaller.properties` file, located by default in the `<location where you have extracted the file>/analyzer_install/` directory.

   3b Add the following text to the properties file:

   `USER_INSTALL_DIR=`*`installation_path`*

4 To run the silent installation, run the following command from the directory containing the properties file:

```
./install -i silent -f analyzerInstaller.properties
```

5 (Conditional) On Linux computers, complete the steps in "Adding XULRunner to Analyzer.ini" on page 72.

6 Activate Analyzer. For more information, see Activating Analyzer in *NetIQ Identity Manager Overview and Planning Guide*.

Adding XULRunner to Analyzer.ini

Before running Analyzer on a Linux platform, you must change the XULRunner mapping.

1 Install XULRunner.

2 Navigate to the `Analyzer` installation directory, by default in the following locations:

root/analyzer

3 Open the Analyzer.ini file in the gedit editor.

4 Add the following line to the end of the list of the parameters:

```
-Djava.util.Arrays.useLegacyMergeSort=true
-Dorg.eclipse.swt.internal.gtk.disablePrinting
-Dorg.eclipse.swt.browser.XULRunnerPath=/opt/xulrunner
```

5 Save the Analyzer.ini file.

6 Launch Analyzer.

Installing Sentinel Log Management for Identity Governance and Administration

You can install Sentinel Log Management using the following methods:

- Installing Sentinel Log Management as a root user
- Installing Sentinel Log Management as a non-root user

Installing Sentinel Log Management as a Root User

1 Download the SentinelLogManagementForIGA8.2.2.0.tar.gz from the NetIQ downloads Website.

2 Navigate to the directory where you want to extract the file.

3 Run the following command to extract the file

```
tar -zxvf SentinelLogManagementForIGA8.2.2.0.tar.gz
```

4 Navigate to the SentinelLogManagementforIGA directory.

5 Run the following command:

```
./install.sh
```

6 Enter y to accept the license agreement and continue with the installation.

The installation might take a few seconds to load the installation packages.

7 Specify 2 to perform a custom configuration of SLM for IGA.

8 Enter 1 to use the default evaluation license key.

or

Enter 2 to enter a purchased license key for SLM for IGA.

9 Specify the password for the administrator user admin and confirm the password again.

10 Specify the password for the database user dbauser and confirm the password again.

The dbauser account is the identity used by SLM for IGA to interact with the database. The password you enter here can be used to perform database maintenance tasks, including resetting the admin password if the admin password is forgotten or lost.

11 Specify the password for the application user appuser and confirm the password again.

12 Change the port assignments by entering the required number.

For example, the default port for Web Server is 8443. To modify the port number for Web Server, specify 4. Enter the new port value for Web Server, for example, 8643.

13 After you have changed the ports, specify 8 for done.

14 Enter 1 to authenticate users using only the internal database.

or

If you have configured an LDAP directory in your domain, enter 2 to authenticate users by using LDAP directory authentication.

The default value is 1.

15 Enter n when you are prompted to enable FIPS 140-2 mode.

16 Enter n when you are prompted to enable scalable storage.

The installation finishes and the server starts. It might take few minutes for all services to start after installation because the system performs a one-time initialization. Wait until the installation finishes before you log in to the Sentinel server.

To access the SLM for IGA main interface, specify the following URL in your web browser:

```
https://<IP_Address/DNS_SLM for IGA_server>:<port>/sentinel/views/
main.html
```

Where <*IP_Address/DNS_SLM for IGA_server*> is the IP address or DNS name of the SLM for IGA server and <*port*> is the port for the SLM for IGA server.

Installing Sentinel Log Management as a Non-root User

You must install Sentinel log management as the `novell` user. NetIQ does not support non-root installations other than the `novell` user, although the installation proceeds successfully.

1 Create a non-root user called `novell`.

```
useradd novell
```

2 Create a non-root group called `novell`.

```
groupadd novell
```

3 Create a directory for installing Sentinel.

```
mkdir /home/slmnonroot
```

4 Assign the `novell` user permissions to the non-root installation directory.

```
chown novell:novell /home/slmnonroot
```

5 Set the appropriate password.

```
sudo passwd <password>
```

6 Log in as `novell` user.

7 Download the `SentinelLogManagementForIGA8.2.2.0.tar.gz` from the NetIQ downloads Website.

8 Navigate to the directory where you want to extract the file.

9 Run the following command to extract the file

```
tar -zxvf SentinelLogManagementForIGA8.2.2.0.tar.gz
```

10 Navigate to the `SentinelLogManagementforIGA` directory.

11 Run the following command:

```
./install.sh --location=/home/slmnonroot
```

12 Follow steps 6 to 16 in the "Installing Sentinel Log Management as a Root User" on page 73.

Installing Java Remote Loader

You install the Java Remote Loader, `dirxml_jremote`, on computers where the operating system is not compatible with the native Remote Loader. However, the Java Remote Loader can also run on the same servers where you might install the native Remote Loader. Identity Manager uses the Java Remote Loader to exchange data between the Identity Manager engine running on one server and the Identity Manager drivers running in another location, where `rdxml` does not run. You can install `dirxml_jremote` on any supported Linux computer with any publicly supported version of Java.

1 On the server that hosts the Identity Manager engine, copy the application shim `.iso` or `.jar` files, located by default in the `/opt/novell/eDirectory/lib/dirxml/classes` directory.

2 Log in to the computer where you want to install the Java Remote Loader (the target computer).

3 Verify that the target computer has a supported version of JRE.

4 To access the installation program, complete one of the following steps:

   4a (Conditional) If you have the `.iso` image file for the Identity Manager installation package, navigate to the directory containing the Java Remote Loader installation files, located by default in `/IDM/packages/java_remoteloader`.

   4b (Conditional) If you downloaded the Java Remote Loader installation files from the NetIQ Downloads website, complete the following steps:

      4b1 Navigate to the `.tgz` file for the downloaded image.

      4b2 Extract the contents of the file to a folder on the local computer.

5 Copy the `dirxml_jremote_dev.tar.gz` file to the desired location on the target computer. For example, copy the file to `/usr/idm`.

6 Copy one of the following files to the desired location on the target computer:

   - `dirxml_jremote.tar.gz`
   - `dirxml_jremote_mvs.tar`

     For information about mvs, untar the `dirxml_jremote_mvs.tar` file, then refer to the `usage.html` document.

7 On the target computer, unzip and extract the `.tar.gz` files.

   For example, `tar -zxvf dirxml_jremote.tar.gz`

8 Place the `.iso` or `.jar` files for the application shim that you copied in Step 1 in the `dirxml/classes` directory under the `lib` directory.

9 To customize the `dirxml_jremote` script so the Java executable is reachable through the RDXML_PATH environment variable, complete one of the following steps:

   9a Enter one of the following commands to set the environment variable RDXML_PATH:

      - `set RDXML_PATH=path`
      - `export RDXML_PATH`

   9b Edit the `dirxml_jremote` script and prepend the path to the Java executable on the script line that executes Java.

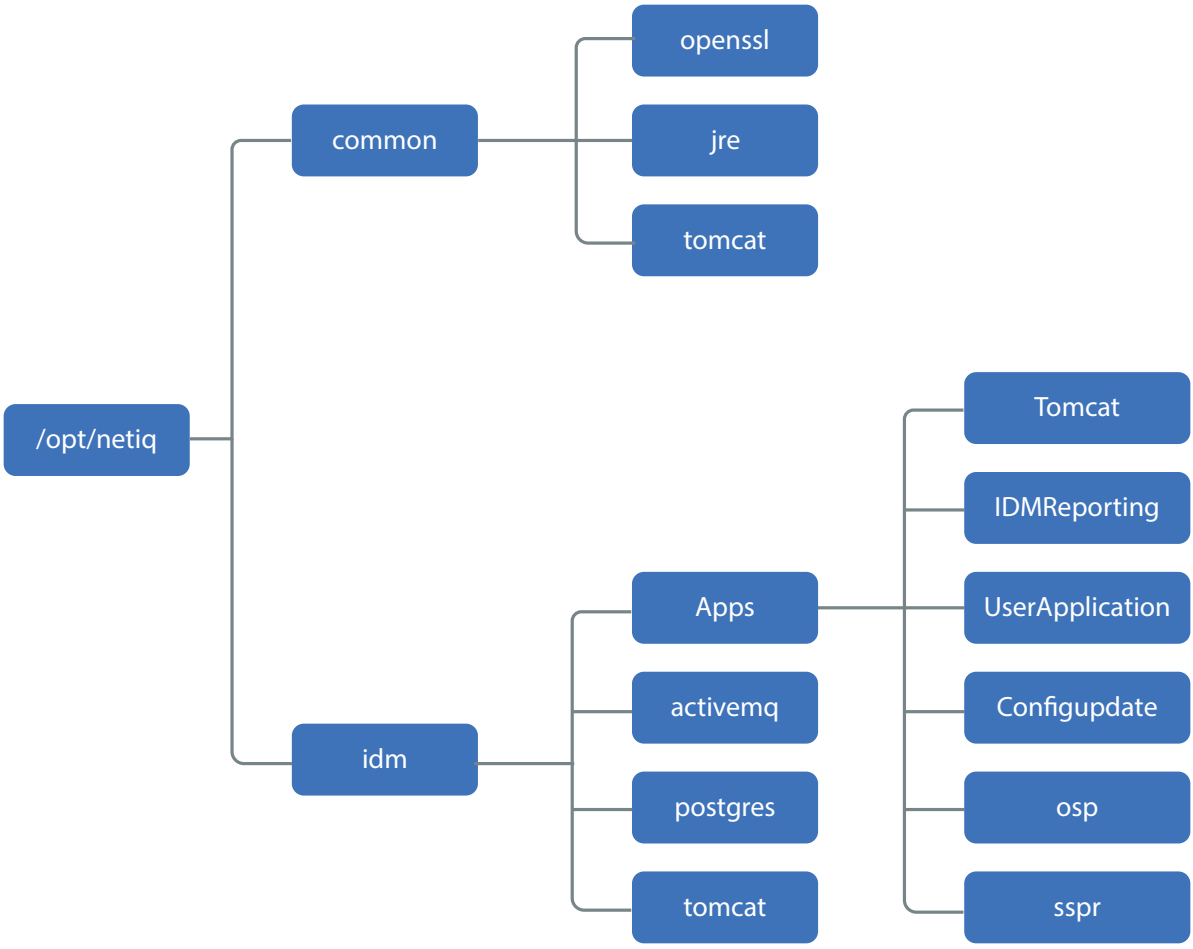
10 You must specify the location of the jar files in the `dirxml_jremote` script from the lib subdirectory of the untarred `dirxml_jremote.tar.gz` directory. For example, `/lib/*.jar`.

11 Configure the sample configuration file `config8000.txt` for use with your application shim.

The sample file is located by default in the `/opt/novell/dirxml/doc` directory. For more information, see "Configuring the Remote Loader and Drivers" on page 94.

Understanding the Directory Structure

The installation process creates the following directory structure:



- `/opt/netiq` directory is the starting point of your directory structure. Every other file and directory is under this directory.
- `common` directory contains supporting software. This software is shared among the components that require them.
- `idm` directory contains component-specific subdirectories that include binary files for installing and configuring the components.

5 Configuring the Identity Manager Components

This section guides you through the process of configuring Identity Manager components. You must review the configuration options for each component before beginning the configuration process. For more information, see "Understanding the Configuration Parameters" on page 77.

Some components such as Designer, Analyzer, Remote Loader, and Fanout Agent might not require configuration.

Using Non-Intuitive Passwords During Configuration

Many of the Identity Manager components require you to specify a password during the configuration phase. For faster configuration, you can instruct the process to apply the same password to all the configuration parameters.

The password must be a minimum of six characters. Do not use words that can be found in the dictionary. Dictionary words are vulnerable to freely available password-cracking tools that often come with dictionary lists. If you must use dictionary words, try combining them with numerals and punctuation.

Understanding the Configuration Parameters

This section defines the parameters that you need to specify to configure the Identity Manager installation. You can use the installation program to configure the components immediately after installing them or configure the components later by running the `configure.sh` script.

NOTE: ◆Identity Applications and Identity Reporting configured in typical configuration mode cannot connect to a database server installed on a different computer.

- The installation process does not allow you to enable auditing for Identity Manager components. You must configure auditing for each component after completing the installation. For more information, see *NetIQ Identity Manager - Configuring Auditing in Identity Manager*.
- Identity Vault is installed automatically with OES. To configure Identity Manager Engine on OES platform, you must select **Custom Configuration** and then select **Add to an Existing Vault**.
- Ensure that the Identity Applications, Identity Reporting, and the databases are uniformly configured with either the FQDN or IP address. In other words, you must not configure these components with a combination of FQDN and IP addresses.
- For containers, it is recommended to specify the FQDN value instead of the IP address.

Table 5-1 describes the parameters required for configuring Identity Manager components in typical mode.

Table 5-1 Typical Configuration

Parameter	Parameter in the Silent Properties File	Typical Configuration
Identity Manager Engine		
Common password	IS_COMMON_PASSWORD	Specifies whether you want to set a common password.
Identity Vault Administrator name	ID_VAULT_ADMIN_LDAP	Specifies the relative distinguished name (RDN) of the administrator object in the tree that has full rights, at least to the context to which this server is added.
Identity Applications		
Common password	IS_COMMON_PASSWORD	Specifies whether you want to set a common password. Ensure that the password meets the considerations specified in the "Using Non-Intuitive Passwords During Configuration" on page 77 section.
Identity Vault Administrator name	ID_VAULT_ADMIN_LDAP	Specifies the relative distinguished name (RDN) of the administrator object in the tree that has full rights, at least to the context to which this server is added.
Hostname (FQDN in lowercase)		Specifies the fully qualified distinguished name or the default IP address of the server.
Application Server DNS/IP address	TOMCAT_SERVLET_HOSTNAME	Specifies the IP address of the Tomcat server.
Identity Applications administrator name	UA_ADMIN	Specifies the name of the administrator account for the identity applications.
Identity Reporting		
Common password	IS_COMMON_PASSWORD	Specifies whether you want to set a common password. Ensure that the password meets the considerations specified in the "Using Non-Intuitive Passwords During Configuration" on page 77 section.
Identity Vault Hostname/IP Address	ID_VAULT_HOST	Specifies the IP address of the server where Identity Vault is installed.
Identity Vault Administrator Name	ID_VAULT_ADMIN_LDAP	Specifies the relative distinguished name (RDN) of the administrator object in the tree that has full rights, at least to the context to which this server is added.
Identity Vault Administrator Password	ID_VAULT_PASSWORD	Specifies the password for the Administrator object. For example, *password*.
Hostname (FQDN in lowercase)		Specifies the fully qualified distinguished name or the default IP address of the server.
Connect to an external One SSO server		Specifies whether you want to a connect to a different One SSO server.

Parameter	Parameter in the Silent Properties File	Typical Configuration
Application server DNS/IP address	TOMCAT_SERVLET_HOSTNAME	Specifies the IP address of the Tomcat server.
One SSO server DNS/IP address	SSO_SERVER_HOST	Specifies the IP address of the server where single sign-on service is installed.
Identity Reporting One SSO Service password	RPT_SSO_SERVICE_PWD	Specifies the password for the authentication service for Identity Reporting.
Identity Reporting Administrator name	RPT_ADMIN	Specifies the administrator name for Identity Reporting. The default value is cn=uaadmin,ou=sa,o=data.
Identity Reporting database account password	RPT_DATABASE_SHARE_PASSWORD	Specifies the database account password for Identity Reporting.

Table 5-2 describes the parameters required for configuring Identity Manager components in custom mode.

Table 5-2 *Custom Configuration*

Parameter	Parameter In the Silent Properties File	Custom Configuration
Identity Manager Engine		
Create a new Identity Vault	TREE_CONFIG	Specifies the Identity Vault to be installed.
Add to an Identity Vault existing on local machine		Specifies whether you want to connect to an existing Identity Vault on the same server where you are installing Identity Manager Engine.
Add to an Identity Vault existing on remote machine		Specifies whether you want to connect to an Identity Vault installed on a different server than Identity Manager Engine.
Identity Vault Tree Name	ID_VAULT_TREENAME	Specifies a new tree for your Identity Vault. The tree name must meet the following requirements: ◆ The tree name must be unique in your network. ◆ The tree name must be 2 to 32 characters long. ◆ The tree name must contain only characters such as letters (A-Z), numbers (0-9), hyphens (-), and underscores (_). **NOTE:** If you are installing Identity Manager on OES, specify the existing tree name.

Parameter	Parameter In the Silent Properties File	Custom Configuration
Identity Vault Administrator Name	`ID_VAULT_ADMIN_LDAP`	Specifies the relative distinguished name (RDN) of the administrator object in the tree that has full rights, at least to the context to which this server is added.
Identity Vault Administrator Password	`ID_VAULT_PASSWORD`	Specifies the password for the Administrator object. For example, *password*.
NDS var folder location	`ID_VAULT_VARDIR`	Specifies the path of this Identity Vault instance on this server. The default path is `/var/opt/novell/eDirectory`.
NDS data location	`ID_VAULT_DIB`	Specifies the path in the local system where you want to install the Directory Information Base (DIB) files.The DIB files are your Identity Vault database files. The default location is `/var/opt/novell/eDirectory/data/dib`.
NCP Port	`ID_VAULT_NCP_PORT`	Specifies the NetWare Core Protocol (NCP) port that the Identity Vault uses to communicate with the Identity Manager components. The default value is `524`.
LDAP non SSL port	`ID_VAULT_LDAP_PORT`	Specifies the port on which the Identity Vault listens for LDAP requests in clear text. The default value is `389`.
LDAP SSL port	`ID_VAULT_LDAPS_PORT`	Specifies the port on which the Identity Vault listens for LDAP requests using Secure Sockets Layer (SSL) protocol. The default value is `636`.
Identity Vault Context DN	`ID_VAULT_SERVER_CONTEXT`	Specifies the context DN of the existing Identity Vault server. The default value is `servers.system`.
Identity Vault HTTP Port	`ID_VAULT_HTTP_PORT`	Specifies the port on which the HTTP stack operates in clear text. The default value is `8028`.
Identity Vault HTTPS Port	`ID_VAULT_HTTPS_PORT`	Specifies the port on which the HTTP stack operates using TLS/SSL protocol. The default value is `8030`.
NDS configuration file with path	`ID_VAULT_CONF`	Specifies the location of the configuration file for Identity Vault. The default value is `/etc/opt/novell/eDirectory/conf/nds.conf`.
Identity Vault driver set name	`ID_VAULT_DRIVER_SET`	Specifies the name for a new Identity Manager driver set object.
RSA Key Size	`ID_VAULT_RSA_KEYSIZE`	Specify the key size for RSA certificates. Allowed values are 2048, 4096, and 8192 bits. The default value is `4096`.
EC Curve	`ID_VAULT_EC_CURVE`	Specify the elliptical curve (EC) limit for EC certificates. Allowed values are P256, P384, and P521. The default value is P384.

Parameter	Parameter In the Silent Properties File	Custom Configuration
Certificate Lifetime	`ID_VAULT_CA_LIFE`	*Applies only if you have selected the* **Create a New Tree** *option.* Specify the certificate life in years.
Identity Vault driver set deploy context	`ID_VAULT_DEPLOY_CTX`	Specifies the LDAP DN of the container where you want to create the driver set object.
Custom driverset ldif file path		Specifies the custom path of the sample `driverset.ldif` file. A driver set is a container that holds Identity Manager drivers. Only one driver set can be active on a server at a time. NetIQ provides a `sample-driverset.ldif` file in the Identity Manager installation kit to help you create or configure a driver set. For information about using this file, see "Creating and Configuring a Driver Set" on page 87.
iManager Web Administration		
HTTP Port Number for Tomcat	`IMAN_TOMCAT_HTTP_PORT`	Specifies the HTTP port for Tomcat Application server. The default value is 8080.
SSL Port Number for Tomcat	`IMAN_TOMCAT_SSL_PORT`	Specifies the HTTPS port for Tomcat Application server. The default value is 8443.
Public Key Algorithm that you want TLS certificate to use	`IMAN_CERT_ALGO`	Specifies whether you want to use RSA or ECDSA as the public key algorithm. By default, the public key algorithm is set to RSA. If you select **RSA**, the certificate uses a 2048-bit RSA key pair. If you select **ECDSA**, the certificate uses a ECDSA key pair with curve secp256r1.
Cipher Suite for TLS communication	`IMAN_CIPHER_SUITE_RSA`	If you select **RSA**, it allows the following cipher levels: ◆ **NONE**: Allows any type of cipher. ◆ **LOW**: Allows a 56-bit or a 64-bit cipher. ◆ **MEDIUM**: Allows a 128-bit cipher. ◆ **HIGH**: Allows ciphers that are greater than 128-bit.
Administrative User Context	`IMAN_USER_CONTEXT`	Specifies the user name that you need to use for logging in to iManager.
Administrative User Tree	`IMAN_DIR_TREE`	Specifies the IP address of the server where the Identity Vault tree exists.
Identity Applications		
Common password	`IS_COMMON_PASSWORD`	Specifies whether you want to set a common password. Ensure that the password meets the considerations specified in the "Using Non-Intuitive Passwords During Configuration" on page 77 section.

Parameter	Parameter In the Silent Properties File	Custom Configuration
Hostname (FQDN in lowercase)	`UA_SERVER_HOST`	Specifies the fully qualified distinguished name or the default IP address of the server. **NOTE:** Ensure that FQDN is specified in lower case. The server hosting your component must also be configured to use FQDN in lower case.
Identity Vault Hostname/IP Address	`ID_VAULT_HOST`	Specifies the IP address or hostname of the server where Identity Vault is installed.
Identity Vault Administrator Name	`ID_VAULT_ADMIN_LDAP`	Specifies the relative distinguished name (RDN) of the administrator object in the tree that has full rights, at least to the context to which this server is added.
Identity Vault Administrator Password	`ID_VAULT_PASSWORD`	Specifies the password for the Administrator object. For example, *password*.
Application server DNS/IP address	`TOMCAT_SERVLET_HOSTNAME`	Specifies the IP address or hostname of the Tomcat server.
OSP custom login screen name	`OSP_CUSTOM_NAME`	Specifies the name that will be displayed on the OSP login screen.
SSPR Configuration password	`CONFIGURATION_PWD`	*Applies only if you have set the common password as* **No**. Specifies the password for password management used by identity applications.
OAuth keystore password	`OSP_KEYSTORE_PWD`	*Applies only if you have set the common password as* **No**. Specifies the password that you want to create for loading the new keystore on the OAuth server.
User search container DN	`USER_CONTAINER`	Specifies the default container for all user objects in the Identity Vault.
Admin search container DN	`ADMIN_CONTAINER`	Specifies the distinguished name of the container in the Identity Vault that contains any administrator User objects that the authentication service (OSP) must authenticate. For example, `o=data`.
Application Server HTTPS port	`TOMCAT_HTTPS_PORT`	Specifies the HTTPS port that you want the Tomcat server to use for communication with client computers. The default value is 8543.
One SSO server SSL port	`SSO_SERVER_SSL_PORT`	Specifies the port that you want the single sign-on service to use. The default value is 8543.
Identity Application One SSO Service password	`SSO_SERVICE_PWD`	*Applies only if you have set the common password as* **No**. Specifies the password for the single sign-on client used by identity applications.

Parameter	Parameter In the Silent Properties File	Custom Configuration
Identity Applications administrator name	`UA_ADMIN`	Specifies the name of the administrator account for the identity applications.
Database Platform	`UA_DB_PLATFORM_OPTION`	Specifies the databases required for Identity Applications.
Configure PostgreSQL on current server	`INSTALL_PG_DB`	Specifies if you want to configure PostgreSQL database on the same server.
Identity Applications database port	`UA_DB_PORT`	Specifies the database port for Identity Applications.
Identity Applications database name	`UA_DATABASE_NAME`	Specifies the name of the database. The default value is `idmuserappdb`.
Identity Applications database and Workflow Engine user name	`UA_WFE_DATABASE_USER`	Specifies the user name for the administrator of the database for the identity applications.
Identity Applications and Workflow Engine database JDBC driver jar file	`UA_WFE_DB_JDBC_DRIVER_JAR`	Specifies the JAR file for the database platform.
Identity Applications and Workflow Engine database administrator password for user postgres	`UA_WFE_DATABASE_ADMIN_PWD`	Specifies the Identity Applications and Workflow Engine database administrator password for user postgres
Create schema	`UA_WFE_DB_CREATE_OPTION`	Indicates when you want to create the database schema for Identity Applications and Workflow Engine as part of the installation process. The available options are **Now**, **Startup**, and **File**.
Create a new database or upgrade/migrate from an existing database	`UA_DB_NEW_OR_EXIST`	Specifies whether you want to create a new database or upgrade from an existing database.
Create a new database or upgrade/migrate from an existing database	`WFE_DB_NEW_OR_EXIST`	Specifies whether you create tables in a new database or update/migrate from an existing database for Workflow Engine. Supported values are new and exist.
Use custom container as root container	`ENABLE_CUSTOM_CONTAINER_CREATION`	Specifies whether you want to use custom container as a root container. By default, the installer creates `o=data` and chooses it as a user container and assigns the password policies and required trustee rights. To create a custom container, choose **Yes**.

Parameter	Parameter In the Silent Properties File	Custom Configuration
Custom container LDIF file path		*Applies only if you have set the custom container as* **Yes**. Specifies the path of the LDIF file for custom container.
Root container	`ROOT_CONTAINER`	Specifies the root container. The default value is `o=data`.
Group search root container DN	`GROUP_ROOT_CONTAINER`	Specifies the DN of the group search root container.
Create the User Application and Roles and Resources Services drivers for Identity Applications	`UA_CREATE_DRIVERS`	Specifies whether you want to install the UA and RRSD drivers. If you select **N**, you must specify the name of the existing User Application driver.
Name of the existing User Application driver	`UA_DRIVER_NAME`	Applies only if you have set the value for creation of UA and RRSD drivers to **No**. Specifies the existing User Application driver DN details.
User Application Workflow Engine ID	`UA_WORKFLOW_ENGINE_ID`	Indicates Identity Applications workflow engine ID.
Identity Applications and Workflow Engine database platform	`UA_WFE_DB_PLATFORM_OPTION`	Indicates the Identity Applications and Workflow Engine database platform. The supported values are Postgres, Oracle, and MsSQL.
Identity Applications and Workflow Engine database port	`UA_WFE_DB_PORT`	Indicates the Identity Applications and Workflow Engine database port.
Workflow Engine database name	`WFE_DATABASE_NAME`	Indicates the Workflow Engine database name. The default database name is `igaworkflowdb`.
Identity Applications and Workflow Engine database password	`UA_WFE_DATABASE_PWD`	Indicates the Identity Applications and Workflow Engine database password.
Identity Applications Custom Certificate	`CUSTOM_UA_CERTIFICATE`	Indicates whether you want to use a custom certificate for Identity Applications
Identity Applications tomcat keystore file with Subject Alternate Name	`UA_COMM_TOMCAT_KEYSTORE_FILE`	Specifies Identity Applications tomcat keystore file with Subject Alternate Name
Identity Applications tomcat keystore password	`UA_COMM_TOMCAT_KEYSTORE_PWD`	Specifies the Identity Applications tomcat keystore password
NGINX port	`NGINX_HTTPS_PORT`	Specifies the NGINX port.
Identity Reporting		

Parameter	Parameter In the Silent Properties File	Custom Configuration
Common password	`IS_COMMON_PASSWORD`	Specifies whether you want to set a common password. Ensure that the password meets the considerations specified in the "Using Non-Intuitive Passwords During Configuration" on page 77 section.
Hostname (FQDN in lowercase)		Specifies the fully qualified distinguished name or the default IP address of the server. **NOTE:** Ensure that FQDN is specified in lower case. The server hosting your component must also be configured to use FQDN in lower case.
Identity Vault Hostname/IP Address	`ID_VAULT_HOST`	Specifies the IP address or hostname of the server where Identity Vault is installed.
Identity Vault Administrator name	`ID_VAULT_ADMIN_LDAP`	Specifies the relative distinguished name (RDN) of the administrator object in the tree that has full rights, at least to the context to which this server is added.
Identity Vault Administrator password	`ID_VAULT_PASSWORD`	Specifies the password for the Administrator object. For example, *password*.
Connect to an external One SSO Server		Specifies whether you want to connect to an external SSO server
Application server DNS/IP address	`TOMCAT_SERVLET_HOSTNAME`	Specifies the IP address or hostname of the Tomcat server.
OSP custom login screen name	`OSP_CUSTOM_NAME`	Specifies the name that will be displayed on the OSP login screen.
User search container DN	`USER_CONTAINER`	Specifies the default container for all user objects in the Identity Vault.
Admin search container DN	`ADMIN_CONTAINER`	Specifies the distinguished name of the container in the Identity Vault that contains any administrator User objects that the authentication service (OSP) must authenticate. For example, `o=data`.
Application Server HTTPS port	`TOMCAT_HTTPS_PORT`	Specifies the HTTPS port that you want the Tomcat server to use for communication with client computers. The default value is 8543.
One SSO server DNS/IP address	`SSO_SERVER_HOST`	Specifies the IP address of the server where single sign-on service is installed.
One SSO server SSL port	`SSO_SERVER_PORT`	Specifies the port that you want the single sign-on service to use. The default value is 8543.
OSP Tomcat keystore file with subject alternate name	`OSP_COMM_TOMCAT_KEYSTORE_FILE`	Specifies the location of the Tomcat keystore file. **NOTE:** The custom certificates are supported only with PKCS type.
OAuth Keystore Password	`OSP_KEYSTORE_PWD`	Specifies the OAuth keystore password.

Parameter	Parameter In the Silent Properties File	Custom Configuration
Application Server Keystore Password	`TOMCAT_SSL_KEYSTORE_PASS`	Specifies the keystore password for the application server.
Identity Applications Tomcat keystore file with subject alternate name	`UA_COMM_TOMCAT_KEYSTORE_FILE`	Specifies the location of the Tomcat keystore file. **NOTE:** The custom certificates are supported only with PKCS type.
Identity Reporting One SSO Service password	`RPT_SSO_SERVICE_PWD`	Specifies the password for the authentication service for Identity Reporting.
Select the database platform for Identity Reporting	`RPT_DATABASE_PLATFORM_OPTION`	Specifies the database that you want to use for Identity Reporting.
Configure PostgreSQL on current server	`INSTALL_PG_DB_FOR_REPORTING`	Specifies if you want to configure PostgreSQL database on the same server.
Identity Reporting database account password	`RPT_DATABASE_SHARE_PASSWORD`	Specifies the database account password for Identity Reporting.
Create a new database or upgrade/migrate from an existing database	`RPT_DATABASE_NEW_OR_EXIST`	Specifies whether you want to create a new database or upgrade from an existing database.
Identity Reporting Administrator name	`RPT_ADMIN`	Specifies the administrator name for Identity Reporting. The default value is `cn=uaadmin,ou=sa,o=data`.
Identity Reporting Administrator password	`RPT_ADMIN_PWD`	Specifies the administrator password for Identity Reporting.
Identity Reporting database name	`RPT_DATABASE_NAME`	Specifies the database name for Identity Reporting. The default value is `idmrptdb`.
Identity Reporting database user	`RPT_DATABASE_USER`	Specifies the administration account that allows Identity Reporting to access and modify data in the databases. The default value is `rptadmin`.
Identity Reporting database host		Specifies the DNS name or IP address of the server where the database has to be created.
Identity Reporting database port	`RPT_DATABASE_PORT`	Specifies the port to connect to the database.The default port is 5432.
Identity Application database JDBC jar file	`RPT_DATABASE_JDBC_DRIVER_JAR`	Specifies the JAR file for the database platform.
Identity Reporting Tomcat keystore file with subject alternate name	`RPT_COMM_TOMCAT_KEYSTORE_FILE`	Specifies the location of the Tomcat keystore file. **NOTE:** The custom certificates are supported only with PKCS type.

Parameter	Parameter In the Silent Properties File	Custom Configuration
Create schema	RPT_DATABASE_CREATE_OPTION	Indicates when you want to create the database schema as part of the installation process. The available options are **Now**, **Startup**, and **File**. If you select the database schema creation option as **Startup** or **File**, you must manually add the datasource to the Identity Data Collection Services page. For more information, see "Manually Adding the DataSource in the Identity Data Collection Services Page" on page 142. If your database is running on a separate server, you must connect to that database. For a remotely installed PostgreSQL database, verify that the database is running. To connect to a remote PostgreSQL database, see "Connecting to a Remote PostgreSQL Database" on page 146. If you are connecting to an Oracle database, ensure that you have created an Oracle database instance. For more information, see Oracle documentation. If you select the database schema creation option as **Startup** or **File**, you must manually create the tables and connect to the database after the configuration. For more information, see "Manually Generating the Database Schema" on page 142.
Default email address	RPT_DEFAULT_EMAIL_ADDRESS	Specifies the email address that you want Identity Reporting to use as the origination for email notifications.
SMTP Server	RPT_SMTP_SERVER	Specifies the IP address or DNS name of the SMTP email host that Identity Reporting uses for notifications.
SMTP Server port	RPT_SMTP_SERVER_PORT	Specifies the port number for the SMTP server. The default port is 465.
Create the MSGW and DCS drivers for Identity Reporting	RPT_CREATE_DRIVERS	Specifies whether you want to create the MSGW and DCS drivers.
NGINX HTTPS port	NGINX_HTTPS_PORT	Specifies the port for NGINX.

Creating and Configuring a Driver Set

Use the `sample-driverset.ldif` file from `IDM/LDIF/` directory of the Identity Manager installation kit to help you create a driver set. The file has the following contents:

```
dn: cn=driverset1,o=system
changetype: add
DirXML-LogLimit: 0
DirXML-ConfigValues::
PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz48Y29u

ZmlndXJhdGlvbi12YWx1ZXM+Cgk8ZGVmaW5pdGlvbnMvPgo8L2NvbmZpZ3VyYXRpb24tdmFsdW
VzPg==
objectClass: DirXML-DriverSet
objectClass: Top
objectClass: Partition
objectClass: nsimPasswordPolicyAux

dn: cn=DirXML-PasswordPolicy,cn=Password Policies,cn=Security
changetype: add
nsimPwdRuleEnforcement: FALSE
nspmSpecialAsLastCharacter: TRUE
nspmSpecialAsFirstCharacter: TRUE
nspmSpecialCharactersAllowed: TRUE
nspmNumericAsLastCharacter: TRUE
nspmNumericAsFirstCharacter: TRUE
nspmNumericCharactersAllowed: TRUE
description: This Password Policy is used by IDM Engine
nspmMaximumLength: 64
nspmConfigurationOptions: 596
passwordUniqueRequired: FALSE
passwordMinimumLength: 1
passwordAllowChange: TRUE
objectClass: nspmPasswordPolicy
objectClass: Top
cn: DirXML-PasswordPolicy
nsimAssignments: cn=driverset1,o=system
```

Creating a Driver Set in a New Installation

In a text editor, open the `sample-driverset.ldif` file and make the following changes. If you have provided the custom LDIF file path, then specify the custom path:

1. Point the driver set DN to the new driver set. For example, change `dn: cn=driverset1,o=system` to `dn:cn=Driverset47,ou=drivers,o=acme`.
2. Change the `nsimAssignments` attribute value to the DN of the new driver set. For example, change `nsimAssignments: cn=driverset1,o=system` to `nsimAssignments: cn=Driverset47,ou=drivers,o=acme`.

NOTE: Copying the content as is might insert some hidden special characters in the file. If you receive a `ldif_record() = 17` error message when you add these attributes to the Identity Vault, insert an extra space between the two DNs.

Configuring a Driver Set on an Existing Server

If Identity Manager is already installed on a server in the eDirectory tree, the DirXML-PasswordPolicy object exists in the tree. You have the following choices:

- **Use the existing password policy**

  Change

  ```
  dn: cn=DirXML-PasswordPolicy,cn=Password Policies,cn=Security
  changetype: modify
  add: nsimAssignments
  nsimAssignments: cn=driverset1,o=system
  ```

- **Use a different password policy**

  Use

  ```
  dn: cn=DirXML-PasswordPolicy,cn=Password Policies,cn=Security
  changetype: add
  ```

In a text editor, open the `sample-driverset.ldif` file and make the following changes:

1. Point the driver set DN to the new driver set.
2. Change the nsimAssignments attribute value to the DN of the new driver set.
3. Change the DirXML-PasswordPolicy attribute to point to the existing DirXML-PasswordPolicy object or a different password policy.

Configuring the Identity Manager Components

You can perform the configuration in the following ways:

- Interactive Configuration
- Silent Configuration

Performing an Interactive Configuration

1. Navigate to the location where you mounted the `Identity_Manager_4.8_Linux.iso` file.
2. Specify the following command at the command line to run the `configure.sh` script:

   ```
   ./configure.sh
   ```

3. Decide whether you want to perform a typical configuration or a custom configuration. The configuration options will vary based on the components that you select for configuration.
4. To configure the components, use the information from "Understanding the Configuration Parameters" on page 77.

NOTE: During the Identity Manager Engine configuration process, the TLS for simple binds is set to **Yes**. If you want to change the value of the TLS for simple binds to **No**, run the following command on the server where Identity Manager Engine is installed:

```
ldapconfig set "Require TLS for Simple Binds with Password=no"
```

Performing a Silent Configuration

1 Navigate to the location where you mounted the `Identity_Manager_4.8_Linux.iso` file.

2 To run the silent installation, execute the following command:

```
./configure.sh -s -f <location of the silent properties file>
```

For example,

`./configure.sh -s -f /home/silent.properties`, where `/home/silent.properties` is the location where you stored the silent properties file.

3 To configure the components, use the information from "Understanding the Configuration Parameters" on page 77.

Configuring SSPR

The following sections provide information about configuring SSPR. Before configuring the components, review the information from "Understanding the Configuration Parameters" on page 77.

NOTE: Ensure that the following containers and user objects are present in the Identity Vault before configuring SSPR:

- User Search Container
- Admin Search Container
- Identity Applications Administrator User

Performing an Interactive Configuration

1 Navigate to the location where you mounted the `Identity_Manager_4.8_Linux.iso`.

2 Navigate to the `sspr` directory.

3 Execute the following command:

```
./configure.sh
```

4 Configure the settings.

Performing a Silent Configuration

Before starting the configuration, ensure that `sspr_silentinstall.properties` (`<iso mounted path>/sspr/`) file is copied to a writable directory. For example, copy the file to `/tmp` directory.

1 Navigate to the location where you mounted the `Identity_Manager_4.8_Linux.iso`.

2 Navigate to the `sspr` directory.

3 Execute the following command:

```
./configure.sh -s -f <location of the silent properties file>
```

For example,

`./configure.sh -s -f /tmp/sspr_silentinstall.properties`, where `/tmp/sspr_silentinstall.properties` is the location where you stored the silent properties file.

4 Configure the settings.

Modifying the Single Sign-on Access Settings on the OSP Server

If SSPR and OSP are installed on separate servers (SSPR is installed on a different server than Identity Applications or Identity Reporting), you must configure the single sign-on access settings on the OSP server. This section helps you ensure that the settings work for your environment.

1 Launch the RBPM Configuration update utility on the server where OSP is installed.

2 Navigate to **SSO Clients** > **Self Service Password Reset**.

3 Specify the SSPR server details in the **OSP OAuth Redirect URL** field. For example, `https://<SSPR Hostname IP>:port/sspr/public/oauth`.

4 Click **OK** to save your changes, then close the configuration utility.

5 Restart Tomcat for the changes to take effect.

6 Final Steps for Completing the Installation

After completing the installation and configuration of Identity Manager components, you must perform certain tasks to make your solution work properly in your environment. For example, configure the drivers you installed to meet the policies and requirements defined by your business processes and configure Sentinel Log Management for IGA to gather audit events.

Post-installation tasks typically include the following items:

- "Configuring the Identity Vault" on page 93
- "Configuring a Non-Administrator User as an Identity Vault Administrator" on page 94
- "Configuring the Remote Loader and Drivers" on page 94
- "Configuring a Connected System" on page 95
- "Configuring Forgotten Password Management" on page 98
- "Configuring Identity Applications" on page 102
- "Configuring the Runtime Environment for Data Collection" on page 132
- "Configuring Identity Reporting" on page 141
- "Completing a Non-root Installation" on page 147
- "Activating Identity Manager" on page 148
- "Creating SSL Certificate for Distributed Deployment" on page 148

Configuring the Identity Vault

- Creating Value Indexes for Identity Vault
- Manually Importing Identity Applications and Identity Reporting Certificates into Identity Vault

Creating Value Indexes for Identity Vault

The identity applications must be able to interact with the objects in your Identity Vault. To improve the performance of the identity applications, the Identity Vault Administrator should create value indexes for the manager, ismanager, and srvprvUUID attributes. Without value indexes on these attributes, the identity applications users can experience impeded performance, particularly in a clustered environment.

You can create these value indexes after completing the identity applications installation by using one of the following methods:

- iManager. Use Index Manager. For more information, see Creating an Index in the *NetIQ eDirectory Administration Guide*.
- Configuration utility. Navigate to Miscellaneous > **Identity Vault Indexes**, then select **Create** from **Server DN** and specify a value for it. Click **OK** and then restart the Identity vault to save your changes.

Manually Importing Identity Applications and Identity Reporting Certificates into Identity Vault

- If you have custom certificates for Identity Applications and Identity Reporting components, import those certificates into cacerts in the Identity Vault (`/opt/netiq/common/jre/lib/security/cacerts`).

  For example, you can use the following keytool command to import certificates into the Identity Vault:

  ```
  keytool -import -trustcacerts -alias <User Application certificate alias name> -keystore <cacerts file> -file <User Application certificate file>
  ```

- If you install SSPR on a different server than the User Application server, import the SSPR application certificate into idm.jks in the User Application (`/opt/netiq/idm/apps/tomcat/conf/idm.jks`).

  For example, you can use the following keytool command to import certificates into User Application:

  ```
  keytool -import -trustcacerts -alias <SSPR certificate alias name> -keystore <idm.jks> -file <SSPR certificate file>
  ```

Configuring a Non-Administrator User as an Identity Vault Administrator

If Identity Applications are configured to use a non-administrator user as an Identity Vault Administrator, the non-administrator user must have [write] rights to the oidpInstanceData attribute in the subtree where the users reside. Otherwise, OSP logins can fail.

To set the write rights on the oidpInstanceData attribute for a non-administrator user:

1. Log in to iManager.
2. In the **Roles and Tasks** view, click select **Rights > Modify Trustees**.
3. Select the non-administrator user object, then click **Add Trustee**.
4. For oidpInstanceData attribute, set the **Compare**, **Read**, and **Write** rights.
5. Click **Apply** to save and apply your changes.

Configuring the Remote Loader and Drivers

Remote Loader allows Identity Manager drivers to access the connected application without requiring to install Identity Vault and Identity Manager engine on the same server as the application. Using Remote Loader requires you to configure the application shim so that it can securely connect with the Identity Manager engine. You must also configure both the Remote Loader and Identity Manager drivers. This information is provided in detail in Configuring the Remote Loader and Drivers in the *NetIQ Identity Manager Driver Administration Guide*.

Configuring a Connected System

Identity Manager enables applications, directories, and databases to share information. For driver-specific configuration instructions, see the Identity Manager Driver Documentation.

Creating and Configuring a Driver Set

A driver set is a container that holds Identity Manager drivers. Only one driver set can be active on a server at a time. You can use the Designer tool to create a driver set.

To support password synchronization to the Identity Vault, Identity Manager requires that driver sets have a password policy. You can use the Default Universal Password Policy package in Identity Manager or create a password policy based on your existing organizational requirement. However, the password policy must include the `DirMXL-PasswordPolicy` object. If the policy object does not exist in the Identity Vault, you can create the object.

- "Creating Driver Set" on page 95
- "Assigning the Default Password Policy to Driver Sets" on page 95
- "Creating the Password Policy Object in the Identity Vault" on page 96
- "Creating a Custom Password Policy" on page 97
- "Creating the Default Notification Collection Object in the Identity Vault" on page 97

Creating Driver Set

Designer for Identity Manager provides many settings to create and configure a driver set. These settings allow you to specify Global Configurations Values, driver set packages, driver set named passwords, log levels, trace levels, and Java Environment Parameters. For more information, see "Configuring Driver Sets" in the *NetIQ Designer for Identity Manager Administration Guide*.

Assigning the Default Password Policy to Driver Sets

You must assign the DirMXL-PasswordPolicy object to each driver set in the Identity Vault. The Identity Manager Default Universal Password Policy package includes this policy object. The default policy installs and assigns a universal password policy to control how the Identity Manager engine automatically generates random passwords for drivers.

Alternatively, to use a custom password policy, you must create the password policy object and the policy. For more information, see "Creating the Password Policy Object in the Identity Vault" on page 96 and "Creating a Custom Password Policy" on page 97.

1. Open your project in Designer.
2. In the Outline pane, expand your project.
3. Expand **Package Catalog > Common** to verify whether the Default Universal Password Policy package exists.
4. (Conditional) If the password policy package is not already listed in Designer, complete the following steps:
   - **4a** Right-click **Package Catalog**.
   - **4b** Select **Import Package**.

4c Select **Identity Manager Default Universal Password Policy**, and then click **OK**.

To ensure that the table displays all available packages, you might need to deselect **Show Base Packages Only**.

5 Select each driver set and assign the password policy.

Creating the Password Policy Object in the Identity Vault

If the `DirMXL-PasswordPolicy` object does not exist in the Identity Vault, you can use Designer or the ldapmodify utility to create the object. For more information about how to do this in Designer, see "Configuring Driver Sets" in *NetIQ Designer for Identity Manager Administration Guide*. To use the ldapmodify utility, use the following procedure:

1 In a text editor, create an LDAP Data Interchange Format (LDIF) file with the following attributes:

```
dn: cn=DirXML-PasswordPolicy,cn=Password Policies,cn=Security
changetype: add
nsimPwdRuleEnforcement: FALSE
nspmSpecialAsLastCharacter: TRUE
nspmSpecialAsFirstCharacter: TRUE
nspmSpecialCharactersAllowed: TRUE
nspmNumericAsLastCharacter: TRUE
nspmNumericAsFirstCharacter: TRUE
nspmNumericCharactersAllowed: TRUE
nspmMaximumLength: 64
nspmConfigurationOptions: 596
passwordUniqueRequired: FALSE
passwordMinimumLength: 1
passwordAllowChange: TRUE
objectClass: nspmPasswordPolicy
```

```
dn: cn=DirXML-PasswordPolicy,cn=Password Policies,cn=Security
changetype: modify
add: nsimAssignments
nsimAssignments: <driverset LDAP dn>
```

NOTE: Copying the content as is might insert some hidden special characters in the file. If you receive a `ldif_record() = 17` error message when you add these attributes to the Identity Vault, insert an extra space between the two DNs.

---

2 To add the DirMXL-PasswordPolicy object in the Identity Vault, import the attributes from the file by performing following action:

From the directory containing the ldapmodify utility, enter the following command:

```
ldapmodify -x -c -h hostname_or_IP_address -p 389 -D
"cn=admin,ou=sa,o=system" -w password -f path_to_ldif_file
```

For example:

```
ldapmodify -x -ZZ -c -h server1.test.com -p 389 -D
"cn=admin,ou=sa,o=system" -w test123 -f /root/dirxmlpasswordpolicy.ldif
```

The ldapmodify utility is located by default in the `/opt/novell/eDirectory/bin` directory.

Creating a Custom Password Policy

Rather than using the default password policy in Identity Manager, you can create a new policy based on your organizational requirements. You can assign a password policy to the entire tree structure, a partition root container, a container, or a specific user. To simplify management, NetIQ recommends that you assign password policies as high in the tree as possible. For more information, see Creating Password Policies in the *Password Management 3.3.2 Administration Guide*.

NOTE: You must also assign the DirXML-PasswordPolicy object to the driver sets. For more information, see "Creating the Password Policy Object in the Identity Vault" on page 96.

Creating the Default Notification Collection Object in the Identity Vault

The Default Notification Collection is an Identity Vault object that contains a set of e-mail notification templates and an SMTP server that is used when sending e-mails generated from the templates. If the Default Notification Collection object does not exist in the Identity Vault, use Designer to create the object.

1. Open your project in Designer.
2. In the Outline pane, expand your project.
3. Right-click the Identity Vault, then click Identity Vault **Properties**.
4. Click **Packages**, then click the **Add Packages** icon.
5. Select all the notification templates packages, and then click **OK**.
6. Click **Apply** to install the packages with the **Install** operation.
7. Deploy the notification templates to the Identity Vault.

Creating a Driver

To create drivers, use the package management feature provided in Designer. For each Identity Manager driver you plan to use, create a driver object and import a driver configuration. The driver object contains configuration parameters and policies for that driver. As part of creating a driver object, install the driver packages and then modify the driver configuration to suit your environment.

The driver packages contain a default set of policies. These policies are intended to give you a good start as you implement your data sharing model. Most of the time, you will set up a driver using the shipping default configuration, and then modify the driver configuration according to the requirements of your environment. After you create and configure the driver, deploy it to the Identity Vault and start it. In general, the driver creation process involves the following actions:

1. Importing the Driver Packages
2. Installing the Driver Packages
3. Configuring the Driver Object
4. Deploying the Driver Object
5. Starting the Driver Object

For additional and driver-specific information, refer to the relevant driver implementation guide from the Identity Manager Drivers Web site.

Defining Policies

Policies enable you to customize the flow of information into and out of the Identity Vault, for a particular environment. For example, one company might use the inetorgperson as the main user class, and another company might use User. To handle this, a policy is created that tells the Identity Manager engine what a user is called in each system. Whenever operations affecting users are passed between connected systems, Identity Manager applies the policy that makes this change.

Policies also create new objects, update attribute values, make schema transformations, define matching criteria, maintain Identity Manager associations, and many other things.

NetIQ recommends that you use Designer to define policies for drivers to meet your business needs. For a detailed guide to Policies, see *NetIQ Identity Manager - Using Designer to Create Policies* guide and *NetIQ Identity Manager Understanding Policies Guide*. For information about the document type definitions (DTD) that Identity Manager uses, see Identity Manager DTD Reference. These resources contain:

- A detailed description of each available policy.
- An in-depth Policy Builder user guide and reference, including examples and syntax for each condition, action, noun, and verb.
- A discussion on creating policies using XSLT style sheets.

Configuring Forgotten Password Management

The Identity Manager installation includes Self Service Password Reset to help you manage the process for resetting forgotten passwords. Alternatively, you can use an external password management system.

- "Using Self Service Password Reset for Forgotten Password Management" on page 98
- "Using an External System for Forgotten Password Management" on page 100
- "Updating SSPR Links in the Dashboard for a Distributed or Clustered Environment" on page 102

Using Self Service Password Reset for Forgotten Password Management

In most cases, you can enable the forgotten password management feature when you install SSPR and the identity applications. However, you might not have specified the URL of the landing page for the identity applications to which SSPR forwards users after a password change. You might also need to enable forgotten password management. This section provides the following information:

- "Configuring Identity Manager to Use Self Service Password Reset" on page 99
- "Configuring Self Service Password Reset for Identity Manager" on page 99
- "Locking the SSPR Configuration" on page 100

Configuring Identity Manager to Use Self Service Password Reset

This section provides information about configuring Identity Manager to use SSPR.

1. Log in to the server where you installed the identity applications.
2. Run the RBPM configuration utility. For more information, see "Running the Identity Applications Configuration Utility" on page 103.
3. In the utility, navigate to **Authentication > Password Management**.
4. For **Password Management Provider**, specify **SSPR**.
5. Select **Forgotten Password**.
6. Navigate to **SSO Clients > Self Service Password Reset**.
7. For **OSP client ID**, specify the name that you want to use to identify the single sign-on client for SSPR to the authentication server. The default value is `sspr`.
8. For **OSP client secret**, specify the password for the single sign-on client for SSPR.
9. For **OSP redirect URL**, specify the absolute URL to which the authentication server redirects a browser client when authentication is complete.

   Use the following format: `protocol://server:port/path`.For example, `http://10.10.10.48:8180/sspr/public/oauth`.
10. Save your changes and close the utility.

Configuring Self Service Password Reset for Identity Manager

This section provides information about configuring SSPR to work with Identity Manager. For example, you might want to modify the password policies and challenge response questions.

When you installed SSPR with Identity Manager, you specified a password that an administrator can use to configure the application. NetIQ recommends that you modify the SSPR settings, then specify an administrator account or group can configure SSPR.

NOTE: If you install SSPR on a different server than user application server, ensure that SSPR application certificate is added to user application `cacerts`.

---

1. Log in to SSPR by using the configuration password that you specified during installation.
2. In the Settings page, modify the settings for the password policy and challenge response questions. For more information about configuring the default values for SSPR settings, see Configuring Self Service Password Reset in the *NetIQ Self Service Password Reset Administration Guide*.
3. Lock the SSPR configuration file (SSPR`Configuration.xml`). For more information about locking the configuration file, see "Locking the SSPR Configuration" on page 100.
4. (Optional) To modify SSPR settings after you lock the configuration, you must set the `configIsEditable` setting to `true` in the SSPR`Configuration.xml` file.
5. Log out of SSPR.
6. For the changes to take effect, restart Tomcat.

Locking the SSPR Configuration

1. Go to **http://<IP/DNS name>:<port>/sspr**. This link takes you to the SSPR portal.
2. Log in to the Identity Manager with an administrator account or log in with your existing login credentials.
3. Click **Configuration Manager** at the top of the page and specify the configuration password that you specified during installation.
4. Click **Configuration Editor** and navigate to **Settings > LDAP Settings**.
5. Lock the SSPR configuration file (`SSPRConfiguration.xml`).

   5a Under the Administrator Permission section, define a filter in LDAP format for a user or a group that has administrator rights to SSPR in the Identity Vault. By default, the filter is set to `groupMembership=cn=Admins,ou=Groups,o=example`.

   For example, set it to `uaadmin` (`cn=uaadmin`) for the User Application administrator.

   This prevents users from modifying the configuration in SSPR except the SSPR admin user who has full rights to modify the settings.

   5b To ensure LDAP query returns results, click **View Matches**.

   If there is any error in the setting, you cannot proceed to the next configuration option. SSPR displays the error details to help you troubleshoot the issue.

   5c Click **Save**.

   5d In the confirmation window that pops up, click **OK**.

   When SSPR is locked, the admin user can see additional options in the Administration user interface such as Dashboard, User Activity, Data Analysis, and so on that were not available for him before SSPR lock down.

6. (Optional) To modify SSPR settings after you lock the configuration, you must set the `configIsEditable` setting to `true` in the `SSPRConfiguration.xml` file.
7. Log out of SSPR.
8. Log in to SSPR again as an admin user defined in Step 3.
9. Click **Close Configuration**, then click **OK** to confirm the changes.
10. For the changes to take effect, restart Tomcat.

Using an External System for Forgotten Password Management

To use an external system, you must specify the location of a WAR file containing Forgot Password functionality. This process includes the following activities:

- "Specifying an External Forgotten Password Management WAR File" on page 101
- "Testing the External Forgot Password Configuration" on page 101
- "Configuring SSL Communication between Application Servers" on page 102

Specifying an External Forgotten Password Management WAR File

If you did not specify these values during installation and want to modify the settings, you can use either the RBPM Configuration utility or make the changes in the User Application as an administrator.

1 (Conditional) To modify the settings in the RBPM Configuration utility, complete the following steps:

   1a Log in to the server where you installed the identity applications.

   1b Run the RBPM configuration utility. For more information, see "Running the Identity Applications Configuration Utility" on page 103.

   1c In the utility, navigate to **Authentication > Password Management**.

   1d For **Password Management Provider**, specify **User Application (Legacy)**.

2 (Conditional) To modify the settings in the User Application, complete the following steps:

   2a Log in as the User Application Administrator.

   2b Navigate to **Administration > Application Configuration > Password Module Setup > Login**.

3 For **Forgotten Password**, specify **External**.

4 For **Forgot Password Link**, specify the link shown when the user clicks **Forgot password** on the login page. When the user clicks this link, the application directs the user to the external password management system. For example:

```
http://localhost:8180/ExternalPwd/jsps/pwdmgt/ForgotPassword.jsp
```

5 For **Forgot Password Return Link**, specify the link shown after the user finishes performing the forgot password procedure. When the user clicks this link, the user is redirected to the link specified. For example:

```
http://localhost/IDMProv
```

6 For **Forgot Password Web Service URL**, specify the URL for the web service that the external forward password WAR uses to call back to the identity applications. Use the following format:

```
https://idmhost:sslport/idm/pwdmgt/service
```

   The return link must use SSL to ensure secure web service communication to the identity applications. For more information, see "Configuring SSL Communication between Application Servers" on page 102.

7 Manually copy `ExternalPwd.war` to the remote application server deploy directory that runs the external password WAR functionality.

Testing the External Forgot Password Configuration

If you have an external password WAR file and want to test the Forgot Password functionality by accessing it, you can access it in the following locations:

- Directly, in a browser. Go to the Forgot Password page in the external password WAR file. For example, `http://localhost:8180/ExternalPwd/jsps/pwdmgt/ForgotPassword.jsp`.
- On the User Application login page, click the link for **Forgot password**.

Configuring SSL Communication between Application Servers

If you use an external password management system, you must configure SSL communication between the Tomcat instances on which you deploy the identity applications and the External Forgotten Password Management WAR file. For more information, refer to the Tomcat documentation.

Updating SSPR Links in the Dashboard for a Distributed or Clustered Environment

The installation process assumes that you deploy SSPR on the same application server as the identity applications and Identity Reporting. By default, the built-in links on the **Applications** page in the Dashboard use a relative URL format that points to SSPR on the local system. For example, `\sspr\private\changepassword`. If you install the applications in a distributed or clustered environment, you must update the URLs for the SSPR links.

For more information, see the *Help for the Identity Applications*.

1. Log in as an administrator to the Dashboard. For example, log in as `uaadmin`.
2. Click **Edit**.
3. In the Edit Home Items page, hover on the item that you want to update, and then click the edit icon. For example, select **Change My Password**.
4. For **Link**, specify the absolute URL. For example, `http://10.10.10.48:8180/sspr/changepassword`.
5. Click **Save**.
6. Repeat for each SSPR link that you want to update.
7. Upon completion, click **I'm done**.
8. Log out, and then log in as a regular user to test the changes.

Configuring Identity Applications

- "Configuring the Settings for the Identity Applications" on page 103
- "Specifying a Location for the Permission Index" on page 128
- "Deploying REST APIs for Identity Applications" on page 128
- "Configuring the Oracle Database With Identity Applications" on page 129
- "Accessing the Oracle Database Using Oracle Service Name" on page 129
- "Manually Creating the Database Schema" on page 130
- "Configuring Single Sign-On Settings for the Identity Applications" on page 131
- "Starting the Identity Applications" on page 132
- "Configuration and Usage Considerations for the Identity Applications" on page 132

Configuring the Settings for the Identity Applications

The Identity Applications Configuration utility helps you manage the settings for the User Application drivers and the identity applications. The installation program for the identity applications invokes a version of this utility so that you can more quickly configure the applications. You can also modify most of these settings after installation.

The file to run the Configuration utility (`configupdate.sh`) is located by default in the `/opt/netiq/idm/apps/configupdate` directory:

NOTE: ◆You should run the `configudate.sh` from the `configupdate` directory only. Running the `configupdate.sh` from a custom location will result in failures.

- In a cluster, the configuration settings must be identical for all members of the cluster.

This section explains the settings in the configuration utility. The settings are organized by tabs. If you install Identity Reporting, the process adds parameters for Reporting to the utility.

Running the Identity Applications Configuration Utility

1. In `configupdate.sh.properties`, ensure that the following options are configured correctly:

   ```
   edit_admin="true"
   ```

   ```
   use_console="false"
   ```

   NOTE: You should configure the value of `-use_console` to be `true` only if you want to run the utility in console mode.

   ---

2. Save and close `configupdate.sh.properties`.
3. At the command prompt, perform the following command to run the configuration utility:

   ```
   ./configupdate.sh
   ```

   NOTE: You might need to wait a few minutes for the utility to start up.

User Application Parameters

When configuring the identity applications, this tab defines the values that the applications use when communicating with the Identity Vault. Some settings are required for completing the installation process.

By default, the tab displays the basic options. To see all settings, click **Show Advanced Options**. This tab includes the following groups of settings:

- "Identity Vault Settings" on page 104
- "Identity Vault DNs" on page 105
- "Identity Vault User Identity" on page 107
- "Identity Vault User Groups" on page 108
- "Identity Vault Certificates" on page 109

- "Email Server Configuration" on page 109
- "Trusted Key Store" on page 111
- "NetIQ Sentinel Digital Signature Certificate & Key" on page 112
- "Miscellaneous" on page 112
- "Container Object" on page 113

Identity Vault Settings

This section defines the settings that enable the identity applications to access the user identities and roles in the Identity Vault. Some settings are required for completing the installation process.

Identity Vault Server

Required

Specifies the hostname or IP address for your LDAP server. For example: `myLDAPhost`.

LDAP port

Specifies the port on which the Identity Vault listens for LDAP requests in clear text. The default value is 389.

LDAP secure port

Specifies the port on which the Identity Vault listens for LDAP requests using Secure Sockets Layer (SSL) protocol. The default value is 636.

If a service already loaded on the server (before you install eDirectory) uses the default port, you must specify a different port.

Identity Vault Administrator

Required

Specifies the credentials for the LDAP Administrator. For example, `cn=admin`. This user must already exist in the Identity Vault.

The identity applications use this account to make an administrative connection to the Identity Vault. This value is encrypted, based on the master key.

Identity Vault Administrator Password

Required

Specifies the password associated the LDAP Administrator. This password is encrypted, based on the master key.

Use Public Anonymous Account

Specifies whether users who are not logged in can access the LDAP Public Anonymous Account.

Secure Administrator Connection

Specifies whether RBPM uses SSL protocol for all communication related to the admin account. This setting allows other operations that do not require SSL to operate without SSL.

NOTE: This option might have adverse performance implications.

Secure User Connection

Specifies whether RBPM uses TLS/SSL protocol for all communication related to the logged-in user's account. This setting allows other operations that do not require TLS/SSL to operate without the protocol.

NOTE: This option might have adverse performance implications.

Identity Vault DNs

This section defines the distinguished names for containers and user accounts that enable communication between the identity applications and other Identity Manager components. Some settings are required for completing the installation process.

Root Container DN

Required

Specifies the LDAP distinguished name of the root container. This is used as the default entity definition search root when no search root is specified in the directory abstraction layer. For example, o=mycompany.

User Container DN

Required

When showing the advanced options, the utility displays this parameter under Identity Vault User Identity.

Specifies the LDAP distinguished name (DN) or fully qualified LDAP name of the user container. The following considerations apply to this setting:

- Users in this container (and below) are allowed to log in to the identity applications.
- If you have started Tomcat hosting the identity applications, you cannot change this setting with the `configupdate.sh` file.
- This container must include the User Application Administrator that you specified as you set up the User Application driver. Otherwise, the specified account cannot execute workflows.

Group Container DN

Required

When showing the advanced options, the utility displays this parameter under Identity Vault User Groups.

Specifies the LDAP distinguished name (DN) or fully qualified LDAP name of the group container. The following considerations apply to this setting:

- Entity definitions within the directory abstraction layer use this DN.
- If you have started Tomcat hosting the identity applications, you cannot change this setting with the `configupdate.sh` file.

User Application Driver

Required

Specifies the distinguished name of the User Application driver.

For example, if your driver is UserApplicationDriver and your driver set is called myDriverSet, and the driver set is in a context of o=myCompany, specify `cn=UserApplicationDriver,cn=myDriverSet,o=myCompany`.

User Application Administrator

Required

Specifies an existing user account in the Identity Vault that has the rights to perform administrative tasks for the specified user container for User Application. The following considerations apply to this setting:

- If you have started Tomcat hosting the User Application, you cannot change this setting with the `configupdate.sh` file.
- To change this assignment after you deploy the User Application, use the **Administration > Security** pages in the User Application.
- This user account has the right to use the **Administration** tab of the User Application to administer the portal.
- If the User Application Administrator participates in workflow administration tasks exposed in iManager, Designer, or the User Application (**Requests & Approvals** tab), you must grant this administrator appropriate trustee rights to object instances contained in the User Application driver. For more information, see the *User Application Administration Guide* for details.

Provisioning Administrator

Specifies an existing user account in the Identity Vault that will manage Provisioning Workflow functions available throughout the User Application.

To change this assignment after you deploy the User Application, use the **Administration > Administrator Assignments** page in the User Application.

Compliance Administrator

Specifies an existing account in the Identity Vault that performs a system role to allow members to perform all functions on the **Compliance** tab. The following considerations apply to this setting:

- To change this assignment after you deploy the identity applications, use the **Administration > Administrator Assignments** page in the User Application.
- During a configuration update, changes to this value take effect only if you do not have a valid Compliance Administrator assigned. If a valid Compliance Administrator exists, then your changes are not saved.

Roles Administrator

Specifies the role that allows members to create, remove, or modify all roles, and grant or revoke any role assignment to any user, group, or container. It also allows its role members to run any report for any user. The following considerations apply to this setting:

- By default, the User Application Admin is assigned this role.

- To change this assignment after you deploy the identity applications, use the **Administration > Administrator Assignments** page in the User Application.
- During a configuration update, changes to this value take effect only if you do not have a valid Roles Administrator assigned. If a valid Roles Administrator exists, then your changes are not saved.

Security Administrator

Specifies the role that gives members the full range of capabilities within the Security domain. The following considerations apply to this setting:

- The Security Administrator can perform all possible actions for all objects within the Security domain. The Security domain allows the Security Administrator to configure access permissions for all objects in all domains within RBPM. The Security Administrator can configure teams, and also assign domain administrators, delegated administrators, and other Security Administrators.
- To change this assignment after you deploy the identity applications, use the **Administration > Administrator Assignments** page in the User Application.

Resources Administrator

Specifies the role that gives members the full range of capabilities within the Resource domain. The following considerations apply to this setting:

- The Resources Administrator can perform all possible actions for all objects within the Resource domain.
- To change this assignment after you deploy the identity applications, use the **Administration > Administrator Assignments** page in the User Application.

RBPM Configuration Administrator

Specifies the role that gives members the full range of capabilities within the Configuration domain. The following considerations apply to this setting:

- The RBPM Configuration Administrator can perform all possible actions on all objects within the Configuration domain. The RBPM Configuration Administrator controls access to navigation items within RBPM. In addition, the RBPM Configuration Administrator configures the delegation and proxy service, the provisioning user interface, and the workflow engine.
- To change this assignment after you deploy the identity applications, use the **Administration > Administrator Assignments** page in the User Application.

RBPM Reporting Administrator

Specifies the Reporting Administrator. By default, the installation program lists this value as the same user as the other security fields.

Identity Vault User Identity

This section defines the values that enable the identity applications to communicate with a user container in the Identity Vault. Some settings are required for completing the installation process.

The utility displays these settings only when you select **Show Advanced Options**.

User Container DN

Required

When not showing the advanced options, the utility displays this parameter under Identity Vault DNs.

Specifies the LDAP distinguished name (DN) or fully qualified LDAP name of the user container. The following considerations apply to this setting:

- Users in this container (and below) are allowed to log in to the identity applications.
- If you have started Tomcat hosting the identity applications, you cannot change this setting with the `configupdate.sh` file.
- This container must include the User Application Administrator that you specified as you set up the User Application driver. Otherwise, the specified account cannot execute workflows.

User Search Scope

Specifies the depth of scope that Identity Vault users can search the container.

User Object Class

Specifies the object class of the LDAP user. Usually the class is `inetOrgPerson`.

Login Attribute

Specifies the LDAP attribute that represents the user's login name. For example, `cn`.

Naming Attribute

Specifies the LDAP attribute used as the identifier when looking up users or groups. This is not the same as the login attribute, which is used only during login. For example, `cn`.

User Membership Attribute

(Optional) Specifies the LDAP attribute that represents the user's group membership. Do not use spaces when specifying the name.

Identity Vault User Groups

This section defines the values that enable the identity applications to communicate with a group container in the Identity Vault. Some settings are required for completing the installation process.

The utility displays these settings only when you select **Show Advanced Options**.

Group Container DN

Required

When not showing the advanced options, the utility displays this parameter under Identity Vault DNs.

Specifies the LDAP distinguished name (DN) or fully qualified LDAP name of the group container. The following considerations apply to this setting:

- Entity definitions within the directory abstraction layer use this DN.
- If you have started Tomcat hosting the identity applications, you cannot change this setting with the `configupdate.sh` file.

Group Container Scope

Specifies the depth of scope that Identity Vault users can search for the group container.

Group Object Class

Specifies the object class of the LDAP group. Usually the class is `groupofNames`.

Group Membership Attribute

(Optional) Specifies the user's group membership. Do not use spaces in this name.

Use Dynamic Groups

Specifies whether you want to use dynamic groups.

You must also specify a value for **Dynamic Group Object Class**.

Dynamic Group Object Class

Applies only when you select **Use Dynamic Groups***.*

Specifies the object class of the LDAP dynamic group. Usually the class is `dynamicGroup`.

Identity Vault Certificates

This section defines the path and password for the JRE keystore. Some settings are required for completing the installation process.

Keystore Path

Required

Specifies the full path to your keystore (`cacerts`) file of the JRE that Tomcat uses to run. You can manually enter the path or browse to the `cacerts` file. The following considerations apply to this setting:

- In environments, you must specify the installation directory of RBPM. The default value is set to the correct location.
- The installation program for the identity applications modifies the keystore file. On Linux, the user must have permission to write to this file.

Keystore Password

Required

Specifies the password for the keystore file. The default is `changeit`.

Email Server Configuration

This section defines the values that enable email notifications, which you can use for email-based approvals.

Notification Template Host

Specifies the name or IP address of Tomcat that hosts the identity applications. For example, `myapplication serverServer`.

This value replaces the $HOST$ token in e-mail templates. The installation program uses this information to create a URL to provisioning request tasks and approval notifications.

Notification Template Port

Specifies the port number of Tomcat that hosts the identity applications.

This values replaces the $PORT$ token in e-mail templates that are used in provisioning request tasks and approval notifications.

Notification Template Secure Port

Specifies the secure port number of Tomcat that hosts the identity applications.

This value replaces the $SECURE_PORT$ token in e-mail templates used in provisioning request tasks and approval notifications.

Notification Template Protocol

Specifies a non-secure protocol included in the URL when sending user email. For example, `http`.

This value replaces the $PROTOCOL$ token in e-mail templates used in provisioning request tasks and approval notifications.

Notification Template Secure Protocol

Specifies the secure protocol included in the URL when sending user email. For example, `https`.

This value replaces the $SECURE_PROTOCOL$ token in e-mail templates used in provisioning request tasks and approval notifications.

Notification SMTP Email From

Specifies the email account that the identity applications use to send email notifications.

SMTP Server Name

Specifies the IP address or DNS name of the SMTP email host that the identity applications use for provisioning emails. Do not use `localhost`.

Server requires authentication

Specifies whether you want the server to require authentication.

You must also specify the credentials for the email server.

User name

Applies only when you enable **Server requires authentication***.*

Specifies the name of a login account for the email server.

Password

Applies only when you enable **Server requires authentication***.*

Specifies the password of an login account for the mail server.

Use SMTP TLS

Specifies whether you want to secure the contents of email messages during transmission between the mail servers.

Email Notification Image Location

Specifies the path to the image that you want to include in email notifications.

When the Identity Applications server and the email server are both set to use secure connection, make sure that the following conditions are met:

- The certificate used to establish a secure connection between the Identity Applications server and the email server is a trusted CA certificate
- Use `https` in the image path. For example, `https://localhost:8543/IDMProv/images`

If Identity Applications is operating on a server that uses http for plain text communication, replace `https` with `http` in the image path. An example of the image path: `http://localhost:8080/IDMProv/images`

Sign email

Specifies whether you want to add a digital signature to outgoing messages.

If you enable this option, you must also specify settings for the keystore and signature key.

Keystore Path

Applies only when you enable **Sign email**.

Specifies the full path to the keystore (`cacerts`) file that you want to use for digitally signing an email. You can manually enter the path or browse to the `cacerts` file.

For example, `/opt/netiq/idm/apps/jre/lib/security/cacerts`.

Keystore Password

Applies only when you enable **Sign email**.

Specifies the password for the keystore file. For example, `changeit`.

Alias of signature key

Applies only when you enable **Sign email**.

Specifies the alias of the signing key in the keystore. For example, `idmapptest`.

Signature key password

Applies only when you enable **Sign email**.

Specifies the password that protects the file containing the signature key. For example, `changeit`.

Trusted Key Store

This section defines the values for the trusted keystore for the identity applications. The utility displays these settings only when you select **Show Advanced Options**.

Trusted Store Path

Specifies the path to the Trusted Key Store that contains all trusted signers' certificates. If this path is empty, the identity applications get the path from System property `javax.net.ssl.trustStore`. If the System property cannot provide the path, the installation program defaults to `jre/lib/security/cacerts`.

Trusted Store Password

Specifies the password for the Trusted Key Store. If you leave this field is empty, the identity applications gets the password from System property `javax.net.ssl.trustStorePassword`. If the System property cannot provide the path, the installation program defaults to `changeit`.

This password is encrypted, based on the master key.

Trusted Store Type

Specifies whether the trusted store path uses a Java keystore (JKS) or PKCS12 for digital signing.

NetIQ Sentinel Digital Signature Certificate & Key

This section defines the values that allows Identity Manager to communicate with Sentinel for auditing events. The utility displays these settings only when you select **Show Advanced Options**.

Sentinel Digital Signature Certificate

Lists the custom public key certificate that you want the OAuth server to use to authenticate audit messages sent to Sentinel.

Sentinel Digital Signature Private Key

Specifies the path to the custom private key file that you want the OAuth server to use to authenticate audit messages sent to Sentinel.

Miscellaneous

The utility displays these settings only when you select **Show Advanced Options**.

OCSP URI

Specifies the Uniform Resource Identifier (URI) to use when the client installation uses the On-Line Certificate Status Protocol (OCSP). For example, `http://host:port/ocspLocal`.

The OCSP URI updates the status of trusted certificates online.

Authorization Config Path

Specifies the fully qualified name of the authorization configuration file.

Identity Vault Indexes

To improve the performance of the identity applications, you can create value indexes for manager, ismanager, and srvprvUUID attributes.

You can create value indexes by using the Configuration utility or iManager after completing the identity applications installation. The following considerations apply to this setting:

- Without indexes on these attributes, identity applications users can experience impeded performance of the identity applications.
- You can create these indexes manually by using iManager after you install the identity applications.
- For best performance, you should create the index during installation.
- The indexes must be in Online mode before you make the identity applications available to users.

- To create an index, select **Create** in the **Server DN** setting and specify a value for **Server DN**. Click **OK** and then restart the Identity Vault for the changes to take effect.
- To delete an index, select **Delete** in the **Server DN** setting and specify a value for **Server DN**. Click **OK** and then restart the Identity Vault for the changes to take effect.

Server DN

Applies only when you want to create or delete an Identity Vault index.

Specifies the eDirectory server where you want the indexes to be created or removed.

You can specify only one server at a time. To configure indexes on multiple eDirectory servers, you must run the RBPM Configuration utility multiple times.

Reinitialize RBPM Security

Specifies whether you want to reset RBPM security when the installation process completes. You must also redeploy the identity applications.

IDMReport URL

Specifies the URL of the Identity Manager Reporting Module. For example, `http://hostname:port/IDMRPT`.

Custom Themes Context Name

Specifies the name of the customized theme that you want to use for displaying the identity applications in the browser.

Log Message Identifier Prefix

Specifies the value that you want to use in the layout pattern for the CONSOLE and FILE appenders in the `idmuserapp_logging.xml` file. The default value is RBPM.

Change RBPM Context Name

Specifies whether you want to change the context name for RBPM.

You must also specify the new name and DN of the Roles and Resource driver.

RBPM Context Name

Applies only when you select **Change RBPM Context Name**.

Specifies the new context name for RBPM.

Role Driver DN

Applies only when you select **Change RBPM Context Name**.

Specifies the DN of the Roles and Resource driver.

Container Object

These parameters apply only during installation.

This section helps you to define the values for container objects or create new container objects.

Selected

Specifies the Container Object Types that you want to use.

Container Object Type

Specifies the container: locality, country, organizationalUnit, organization, or domain.

You can also define your own containers in iManager and add them under **Add a new Container Object**.

Container Attribute Name

Specifies the name of the Attribute Type associated with the specified Container Object Type.

Add a New Container Object: Container Object Type

Specifies the LDAP name of an object class from the Identity Vault that can serve as a new container.

Add a New Container Object: Container Attribute Name

Specifies the name of the Attribute Type associated with the new Container Object Type.

Reporting Parameters

When configuring the identity applications, this tab defines the values for managing Identity Reporting. The utility adds this tab when you install Identity Reporting.

By default, the tab displays the basic options. To see all settings, click **Show Advanced Options**. This tab includes the following groups of settings:

- "Email Delivery Configuration" on page 114
- "Report Retention Values" on page 115
- "Modify Locale" on page 115
- "Role Configuration" on page 115
- "Outbound Proxy" on page 115

Email Delivery Configuration

This section defines the values for sending notifications.

SMTP Server Hostname

Specifies the DNS name or IP address of the email server than you want Identity Reporting to use when sending notification. Do not use `localhost`.

SMTP Server Port

Specifies the port number for the SMTP server.

SMTP Use SSL

Specifies whether you want to use TLS/SSL protocol for communication with the email server.

Server Needs Authentication

Specifies whether you want to use authentication for communications with the email server.

SMTP User Name

Specifies the email address that you want to use for authentication.

You must specify a value. If the server does not require authentication, you can specify an invalid address.

SMTP User Password

Applies only when you specify that the server requires authentication.

Specifies the password for the SMTP user account.

Default Email Address

Specifies the email address that you want Identity Reporting to use as the origination for email notifications.

Report Retention Values

This section defines the values for storing completed reports.

Report Unit, Report Lifetime

Specifies the amount of time that Identity Reporting keeps completed reports before deleting them. For example, to specify six months, enter 6 in the **Report Lifetime** field and then select **Month** in the **Report Unit** field.

Location of Reports

Specifies a path where you want to store the report definitions. For example, `/opt/netiq/IdentityReporting`.

Modify Locale

This section defines the values for the language that you want Identity Reporting to use. Identity Reporting uses the specific locales in searches. For more information, see the Administrator Guide to NetIQ Identity Reporting.

Role Configuration

This section defines the values for the authentication sources that Identity Reporting uses to generate reports.

Add Authentication Source

Specifies the type of authentication source that you want to add for reporting. Authentication sources can be

- **Default**
- **LDAP Directory**
- **File**

Outbound Proxy

Applies only when you use Identity Manager 4.8.1 or later versions.

This section defines the values to use reverse proxy server that Identity Reporting uses to download reports.

Use Proxy

Specifies the option to use Reverse Proxy server for reporting.

- **Hostname or IP address**
- **Port**

- **Use TLS**

  Applies only when you want to use TCP as your network protocol.

Authentication Parameters

When configuring the identity applications, this tab defines the values that Tomcat uses to direct users to the identity application and password management pages.

By default, the tab displays the basic options. To see all settings, click **Show Advanced Options**. This tab includes the following groups of settings:

- "Authentication Server" on page 116
- "Authentication Configuration" on page 117
- "Authentication Method" on page 117
- "Password Management" on page 119
- "Sentinel Digital Signature Certificate and Key" on page 120

Authentication Server

This section defines settings for the identity applications to connect to the authentication server.

OAuth server host identifier

Required

Specifies the relative URL of the authentication server that issues tokens to OSP. For example, 192.168.0.1.

OAuth server TCP port

Specifies the port for the authentication server.

OAuth server is using TLS/SSL

Specifies whether the authentication server uses TLS/SSL protocol for communication.

Optional TLS/SSL truststore file

Applies only when you select **OAuth server is using TLS/SSL** *and the utility is showing the advanced options.*

Optional TLS/SSL truststore password

Applies only when you select **OAuth server is using TLS/SSL** *and the utility is showing the advanced options.*

Specifies the password used to load the keystore file for the TLS/SSL authentication server.

NOTE: If you do not specify the keystore path and password, and the trust certificate for the authentication server is not in the JRE trust store (cacerts), the identity applications fail to connect to the authentication service that uses TLS/SSL protocol.

Authentication Configuration

This section defines settings for the authentication server.

LDAP DN of Admins Container

Required

Specifies the distinguished name of the container in the Identity Vault that contains any administrator User objects that OSP must authenticate. For example, `ou=sa,o=data`.

Duplicate resolution naming attribute

Specifies the name of the LDAP attribute used to differentiate between multiple eDirectory User objects with the same `cn` value. The default value is `mail`.

Restrict authentication sources to contexts

Specifies whether searches in the user and administrator containers in the Identity Vault are restricted to only User objects in those containers or searches should also include subcontainers.

Session Timeout (minutes)

Specifies the number of minutes of inactivity in a session before the server times out the user's session. The default value is 20 minutes.

Access token lifetime (seconds)

Specifies the number of seconds an OSP access token remains valid. The default value is 60 seconds.

Refresh token lifetime (hours)

Specifies the number of seconds an OSP refresh token remains valid. The refresh token is used internally by OSP. The default value is 48 hours.

Authentication Method

This section defines the values that enable OSP to authenticate users who log in to the browser-based components of Identity Manager.

Method

Specifies the type of authentication that you want Identity Manager to use when a user logs on.

- **Name and Password**: OSP verifies authentication with the Identity Vault.
- **Kerberos**: OSP accepts authentication from both a Kerberos ticket server and the identity vault.
- **SAML 2.0**: OSP accepts authentication from both a SAML identity provider and the identity vault.

Enable reCAPTCHA

Applies only when you specify **Name and Password**.

Specifies whether you want to enable reCAPTCHA on the login page.

reCAPTCHA provides an additional layer of security by requesting users to confirm that they are not a robot. It displays images that users must select based on a matching criteria. If a response succeeds, Access Manager authenticates the user's authentication credentials. If a response fails, Access Manager does not authenticate the user credentials, and redirects to the login page.

Enable two-factor authentication

Applies only when you specify **Name and Password***.*

Specifies whether you want to enable two-factor authentication.

This requires some configuration to be done in the **Second Factor** tab. For more information, see "Second Factor Parameters" on page 125.

Mapping attribute name

Applies only when you specify **Kerberos** *or* **SAML***.*

Specifies the name of the attribute that maps to the Kerberos ticket server or SAML representations at the identity provider.

Enable fallback reCAPTCHA

Applies only when you specify **Kerberos***.*

Specifies whether you want to enable reCAPTCHA with the fallback username and password when Kerberos cannot be used.

Number of attempts before required

Applies only when you select the **Enable fallback reCAPTCHA** *check box.*

Specifies the number of unsuccessful login attempts before reCAPTCHA is enabled. Setting the value to zero indicates that the reCAPTCHA is always required.

Site Key

Applies only when you select the **Enable fallback reCAPTCHA** *check box.*

Specifies the reCAPTCHA site key value obtained from the Google reCAPTCHA website.

Private Key

Applies only when you select the **Enable fallback reCAPTCHA** *check box.*

Specifies the reCAPTCHA private key value obtained from the Google reCAPTCHA website.

Enable fallback two-factor authentication

Applies only when you specify **Kerberos***.*

Specifies whether you want to enable two-factor authentication with the fallback username and password when Kerberos cannot be used.

This requires some configuration to be done in the **Second Factor** tab. For more information, see "Second Factor Parameters" on page 125.

Use logout landing page

Applies only when you specify **Kerberos***.*

Specifies if you want to enable a landing page rather than redirecting to the login page after a successful logout.

Landing Page

Applies only when you specify **SAML**.

- **None:** Specifies that the landing page will not be used. Select this option if the IDP URL is indicated.
- **Internal:** Specifies that the internal OSP landing page will be used.
- **External:** Specifies that you will be redirected to an external OSP landing page.

URL

Applies only when you select **External** *in the* **Landing page** *field.*

Specifies the URL of the external landing page.

Metadata source

Applies only when you specify **SAML**.

Specifies the source of the IDP metadata. You can either load the metadata from a URL or copy a previously obtained metadata.

Metadata URL

Applies only when you specify **URL** *in the* **Metadata URL** field.

Specifies whether you want to load the metadata from the URL and save it to the configuration before you exit the application.

Load on save

Applies only when you specify **URL** *in the* **Metadata URL** field.

Specifies the URL that OSP uses to redirect the authentication request to SAML.

IDP Metadata

Applies only when you specify **Copy/Paste** *in the* **Metadata URL** field.

Specifies the data you want to paste, that is obtained from the SAML IDP.

Configure Access Manager on exit

Applies only when you specify **Copy/Paste** *in the* **Metadata URL** field.

Specifies whether you want to automatically configure a SAML service provider definition in Access Manager.

Password Management

This section defines the values that enable users to modify their passwords as a self-service operation.

Password Management Provider

Specifies the type of password management system that you want to use.

User Application (Legacy): Uses the password management program that Identity Manager traditionally has used. This option also allows you to use an external password management program.

Forgotten Password

This check box parameter applies only when you want to use SSPR.

Specifies whether you want users to recover a forgotten password without contacting a help desk.

You must also configure the challenge-response policies for the Forgotten Password feature. For more information, see the *NetIQ Self Service Password Reset Administration Guide*.

Forgotten Password

This menu list applies only when you select **User Application (Legacy)**.

Specifies whether you want to use the password management system integrated with the User Application or an external system.

- **Internal**: Use the default internal Password Management functionality, `./jsps/pwdmgt/ForgotPassword.jsp` (without the http(s) protocol at the beginning). This redirects the user to the Forgot Password functionality built into the User Application, rather than to an external WAR.
- **External**: Use an external Forgot Password WAR to call back the User Application through a web service. You must also specify the settings for the external system.

Forgotten Password Link

Applies only when you want to use an external password management system.

Specifies the URL that points to the Forgot Password functionality page. Specify a `ForgotPassword.jsp` file in an external or internal password management WAR.

Forgotten Password Return Link

Applies only when you want to use an external password management system.

Specifies the URL for the **Forgot Password Return Link** that the user can click after performing a forgot password operation.

Forgotten Password Web Service URL

Applies only when you want to use an external password management system.

Specifies the URL that the External Forgot Password WAR will use to call back to the User Application to perform core forgot password functionalities. Use the following format:

```
https://<idmhost>:<sslport>/<idm>/
pwdmgt/service
```

Sentinel Digital Signature Certificate and Key

This section defines the values that allows Identity Manager to communicate with Sentinel for auditing events.

Sentinel Digital Signature Certificate

Specifies a custom public key certificate that you want the OSP server to use to authenticate audit messages sent to the audit system.

For information about configuring certificates for Novell Audit, see "Managing Certificates" in the *Novell Audit Administration Guide*.

Sentinel Digital Signature Private Key

Specifies the path to the custom private key file that you want the OSP server to use to authenticate audit messages sent to the audit system.

SSO Clients Parameters

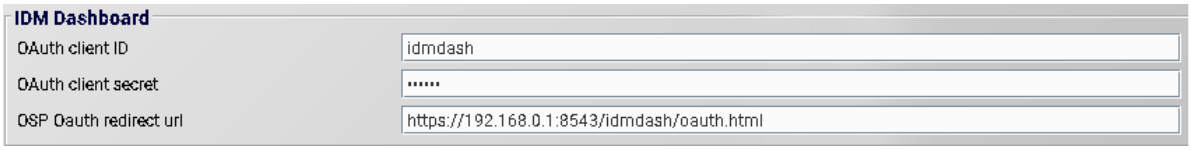
When configuring the identity applications, this tab defines the values for managing single sign-on access to the applications.

By default, the tab displays the basic options. To see all settings, click **Show Advanced Options**. This tab includes the following groups of settings:

- "IDM Dashboard" on page 121
- "IDM Administrator" on page 122
- "RBPM" on page 122
- "Reporting" on page 123
- "IDM Data Collection Service" on page 124
- "DCS Driver" on page 124
- "Self Service Password Reset" on page 124

IDM Dashboard

This section defines the values for the URL that users need to access the Identity Manager Dashboard, which is the primary login location for the identity applications.



OAuth client ID

Required

Specifies the name that you want to use to identify the single sign-on client for the Dashboard to the authentication server. The default value is `idmdash`.

OAuth client secret

Required

Specifies the password for the single sign-on client for the Dashboard.

OSP OAuth redirect URL

Required

Specifies the absolute URL to which the authentication server redirects a browser client when authentication is complete.

Use the following format: `protocol://server:port/path`. For example, `https://192.168.0.1:8543/idmdash/oauth.html`.

IDM Administrator

This section defines the values for the URL that users need to access the Identity Manager Administrator page.

OAuth client ID

Required

Specifies the name that you want to use to identify the single sign-on client for the Identity Manager Administrator to the authentication server. The default value is `idmadmin`.

OAuth client secret

Required

Specifies the password for the single sign-on client for the Identity Manager Administrator.

OSP OAuth redirect URL

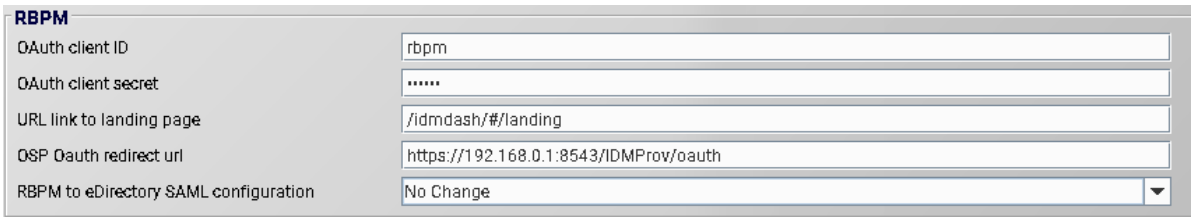
Required

Specifies the absolute URL to which the authentication server redirects a browser client when authentication is complete.

Use the following format: `protocol://server:port/path`. For example, `https://192.168.0.1:8543/idmadmin/oauth.html`.

RBPM

This section defines the values for the URL that users need to access the User Application.



OAuth client ID

Required

Specifies the name that you want to use to identify the single sign-on client for the User Application to the authentication server. The default value is `rbpm`.

OAuth client secret

Required

Specifies the password for the single sign-on client for the User Application.

URL link to landing page

Required

Specifies the relative URL to use to access the Dashboard from the User Application. The default value is `/landing`.

OSP OAuth redirect URL

Required

Specifies the absolute URL to which the authentication server redirects a browser client when authentication is complete.

Use the following format: `protocol://server:port/path`. For example, `https://192.168.0.1:8543/IDMProv/oauth`.

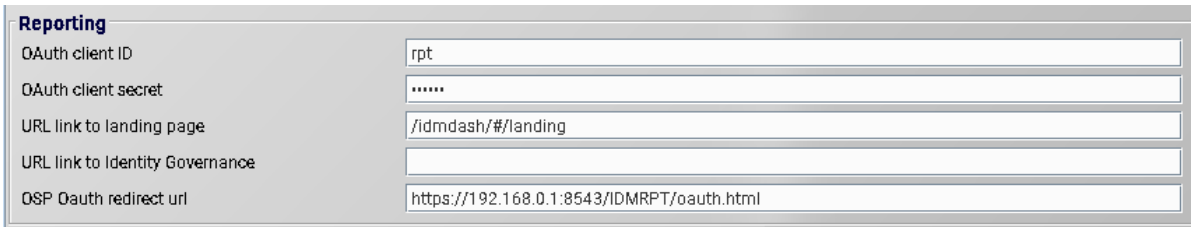
RBPM to eDirectory SAML configuration

Required

Specifies the RBPM to Identity Vault SAML settings required for SSO authentication.

Reporting

This section defines the values for the URL that users need to access Identity Reporting. The utility display these values only if you add Identity Reporting to your Identity Manager solution.



OAuth client ID

Required

Specifies the name that you want to use to identify the single sign-on client for the Identity Reporting to the authentication server. The default value is `rpt`.

OAuth client secret

Required

Specifies the password for the single sign-on client for Identity Reporting.

URL link to landing page

Required

Specifies the relative URL to use to access the Dashboard from Identity Reporting. The default value is `/idmdash/#/landing`.

If you installed Identity Reporting and the identity applications in separate servers, then specify an absolute URL. Use the following format: `protocol://server:port/path`. For example, `https://192.168.0.1:8543/IDMRPT/oauth`.

OSP OAuth redirect url

Required

Specifies the absolute URL to which the authentication server redirects a browser client when authentication is complete.

Use the following format: `protocol://server:port/path`. For example, `https://192.168.0.1:8543/IDMRPT/oauth`.

IDM Data Collection Service

This section defines the values for the URL that users need to access the Identity Manager Data Collection Service.

OAuth client ID

Required

Specifies the name that you want to use to identify the single sign-on client for Identity Manager Data Collection Service to the authentication server. The default value is `idmdcs`.

OAuth client secret

Required

Specifies the password for the single sign-on client for the Identity Manager Data Collection Service.

OSP OAuth redirect URL

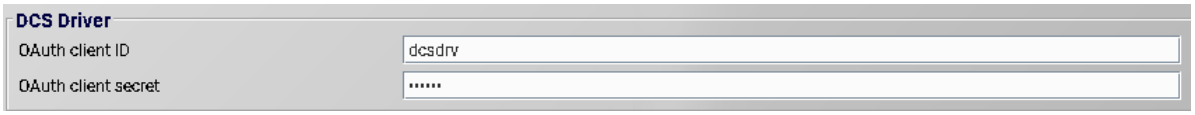
Required

Specifies the absolute URL to which the authentication server redirects a browser client when authentication is complete.

Use the following format: `protocol://server:port/path`. For example, `https://192.168.0.1:8543/idmdcs/oauth.html`.

DCS Driver

This section defines the values for managing the Data Collection Services driver.



OAuth client ID

Specifies the name that you want to use to identify the single sign-on client for the Data Collection Service driver to the authentication server. The default value for this parameter is `dcsdrv`.

OAuth client secret

Specifies the password for the single sign-on client for the Data Collection Service driver.

Self Service Password Reset

This section defines the values for the URL that users need to access SSPR.

OAuth client ID

Required

Specifies the name that you want to use to identify the single sign-on client for SSPR to the authentication server. The default value is `sspr`.

OAuth client secret

Required

Specifies the password for the single sign-on client for SSPR.

OSP OAuth redirect URL

Required

Specifies the absolute URL to which the authentication server redirects a browser client when authentication is complete.

Use the following format: `protocol://server:port/path`. For example, `https://192.168.0.1:8543/sspr/public/oauth.html`.

Second Factor Parameters

Ensure that you have created the methods, chain, and events in Advanced Authentication before proceeding. You must configure OSP to accept the authentications from Advanced Authentication.

By default, the tab displays the basic options. To see all settings, click Show Advanced Options. This tab includes the following groups of settings:

AAF Administrator

This section defines settings for the Advanced Authentication Administrator:

Admin name (Repository\name)

Required

Specifies the repository-qualified name of the Advanced Authentication administrator account that OSP uses to interface with Advanced Authentication. Typically, the account is in the LOCAL repository.

The default Advanced Authentication administrator account is named admin. If you used this account, then the **Admin name** value is:

LOCAL\admin (repository name + \ + user name)

Admin Password

Required

Specifies the password for the Advanced Authentication administrative user you specified above.

AAF User Repository

This section define settings for the Advanced Authentication user repository:

User repository name

Required

Specifies the name of the repository in Advanced Authentication you created. This repository corresponds to the Identity Vault for Identity Manager.

AAF Servers

This section defines settings for the Advanced Authentication servers:

Allow test TLS certificate

Required

Specifies whether you want to ignore an invalid test certificate subject from the AAF server. This applies only for initial configuration and testing.

Click **Add**, then specify the DNS name or IP address of the Advanced Authentication server. If you use a different port than 443, specify that port as well.

(Conditional) If you have clustered the Advanced Authentication server, then click **Add** again, and specify each DNS name or IP address for each server in the cluster.

Show tuning parameters

Required

Specifies whether you want to enable the tuning parameters.

Logout session cleanup (minutes): *Applies only if you have selected the* **Show tuning parameters** *check box.*

Specifies the duration after which the active AAF logon sessions are considered for timeout and cleanup issues.

Heartbeat interval (milliseconds): *Applies only if you have selected the* **Show tuning parameters** *check box.*

Specifies the duration after which the heartbeat request is sent to an AAF server to check for availability.

AAF Endpoint

This section define settings for the Advanced Authentication endpoints:

Create new endpoint

Required

Specifies whether you want to create a new endpoint for two-factor authentication.

Identifier: *Applies only if you have not selected the* **Create new endpoint** *check box.*

Specifies the endpoint identifier as configured in AAF administration.

Secret: *Applies only if you have not selected the* **Create new endpoint** *check box.*

Specifies the endpoint secret as configured in AAF administration.

Name: *Applies only if you have selected the* **Create new endpoint** *check box.*

Specifies the name of the new endpoint used for identifying the endpoint in the AAF administration pages.

Description: *Applies only if you have selected the* **Create new endpoint** *check box.*

Specifies the description for the new endpoint that you specified above.

Second Factor Conditions

This section defines settings for the second factor conditions.

All users, all the time

Required

Specifies whether you want to enable all users to provide a second factor authentication at all times.

User Login Condition: *Applies only if you have not selected the* **All users, all the time** *check box*.

Specifies that you can define certain expressions and conditions for Identity Manager to use the second factor authentication.

Second Factor Authentication Methods

This section define settings for the Advanced Authentication methods.

Specifies whether you want to enable the second factor authentication for different methods.

To disable the second factor authentication for a method, deselect the check box next to the method name.

Identity Manager uses the relative priority of second factor methods if a user has enrolled in more than one method.

CEF Auditing Parameters

This section defines the values for managing the CEF auditing parameters for the single sign-on client.

Send audit events

Specifies whether you want to use CEF for auditing events.

Destination host

Specifies the DNS name or the IP address of the auditing server.

Destination port

Specifies the port of the auditing server.

Network protocol

Specifies the network protocol used by the auditing server to receive CEF events.

Use TLS

Applies only when you want to use TCP as your network protocol.

Specifies if the auditing server is configured to use TLS with TCP. Select **Use TLS** > Show Advanced Options, and provide the **Identity Manager Keystore file name** and the **Identity Manager Keystore password**.

Intermediate event store directory

Specifies the location of the cache directory before the CEF events are sent to the auditing server. If you are providing an intermediate event store directory of your choice, you must first ensure that the permissions and ownership are changed to `novlua` for that directory. To change the permission of the directory, run the following commands:

```
chown novlua:novlua <directory_path>
chmod 755 <directory_path>
```

where `<directory_path>` is the path to the intermediate event store directory.

For complete documentation on configuring CEF auditing, see Configuring Identity Applications in *NetIQ Identity Manager - Configuring Auditing in Identity Manager*.

Specifying a Location for the Permission Index

When you install the identity applications, the process creates a permission index for Tomcat. If you do not specify a location for the index, the installation creates a folder in a temporary directory. For example, `/opt/netiq/idm/apps/tomcat/temp/permindex` on Tomcat.

In a test environment, the location usually does not matter. However, in a production or staging environment, you might not want to place the permission index in a temporary directory.

To specify a location for the index:

1. Stop Tomcat.
2. In a text editor, open the `ism-configuration.properties` file.
3. At the end of the file, add the following text:

   `com.netiq.idm.cis.indexdir = path/permindex`

   For example:

   `com.netiq.idm.cis.indexdir = /opt/netiq/idm/apps/tomcat/temp/permindex`
4. Save and close the file.
5. Delete the existing `permindex` folder in the temporary directory.
6. Start Tomcat.

To enable the permission index for clustering, see Chapter 18, "Sample Identity Applications Cluster Deployment Solution on Tomcat Application Server," on page 285.

Deploying REST APIs for Identity Applications

The identity applications components incorporate several REST APIs that enable different features within Identity Applications. The REST services use OAUTH2 protocol to provide authentication. You can invoke these APIs using a browser or curl command in scripts to automate the administrative tasks. The REST APIs and the corresponding documentation are available in the `idmappsdoc.war` file. The war is automatically deployed when Identity Applications are installed. For more information, see the REST API documentation.

To access the REST API documentation on the server where identity applications are installed, specify `https://<identity applications servername>:<Port>/idmappsdoc`, in the address bar of your browser. For example: `https://192.168.0.1:8543/idmappsdoc`.

NOTE: To export the workflow database tables, add the below entry in the `ism-configuration.properties` file:

```
com.microfocus.workflow.migration.tables = <list-of-tables-to-exported-comma-separated>
```

For example:

```
com.microfocus.workflow.migration.tables = afmodel,afform,
afprocess,afdocument,afactivity,afactivitytimertasks,afbranch,afcomment,af
provisioningstatus,afquorum,afworktask,configuration,email_approval_token,
localization,processed_eba_mails
```

Configuring the Oracle Database With Identity Applications

Perform the following steps to configure the Oracle database:

1. Download and install the latest version of the Oracle database.

   The default SID for the database is `orcl`.

2. Create the database instance:

   2a. From the command line, run the following command to launch the Database Configuration Assistor.

   ```
   dbca
   ```

   2b. In the **Global Database Name** field, specify the database name. For example, `idappsdb`.

3. Create the database user:

   3a. Log in to the Oracle SQL developer tool.

   3b. Run the following commands:

   ```
   CREATE USER idmadmin IDENTIFIED BY <password>;
   GRANT CONNECT, RESOURCE to idmadmin;
   ALTER USER idmadmin quota 100M on USERS;
   ```

   where,

   `idmadmin` represents the user account, and

   `<password>` represents the user password.

4. Repeat steps 2 and 3 to configure the Workflow Engine database.

NOTE: When you are configuring Identity Applications with Oracle database, ensure that you use the same details as specified in the above section.

Accessing the Oracle Database Using Oracle Service Name

You can connect to the Oracle database by using Oracle System ID (SID) and Oracle Service Name. If you want to access the database by using a service name, complete the identity applications installation to one database instance by connecting through SID. After the installation is completed, perform the following actions:

1. Create a service name in the Oracle database by running the following command:

   ```
   alter system set service_names='SERVICE1' scope=both sid='*';
   ```

   where `SERVICE 1` is the name of the Oracle service.

   NOTE: You can specify the service name in uppercase or lowercase. It is not case-sensitive.

   ---

2. Define the service name in Tomcat's `server.xml` file by modifying the Oracle data source details in the file:

   ```
   url="jdbc:oracle:thin:@IP:PORT/service1"
   ```

3. Restart Tomcat.

4 Verify that the service name is included in the `catalina.out` log file.

5 Verify that the identity applications are properly connected to the database.

Manually Creating the Database Schema

When you install the identity applications, you can postpone connecting to the database or creating tables in the database. If you do not have permissions to the database, you might need to choose this option. The installation program creates a SQL file that you can use to create the database schema. You can also recreate the database tables after installation without having to reinstall. To do so, you delete the database for the identity applications and create a new database with the same name.

Using the SQL File to Generate the Database Schema

This section assumes that the installation program created a SQL file that you can execute to generate the database schema. If you do not have the SQL file, see "Manually Creating the SQL File to Generate the Database Schema" on page 131.

NOTE: Do not use SQL*Plus to execute the SQL file. The line lengths in the file exceed 4000 characters.

---

1 Stop the Application Server.

2 Login to the Database Server.

3 Delete the database that is used by the identity applications.

4 Create a new database with the same name as the one that was deleted in Step 3.

5 Navigate to the SQL script that the installation process created, by default in the `/installation_path/userapp/sql` directory.

6 (Conditional) For an Oracle database, insert a backslash (/) after the definition of the function `CONCAT_BLOB`. For example:

```
  -- Changeset icfg-data-load.xml::700::IDMRBPM
CREATE OR REPLACE FUNCTION CONCAT_BLOB(A IN BLOB, B IN BLOB) RETURN BLOB
AS
                C BLOB;
            BEGIN
                DBMS_LOB.CREATETEMPORARY(C, TRUE);
                DBMS_LOB.APPEND(C, A);
                DBMS_LOB.APPEND(C, B);
                RETURN c;
            END;
/
```

7 Have the database administrator run the SQL script to create and configure the User Application database.

8 Restart Tomcat.

Manually Creating the SQL File to Generate the Database Schema

You can recreate the database tables after installation without having to reinstall and without having the SQL file. This section helps you create the database schema in the event that you do not have the SQL file.

1. Stop Tomcat.
2. Log in to the server that hosts your identity applications database.
3. Delete the existing database.
4. Create a new database with the same name as the one that you deleted in Step 3.
5. In a text editor, open the `NetIQ-Custom-Install.log` file, located by default at the root of the installation directory for the identity applications. For example:

   ```
   /opt/netiq/idm/apps/UserApplication
   ```

6. Search and copy the below command from the `NetIQ-Custom-Install.log` file:

   ```
   /opt/netiq/common/jre/bin/java -Xms256m -Xmx256m  -
   Dwar.context.name=IDMProv -Ddriver.dn="cn=User Application
   Driver,cn=driverset1,o=system" -Duser.container="o=data" -jar /opt/
   netiq/idm/apps/UserApplication/liquibase.jar --
   databaseClass=liquibase.database.core.PostgresDatabase --
   driver=org.postgresql.Driver --classpath=/opt/netiq/idm/apps/
   postgresql/postgresql-9.4.1212jdbc42.jar opt/netiq/idm/apps/
   UserApplication/IDMProv.war --changeLogFile=DatabaseChangeLog.xml  --
   url="jdbc:postgresql://localhost:5432/idmuserappdb" --
   contexts="prov,newdb" --logLevel=info --logFile=/opt/netiq/idm/apps/
   UserApplication/db.out --username=******** --password=******** update
   ```

7. Log in to the server where you installed the database for the identity applications.
8. In a terminal, paste the command string that you copied.

   NOTE: The command should be `updateSQL`. If it is `update`, change the command to `updateSQL`.

9. In the command, replace the asterisks (*) that represent the database username and password with the actual values required to authenticate. Also, ensure the name of the SQL file is unique.
10. Execute the command.
11. (Conditional) If the process generates a SQL file instead of populating the database, provide the file to your database administrator to import into the database server. For more information, see "Using the SQL File to Generate the Database Schema" on page 130.
12. After the database administrator imports the SQL file, start Tomcat.

Configuring Single Sign-On Settings for the Identity Applications

The installation process installs an authentication service (OSP) for single sign-on access in Identity Manager. However, you can also configure the OSP authentication server to accept authentication from the Kerberos ticket server or SAML. To configure the single sign-on settings for the identity applications after installation, Configuring Single Sign-on Access in Identity Manager in the *NetIQ Identity Manager - Administrator's Guide to the Identity Applications*.

Starting the Identity Applications

Ensure that you restart the Tomcat service and ActiveMQ service after you configure the identity applications.

```
systemctl restart netiq-tomcat.service
```

```
systemctl restart netiq-activemq.service
```

Configuration and Usage Considerations for the Identity Applications

The following considerations apply to the configurations and initial usage of the identity applications.

- During the installation process, the installation program writes log files to the installation directory. These files contain information about your configuration. After you configure your Identity Applications environment, you should consider deleting these log files or storing them in a secure location. During the installation process, you might choose to write the database schema to a file. Since this file contains descriptive information about your database, you should move the file to a secure location after the installation process is complete.
- (Conditional) To audit the identity applications, you must have Identity Reporting and an auditing service installed in your environment and configured to capture the events. You must also configure the identity applications for auditing.
- Before users can access the identity applications, you must complete the following activities:
  - Ensure that all necessary Identity Manager drivers are installed.
  - Ensure that the indexes for the Identity Vault are in Online mode. For more information about configuring an index during or after installation, see "Creating Value Indexes for Identity Vault" on page 93.
  - Enable cookies on all browsers. The applications do not work when cookies are disabled.
- If you have installed Identity Applications and SSPR on different servers, then you must import the SSPR trusted certificate into the `idm.jks` of Identity Applications server.

Configuring the Runtime Environment for Data Collection

This section provides information about additional configuration steps you should perform to ensure that the runtime environment is operating correctly. It also provides troubleshooting techniques, as well as some information about database tables that are of particular interest.

This process includes the following activities:

- "Configuring the Data Collection Services Driver to Collect Data from the Identity Applications" on page 133
- "Migrating the Data Collection Service Driver" on page 134
- "Adding Support for Custom Attributes and Objects" on page 136
- "Adding Support for Multiple Driver Sets" on page 138
- "Configuring the Drivers to Run in Remote Mode with SSL" on page 140

If you have problems with one or more of the drivers that are difficult to understand, see "Troubleshooting the Drivers" in the *NetIQ Identity Reporting Module Guide*.

Configuring the Data Collection Services Driver to Collect Data from the Identity Applications

For the identity applications to function properly with Identity Reporting, you must configure the DCS driver to support the OAuth protocol.

NOTE: ◆You only need to install and configure the DCS driver if you use Identity Reporting in your environment.

◆ If you have multiple DCS drivers configured in your environment, you must complete the following steps for each driver.

---

1. Log in to Designer.
2. Open your project in Designer.
3. (Conditional) If you have not already upgraded your DCS driver to the supported patch version, complete the following steps:
   - **3a** Download the latest DCS driver patch file.
   - **3b** Extract the patch file to a location on your server.
   - **3c** In a terminal, navigate to the location of the extracted patch RPM for your environment and run the following command:

     ```
     rpm -Uvh novell-DXMLdcs.rpm
     ```

   - **3d** Restart the Identity Vault.
   - **3e** In Designer, ensure that you have installed a supported version of the Data Collection Service Base package. If necessary, install the latest version before continuing. For more information about software requirements, see the "Considerations for Installing Identity Reporting Components" on page 56.
   - **3f** Redeploy and restart the DCS driver in Designer.
4. In the **Outline** view, right-click the DCS driver, then select **Properties**.
5. Click **Driver Configuration**.
6. Click the **Driver Parameters** tab.
7. Click **Show connection parameters**, then select **show**.
8. Click **SSO Service Support**, then select **Yes**.
9. Specify the IP address and port for Identity Reporting.
10. Specify a password for the SSO Service Client. The default password is `driver`.
11. Click **Apply**, then click **OK**.
12. In the **Modeler** view, right-click the DCS driver, then select **Driver > Deploy**.
13. Click **Deploy**.
14. If prompted to restart the DCS driver, click **Yes**.
15. Click **OK**.

Migrating the Data Collection Service Driver

For the objects to synchronize into the Identity Information Warehouse, you must migrate the Data Collection Service driver.

1 Log in to iManager.

2 In the **Overview** panel for the Data Collection Service Driver, select **Migrate From Identity Vault**.

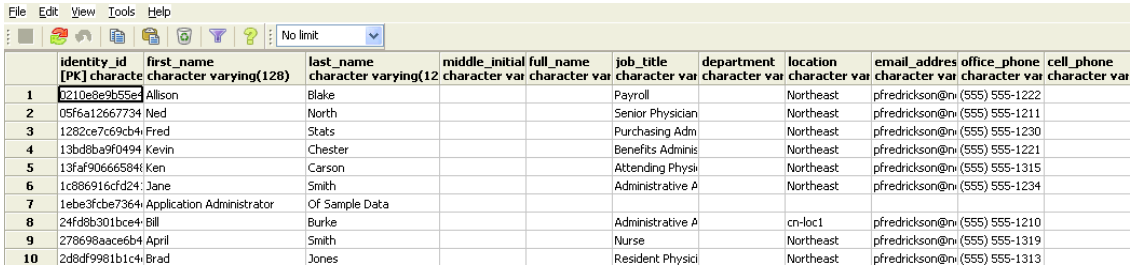
3 Select the organizations that contain relevant data, and click **Start**.

NOTE: Depending on the amount of data that you have, the migration process could take several minutes. Be sure to wait until the migration process is complete before you proceed.

---

4 Wait for the migration process to complete.

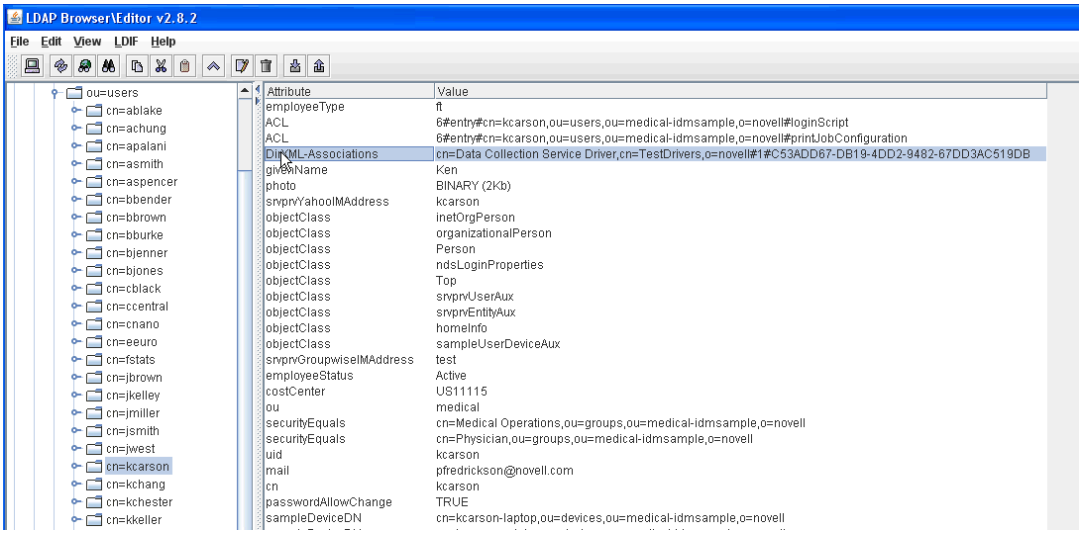
5 In the **idmrpt_identity** and **idmrpt_acct** tables, which provide information about the identities and accounts in the Identity Vault, ensure they contain the following type of information:



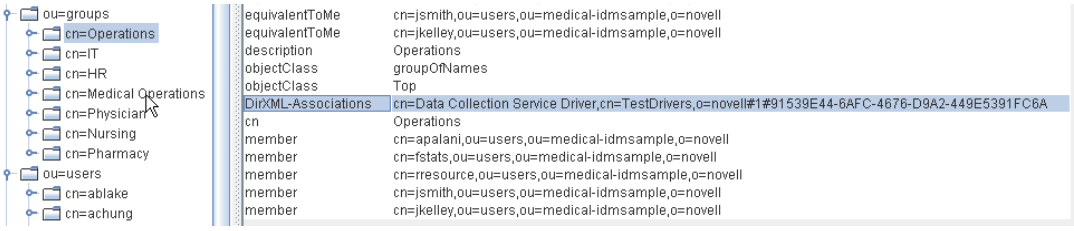
	identity_id [PK] characte	first_name character varying(128)	last_name character varying(12	middle_initial character var	full_name character var	job_title character var	department character var	location character var	email_addres character var	office_phone character var	cell_phone character var
1	0210e8e9b55e4	Allison	Blake			Payroll		Northeast	pfredrickson@n	(555) 555-1222	
2	05f6a12667734	Ned	North			Senior Physician		Northeast	pfredrickson@n	(555) 555-1211	
3	1282ce7c69cb4	Fred	Stats			Purchasing Adm		Northeast	pfredrickson@n	(555) 555-1230	
4	13bd8ba9f0494	Kevin	Chester			Benefits Adminis		Northeast	pfredrickson@n	(555) 555-1221	
5	13faf90666584	Ken	Carson			Attending Physi		Northeast	pfredrickson@n	(555) 555-1315	
6	1c886916cfd24	Jane	Smith			Administrative A		Northeast	pfredrickson@n	(555) 555-1234	
7	1ebe3fcbe7364	Application Administrator	Of Sample Data								
8	24fd8b301bce4	Bill	Burke			Administrative A		cn-loc1	pfredrickson@n	(555) 555-1210	
9	278698aace6b4	April	Smith			Nurse		Northeast	pfredrickson@n	(555) 555-1319	
10	2d8df9981b1c4	Brad	Jones			Resident Physici		Northeast	pfredrickson@n	(555) 555-1313	

6 In the LDAP browser, verify that the migration process adds the following references for DirXML-Associations:

- For each user, verify the following type of information:



- For each group, verify the following type of information:



7 Ensure that the data in the **idmrpt_group** table appears similar to the following information:

group_name character var	group_desc character var	dynamic_grou boolean	dynamic_rule character var	nested_group boolean	idmrpt_valid_from timestamp without tin	idmrpt_deleted boolean	idmrpt_syn_state smallint
Pharmacy	Pharmacy	FALSE		FALSE	2010-07-07 21:28:11	FALSE	1
IT	Information Tec	FALSE		FALSE	2010-07-07 21:28:11	FALSE	1
HR	Human Resourc	FALSE		FALSE	2010-07-07 21:28:11	FALSE	1
Physician	Physician	FALSE		FALSE	2010-07-07 21:28:11	FALSE	1
Operations	Operations	FALSE		FALSE	2010-07-07 21:28:11	FALSE	1
Medical Operatic	Medical Operatic	FALSE		FALSE	2010-07-07 21:28:11	FALSE	1
Nursing	Nursing	FALSE		FALSE	2010-07-07 21:28:11	FALSE	1

This table shows the name for each group, as well as flags indicating whether the group is dynamic or nested. It also shows whether the group has been migrated. The synchronization status (idmrpt_syn_state) could possibly be set to 0 if an object had been modified in the User Application but not yet migrated. For example, if a user were added to a group, and the driver had not been migrated yet, this value might be set to 0.

8 (Optional) Verify the data in the following tables:

- idmrpt_approver
- idmrpt_association
- idmrpt_category
- idmrpt_container
- idmrpt_idv_drivers
- idmrpt_idv_prd
- idmrpt_role
- idmrpt_resource
- idmrpt_sod

9 (Optional) Verify that the **idmrpt_ms_collect_state** table, which shows information about the data collection state for the Managed System Gateway Driver, contains now rows.

This table includes data about which REST endpoints for managed systems have been executed. At this point, the table has no rows because you have not started the collection process for this driver.

Adding Support for Custom Attributes and Objects

You can configure the Data Collection Service driver to gather and persist data for custom attributes and objects that are not part of the default data collection scheme. To do this, you need to modify the Data Collection Service driver filter. Modifying the filter does not trigger object synchronization immediately. Instead, the newly added attributes and objects are sent to the data collection services when add, modify, or delete events occur in the Identity Vault.

When you add support for custom attributes and objects, you need to modify the reports in order to include the extended attribute and object information. The following views provide current and historic data on the extended objects and attributes:

- idm_rpt_cfg.idmrpt_ext_idv_item_v
- idm_rpt_cfg.idmrpt_ext_item_attr_v

This process includes the following activities:

- "Configuring the Driver to Use Extended Objects" on page 136
- "Including a Name and Description in the Database" on page 137
- "Adding Extended Attributes to Known Object Types" on page 137

Configuring the Driver to Use Extended Objects

You can add any object or attribute to the Data Collection Service filter policy. When you add a new object or attribute, you need to make sure you map the GUID (with subscriber sync) and the Object Class (with subscriber notify), as shown in the following example:

```
<filter-class class-name="Device" publisher="ignore" publisher-create-
homedir="true" publisher-track-template-member="false" subscriber="sync">
<filter-attr attr-name="CN" merge-authority="default" publisher="ignore"
publisher-optimize-modify="true" subscriber="sync"/>
<filter-attr attr-name="Description" merge-authority="default"
publisher="ignore" publisher-optimize-modify="true" subscriber="sync"/>
<filter-attr attr-name="GUID" merge-authority="default" publisher="ignore"
publisher-optimize-modify="true" subscriber="sync"/>
<filter-attr attr-name="Object Class" merge-authority="default"
publisher="ignore" publisher-optimize-modify="true" subscriber="notify"/>
<filter-attr attr-name="Owner" merge-authority="default"
publisher="ignore" publisher-optimize-modify="true" subscriber="sync"/>
<filter-attr attr-name="Serial Number" merge-authority="default"
publisher="ignore" publisher-optimize-modify="true" subscriber="sync"/>
<filter-attr attr-name="sampleDeviceModel" from-all-classes="true" merge-
authority="default" publisher="ignore" publisher-optimize-modify="true"
subscriber="sync"/>
<filter-attr attr-name="sampleDeviceType" from-all-classes="true" merge-
authority="default" publisher="ignore" publisher-optimize-modify="true"
subscriber="sync"/>
</filter-class>
```

Including a Name and Description in the Database

If you want the object to have a name and description in the database, you need to add a schema mapping policy for _dcsName and _dcsDescription. The schema mapping policy maps the attribute values on the object instance to the columns idmrpt_ext_idv_item.item_name and idmrpt_ext_idv_item.item_desc, respectively. If you do not add a schema mapping policy, the attributes will be populated in the child table idmrpt_ext_item_attr.

For example:

```
<attr-name class-name="Device">
<nds-name>CN</nds-name>
<app-name>_dcsName</app-name>
</attr-name>
<attr-name class-name="Device">
<nds-name>Description</nds-name>
<app-name>_dcsDescription</app-name>
</attr-name>
```

The following example of SQL allows you to show these object and attribute values in the database:

```
SELECT
    item.item_dn,
    item.item_name,
    item.item_desc,
    attr.attribute_name,
    itemAttr.attribute_value,
    item.idmrpt_deleted as item_deleted,
    itemAttr.idmrpt_deleted as attr_deleted,
    item.item_desc,
    obj.object_class
FROM
    idm_rpt_data.idmrpt_ext_idv_item as item,
idm_rpt_data.idmrpt_ext_item_attr itemAttr, idm_rpt_data.idmrpt_ext_attr
as attr, idm_rpt_data.idmrpt_ext_obj as obj
WHERE
    item.object_id = obj.object_id and itemAttr.attribute_id =
attr.attribute_id and itemAttr.cat_item_id = item.item_id
ORDER BY
    item.item_dn, item.item_name
```

Adding Extended Attributes to Known Object Types

If an attribute is added to the filter policy on the Data Collection Service driver and not explicitly mapped to the reporting database in the XML reference file (`IdmrptIdentity.xml`), the value is populated and maintained in the idmrpt_ext_item_attr table, with an attribute reference in the idmrpt_ext_attr table.

The following example of SQL shows these extended attributes:

```
SELECT
    acct.idv_acct_dn,
    attrDef.attribute_name,
    attribute_value,
    attrVal.idmrpt_valid_from,
    cat_item_attr_id,
    attrVal.idmrpt_deleted,
    attrVal.idmrpt_syn_state
FROM
    idm_rpt_data.idmrpt_ext_item_attr as attrVal,
idm_rpt_data.idmrpt_ext_attr as attrDef, idm_rpt_data.idmrpt_identity as
idd, idm_rpt_data.idmrpt_idv_acct as acct
WHERE attrVal.attribute_id = attrDef.attribute_id and idd.identity_id =
acct.identity_id and attrVal.cat_item_id = acct.identity_id and
cat_item_type_id = 'IDENTITY'
```

In addition to the User object, you can add extended attributes to the filter policy on the following objects and populate the database with these attributes:

- nrfRole
- nrfResource
- Containers

  NOTE: The installed product provides support for organizationUnit, Organization, and Domain. The container types are maintained in the idmrpt_container_types table.

- Group
- nrfSod

You can see the association of the extended attributes to the parent table or object by looking at the idmrpt_cat_item_types.idmrpt_table_name column. This column describes how to join the idm_rpt_data.idmrpt_ext_item_attr.cat_item_id column to the primary key of the parent table.

Adding Support for Multiple Driver Sets

The Data Collection Service Scoping package (NOVLDCSSCPNG) provides static and dynamic scoping capabilities for enterprise environments with multiple driversets and multiple pairs of Data Collection Service Drivers and Managed System Gateway Drivers.

During or after installation, you need to determine the role for the Data Collection Service Driver that the package is being installed on. You need to select one of the following roles:

- **Primary** The driver synchronizes everything except subtrees of other driver sets. A primary Data Collection Service Driver may well service a whole Identity Vault or it may work in conjunction with one or multiple secondary drivers.
- **Secondary** The driver synchronizes only its own driver set, but nothing else. A secondary Data Collection Service Driver usually requires a primary driver to run in a different driverset or no data outside the local driver set is sent to the Data Collection Service.
- **Custom** Allows the administrator to define custom scoping rules. The only implicit scope is the local driver set, everything else is considered out-of-scope, unless it is explicitly added to the list of custom scopes. A custom scope is the distinguished name in slash format of a container in the Identity Vault whose subordinates or subtree should be synchronized.

The scoping package is only required in some configuration scenarios, as described below:

- **Single server with a single driver set Identity Vault:** For this scenario, you do not need scoping, and, therefore, you do not need to install the scoping package.
- **Multiple servers with a single driver set Identity Vault:** For this scenario, you need to follow these guidelines:
  - Make sure the Identity Manager server holds replicas of all partitions from which data should be collected.
  - For this scenario, no scoping is required, so do not install the scoping package
- **Multiple servers with a multiple driver set Identity Vault:** In this scenario, there are two basic configurations:
  - All servers hold a replica of all partitions from which data should be collected.

    For this configuration, you need to follow these guidelines:
    - Scoping is required to avoid the same change being processed by multiple DCS drivers.
    - You need to install the scoping package on all DCS drivers.
    - You need to select one DCS driver to be the Primary driver.
    - You need to configure all other DCS drivers to be Secondary drivers.
  - All servers *do not* hold a replica of all partitions from which data should be collected.

    Within this configuration, there are two possible situations:
    - All partitions from which data should be collected are being held by *only one* Identity Manager server

      In this case, you need to follow these guidelines:
      - Scoping is required to avoid the same change being processed by multiple DCS drivers.
      - You need to install the scoping package on all DCS drivers.
      - You need to configure all DCS drivers to be Primary drivers.
    - All partitions from which data should be collected are *not being held by only one* Identity Manager server (some partitions are held by more than one Identity Manager server).

      In this case, you need to follow these guidelines:
      - Scoping is required to avoid the same change being processed by multiple DCS drivers.
      - You need to install the scoping package on all DCS drivers.
      - You need to configure all DCS drivers to be Custom drivers.

        You need to define custom scoping rules for each driver and be sure not to create any overlapping scopes.

Configuring the Drivers to Run in Remote Mode with SSL

When running in remote mode, you can configure the Data Collection Service and Managed System Gateway drivers to use SSL. This section provides steps for configuring the drivers to run in remote mode with SSL.

To configure SSL using a Keystore for the Managed System Gateway Driver:

1. Create a server certificate in iManager.
   - **1a** In the **Roles and Tasks** view, click **NetIQ Certificate Server > Create Server Certificate**.
   - **1b** Browse to and select the server object where the Managed System Gateway Driver is installed.
   - **1c** Specify a certificate nickname.
   - **1d** Select **Standard** as the creation method, then click **Next**.
   - **1e** Click **Finish**, then click **Close**.
2. Export the server certificate using iManager.
   - **2a** In the **Roles and Tasks** view, click **NetIQ Certificate Access > Server Certificates**.
   - **2b** Select the certificate created in Step 1 on page 140 and click **Export**.
   - **2c** In the **Certificates** menu, select the name of your certificate.
   - **2d** Ensure that **Export private key** is checked.
   - **2e** Enter a password and click **Next**.
   - **2f** Click **Save the exported certificate**, and save the exported pfx certificate.
3. Import the pfx certificate exported in Step 2 on page 140 into the java key-store.
   - **3a** Use the keytool available with Java. You must use JDK 6 or later.
   - **3b** Enter the following command at a command prompt:

     ```
     keytool -importkeystore -srckeystore pfx certificate -srcstoretype
     PKCS12 -destkeystore Keystore Name
     ```

     For example:

     ```
     keytool -importkeystore -srckeystore cert.pfx -srcstoretype PKCS12
     -destkeystore msgw.jks
     ```

   - **3c** Enter the password when prompted to do so.
4. Modify the Managed System Gateway Driver configuration to use the keystore using iManager.
   - **4a** From **Identity Manager Overview**, click the driverset containing the Managed System Gateway Driver.
   - **4b** Click on the driver state icon and select **Edit properties > Driver configuration**.
   - **4c** Set **Show Connection Parameters** to true and set the **Driver configuration mode** to remote.
   - **4d** Enter the complete path of the keystore file and the password.
   - **4e** Save and restart the driver.

5 Modify the Data Collection Service Driver configuration to use the keystore using iManager.

   5a From **Identity Manager Overview**, click the driverset containing the Managed System Gateway Driver.

   5b Click on the driver state icon and select **Edit properties > Driver configuration**.

   5c Under the **Managed System Gateway Registration** header, set **Managed System Gateway Driver Configuration Mode** to remote.

   5d Enter the complete path of the keystore, password and the alias enter in Step 1c on page 140.

   5e Save and restart the driver.

Configuring Identity Reporting

After installing Identity Reporting, you can still modify many of the installation properties. To make changes, run the configuration update utility (`configupdate.sh`) file.

If you change any setting for Identity Reporting with the configuration tool, you must restart Tomcat for the changes to take effect. However, you do not need to restart the server after making changes in the web user interface for Identity Reporting.

- "Configuring the Managed System Gateway Driver" on page 141
- "Manually Adding the DataSource in the Identity Data Collection Services Page" on page 142
- "Running Reports on an Oracle Database" on page 142
- "Manually Generating the Database Schema" on page 142
- "Deploying REST APIs for Identity Reporting" on page 146
- "Connecting to a Remote PostgreSQL Database" on page 146

Configuring the Managed System Gateway Driver

The installer creates and configures the Data Collection Services Driver and the Managed System Gateway Driver for Identity Reporting.

Once the Managed System Gateway driver is configured, NetIQ recommends you to modify the KMO to **SSL EC Certificate DNS**. To modify the KMO, perform the following steps:

1 Log in to iManager.

2 Click **Identity Manager Administration**> **Identity Manager Overview**.

3 Browse to and select the driver set object, then click **Search**.

4 In the upper right corner of the Managed System Gateway driver icon, click **Edit Properties.**

5 In the **Driver Configuration** tab, perform the following steps:

   5a Select `show` from the **Show connection parameters** field.

   5b In the **KMO Name** field, change the value from **SSL CertificateDNS** to **SSL EC Certificate DNS**.

   5c Click **Apply** and then click **OK**.

Manually Adding the DataSource in the Identity Data Collection Services Page

1. Log in to Identity Reporting application.
2. Click **Data Sources**.
3. Click **Add**.
4. In the **Add Data Source** dialog box, click the **Select from predefined list** radio button.
5. Select **IDMDCSDataSource**.
6. Click **Save**.

Running Reports on an Oracle Database

Identity Reporting provides the ability to run reports against remote Oracle databases. Ensure that you have the `ojbc8.jar` file on the server where you are running the Oracle Database.

Manually Generating the Database Schema

To manually generate the database schema after installation, perform one of the following procedures for your database:

- "Configuring Create_rpt_roles_and_schemas.sql Schema against PostgreSQL Database" on page 142
- "Configuring Create_rpt_roles_and_schemas.sql Schema against Oracle Database" on page 143
- "Configuring Create_rpt_roles_and_schemas.sql Schema against MS SQL Database" on page 144
- "Clearing the Database Checksums" on page 144
- "(Optional) Increasing the Column Data Size" on page 145

Configuring Create_rpt_roles_and_schemas.sql Schema against PostgreSQL Database

1. Add the required roles to the database using the `create_dcs_roles_and_schemas.sql` and `create_rpt_roles_and_schemas.sql` SQLs located in `/opt/netiq/idm/apps/IDMReporting/sql/`.
2. Log in to PGAdmin as a postgres user.
3. Run the Query tool.
4. To create `Create_rpt_roles_and_schemas` and `Create_dcs_roles_and_schemas` procedures, copy the content from these SQLs to the Query tool and execute against the connected database.
5. To create `IDM_RPT_DATA`, `IDM_RPT_CFG`, and `IDMRPTUSER` roles, execute the following commands in the given order:

   ```
   Select CREATE_DCS_ROLES_AND_SCHEMAS('<Set pwd for IDM_RPT_DATA>', '<Set pwd for IDMRPTUSER>');
   ```

```
Select CREATE_RPT_ROLES_AND_SCHEMAS('<Set pwd for IDM_RPT_CFG>');
```

For example, if the password for IDM_RPT_DATA, IDMRPTUSER, and IDM_RPT_CFG are *password*, *password1*, and *password2* respectively, then you must execute the following commands:

```
Select CREATE_DCS_ROLES_AND_SCHEMAS('password', 'password1');
```

```
Select CREATE_RPT_ROLES_AND_SCHEMAS('password2');
```

6 Copy the content of get_formatted_user_dn.sql from /opt/netiq/idm/apps/IDMReporting/sql/ to the Query tool and execute against the connected database.

NOTE: The get_formatted_user_dn.sql function must be added manually when you select database schema creation option as **File**. If you select the database schema creation option as **Now** or **Startup**, the installer will add this function to the database.

Configuring Create_rpt_roles_and_schemas.sql Schema against Oracle Database

1 Add the required roles to the database using create_dcs_roles_and_schemas-orcale.sql and create_rpt_roles_and_schemas-orcale.sql from /opt/netiq/idm/apps/IDMReporting/sql/.

2 Log in to SQL Developer as a database admin (sysdba) user.

3 To create Create_rpt_roles_and_schemas and Create_dcs_roles_and_schemas procedures, copy the content from these SQLs to SQL Developer and execute against the connected database.

4 To create IDM_RPT_DATA, IDM_RPT_CFG, and IDMRPTUSER roles, execute the following commands in the given order:

```
begin
CREATE_DCS_ROLES_AND_SCHEMAS('<Set pwd for IDM_RPT_DATA>', '<Set pwd
for IDMRPTUSER>');
end;

begin
CREATE_RPT_ROLES_AND_SCHEMAS('<Set pwd for IDM_RPT_CFG>');
end;
```

For example, if the password for IDM_RPT_DATA, IDMRPTUSER, and IDM_RPT_CFG are *password*, *password1*, and *password2* respectively, then you must execute the following commands:

```
begin
CREATE_DCS_ROLES_AND_SCHEMAS('password', 'password1');
end;

begin
CREATE_RPT_ROLES_AND_SCHEMAS('password2');
end;
```

5 Assign the following permission:

```
GRANT CREATE PUBLIC SYNONYM to IDM_RPT_CFG;
```

6 Copy the content of `get_formatted_user_dn-oracle.sql` to SQL Developer from `/opt/netiq/idm/apps/IDMReporting/sql/` and execute against the connected database.

NOTE: The `get_formatted_user_dn-oracle.sql` function must be manually added to the database when you select database schema creation option as **File**. If you select the database schema creation option as **Now** or **Startup**, the installer will add this function to the database.

Configuring Create_rpt_roles_and_schemas.sql Schema against MS SQL Database

1 Execute `delete_create_dcs_roles_and_schemas-mssql.sql` and `delete_get_formatted_user_dn-mssql.sql`.

2 Add the required roles to the database using `create_dcs_roles_and_schemas.mssql` and `create_rpt_roles_and_schemas.mssql` from `/opt/netiq/idm/apps/IDMReporting/sql/`.

3 Log in to SQL Developer as a database admin user.

4 To create `Create_rpt_roles_and_schemas` and `Create_dcs_roles_and_schemas` procedures, copy the content from `create_dcs_roles_and_schemas.mssql` and `create_rpt_roles_and_schemas.mssql` to SQL Developer and execute against the connected database.

5 To create `IDM_RPT_DATA`, `IDM_RPT_CFG`, and `IDMRPTUSER` roles, execute the following commands in the given order:

```
execute CREATE_DCS_ROLES_AND_SCHEMAS '<Set pwd for IDM_RPT_DATA>',
'<Set pwd for IDMRPTUSER>'
```

```
execute CREATE_RPT_ROLES_AND_SCHEMAS \'<Set pwd for IDM_RPT_CFG>\'
```

6 Copy the content of `get_formatted_user_dn.sql` to SQL Developer from `/opt/netiq/idm/apps/IDMReporting/sql/` and execute against the connected database.

Clearing the Database Checksums

1 Locate the following `.sql` files in `/opt/netiq/idm/apps/IDMReporting/sql`.

- `DbUpdate-001-run-as-idm_rpt_data.sql`
- `DbUpdate-01-run-as-idm_rpt_cfg.sql`
- `DbUpdate-02-run-as-idm_rpt_cfg.sql`
- `DbUpdate-03-run-as-idm_rpt_data.sql`
- `DbUpdate-04-run-as-idm_rpt_data.sql`
- `DbUpdate-05-run-as-idm_rpt_data.sql`
- `DbUpdate-06-run-as-idm_rpt_cfg.sql`

2 Clear the database checksums

2a To run the clearchsum command with each `.sql`, append the following line at the beginning of each file:

```
update DATABASECHANGELOG set MD5SUM = NULL
go
```

The modified content should look similar to the following:

```
--
*****************************************************************
**
-- Update Database Script
--
*****************************************************************
**
-- Change Log: IdmDcsDataDropViews.xml
-- Ran at: 2/23/18 5:17 PM
-- Against: IDM_RPT_CFG@jdbc:oracle:thin:@192.168.0.1:1521/orcl
-- Liquibase version: 3.5.1
--
*****************************************************************
**
update databasechangelog set md5sum = null
go
```

2b Run each `.sql` with the corresponding user.

3 Commit the changes to the database.

(Optional) Increasing the Column Data Size

In previous versions of Identity Manager, long data fields failed to synchronize data with the Identity Reporting server due to character limitations. Identity Manager 4.8 provides an option to increase the character limitation with PostgreSQL, Oracle, and MS SQL databases for the following tables and their respective columns:

Name of the Table	Name of the Column
idm_rpt_data.idmrpt_idv_ent_bindings	ent_param_str
idm_rpt_data.idmrpt_idv_ent_bindings	ent_param_val
idm_rpt_data.idmrpt_idv_identity_trust	idv_ent_ref
idm_rpt_data.idmrpt_idv_identity_trust	trust_params
idm_rpt_data.idmrpt_idv_ent_bindings_hist	ent_param_str
idm_rpt_data.idmrpt_idv_ent_bindings_hist	ent_param_val
idm_rpt_data.idmrpt_idv_identity_trust_hist	idv_ent_ref
idm_rpt_data.idmrpt_idv_identity_trust_hist	trust_params

- For PostgreSQL, the character limitation has been increased automatically with Identity Manager 4.8 for all the fields as mentioned in the above table.

- For Oracle, you must run the `alter_column_length-oracle.sql` script from `/opt/netiq/idm/apps/IDMReporting/sql/` directory to increase the character limitation for all the columns as mentioned in above table.
- For MS SQL, you must run the `alter_column_length-mssql.sql` script from `/opt/netiq/idm/apps/IDMReporting/sql/` directory to increase the character limitation for indexed columns only. In this case, ENT_PARAM_STR is the only indexed column under the table `idmrpt_idv_ent_bindings`.

Deploying REST APIs for Identity Reporting

Identity Reporting incorporates several REST APIs that enable different features within the reporting functionality. These REST API uses the OAuth2 protocol for authentication.

On Tomcat, the `rptdoc war` and the `dcsdoc war` are automatically deployed when Identity Reporting is installed.

Connecting to a Remote PostgreSQL Database

If your PostgreSQL database is installed on a separate server, you need to change the default settings in the `postgresql.conf` and `pg_hba.conf` files in the remote database.

1. Change the listening address in the `postgresql.conf` file.

   By default, PostgreSQL allows to listen for the localhost connection. It does not allow a remote TCP/IP connection. To allow a remoteTCP/IP connection, add the following entry to the `/opt/netiq/idm/postgres/data/postgresql.conf` file:

   ```
   listen_addresses = '*'
   ```

   If you have multiple interfaces on the server, you can specify a specific interface to be listened.

2. Add a client authentication entry to the `pg_hba.conf` file.

   By default, PostgreSQL accepts connections only from the `localhost`. It refuses remote connections. This is controlled by applying an access control rule that allows a user to log in from an IP address after providing a valid password (the md5 keyword). To accept a remote connection, add the following entry to the `/opt/netiq/idm/postgres/data/pg_hba.conf` file.

   ```
   host all all 0.0.0.0/0 md5
   ```

   For example, 192.168.104.24/26 trust

   This works only for IPv4 addresses. For IPv6 addresses, add the following entry:

   ```
   host all all ::0/0 md5
   ```

   If you want to allow connection from multiple client computers on a specific network, specify the network address in the CIDR-address format in this entry.

   The pg_hba.conf file supports the following client authentication formats.

   - local database user authentication-method [authentication-option]
   - host database user CIDR-address authentication-method [authentication-option]

- hostssl database user CIDR-address authentication-method [authentication-option]
- hostnossl database user CIDR-address authentication-method [authentication-option]

Instead of CIDR-address format, you can specify the IP address and the network mask in separate fields using the following format:

- host database user IP-address IP-mask authentication-method [authentication-option]
- hostssl database user IP-address IP-mask authentication-method [authentication-option]
- hostnossl database user IP-address IP-mask authentication-method [authentication-option]

3 Test the remote connection.

3a Restart the remote PostgreSQL server.

3b Log in to the server remotely using the username and password.

Completing a Non-root Installation

When you install the Identity Manager engine and plug-ins as a non-`root` user, the process perform all intended installation activities. This section guides you through the manual process required to complete the installation.

Creating a Container for Password Policies

Identity Manager requires password policy objects in the Identity Vault. However, the non-`root` installation process does not create a container for password policies.

1 Log in to the Identity Manager tree in iManager.

2 Navigate to the security container in the Identity Vault.

3 Create a container for password policies.

Adding Support for Graphics in Email Notifications

If you install the Identity Vault and the Identity Manager engine as a non-`root` user, email notifications might fail to include the graphics or images provided in the email template. For example, when running the `do-send-email-from-template` action, Identity Manager sends the email but the included images are blank. You must update the driverset to ensure graphic support.

1 Log into your project in Designer.

2 In the Outline pane, expand **Identity Vault**.

3 Right-click **Driver Set**.

4 Select **Properties > Java**.

5 For JVM options, enter the following content:

```
-Dcom.novell.nds.dirxml.util.mail.templatepath=path_to_graphics_files
```

For example:

```
-Dcom.novell.nds.dirxml.util.mail.templatepath=/prod/eDirectory/opt/
novell/eDirectory/lib/dirxml/rules/manualtask/mt_files
```

6 Click **OK**.

7 Deploy the changes to the driverset:

   7a Right-click **Driver Set**.

   7b Select **Live > Deploy**.

   7c Select **Deploy**.

8 Restart the Identity Vault.

Activating Identity Manager

You do not need an activation code to install or initially run Identity Manager. However, without an activation code, Identity Manager stops functioning 90 days after installation. You can activate Identity Manager at any time during the 90 days or afterward. For more information, see Activating Identity Manager in *NetIQ Identity Manager Overview and Planning Guide*.

Creating SSL Certificate for Distributed Deployment

The following procedure explains how to create an SSL Certificate for distributed deployment:

1 Run the following command to set the Java path:

```
export PATH=/opt/netiq/common/jre/bin:$PATH
```

   Ensure that java path is set on the server. For example:

```
export PATH=/opt/netiq/common/jre/bin/:$PATH
```

2 Create a sample working directory.

```
mkdir -p /opt/certs
```

3 Generate the PKCS keystore:

```
keytool -genkey -alias < alias_name > -keyalg RSA -keystore <absolute
path of keystore > -validity 3650 -keysize 1024 -dname "CN=<Subject
Alternate Name>" -keypass <keypass> -storepass <storepass>
```

4 Generate the CSR (Certificate Signing Request):

```
keytool -certreq -v -alias < alias_name >  -file < name of the csr file
> -keypass < keypass >  -keystore < absolute path of the keystore >  -
storepass < storepass >
```

5 Generate a self-signed certificate:

   5a Log in to iManager as an administrator.

   5b Navigate to **Roles and Tasks** > **NetIQ Certificate Server** > **Issue Certificate.**

   5c Browse to the `.csr` file created in step 4. For example: `ua.csr`.

   5d Click **Next**.

   5e Specify the key usage and click **Next**.

   5f For the certificate type, select **Unspecified**.

5g Click **Next**.

5h Specify the validity of the certificate and click **Next.**

5i Select the File in binary DER format radio button.

5j Click **Next**.

5k Click **Finish**.

5l Download the certificate and copy the downloaded certificate to the /data directory.

6 Export the root certificate in `.der` format:

6a Log in to iManager as an administrator.

6b Navigate to **Roles and Tasks** > **NetIQ Certificate Access** > **Server Certificates**.

6c Select the **SSL CertificateDNS** check box and click **Export**.

6d In the **Certificates** drop-down list, select Organizational CA.

6e In the Export Format drop-down list, select **DER**.

6f Click **Next**.

6g Download the certificate and copy the downloaded certificate to the /data directory.

7 Import the certificates into the PKCS keystore as explained earlier.

```
keytool -import -trustcacerts -alias root -keystore <name of the
keystore> -file <root certificate > -storepass <password> -noprompt
```

For example:

```
keytool -import -alias <alias name> -keystore <name of the keystore> -
file <self-signed certificate> -storepass <password> -noprompt
```

8 Repeat the above procedure for each of the `osp`, `userapp` and `reporting` if they are used in different machines.

IV Deploying Identity Manager Containers

Identity Manager provides the flexibility of deploying Identity Manager Components through a containerized mechanism. Identity Manager uses Docker for managing containers. The Identity Manager components, that support containerization, are delivered as Docker images. The Docker images are self-sufficient to run on their own.

The Docker images are available for the following Identity Manager components:

- Identity Manager Engine
- Remote Loader
- iManager
- One SSO Provider (OSP)
- Fanout Agent
- ActiveMQ
- PostgreSQL (Redistribution)
- Identity Applications
- Self Service Password Reset (SSPR)
- Form Renderer
- Identity Reporting

NOTE: The Identity Configuration Generator image is used for generating the silent properties file.

All the functionalities and operations that can be achieved through the enterprise mode of installation are also available through the containerized mechanism.

7 Planning Your Container Deployment

The following sections describe the high-level planning required for a container-based deployment in Docker environment:

- "System Requirements" on page 153
- "Obtaining the Docker Images" on page 153

System Requirements

You must ensure that the following requirements are met for deploying the containers:

Software	Certified Versions
Docker	19.03.1 or later

Obtaining the Docker Images

Perform the following steps to obtain the Docker images:

1. Download the `Identity_Manager_4.8_Containers.tar.gz` from the download page.
2. Run the following command to extract the `.tar` file:

   ```
   tar -zxvf Identity_Manager_4.8_Containers.tar.gz
   ```

8 Identity Manager Containers Deployment

The Identity Manager containers deployment process requires pre-installation, installation, and post-installation work. The examples used in this section provides information on deploying containers in an Advanced Edition. However, the containers can also be deployed in Standard edition.

Some containers are dependent on others. The following table provides details on those containers that are dependent on other containers.

Table 8-1 *Dependent Containers*

Container	Dependent containers
OSP	◆ Identity Engine ◆ iManager
Identity Applications	◆ OSP ◆ Databases for Identity Applications
Form Renderer	Identity Applications
Identity Reporting	◆ Identity Applications ◆ Databases for Identity Reporting
SSPR	Identity Applications

Best Practices

This section includes some tips and best practices for deploying Docker containers:

- NetIQ recommends you to set a limit on the amount of CPU used for a container. This can be achieved by using the `--cpuset-cpus` flag in the docker run command.
- To set a restart policy for a container, use the `--restart` flag in the docker run command. It is recommended to choose the on-failure restart policy and limit the restart attempts to 5.
- To set a limit on the memory used by a container, use the `--memory` flag in the docker run command.
- If you want to back up the trace files for the deployed drivers, then you can place the trace file under `/config/idm/` or manually copy the trace file to the volumized folder.
- To set a limit on the number of processes allowed to run at any point in time, use the --pids-limit flag in the docker run command. It is recommended to limit the PID value to 300.
- For Identity Manager Engine container, if you want to view the `environ` file located at the / process directory of the `/proc` file system, use the `--cap-add=SYS_PTRACE` flag in the docker run command. By default, most of the privileges are restricted and only the required privileges are enabled. For more information, see Docker documentation.

- Ensure that the third party jar files are volume mounted so that they are available when the container is started every time. For example, if the `ojdbc.jar` is present in the `/opt/netiq/idm/apps/tomcat/lib` directory of the container, then you must volume mount the jar file using the following command:

  ```
  -v /host/ojdbc.jar:/opt/netiq/idm/apps/tomcat/lib/ojdbc.jar
  ```

For example, run the following sample command containing all the above arguments for deploying containers:

```
docker run -itd --cap-add=SYS_PTRACE --pids-limit<tune container pids
limit> --memory=<maximum amout of memory container can use> --restart=on-
failure:5 --cpuset-cpus=<CPUs in which to allow execution> --
network=<connect a container to network> -v <bind mount a volume> --
name=<assign a name to the container> <image name>
```

Managing Container Volume Data

Docker supports several mechanisms for data storage and persistence. One such mechanism of persisting container data is by using shared volumes in containers.

The examples used in this guide assumes that you create and use shared volumes. For example, create a shared volume called `/data` on your Docker host.

```
mkdir /data
```

The shared volume created requires permissions. use the following command to provide:

```
chmod -R 777 /data
```

However, you can use other volumes that Docker supports. For more information, see Docker documentation.

NOTE: Ensure that you have read-write permissions for the shared volumes.

Creating the Silent Properties File

Most of the Identity Manager containers offer an interactive mode of installation. However, NetIQ recommends the use of a silent properties file for the deployment of different containers.

NOTE: When the silent.properties file is generated, it will be available in the `/data` of the Docker host.

---

1. From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.
2. Run the following command to load the image:

   ```
   docker load --input IDM_48_idm_conf_generator.tar.gz
   ```

3. Deploy the container using the following sample command:

NOTE: Ensure that you specify the machine FQDN as a value for the hostname.

```
docker run -it --name=idm_conf_generator --
hostname=identitymanager.example.com -v /data:/config
idm_conf_generator:idm-4.8.0
```

4 Navigate to the `idm` directory.

5 Run the `create_silent_props.sh` file:

```
./create_silent_props.sh
```

6 Enter y to proceed with the installation and configuration of the components.

7 To install JRE, enter y.

8 Specify the path for the properties file.

NOTE: Ensure that you create the `silent.properties` file in the shared volume location, for example, `/config`.

9 Specify the following settings to create the silent properties file:

Parameter	Description
Silent Property file name with absolute path	Specify the path for the silent properties file.
Configure the Silent properties for Docker Containers	Specifies whether you want to configure the properties file for Docker containers.
Generate inputs for Kubernetes Orchestration	*Applies only if you have selected y in the* **Configure the Silent properties for Docker Containers** *option.* Specifies whether you want to generate the YAML file for Kubernetes.
Directory name with absolute path for creating kube yaml file	*Applies only if you have selected* **y** *in the* **Generate inputs for Kubernetes Orchestration** *option.* Specifies the path for creating the YAML file for Kubernetes. **NOTE:** It is recommended that you provide different paths for the Identity Applications and Identity Reporting YAML files.
Kubernetes volume mount path	*Applies only if you have selected y in the Generate inputs for Kubernetes Orchestration option.* Specifies the path for the Kubernetes volume.
Identity Manager Engine hostname for Kubernetes deployment	*Applies only if you have selected* **y** *in the* **Generate inputs for Kubernetes Orchestration** *option.* Specifies the hostname of the Identity Manager Engine.

10 Decide the Identity Manager server edition you want to install. Enter y for Advanced Edition and n for Standard Edition.

11 Decide if you want to configure the components in a typical or custom mode.

12 From the list of components available for installation, select the required components:

- To install Engine, select **Identity Manager Engine**.
- To install Identity Reporting, select **Identity Reporting**.
- To install Identity Applications, select **Identity Applications**.

For information about the configuration parameters, see "Understanding the Configuration Parameters" on page 77.

NOTE: ◆You must generate the `silent.properties` file for all components at once.

- Use FQDN for all IP related configuration prompts.
- The `SSO_SERVER_SSL_PORT`, `TOMCAT_HTTPS_PORT`, `UA_SERVER_SSL_PORT`, and `RPT_TOMCAT_HTTPS_PORT` must be unique ports. For example, modify the `SSO_SERVER_SSL_PORT` to 8543, `TOMCAT_HTTPS_PORT` and `UA_SERVER_SSL_PORT` to 18543, and `RPT_TOMCAT_HTTPS_PORT` to 28543 respectively.
- (Conditional) If you are deploying containers on a single server using the host network mode, you must specify the `tomcat.ks` path as `/opt/netiq/idm/apps/tomcat/conf/tomcat.ks` for the certificate-related prompts specific to OSP, Identity Applications, and Identity Reporting.

---

13 (Conditional) If you are deploying containers on a single server using the host network mode, you must perform the following steps after the silent properties file is generated:

- Ensure that the value for the `CUSTOM_OSP_CERTIFICATE` is set to `n`.
- Add the following entries at the end of the `silent.propertes` file:

```
SKIP_PORT_CHECK=1
CUSTOM_UA_CERTIFICATE="n"
TOMCAT_SSL_KEYSTORE_PASS="<password>"
CUSTOM_RPT_CERTIFICATE="n"
```

Modes of Container Deployment

NetIQ recommends you to use the following modes for deploying containers:

- Deploying containers on a single server
- Deploying containers on distributed servers

Deploying Containers on a Single Server

In this example, all the Identity Manager containers are deployed on a single Docker host using the host network mode.

Prerequisites

- Ensure that the hostname is in FQDN format as shown below:

```
<IP of the host> <FQDN> <short_name>
```

For example:

```
172.120.0.1     identitymanager.example.com      identitymanager
```

- You must generate the silent properties file before you deploy the containers. For more information on generating the silent properties file, see the "Creating the Silent Properties File" on page 156.

The containers must be deployed in the following order:

- "Deploying Identity Manager Engine Container" on page 159
- "Deploying Remote Loader Container" on page 160
- "Deploying Fanout Agent Container" on page 160
- "Deploying iManager Container" on page 161
- "Deploying OSP Container" on page 162
- "Deploying PostgreSQL Container" on page 162
- "Deploying Identity Applications Container" on page 163
- "Deploying Form Renderer Container" on page 164
- "Deploying ActiveMQ Container" on page 165
- "Deploying Identity Reporting Container" on page 165
- "Deploying SSPR Container" on page 166

Deploying Identity Manager Engine Container

1. Generate the silent properties file. For more information, see Creating the Silent Properties File.
2. From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.
3. Run the following command to load the image:

   ```
   docker load --input IDM_48_identityengine.tar.gz
   ```

4. Deploy the container using the following command:

   ```
   docker run -d --network=host --name=engine-container -v /data:/config -
   e SILENT_INSTALL_FILE=/config/silent.properties identityengine:idm-
   4.8.0
   ```

5. To verify whether the container was successfully deployed, check the log files by running the following command:

   ```
   tail -f /data/idm/log/idmconfigure.log
   ```

6. To log in to the container, run the following command:

   ```
   docker exec -it <container> <command>
   ```

   For example,

   ```
   docker exec -it engine-container bash
   ```

NOTE: To run the Identity Vault utilities such as `ndstrace` or `ndsrepair`, log in to the container as a non-root user called as `nds`. These utilities cannot be run if you are logged in as a root user. To log in to the container as a `nds` user, run the `docker exec -it engine-container sudo nds` command.

Deploying Remote Loader Container

1. From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.
2. Run the following command to load the image:

   ```
   docker load --input IDM_48_remoteloader.tar.gz
   ```

3. Deploy the container using the following command:

   ```
   docker run -d --network=host --name=rl-container -v /data:/config
   remoteloader:idm-4.8.0
   ```

   This deploys the 64-bit and 32-bit version of the Remote Loader. The driver files can be found at the `/opt/novell/eDirectory/lib/dirxml/classes/` directory of the container.
4. To log in to the container, run the following command:

   ```
   docker exec -it <container> <command>
   ```

   For example,

   ```
   docker exec -it rl-container bash
   ```

5. Configure Remote Loader. For more information, see Configuring the Remote Loader and Drivers in the *NetIQ Identity Manager Driver Administration Guide*.
6. Ensure that the configuration files are available in the shared volume.

Deploying Fanout Agent Container

1. From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.
2. Run the following command to load the image:

   ```
   docker load --input IDM_48_fanoutagent.tar.gz
   ```

3. Deploy the container using the following command:

   ```
   docker run -d --network=host --name=foa-container -v /data:/config
   fanoutagent:idm-4.8.0
   ```

4. To log in to the container, run the following command:

   ```
   docker exec -it <container> <command>
   ```

   For example,

   ```
   docker exec -it foa-container bash
   ```

5. Configure the Fanout Agent. For more information, see Configuring the Fanout Agent in the *NetIQ Identity Manager Driver for JDBC Fanout Implementation Guide*.

Deploying iManager Container

1. From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.
2. Run the following command to load the image:

   ```
   docker load --input IDM_48_iManager320.tar
   ```

3. Create a `.env` file with the required configuration to suit your environment. For example, the `iManager.env` is created in the `/data` directory.

   ```
   # Certificate Public Key Algorithm
   # Allowed Values: RSA, ECDSA256, ECDSA384
   CERTIFICATE_ALGORITHM=RSA
   # Cipher Suite
   # Allowed Values:
   # For RSA - NONE, LOW, MEDIUM HIGH
   # For ECDSA256 - SUITEB128ONLY
   # For ECDSA384 - SUITEB128, SUITEB192
   CIPHER_SUITE=NONE
   # Tomcat Server HTTP Port
   TOMCAT_HTTP_PORT=8080
   # Tomcat Server SSL Port
   TOMCAT_SSL_PORT=8743
   # iManager Authorized User (admin_name.container_name.tree_name)
   AUTHORIZED_USER=
   ```

4. Create a sub-directory under the shared volume `/data`, for example, `iManager`.
5. Deploy the container using the following command:

   ```
   docker run -d --network=host --name=iman-container -v /data:/config -v
   /data/iManager.env:/etc/opt/novell/iManager/conf/iManager.env
   imanager:3.2.0
   ```

6. To install the Identity Manager plug-ins, perform the following steps:

   6a Log in to iManager.

   ```
   https://identitymanager.example.com:8743/nps/
   ```

   6b Click **Configure**.

   6c Click **Plug-in Installation** and then click **Available NetIQ Plug-in Modules**.

   6d Select all the plug-ins from the **NetIQ Plug-in Modules** list and then click **Install**.

   To obtain the plug-ins offline, perform the following steps:

   1. Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.
   2. Mount the downloaded `.iso`.
   3. From the mounted location, navigate to the `/iManager/plugins` directory and obtain the required plug-ins.

   Alternatively, you can install the plug-ins from the iManager plug-ins website.

7. Restart the iManager container.

   ```
   docker restart iman-container
   ```

8 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it iman-container bash
```

For more information about deploying the iManager container, see the Deploying iManager Using Docker Container in the *NetIQ iManager Installation Guide*.

Deploying OSP Container

1 Generate the silent properties file. For more information, see Creating the Silent Properties File.

2 Ensure that the SSO_SERVER_SSL_PORT property is set to a unique port.

3 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

4 Run the following command to load the image:

```
docker load --input IDM_48_osp.tar.gz
```

5 Deploy the container using the following command:

```
docker run -d --network=host --name=osp-container -v /data:/config -e
SILENT_INSTALL_FILE=/config/silent.properties osp:idm-4.8.0
```

6 To verify whether the container was successfully deployed, check the log files by running the following command:

```
tail -f /data/osp/log/idmconfigure.log
```

7 Stop the container using the following command:

```
docker stop osp-container
```

8 Run the following command to modify the Tomcat shutdown port in the `server.xml` file. In the following example, the port 8005 will be changed to 18005:

```
sed -i "s~8005~18005~g" /data/osp/tomcat/conf/server.xml
```

9 Start the container using the following command:

```
docker start osp-container
```

10 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it osp-container bash
```

Deploying PostgreSQL Container

1 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

2 Run the following command to load the image:

```
docker load --input IDM_48_postgres.tar.gz
```

3 Create a sub-directory under the shared volume `/data`, for example, `postgres`.

```
mkdir postgres
```

4 Deploy the container using the following sample command:

```
docker run -d --network=host --name=postgresql-container -e
POSTGRES_PASSWORD=<password> -v /data/postgres:/var/lib/postgresql/data
postgres:9.6.12-alpine
```

For example,

```
docker run -d --network=host --name=postgresql-container -e
POSTGRES_PASSWORD=novell -v /data/postgres:/var/lib/postgresql/data
postgres:9.6.12-alpine
```

5 Create the `idmdamin` user for Identity Applications.

```
docker exec -it postgresql-container psql -U postgres -c "CREATE USER
idmadmin WITH ENCRYPTED PASSWORD '<password>'"
```

6 Create the Identity Applications, Workflow, and Identity Reporting databases.

```
docker exec -it postgresql-container psql -U postgres -c "CREATE
DATABASE idmuserappdb"
```

```
docker exec -it postgresql-container psql -U postgres -c "CREATE
DATABASE igaworkflowdb"
```

```
docker exec -it postgresql-container psql -U postgres -c "CREATE
DATABASE idmrptdb"
```

NOTE: These databases are used while you configure the Identity Applications and Identity Reporting containers.

---

7 Grant all the privileges on the databases for the `idmadmin` user:

```
docker exec -it postgresql-container psql -U postgres -c "GRANT ALL
PRIVILEGES ON DATABASE idmuserappdb TO idmadmin"
```

```
docker exec -it postgresql-container psql -U postgres -c "GRANT ALL
PRIVILEGES ON DATABASE igaworkflowdb TO idmadmin"
```

8 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it postgresql-container bash
```

Deploying Identity Applications Container

1 Generate the silent properties file. For more information, see Creating the Silent Properties File.

2 Ensure that the UA_SERVER_SSL_PORT property is set to a unique port.

3 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

4 Run the following command to load the image:

```
docker load --input IDM_48_identityapplication.tar.gz
```

5 Deploy the container using the following command:

```
docker run -d --network=host --name=idapps-container -v /data:/config -
e SILENT_INSTALL_FILE=/config/silent.properties
identityapplication:idm-4.8.0
```

6 To verify whether the container was successfully deployed, check the log files by running the following command:

```
tail -f /data/userapp/log/idmconfigure.log
```

7 Stop the container using the following command:

```
docker stop idapps-container
```

8 Run the following command to modify the Tomcat shutdown port in the `server.xml` file. In the following example, the port 8005 will be changed to 28005:

```
sed -i "s~8005~28005~g" /data/userapp/tomcat/conf/server.xml
```

9 Start the Docker container using the following command:

```
docker start idapps-container
```

10 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it idapps-container bash
```

NOTE: To modify any settings in the configuration update utility, launch `configupdate.sh` from the `/opt/netiq/idm/apps/configupdate/` directory of the Identity Applications container. The configuration update utility can be launched in console mode only.

Deploying Form Renderer Container

1 Generate the silent properties file. Select **Identity Applications** while generating the silent properties file. For more information, see Creating the Silent Properties File.

2 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

3 Run the following command to load the image:

```
docker load --input IDM_48_formrenderer.tar.gz
```

4 Deploy the container using the following command:

```
docker run -d --network=host --name=fr-container -v /data:/config -e
SILENT_INSTALL_FILE=/config/silent.properties formrenderer:idm-4.8.0
```

5 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it fr-container bash
```

Deploying ActiveMQ Container

1. From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.
2. Run the following command to load the image:

   ```
   docker load --input IDM_48_activemq.tar.gz
   ```

3. Deploy the container using the following command:

   ```
   docker run -d --network=host --name=amq-container -v /data:/config --
   env-file /data/silent.properties activemq:idm-4.8.0
   ```

4. To log in to the container, run the following command:

   ```
   docker exec -it <container> <command>
   ```

   For example,

   ```
   docker exec -it amq-container bash
   ```

5. Configure ActiveMQ. For more information, see Setting Up ActiveMQ Startup Service in the *NetIQ Identity Manager Driver for JDBC Fanout Implementation Guide*.

Deploying Identity Reporting Container

1. Generate the silent properties file. For more information, see Creating the Silent Properties File.
2. Ensure that the `TOMCAT_HTTPS_PORT` property is set to a unique port.
3. From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.
4. Run the following command to load the image:

   ```
   docker load --input IDM_48_identityreporting.tar.gz
   ```

5. Deploy the container using the following command:

   ```
   docker run -d --network=host --name=rpt-container -v /data:/config -e
   SILENT_INSTALL_FILE=/config/silent.properties identityreporting:idm-
   4.8.0
   ```

6. To verify whether the container was successfully deployed, check the log files by running the following command:

   ```
   tail -f /data/reporting/log/idmconfigure.log
   ```

7. Stop the container using the following command:

   ```
   docker stop rpt-container
   ```

8. Run the following command to modify the Tomcat shutdown port in the `server.xml` file. In the following example, the port 8005 will be changed to 38005:

   ```
   sed -i "s~8005~38005~g" /data/reporting/tomcat/conf/server.xml
   ```

9. *(Conditional) Applies only if you are using Identity Vault as the Certificate Authority.*

   Add the `-Dcom.sun.net.ssl.checkRevocation=false` parameter in the `export CATALINA_OPTS` entry of the `setenv.sh` file. In this example, the `setenv.sh` file is located under the `/data/reporting/tomcat/bin/` directory.

10 Start the Docker container using the following command:

```
docker start rpt-container
```

11 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it rpt-container bash
```

Deploying SSPR Container

Perform the following tasks to deploy the SSPR container:

1 Generate the silent properties file for SSPR. For more information, see Creating the Silent Properties File.

2 Create a sub-directory under the shared volume `/data`, for example, `sspr`.

```
mkdir sspr
```

3 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

4 Run the following command to load the image:

```
docker load --input IDM_48_sspr.tar.gz
```

5 Deploy the container using the following sample command:

```
docker run -d --network=host --name=sspr-container -v /data/sspr:/
config sspr/sspr-webapp:latest
```

6 Run the following command from the Docker host to copy the `silent.properties` file from the Docker host to SSPR container:

```
docker cp /data/silent.properties sspr-container:/tmp
```

7 Load the silent properties file to the SSPR container.

```
docker exec -it sspr-container /app/command.sh ImportPropertyConfig /
tmp/silent.properties
```

NOTE: Check if the `SSPRConfiguration.xml` is created under the `/config` directory of SSPR container and verify the content of the file.

---

8 Import the OAuth certificate to SSPR:

8a From the Docker host, edit the `SSPRConfiguration.xml` file located at `/data/sspr/` directory and set the value of the `configIsEditable` flag to **true** and save the changes.

8b Launch a browser and enter the `https://identitymanager.example.com:8443/sspr` URL.

8c Log in using administrator credentials, for example, `uaadmin`.

8d Click on the user, for example, `uaadmin`, on the top-right corner and then click **Configuration Editor**.

8e Specify the configuration password and click **Sign In**.

8f Click **Settings** > **Single Sign On (SSO) Client** > **OAuth** and ensure that all URLs use the HTTPS protocol and correct ports.

8g Under **OAuth Server Certificate**, click **Import from Server** to import a new certificate and then click **OK**.

8h Click at the top-right corner to save the certificate.

8i Review the changes and click **OK**.

8j After the SSPR application is restarted, edit the `SSPRConfiguration.xml` file and set the value of the `configIsEditable` flag to **false** and save the changes.

Deploying Containers on Distributed Servers

NetIQ recommends you to use host network mode for Identity Manager Engine container and overlay network for all other Identity Manager containers. In the examples used in this guide, we will deploy the Identity Manager container on Docker Host A and other Identity Manager containers on Docker Host B.

Perform the following steps to set up an overlay network:

1. Run the following command on Docker Host A:

   ```
   docker run -d -p <host port>:8500 -h consul --name <container name>
   progrium/consul -server -bootstrap
   ```

   For example:

   ```
   docker run -d -p 8500:8500 -h consul --name consul progrium/consul -
   server -bootstrap
   ```

2. On Docker Host B, edit the **docker** file located at `/etc/sysconfig/` directory and add the following line:

   ```
   DOCKER_OPTS="-H tcp://0.0.0.0:2375 -H unix:///var/run/docker.sock --
   cluster-advertise <Master Server Network Interface>:2375 --cluster-
   store consul://<Docker Host A IP Address>:<Docker Host A Port>"
   ```

   For example:

   ```
   DOCKER_OPTS="-H tcp://0.0.0.0:2375 -H unix:///var/run/docker.sock --
   cluster-advertise eth0:2375 --cluster-store consul://172.120.0.1:8500"
   ```

3. Restart the Docker service on Docker Host B:

   ```
   systemctl restart docker
   ```

4. In Docker Host B, run the following command to check whether Docker Host B is added to the cluster:

   ```
   docker info
   ```

   The sample output will be as follows:

   ```
   Cluster store: consul://<Docker HOST A IP Address>:8500
   Cluster advertise: <Docker HOST B IP Address>:2375
   ```

5. Create an overlay network on Docker Host B:

```
docker network create -d overlay --subnet=<subnet in CID format that
represents a network segment> --gateway=<ipv4 gateway> <name of the
overlay network>
```

For example:

```
docker network create -d overlay --subnet=192.168.0.0/24 --
gateway=192.168.0.1 idmoverlaynetwork
```

6 Run the following command to verify whether the overlay network is created:

```
docker network ls
```

Prerequisites

- The `/etc/hosts` file of all the Docker hosts in your Docker deployment must be updated with the details of all the containers running on that host. Ensure that the hostname for all containers are in Fully Qualified Domain Name (FQDN) format only.

  The host file entries can follow the below format for all the components:

  ```
  <IP of the container> <FQDN> <short_name>
  ```

  In the sample deployment used in this guide, add the following entries in the `/etc/hosts` file:

  ```
  172.120.0.1     identityengine.example.com       identityengine
  192.168.0.2     remoteloader.example.com         remoteloader
  192.168.0.3     fanoutagent.example.com          fanoutagent
  192.168.0.4     imanager.example.com             imanager
  192.168.0.5     osp.example.com                  osp
  192.168.0.6     postgresql.example.com           postgresql
  192.168.0.7     identityapps.example.com         identityapps
  192.168.0.8     formrenderer.example.com         formrenderer
  192.168.0.9     activemq.example.com             activemq
  192.168.0.10    identityreporting.example.com    identityreporting
  192.168.0.11    sspr.example.com                 sspr
  ```

- Ensure that the third party jar files are volume mounted so that they are available when the container is started every time. For example, if the `ojdbc.jar` is present in the `/opt/netiq/idm/apps/tomcat/lib` directory of the container, then you must volume mount the jar file using a sample command such as:

  ```
  -v /host/ojdbc.jar:/opt/netiq/idm/apps/tomcat/lib/ojdbc.jar
  ```

- You must generate the silent properties file before you deploy the containers. For more information on generating the silent properties file, see the "Creating the Silent Properties File" on page 156.

Exposing the Ports to Be Accessed by Containers

As a prerequisite, you must know the ports that you want to use for a container. You must expose the required ports and map the container ports with the ports on the Docker host. The following table provides information on ports that you must expose on the Docker hosts based on the examples provided in the guide.

Container	Default ports assumed as per the example deployment
Remote Loader	8090
Fanout Agent	Not applicable
iManager	8743
OSP	8543
Identity Applications	18543
Identity Reporting	28543
Form Renderer	8600
ActiveMQ	◆ 8161 ◆ 61616
PostgreSQL	5432
SSPR	8443 **NOTE:** SSPR container runs only on 8443 port.

However, you can customize the ports based on your requirement. The following considerations apply while you expose the ports:

- Ensure that you expose those ports which are not in use.
- The container port must be mapped to the same port on the Docker host. For example, the 8543 port on the container must be mapped to the 8543 port on the Docker host.

The containers must be deployed in the following order:

- "Deploying Identity Manager Engine Container" on page 170
- "Deploying Remote Loader Container" on page 170
- "Deploying Fanout Agent Container" on page 171
- "Deploying iManager Container" on page 171
- "Deploying OSP Container" on page 176
- "Deploying PostgreSQL Container" on page 177
- "Deploying Identity Applications Container" on page 178
- "Deploying Form Renderer Container" on page 179
- "Deploying ActiveMQ Container" on page 179
- "Deploying Identity Reporting Container" on page 180
- "Deploying SSPR Container" on page 181

Deploying Identity Manager Engine Container

1. Generate the silent properties file. For more information, see Creating the Silent Properties File.
2. From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.
3. Run the following command to load the image:

   ```
   docker load --input IDM_48_identityengine.tar.gz
   ```

4. Deploy the container using the following command:

   ```
   docker run -d --network=host --name=engine-container -v /etc/hosts:/
   etc/hosts -v /data:/config -e SILENT_INSTALL_FILE=/config/
   silent.properties identityengine:idm-4.8.0
   ```

5. To verify whether the container was successfully deployed, check the log files by running the following command:

   ```
   tail -f /data/idm/log/idmconfigure.log
   ```

6. To log in to the container, run the following command:

   ```
   docker exec -it <container> <command>
   ```

   For example,

   ```
   docker exec -it engine-container bash
   ```

NOTE: To run the Identity Vault utilities such as `ndstrace` or `ndsrepair`, log in to the container as a non-root user called as `nds`. These utilities cannot be run if you are logged in as a root user. To log in to the container as a `nds` user, run the `docker exec -it engine-container sudo nds` command.

Deploying Remote Loader Container

1. From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.
2. Run the following command to load the image:

   ```
   docker load --input IDM_48_remoteloader.tar.gz
   ```

3. Deploy the container using the following command:

   ```
   docker run -d --ip=192.168.0.2 --network=idmoverlaynetwork --
   hostname=remoteloader.example.com -p 8090:8090 --name=rl-container -v /
   etc/hosts:/etc/hosts -v /data:/config remoteloader:idm-4.8.0
   ```

   This deploys the 64-bit and 32-bit version of the Remote Loader. The driver files can be found at the `/opt/novell/eDirectory/lib/dirxml/classes/` directory of the container.

4. To log in to the container, run the following command:

   ```
   docker exec -it <container> <command>
   ```

   For example,

```
docker exec -it rl-container bash
```

5 Configure Remote Loader. For more information, see Configuring the Remote Loader and Drivers in the *NetIQ Identity Manager Driver Administration Guide*.

Deploying Fanout Agent Container

1 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

2 Run the following command to load the image:

```
docker load --input IDM_48_fanoutagent.tar.gz
```

3 Deploy the container using the following command:

```
docker run -d --ip=192.168.0.3 --network=idmoverlaynetwork --
hostname=fanoutagent.example.com --name=foa-container -v /etc/hosts:/
etc/hosts -v /data:/config fanoutagent:idm-4.8.0
```

4 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it foa-container bash
```

5 Configure the Fanout Agent. For more information, see Configuring the Fanout Agent in the *NetIQ Identity Manager Driver for JDBC Fanout Implementation Guide*.

Deploying iManager Container

1 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

2 Run the following command to load the image:

```
docker load --input IDM_48_iManager320.tar
```

3 Create a `.env` file with the required configuration to suit your environment. For example, the `iManager.env` is created in the `/data` directory.

```
# Certificate Public Key Algorithm
# Allowed Values: RSA, ECDSA256, ECDSA384
CERTIFICATE_ALGORITHM=RSA
# Cipher Suite
# Allowed Values:
# For RSA - NONE, LOW, MEDIUM HIGH
# For ECDSA256 - SUITEB128ONLY
# For ECDSA384 - SUITEB128, SUITEB192
CIPHER_SUITE=NONE
# Tomcat Server HTTP Port
TOMCAT_HTTP_PORT=8080
# Tomcat Server SSL Port
TOMCAT_SSL_PORT=8743
# iManager Authorized User (admin_name.container_name.tree_name)
AUTHORIZED_USER=
```

4 Create a sub-directory under the shared volume `/data`, for example, `iManager`.

5 Deploy the container using the following command:

```
docker run -d --ip=192.168.0.4 --name=iman-container --
network=idmoverlaynetwork --hostname=imanager.example.com -v /etc/
hosts:/etc/hosts -v /data:/config -v /data/iManager.env:/etc/opt/
novell/iManager/conf/iManager.env -p 8743:8743 imanager:3.2.0
```

6 To install the Identity Manager plug-ins, perform the following steps:

   6a Log in to iManager.

   `https://imanager.example.com:8743/nps/`

   6b Click **Configure**.

   6c Click **Plug-in Installation** and then click **Available NetIQ Plug-in Modules**.

   6d Select all the plug-ins from the **NetIQ Plug-in Modules** list and then click **Install**.

   To obtain the plug-ins offline, perform the following steps:

   1. Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.
   2. Mount the downloaded `.iso`.
   3. From the mounted location, navigate to the `/iManager/plugins` directory and obtain the required plug-ins.

   Alternatively, you can install the plug-ins from the iManager plug-ins website.

7 Restart the iManager container.

```
docker restart iman-container
```

8 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it iman-container bash
```

For more information about deploying the iManager container, see the Deploying iManager Using Docker Container in the *NetIQ iManager Installation Guide*.

Generating Certificates With Identity Vault Certificate Authority

(Conditional) This section applies only if you are using Identity Vault as the Certificate Authority.

The following components require you to generate certificates before they are deployed. Before you generate the certificates for the following components, ensure that you deploy the Identity Manager Engine and iManager containers.

- OSP
- Identity Applications
- Identity Reporting

Generating Certificates for OSP

Perform the following steps to generate the certificates:

1. Log in to the iManager container.

   ```
   docker exec -it <container> <command>
   ```

   For example,

   ```
   docker exec -it iman-container bash
   ```

2. Ensure that you set the Java path. For example, run the following command:

   ```
   export PATH=<java installed location>/bin:$PATH
   ```

   For example,

   ```
   export PATH=/opt/netiq/common/jre/bin/:$PATH
   ```

   NOTE: Ensure that the Java version installed is Azul Zulu 1.80_222 or later.

3. Generate the PKCS keystore:

   ```
   keytool -genkey -alias osp -keyalg RSA -storetype pkcs12 -keystore /
   config/tomcat-osp.ks -validity 3650 -keysize 2048 -dname
   "CN=osp.example.com" -keypass <password> -storepass <password>
   ```

4. Generate a certificate signing request:

   ```
   keytool -certreq -v -alias osp -file /config/osp.csr -keypass
   <password> -keystore /config/tomcat-osp.ks -storepass <password>
   ```

5. Generate a self-signed certificate:

   5a Launch iManager from Docker host and log in as an administrator.

   5b Navigate to **Roles and Tasks** > **NetIQ Certificate Server** > **Issue Certificate**.

   5c Browse to the `.csr` file created in step 4. For example, `osp.csr`.

   5d Click **Next**.

   5e Specify the key usage and click **Next**.

   5f For the certificate type, select **Unspecified**.

   5g Click **Next**.

   5h Specify the validity of the certificate and click **Next**.

   5i Select the **File in binary DER format radio** button.

   5j Click **Next**.

   5k Click **Finish**.

   5l Download the certificate and copy the downloaded certificate to the `/data` directory.

6. Export the root certificate in `.der` format:

   6a Launch iManager from Docker host and log in as an administrator.

   6b Navigate to **Roles and Tasks** > **NetIQ Certificate Access** > **Server Certificates**.

   6c Select the **SSL CertificateDNS** check box and click **Export**.

   6d In the **Certificates** drop-down list, select the Organizational CA.

   6e In the **Export Format** drop-down list, select DER.

6f Click **Next**.

6g Download the certificate and copy the downloaded certificate to the `/data` directory.

7 Import the certificates into the PKCS keystore you created in step 2:

```
keytool -import -trustcacerts -alias root -keystore /config/tomcat-
osp.ks -file /config/cert.der -storepass <password> -noprompt
```

```
keytool -import -alias osp -keystore /config/tomcat-osp.ks -file /
config/osp.der -storepass <password> -noprompt
```

NOTE: Ensure that the keystore is available in the path that was specified as an input for deployment.

Generating Certificates for Identity Applications

Perform the following steps to generate the certificates:

1 Log in to the iManager container.

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it iman-container bash
```

2 Ensure that you set the Java path. For example, run the following command:

```
export PATH=<java installed location>/bin:$PATH
```

For example,

```
export PATH=/opt/netiq/common/jre/bin/:$PATH
```

NOTE: Ensure that the Java version installed is Azul Zulu 1.80_222 or later.

---

3 Generate the PKCS keystore:

```
keytool -genkey -alias ua -keyalg RSA -storetype pkcs12 -keystore /
config/tomcat-ua.ks -validity 3650 -keysize 2048 -dname
"CN=identityapps.example.com" -keypass <password> -storepass <password>
```

4 Generate a certificate signing request:

```
keytool -certreq -v -alias ua -file /config/ua.csr -keypass <password>
-keystore /config/tomcat-ua.ks -storepass <password>
```

5 Generate a self-signed certificate:

5a Log in to iManager as an administrator.

5b Navigate to **Roles and Tasks** > **NetIQ Certificate Server** > **Issue Certificate**.

5c Browse to the `.csr` file created in step 4. For example, `ua.csr`.

5d Click **Next**.

5e Specify the key usage and click **Next**.

5f For the certificate type, select **Unspecified**.

5g Click **Next**.

5h Specify the validity of the certificate and click **Next**.

5i Select the **File in binary DER format radio** button.

5j Click **Next**.

5k Click **Finish**.

5l Download the certificate and copy the downloaded certificate to the `/data` directory.

6 Export the root certificate in `.der` format:

6a Log in to iManager as an administrator.

6b Navigate to **Roles and Tasks** > **NetIQ Certificate Access** > **Server Certificates**.

6c Select the **SSL CertificateDNS** check box and click **Export**.

6d In the **Certificates** drop-down list, select the Organizational CA.

6e In the Export Format drop-down list, select DER.

6f Click **Next**.

6g Download the certificate and copy the downloaded certificate to the `/data` directory.

7 Import the certificates into the PKCS keystore in step 2:

```
keytool -import -trustcacerts -alias root -keystore /config/tomcat-
ua.ks -file /config/cert.der -storepass <password> -noprompt
```

```
keytool -import -alias ua -keystore /config/tomcat-ua.ks -file /config/
ua.der -storepass <password> -noprompt
```

NOTE: Ensure that the certificates are available in the path that was specified as an input for deployment.

Generating Certificates for Identity Reporting

Perform the following steps to generate the certificates:

1 Log in to the iManager container.

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it iman-container bash
```

2 Ensure that you set the Java path. For example, run the following command:

```
export PATH=<java installed location>/bin:$PATH
```

For example,

```
export PATH=/opt/netiq/common/jre/bin/:$PATH
```

NOTE: Ensure that the Java version installed is Azul Zulu 1.80_222 or later.

---

3 Generate the PKCS keystore:

```
keytool -genkey -alias rpt -keyalg RSA -storetype pkcs12 -keystore /
config/tomcat-rpt.ks -validity 3650 -keysize 2048 -dname
"CN=identityreporting.example.com" -keypass <password> -storepass
<password>
```

4 Generate a certificate signing request:

```
keytool -certreq -v -alias rpt -file /config/rpt.csr -keypass
<password> -keystore /config/tomcat-rpt.ks -storepass <password>
```

5 Generate a self-signed certificate:

   5a Log in to iManager as an administrator.

   5b Navigate to **Roles and Tasks** > **NetIQ Certificate Server** > **Issue Certificate**.

   5c Browse to the `.csr` file created in step 4. For example, `rpt.csr`.

   5d Click **Next**.

   5e Specify the key usage and click **Next**.

   5f For the certificate type, select **Unspecified**.

   5g Click **Next**.

   5h Specify the validity of the certificate and click **Next**.

   5i Select the **File in binary DER format radio** button.

   5j Click **Next**.

   5k Click **Finish**.

   5l Download the certificate and copy the downloaded certificate to the `/data` directory.

6 Export the root certificate in `.der` format:

   6a Log in to iManager as an administrator.

   6b Navigate to **Roles and Tasks** > **NetIQ Certificate Access** > **Server Certificates**.

   6c Select the **SSL CertificateDNS** check box and click **Export**.

   6d In the **Certificates** drop-down list, select the Organizational CA.

   6e In the Export Format drop-down list, select DER.

   6f Click **Next**.

   6g Download the certificate and copy the downloaded certificate to the `/data` directory.

7 Import the certificates into the PKCS keystore you created in step 2:

```
keytool -import -trustcacerts -alias root -keystore /config/tomcat-
rpt.ks -file /config/cert.der -storepass <password> -noprompt
```

```
keytool -import -alias rpt -keystore /config/tomcat-rpt.ks -file /
config/rpt.der -storepass <password> -noprompt
```

NOTE: Ensure that the certificates are available in the path that was specified as an input for deployment.

Deploying OSP Container

NOTE: Before you deploy the OSP container, ensure that you generate the required certificates. For more information, see Generating Certificates for OSP.

---

1 Generate the silent properties file. For more information, see Creating the Silent Properties File.

2 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

3 Run the following command to load the image:

```
docker load --input IDM_48_osp.tar.gz
```

4 Deploy the container using the following command:

```
docker run -d --ip=192.168.0.5 --network=idmoverlaynetwork --
hostname=osp.example.com -p 8543:8543 --name=osp-container -v /etc/
hosts:/etc/hosts -v /data:/config -e SILENT_INSTALL_FILE=/config/
silent.properties osp:idm-4.8.0
```

5 To verify whether the container was successfully deployed, check the log files by running the following command:

```
tail -f /data/osp/log/idmconfigure.log
```

6 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it osp-container bash
```

Deploying PostgreSQL Container

1 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

2 Run the following command to load the image:

```
docker load --input IDM_48_postgres.tar.gz
```

3 Create a sub-directory under the shared volume `/data`, for example, `postgres`.

```
mkdir postgres
```

4 Deploy the container using the following sample command:

```
docker run -d --ip=192.168.0.6 --network=idmoverlaynetwork --
hostname=postgresql.example.com --name=postgresql-container -p
5432:5432 -e POSTGRES_PASSWORD=<password> -v /data/postgres:/var/lib/
postgresql/data -v /etc/hosts:/etc/hosts -v /data:/config
postgres:9.6.12-alpine
```

For example,

```
docker run -d --ip=192.168.0.6 --network=idmoverlaynetwork --
hostname=postgresql.example.com --name=postgresql-container -p
5432:5432 -e POSTGRES_PASSWORD=novell -v /data/postgres:/var/lib/
postgresql/data -v /etc/hosts:/etc/hosts -v /data:/config
postgres:9.6.12-alpine
```

5 Create the `idmdamin` user for Identity Applications.

```
docker exec -it postgresql-container psql -U postgres -c "CREATE USER
idmadmin WITH ENCRYPTED PASSWORD '<password>'"
```

6 Create the Identity Applications, Workflow, and Identity Reporting databases.

```
docker exec -it postgresql-container psql -U postgres -c "CREATE
DATABASE idmuserappdb"
```

```
docker exec -it postgresql-container psql -U postgres -c "CREATE
DATABASE igaworkflowdb"
```

```
docker exec -it postgresql-container psql -U postgres -c "CREATE
DATABASE idmrptdb"
```

NOTE: These databases are used while you configure the Identity Applications and Identity Reporting containers.

7 Grant all the privileges on the databases for the `idmadmin` user:

```
docker exec -it postgresql-container psql -U postgres -c "GRANT ALL
PRIVILEGES ON DATABASE idmuserappdb TO idmadmin"
```

```
docker exec -it postgresql-container psql -U postgres -c "GRANT ALL
PRIVILEGES ON DATABASE igaworkflowdb TO idmadmin"
```

8 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it postgresql-container bash
```

Deploying Identity Applications Container

NOTE: Before you deploy the Identity Applications container, ensure that you generate the required certificates. For more information, see Generating Certificates for Identity Applications.

1 Generate the silent properties file. For more information, see Creating the Silent Properties File.

NOTE: Specify the exposed port, 18543, as the value for the application server port.

2 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

3 Run the following command to load the image:

```
docker load --input IDM_48_identityapplication.tar.gz
```

4 Deploy the container using the following command:

```
docker run -d --ip=192.168.0.7 --network=idmoverlaynetwork --
hostname=identityapps.example.com -p 18543:18543 --name=idapps-
container -v /etc/hosts:/etc/hosts -v /data:/config -e
SILENT_INSTALL_FILE=/config/silent.properties identityapplication:idm-
4.8.0
```

5 To verify whether the container was successfully deployed, check the log files by running the following command:

```
tail -f /data/userapp/log/idmconfigure.log
```

6 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it idapps-container bash
```

7 Run the following command:

NOTE: Before performing this step, ensure that the container is deployed successfully.

```
/opt/netiq/common/jre/bin/keytool -importkeystore -srckeystore /config/
tomcat-osp.ks -srcstorepass <password> -destkeystore /opt/netiq/idm/
apps/tomcat/conf/idm.jks -deststorepass <password>
```

8 Type `yes` to overwrite the entry for the `root` alias.

9 Restart the Identity Applications container.

```
docker restart idapps-container
```

NOTE: To modify any settings in the configuration update utility, launch `configupdate.sh` from the `/opt/netiq/idm/apps/configupdate/` directory of the Identity Applications container. The configuration update utility can be launched in console mode only.

Deploying Form Renderer Container

1 Generate the silent properties file. For more information, see Creating the Silent Properties File.

2 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

3 Run the following command to load the image:

```
docker load --input IDM_48_formrenderer.tar.gz
```

4 Deploy the container using the following command:

```
docker run -d --ip=192.168.0.8 --network=idmoverlaynetwork --
hostname=formrenderer.example.com -p 8600:8600 --name=fr-container -v /
etc/hosts:/etc/hosts -v /data:/config -e SILENT_INSTALL_FILE=/config/
silent.properties formrenderer:idm-4.8.0
```

5 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it fr-container bash
```

Deploying ActiveMQ Container

1 Generate the silent properties file. For more information, see Creating the Silent Properties File.

2 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

3 Run the following command to load the image:

```
docker load --input IDM_48_activemq.tar.gz
```

4 Deploy the container using the following command:

```
docker run -d --ip=192.168.0.9 --network=idmoverlaynetwork --
hostname=activemq.example.com -p 8161:8161 -p 61616:61616 --name=amq-
container -v /etc/hosts:/etc/hosts -v /data:/config --env-file /data/
silent.properties activemq:idm-4.8.0
```

5 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it amq-container bash
```

6 Configure ActiveMQ. For more information, see Setting Up ActiveMQ Startup Service in the *NetIQ Identity Manager Driver for JDBC Fanout Implementation Guide*.

Deploying Identity Reporting Container

NOTE: Before you deploy the Identity Reporting container, ensure that you generate the required certificates. For more information, see Generating Certificates for Identity Reporting.

1 Generate the silent properties file. For more information, see Creating the Silent Properties File.

NOTE: Specify the exposed port, 28543, as the value for the application server port.

2 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

3 Run the following command to load the image:

```
docker load --input IDM_48_identityreporting.tar.gz
```

4 Deploy the container using the following command:

```
docker run -d --ip=192.168.0.10 --network=idmoverlaynetwork --
hostname=identityreporting.example.com -p 28543:28543 --name=rpt-
container -v /etc/hosts:/etc/hosts -v /data:/config -e
SILENT_INSTALL_FILE=/config/silent.properties identityreporting:idm-
4.8.0
```

5 To verify whether the container was successfully deployed, check the log files by running the following command:

```
tail -f /data/reporting/log/idmconfigure.log
```

6 To log in to the container, run the following command:

```
docker exec -it <container> <command>
```

For example,

```
docker exec -it rpt-container bash
```

7 Run the following command:

NOTE: Before performing this step, ensure that the container is deployed successfully.

```
/opt/netiq/common/jre/bin/keytool -importkeystore -srckeystore /config/
tomcat-osp.ks -srcstorepass <password> -destkeystore /opt/netiq/idm/
apps/tomcat/conf/idm.jks -deststorepass <password>
```

8 Type yes to overwrite the entry for the `root` alias.

9 Restart the Identity Reporting container.

```
docker restart rpt-container
```

Deploying SSPR Container

Perform the following tasks to deploy the SSPR container:

1 Generate the silent properties file for SSPR. Select **Identity Applications** while generating the silent properties file. For more information, see Creating the Silent Properties File.

2 Create a sub-directory under the shared volume `/data`, for example, `sspr`.

```
mkdir sspr
```

3 From the location where you have extracted the `Identity_Manager_4.8_Containers.tar.gz file`, navigate to the `Identity_Manager_4.8_Containers` directory.

4 Run the following command to load the image:

```
docker load --input IDM_48_sspr.tar.gz
```

5 Deploy the container using the following sample command:

```
docker run -d --ip=192.168.0.11 --network=idmoverlaynetwork --
hostname=sspr.example.com --name=sspr-container -v /etc/hosts:/etc/
hosts -v /data/sspr:/config -p 8443:8443 sspr/sspr-webapp:latest
```

6 Run the following command from the Docker host to copy the `silent.properties` file from the Docker host to SSPR container:

```
docker cp /data/silent.properties sspr-container:/tmp
```

7 Load the silent properties file to the SSPR container.

```
docker exec -it sspr-container /app/command.sh ImportPropertyConfig /
tmp/silent.properties
```

NOTE: Check if the SSPR`Configuration.xml` is created under the `/config` directory of SSPR container and verify the content of the file.

---

8 Import the OAuth certificate to SSPR:

8a From the Docker host, edit the SSPR`Configuration.xml` file located at `/data/sspr` directory and set the value of the `configIsEditable` flag to **true** and save the changes.

8b Launch a browser and enter the `https://sspr.example.com:8443/sspr` URL.

8c Log in using administrator credentials, for example, `uaadmin`.

8d Click on the user, for example, `uaadmin`, on the top-right corner and then click **Configuration Editor**.

8e Specify the configuration password and click **Sign In**.

8f Click **Settings** > **Single Sign On (SSO) Client** > **OAuth** and ensure that all URLs use the HTTPS protocol and correct ports.

8g Under **OAuth Server Certificate**, click **Import from Server** to import a new certificate and then click **OK**.

8h Click at the top-right corner to save the certificate.

8i Review the changes and click **OK**.

8j After the SSPR application is restarted, edit the SSPR`Configuration.xml` file and set the value of the `configIsEditable` flag to **false** and save the changes.

V Upgrading Identity Manager

This section provides information for upgrading Identity Manager components.

9 Preparing to Upgrade Identity Manager

This section provides information to help you prepare for upgrading your Identity Manager solution to the latest version.

WARNING: You must always rely on Identity Manager patch channels to update the components that are installed with Identity Manager 4.8. Otherwise, you can encounter severe conflicts during regular Identity Manager patch updates.

- "Checklist for Upgrading Identity Manager" on page 185
- "Understanding Upgrade Process" on page 186
- "Supported Upgrade Paths" on page 187
- "Backing Up the Current Configuration" on page 191

Checklist for Upgrading Identity Manager

To perform the upgrade, NetIQ recommends that you complete the steps in the following checklist:

	Checklist Items
❑	1. Understand the upgrade process. For more information, see "Understanding Upgrade Process" on page 186.
❑	2. Review the supported upgrade paths for upgrading to Identity Manager 4.8. For information about the supported upgrade paths, see "Supported Upgrade Paths" on page 187.
❑	3. Ensure that you have the installation kit to upgrade Identity Manager. For more information, see Where to Get Identity Manager in the *NetIQ Identity Manager Overview and Planning Guide*.
❑	4. Back up the current project, driver configuration, and databases. For more information, see "Backing Up the Current Configuration" on page 191.
❑	5. Upgrade Designer to the latest version.
❑	6. Upgrade Sentinel Log Management for IGA to the latest version. For more information, see "Upgrading Sentinel Log Management for IGA" on page 218.
❑	7. Upgrade Identity Vault (eDirectory) to 9.2. For more information, see "Upgrading the Identity Vault" on page 197.
❑	8. Stop the drivers that are associated with the server where you installed the Identity Manager engine. For more information, see "Stopping the Drivers" on page 204.

	Checklist Items
❑	9. Upgrade the Identity Manager engine. For more information, see "Upgrading the Identity Manager Engine" on page 197. **NOTE:** If you are migrating the Identity Manager engine to a new server, you can use the same eDirectory replicas that are on the current Identity Manager server. For more information, see "Migrating the Identity Manager Engine to a New Server" on page 232.
❑	10. (Conditional) If any of the drivers in the driver set for the Identity Manager Engine are Remote Loader drivers, upgrade the Remote Loader servers for each driver. For more information, see "Upgrading the Remote Loader" on page 200.
❑	11. Upgrade iManager to 3.2. For more information, see "Upgrading iManager" on page 202.
❑	12. Update the iManager plug-ins to match the version of iManager. For more information, see "Updating iManager Plug-ins after an Upgrade or Re-installation" on page 203.
❑	13. (Conditional) Upgrade the packages on the existing drivers if a newer version of packages is available. For more information, see "Upgrading the Identity Manager Drivers" on page 206. This is only required if you want to use the functionality included in the new package for your existing driver.
❑	14. Upgrade the Identity Applications. For more information, see "Upgrading Identity Applications" on page 207.
❑	15. Upgrade Identity Reporting. For more information, see "Upgrading Identity Reporting" on page 217.
❑	16. Start the drivers associated with the Identity Applications and the Identity Manager engine. For more information, see "Starting the Drivers" on page 205.
❑	17. (Conditional) If you migrated the Identity Manager engine or the identity applications to a new server, add the new server to the driver set. For more information, see "Adding New Servers to the Driver Set" on page 221.
❑	18. (Conditional) If you have custom policies and rules, restore your customized settings. For more information, see "Restoring Custom Policies and Rules to the Driver" on page 222.
❑	19. Upgrade Analyzer. For more information, see "Upgrading Analyzer" on page 221.
❑	20. Activate your upgraded Identity Manager solution. For more information, see Activating Identity Manager in *NetIQ Identity Manager Overview and Planning Guide*.

Understanding Upgrade Process

When you want to install a newer version of an existing Identity Manager installation, you usually perform an **upgrade**. However, when the new version of Identity Manager does not support a direct upgrade path from your existing version, you must first upgrade to a version from which upgrade to 4.8 is possible. Alternatively, you can perform a migration to a new machine. NetIQ defines **migration** as the process of installing Identity Manager on a new server and then migrating the existing data to the new server.

Upgrade

- **Identity Manager 4.8 Standard Edition:** If you want to upgrade from Identity Manager 4.7 Standard Edition to Identity Manager 4.8 Standard Edition, perform the steps listed in the Upgrading Identity Manager 4.7 Standard Edition to Identity Manager 4.8 Standard Edition section of the *Quick Start Guide for Installing and Upgrading NetIQ Identity Manager 4.8 Standard Edition*.
- **Identity Manager 4.7 Advanced Edition:** If you currently have Identity Manager 4.7 Advanced Edition, you can directly upgrade it to Identity Manager 4.8 Advanced Edition. For more information, see "Checklist for Upgrading Identity Manager" on page 185.

Migration

In some cases, you cannot perform a direct upgrade. In such scenarios, migration is preferred. For example, if you previously installed Identity Manager on a server running an operating system that is no longer supported, you must perform a migration instead of an upgrade.

If you have multiple servers associated with a driver set, you can perform an upgrade or a migration on one server at a time. If you cannot upgrade the servers at the same time, the drivers continue to work with the different versions of Identity Manager until the upgrades for each server are completed.

IMPORTANT: If you enable features for drivers that are supported only on Identity Manager 4.8 or later, the drivers stop working on the servers with mixed versions. The older engines cannot handle the new functionality. This breaks the drivers until all servers are upgraded to Identity Manager 4.8 or later.

Switch From Advanced Edition to Standard Edition

Identity Manager allows you to switch from Advanced Edition to Standard Edition during the product evaluation period or after activating Advanced Edition.

IMPORTANT: If you have already applied Advanced Edition activation, you need not move to Standard Edition as all Standard Edition functionality is available in Advanced Edition. You must switch to Standard Edition only if you do not want any Advanced Edition functionality in your environment and want to scale down your Identity Manager deployment. For more information, see Chapter 11, "Switching from Advanced Edition to Standard Edition," on page 225.

Supported Upgrade Paths

Identity Manager 4.8 support upgrade from 4.7.x and 4.6.4 versions. Before starting the upgrade, NetIQ recommends that you review the information from the release notes for your current version.

NOTE: Upgrading Identity Manager to 4.8 version requires you to apply the Identity Manager 4.8 Upgrade Enablement Patch. Conditions for applying this patch depends on your current version of Identity Manager. For more information, see NetIQ Identity Manager 4.8 Upgrade Enablement Patch Release Notes.

- "Upgrading from Identity Manager 4.7.x Versions" on page 188
- "Upgrading from Identity Manager 4.6.x Versions" on page 189

Upgrading from Identity Manager 4.7.x Versions

The following table lists the component-wise upgrade paths for Identity Manager 4.7.x versions:

Component	Base Version	Upgraded Version
Identity Manager Engine	4.7.x	1. Upgrade the operating system to a supported version. 2. Upgrade Identity Vault to 9.2. 3. Upgrade Identity Manager Engine to 4.8.
Remote Loader/Fanout Agent	4.7.x	Install 4.7 Remote Loader/Fanout Agent.
Designer	4.7.x	Install Designer 4.8.
Analyzer	4.7.x	Install Analyzer 4.8.
Identity Applications	4.7.x	Before you upgrade Identity Applications, ensure that the Identity Vault and Identity Manager engine are upgraded to 9.2 and 4.8 respectively. 1. Upgrade the operating system to a supported version. 2. Upgrade the database to a supported version. For the supported database versions, see the NetIQ Identity Manager Technical Information website. 3. (Conditional) If SSPR is installed on a separate server, upgrade the component to 4.8 version. 4. Update the User Application driver and Roles and Resources driver packages. 5. Upgrade Identity Applications to 4.8. 6. Stop Tomcat.

Component	Base Version	Upgraded Version
Identity Reporting	4.7.x	1. Upgrade the operating system to a supported version. 2. Upgrade the database to a supported version. For more information about the supported database versions, see the NetIQ Identity Manager Technical Information website. 3. Upgrade SLM for IGA to a supported version. 4. Upgrade Identity Reporting to 4.8. 5. (Conditional) Create a data synchronization policy from the Identity Manager Data Collection Services page.

Before starting the upgrade, NetIQ recommends that you review the information from the release notes for your version from the NetIQ documentation page.

Upgrading from Identity Manager 4.6.x Versions

The following table lists component-wise upgrade paths for Identity Manager 4.6.x versions:

Component	Base Version	Intermediate Step	Upgraded Version
Identity Manager Engine	4.6.x, where x is 0 to 3	4.6.4	1. Upgrade the operating system to a supported version. 2. Upgrade Identity Vault to 9.2. 3. Upgrade Identity Manager Engine to 4.8.
Remote Loader/ Fanout Agent	4.6.x, where x is 0 to 3	4.6.4	Install 4.8 Remote Loader/Fanout Agent.
Designer	4.6.x, where x is 0 to 3		Install Designer 4.8.

Component	Base Version	Intermediate Step	Upgraded Version
Identity Applications	4.6.x, where x is 0 to 3	4.6.4	Before you upgrade Identity Applications, ensure that Identity Vault and Identity Manager engine are upgraded to 9.2 and 4.8 versions respectively. 1. Upgrade the operating system to a supported version. 2. Update the User Application driver and Roles and Resources driver packages. 3. Upgrade the database to a supported version. For the supported database versions, see the NetIQ Identity Manager Technical Information website. 4. (Conditional) If SSPR is installed on a separate server, upgrade the component to the 4.8 version. 5. Upgrade Identity Applications to 4.8. 6. Stop Tomcat.
Identity Reporting	4.6.x, where x is 0 to 3	4.6.4	1. Upgrade the operating system to a supported version. 2. Upgrade the database to a supported version. For more information about the supported database versions, see the NetIQ Identity Manager Technical Information website. 3. Upgrade SLM for IGA to a supported version. 4. Migrate Identity Reporting to 4.8. 5. (Conditional) Create a data synchronization policy from the Identity Manager Data Collection Services page.

Before starting the upgrade, NetIQ recommends that you review the information from the release notes for your version from the NetIQ documentation page.

Backing Up the Current Configuration

Before upgrading, NetIQ recommends that you back up the current configuration of your Identity Manager solution. There are no additional steps required to back up the User Application. All User Application configuration is stored in the User Application driver. You can create the backup in the following ways:

- "Exporting the Designer Project" on page 191
- "Exporting the Driver Configuration" on page 192

Exporting the Designer Project

A Designer project contains the schema and all driver configuration information. Creating a project of your Identity Manager solution allows you to export all of the drivers in one step instead of creating a separate export file for each driver.

- "Exporting the Current Project" on page 191
- "Creating a New Project from the Identity Vault" on page 191

Exporting the Current Project

If you already have a Designer project, verify that the information in the project is synchronized with what is in the Identity Vault:

1. In Designer, open your project.
2. In the Modeler, right-click the Identity Vault, then select **Live > Compare**.
3. Evaluate the project and reconcile any differences, then click **OK**.

   For more information, see "Using the Compare Feature When Deploying" in the *NetIQ Designer for Identity Manager Administration Guide*.
4. On the toolbar, select **Project > Export**.
5. Click **Select All** to select all resources to export.
6. Select where to save the project and in what format, then click **Finish**.

   Save the project in any location, other than the current workspace. When you upgrade to Designer, you must create a new workspace location. For more information, see "Exporting a Project" in the NetIQ Designer for Identity Manager Administration Guide.

Creating a New Project from the Identity Vault

If you do not have a Designer project of your current Identity Manager solution, you must create a project to back up your current solution.

1. Install Designer.
2. Launch Designer, then specify a location for your workspace.
3. Select whether you want to check for online updates, then click **OK**.
4. On the Welcome page, click **Run Designer**.
5. On the toolbar, select **Project > Import Project > Identity Vault**.

6 Specify a name for the project, then either use the default location for your project or select a different location.

7 Click **Next**.

8 Specify the following values for connecting to the Identity Vault:

- **Host Names:** which represents the IP address or DNS name of the Identity Vault server
- **User name:** which represents the DN of the user used to authenticate to the Identity Vault
- **Password:** which represents the password of the authentication user

9 Click **Next**.

10 Leave the Identity Vault Schema and the Default Notification Collection selected.

11 Expand the Default Notification Collection, then de-select the languages you do not need.

The Default Notification Collections are translated into many different languages. You can import all languages or select only the languages that you use.

12 Click **Browse**, then browse to and select a driver set to import.

13 Repeat Step 12 for each driver set in this Identity Vault, then click **Finish**.

14 Click **OK** after the project is imported.

15 If you only have one Identity Vault, you are finished. If you have multiple Identity Vaults, proceed with Step 16.

16 Click **Live > Import** on the toolbar.

17 Repeat Step 8 through Step 14 for each additional Identity Vault.

Exporting the Driver Configuration

Creating an export of the drivers takes a backup of your current configuration. However, Designer currently does not create a backup of the Roles Based Entitlements driver and policies. Use iManager to verify that you have an export of the Roles Based Entitlement driver.

- "Using Designer to Export the Driver Configurations" on page 192
- "Using iManager to Create an Export of the Driver" on page 193

Using Designer to Export the Driver Configurations

1 Verify that your project in Designer has the most current version of your driver. For more information, see "Importing a Library, a Driver Set, or a Driver from the Identity Vault" in the *NetIQ Designer for Identity Manager Administration Guide*.

2 In the Modeler, right-click the line of the driver that you are upgrading.

3 Select **Export to a Configuration File**.

4 Browse to a location to save the configuration file, then click **Save**.

5 Click **OK** on the results page.

6 Repeat Step 1 through Step 5 for each driver.

Using iManager to Create an Export of the Driver

1. In iManager, select **Identity Manager > Identity Manager Overview**.
2. Browse to and select the location in the tree to search for Driver Set objects, then click the search icon ▸.
3. Click the Driver Set object that holds the driver you want to upgrade.
4. Click the driver you want to upgrade, then click **Export**.
5. Click **Next**, then select **Export all contained policies, linked to the configuration or not**.
6. Click **Next**, then click **Save As**.
7. Select **Save to Disk**, then click **OK**.
8. Click **Finish**.
9. Repeat Step 1 through Step 8 for each driver.

10 Upgrading Identity Manager Components

This section provides specific information for upgrading individual components of Identity Manager. This section also provides steps that you might need to take after performing an upgrade.

- "Considerations for Upgrade" on page 195
- "Upgrade Sequence" on page 196
- "Upgrading Designer" on page 196
- "Upgrading Identity Manager Engine" on page 197
- "Stopping and Starting Identity Manager Drivers" on page 204
- "Upgrading the Identity Manager Drivers" on page 206
- "Upgrading Identity Applications" on page 207
- "Upgrading Identity Reporting" on page 217
- "Upgrading Analyzer" on page 221
- "Adding New Servers to the Driver Set" on page 221
- "Restoring Custom Policies and Rules to the Driver" on page 222

Considerations for Upgrade

Review the following considerations before beginning to upgrade the Identity Manager components:

- You can upgrade the components only in the console mode. Silent upgrade is not supported.
- You must upgrade one component at a time.
- Identity Vault must be separately upgraded. This version supports eDirectory 9.2.
- If Identity Vault is configured on BTRFS file system, the upgrade process will not succeed. This version supports only Ext3, Ext4, and XFS file systems.
- Ensure that there are no events in the cache file before you begin upgrading Identity Manager Engine. If your driver is using MapDB, ensure that your upgraded driver works correctly with the upgraded Engine. For more information, see "Working with MapDB 3.0.5" on page 198.
- You must review the recommended server setup before upgrading the components. Identity Manager mandates that Identity Applications and OSP are installed on the same computer. However, Identity Reporting supports a locally or a remotely installed OSP.
- After upgrading Identity Manager to the 4.8 version, the new JRE files are located in the `/opt/netiq/common/jre/` directory. If required, back up the old JRE files and any customizations from the `/opt/netiq/idm/jre/` directory and then delete the files.
- Ensure that the Identity Applications, Identity Reporting, and the databases are uniformly configured with either the FQDN or IP address. In other words, you must not configure these components with a combination of FQDN and IP addresses.

- If you are using a supported version of Oracle or Microsoft SQL Server database, you must configure two database instances for Identity Applications to work correctly; Identity Applications (`idmuserappdb`) database and the Workflow (`igaworkflowdb`) database. Ensure that you configure the database instances on the same server.
- If you are using NAM Reverse Proxy configuration with Identity Applications to load forms, you must utilize NAM 5.0 version.

Upgrade Sequence

You must upgrade only one Identity Manager component at a time. Upgrade the components in the following sequence:

1. Designer
2. Sentinel Log Management for IGA
3. Identity Vault
4. Identity Manager Engine
5. Remote Loader
6. Fanout Agent
7. iManager
8. Identity Applications (for Advanced Edition)
9. Identity Reporting (also installs OSP for Standard Edition)
10. Analyzer
11. (Conditional) SSPR (required for Standard Edition)

Upgrading Designer

1 Log in as an administrator to the server where Designer is installed.

2 To create a backup copy of your projects, export your projects.

For more information about exporting, see "Exporting a Project" in the *NetIQ Designer for Identity Manager Administration Guide*.

3 Launch the Designer installation program. For more information, see "Installing Designer" on page 71.

After upgrading to the current version of Designer, you must import all Designer projects from the older version. When you initiate the import process, Designer runs the Project Converter Wizard, which converts the older projects to the current version. In the wizard, select **Copy project into the workspace**. For more information about the Project Converter, see the *NetIQ Designer for Identity Manager Administration Guide*.

Upgrading Identity Manager Engine

Ensure that you upgrade Identity Vault before upgrading the Identity Manager engine. The Identity Manager engine upgrade process updates the driver shim files that are stored in the file system on the host computer.

Upgrading the Identity Vault

1. Download the `Identity_Manager_4.8_Linux.iso` as instructed in Where to Get Identity Manager in the *NetIQ Identity Manager Overview and Planning Guide*.
2. Mount the downloaded `.iso`.
3. From the root directory of the `.iso` file, navigate to the `IDVault/setup` directory.
4. Run the following command:

   ```
   ./nds-install
   ```

5. Accept the license agreement and proceed with the installation.
6. Specify **adminDN**. For example, `cn=admin.ou=sa.o=system`.
7. Specify y when prompted for stopping eDirectory instances and upgrading NICI.
8. Specify if you want to configure **Enhanced Background Authentication**.

NOTE: Run `ndsconfig` upgrade after `nds-install`, if DIB upgrade fails and the `nds-install` command prompts to do so. If eDirectory services are not starting after an upgrade, run the `ndsconfig upgrade` command. For more information, see the NetIQ eDirectory Installation Guide.

Upgrading the Identity Manager Engine

Verify that the drivers are stopped. For more information, see "Stopping the Drivers" on page 204.

Perform the following steps to upgrade the Identity Manager Engine:

1. Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.
2. Mount the downloaded `.iso`.
3. Run the following command:

   ```
   ./install.sh
   ```

4. Read through the license agreement.
5. Enter y to accept the license agreement.
6. Specify whether you want upgrade the Identity Manager components. The available options are **y** and **n**.

7 Select Identity Manager Engine.

8 Specify the following details:

Identity Vault Administrator: Specify the Identity Vault administrator name.

Identity Vault Administrator Password: Specify the Identity Vault Administrator password.

The engine upgrade process retains some of the existing MapDB cache files (`dx*`) in the Identity Vault's DIB directory. You must manually remove these files for a driver using MapDB after upgrading the driver. For more information, see "Working with MapDB 3.0.5" on page 198.

Working with MapDB 3.0.5

Identity Manager adds support for MapDB 3.0.5. In addition to Identity Manager Engine, MapDB is used by the following Identity Manager drivers:

- Data Collection Services
- JDBC
- LDAP
- Managed System Gateway
- Office 365 and Azure Active Directory
- Salesforce

If you are using any of these drivers, you must review the following sections before upgrading the driver:

- "Understanding Identity Manager 4.8 Engine Support for Driver Versions" on page 198
- "Manually Removing the MapDB Cache Files" on page 198

Understanding Identity Manager 4.8 Engine Support for Driver Versions

Review the following considerations before upgrading an Identity Manager driver that uses MapDB:

- Drivers shipped with Identity Manager 4.8 are compatible with Identity Manager 4.8 Engine or Remote Loader. You must follow the driver upgrade steps from the specific driver implementation guide.
- Drivers shipped before Identity Manager 4.8 are not compatible with Identity Manager 4.8 Engine or Remote Loader.
- Drivers shipped with Identity Manager 4.8 are not backward compatible with Identity Manager 4.7.x Engine or Remote Loader.
- Drivers shipped with Identity Manager 4.8 are not backward compatible with Identity Manager 4.6.x Engine or Remote Loader.

Manually Removing the MapDB Cache Files

The Identity Manager Engine upgrade process leaves some of the existing MapDB cache files (`dx*`) in the Identity Vault's DIB directory (`/var/opt/novell/eDirectory/data/dib`). You must manually remove these files for your driver after upgrading the driver. This action ensures that your driver works correctly with Identity Manager 4.8 engine.

The following table lists the MapDB cache files that must be removed:

Identity Manager Driver	MapDB State Cache File To Remove
Data Collection Services	DCSDriver_<driver instance guid>-*
	<driver instance guid>-*
JDBC	jdbc_<driver instance guid>_*
LDAP	ldap_<driver instance guid>*
Managed System Gateway	MSGW-<driver-instance-guid>.*
Office 365 and Azure Active Directory	<Azure driver name>_obj.db.*
Salesforce	<Salesforce driver name>.*
	<Salesforce driver name>

where * represents the version of the MapDB state cache file. Any MapDB state cache files containing the above names, regardless of the versions, must be removed. In case of a Salesforce driver, the MapDB state cache files are also represented by the driver name. Below are some examples of these files.

- DCSDriver_<driver instance guid>-0.t, <driver instance guid>-1.p
- jdbc_<driver instance guid>_0.t, jdbc_<driver instance guid>_0, jdbc_<driver instance guid>_1
- ldap_<driver instance guid>b, ldap_<driver instance guid>b.p
- MSGW-<driver instance guid>.p, MSGW-<driver instance guid>.t
- <Azure driver name>_obj.db.t, <Azure driver name>_obj.db.p
- <Salesforce driver name>.p, <Salesforce driver name>.t, Salesforce driver1

Upgrading the Identity Manager Engine as a Non-root User

You need to perform the following steps to install the Identity Manager Engine as a non-root user:

1. Update NICI.
2. Update eDirectory as a non-root user.
3. Update Identity Manager Engine as a non-root user.

Upgrading NICI

Before installing Identity Manager engine as a non-root user make sure that NICI 3.1 or later installed in your setup or else perform the following steps to install:

1 Run the following command to stop the eDirectory instance:

```
ndsmanage stopall
```

2 From the iso that you have mounted, navigate to the /IDVault/setup/ directory.

3 Run the following command:

```
rpm -ivh nici64-3.1.0-1.00.x86_64.rpm
```

Upgrading eDirectory as a Non-root User

Perform the following steps to upgrade eDirectory as a non-root user:

1 Log in as a non-root user.

2 From the `iso` that you have mounted, navigate to the `/IDVault/` directory.

3 Copy the `eDir_NonRoot.tar.gz` file to a non-root home directory.

4 Run the following command to extract the .tar.gz file.

```
tar -zxvf eDir_NonRoot.tar.gz
```

5 Restart eDirectory.

```
ndsmanage stopall
ndsmanage startall
```

Upgrading Identity Manager Engine

Perform this action only if you have installed Identity Manager engine as a non-root user.

1 Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.

2 Mount the downloaded `.iso`.

3 Run the following command:

```
./idm-nonroot-install.sh
```

4 Select **Identity Manager Engine** and press Enter.

5 Specify the non-root install location for Identity Manager engine. For example, `/home/user/eDirectory/`.

6 Specify y to complete the upgrade.

Upgrading the Remote Loader

If you are running the Remote Loader, you need to upgrade the Remote Loader files.

1 Create a backup of the Remote Loader configuration files.

2 Verify that the drivers are stopped. For more information, see "Stopping the Drivers" on page 204.

3 Stop the Remote Loader service or daemon for each driver.

```
rdxml -config path_to_configfile -u
```

4 Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.

5 Mount the downloaded `.iso`.

6 Run the following command:

```
./install.sh
```

7 Read through the license agreement.

8 Enter y to accept the license agreement.

9 Specify whether you want upgrade the Identity Manager components. The available options are **y** and **n**.

10 Select Remote Loader.

11 After the installation finishes, verify that your configuration files contain your environment's information.

12 (Conditional) If there is a problem with the configuration file, copy the backup file that you created in step 1. Otherwise, continue with the next step.

13 Start the Remote Loader service or daemon for each driver.

```
rdxml -config path_to_config_file
```

IMPORTANT: If your driver uses MapDB, manually remove the existing MapDB state cache files for the driver after upgrading the driver. This is required because Identity Manager engine upgrade process does not remove all of these files from the Identity Vault's DIB directory. For more information, see "Working with MapDB 3.0.5" on page 198.

Upgrading the Java Remote Loader

1 Create a backup of the Java Remote Loader configuration files.

2 Verify that the drivers are stopped. For more information, see "Stopping the Drivers" on page 204.

3 Stop the Remote Loader service or daemon for each driver.

```
dirxml_jremote -config path_to_configfile -u
```

4 Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.

5 Mount the downloaded `.iso`.

6 Navigate to the `/IDM/packages/java_remoteloader` directory.

7 Copy and replace the `dirxml_jremote_dev.tar.gz` file in your existing Java Remote Loader installed directory.

8 Based on the file present in your existing setup, copy and replace one of the following files in your existing Java Remote Loader installed directory:

- `dirxml_jremote.tar.gz`
- `dirxml_jremote_mvs.tar`

9 Extract the files that you have copied in step 7 and step 8.

For example, `tar -zxvf dirxml_jremote.tar.gz`

10 (Conditional) If there is a problem with the configuration file, copy the backup file that you created in step 1. Otherwise, continue with the next step.

NOTE: Use the `version.txt` file to ensure that you have the latest version of Java Remote Loader.

---

11 Start the Remote Loader service or daemon for each driver.

```
dirxml_jremote -config path_to_config_file
```

Upgrading iManager

The upgrade process for iManager uses the existing configuration values in the `configiman.properties` file, such as port values and authorized users. Before upgrading iManager to the 3.2 version, NetIQ recommends that you:

- Upgrade eDirectory to the 9.2 version.
- Back up the `server.xml` and `context.xml` configuration files.

The upgrade process includes the following activities:

- "Upgrading iManager" on page 202
- "Updating Role-Based Services" on page 202
- "Re-installing or Migrating Plug-ins for Plug-in Studio" on page 203
- "Updating iManager Plug-ins after an Upgrade or Re-installation" on page 203

Upgrading iManager

Before upgrading iManager, ensure that the computer meets the prerequisites and system requirements.

NOTE: The upgrade process uses the HTTP port and SSL port values that were configured in the previous version of iManager.

---

1. Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads Website.
2. Mount the downloaded `.iso`.
3. Run the following command:

   ```
   ./install.sh
   ```

4. Read through the license agreement.
5. Enter y to accept the license agreement.
6. Specify iManager to proceed with the upgrade.

After upgrading iManager, you must perform the following steps:

- On RHEL servers, you must reboot the system.
- On SLES servers, you must manually remove Tomcat 8 from the following location and then reboot the system: `/etc/systemd/system/multi-user.target.wants/novell-tomcat8-service.service`

Updating Role-Based Services

NetIQ recommends that you update your RBS modules to the latest version so that you can see and use all of the available functionality in iManager.

NOTE: ◆When updating or re-installing iManager, the installation program does not update existing plug-ins. To update plug-ins manually, launch iManager and navigate to **Configure** > **Plug-in Installation** > **Available Novell Plug-in Modules**.

- Different installations of iManager might have a different number of plug-ins locally installed. As a result, you might see discrepancies in the module report for any given collection from the **Role Based Services** > **RBS Configuration** page. For the numbers to match between iManager installations, ensure that you install the same subset of plug-ins on each iManager instance in the tree.

To check for and update outdated RBS objects:

1. Log in to iManager.
2. In the Configure view, select **Role Based Services > RBS Configuration**.

   Review the table in the 2.*x* Collections tabbed page for any out-of-date modules.
3. To update a module, complete the following steps:
   - **3a** For the Collection that you want to update, select the number in the **Out-Of-Date** column.

     iManager displays the list of outdated modules.
   - **3b** Select the module you that want to update.
   - **3c** Click **Update** at the top of the table.

Re-installing or Migrating Plug-ins for Plug-in Studio

You can migrate or replicate Plug-in Studio plug-ins to another iManager instance, as well as to a new or updated version of iManager.

1. Log in to iManager.
2. In the iManager Configure view, select **Role Based Services > Plug-in Studio.**

   The Content frame displays the Installed Custom Plug-ins list, including the location of the RBS collection to which the plug-ins belong.
3. Select the plug-in that you want to re-install or migrate, then click **Edit**.

   NOTE: You can edit only one plug-in at a time.
4. Click **Install**.
5. Repeat these steps for every plug-in that you need to re-install or migrate.

Updating iManager Plug-ins after an Upgrade or Re-installation

When you upgrade or re-install your iManager, the installation process does not update the existing plug-ins. Ensure that the plug-ins match the correct iManager version.

NOTE: This is the only method for updating Identity Manager plug-ins from iManager on Open Enterprise Server 2018.

1 Open iManager.
2 Navigate to **Configure > Plug-in Installation > Available Novell Plug-in Modules**.
3 Update the plug-ins.

Stopping and Starting Identity Manager Drivers

You might need to start or stop the Identity Manager drivers to ensure that an upgrade, migration, or an installation process can modify or replace the correct files. This section explains the following activities:

- "Stopping the Drivers" on page 204
- "Starting the Drivers" on page 205

Stopping the Drivers

Before you modify any files for a driver, it is important to stop the drivers.

- "Using Designer to Stop the Drivers" on page 204
- "Using iManager to Stop the Drivers" on page 204

Using Designer to Stop the Drivers

1 In Designer, select the Identity Vault object in the **Outline** tab.
2 In the Modeler toolbar, click the **Stop All Drivers** icon .

   This stops all drivers that are part of the project.
3 Set the drivers to manual start to ensure that the drivers do not start until the upgrade process is complete:
   - **3a** Double-click the driver icon in the **Outline** tab.
   - **3b** Select **Driver Configuration > Startup Options**.
   - **3c** Select **Manual**, then click **OK**.
   - **3d** Repeat Step 3a through Step 3c for each driver.

Using iManager to Stop the Drivers

1 In iManager, select **Identity Manager > Identity Manager Overview**.
2 Browse to and select the location in the tree to search for Driver Set objects, then click the search icon .
3 Click the Driver Set object.
4 Click **Drivers > Stop all drivers**.
5 Repeat Step 2 through Step 4 for each Driver Set object.

6 Set the drivers to manual start to ensure that the drivers do not start until the upgrade process is complete:

   6a In iManager, select **Identity Manager > Identity Manager Overview**.

   6b Browse to and select the location in the tree to search for Driver Set objects, then click the search icon ▸.

   6c Click the Driver Set object.

   6d In the upper right corner of the driver icon, click **Edit properties**.

   6e On the Driver Configuration page under **Startup Options**, select **Manual**, then click **OK**.

   6f Repeat Step 6a through Step 6e for each driver in your tree.

Starting the Drivers

After all of the Identity Manager components are updated, restart the drivers. NetIQ recommends that you test the drivers after they are running to verify that all of the policies still work.

- "Using Designer to Start the Drivers" on page 205
- "Using iManager to Start the Drivers" on page 205

Using Designer to Start the Drivers

1 In Designer, select the Identity Vault object in the **Outline** tab.

2 Click the **Start All Drivers** icon in the Modeler toolbar. This starts all of the drivers in the project.

3 Set the driver startup options:

   3a Double-click the driver icon in the **Outline** tab.

   3b Select **Driver Configuration > Startup Option**.

   3c Select **Auto start** or select your preferred method of starting the driver, then click **OK**.

   3d Repeat Step 3a through Step 3c for each driver.

4 Test the drivers to verify the policies are working as designed. For information on how to test your policies, see "Testing Policies with the Policy Simulator" in *NetIQ Identity Manager - Using Designer to Create Policies*.

Using iManager to Start the Drivers

1 In iManager, select **Identity Manager > Identity Manager Overview**.

2 Browse to and select the location in the tree to search for Driver Set objects, then click the search icon ▸.

3 Click the Driver Set object.

4 Click **Drivers > Start all drivers** to start all of the drivers at the same time.

   or

   In the upper right corner of the driver icon, click **Start driver** to start each driver individually.

5 If you have multiple drivers, repeat Step 2 through Step 4.

6 Set the driver startup options:

   6a In iManager, select **Identity Manager > Identity Manager Overview**.

   6b Browse to and select the location in the tree to search for Driver Set objects, then click the search icon ▸.

   6c Click the Driver Set object.

   6d In the upper right corner of the driver icon, click **Edit properties**.

   6e On the Driver Configuration page, under **Startup Options**, select **Auto start** or select your preferred method of starting the driver, then click **OK**.

   6f Repeat Step 6b through Step 6e for each driver.

7 Test the drivers to verify the policies are working as designed.

   There is no policy simulator in iManager. To test the policies, cause events to happen that make the policies execute. For example, create a user, modify a user, or delete a user.

Upgrading the Identity Manager Drivers

NetIQ delivers new driver content through **packages**. You manage, maintain, and create packages in Designer. Although iManager is package-aware, Designer does not maintain any changes to driver content that you make in iManager. For more information about managing packages, see "Understanding Packages" in the *NetIQ Designer for Identity Manager Administration Guide*.

You can upgrade your drivers to packages in the following ways:

- "Creating a New Driver" on page 206
- "Replacing Existing Content with Content from Packages" on page 206
- "Keeping the Current Content and Adding New Content with Packages" on page 207

Creating a New Driver

The simplest and cleanest way to upgrade drivers to packages is to delete your existing driver and create a new driver with packages. Add all the functionality you want in the new driver. The steps are different for each driver. For instructions, see the individual driver guides on the Identity Manager Drivers documentation website. The driver now functions as before, but with content from packages instead of from a driver configuration file.

Replacing Existing Content with Content from Packages

If you need to keep the associations created by the driver, you do not need to delete and re-create the driver. You can keep the associations and replace the driver content with packages.

To replace the existing content with content from packages:

1 Create a backup of the driver and all of the customized content in the driver.

   For instructions, see "Exporting the Driver Configuration" on page 192.

2 In Designer, delete all objects stored inside of the driver. Delete the policies, filters, entitlements, and all other items stored inside of the driver.

NOTE: Designer provides the auto-import facility for importing the latest packages. You do not need to manually import the driver packages into the package catalog.

For more information, see "Importing Packages into the Package Catalog" in the *NetIQ Designer for Identity Manager Administration Guide*.

3 Install the latest packages to the driver.

These steps are specific for each driver. For instructions, see each driver guide at the Identity Manager Drivers documentation website.

4 Restore any custom policies and rules to the driver. For instructions, see "Restoring Custom Policies and Rules to the Driver" on page 222.

Keeping the Current Content and Adding New Content with Packages

Before you install a package, create a backup of the driver configuration file. When you install a package, it can overwrite existing policies, which might cause the driver to stop working. If a policy is overwritten, you can import the backup driver configuration file and recreate the policy.

Before you begin, make sure that any customized policies have different policy names than the default policies. When a driver configuration is overlaid with a new driver file, the existing policies are overwritten. If your custom policies do not have a unique name, you will lose them.

To add new content to the driver with packages:

1 Create a backup of the driver and all of the customized content in the driver.

For instructions, see "Exporting the Driver Configuration" on page 192.

NOTE: Designer provides the auto-import facility for importing the latest packages. You do not need to manually import the driver packages into the package catalog.

For more information, see "Importing Packages into the Package Catalog" in the *NetIQ Designer for Identity Manager Administration Guide*.

2 Install the packages on the driver.

For instructions, see each driver guide at the Identity Manager Drivers documentation website.

3 Add the desired packages to the driver. These steps are specific for each driver.

For more information, see the Identity Manager Drivers documentation website.

The driver contains the new functionality added by the packages.

Upgrading Identity Applications

This section provides information about upgrading Identity Applications and supporting software, which includes updating the following components:

- Identity Manager User Application
- Self-Service Password Reset (SSPR)
- Tomcat, JDK, and ActiveMQ

- PostgreSQL database
- One SSO Provider (OSP)

IMPORTANT: Identity Manager 4.8 requires Identity Applications and OSP installed on the same computer. When upgrading to this version, use OSP that is installed when Identity Applications are upgraded and then copy the OSP settings from your existing OSP server to the new OSP server. For more information, see "Post-Upgrade Tasks for Identity Applications Components" on page 214.

This section provides information about the following topics:

- "Considerations for Upgrade" on page 208
- "System Requirements" on page 209
- "Understanding the Upgrade Program" on page 209
- "Upgrading PostgreSQL" on page 209
- "Upgrading the Identity Applications Components" on page 210
- "Post-Upgrade Tasks for Identity Applications Components" on page 214
- "Verifying the Version Numbers After Upgrade" on page 217

Considerations for Upgrade

The Identity Applications upgrade process can vary based on how you want to upgrade the identity applications components. The following considerations apply before you upgrade Identity Applications:

- Before you begin with the upgrade process for Identity Applications, you must create the `igaworkflowdb` if you are using Oracle or MS SQL databases and assign all the privileges to the `idmadmin` user to own the databases.
- If your Identity Applications and SSPR are installed on different servers, you can choose to upgrade SSPR separately.
- During upgrade, ensure to verify and then modify the database name and other default values, as appropriate.
- Identity Manager supports a local installation of OSP on the Identity Applications server. The upgrade program does not support a standalone upgrade of OSP to this version and installs a new copy of OSP while upgrading Identity Applications. To restore your existing OSP settings to the newly installed OSP, see One SSO Provider in the "Post-Upgrade Tasks for Identity Applications Components" on page 214.

Table 10-1 *Upgrade Process for Identity Applications*

Identity Applications Deployment	Upgrade Process
Identity Applications, SSPR, and OSP are installed on the same server	To upgrade all the components, follow the steps from "Upgrading Identity Applications" on page 210.

Identity Applications Deployment	Upgrade Process
Identity Applications and OSP are installed on the same server. SSPR is installed on a different server.	1. To upgrade Identity Applications and OSP, follow the steps from "Upgrading Identity Applications" on page 207. 2. To upgrade SSPR on a different server, follow the steps from "Upgrading SSPR" on page 214.
Identity Applications are installed on a different server than SSPR and OSP. In this case, SSPR can be installed on the Identity Applications server or a separate server. However, OSP must be installed on the Identity Applications server.	1. To upgrade Identity Applications and OSP, follow the steps from "Upgrading Identity Applications" on page 207. 2. To upgrade SSPR on a different server, follow the steps from "Upgrading SSPR" on page 214. 3. Launch configuration update utility and provide details of the new server where OSP is installed. In this case, the new server is the server where Identity Applications is installed. For more information, see "SSO Clients Parameters" on page 121.

System Requirements

The upgrade process creates a backup of the current configuration for the installed components. Ensure that your server has sufficient space to store the backup and additional free space available for upgrade. For more information, see the NetIQ Identity Manager Technical Information website (https://www.microfocus.com/en-us/products/netiq-identity-manager/specs).

Understanding the Upgrade Program

The upgrade process reads the configuration values from the existing components. This information includes `ism-configuration.properties`, `server.xml`, `SSPRConfiguration` and other configuration files. When you use these configuration files, the upgrade process internally invokes the upgrade program for the specified components. The upgrade program also creates a backup of the current installation.

Upgrading PostgreSQL

Perform the following steps to upgrade PostgreSQL:

1. Download and extract the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.
2. Navigate to the `/common/scripts` directory.
3. Run the following command:

   ```
   ./pg-upgrade.sh
   ```

4. Specify the following details to complete the installation:

Existing Postgres install location: Specify the location where PostgreSQL is installed. The default location is `/opt/netiq/idm/postgres`.

NOTE: Ensure that the postgres user has appropriate permissions to the `/opt/netiq/idm/postgres` directory.

Existing Postgres Data Directory: Specify the location of the PostgreSQL data directory. The default location is `/opt/netiq/idm/postgres/data`.

Existing Postgres Database Password: Specify the PostgreSQL password.

New Postgres Data Directory: Specify the new PostgreSQL data directory. For example, `/opt/netiq/idm/postgres_new/data`.

Upgrading the Identity Applications Components

- Upgrading the Driver Packages for Identity Applications
- Upgrading Identity Applications
- Upgrading SSPR

Upgrading the Driver Packages for Identity Applications

You must stop Tomcat and update the packages for the User Application Driver and Role and Resource Service drivers to the latest version. For information about upgrading packages to the latest version, see Upgrading Installed Packages of the *NetIQ Designer for Identity Manager Administration Guide*.

After upgrading the User Application driver packages, you must manually add the workflow templates package:

1. In Designer, navigate to the User Application driver > Properties.
2. Click Packages, then click the ✚.
3. Select the **Show only applicable package versions** check box.
4. Select the **Create Workflow Templates**.
5. Click **OK** and then click **Finish** to complete the installation.
6. Deploy the User Application driver.

IMPORTANT: If any Email notifications template is installed or upgraded as part of User Application Driver upgrade, then you need to deploy **Default Notification Collection** object.

Upgrading Identity Applications

The following procedure describes how to upgrade Identity Applications.

1. Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.
2. Mount the downloaded `.iso`.
3. Run the following command:

   ```
   ./install.sh
   ```

4 Read through the license agreement.

5 Enter y to accept the license agreement.

6 Specify whether you want to upgrade the Identity Applications. The available options are **y** and **n**.

7 If you proceed with the upgrade, specify the following details:

OSP Installation Folder for Backup

This applies only when you have OSP and Identity Applications on the same server.

Specify the OSP backup folder to store the OSP backup data.

SSPR Installation Folder

This applies only when you have SSPR and Identity Applications on the same server.

Specify the SSPR installation folder.

SSPR not found on system. Do you want to install & configure it?

This applies only when you have Identity Applications and SSPR on different servers.

If you select **y**, then SSPR will be installed on the same server as Identity Applications. You need to copy the existing customization settings to the new SSPR installed server.

- **SSPR Configuration Password**: Specify the SSPR configuration password.
- **One SSO Server DNS/IP Address**: Specify the IP address of the server where OSP is installed.
- **One SSO Server SSL Port**: Specify the OSP SSL port.

If you select **n**, then SSPR will not be installed and Identity Applications will be upgraded.

User Application Installation Folder

Specify the User Application installation folder.

Identity Applications One SSO Service Password

Specify the One SSO password. The specified password will update the Client Secrets for all the clients that you have configured in the Configuration Update utility. If required, you can reset the password for the respective clients from the Configuration Update utility. For more information, see "SSO Clients Parameters" on page 121.

Identity Applications Database JDBC jar file

Specify the database JAR file. For example, if you are using PostgresQL database and it is installed on the same server, the default location of the existing database jar file is `/opt/netiq/idm/postgres/postgresql-9.4.1212.jar`.

Create Schema for Identity Applications

Specify when you want to create database schema. The available options are **Now**, **Startup**, and **File**. The default option is **Now**.

NOTE: You must create `igaworkflowdb` if you are using msSQL or Oracle assigning the idmadmin user all the privileges for this database.

Identity Applications Database User Password

Specify the database user password.

Identity Applications Database Administrator Password

Specify the database administrator password.

8 Start Tomcat. If you opt to create the database schema immediately, select **Now**.

```
systemctl start netiq-tomcat.service
```

(Optional) During upgrade, in case you select the **Startup** or **Write to file** option for creating the database schema, you must perform the required steps for migration of data to the workflow database. The following sections provide details on the data migration when you are using the **Startup** or **Write to file** options.

9 Restart the NGINX service.

```
systemctl restart netiq-nginx.service
```

Database Startup

IMPORTANT: Follow the procedure in this section to migrate data to the workflow database using the workflow migration utility provided with the Identity Manager 4.8 Linux and Windows ISO files. However, an updated workflow migration tool is now available in the Identity Applications Tools Patch 1 package (1.0.0.0100). This patch adds a new workflow migration command utility with new options to assist you in faster migration. It also addresses a key issue with the database migration failure when migrating large databases. You can download this patch from the Software License and Download portal. We recommend that you use the latest workflow migration utility for a smooth database migration. For more information, see NetIQ Identity Manager 4.8 Identity Applications Tools Patch 1 Release Notes.

Perform the following steps to migrate data while using the **Startup** option:

1 Copy the `WorkflowMigration.zip` file from the `<location where you have mounted the ISO>/user_application/IDM_Tools/` directory to `/home` directory and unzip the file.

2 Copy the `jdbc.jar` (for example, `sqljdbc42.jar`) file to `/home/WorkflowMigration/WEB-INF/lib/` directory and rename the file as `jdbcDriver.jar`.

3 Stop the Roles and Resource Service driver.

4 Start Tomcat.

5 Run the following commands to export the data:

NOTE: Ensure that the database user has all the privileges to modify the database.

```
[postgres]  /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar
-e test.zip -surl jdbc:postgresql://ip:port/idmuserappdb -suser
idmadmin -spwd ***** -sdb postgres
```

```
[oracle] /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar -e
test.zip -surl jdbc:oracle:thin:@IP:1521:idmdb -suser idmadmin -spwd
***** -sdb oracle
```

```
[mssql] /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar -e
test.zip -surl "jdbc:sqlserver://IP:1433;DatabaseName=idmuserappdb" -
suser idmadmin -spwd ***** -sdb mssql
```

6 Run the following commands to import the data:

```
[postgres]  /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar
-i test.zip -durl jdbc:postgresql://ip:port/igaworkflowdb -duser
idmadmin -dpwd ***** -ddb postgres
```

```
[oracle]   /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar -
i test.zip -durl jdbc:oracle:thin:@IP:1521:igaworkflowdb -duser
idmadmin -dpwd ***** -ddb oracle
```

```
[mssql] /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar -i
test.zip -durl "jdbc:sqlserver://IP:1433;DatabaseName=igaworkflowdb" -
duser idmadmin -dpwd ***** -ddb mssql
```

7 Start the Roles and Resource Services driver.

Write to SQL File

Perform the following steps to migrate data while using the **Write to SQL** option:

1 Copy the `WorkflowMigration.zip` file from the `<location where you have mounted the ISO>/user_application/IDM_Tools/` directory to `/home` directory and unzip the file.

2 Copy the `jdbc.jar` (for example, `sqljdbc42.jar`) file to `/home/WorkflowMigration/WEB-INF/lib/` directory and rename the file as `jdbcDriver.jar`.

3 Execute the `ua_databaseschema.sql` and `wfe_databaseschema.sql` scripts using an admin tool such as pgAdmin and verify whether the schema is created properly.

NOTE: While executing the `ua_databaseschema.sql` and `wfe_databaseschema.sql` using pgAdmin tool, ensure to comment out the first line in both the SQL files. For example, `Starting Liquibase at Fri, 11 Oct 2019 15:59:26 IST (version 3.7.0 built at 2019-07-16 02:32:57)` line should be commented out before executing the SQL files.

---

4 Run the following commands to export the data:

NOTE: Ensure that the database user has all the privileges to modify the database.

```
[postgres]  /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar
-e test.zip -surl jdbc:postgresql://ip:port/idmuserappdb -suser
idmadmin -spwd ***** -sdb postgres
```

```
[oracle] /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar -e
test.zip -surl jdbc:oracle:thin:@IP:1521:idmdb -suser idmadmin -spwd
***** -sdb oracle
```

```
[mssql] /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar -e
test.zip -surl "jdbc:sqlserver://IP:1433;DatabaseName=idmuserappdb" -
suser idmadmin -spwd ***** -sdb mssql
```

5 Run the following commands to import the data:

```
[postgres]  /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar
-i test.zip -durl jdbc:postgresql://ip:port/igaworkflowdb -duser
idmadmin -dpwd ***** -ddb postgres
```

```
[oracle]   /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar -
i test.zip -durl jdbc:oracle:thin:@IP:1521:igaworkflowdb -duser
idmadmin -dpwd ***** -ddb oracle
```

```
[mssql] /opt/netiq/common/jre/bin/java -jar Workflow-Migration.jar -i
test.zip -durl "jdbc:sqlserver://IP:1433;DatabaseName=igaworkflowdb" -
duser idmadmin -dpwd ***** -ddb mssql
```

6 Start Tomcat.

Upgrading SSPR

Use this method when SSPR is installed on a different server than the identity applications server in an Advanced Edition.

This is the only method to upgrade SSPR in a Standard Edition.

To upgrade SSPR:

1 Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.

2 Mount the downloaded `.iso`.

3 From the root directory of the `.iso` file, navigate to the `sspr` directory.

4 Run the following command:

```
./install.sh
```

5 Read through the license agreement.

6 Enter `y` to accept the license agreement.

7 Specify **y** to upgrade SSPR.

8 Specify **Identity Vault Administrator Password** and complete the upgrade.

Post-Upgrade Tasks for Identity Applications Components

Perform the following tasks before starting to use Identity Applications:

- Manually delete the previous version of Tomcat and ActiveMQ services. For example, run the following commands:

  ```
  /etc/init.d/idmapps_tomcat_init
  ```

  ```
  /etc/init.d/idmapps_activemq_init
  ```

- You must manually restore the customized settings for Tomcat, SSPR, OSP, and Kerberos.
- A certificate with CN as Identity Applications should be present in the keystore (`idm.jks`) of the Identity Applications server. As part of enhanced Java security, now Identity Applications requires trusted certificate to communicate with OSP.
- Use the existing Identity Applications keystore file to import the signed certificate to idm.jks. For example:

  ```
  ./keytool -import -alias mycerts -keystore /opt/netiq/idm/apps/tomcat/
  conf/idm.jks -file /opt/certs/chap8.der
  ```

  NOTE: This step is required for upgrading 4.6.x to 4.8.

- If you are upgrading Identity Applications in a clustered environment, then you must perform the following steps after upgrading Identity Applications:
  - Navigate to the `/opt/netiq/idm/apps/tomcat/conf` directory and add the following line in the `Context` tag of the `context.xml` file:

    ```
    <Manager notifyListenersOnReplication="true"
    expireSessionsOnShutdown="false"
    className="org.apache.catalina.ha.session.DeltaManager"/>
    ```

  - Navigate to the `/opt/netiq/idm/apps/tomcat/conf` directory and add the following lines in the `Cluster` tag of the `server.xml` file:

    ```
    <Channel className="org.apache.catalina.tribes.group.GroupChannel">
                <Membership
    className="org.apache.catalina.tribes.membership.McastService"
                            address="228.0.0.4"
                            port="45564"
                            frequency="500"
                            dropTime="3000"/>
                <Receiver
    className="org.apache.catalina.tribes.transport.nio.NioReceiver"
                          address="auto"
                          port="5000"
                          selectorTimeout="100"
                          maxThreads="6"/>

                <Sender
    className="org.apache.catalina.tribes.transport.ReplicationTransmit
    ter">
                  <Transport
    className="org.apache.catalina.tribes.transport.nio.PooledParallelS
    ender"/>
                </Sender>
                <Interceptor
    className="org.apache.catalina.tribes.group.interceptors.TcpFailure
    Detector"/>
                <Interceptor
    className="org.apache.catalina.tribes.group.interceptors.MessageDis
    patchInterceptor"/>
                <Interceptor
    className="org.apache.catalina.tribes.group.interceptors.Throughput
    Interceptor"/>
              </Channel>

              <Valve
    ```

```
className="org.apache.catalina.ha.tcp.ReplicationValve"

filter=".*\.gif|.*\.js|.*\.jpeg|.*\.jpg|.*\.png|.*\.htm|.*\.html|.*
\.css|.*\.txt"/>

          <Deployer
className="org.apache.catalina.ha.deploy.FarmWarDeployer"
                    tempDir="/tmp/war-temp/"
                    deployDir="/tmp/war-deploy/"
                    watchDir="/tmp/war-listen/"
                    watchEnabled="false"/>

          <ClusterListener
className="org.apache.catalina.ha.session.ClusterSessionListener"/>
```

- If your database is configured over SSL, replace `ssl=true` with `sslmode=require` in the `server.xml` file from PATH located at `/opt/netiq/idm/apps/tomcat/conf/`.

  For example, change

  ```
  jdbc:postgresql://<postgres db>:5432/idmuserappdb?ssl=true
  ```

  to

  ```
  jdbc:postgresql://<postgres db>:5432/idmuserappdb?sslmode=require
  ```

Tomcat

- In a cluster environment, manually uncomment the `Cluster` tag in `server.xml` and copy `osp.jks` on to all nodes from the first node located at `/opt/netiq/idm/apps/osp_backup_<date>`.
- If you have customized keystore files, include the correct path in the new `server.xml` file.

SSPR

If Identity Applications and SSPR are deployed on different servers, and you choose to restore the existing SSPR customized settings to the new server where SSPR is installed, ensure that you modify the SSPR settings on the new SSPR server by using the ConfigUpdate utility. For more information, see "SSO Clients Parameters" on page 121.

One SSO Provider

If Identity Applications and OSP are deployed on different servers in your pre-upgrade setup, copy the existing OSP settings to the new server where OSP is installed (Identity Applications server), then run the `merge_jars` method from the installation kit on this server to restore your settings.

1. Stop Tomcat on the server where you upgraded Identity Applications. (OSP is installed with Identity Applications upgrade)
2. Restore the customization.
   - **2a** Navigate to the OSP installation directory in your existing OSP server and locate the `osp-custom-resource.jar` file.

For example, `/opt/netiq/backup_idm/osp/osp-extras/l10n-resources/osp-custom-resource.jar`.

2b Copy the `osp-custom-resource.jar` file to a location on the server where you upgraded Identity Applications.

2c Navigate to `<location where you have mounted the Identity_Manager_4.8_Linux.iso>/osp/scripts/merge_cust_loc.sh`.

This script contains `merge_jars` method that takes care of merging the existing customization with the newly installed OSP.

2d Open a command prompt and run the following command:

```
merge_jars ${IDM_BACKUP_FOLDER in the remote OSP server}/tomcat/lib/
osp-custom-resource.jar ${IDM 4.8_OSP_INSTALLED_HOME}/osp-extras/
l10n-resources/osp-custom-resource.jar)
```

For example:

```
merge_jars /opt/netiq/backup_idm/osp/osp-extras/l10n-resources/osp-
custom-resource.jar /opt/netiq/idm/apps/osp/osp-extras/l10n-
resources/osp-custom-resource.jar
```

where `backup_idm` directory contains OSP settings in the existing OSP server.

3 Start Tomcat on the new server where OSP is installed.

For updating other settings, see "SSO Clients Parameters" on page 121.

Kerberos

The upgrade utility creates a new Tomcat folder on your computer. If any of the Kerberos files such as `keytab` and `Kerberos_login.config` resided in the old Tomcat folder, copy those files to the new Tomcat folder from the backed-up folder.

Verifying the Version Numbers After Upgrade

After upgrading to Identity Manager 4.8, verify that the components are upgraded to the following versions:

- Tomcat – `9.0.22`
- ActiveMQ – `5.15.9`
- Java – `1.8.0_222`
- One SSO Provider – `6.3.6`
- Self-Service Password Reset – `4.4.0.3`

Upgrading Identity Reporting

Identity Reporting includes two drivers. Perform the upgrade in the following order:

NOTE: Ensure that your database is upgraded to a supported version.

1. Upgrade your database to a supported version. For information on upgrading PostgreSQL database, see "Upgrading PostgreSQL" on page 209.
2. Upgrade Sentinel Log Management for IGA. For more information, see "Upgrading Sentinel Log Management for IGA" on page 218.
3. Upgrade Identity Reporting. For more information, see "Upgrading Identity Reporting" on page 219.
4. Configure the Managed System Gateway driver. For more information, see "Configuring the Managed System Gateway Driver" on page 141.
5. Configure Data Collection. For more information, see Configuring Settings and Data Collection in the Administrator Guide to NetIQ Identity Reporting.

Considerations for Upgrade

During upgrade, ensure that you specify the correct location for the `postgresql-9.4.1212.jar` file. The default location is `/opt/netiq/idm/postgres/`. The database connection will fail in the following scenarios:

- if you provide the incorrect path
- if you provide the incorrect jar file
- if the firewall is enabled
- if the database does not accept connections from remote machines

Upgrading Sentinel Log Management for IGA

1. Download the `SentinelLogManagementForIGA8.2.2.0.tar.gz` from the NetIQ downloads Website.
2. Navigate to a directory where you want to extract the file.
3. Run the following command to extract the file.

   ```
   tar -zxvf SentinelLogManagementForIGA8.2.2.0.tar.gz
   ```

4. Navigate to the `SentinelLogManagementforIGA` directory.
5. To install SLM for IGA, run the following command:

   ```
   ./install.sh
   ```

6. Specify the language that you want to use for installation, then press `Enter`.
7. Enter `y` to accept the license agreement and complete the upgrade.

NOTE: After SLM for IGA is upgraded, manually import the latest collectors.

1. Navigate to the directory where you have extracted the `SentinelLogManagementForIGA8.2.2.0.tar.gz` file.

2. Navigate to the `/content/` directory.
3. Import and configure the collectors. For more information, see Installing and Configuring the Sentinel Collectors in *NetIQ Identity Manager - Configuring Auditing in Identity Manager*.

Upgrading Identity Reporting

1 Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.

2 Mount the downloaded `.iso`.

3 Run the following command:

```
./install.sh
```

4 Read through the license agreement.

5 Enter y to accept the license agreement.

6 Specify whether you want upgrade the Identity Manager components. The available options are **y** and **n**.

7 Select Identity Reporting to proceed with the upgrade.

8 Specify the following details:

OSP Installed: Specify if OSP is installed.

OSP Install Folder: Specify the backup installation folder for OSP.

Reporting Installation Folder for backup: Specify the Reporting Installation folder.

Create schema for Identity Reporting: Specify whether you want to create the schema for your database now or later. The available options are **Now**, **Startup**, and **File**.

Identity Reporting Database JDBC jar file: Specify the database JAR file for Identity Reporting. The default location of the existing database jar file is `/opt/netiq/idm/apps/postgres/postgresql-9.4.1212.jar`.

Identity Reporting Database user: Specify the name of the Reporting database user.

Identity Reporting Database account password: Specify the Reporting database password.

NOTE: After upgrading Identity Manager to 4.8,

- Data synchronization policy will not be visible in IDMDCS UI. If you are planning to create a new policy, you must remove the existing data synchronization policy in Sentinel server and create a new data synchronization policy using IDMDCS UI after configuring Identity Reporting.
- The `com.netiq.rpt.ssl-keystore.type` property in `ism-configuration.properties` file will retain the value (JKS/PKCS12) that was set prior to upgrade.

Post-upgrade Steps for Reporting

- In a distributed setup, after upgrading Identity Applications and Identity Reporting from 4.7.x to 4.8, perform the following steps:
  1. Run the following command to import the OSP certificate from the `idm.jks` file of the Identity Applications and place it in a new Java Keystore file:

```
/opt/netiq/common/jre/bin/keytool -importkeystore -srckeystore /
opt/netiq/idm/apps/tomcat/conf/idm.jks -srcstorepass novell-
destkeystore ./idm.jks -deststorepass novell -srcalias "cn=<user-
name>, o=<organization-name>" -destalias "cn=<user-name>" -noprompt
```

For example:

```
/opt/netiq/common/jre/bin/keytool -importkeystore -srckeystore /
opt/netiq/idm/apps/tomcat/conf/idm.jks -srcstorepass novell-
destkeystore ./idm.jks -deststorepass novell -srcalias "cn=sean,
o=novell" -destalias "cn=sean" -noprompt
```

2. Replace the existing Java Keystore file in the Identity Reporting server with this newly created keystore file and restart the Identity Reporting server.

- During upgrade, if you have selected **Database Schema** creation as **Startup** or **File**, ensure you do the following:
  1. Log in to Identity Reporting.
  2. Delete the existing datasource and report definitions from the Identity Reporting repository.
  3. Add the new Identity Manager Data Collection Services datasource.
- After upgrading Identity Reporting to 4.8, navigate to the `ism-configuration.properties` file located at `/opt/netiq/idm/apps/tomcat/conf/` directory and perform the following actions:
  - Change the value of the `com.netiq.rpt.landing.url` property as follows:

    ```
    com.netiq.rpt.landing.url = ${com.netiq.idm.osp.url.host}/idmdash/
    #/landing
    ```

  - Change the value of the `com.netiq.idmdcs.landing.url` property as follows:

    ```
    com.netiq.idmdcs.landing.url = ${com.netiq.idm.osp.url.host}/
    idmdash/#/landing
    ```

  - Specify the value for the `com.netiq.rpt.redirect.url` property in the following format: `https:<hostname>:<port>/path`

    For example, `com.netiq.rpt.redirect.url = https://192.168.0.1:8543/IDMRPT/oauth.html`

  After making the required changes, save the file and restart Tomcat.
- If your database is configured over SSL, replace `ssl=true` with `sslmode=require` in the `server.xml` file from PATH located at `/opt/netiq/idm/apps/tomcat/conf/`.

  For example, change

  ```
  jdbc:postgresql://<postgres db>:5432/idmrptdb?ssl=true
  ```

  to

  ```
  jdbc:postgresql://<postgres db>:5432/idmrptdb?sslmode=require
  ```

Verifying the Upgrade for Identity Reporting

1. Launch Identity Reporting.
2. Verify that old and new reports are being displayed in the tool.

3 Look at the **Calendar** to see whether your scheduled reports appear.

4 Ensure that the **Settings** page displays your previous settings for managed and unmanaged applications.

5 Verify that all other settings look correct.

6 Verify whether the application lists your completed reports.

Upgrading Analyzer

1 Download the `Identity_Manager_4.8_Linux_Analyzer.tar.gz` from the NetIQ download website.

2 Extract the `.zip` file to the directory that contains the Analyzer installation files, such as the plug-ins, uninstallation script, and other Analyzer files.

3 Restart Analyzer.

4 To verify that you successfully applied the new patch, complete the following steps:

   4a Launch Analyzer.

   4b Click **Help > About Analyzer**.

   4c Check whether the program displays the new version.

Adding New Servers to the Driver Set

When you upgrade or migrate Identity Manager to new servers, you must update the driver set information. This section guides you through the process. You can use Designer or iManager to update the driver set.

For example, if you are using iManager, perform the following steps:

1 In iManager, click to display the Identity Manager Administration page.

2 Click **Identity Manager Overview**.

3 Browse to and select the container that holds the driver set.

4 Click the driver set name to access the Driver Set Overview page.

5 Click **Servers > Add Server**.

6 Browse to and select the new Identity Manager server, then click **OK**.

Removing the Old Server from the Driver Set

After the new server is running all of the drivers, you can remove the old server from the driver set.

- "Using Designer to Remove the Old Server from the Driver Set" on page 222
- "Using iManager to Remove the Old Server from the Driver Set" on page 222
- "Decommissioning the Old Server" on page 222

Using Designer to Remove the Old Server from the Driver Set

1. In Designer, open your project.
2. In the Modeler, right-click the driver set, then select **Properties**.
3. Select **Server List**.
4. Select the old Identity Manager server in the **Selected Servers** list, then click the **<** to remove the server from the **Selected Servers** list.
5. Click **OK** to save the changes.
6. Deploy the change to the Identity Vault.

   For more information, see "Deploying a Driver Set to an Identity Vault" in the *NetIQ Designer for Identity Manager Administration Guide*.

Using iManager to Remove the Old Server from the Driver Set

1. In iManager, click to display the Identity Manager Administration page.
2. Click **Identity Manager Overview**.
3. Browse to and select the container that holds the driver set.
4. Click the driver set name to access the Driver Set Overview page.
5. Click **Servers > Remove Server**.
6. Select the old Identity Manager server, then click **OK**.

Decommissioning the Old Server

At this point, the old server is not hosting any drivers. If you no longer need this server, you must complete additional steps to decommission it:

1. Remove the eDirectory replicas from this server.

   For more information, see "Deleting Replicas" in the *NetIQ eDirectory Administration Guide*.
2. Remove eDirectory from this server.

Restoring Custom Policies and Rules to the Driver

After installing or upgrading to new packages for your drivers, you must restore any custom policies or rules to the driver after you overlay the new driver configuration file. If these policies have different names, they are still stored in the driver, but the links are broken and need to be reestablished.

- "Using Designer to Restore Custom Policies and Rules to the Driver" on page 223
- "Using iManager to Restore Custom Policies and Rules to the Driver" on page 223

Using Designer to Restore Custom Policies and Rules to the Driver

You can add policies into the policy set. You should perform these steps in a test environment before you move the upgraded driver to your production environment.

1. In the **Outline** view, select the upgraded driver, then click the **Show Policy Flow** icon .
2. Right-click the policy set where you need to restore the customized policy to the driver, then select **Add Policy > Copy Existing**.
3. Browse to and select the customized policy, then click **OK**.
4. Specify the name of the customized policy, then click **OK**.
5. Click **Yes** in the file conflict message to save your project.
6. After the Policy Builder opens the policy, verify that the information is correct in the copied policy.
7. Repeat Step 2 through Step 6 for each customized policy you need to restore to the driver.
8. Start the driver and test the driver.

   For more information on starting the driver, see <TBD - provide a link to the book where you will have the Starting the Drivers section>. For more information on testing the driver, see "Testing Policies with the Policy Simulator" in *NetIQ Identity Manager - Using Designer to Create Policies*.
9. After you verify that the policies work, move the driver to the production environment.

Using iManager to Restore Custom Policies and Rules to the Driver

Perform these steps in a test environment before you move the upgraded driver to your production environment.

1. In iManager, select **Identity Manager > Identity Manager Overview**.
2. Browse to and select the location in the tree to search for Driver Set objects, then click the search icon .
3. Click the Driver Set object that contains the upgraded driver.
4. Click the driver icon, then select the policy set where you need to restore the customized policy.
5. Click **Insert**.
6. Select **Use an existing policy**, then browse to and select the custom policy.
7. Click **OK**, then click **Close**.
8. Repeat Step 3 through Step 7 for each custom policy you need to restore to the driver.
9. Start the driver and test the driver.

   For information on starting the driver, see <TBD - provide a link to the book where you will have the Starting the Drivers section>. There is no policy simulator in iManager. To test the policies, cause events to happen that make the policies execute. For example, create a user, modify a user, or delete a user.
10. After you verify that the policies work, move the driver to the production environment.

11 Switching from Advanced Edition to Standard Edition

You should switch to Standard Edition only if you do not want any Advanced Edition functionality in your environment and want to scale down your Identity Manager deployment.

1. (Conditional) If you have already applied the Advanced Edition activation, remove the activation.
2. (Conditional) To switch to the Standard Edition evaluation mode, perform the following actions:
   - **2a** Navigate to the Identity Vault `dib` directory.

     `/var/opt/novell/eDirectory/data/dib`
   - **2b** Create a new file, name it `.idme`, and add 2 (numeric) to the file.
   - **2c** Restart Identity Vault.
   - **2d** Continue with Step 4.
3. (Conditional) If you have already purchased a Standard Edition activation, apply the activation.
4. Stop Tomcat.
5. Remove the following WAR files and Webapps folder from the `/opt/netiq/idm/apps/tomcat/webapps` directory:
   - `IDMProv*`
   - `IDMRPT*`
   - `dash*`
   - `idmdash*`
   - `landing*`
   - `rra*`
   - `rptdoc*`
6. Move the following existing folders to a backup directory:
   - `IDMReporting`
   - `UserApplication`
7. Copy the `ism-configuration.properties` file from `<install folder>/tomcat/conf` directory to a backup directory.
8. Install Identity Reporting. For more information, see Chapter 4, “Installing Identity Manager,” on page 65.
9. Start `configupdate.sh` from the `<reporting install folder>/bin` directory and specify values for the following parameters:

   Reporting tab: Specify the settings in the following sections:
   - ID Vault
   - Identity Vault User Identity

- Report Administrators
  - **Report Admin Role Container DN**. For example, `ou=sa,o=data`
  - **Report Administrators**. For example, `cn=uaadmin,ou=sa,o=data`

Authentication tab: Specify the settings in the following sections:

- Authentication Server
  - **OAuth server host identifier**. For example, IP address or DNS name of the authentication server such as `192.168.0.1`
  - **OAuth server TCP port**
  - **OAuth server is using TLS/SSL**
- Authentication Configuration
  - **OAuth keystore file**. For example, `/opt/netiq/idm/apps/osp/osp.jks`
  - **Key alias of key for use by OAuth**
  - **Key password of key for use by OAuth**
  - **Session Timeout (minutes)**. For example, 60 minutes.

SSO Clients tab: Specify the settings in the following sections:

- Reporting
  - **URL link to landing page**. For example, `http://192.168.0.1:8180/IDMRPT`
- Self Service Password Reset
  - **OAuth client ID**. For example, *sspr*
  - OAuth client secret For example, *<sspr client secret>*
  - **OSP OAuth redirect url**. For example, `http://192.168.0.1:8180/sspr/public/oauth`

For more information about Configuration Utility, see "Running the Identity Applications Configuration Utility" on page 103.

10 Save the changes and exit the Configuration Utility.

11 Start Tomcat.

VI Migrating Identity Manager Data to a New Installation

This section provides information on migrating existing data in Identity Manager components to a new installation. Most migration tasks apply to the Identity Applications. To upgrade Identity Manager components, see Part V, "Upgrading Identity Manager," on page 183. For more information about the difference between upgrade and migration, see "Understanding Upgrade Process" on page 186.

12 Preparing to Migrate Identity Manager

This section provides information to help you prepare for migrating your Identity Manager solution to the new installation.

Checklist for Performing a Migration

To perform a migration, NetIQ recommends that you complete the steps in the following checklist.

	Checklist Items
❑	1. Ensure that you have the latest installation kit to migrate your Identity Manager data.
❑	2. Upgrade eDirectory to the latest supported version for the Identity Vault. For more information, see "Upgrading the Identity Vault" on page 197.
❑	3. Add the eDirectory replicas that are on the current Identity Manager server to the new server. For more information, see "Migrating the Identity Manager Engine to a New Server" on page 232.
❑	4. Install Identity Manager on the new server. For more information, see "Planning to Install Identity Manager" on page 35.
❑	5. (Conditional) If any of the drivers in the driver set are Remote Loader drivers, upgrade the Remote Loader server for each driver. For more information, see "Upgrading the Remote Loader" on page 200.
❑	6. (Conditional) If you are running User Application on your old server, update the component and its drivers. For more information, see "Prerequisites" on page 231.
❑	7. Change the server-specific information for each driver. For more information, see "Copying the Server-specific Information in Designer" on page 233.
❑	8. (Conditional) If you are running User Application, update the server-specific information from the old server to the new server for User Application. For more information, see "Copying Server-specific Information for the Driver Set" on page 232.
❑	9. Update your drivers to the package format. For more information, see "Upgrading the Identity Manager Drivers" on page 206.
❑	10. (Conditional) If you have custom policies and rules, restore your customize settings. For more information, see "Restoring Custom Policies and Rules to the Driver" on page 222.
❑	11. Install Identity Reporting and associated drivers. For more information, see "Migrating Identity Reporting" on page 237.
❑	12. Remove the old server from the driver set. For more information, see "Removing the Old Server from the Driver Set" on page 221.
❑	13. Activate your upgraded Identity Manager solution. For more information, see Activating Identity Manager in *NetIQ Identity Manager Overview and Planning Guide*.

13 Migrating Identity Manager to a New Server

This section provides information for migrating from the User Application to the identity applications on a new server. You might also need to perform a migration when you cannot upgrade an existing installation. This section includes the following activities:

- "Prerequisites" on page 231
- "Preparing Your Designer Project for Migration" on page 231
- "Migrating the Identity Manager Engine to a New Server" on page 232
- "Copying Server-specific Information for the Driver Set" on page 232
- "Updating the User Application Drivers" on page 234
- "Migrating Identity Applications" on page 235
- "Migrating Identity Reporting" on page 237

Prerequisites

- Back up the directories and databases of your Identity Manager solution.
- Ensure that you have installed the latest versions of the Identity Manager components, except for the identity applications.

NOTE: To continue using your current User Application database, specify **Existing Database** in the installation program. For more information, see Chapter 4, "Installing Identity Manager," on page 65.

- Run a health check of the Identity Vault to ensure that the schema extends properly. Use TID 3564075 to complete the health check.
- Import your existing User Application drivers into Designer.

Preparing Your Designer Project for Migration

You must archive the Designer project. It represents the pre-migration state of the drivers.

Before you migrate the driver, you need to perform some setup steps to prepare the Designer project for migration.

NOTE: If you do not have an existing Designer project to migrate, create a new project by using **File > Import > Project (From Identity Vault)**.

---

1. Launch Designer.

2 (Conditional) If you have an existing Designer project that contains the User Application that you want to migrate, back up the project:

   2a Right-click the name of the project in Project view, then select **Copy Project**.

   2b Specify a name for the project, then click **OK**.

3 To update the schema for your existing project, complete the following steps:

   3a In the Modeler view, select the Identity Vault.

   3b Select **Live > Schema > Import**.

4 (Optional) To verify that the version number for Identity Manager is correct in your project, complete the following steps:

   4a In the Modeler view, select the Identity Vault and then click **Properties**.

   4b In the left navigation menu, select **Server List**.

   4c Select a server and then click **Edit**.

   The **Identity Manager version** field should show the latest version.

Migrating the Identity Manager Engine to a New Server

1 Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.

2 Mount the downloaded `.iso`.

3 Install Identity Manager Engine from the `.iso`.

```
./install.sh
```

4 Configure Identity Manager Engine.

```
configure.sh
```

5 Create a read-write replica of the driverset partition and data partition on the new server.

6 Add the new server to the designer project.

Copying Server-specific Information for the Driver Set

You must copy all server-specific information that is stored in each driver and driver set to the new server's information. This also includes GCVs and other data on the driver set that will not be there on the new server and need to be copied. The server-specific information is contained in:

- Global configuration values
- Engine control values
- Named passwords
- Driver authentication information
- Driver startup options
- Driver parameters
- Driver set data

You can do this in Designer or iManager. If you use Designer, it is an automated process. If you use iManager, it is a manual process. If you are migrating from an Identity Manager server earlier than 3.5 version to an Identity Manager server greater than or equal to 3.5, you should use iManager. For all other supported migration paths, you can use Designer.

- "Copying the Server-specific Information in Designer" on page 233
- "Changing the Server-specific Information in iManager" on page 233
- "Changing the Server-specific Information for the User Application" on page 234

Copying the Server-specific Information in Designer

This procedure affects all drivers stored in the driver set.

1. In Designer, open your project.
2. In the **Outline** tab, right-click the server, then select **Migrate**.
3. Read the overview to see what items are migrated to the new server, then click **Next**.
4. Select the target server from the list available servers, then click **Next**.

   The only servers listed are servers that are not currently associated with a driver set and are equal to or newer than the source server's Identity Manager version.
5. Select one of the following options:
   - **Make the target server active:** Copies the settings from the source server to the target server and disables the drivers on the source server. NetIQ recommends using this option.
   - **Keep the source server active:** Does not copy the settings and disables all drivers on the target server.
   - **Makes both target and source servers active:** Copies settings from the source server to the target server without disabling the drivers on the source or target servers. If this option is used, then you must disable the driver on either the source server or the destination server, until each driver successfully runs on the new server. NetIQ recommends that you do not cache events for the driver on two servers at one time.

     NOTE: When the driver is active on a server, it caches events. Disabling the driver stops the caching of events on the server and deletes the cache (. TA0) file.
6. Click **Migrate**.
7. Deploy the changed drivers to the Identity Vault.

   For more information, see "Deploying a Driver to an Identity Vault" in the *NetIQ Designer for Identity Manager Administration Guide*.
8. Start the drivers.

   For more information, see "Starting the Drivers" on page 205.

Changing the Server-specific Information in iManager

1. In iManager, click to display the Identity Manager Administration page.
2. Click **Identity Manager Overview**.
3. Browse to and select the container that holds the driver set.

4 Click the driver set name to access the Driver Set Overview page.

5 Click the upper right corner of the driver, then click **Stop driver**.

6 Click the upper right corner of the driver, then click **Edit properties**.

7 Copy or migrate all server-specific driver parameters, global configuration values, engine control values, named passwords, driver authentication data, and driver startup options that contain the old server's information to the new server's information. Global configuration values and other parameters of the driver set, such as max heap size, Java settings, and so on, must have identical values to those of the old server.

8 Click **OK** to save all changes.

9 Click the upper right corner of the driver to start the driver.

10 Repeat Step 5 through Step 9 for each driver in the driver set.

Changing the Server-specific Information for the User Application

You must reconfigure the User Application to recognize the new server. Run `configupdate.sh`.

1 Navigate to the configuration update utility located by default in the installation subdirectory of the User Application.

2 At a command prompt, launch the configuration update utility:

```
configupdate.sh
```

3 Specify the values as described in "Configuring the Settings for the Identity Applications" on page 103.

Updating the User Application Drivers

1 Upgrade the User Application driver and Roles and Resource driver packages. For more information, see Upgrading Installed Packages of the NetIQ Designer for Identity Manager Administration Guide.

NOTE: While upgrading the packages, ensure that you specify the details of the new Identity Applications server.

---

2 Deploy the drivers.

Deploying the Drivers for Identity Applications

1 Open the project in Designer and run the Project Checker on the migrated objects.

For more information, see "Validating Provisioning Objects" in the *NetIQ Identity Manager - Administrator's Guide to Designing the Identity Applications*. If validation errors exist for the configuration, you are informed of the errors. These errors must be corrected before you can deploy the driver.

2 In the **Outline** view, right-click the User Application driver.

3 Select **Deploy**.

4 Repeat this process for each User Application driver in the driver set. Once the user Application driver is deployed, repeat this process for Roles and Resources Service driver.

Migrating Identity Applications

Do not use case-sensitive collation for your database. Case-sensitive collation is not supported. The case-sensitive collation might cause duplicate key errors during migration. If a duplicate key error is encountered, check the collation and correct it, then re-install the identity applications. The only supported collation is SQL_Latin1_General_CP1_CI_AS.

Before you migrate Identity Applications, you must install the `libssl.so.1.0.0` and `libcrypto.so.1.0.0` libraries from the `/opt/netiq/common/openssl/lib64/` directory.

The migration of Identity Applications involves the following:

- "Migrating the Database to the New Server" on page 235
- "Installing Identity Applications On the New Server" on page 236

Migrating the Database to the New Server

If your User Application database is on PostgreSQL, perform the following steps:

1 Log in as `postgres` user to the server where PostgreSQL is installed.

```
#su - postgres
```

2 Export the data to a `.sql` file. Ensure that the Postgres user has full access to the directory where you want to export the file:

```
pg_dump -p <portnumber> -U <username> -d <dbname> -f <export location>
```

For example,

```
pg_dump -p 5432 -U postgres -d idmuserappdb -f /tmp/idmuserappdb.sql
```

3 Log in to the new server where you want to install PostgreSQL.

4 Install PostgreSQL.

4a Navigate to the location where you have mounted the `Identity_Manager_4.8_Linux.iso`.

4b Navigate to the `/common/packages/postgres/` directory.

4c Install PostgreSQL using the following command:

```
rpm -ivh netiq-postgresql-9.6.6-0.noarch.rpm
```

4d Associate the group to postgres user using the following command:

```
/usr/sbin/usermod -a -G postgres postgres
```

4e Change the postgres user's home directory path to `/opt/netiq/idm/postgres/` in the `/etc/passwd` file.

4e1 Navigate to the `/etc/` directory.

4e2 Edit the `passwd` file.

```
vi /etc/passwd
```

4e3 Change the home directory of the postgres user to `/opt/netiq/idm/postgres/`.

4f Log in as `postgres` user.

For example,

```
su - postgres
```

4g Create a `data` directory in the PostgreSQL installed location.

`mkdir -p <POSTGRES_HOME>/data`, where `<POSTGRES_HOME>` is `/opt/netiq/idm/postgres`

For example,

```
mkdir -p /opt/netiq/idm/postgres/data
```

4h Export the PostgreSQL home directory.

```
export PGHOME=<postgres home directory path>
```

For example,

```
export PGHOME=/opt/netiq/idm/postgres
```

4i Export the PostgreSQL password:

```
export PGPASSWORD=<enter the database password>
```

4j Initialize the database.

```
LANG=en_US.UTF-8 <POSTGRES_HOME>/bin/initdb -D <POSTGRES_HOME>/data
```

For example:

```
LANG=en_US.UTF-8 /opt/netiq/idm/postgres/bin/initdb -D /opt/netiq/idm/postgres/data
```

4k Navigate to the `/opt/netiq/idm/postgres/bin` directory.

4l Create a database for the following components:

```
$ createdb idmuserappdb
$ psql -s idmuserappdb
# create user idmadmin password 'somepassword';
# GRANT ALL PRIVILEGES ON DATABASE idmuserappdb TO idmadmin;
# ALTER DATABASE idmuserappdb OWNER TO idmadmin;
```

5 Import the data to the new PostgreSQL database.

5a Copy the file exported in step 2 to a location where postgres user has full access.

5b Execute the following command to import data to the PostgreSQL database.

```
psql -d <dbname> -U <username> -f <full path where the exported file is located> -W
```

For example,

```
psql -d idmuserappdb -U idmadmin -f /tmp/idmuserappdb.sql -W
```

Installing Identity Applications On the New Server

1 Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.

2 Mount the `.iso`.

3 Copy the contents of the iso to a different directory which has write access.

For example,

```
cp -rp /mnt /home
```

4 Edit the contents of the configuration file to skip the deployment of User Application and roles and Resources Service driver.

NOTE: By default, Identity Applications installation creates and deploys the drivers for Role and Resource Service and User Application.

4a Navigate to the `/mnt/user_application` directory.

4b Edit the `configure.sh` file.

```
vi configure.sh
```

4c Comment out the following line:

```
install_service_drivers "UA" "${ID_VAULT_ADMIN_LDAP}"
"${ID_VAULT_PASSWORD}" "${ID_VAULT_HOST}" ${ID_VAULT_LDAPS_PORT}
"cn=${ID_VAULT_DRIVER_SET},${ID_VAULT_DEPLOY_CTX}"
```

4d Save the `configure.sh` file.

5 Install Identity Applications from the `/mnt` directory.

```
./install.sh
```

6 Configure Identity Applications from the `/mnt` directory.

```
./configure.sh
```

7 Select **Custom configuration** and choose **No** for the following prompt:

```
Do you want to configure PostgreSQL database on current server?
```

8 Navigate to the configuration update utility located at `/opt/netiq/idm/apps/configupdate` directory and ensure that the configuration settings are correct:

```
./configupdate.sh
```

Migrating Identity Reporting

The migration of Identity Reporting involves the following:

- "Updating the Drivers for Identity Reporting" on page 237
- "Deploying the Drivers for Identity Reporting" on page 238
- "Migrating Your Existing Data to a New Database" on page 238
- "Setting up the New Reporting Server" on page 242
- "Creating the Data Synchronization Policy" on page 242

Updating the Drivers for Identity Reporting

1 Upgrade the Data Collection Services and Managed Services Gateway driver packages. For more information, see Upgrading Installed Packages of the NetIQ Designer for Identity Manager Administration Guide.

NOTE: While upgrading the packages, ensure that you specify the details of the new Identity Reporting server.

2 Deploy the drivers. For more information, see "Deploying the Drivers for Identity Reporting" on page 238.

3 (Conditional) If you are migrating from 4.5.x and desire to migrate the EAS data, perform the steps from "Migrating Your Existing Data to a New Database" on page 238.

Deploying the Drivers for Identity Reporting

1 Open the project in Designer and run the Project Checker on the migrated objects.

For more information, see "Validating Provisioning Objects" in the *NetIQ Identity Manager - Administrator's Guide to Designing the Identity Applications*. If validation errors exist for the configuration, you are informed of the errors. These errors must be corrected before you can deploy the driver.

2 In the **Outline** view, right-click the Data Collection Services driver.

3 Select **Deploy**.

4 Repeat this process for each Data Collection Services driver in the driver set. Once the Data Collection Service driver is deployed, repeat this process for Managed Service Gateway driver.

Migrating Your Existing Data to a New Database

You must create the required roles and table spaces to ensure there are no failures during migration.

Prepare the New PostgreSQL Database

1 Stop EAS to ensure that none of the events are sent to the EAS server.

2 Using iManager, stop the DCS driver:

2a Log in to iManager.

2b Stop the DCS driver.

2c Edit the driver properties to change the startup option to **Manual**.

This step ensures that the driver does not start automatically.

3 Run the following SQL commands to create the required roles, table space, and database using `PGAdmin`.

This step ensures there are no failures during migration.

3a Run the following commands to create the required roles:

```
CREATE ROLE esec_app
  NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE;

CREATE ROLE esec_user
  NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE;

CREATE ROLE admin LOGIN
  ENCRYPTED PASSWORD '<specify the password for admin>'
  NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE;
GRANT esec_user TO admin;

CREATE ROLE appuser LOGIN
  ENCRYPTED PASSWORD '<specify the password for appuser>'
  NOSUPERUSER INHERIT NOCREATEDB CREATEROLE;
GRANT esec_app TO appuser;

CREATE ROLE dbauser LOGIN
  ENCRYPTED PASSWORD '<specify the password for dbauser>'
  SUPERUSER INHERIT CREATEDB CREATEROLE;

CREATE ROLE idmrptsrv LOGIN
  ENCRYPTED PASSWORD '<specify the password for idmrptsrv>'
  NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE;
GRANT esec_user TO idmrptsrv;

CREATE ROLE idmrptuser LOGIN
  ENCRYPTED PASSWORD '<specify the password for idmrptuser>'
  NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE;

CREATE ROLE rptuser LOGIN
  ENCRYPTED PASSWORD '<specify the password for rptuser>'
  NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE;
GRANT esec_user TO rptuser;
```

3b (Conditional) Run the following command for creating table spaces:

```
 CREATE TABLESPACE sendata1
  OWNER dbauser
  LOCATION '<provide the location where table space has to be
created>';
```

For example,

```
CREATE TABLESPACE sendata1
  OWNER dbauser
  LOCATION '</opt/netiq/idm/apps/postgres/data>';
```

3c (Conditional) If you want to migrate the existing EAS data, NetIQ recommends that you run the following command to create a SIEM database:

```
CREATE DATABASE "SIEM"
  WITH OWNER = dbauser
       ENCODING = 'UTF8'
       TABLESPACE = sendata1
       CONNECTION LIMIT = -1;
```

3d Run the following command to create a Reporting database:

```
CREATE DATABASE "idmrptdb"
  WITH OWNER = dbauser
       ENCODING = 'UTF8'
       CONNECTION LIMIT = -1;
```

Exporting EAS Data

Perform the following actions only if you are currently running Identity Manager 4.5.x and want to migrate your existing EAS data to a SIEM database:

- "Exporting EAS Data" on page 240
- "Importing EAS Data into the New PostgreSQL Database" on page 240

Exporting EAS Data

1. Stop EAS to ensure that none of the events are sent to the EAS server.
2. Using iManager, stop the DCS driver:
   - **2a** Log in to iManager.
   - **2b** Stop the DCS driver.
   - **2c** Edit the driver properties to change the startup option to **Manual**.

     This step ensures that the driver does not start automatically.
3. Export the data from EAS database to a file:
   - **3a** Log in to the EAS user account:

     ```
     # su - novleas
     ```
   - **3b** Specify a location where the EAS user has full access, for example, `/home/novleas`.
   - **3c** Navigate to the PostgreSQL installation directory and execute the following commands:

     For example,

     ```
     export PATH=/opt/novell/sentinel_eas/3rdparty/postgresql/bin/:$PATH
     ```

     ```
     export LD_LIBRARY_PATH=/opt/novell/sentinel_eas/3rdparty/
     postgresql/lib/:$LD_LIBRARY_PATH
     ```
   - **3d** Export the data to a `.sql` file using the following command:

     ```
     ./pg_dump -p <portnumber> -U <username> -d <dbname> -f <export
     location>
     ```

     For example,

     ```
     ./pg_dump -p 15432 -U dbauser SIEM -f /home/novleas/SIEM.sql
     ```

Importing EAS Data into the New PostgreSQL Database

1. Stop EAS to ensure that none of the events are sent to the EAS server.
2. Using iManager, stop the DCS driver:
   - **2a** Log in to iManager.
   - **2b** Stop the DCS driver.

2c Edit the driver properties to change the startup option to **Manual**.

This step ensures that the driver does not start automatically.

3 Import the EAS data to the new PostgreSQL database:

3a Copy the exported `.sql` file to a location where the postgres user has full access. For example, `/opt/netiq/idm/postgres`

3b Execute the following command to import the EAS data to the PostgreSQL database.

```
psql -d <dbname> -U <username> -f <full path where the exported file is located>
```

For example,

```
psql -d SIEM -U postgres -f /opt/netiq/idm/apps/postgres/SIEM.sql
```

4 Check for any migration log errors and resolve them.

Exporting the Reporting Data

Perform the following actions only if you are currently running Identity Manager 4.6.x and want to migrate your existing reporting data to a new server:

- "Exporting the Reporting Data" on page 241
- "Importing the Data into the New Reporting Server" on page 241

Exporting the Reporting Data

1 Log in as `postgres` user to the server where PostgreSQL is installed.

```
#su - postgres
```

2 Export the data to a `.sql` file. Ensure that the Postgres user has full access to the directory where you want to export the file:

```
pg_dump -p <portnumber> -U <username> -d <dbname> -f <export location>
```

For example,

```
pg_dump -p 5432 -U dbauser -W idmrptdb -f /tmp/idmrptdb.sql
```

Importing the Data into the New Reporting Server

1 Log in as `postgres` user to the server where PostgreSQL is installed.

```
#su - postgres
```

2 Import the data to the new PostgreSQL database.

2a Copy the exported `.sql` file to a location where postgres user has full access.

2b Execute the following command to import data to the PostgreSQL database.

```
psql -d <dbname> -U <username> -f <full path where the exported file is located>
```

For example,

```
psql -d idmrptdb -U dbauser -f /tmp/idmrptdb.sql
```

3 Check for any migration log errors and resolve them.

Setting up the New Reporting Server

1. Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.
2. Mount the `.iso`.
3. From the ISO mounted location, run the following command:

   ```
   ./install.sh
   ```

4. Configure Identity Reporting.

   ```
   ./configure.sh
   ```

5. Select **Custom configuration** and choose **No** for the following prompts:

   ```
   Do you want to configure PostgreSQL database on current server?
   Do you want to install a new driverset?
   ```

   NOTE: By default, Identity Reporting installation creates and deploys the drivers for Managed Services Gateway and Data Collection Services.

6. Navigate to the configuration update utility located at `/opt/netiq/idm/apps/configupdate` directory and ensure that the configuration settings are correct:

   ```
   ./configupdate.sh
   ```

Creating the Data Synchronization Policy

After the reporting server is configured, you need to create the data synchronization policy for forwarding events from SLM for IGA to the reporting database.

VII Deploying Identity Manager on AWS EC2

This section explains the planning and implementation of Identity Manager on AWS cloud.

- Chapter 14, “Planning and Implementation of Identity Manager on AWS EC2,” on page 245
- Chapter 15, “Example Scenarios of Hybrid Identity Manager,” on page 263

14 Planning and Implementation of Identity Manager on AWS EC2

Identity Manager adds support for deploying the following Identity Manager components as services on Amazon Web Services (AWS) EC2:

- Identity Vault
- Identity Manager engine
- Identity Manager drivers and Remote Loader
- iManager
- Designer
- Identity Applications
- Identity Reporting

Identity Manager supports the following operating systems on AWS EC2:

- SUSE Linux Enterprise Server (SLES) 12.x
- Red Hat Enterprise Linux (RHEL) 8.x

Prerequisites

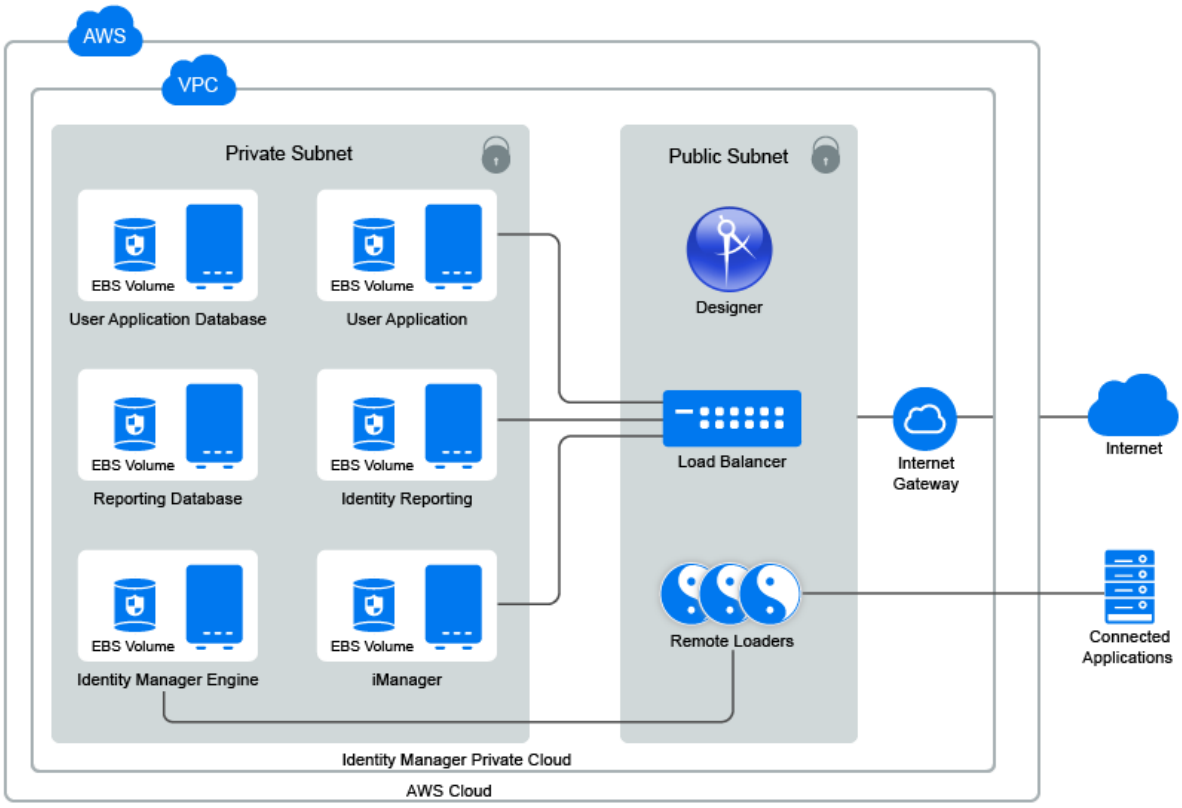
In addition to the system requirements of Identity Manager components, ensure that you meet the following prerequisites:

- An administrative account on AWS EC2.
- `Identity_Manager_4.8_Linux.iso` and Designer are downloaded, extracted, and available on Identity Manager component instances.
- An SSH client to connect to the AWS EC2 instances from your local client machine.

Deployment Procedure

Identity Manager components can be deployed on a private or a public network based on your requirement. Figure 14-1, “Identity Manager Deployment on AWS EC2,” on page 246 illustrates a sample deployment that is used in the subsequent sections.

Figure 14-1 *Identity Manager Deployment on AWS EC2*



Identity Manager components can be deployed in different combinations depending on how the components are distributed on different servers. However, the deployment procedure is the same for all scenarios.

The deployment procedure consists of the following steps:

- "Preparing AWS Virtual Private Cloud" on page 247
- "Creating and Deploying Instances" on page 249
- "Preparing the EC2 Instances" on page 250
- "Setting Up Identity Manager Components" on page 252
- "Setting Up Database for Identity Applications and Identity Reporting" on page 252
- "Setting Up Designer" on page 254
- "Creating an AWS EC2 Load Balancer" on page 254
- "(Optional) Creating Alias DNS with the Registered Hosted Zone" on page 259
- "Accessing Identity Manager Components" on page 260

Preparing AWS Virtual Private Cloud

This section outlines general steps to set up AWS VPC to use with Identity Manager. For more information, see the Amazon Elastic Compute Cloud Documentation.

Perform the following steps to create AWS VPC services:

1. Log in to the AWS Management Console.
2. Click **Services** and create the following services:

Service	Steps
VPC	1. Click **Services > VPC** under **Networking & Content Delivery**. 2. Click **Start VPC Wizard**. 3. Select a VPC configuration type and click **Select**. 4. Specify the details in the form, and then click **Create VPC**. This creates a private network of the specified size. VPC and subnet creation use the CIDR notation for address ranges. The largest VPC size is a /16 network. For more information, see the Amazon Virtual Private Cloud Documentation (https://docs.aws.amazon.com/vpc/index.html).
IMPORTANT: Creating a VPC using **Start VPC Wizard** creates Subnets, Internet gateways, and Route table for the VPC. You can view or edit these items as follows:	
Subnets	To deploy Identity Manager components as shown in Figure 14-1, create three subnets in VPC. For example, privateSN, publicSN1, and publicSN2. Perform the following steps to create a subnet. 1. In the left menu, click **Subnets**. 2. Click **Create Subnet**. 3. Specify **Name tag** to identify the subnet. 4. Specify **IPv4 CIDR block** within VPC. For example: `10.0.0.0/24` You must create public subnets in different availability zones. 5. Click **Yes, then click Create**. 6. (Conditional) For public subnets, enable auto-assign public IP address: a. Select **Subnet Actions > Modify auto-assign IP settings**. b. Select **Enable auto-assign public IPv4 address**. c. Click **Save**. Repeat these steps to create additional subnets.

Service	Steps
Internet gateways	1. In the left menu, click **Internet Gateways**. 2. Click **Create Internet Gateway**. 3. Specify **Name tag,** then click **Create**. 4. Select the newly created Internet gateway and attach it to VPC: a. Select **Actions > Attach to VPC**. b. Select the VPC from the list and click **Attach**.
Route table	1. In the left menu, click **Route Tables**. 2. Select the route table that was automatically created for this VPC. 3. In the **Routes** tab: a. Click **Edit**. b. Click **Add another route**. c. In **Destination**, specify `0.0.0.0/0`. d. In **Target**, select the Internet Gateway table that is associated with this VPC. See, **Internet gateways**. e. Click **Save**. 4. In the **Subnet Association** tab: a. Click **Edit**. b. Locate the subnet that you want to associate with this VPC and click **Save**.
(Optional) Hosted Zones	If you have a registered domain, you can use it to host Identity Manager components by performing the following actions: 1. Click **Services > Route 53 > Hosted Zones**. 2. Click **Create Hosted Zone**, specify the details such as: ◆ **Domain Name:** Specify the domain name. ◆ **Comment:** Add a comment. ◆ **Type:** Specify the type of the hosted zone. 3. Click **Create**.
Elastic IP address	1. Click **Services > EC2**. 2. In the left menu, select **Elastic IPs**. 3. Click **Allocate New address**. 4. Click **Allocate**. A static IPv4 address is allocated that is not used by any other resource. 5. Click **Close**.

Creating and Deploying Instances

This section outlines steps to create and deploy instances for a basic setup of Identity Manager, which includes the Identity Manager engine, iManager, Identity Applications, Reporting, User Application database, and Reporting database.

Perform the following steps to create instances for Identity Manager components.

1. Click **Services** > **EC2**.
2. Click **Launch Instance**.
3. Select the SLES 12 SPx or RHEL 7.x image.
4. Select the instance type that meets the requirements of the base operating system and deployment of Identity Manager components. See System Requirements.
5. Click **Next: Configure Instance Details**.

   Ensure that the instance is using the correct VPC and subnet. This page auto-populates the subnet settings.

Field	Action
Auto-assign Public IP	Set to `Enable` for the public. This setting automatically populates the subnet settings. For private subnet, set the value to `Disable`.

6. Click **Next: Add Storage**.

   The default storage size is 10 GB. Change the storage size as per your requirement. See System Requirements.
7. Click **Next: Add Tags**.

   Add tags as desired. Tags enable you to organize instances. For example, you can add the following two tags to each instance:

   - A tag indicating what the instance is being used for
   - A tag indicating who is the owner of this machine
8. Click **Next: Configure Security Group**.

   Security groups are virtual firewall rules for groups of instances. It is recommended to create a separate security group for each group of instances with the same firewall requirements.

   For example, you can configure a security group for all nodes of the Identity Manager engine, one security group for all nodes of Identity Applications, and one security group for all nodes of Identity Reporting. By default, a new security group only allows incoming traffic on port 22, so that you can only connect to the instance by using SSH.

   For more information, see Amazon EC2 Security Groups for Linux Instances (https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html).
9. Create a new security group; specify a name and description for it.

   Add additional port rules before installing the following Identity Manager components:

Component	Port	Description
LDAP for Identity Vault	TCP 636	Required for the secured LDAP communication.
iManager	TCP 8443	Required for the HTTPS communication to access iManager.
Identity Applications	TCP 8543	Required for the HTTPS communication to access Identity Applications.
Identity Reporting	TCP 8643	Required for the HTTPS communication to access Identity Reporting.
PostgreSQL Database	TCP 5432	Required for the secured database communication to access PostgreSQL.

10 Click **Review and Launch**.

11 After reviewing the details, click **Launch**.

12 Select an existing key pair or create a new one.

This key pair is used for SSH access to the instance. You can use the same key pair with multiple machines.

13 Click **Download Key Pair**.

IMPORTANT: You can connect to and manage your instances only using the private key. Therefore, do not lose the private key after downloading it.

14 Attach the Elastic IP address that is created when the instance is initializing.

15 Repeat Step 1 to Step 13 and create other instances.

Preparing the EC2 Instances

Launch an instance and verify the software repositories. To verify the configured software repositories, perform the following:

1 Log in to an instance using the key pair.

2 Switch to `root` user.

3 Verify that the following updates are available in your operating system:

SLES12-SP3-Pool and SLES12-SP3-Updates on SLES: To verify, run `zypper lr -n` command.

rhui-REGION-rhel-server-releases/7Server/x86_64 on RHEL: To verify, run `yum repolist` command.

NOTE: If repositories are not present in your operating system, verify that the configured elastic IP address is attached to the instance and then restart the instance.

4 Install the following prerequisites for your operating system:

SLES

Use `zypper` command to install `glibc-32bit` library.

Red Hat

Use `yum install` command to install the following prerequisites:

- `unzip`
- `ksh`
- `bc`
- `glibc-*.i686`
- `libXau-1.0.8-2.1.el7.i686`
- `libxcb.i686`
- `libX11.i686`
- `libXtst.i686`
- `libXrender.i686`
- `libgcc.i686`
- `lsof`

For Identity Manager engine, you can edit the `prerequisite.sh` script and remove the occurrences of `compat-libstdc++-33.x86_64`. This package is no longer necessary for Identity Manager installation.

5 Set up `/etc/hosts` and `hostname`:

5a Use the private IP address of the instance to secure Identity Manager servers within the firewall.

5b Assign a DNS name to the instance and update the `hosts` file.

For example:

```
# 10.0.0.1 identityEngine.example.com identityEngine
```

5c Set `hostname` and `domain name`.

SLES

```
yast lan
```

RHEL

```
hostnamectl set-hostname idmengine.example.com
```

6 (Conditional) Create an encrypted Elastic Block Store (EBS) volume to encrypt the data in the cloud.

6a Click **Services > EC2**.

6b In **Elastic Block Store**, select **Volumes** and click **Create Volume**.

6c Specify the required size for your volume.

6d Select **Encrypt this volume** and click **Create Volume**.

6e Select the newly created volume in the list.

6f In **Actions**, click **Attach Volume** to attach the volume to EC2 instance.

6g Repeat these steps for each instance.

For more information about EBS, see Amazon EBS.

7 Format the volume and mount the partition by using your operating system tools:

SLES

Run the `yast disk` command to format the volume.

RHEL

Run `mkfs` to format and add to `/etc/fstab`. For more information, see Red Hat Enterprise Linux Deployment Guide.

NOTE: ◆Mount Identity Manager engine data partition. By default, the data partition is `/var/opt/novell/`.

◆ Mount other Identity Manager components in `/opt/netiq/`.

---

8 Update the `/etc/hosts` file on all instances with DNS to IP address of all machines.

Setting Up Identity Manager Components

Before installing the Identity Manager components, perform the following steps:

1 Download the `Identity_Manager_4.8_Linux.iso` on the instance where you want to install the Identity Manager component.

2 Mount the downloaded `.iso` file.

3 (Conditional) Create the database for Identity Applications and Identity Reporting. For more information, see "Setting Up Database for Identity Applications and Identity Reporting" on page 252.

4 From the root directory of the `.iso` file, run `./install.sh` command.

5 Read through the license agreement and type y to accept the license agreement.

6 Select custom installation option and select the component that you want to install on the instance. Configure the component. For more information, see Table 5-2, "Custom Configuration," on page 79.

7 Run `configupdate.sh` on Identity Reporting and Identity Applications to set all clients.

Setting Up Database for Identity Applications and Identity Reporting

If you want to install the PostgreSQL database for Identity Applications and Identity Reporting on an external server, you should perform the following steps before installing:

1 Navigate to the location where you have mounted the `Identity_Manager_4.8_Linux.iso`.

2 Locate the `/common/packages/postgres/` directory and install PostgreSQL.

```
rpm -ivh netiq-postgresql-9.6.6-0.noarch.rpm
```

3 Associate the group to `postgres` user by running the following command:

```
/usr/sbin/usermod -a -G postgres postgres
```

4 Change the `postgres` user's home directory path to `/opt/netiq/idm/postgres/` in the `/etc/passwd` file.

4a Navigate to the `/etc/` directory.

4b Edit the `passwd` file.

```
vi /etc/passwd
```

4c Change the home directory of the `postgres` user to `/opt/netiq/idm/postgres/`.

5 Log in as `postgres` user.

For example:

```
su - postgres
```

6 Create a `data` directory in the PostgreSQL install location.

`mkdir -p <POSTGRES_HOME>/data`, where `<POSTGRES_HOME>` is `/opt/netiq/idm/postgres`

For example:

```
mkdir -p /opt/netiq/idm/postgres/data
```

7 Export the PostgreSQL home directory

```
export PGHOME=<postgres home directory path>
```

For example:

```
export PG_HOME=/opt/netiq/idm/postgres
```

8 Export the PostgreSQL password:

```
export PGPASSWORD=<enter the database password>
```

9 Initialize the database.

```
"LANG=en_US.UTF-8 <POSTGRES_HOME>/bin/initdb -D <POSTGRES_HOME>/data"
```

For example:

```
"LANG=en_US.UTF-8 /opt/netiq/idm/postgres/bin/initdb -D /opt/netiq/idm/postgres/data"
```

10 Create a database for the following components in the `/opt/netiq/idm/postgres/` directory.

Identity Applications

```
$ createdb idmuserappdb
$ psql -s idmuserappdb
# create user idmadmin password 'somepassword';
# GRANT ALL PRIVILEGES ON DATABASE idmuserappdb TO idmadmin;
# ALTER DATABASE idmuserappdb OWNER TO idmadmin;
```

Identity Reporting

```
$ createdb idmrptdb
```

11 Log out as `postgres` user.

12 Modify the `postgresql.conf` file to allow the PostgreSQL instance to listen on network instances other than `localhost`.

12a Navigate to the `/opt/netiq/idm/postgres/data/` directory.

12b Edit the `postgresql.conf` file:

```
vi postgresql.conf
```

12c Add the following line in the file:

```
listen_addresses = '*'
```

13 Create `pg_log` directory under `<postgres home directory path>/data`.

   For example:

   ```
   mkdir -p /opt/netiq/idm/postgres/data/pg_log
   ```

14 Change the permissions for the `pg_log` directory.

   ```
   chown -R postgres:postgres <postgres directory path>/data/pg_log
   ```

   For example:

   ```
   chown -R postgres:postgres /opt/netiq/idm/postgres/data/pg_log
   ```

15 Start the PostgreSQL service.

   ```
   systemctl start netiq-postgresql
   ```

   This will start the new PostgreSQL service.

Setting Up Designer

You must install Designer on a Windows machine to use it.

1 On a public subnet, launch a supported Windows instance.

   For the Windows security group, use `rdesktop` port only. For example, 3389

2 Install Designer on a Windows instance. For more information, see Installing Designer in *NetIQ Identity Manager Setup Guide for Windows*.

Creating an AWS EC2 Load Balancer

You can create a load balancer to balance the load of incoming requests across Identity Manager components. The load balancer can be used to secure the Identity Manager servers from public access.

The following procedures explain the configuration details required to set up a load balancer for the sample deployment scenario:

Create a Certificate for a Load Balancer to Use Secured Communication

The load balancer uses this certificate to establish a secured communication among Identity Manager components. You can create a certificate for a load balancer in three ways:

- "Using AWS Certificate Manager (ACM)" on page 254
- "Uploading an External Certificate to ACM" on page 255
- "Uploading an External Certificate to IAM" on page 255

Using AWS Certificate Manager (ACM)

1 Click **Services > Certificates Manager**.

2 Click **Request Certificate**.

3 Specify the DNS name for which you want to create a certificate.

4 Verify the DNS authority.

Uploading an External Certificate to ACM

1 Click **Services > Certificates Manager**.

2 Click **Import Certificate**.

3 Specify the certificate details.

Uploading an External Certificate to IAM

1 Click **Services**.

2 In **Security, Identity & Compliance**, click **IAM**.

3 Specify the certificate details using IAM API. For more information, see Uploading Server Certificate Using IAM API.

Creating Target Groups

A target group provides a way to associate the load balancer to the IP addresses of instances (targets) among which the load will be distributed.

Perform the following steps to create a target group:

1 In the EC2 Dashboard, click **Target Groups** under **LOAD BALANCING**.

2 Click **Create target group**.

3 Specify the following details:

Field	Description
Target group name	Specify a name for the target group. You can specify the name of a component for which this target group is configured. For example, Identity Applications, Identity Reporting, or iManager.
Protocol	Select **HTTPS**.
Port	Specify the port on which the server is configured for listening. Following are the example port values used for different Identity Manager Components: ◆ **Identity Applications:** 8543 ◆ **Identity Reporting:** 8643 ◆ **iManager:** 8443
Target type	Select **Instance**.
VPC	Select the same VPC that you have selected for the instances of Identity Manager components.
Health Check Settings	
Protocol	Select **HTTPS**. The load balancer uses this protocol while performing health checks.

Field	Description
Path	Specify the destination path for health checks. Following are the default paths of the Identity Manager components to perform health checks: ◆ **Identity Applications:** `/idmdash/index.html` ◆ **Identity Reporting:** `/IDMRPT/index.html` ◆ **iManager:** `/nps/login.html`
Advanced health check settings	Keep the default values.

4 Click Create.

5 Enable session stickiness.

5a Select the target group you have created.

5b In the Description tab, click Edit attributes.

5c Select Enable for Stickiness.

6 Repeat these steps to create target groups for each application.

NOTE: If SSPR is installed on a different server, create a separate target group for this component.

Create the Load Balancer

Perform the following steps to create a load balancer:

1 In the left menu, click Load Balancers.

2 Click Create Load Balancers.

3 Click Create under Application Load Balancer.

4 Specify the following details:

Field	Description
Name	Specify a name for the load balancer.
Scheme	Select internet-facing.

Field	Description
Listeners	To add more listeners to your load balancer, click **Add Listener**. Specify the listener ports as follows: **For iManager:** ◆ **Load Balancer Protocol**: HTTPS ◆ **Load Balancer Port**: 8443 **For Identity Applications:** ◆ **Load Balancer Protocol**: HTTPS ◆ **Load Balancer Port**: 8543 **For Identity Reporting:** ◆ **Load Balancer Protocol**: HTTPS ◆ **Load Balancer Port**: 8643
Availability Zones	1. Select the same **VPC** that you have created earlier for Identity Manager components. 2. Select the **Availability Zone** in which public subnets are available. **NOTE:** You must select at least two subnets.
Tags	(Optional) You can add a tag to identify your load balancer.

5 Click **Next: Configure Security Settings**.

6 Specify the certificate details to use HTTPS protocol. You can perform one of the following:

- Select the certificate type that you created in "Create a Certificate for a Load Balancer to Use Secured Communication" on page 254.
- Upload the certificate to IAM or ACM – Specify the certificate details.

7 Click **Next: Configure Security Groups**.

8 In **Assign a security group**, select **Create a new security group**.

9 (Optional) Specify the name and description for the load balancer.

10 Add rules to the security group that routes the traffic to the configured listeners:

Field	Description
Type	Select **Custom TCP Rule**.
Protocol	This displays the protocol type used for the rule.
Port Range	Select the port range for the Identity Manager Components: ◆ **iManager:** 8443 ◆ **Identity Applications:** 8543 ◆ **Identity Reporting:** 8643
Source	Select **Anywhere** to connect to the instance where the Identity Manager component is deployed.

11 Click **Next: Configure Routing**.

12 In **Target group**, specify the following details:

Field	Description
Target group	Select **Existing target group**. This list displays the target groups created for Identity Manager Components in "Creating Target Groups" on page 255.
Name	Select a target group from the list. You can select only one target group here. For example, select the target group that you have created for Identity Applications. After creating the load balancer, you will need to modify the listener port 8443 to use the target group that is configured for the HTTPS protocol. See Step 18 on page 258 of this section.
Protocol	Populated with the value that you have configured in the specified target group. Review to ensure that the value is listed correctly.
Port	Populated with the value that you have configured in the specified target group. Review to ensure that the value is listed correctly.
Target type	Populated with the value that you have configured in the specified target group. Review to ensure that the correct value is listed.

13 Under **Health Checks**, review the following details:

Field	Description
Protocol	Populated with **HTTPS** or **HTTP** based on the configuration of the target group you have selected in Step 12. See "Creating Target Groups" on page 255.
Path	Populated with the health URL that you have configured in the target group selected in Step 12. See "Creating Target Groups" on page 255.
Advanced health check settings	Keep the default values.

14 Click **Next: Register Targets**.

The list of all targets registered with the target group that is selected. You can modify this list only after creating the load balancer.

15 Click **Next: Review**.

16 Verify that the load balancer details are correct.

17 Click **Create** and then click **Close**.

18 (Conditional) If you skipped to create listeners ports for Identity Manager components or you want to add new listener ports, update the listener ports to use the appropriate target groups:

18a Select the load balancer you have created.

18b Select the **Listeners** tab.

18c Click **Add Listener** and specify the required details for each listener, see Step 4.

18d Select the certificate that is used for the load balancer, see "Create a Certificate for a Load Balancer to Use Secured Communication" on page 254.

18e Click **Create**.

If SSPR is configured on a separate machine, you might need to add a listener.

IMPORTANT: To use a single load balancer in a distributed setup, create a separate DNS alias record to differentiate the servers in the setup. Otherwise, create a separate load balancer for each web application.

(Optional) Creating Alias DNS with the Registered Hosted Zone

If you have a registered site, you can use it to create an individual record set for each Identity Manager component.

1. Click **Services > Route 53**.
2. In the left-side menu, click **Hosted Zones** and select the hosted zone that is created while setting up AWS EC2 services. See, "Preparing AWS Virtual Private Cloud" on page 247.
3. Click **Go to Record Sets**.
4. Click **Create Record Set**:

Field	Description
Name	Specify a meaningful name for your record set. For example: Name Identity Applications record set as `rbpm`.
Type	Select **A – IPv4 address**.
Alias	Select **Yes**.
Alias Target	Select the load balancer which is configured to connect Identity Manager components
Routing Policy	Select **Simple**

5. Click **Create**.
6. Repeat the Step 4 and Step 5 to create a record set for each Identity Manager instance.
7. Run `configupdate.sh` on Identity Applications, Identity Reporting, and OSP instances and update SSO configuration with the public DNS name.
8. Restart Tomcat.
9. Verify the configuration by accessing the applications using the public DNS.

```
https://<public-DNS-name>:<port>/<application-context-name>
```

Accessing Identity Manager Components

You can access the Identity Manager instances using the public DNS name of the load balancer or the alias DNS record set. To allow Identity Manager instances to communicate with one another, edit the `/etc/hosts` files on each instance and add an entry to resolve its hostname to its private IP address.

Update the following instances to internally access the other instances

Instance	Description

OSP	The OSP instance requires access to the SSPR instance to reset passwords.
	Host file location: `/etc/hosts`
	Modify the hosts file with the following entry:
	`<IP_address>  <Private_DNS_Name> <Public_DNS_Name>`
	For example:
	`10.0.1.5  sspr.privatedns.local sspr.publicdns.com`

Identity Applications	The Identity Applications instance requires an access to OSP instance for login purposes.
	Host file location: `/etc/hosts`
	Modify the hosts file with the following entry:
	`<IP_address>  <Private_DNS_Name> <Public_DNS_Name>`
	For example:
	`10.0.1.6  osp.privatedns.local osp.publicdns.com`

Identity Reporting	The Identity Reporting instance requires an access to OSP instance for login purposes.
	Host file location: `/etc/hosts`
	Modify the hosts file with the following entry:
	`<IP_address>  <Private_DNS_Name> <Public_DNS_Name>`
	For example:
	`10.0.1.6  osp.privatedns.local  osp.publicdns.comm`

Security Considerations

NetIQ recommends that you review the following considerations for deploying Identity Manager components on AWS cloud:

- Identity Manager components are configured on a private network with no public access or attached to an Elastic IP address.
- Web applications such as Identity Applications, Identity Reporting, or iManager are accessed through a load balancer.

- Identity Manager components are configured to use a secured communication channel.
- Data is configured on a separate encrypted EBS volume for each component.
- The following ports are available on the Identity Manager servers to use within the subnet.

Port	Application
636	LDAP
8543	Identity Applications
8643	Identity Reporting
5432	PostgreSQL
8443	iManager

15 Example Scenarios of Hybrid Identity Manager

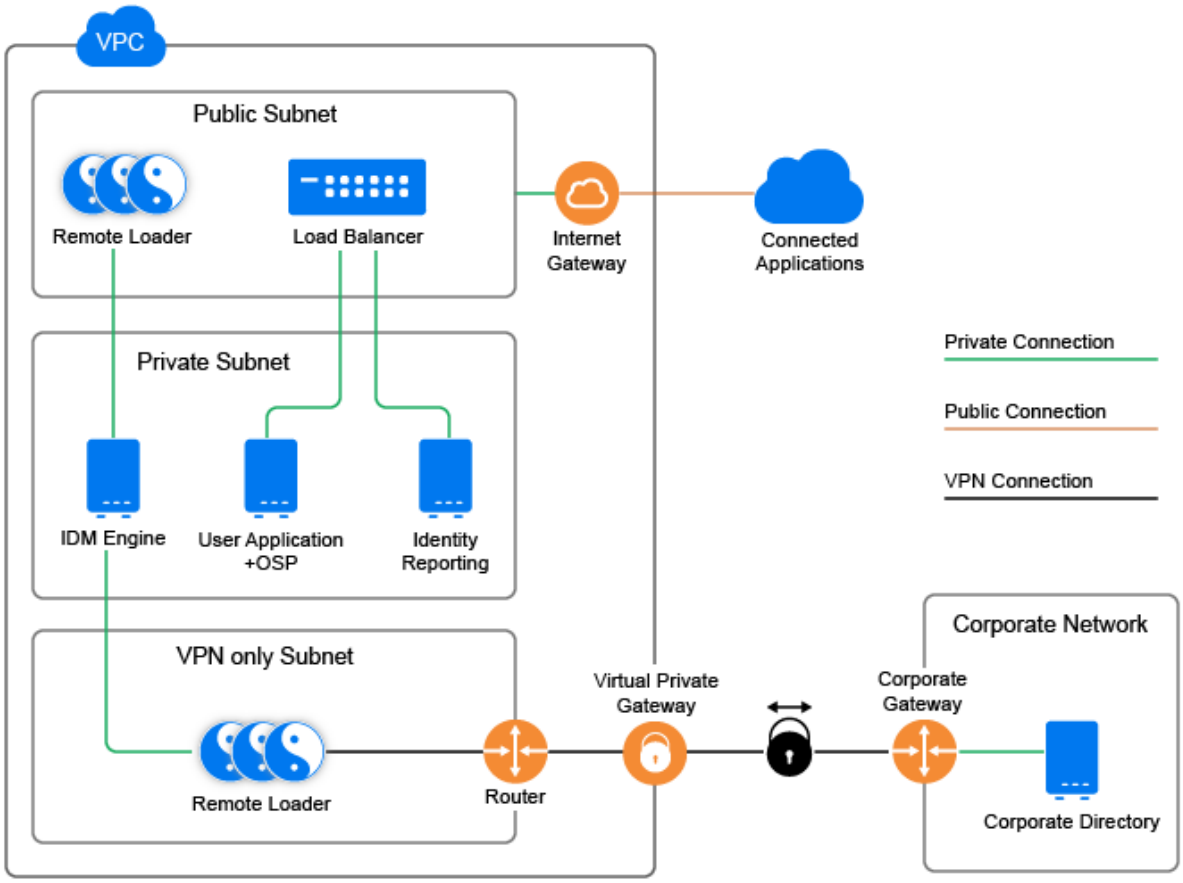
You can configure Identity Manager components where the identities are synchronized seamlessly between your enterprise premise and AWS cloud. Implementing this type of hybrid scenarios requires you to configure a VPN connection between AWS subnet and the enterprise network. This section explains the following hybrid Identity Manager scenarios:

- "Using Remote Loader Connection" on page 263
- "Using Multi-Server Driver Set Connection" on page 264
- "Using eDirectory Driver Connection" on page 266

Using Remote Loader Connection

Remote Loader is installed on a subnet where VPN is configured. When you enable the synchronization, the Remote Loader driver shim connects to the application that is running on the enterprise network and synchronizes the identities between Identity Manager on AWS cloud and the application.

Figure 15-1 Hybrid Scenario Using Remote Loader Connection



This scenario is suitable for systems with fewer connected applications and requires you to open a listener port for Remote Loader. The connection allows only configured attributes to pass during the synchronization.

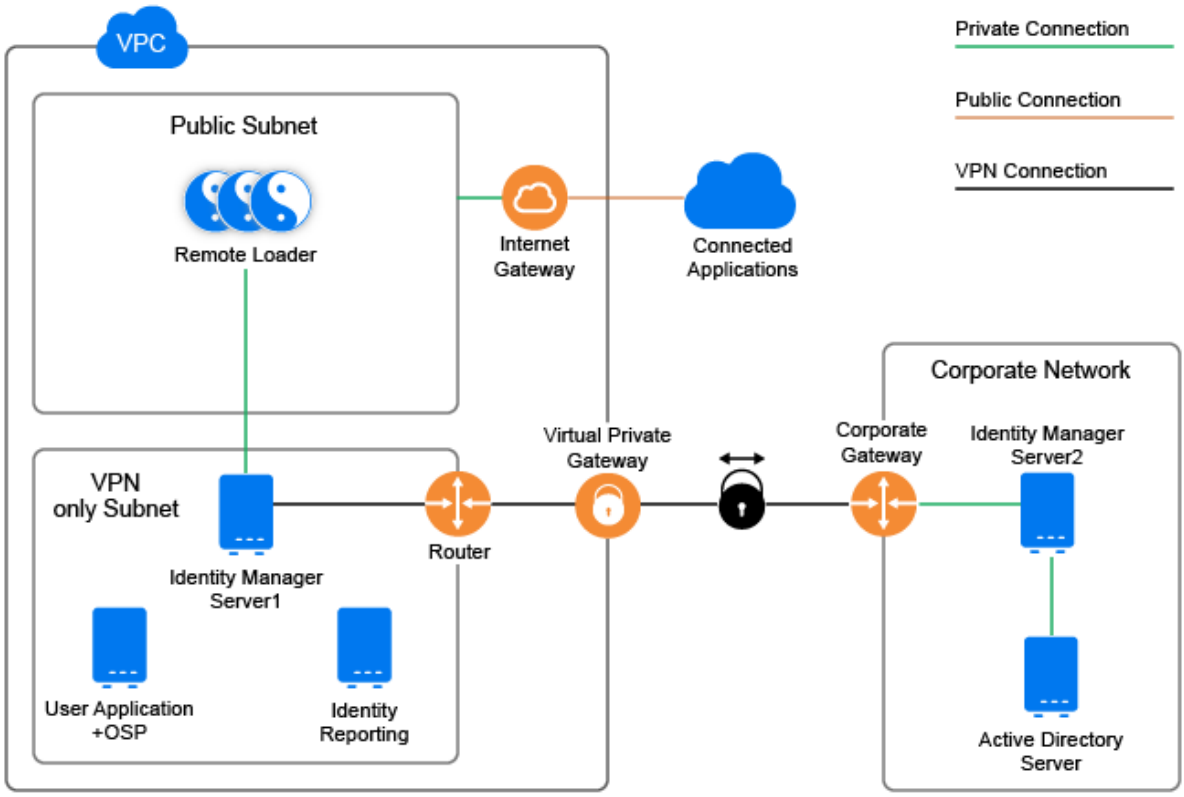
Limitations:

- This scenario applies to the drivers that support the use of Remote Loader.
- A large number of connected applications increase the traffic to Remote Loader.

Using Multi-Server Driver Set Connection

In this scenario, at least two Identity Manager servers use the same driver set where one server is installed on AWS cloud and the other server is installed on the enterprise premise. This includes full replica servers that use the Identity Vault replication channel to synchronize the identities through VPN connection. The Identity Manager server that is running on the enterprise network or AWS cloud synchronizes the identities across their respective connected applications.

Figure 15-2 Hybrid Scenario Using Multi-Server Driver Set Connection

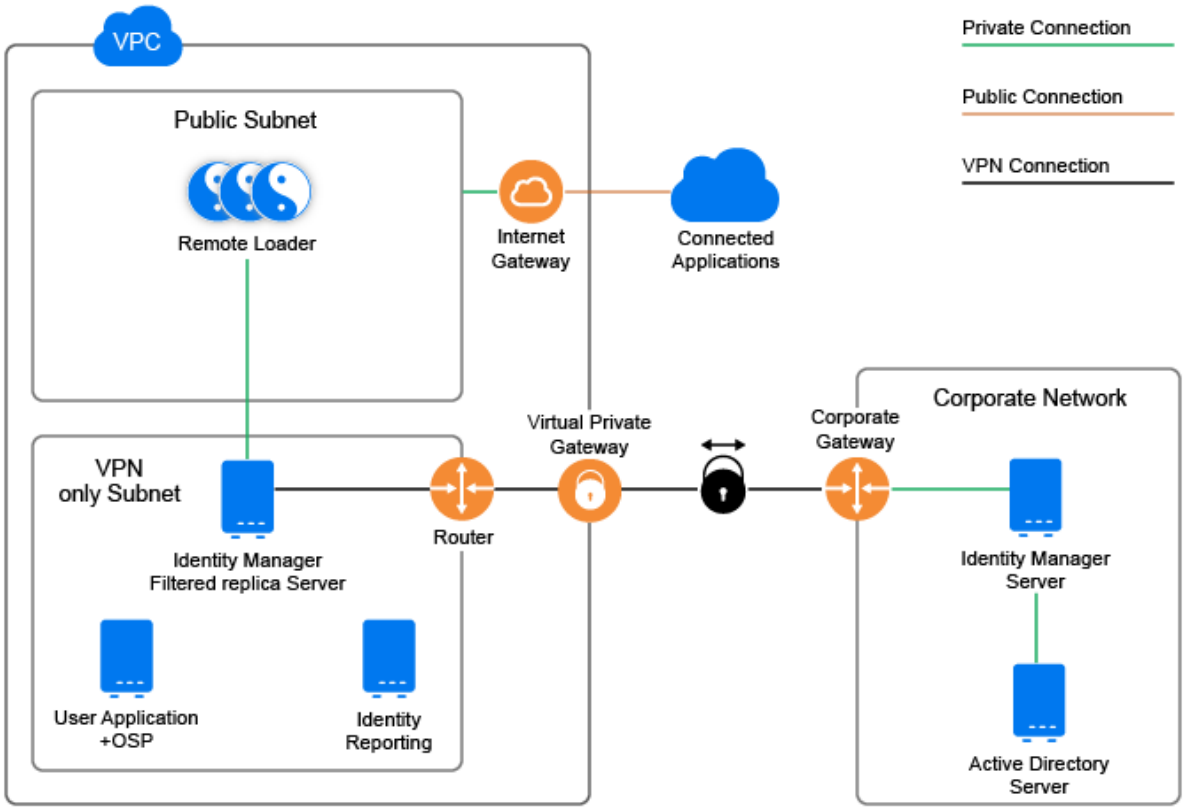


This configuration uses VPN connection only for synchronizing the delta changes between the Identity Manager servers on either side.

With Filtered Replication

This is a variant of multi-server driverset scenario and includes a filtered read-write replica of the data partition on the server in the AWS cloud. For driverset partition, you should always use full replica partition on either side.

Figure 15-3 Hybrid Scenario Using Controlled Replication

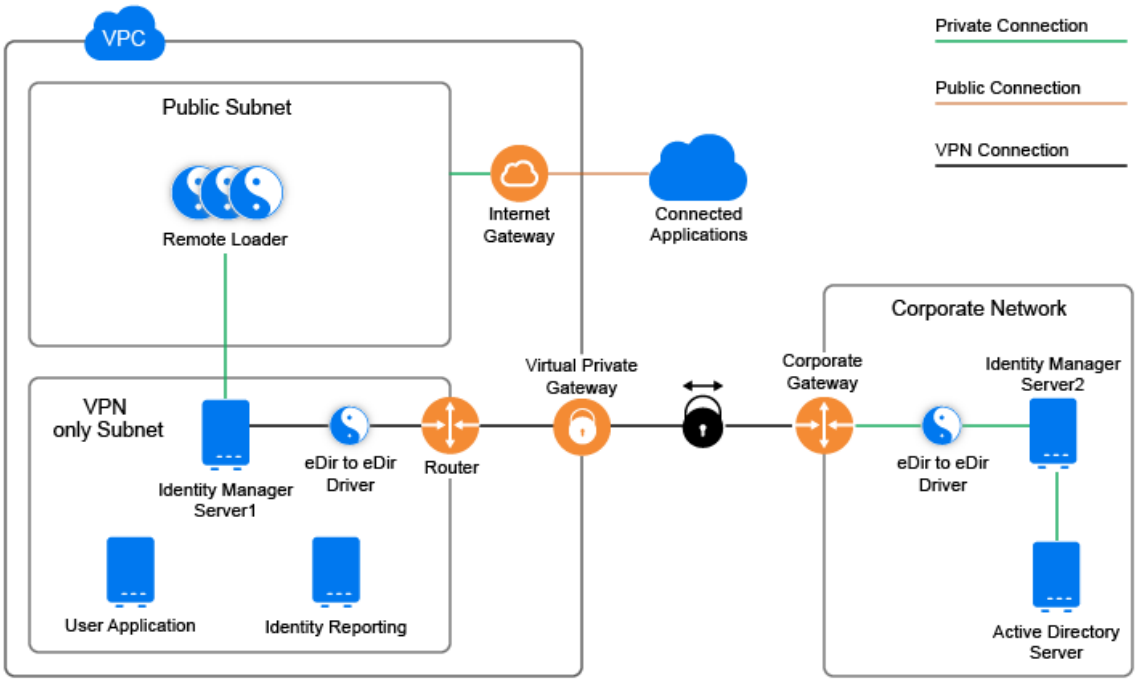


This scenario adds more control over the attributes to synchronize. For example, you can prevent sensitive attributes from synchronizing with the Identity Manager server on AWS cloud.

Using eDirectory Driver Connection

This scenario is suitable if you have Identity Manager servers installed on two separate eDirectory trees where one tree belongs to AWS cloud and the other tree belongs to the enterprise network. This configuration uses eDirectory driver to synchronize the identities between AWS cloud and the enterprise network through a VPN connection. The Identity Manager server that is running on the enterprise network or AWS cloud synchronizes the identities across their respective connected applications.

Figure 15-4 *Hybrid Scenario Using eDirectory Driver Connection*



The communication between the AWS cloud and the enterprise network is limited. It only synchronizes the delta changes. You can control the attributes to synchronize by configuring the driver filter. You can also leverage the policy engine to define additional controls for synchronizing attributes. For example, limit the password attribute from synchronizing and allow users to use different passwords to access Identity Manager servers from AWS cloud and the enterprise network.

VIII Deploying Identity Manager for High Availability

High availability ensures efficient manageability of critical network resources including data, applications, and services. NetIQ supports high availability for your Identity Manager solution through clustering or Hypervisor clustering, such as VMWare Vmotion. When planning a high-availability environment, the following considerations apply:

- You can install the following components in a high-availability environment:
  - Identity Vault
  - Identity Manager engine
  - Remote Loader
  - Identity applications, except Identity Reporting
- To manage the availability of your network resources for your Identity Manager environment, use the SUSE Linux Enterprise High Availability Extension with SUSE Linux Enterprise Server (SLES) 12 SP3 or later with the latest patches installed.
- When you run the Identity Vault in a clustered environment, the Identity Manager engine is also clustered.

NOTE: Identity Manager does not support load balancing LDAP or LDAPS communication between Identity Vault and Identity Applications.

For more information about...	See...
Determining the server configuration for Identity Manager components	see High Availability Configuration in *NetIQ Identity Manager Overview and Planning Guide*.
Running the Identity Vault in a cluster	Sample Identity Manager Cluster Deployment Solution on SLES 12 SP3 or Later Versions Deploying eDirectory on High Availability Clusters in the *NetIQ eDirectory Installation Guide*
Running the identity applications in a cluster	Sample Identity Applications Cluster Deployment Solution on Tomcat Application Server

For more information on implementing high availability and disaster recovery in your Identity Manager environment, contact NetIQ Technical Support (https://www.netiq.com/support/).

This following chapters provide the steps for installing and configuring Identity Manager components in a high availability environment:

- Chapter 16, “Preparing for Installing Identity Manager in a Cluster Environment,” on page 271
- Chapter 17, “Sample Identity Manager Cluster Deployment Solution on SLES 12 SP3 or Later Versions,” on page 275
- Chapter 18, “Sample Identity Applications Cluster Deployment Solution on Tomcat Application Server,” on page 285

16 Preparing for Installing Identity Manager in a Cluster Environment

- Prerequisites
- Preparing a Cluster for the Identity Applications

Prerequisites

- "Identity Vault" on page 271
- "Identity Applications" on page 272
- "Database for Identity Applications" on page 272

Identity Vault

Before installing the Identity Vault in a clustered environment, NetIQ recommends reviewing the following considerations:

- You must have external shared storage supported by the cluster software, with sufficient disk space to store all Identity Vault and NICI data:
  - The Identity Vault DIB must be located on the cluster shared storage. State data for the Identity Vault must be located on the shared storage so that it is available to the cluster node that is currently running the services.
  - The root Identity Vault instance on each of the cluster nodes must be configured to use the DIB on the shared storage.
  - You must also share NICI (NetIQ International Cryptographic Infrastructure) data so that server-specific keys are replicated among the cluster nodes. NICI data used by all cluster nodes must be located on the cluster shared storage.
  - NetIQ recommends storing all other eDirectory configuration and log data on the shared storage.
- You must have a virtual IP address.
- (Conditional) If you are using eDirectory as the support structure for the Identity Vault, the `nds-cluster-config` utility supports configuring the root eDirectory instance only. eDirectory does not support configuring multiple instances and non-root installations of eDirectory in a cluster environment.

For more information about installing the Identity Vault in a clustered environment, see Deploying eDirectory on High Availability Clusters in the *NetIQ eDirectory Installation Guide*.

Identity Applications

You can install the database for the identity applications in an environment supported by Tomcat clusters with the following considerations:

- The cluster must have a unique cluster partition name, multicast address, and multicast port. Using unique identifiers separates multiple clusters to prevent performance problems and anomalous behavior.
  - For each member of the cluster, you must specify the same port number for the listener port of the identity applications database.
  - For each member of the cluster, you must specify the same hostname or IP address of the server hosting the identity applications database.
- You must synchronize the clocks of the servers in the cluster. If server clocks are not synchronized, sessions might time out early, causing HTTP session failover not to work properly.
- NetIQ recommends to not use multiple log ins across browser tabs or browser sessions on the same host. Some browsers share cookies across tabs and processes, so allowing multiple logins might cause problems with HTTP session failover (in addition to risking unexpected authentication functionality if multiple users share a computer).
- The cluster nodes reside in the same subnet.
- A failover proxy or a load balancing solution is installed on a separate computer.

Database for Identity Applications

Database clustering is a feature of each respective database server. NetIQ does not officially test with any clustered database configuration because clustering is independent of the product functionality. Therefore, we support clustered database servers with the following caveats:

- By default, the maximum number of connections is set to 100. This value might be too low to handle the workflow request load in a cluster. You might see the following exception:

  ```
  (java.sql.SQLException: Data source rejected establishment of
  connection, message from server: "Too many connections."
  ```

  To increase the maximum number of connections, set the `max_connections` variable in the `my.cnf` file to a higher value.
- Some features or aspects of your clustered database server might need to be disabled. For example, Transactional Replication must be disabled on certain tables due to constraint violations when trying to insert a duplicate key.
- We do not provide assistance on the installation, configuration, or optimization of the clustered database server, including installation of our products into a clustered database server.
- We exert our best effort to resolve any issues that might arise with the use of our products in a clustered database environment. Troubleshooting methods in a complex environment often require cooperative work to resolve issues. NetIQ provides expertise to analyze, plan, and troubleshoot the NetIQ products. The customer must provide expertise to analyze, plan and troubleshoot any third-party products. We ask customers to reproduce issues or analyze behavior of their components in a non-clustered environment to help isolate potential cluster setup issues from NetIQ product issues.

Preparing a Cluster for the Identity Applications

The identity applications supports HTTP session replication and session failover. If a session is in process on a node and that node fails, the session can be resumed on another server in the cluster without intervention. Before installing the identity applications in a cluster, you should prepare the environment.

- "Understanding Cluster Groups in Tomcat Environments" on page 273
- "Setting System Properties for Workflow Engine IDs" on page 273

Understanding Cluster Groups in Tomcat Environments

The User Application cluster group uses a UUID name to minimize the risk of conflicts with other cluster groups that users might add to their servers. You can modify the configuration settings for User Application cluster group using the User Application administration features. Changes to the cluster configuration take effect for a server node only when you restart that node.

Setting System Properties for Workflow Engine IDs

Each server that hosts the identity applications in the cluster can run a workflow engine. To ensure performance of the cluster and the workflow engine, every server in the cluster should use the same partition name and partition UDP group. Also, each server in the cluster must be started with a unique ID for the workflow engine, because clustering for the workflow engine works independently of the cache framework for the identity applications.

To ensure that your workflow engines run appropriately, you must set system properties for Tomcat.

1. Create a new JVM system property for each identity applications server in the cluster.
2. Name the system property `com.novell.afw.wf.`*`engine-id`* where the engine ID is a unique value.
3. The Identity Applications and Identity Reporting are there in the Identity Vault and Identity Manager Engine.

17 Sample Identity Manager Cluster Deployment Solution on SLES 12 SP3 or Later Versions

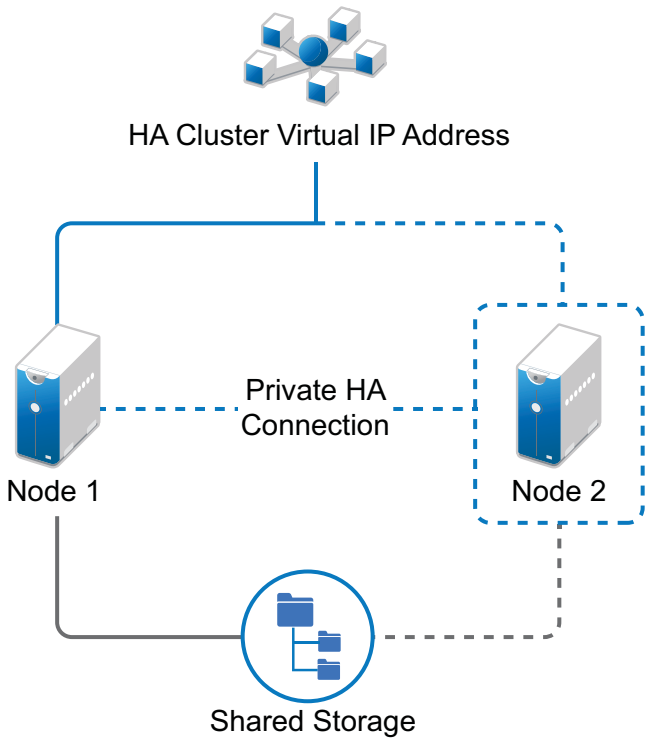
The chapter provides step-by-step instructions on how to configure eDirectory and Identity Manager into a supported SUSE Linux Enterprise Server (SLES) cluster environment with shared storage and an example of a clustered Identity Manager deployment.

- "Prerequisites" on page 276
- "Installation Procedure" on page 276

For a production-level Linux High Availability (HA) solution with shared storage, implementing a fencing mechanism in the cluster is recommended. Although there are different methods of implementing fencing mechanisms in the cluster, in our example, we use a STONITH resource which uses the Split Brain Detector (SBD).

Figure 17-1 on page 275 shows a sample cluster deployment solution.

Figure 17-1 *Sample cluster deployment solution*



Prerequisites

- Two servers running SLES 12 SP3 64-bit for nodes
- One server running SLES 12 SP3 64-bit for iSCSI Server
- SLES12 SP3 64-bit HA extension ISO image file
- Six static IPs:
  - Two static IP addresses for each node.
  - One static IP address for the cluster. This IP address is dynamically assigned to the node currently running eDirectory.
  - One IP address for iSCSI Server.

Installation Procedure

This section explains the steps to install Identity Manager in a cluster environment. For more information about configuring the SLES High Availability Extension, see the SUSE Linux Enterprise High Availability Extension guide.

- Configuring the iSCSI Server
- Configuring the iSCSI initiator on all Nodes
- Partitioning the Shared Storage
- Installing the HA Extension
- Setting up Softdog Watchdog
- Configuring the HA Cluster
- Installing and Configuring Identity Vault and Identity Manager Engine on Cluster Nodes
- Configuring the Identity Vault Resource
- Primitives for eDirectory and Shared Storage Child Resources
- Changing the Location Constraint Score

Configuring the iSCSI Server

An iSCSI target is a device that is configured as a common storage for all nodes in a cluster. It is a virtual disk that is created on the Linux server to allow remote access over an Ethernet connection by an iSCSI initiator.An iSCSI initiator is any node in the cluster that is configured to contact the target (iSCSI) for services. The iSCSI target should be always up and running so that any host acting as an initiator can contact the target. Before installing iSCSI target on the iSCSI server, ensure that the iSCSI target has sufficient space for a common storage.Install the iSCSI initiator packages on the other two nodes after installing SLES 12 SP3.

During the SLES 12 SP3 installation:

1. Create a separate partition and specify the partition path as the iSCSI shared storage partition.
2. Install the iSCSI target packages.

To configure the iSCSI server:

1. Create a block device on the target server.
2. Type the `yast2 disk` command in terminal.
3. Create a new Linux partition, and select **Do not format**.
4. Select **Do not mount the partition**.
5. Specify the partition size.
6. Type the `yast2 iscsi-server` or `yast2 iscsi-lio-server` command in terminal.
7. Click the **Service** tab, then select **When Booting** in the **Service Start** option.
8. In the **Targets** tab, click **Add** to enter the partition path (as created during the SLES installation).
9. In the **Modify iSCSI Target Initiator Setup** page, specify iSCSI client initiator host names for the target server and then click **Next**.

   For example, *iqn.sles12sp3node2.com* and *iqn.sles12sp3node3.com*.
10. Click **Finish**.
11. Run the `cat /proc/net/iet/volume` command in the terminal to verify if the iSCSI target is installed

Configuring the iSCSI initiator on all Nodes

You must configure the iSCSI initiator on all cluster nodes to connect to the iSCSI target.

To configure the iSCSI initiator:

1. Install the iSCSI initiator packages.
2. Run the `yast2 iscsi-client` in terminal.
3. Click the **Service** tab and select **When Booting** in the **Service Start** option.
4. Click the **Connected Targets** tab, and click **Add** to enter the IP address of the iSCSI target server.
5. Select **No Authentication**.
6. Click **Next**, then click **Connect**.
7. Click **Toggle Start-up** to change the start-up option from manual to automatic, then click **Next**.
8. Click **Next**, then click **OK**.
9. To check the status of the connected initiator on the target server, run the `cat /proc/net/iet/session` command on the target server. The list of initiators that are connected to iSCSI server are displayed.

Partitioning the Shared Storage

Create two shared storage partitions: one for SBD and the other for Cluster File System.

To partition the shared storage:

1. Run the `yast2 disk` command in terminal.
2. In the **Expert Partitioner** dialog box, select the shared volume. In our example, select **sdb** from the **Expert Partitioner** dialog box.

3 Click **Add**, select **Primary partition** option, and click **Next**.

4 Select **Custom size**, and click **Next**. In our example, the custom size is 100 MB.

5 Under **Formatting options**, select **Do not format partition**. In our example, the File system ID is 0x83 Linux.

6 Under **Mounting options**, select **Do not mount partition**, then click **Finish**.

7 Click **Add**, then select **Primary partition**.

8 Click **Next**, then select **Maximum Size**, and click **Next**.

9 In **Formatting options**, select **Do not format partition**. In our example, specify the File system ID as 0x83 Linux.

10 In **Mounting options**, select **Do not mount partition**, then click **Finish**.

Installing the HA Extension

To install the HA extension:

1 Go to the SUSE Downloads website.

SUSE Linux Enterprise High Availability Extension (SLE HA) is available for download for each available platform as two ISO images. Media 1 contains the binary packages and Media 2 contains the source code.

NOTE: Select and install the appropriate HA extension ISO file based on your system architecture.

---

2 Download the Media 1 ISO file on each server.

3 Open **YaST Control Center** dialog box, click **Add-on products > Add**.

4 Click **Browse** and select the DVD or local ISO image, then click **Next**.

5 In the **Patterns** tab, select **High Availability** under **Primary Functions**.

Ensure that all the components under high availability are installed.

6 Click **Accept**.

Setting up Softdog Watchdog

In SLES HA Extension, the Watchdog support in the kernel is enabled by default. It is shipped with a number of different kernel modules that provide hardware-specific watchdog drivers. The appropriate watchdog driver for your hardware is automatically loaded during system boot.

1 Enable the softdog watchdog:

```
echo softdog > /etc/modules-load.d/watchdog.conf
systemctl restart systemd-modules-load
```

2 Test if the softdog module is loaded correctly:

```
lsmod | grep dog
```

Configuring the HA Cluster

This example assumes that you are configuring two nodes in a cluster.

Setting up the first node:

1. Log in as root to the physical or virtual machine you want to use as cluster node.
2. Run the following command:

   `ha-cluster-init`

   The command checks for NTP configuration and a hardware watchdog service. It generates the public and private SSH keys used for SSH access and Csync2 synchronization and starts the respective services.
3. Configure the cluster communication layer:
   - **3a** Enter a network address to bind to.
   - **3b** Enter a multicast address. The script proposes a random address that you can use as default.
   - **3c** Enter a multicast port. By default, the port is 5405.
4. Set up SBD as the node fencing mechanism:
   - **4a** Press *y* to use SBD.
   - **4b** Enter a persistent path to the partition of your block device that you want to use for SBD. The path must be consistent for both the nodes in the cluster.
5. Configure a virtual IP address for cluster administration:
   - **5a** Press *y* to configure a virtual IP address.
   - **5b** Enter an unused IP address that you want to use as administration IP for SUSE Hawk GUI. For example, *192.168.1.3*.

     Instead of logging in to an individual cluster node, you can connect to the virtual IP address.

Once the first node is up and running, add the second cluster node using the `ha-cluster-join` command.

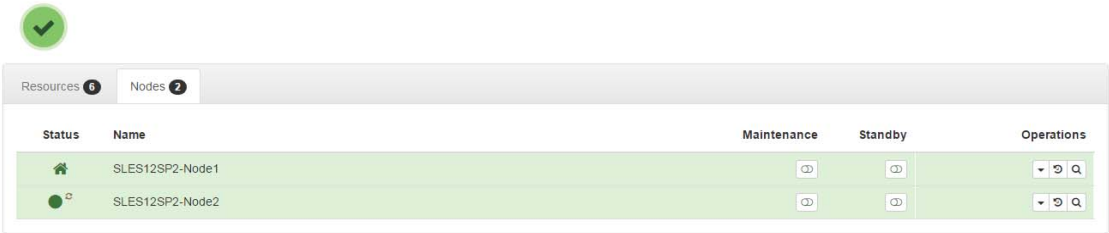
Setting up the second node:

1. Log in as root to the physical or virtual machine through which you want to connect to the cluster.
2. Run the following command:

   `ha-cluster-join`

   If NTP is not configured, a message appears. The command checks for a hardware watchdog device and notifies if it is not present.
3. Enter the IP address of the first node.
4. Enter the root password of the first node.

5 Log in to SUSE Hawk GUI and then click **Status** > **Nodes**. For example, `https://192.168.1.3:7630/cib/live`.



Installing and Configuring Identity Vault and Identity Manager Engine on Cluster Nodes

1 Install Identity Manager Engine on cluster nodes:

   1a Download the `Identity_Manager_4.8_Linux.iso` from the NetIQ Downloads website.

   1b Mount the downloaded `.iso`.

   1c From the ISO mounted location, run the following command:

   ```
   ./install.sh
   ```

   1d Read through the license agreement.

   1e Enter y to accept the license agreement.

   1f Decide the Identity Manager server edition you want to install. Enter y for Advanced Edition and n for Standard Edition.

   1g Select **Identity Manager Engine** from the list and proceed with the installation.

   This step installs the supported Identity Vault version.

2 Configure Identity Manager Engine on all nodes.

   2a Navigate to the location where you have mounted the `Identity_Manager_4.8_Linux.iso`.

   2b From the ISO mounted location, run the following command:

   ```
   ./configure.sh
   ```

   2c Decide whether you want to perform a typical configuration or a custom configuration. The configuration options will vary based on the components that you select for configuration.

   2d Select the **Identity Manager Engine** component from the list.

   2e If your are configuring the Identity Vault for the first time, select the **Create a new Identity Vault** option. If you have installed Identity Vault previously and want to connect to that Identity Vault instance, select the **Add to an Identity Vault existing on local machine** or **Add to an Identity Vault existing on local machine** option.

3 Navigate to the `/etc/opt/novell/eDirectory/conf` directory.

4 Edit the `nds.conf` file and specify the virtual IP address of the cluster in the `n4u.nds.preferred-server` field.

5 Stop the Identity Vault service.

```
ndsmanage stopall
```

6 Back up all the folders and files from the `/var/opt/novell/nici`, `/etc/opt/novell/eDirectory/conf`, and `/var/opt/novell/eDirectory/` directories.

7 Navigate to the `/opt/novell/eDirectory/bin` directory.

8 Run the following command:

```
nds-cluster-config -s /<shared cluster path>
```

where, `<shared cluster path>` indicates the location that you want use for the Identity Vault shared cluster data.

9 Start the Identity Vault service.

```
ndsmanage startall
```

For more information on configuring Identity Vault in a clustered setup, see "Deploying eDirectory on High Availability Clusters" in the eDirectory Installation Guide.

Configuring the Identity Vault Resource

1 Log in to SUSE Hawk GUI.

2 Click **Add Resource** and create a new group.

   2a Click ⊕ next to the **Group**.

   2b Specify a group ID. For example, *Group-1*.

   Ensure that the following child resources are selected when you create a group:

   - *stonith-sbd*
   - *admin_addr (Cluster IP address)*

3 In the **Meta Attributes** tab, set the **target-role** field to *Started* and **is-managed** field to *Yes*.

4 Click **Create**.

5 Click **Edit Configuration** and then click ✎ next to the group you created in step 2.

6 In the **Children** field, add the following child resources:

   - *shared-storage*
   - *eDirectory-resource*

   For example, the resources should be added in the following order within the group:

   - *stonith-sbd*
   - *admin_addr (Cluster IP address)*
   - *shared-storage*
   - *eDirectory-resource*

   You can change the resource names if necessary. Every resource has a set of parameters that you need to define. For information about examples for shared-storage and eDirectory resources, see Primitives for eDirectory and Shared Storage Child Resources.

Primitives for eDirectory and Shared Storage Child Resources

The *stonith-sbd* and admin_addr resources are configured by HA Cluster commands by default when the cluster node is initialized.

Table 17-1 *Example for shared-storage*

Resource ID	Name of the shared storage resource
Class	ocf
Provider	heartbeat
Type	Filesystem
device	/dev/sdc1
directory	/shared
fstype	xfs
operations	◆ start (60, 0) ◆ stop (60, 0) ◆ monitor (40, 20)
is-managed	Yes
resource-stickiness	100
target-role	Started

Table 17-2 *Example for eDirectory-resource*

Resource ID	Name of the eDirectory resource
Class	systemd
Type	ndsdtmpl-shared-conf-nds.conf@-shared-conf-env
operations	◆ start (100, 0) ◆ stop (100, 0) ◆ monitor (100, 60)
target-role	Started
is-managed	Yes
resource-stickiness	100
failure-timeout	125
migration-threshold	0

Changing the Location Constraint Score

Change the location constraint score to 0.

1. Log in to SUSE Hawk GUI.
2. Click **Edit Configuration**.
3. In the **Constraints** tab, click ✎ next to the node 1 of your cluster.
4. In the **Simple** tab, set the score to *0*.
5. Click **Apply**.

Ensure that you set the score to *0* for all the nodes in your cluster.

NOTE: When you migrate the resources from one node to another from the SUSE Hawk GUI using the **Status** > **Resources** > **Migrate** option, the location constraint score will change to *Infinity* or -*Infintity*. This will give preference to only one of the nodes in the cluster and will result in delays in eDirectory operations.

---

18 Sample Identity Applications Cluster Deployment Solution on Tomcat Application Server

This chapter provides instructions on how to configure the identity applications into a cluster environment on Tomcat with an example deployment.

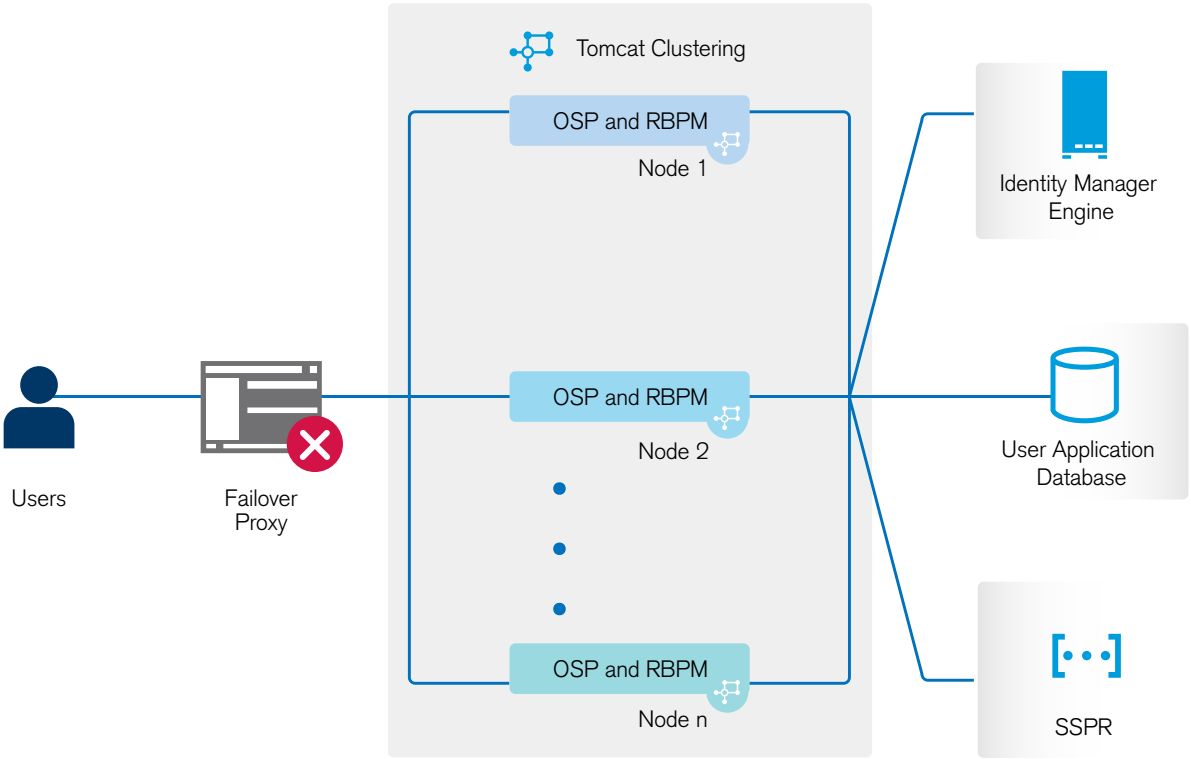
Clustering allows you to run the identity applications on several parallel servers (cluster nodes) to achieve high availability. To build a cluster, you need to group several Tomcat instances (nodes) together. The load is distributed across different servers, and even if any of the servers fail, the identity applications are accessible through other cluster nodes. For failover, you can create a cluster of the identity applications and configure them to act as a single server. However, this configuration does not include Identity Reporting.

It is recommended to use a load balancer software that processes all user requests and dispatches them to the server nodes in the cluster. The load balancer is typically part of the cluster. It understands the cluster configuration as well as failover policies. You can select a solution that best suits you.

Figure 18-1 shows a sample deployment with a two-node cluster with the following assumptions:

- All the communication is routed through the load balancer.
- Components such as Identity Manager engine and the User Application are installed on separate servers. This is a recommended approach for a production-level deployment.
- You are familiar with the installation procedures for eDirectory, Identity Manager engine, identity applications, Tomcat application server, and databases for the User Application.
- SSPR (Single Sign-On Password Reset) is installed on a separate computer. For a production-level deployment, this is the recommended approach.
- PostgreSQL is used as a database for the User Application. However, you can use any of the supported databases, such as Oracle or MsSQL.
- All the User Application nodes communicate to the same instance of eDirectory and the User Application database. Based on your requirement, you can increase the number of User Application instances.

Figure 18-1 Sample cluster deployment solution



NOTE: A two-node cluster is the minimum configuration used for high availability. However, the concepts in this section can easily be extended to a cluster with additional nodes.

To help you understand the step-by-step configuration, this sample deployment is referred throughout the subsequent sections of the document.

Prerequisites

You can install the database for the identity applications in an environment supported by Tomcat clusters with the following considerations:

- The cluster must have a unique cluster partition name, multicast address, and multicast port. Using unique identifiers separates multiple clusters to prevent performance problems and anomalous behavior.
  - For each member of the cluster, you must specify the same port number for the listener port of the identity applications database.
  - For each member of the cluster, you must specify the same hostname or IP address of the server hosting the identity applications database.
- Clock time is synchronized among the servers in the cluster. Otherwise, sessions might time out early, causing HTTP session failover not to work properly.
- NetIQ recommends to not use multiple log ins across browser tabs or browser sessions on the same host. Some browsers share cookies across tabs and processes, so allowing multiple logins might cause problems with HTTP session failover (in addition to risking unexpected authentication functionality if multiple users share a computer).

- The cluster nodes reside in the same subnet.
- A failover proxy or a load balancing solution is installed on a separate computer.
- To achieve clustering for forms, start two instances of load balancer on the server, one for the Identity Applications and the other for the form renderer.

Preparing a Cluster

The identity applications supports HTTP session replication and session failover. If a session is in process on a node and that node fails, the session can be resumed on another server in the cluster without intervention. Before installing the identity applications in a cluster, you should prepare the environment.

- "Understanding Cluster Groups in Tomcat Environments" on page 287
- "Setting System Properties for Workflow Engine IDs" on page 287

Understanding Cluster Groups in Tomcat Environments

The User Application cluster group uses a UUID name to minimize the risk of conflicts with other cluster groups that users might add to their servers. You can modify the configuration settings for User Application cluster group using the User Application administration features. Changes to the cluster configuration take effect for a server node only when you restart that node.

Setting System Properties for Workflow Engine IDs

Each server that hosts the identity applications in the cluster can run a workflow engine. To ensure performance of the cluster and the workflow engine, every server in the cluster should use the same partition name and partition UDP group. Also, each server in the cluster must be started with a unique ID for the workflow engine, because clustering for the workflow engine works independently of the cache framework for the identity applications.

To ensure that your workflow engines run appropriately, you must set system properties for Tomcat.

1. Create a new JVM system property for each identity applications server in the cluster.
2. Name the system property `com.novell.afw.wf.`*`engine-id`* where the engine ID is a unique value.

Installation Procedure

This section provides step-by-step instructions of installing a new instance of the identity applications on Tomcat and then configuring it for clustering.

1. Install the Identity Manager engine. For a production-level deployment, it is recommended to install Identity Manager engine on a separate server.
2. Install database for Identity Applications. You can use the PostgreSQL database installed with the Identity Applications. However, it is recommended to install database on a separate server.
3. On Node1, install and configure Identity Applications.

During configuration, ensure that you:

- select the new database option
- provide a unique Workflow Engine ID. For example, `Node1`.
- have the database jar file available in all the User Application nodes in the cluster. For PostgreSQL, the `postgresql-9.4.1212.jar` is located at `/opt/netiq/idm/postgres`.

Identity Applications encrypt sensitive data using a master key. The installation program will create a new master key during Identity Applications configuration. In a cluster, the User Application clustering requires every instance of the User Application to use the same master key. Master key is stored under the property `com.novell.idm.masterkey` in the `ism-configuration.properties` file located at `/opt/netiq/idm/apps/tomcat/conf/` directory.

4. On Node2, install and configure Identity Applications.

   During configuration, ensure that you:

   - select the existing database option
   - provide a unique Workflow Engine ID. For example, `Node2`.
   - have the database jar file available in all the User Application nodes in the cluster. For PostgreSQL, the `postgresql-9.4.1212.jar` is located at `/opt/netiq/idm/postgres`.

   After completing the Node2 User Application configuration, copy the master key value from the Node1 `ism-configuration.properties` and replace the corresponding master key value stored in Node 2's `ism-configuration.properties`.

   Master key is stored under the property `com.novell.idm.masterkey` in the `ism-configuration.properties` (`/opt/netiq/idm/apps/tomcat/conf/`).

5. In load balancer server, start an instance of load balancer with Identity Applications port number and other instance of load balancer with form renderer port number for all clustered nodes. For example,

   - `./balance 8543 apps1-au.edu.in:8543 ! apps2-au.edu.in:8543`
   - `./balance 8600 apps1-au.edu.in:8600 ! apps2-au.edu.in:8600`

6. Install SSPR on a separate computer.

   After completing the SSPR installation, launch SSPR (`https://<IP>:<port>/sspr/private/config/editor`) and log in. Click **Configuration Editor** > **Settings** > **Security** > **Web Security** > **Redirect Whitelist**.

   a. Click **Add value** and specify the following URL:

      `https://<dns of the failover>:<port>/osp`

   b. Save the changes.

   c. In the SSPR Configuration page, click **Settings** > **Single Sign On (SSO) Client** > **OAuth** and modify the **OAuth Login URL**, **OAuth Code Resolve Service URL**, and **OAuth Profile Service URL** links by replacing the IP addresses with the DNS name of the server where the load balancer software is installed.

   d. Click **Settings** > **Application** > **Application** and update the **Forward URL** and **Logout URL** by replacing the IP addresses with the DNS name of the server where the load balancer software is installed. Update the **Site URL** by providing the IP address or hostname of the server/system where SSPR is installed.

e. To update the SSPR information on Node1, launch the Configuration utility located at `/opt/netiq/idm/apps/configupdate/`

```
./configupdate.sh
```

NOTE: You should run the `configudate.sh` file from the configupdate directory only. Running the `configupdate.sh` from a custom location will result in failures.

f. Click **SSO clients** > **Self Service Password Reset** and enter values for **Client ID**, **Password**, and **OSP Auth Redirect URL** parameters. For more information, see "Self Service Password Reset" on page 124.

NOTE: Verify that the values for these parameters are updated in Node2.

7. In Node1, stop Tomcat and generate a new `osp.jks` file by specifying the DNS name of the load balancer server by using the following command:

```
/opt/netiq/common/jre/bin/keytool -genkey -storetype PKCS12 -keyalg RSA
-keysize 2048 -keystore osp.jks -storepass <password> -keypass
<password> -alias osp -validity 1800 -dname "cn=<loadbalancer IP/DNS>"
```

For example: `/opt/netiq/common/jre/bin/keytool -genkey -storetype PKCS12 -keyalg RSA -keysize 2048 -keystore osp.jks -storepass changeit -keypass changeit -alias osp -validity 1800 -dname "cn=mydnsname"`

NOTE: Ensure that the key password is the same as the one provided during OSP installation. Alternatively, this can also be changed using Configuration Update utility including the keystore password.

8. (Conditional) To verify if the `osp.jks` file is updated with the changes, run the following command:

```
/opt/netiq/common/jre/bin/keytool -list -v -keystore osp.jks -storepass
changeit
```

9. Take a backup of the original `osp.jks` file located at `/opt/netiq/idm/apps/osp/` and copy the new `osp.jks` file to this location.
10. Copy the new `osp.jks` file located at `/opt/netiq/idm/apps/osp/` from Node1 to other User Application nodes in the cluster.
11. On each clustered node,

a. Navigate to the `/opt/netiq/idm/apps/sites` directory and edit the `ServiceRegistry.json` file to add the load balancer details.

```
{"serviceRegisteries":[{"serviceID":"IDM","restUrl":"https://<DNS
of the load balancer>:8543/IDMProv"}]}
```

b. Navigate to the `/opt/netiq/idm/apps/sites` directory and edit the `config.ini` file to add the load balancer DNS and port number.

```
OSPIssuerUrl=https://<DNS of the load balancer>:8543/osp/a/idm/
auth/oauth2
OSPRedirectUrl=https://<DNS of the load balancer>:8600/forms/
oauth.html
ClientID=forms
OSPLogoutUrl=https://<DNS of the load balancer>:8543/osp/a/idm/
auth/app/logout
```

12. Launch the Configuration utility in Node1 and change all of the URL settings, such as URL link to landing page and OAuth Redirect URL to the load balancer DNS name under the SSO Client tab.

    a. Save the changes in the Configuration utility. Check the `ism-configuration properties` file for the changes and modify if any URLs are still pointing to Node 1 DNS and port.

    b. To reflect this change in all other nodes of the cluster, copy the `ism-configuration properties` file located in `/TOMCAT_INSTALLED_HOME/conf` from Node1 to other User Application nodes in the cluster.

NOTE: ◆You copied the `ism-configuration.properties` file from Node1 to the other nodes in the cluster. If you specified custom installation paths during the User Application installation, ensure that referential paths are corrected by using Configuration update utility in the cluster nodes.

- After copying the `ism-configuration.properties` file from one node to another, ensure that the file has `novlua:novlua` permissions.
- In this scenario, both OSP and User Application are installed on the same server; therefore, the same DNS name is used for redirect URLs.
- If OSP and User Application are installed on separate servers, change the OSP URLs to a different DNS name pointing to the load balancer. Do this for all the servers where OSP is installed. Doing this ensures that all OSP requests are dispatched through load balancer to the OSP cluster DNS name. This involves having a separate cluster for OSP nodes.

---

13. Assign the `novlua` permission to the `osp.jks` file:

    ```
    chown novlua:novlua osp.jks
    ```

14. Perform the following actions in the `setenv.sh` file located at `/TOMCAT_INSTALLED_HOME/bin/` directory:

    a. To ensure that the `mcast_addr` binding is successful, JGroups requires that the `preferIPv4Stack property` be set to **true**. To do so, add the JVM property "-`Djava.net.preferIPv4Stack=true`" in the `setenv.sh` file in all nodes.

    b. Add `-Dcom.novell.afw.wf.engine-id="`*`Engine1`*`"` in the `setenv.sh` file on Node1. Similarly, add a unique engine name for each node of the cluster. For example, for Node2, you can add the engine name as *`Engine2`*.

15. Enable clustering in the User Application.

    a. Start Tomcat on Node1.

    Do not start any other servers.

    b. Log in to the User Application as a User Application Administrator.

c. Click the **Configuration** > **Caching and Cluster** option.

   The User Application displays the Caching Management page.

d. Click **Cluster Cache Configuration** and select **True** for the **Cluster Enabled** property.

e. Click **Save**.

f. Restart Tomcat.

NOTE: If you have selected Enable Local settings, repeat this procedure for each server in the cluster.

The User Application cluster uses JGroups for cache synchronization across nodes using default UDP. In case you want to change this protocol to use TCP, see Configuring User Application Caching to use TCP in the NetIQ Identity Manager - Administrator's Guide to the Identity Applications.

---

16. Enable the permission index for clustering.

    a. Log in to iManager on IDVault and navigate to **View Objects**.

    b. Under **System**, navigate to the driver set containing the User Application driver.

    c. Select **AppConfig** > **AppDefs > > Configuration**.

    d. Select the XMLData attribute and set the `com.netiq.idm.cis.clustered` property to **true**.

       For example:

       ```
       <property>
       <key>com.netiq.idm.cis.clustered</key>
       <value>true</value>
       </property>
       ```

    e. Click **OK**.

    f. Click **Apply** > **OK**.

17. Enable Tomcat cluster.

    Open the Tomcat `server.xml` file from `/TOMCAT_INSTALLED_HOME/conf/` and uncomment this line in this file on all the cluster nodes:

    ```
    <Cluster className="org.apache.catalina.ha.tcp.SimpleTcpCluster"/>
    ```

    For advanced Tomcat clustering configuration, follow the steps from https://tomcat.apache.org/tomcat-8.5-doc/cluster-howto.html.

18. Restart Tomcat on all the nodes.

19. Configure the User Application Driver for clustering.

    In a cluster, the User Application driver must be configured to use the DNS name of the load balancer for the cluster. You configure the User Application driver using iManager.

    a. Log in to iManager that manages your Identity Manager engine.

    b. Click the **Identity Manager node** in the iManager navigation frame.

    c. Click **Identity Manager Overview.**

    d. Use the search page to display the Identity Manager Overview for the driver set that contains your User Application driver and Roles and Resource Service Driver.

e. Click the round status indicator in the upper right corner of the driver icon:

   A menu is displayed that lists commands for starting and stopping the driver, and editing driver properties.

f. Select **Edit Properties**.

g. In the Driver Parameters section, change **Host** to the host name or IP address of the Load Balancer.

h. Click **OK**.

i. Restart the driver.

20. To change the URL of Roles and Resource Service Driver, repeat steps from 19a to 19f and click **Driver Configuration** and update the **User application URL** with the load balancer DNS name.
21. Ensure session stickiness is enabled for the cluster created in the load balancer software for the User Application nodes.
22. Configure client settings on Identity Manager dashboard. For more information, see Configuring Client Settings Mode in the NetIQ Identity Manager - Administrator's Guide to the Identity Applications.

Enabling Permission Index for Clustering

1. Log in to iManager on Node1 and navigate to **View Objects**.
2. Under **System**, navigate to the driver set containing the User Application driver.
3. Select **AppConfig** > **AppDefs > > Configuration**.
4. Select the XMLData attribute and set the `com.netiq.idm.cis.clustered` property to **true**.

   For example:

   ```
   <property>
   <key>com.netiq.idm.cis.clustered</key>
   <value>true</value>
   </property>
   ```

5. Click **OK**.

Configuring the User Application Driver for Clustering

In a clustered environment, you can use a single User Application driver with multiple instances of the User Application. The driver stores various kinds of information (such as workflow configuration and cluster information) that is application-specific. You must configure the driver to use the host name or IP address of the dispatcher or load balancer for the cluster.

1. Log in to the instance of iManager that manages your Identity Vault.
2. In the navigation frame, select **Identity Manager**.
3. Select **Identity Manager Overview.**
4. Use the search page to display the Identity Manager Overview for the driver set that contains your User Application driver.
5. Click the round status indicator in the upper right corner of the driver icon.

6 Select **Edit Properties**.

7 For **Driver Parameters**, change **Host** to the host name or IP address of the dispatcher.

8 Click **OK**.

9 Restart the driver.

Configuring OSP and SSPR for Clustering

Identity Manager supports SSPR configuration in a Tomcat cluster environment.

Configuring SSPR to Support Clustering

To update the SSPR information in the first node of the cluster, launch the Configuration utility from `/opt/netiq/idm/apps/configupdate/configupdate.sh`.

In the window that opens, click **SSO clients** > **Self Service Password Reset** and enter values for **Client ID**, **Password**, and **OSP Auth redirect URL** parameters.

Configuring Tasks on Cluster nodes

Perform the following configuration tasks on the cluster nodes:

1 To update the Forgotten Password link with the SSPR IP address, log in to the User Application on the first node and click **Administration** > **Forgot Password**.

For more information on SSPR configuration, see "Configuring Forgotten Password Management" on page 98.

2 To change the Change my password link, see "Updating SSPR Links in the Dashboard for a Distributed or Clustered Environment" on page 102.

3 Verify that the Forgot Password link and Change my password links are updated with the SSPR IP address on the other nodes in the cluster.

NOTE: If the Change Password and Forgot Password links are already updated with the SSPR IP address, no changes are required.

---

4 In the first node, stop Tomcat and generate a new `osp.jks` file by specifying the DNS name of the load balancer server by using the following command:

```
/opt/netiq/common/jre/bin/keytool -genkey -keyalg RSA -keysize 2048 -
keystore osp.jks -storepass <password> -keypass <password> -alias osp -
validity 1800 -dname "cn=<loadbalancer IP/DNS>"
```

For example : `/opt/netiq/common/jre/bin/keytool -genkey -keyalg RSA -keysize 2048 -keystore osp.jks -storepass changeit -keypass changeit -alias osp -validity 1800 -dname "cn=mydnsname"`

NOTE: Ensure that the key password is the same as the one provided during OSP installation. Alternatively, this can also be changed using Configuration Update utility including the keystore password.

---

5 (Conditional) To verify if the `osp.jks` file is updated with the changes, run the following command:

```
/opt/netiq/common/jre/bin/keytool -list -v -keystore osp.jks -storepass changeit
```

6 Take backup of the original `osp.jks` file located at `/opt/netiq/idm/apps/osp` and copy the new `osp.jks` file to this location. The new `osp.jks` file was created in Step 4.

7 Copy the new `osp.jks` file located at `/opt/netiq/idm/apps/osp` from the first node to all other User Application nodes in the cluster.

8 Launch the Configuration utility in the first node and change all of the URL settings, such as URL link to landing page and OAuth redirect URL to the load balancer DNS name under the SSO Client tab.

8a Save the changes in the Configuration utility.

8b To reflect this change in all other nodes of the cluster, copy the `ism-configuration properties` file located in `/TOMCAT_INSTALLED_HOME/conf` from the first node to all other User Application nodes.

NOTE: You copied the `ism.properties` file from the first node to the other nodes in the cluster. If you specified custom installation paths during User Application installation, ensure that referential paths are corrected by using Configuration update utility in the cluster nodes.

In this scenario, both OSP and User Application are installed on the same server; therefore, the same DNS name is used for redirect URLs.

If OSP and User Application are installed on separate servers, change the OSP URLs to a different DNS name pointing to load balancer. Do this for all the servers where OSP is installed. This ensures that all OSP requests are dispatched through load balancer to the OSP cluster DNS name. This involves having a separate cluster for OSP nodes.

---

9 Perform the following actions in the `setenv.sh` file in the `/TOMCAT_INSTALLED_HOME/bin/` directory:

9a To ensure that the mcast_addr binding is successful, JGroups requires that the `preferIPv4Stack` property be set to **true**. To do so, add the JVM property `"-Djava.net.preferIPv4Stack=true"` in the `setenv.sh` file in all nodes.

9b Add "`-Dcom.novell.afw.wf.engine-id=`*`Engine`*" in the `setenv.sh` file in the first node.

The engine name should be unique. Provide the name that was given during the installation of the first node. The default name is "Engine" in case no name was specified.

Similarly, add a unique engine name for other nodes in the cluster. For example, for second node, the engine name can be Engine2.

19 Uninstalling Identity Manager Components

This section describes the process for uninstalling the components of Identity Manager. Some components have prerequisites for uninstallation. Ensure that you review full section for each component before beginning the uninstallation process.

NOTE: Ensure that you perform the following actions before starting the uninstallation process for Identity Manager components:

- Stop Tomcat, PostgreSQL, and ActiveMQ services.
- Take a backup of the install log files from the `/var/opt/netiq/idm/log/` directory.

Removing Objects from the Identity Vault

The first step in uninstalling Identity Manager is to delete all Identity Manager objects from the Identity Vault. When the driver set is created, the wizard prompts you to make the driver set a partition. If any driver set objects are also partition root objects in eDirectory, the partition must be merged into the parent partition before you can delete the driver set object.

To remove objects from the Identity Vault:

1. Perform a health check on the eDirectory database, then fix any errors that occur before proceeding.

   For more information, see "Keeping eDirectory Healthy" in the *NetIQ eDirectory Administration Guide*.

2. Log in to iManager as an administrator with full rights to the eDirectory tree.
3. Select **Partitions and Replica > Merge Partition**.
4. Browse to and select the driver set object that is the partition root object, then click **OK**.
5. Wait for the merge process to complete, then click **OK**.
6. Delete the driver set object.

   When you delete the driver set object, the process deletes all the driver objects associated with that driver set.

7. Repeat Step 3 through Step 6 for each driver set object that is in the eDirectory database, until they are all deleted.
8. Repeat Step 1 to ensure that all merges completed and all of the objects have been deleted.

Uninstalling the Identity Manager Engine

The installer provides an uninstallation script for Identity Manager. This script allows you to remove all services, packages, and directories that were created during the installation.

NOTE: Before uninstalling the Identity Manager engine, prepare the Identity Vault. For more information, see "Removing Objects from the Identity Vault" on page 295.

To uninstall Identity Manager Engine:

1. Navigate to the location where you have mounted the iso for installation.
2. From the root directory of the `.iso` file, run the following command:

   ```
   ./uninstall.sh
   ```

3. Specify the value for **Identity Manager Engine**.
4. If you want to uninstall Identity Vault along with the Identity Manager Engine, specify **y** for the **Do you want to deconfigure and uninstall IDVault** parameter and specify the following details:
   - **Identity Vault Tree Name:** Specify the Identity Vault tree name.
   - **Identity Vault Administrator name:** Specify the Identity Vault administrator name.
   - **Identity Vault Administrator password:** Specify the Identity Vault administrator password.

Uninstalling the Identity Applications

1. Navigate to the location where you have mounted the `.iso` for installation.
2. From the root directory of the `.iso` file, run the following command:

   ```
   ./uninstall.sh
   ```

3. Specify the value for **Identity Applications**.

Uninstalling the Identity Reporting Components

You must uninstall the Identity Reporting components in the following order:

1. Delete the drivers. For more information, see "Deleting the Reporting Drivers" on page 297.
2. Delete Identity Reporting. For more information, see "Uninstalling Identity Reporting" on page 297.
3. Delete Sentinel. For more information, see "Uninstalling Sentinel Log Management for IGA" on page 297.

NOTE: To conserve disk space, the installation programs for Identity Reporting do not install a Java virtual machine (JVM). Therefore, to uninstall one or more components, ensure that you have a JVM available and also make sure that the JVM is in the PATH. If you encounter an error during an uninstallation, add the location of a JVM to the local PATH environment variable, then run the uninstallation program again.

Deleting the Reporting Drivers

You can use Designer or iManager to delete the Data Collection and Managed System Gateway drivers.

1. Stop the drivers. Depending on the component that you use, complete one of the following actions:
   - **Designer:** For each driver, right-click the driver line, then click **Live > Stop Driver**.
   - **iManager:** On the Driver Set Overview page, click the upper right corner of each driver image, then click **Stop Driver**.
2. Delete the drivers. Depending on the component that you use, complete one of the following actions:
   - **Designer:** For each driver, right-click the driver line, then click **Delete**.
   - **iManager:** On the Driver Set Overview page, click **Drivers > Delete drivers**, then click the driver that you want to delete.

Uninstalling Identity Reporting

Before deleting Identity Reporting, ensure you have deleted the Data Collection and Managed System Gateway drivers. For more information, see "Deleting the Reporting Drivers" on page 297.

1. Navigate to the location where you have mounted the `.iso` for installation.
2. From the root directory of the `.iso` file, run the following command:

   ```
   ./uninstall.sh
   ```
3. Specify the value for **Identity Reporting**.

Uninstalling Sentinel Log Management for IGA

1. Log in to the Sentinel server.
2. Navigate to the directory containing the uninstallation script:

   ```
   /opt/novell/sentinel/setup/
   ```
3. Execute the following command:

   ```
   ./uninstall.sh
   ```
4. When prompted to reconfirm that you want to proceed with the uninstall, press `y`.

   The script first stops the service and then removes it completely.

Uninstalling Designer

1. Close Designer.
2. Uninstall Designer.

   Navigate to the directory containing the uninstallation script, by default `<installation_directory>/designer/UninstallDesigner/Uninstall Designer for Identity Manager`.

   To execute the script, enter `./uninstall`

Uninstalling Analyzer

1. Close Analyzer.
2. Uninstall Analyzer according to the operating system:

   Navigate to the `Uninstall Analyzer for Identity Manager` script, located by default in the `<installation_directory>/analyzer/UninstallAnalyzer` directory.

   To execute the script, enter `./Uninstall`

20 Troubleshooting

This section provides useful information for troubleshooting problems with installing Identity Manager. For more information about troubleshooting Identity Manager, see the guide for the specific component.

Locating Log Files

Identity Manager maintains log files to help with debugging any issues. The log files are located at the following locations:

- Install log files for all Identity Manager components: `/var/opt/netiq/idm/log/idminstall.log`
- Configure log files for all Identity Manager components: `/var/opt/netiq/idm/log/idmconfigure.log`
- User Application: `/opt/netiq/idm/apps/tomcat/logs/catalina.out` and `/opt/netiq/idm/apps/tomcat/logs/idapps.out`
- Identity Reporting: `/opt/netiq/idm/apps/tomcat/logs/localhost.<date>.log`
- DCS: `/opt/netiq/idm/apps/tomcat/logs/catalina.out`
- eDirectory: `/var/opt/novell/eDirectory/log/ndsd.log`
- iManager: `/var/opt/novell/tomcat/logs/catalina.out`
- Tomcat: `/opt/netiq/idm/apps/tomcat/logs/catalina.out`
- OSP: `/opt/netiq/idm/apps/tomcat/logs/osp-idm.log`
- SSPR: `/opt/netiq/idm/apps/tomcat/logs/catalina.out`
- Sentinel Log Management for IGA: `/var/opt/novell/sentinel/log/server0.0log`

Troubleshooting Identity Manager Engine

The following table lists the issues you might encounter and the suggested actions for working on these issues. If the problem persists, contact your NetIQ representative.

Issue	Suggested Actions
While installing Identity Manager Engine on OES, the following message is reported on the console and in the `idminstall.log` file located at `/var/opt/netiq/idm/log/` directory.	Ensure that the `/etc/OES-brand` file exists on the OES server. If the file is not present, create a new file and try installing Identity Manager Engine again.

Issue	Suggested Actions
The Identity Manager Engine fails to start when the eDirectory initialization process is in progress. This issue is mostly observed when the eDirectory DIB is very large.	Perform the following steps to workaround this issue: 1. Navigate to the `/etc/opt/novell/eDirectory/conf/` directory. 2. In the `env_idm` file, add the **SLEEP_BEFORE_ENGINE_STARTUP** environment variable and set the value of the variable from 0 to 600. The value is denoted in seconds. **NOTE:** If you provide an invalid value or a value greater than 600, the value defaults to 600. 3. Restart eDirectory. 4. (Conditional) Check the `ndsd.log` to view the messages and logs.
When you run Identity Manager Engine on Linux systems, the `/tmp` directory runs out of disk space in spite of the available space. You can check this status using `df` (disk free) and `du` (disk used) commands. The `df` command shows no available space while the `du` command shows that not all the space allocated for `/tmp` is used. This issue occurs because every Identity Manager driver that is instantiated loads several libraries in the memory. The JVM temporarily copies these drivers to `/tmp` directory and then deletes them. The deleted files continue to use the memory until the JVM process that created those files is terminated. You can use the `lsof` command to determine this behavior. Files in this state are marked as deleted. The total disk space consumed depends on the number of drivers running on the server.	The space consumed is relatively static. Therefore, ensure that you provide sufficient extra space in `/tmp` directory. If the issue persists, restart eDirectory.
In a multi-server environment, an unrecognized extended exception is displayed.	Ensure that the primary server has a read-write partition for the secondary server: 1. Log in to iManager. 2. Click **Roles and Tasks** > **Partitions and Replicas** > **Replica View**. 3. Select the secondary server. 4. Assign read-write permissions to the server. **NOTE:** Ensure that you have added the secondary server in the driver set.

Issue	Suggested Actions
When you execute a large query through dxcmd, dxcmd exits with a 625 `ERR_TRANSPORT_FAILURE` error and no result file is generated. The Identity Manager engine uses the environment variable `NCPCLIENT_REQ_TIMEOUT` value as the default time to execute a dxcmd query. By default a NCP connection has a timeout of 115 seconds. If the time taken for executing the query and returning the result exceeds this value, the error will be seen.	Increase the timeout value by setting the environment variable `NCPCLIENT_REQ_TIMEOUT` to a number of seconds larger than the total time the query takes. Setting the environment variable permanently for dxcmd can be accomplished by adding export `NCPCLIENT_REQ_TIMEOUT=value` to the dxcmd script `/opt/novell/eDirectory/bin/dxcmd`. It is also possible to set the variable manually in the terminal from which the script is being executed by running `export NCPCLIENT_REQ_TIMEOUT=value` prior to executing dxcmd.

Troubleshooting the Identity Applications and Identity Reporting

The following table lists the issues you might encounter and the suggested actions for working on these issues. If the problem persists, contact your NetIQ representative.

Issue	Suggested Actions
If the LDAP Server Name specified in the Certificate Subject and the Application Configuration are different, the Identity Applications fails to connect to the Identity Vault after upgrading Identity Manager.	Java enables endpoint identification on LDAPS connections and thus mandates that the server name that you specify while connecting to the Identity Manager server and the server name returned in the certificate are the same. If the server names are different, perform the following steps: 1. Navigate to the `/opt/netiq/idm/apps/configupdate` directory. 2. Run the following command to launch the Configuration Update utility. `./configupdate.sh` 3. Navigate to the **User Application** tab, click **Identity Vault server**, and change the server name to the one specified in the LDAP server certificate subject. This action will update the `DirectoryService/realms/jndi/params/AUTHORITY property` in the `ism-configuration.properties` file. 4. Click **OK**.

Issue	Suggested Actions
When Identity Applications and Identity Reporting are installed on the same server and you perform configuration changes using the configuration update utility located at `<reporting install folder>/bin` directory, the Identity Manager Dashboard fails to launch. The following error is reported in `catalina.out` log file: `EboPortalBootServlet [RBPM] +++++WARNING!!!!: This portal application context, IDMProv, does not match the portal.context property set in the PortalService-conf/config.xml file. Only one portal per database is allowed. Data has been loaded using the previous portal context. To correct this you must revert back to the previous portal name of, NoCacheFilter, please consult the documentation.`	For any configuration changes, use the configuration update utility located at `/opt/netiq/idm/apps/configupdate/` directory.
Cannot change the password of a property using the configuration update utility.	You can change the password of a property, for example `com.netiq.rpt.ssl-keystore` from the command line by performing the following steps: 1. Use the below utility to encrypt your password: `/opt/netiq/common/jre/bin/java -jar tomcat/lib/obscurity-0.7.0-uber.jar <<Password>>` 2. Navigate to the `ism-configuration.properties` file located at `/opt/netiq/idm/apps/tomcat/conf/` directory. 3. Modify the `ism-configuration.properties` file and add the encrypted password specified in step 2 for the `com.netiq.rpt.ssl-keystore.pwd` parameter. 4. Save the file and restart Tomcat.

Issue	Suggested Actions
If Identity Reporting is installed on a standalone server and you launch the Identity Reporting or the IDM DCS URL from the dashboard, then the URL fails to launch.	Perform the following steps after you launch the Identity Reporting or IDM DCS URL: 1. Navigate to the address bar. 2. Modify the URL and manually provide the host name and port details of the server where Identity Reporting is installed. In addition to that, navigate to the `configupdate.sh.properties` file on the server where Identity Applications is installed and add the entry called `rpt` in the `sso_apps` parameter and then save the changes. For example, **sso_apps=ua,rpt**
If Identity Applications and Identity Reporting are installed on the same server and CEF auditing is enabled for OSP and Identity Applications, then the Reporting component fails to launch.	Perform the following steps to workaround this issue: 1. Navigate to the `idmrptcore_logging.xml` file located at `/opt/netiq/idm/apps/tomcat/conf` directory. 2. Add the `<keystore file>` parameter and specify the keystore file path in the `idmrptcore_logging.xml` file. For example, add the following line: `<keystore-file>/opt/netiq/idm/app/tomcat/conf/idm.jks</keystore-file>` 3. Restart Tomcat.
If your Identity Applications and Identity Reporting are installed on the same server and you choose the database creation option as **Startup**, you will notice some exceptions in the log.	To clear the exceptions, manually restart Tomcat.
If your existing Identity Applications or Identity Reporting configuration has been configured without ports, and you try to upgrade to Identity Manager, the IP address and ports mentioned under the **Authentication** and **SSO Clients** tab in the configuration update utility displays incorrect values.	Once you upgrade Identity Applications and Identity Reporting, perform the following steps: 1. Navigate to the `/opt/netiq/idm/apps/configupdate` directory. 2. Run the following command: `./configupdate.sh` 3. In the **Authentication** tab, specify the correct IP address and port in the **OAuth server host identifier** and **OAuth server TCP port** fields respectively. 4. In the **SSO Clients** tab, ensure that URLs for IDM Administrator, Reporting, and IDM Data Collection Services are in correct format. 5. Restart Tomcat.

Issue	Suggested Actions
You want to modify one or more of the following the User Application configuration settings created during installation: ◆ Identity Vault connections and certificates ◆ E-mail settings ◆ Identity Manager Engine User Identity and User Groups ◆ Access Manager or iChain settings	Run the configuration utility independent of the installer. **Linux**: Run the following command from the installation directory (by default, `/opt/netiq/idm/apps/configupdate/`): `./configupdate.sh`
Starting Tomcat causes the following exception: `port 8180 already in use`	Shut down any instances of Tomcat (or other server software) that might already be running. If you reconfigure Tomcat to use a port other than 8180, edit the `config` settings for the User Application driver.
When Tomcat starts, the application reports it cannot find trusted certificates.	Ensure that you start Tomcat by using the JDK specified during the installation of the User Application.
Cannot log in to the portal admin page.	Ensure that the User Application Administrator account exists. This account is not the same as your iManager administrator account.
Cannot create new users even with administrator account.	The User Application Administrator must be a trustee of the top container and should have Supervisor rights. You can try setting the User Application Administrator's rights equivalent to the LDAP Administrator's rights (using iManager).
Starting application server throws keystore errors.	Your application server is not using the JDK specified during the installation of the User Application. Use the `keytool` command to import the certificate file: `keytool -import -trustcacerts -alias aliasName -file certFile -keystore ..\lib\security\cacerts -storepass changeit` ◆ Replace *aliasName* with a unique name of your choice for this certificate. ◆ Replace *certFile* with the full path and name of your certificate file. ◆ The default keystore password is `changeit` (if you have a different password, specify it).

Issue	Suggested Actions
Email notification not sent.	Run the `configupdate` utility to check whether you supplied values for the following User Application configuration parameters: **Email From** and **Email Host**. **Linux**: Run the following command from the installation directory (by default, `/opt/netiq/idm/apps/configupdate/`): `./configupdate.sh`
IG SSO Clients tab is not seen in the configuration update utility	In the `configupdate.sh.properties` file, add an entry for `ig` in the `sso_apps` parameter and then save the changes. If the `sso_apps` parameter already contains the Identity Applications and Identity Reporting entries, add the Identity Governance entry to the list. For example, **sso_apps=ua,rpt,ig**

Troubleshooting Login

The following table lists the issues you might encounter and the suggested actions for working on these issues. If the problem persists, contact your NetIQ representative.

Issue	Suggested Actions
Launching Designer displays the following error and the Designer readme will not be shown: `Exception... "Update manifest is missing a required addons property."`	Ignore the error and launch Designer. There is no functionality loss.
User is unable to login in large scale environment (>2 million objects)	Add an index for `mail(Internet Mail Address)` attribute with the rule set as `Value` in both eDirectory master and replica servers.
When you sign out from Identity Applications page, SSPR shows an error `5053 ERROR_APP_UNAVALIABLE`.	Ignore this error. It does not cause any functionality loss.

Issue	Suggested Actions
Challenge Responses are not prompted at the first login to the Identity Applications	1. Ensure that the SSPR server has a certificate created using FQDN. 2. Log in to the User Application server and launch ConfigUpdate (`/opt/netiq/idm/apps/configupdate/`)utility. 3. Navigate to **SSO Clients > Self Service Password Reset** and make sure the settings are correct. If SSPR is installed on a separate server, make sure that the SSPR certificate is imported into `idm.jks` located in the User Application server at `/opt/netiq/idm/apps/tomcat/conf`.

Issue	Suggested Actions
Browser displays a blank page when SSPR URL is accessing	This occurs when SSPR is not properly configured with OSP. The SSPR log shows the following information: `2018-01-24T22:24:02Z, ERROR, oauth.OAuthConsumerServlet, 5071 ERROR_OAUTH_ERROR (unexpected error communicating with oauth server: password.pwm.error.PwmUnrecoverableException: 5071 ERROR_OAUTH_ERROR (io error during oauth code resolver http request to oauth server: Certificate for <IP> doesn't match any of the subject alternative names: [IP]))` 1. Verify that the Tomcat server where OSP is running has a valid certificate created using FQDN. Log in to the User Application server and launch ConfigUpdate utility. Navigate to **SSO Clients > Self Service Password Reset** and make sure the settings are correct. 2. Log in to SSPR by overriding the OSP login method. (for example, `https://<sspr sserver ip>:<port>/sspr/private/Login?sso=false`) 3. Navigate to **Configuration Editor** in the top right corner of the page. 4. Specify **Configure Password**, then click **Sign In**. 5. Navigate to **LDAP > LDAP Directories > Default > Connection**. 6. If the LDAP certificate is not correct, click **Clear**. 7. To reimport the certificate, click **Import From Server**. 8. Navigate to **Settings >Single Sign On (SSO)Client > OAuth** and verify that the certificate under **OAUTH Web Service Server Certificate** is correct. 9. If the certificate is not correct, click **Clear**. 10. To reimport the certificate, click **Import From Server**.
Error when ConfigUpdate utility is launched from a different directory	The ConfigUpdate utility reports errors. It does not save any changes. For example, if you launch the configupdate utility using the `/opt/netiq/idm/apps/configupdate/configupdate.sh` command, it does not launch. Instead, navigate to the `/opt/netiq/idm/apps/configupdate/` directory and then run `./configupdate.sh` command.

Troubleshooting Installation and Uninstallation

The following table lists the issues you might encounter and the suggested actions for working on these issues. If the problem persists, contact your NetIQ representative.

Issue	Suggested Actions
You are unable to access Identity Manager Dashboard after restarting the system, where Identity Manager 4.8.x is installed on RHEL 8 operating system. The `catalina.log` file contains the following warning message: `WARN [com.novell.soa.af.impl.core.WorkflowEngineImpl] (main) [WORKFLOW] Duplicate engine id detected. This engine may not have been shutdown cleanly or another engine is running with engine-id: ENGINE . Waiting 60000 ms for heartbeat to timeout.` This issue occurs if the Tomcat service is not stopped before restarting the system. When the workflow engine starts up, it encounters another node in the cluster with an engine ID that is already in use and logs a warning.	It is always recommended to stop the Tomcat service before restarting the system. If you do encounter the warning, you can resolve it by changing the engine status in the afenginestate table in the `igaworkflowdb` database. Log in to a database admin tool such as pgAdmin and manually update the engine status to shutdown state.
If you are installing Identity Manager 4.8 on RHEL 8.3, the PostgreSQL service does not start correctly.	Perform the following steps: 1. Install Identity Manager 4.8 on RHEL 8. 2. Upgrade the RHEL OS version to 8.3. 3. Upgrade Identity Manager version to 4.8.3.

Tomcat and ActiveMQ services are in disabled state when you have installed Identity Manager on SLES 15 or SLES 15 SPx (where x denotes all SLES 15 versions supported with Identity Manager).	Perform the following steps:
	1. Log in to the server where Tomcat and ActiveMQ services are disabled.
	2. Install the `insserv-compat*` RPM. The * symbol denotes the latest version of the RPM.
	NOTE: NetIQ recommends you to obtain the dependent packages from your operating system subscription service to ensure continued support from your operating system vendor. If you do not have a subscription service, you can find the recent packages from a website such as http://rpmfind.net/linux.
	3. Run the following commands to enable the Tomcat and ActiveMQ services:
	`systemctl enable netiq-tomcat.service`
	`systemctl enable netiq-activemq.service`

Issue	Suggested Actions

If you are installing Identity Manager 4.8 on RHEL 8.1, the following error messages are displayed when:
◆ you run the `RHEL-Prerequisite.sh` script:
`dnf-utils rpm required but not found`
`RHEL 8.1 iso does not contain dnf-utils packages, its replaced by yum-utils`
◆ you install the Identity Manager Engine component:
`Some of the dependencies required for installation of Identity Vault are not found`
	Ignore the error messages and proceed with the installation process.

After you upgrade Identity Applications to 4.8 version, the workflow forms fail to render on the Identity Applications. The client password (CientPass) present in `/opt/netiq/idm/apps/sites/config.ini` is left blank at some scenarios.	To resolve this issue, perform the following steps: 1. Navigate to the `/opt/netiq/idm/apps/sites/` directory. 2. Edit the config.ini fle and provide the encoded base64 password for the `ClientPass` parameter. **NOTE:** Specify the same password that you specified for the **forms client** in the configuration update utility. 3. Restart the NGINX service. `systemctl restart netiq-nginx.service` For more information, see TID 7024492.
In a container environment, if you have exceeded the idle time set for the Identity Manager dashboard and the session times out, the **Extend** button does not work as expected.	Add the `com.netiq.idm.session-timeout` property in the `ism-configuration.properties` file of the OSP container.
In a multi-server environment, while trying to deploy a driver to secondary server, LDAP exceptions are displayed.	After the secondary server is installed and partition is added in the Identity Vault for secondary server, you must restart `ndsd` on both the servers. This also applies if you are installing Identity Manager engine in a container deployment.
In a clustered environment, when you disconnect a node from the network (for example, node 2), and then create roles on the active node (for example, node 1), then the newly-added roles are not synchronized on node 2 when it is connected back to the network.	Perform the following steps on node 2: 1. Delete the `.../temp/permindex files`. 2. Restart Tomcat.

Issue	Suggested Actions
During the configuration of Identity Applications, if you want to connect to an SSPR installed on a remote server, the SSPR configuration will be skipped. However, the `sspr.war` that was installed during the installation process will be deployed when you restart Tomcat.	If you want to connect to an SSPR installed on a remote server, remove the `sspr.war` from the `/opt/netiq/idm/apps/tomcat/webapps` directory before configuring Identity Applications.
During the configuration of Identity Applications, if you use a custom sub-container for Identity Applications Administrator, for example, `cn=uaadmin,ou=univ,o=data`, then `uaadmin` will be created under the default container (`ou=sa,o=data`).	Perform one of the following steps if you want to use the custom sub-container for Identity Applications Administrator: ◆ Before configuring Identity Applications, create the `ou=univ,o=data` custom sub-container. ◆ During the configuration process, specify **No** for the **Do you want to use custom container as root container** prompt. The custom LDIF file that you will import should contain the custom root container and the sub-container details.
The silent installation process does not check for the system requirements when the silent properties file is created. During silent installation, the log file displays an error message stating that the system requirements are not met.	When you encounter this issue, manually add the `IS_SYSTEM_CHECK_DONE` parameter in the `silent.properties` file. To skip the system requirement check, set the value for the `IS_SYSTEM_CHECK_DONE` parameter to 1.
When you uninstall and reinstall Identity Applications or Identity Reporting, the configuration process fails when setting up database users and schema. This issue is observed when you perform a typical configuration during the re-installation of the component.	When you are reinstalling the Identity Applications or Identity Reporting component, you must perform a custom configuration.
Uninstallation process reports as incomplete but the log file shows no failures.	The process failed to delete the `netiq` directory that contains the installation files by default. You can delete the directory if you have removed all NetIQ software from your computer.

Troubleshooting Upgrade

The following table lists the issues you might encounter and the suggested actions for working on these issues. If the problem persists, contact your NetIQ representative.

Issue	Suggested Actions
Identity Manager authorizes and securely communicates with its components using digital certificates. The Identity Vault certificates must be imported into the `idm.jks` and `tomcat.ks` keystore files. However, when attempting to access Identity Applications after importing the certificates, you might hit the following error: `javax.net.ssl.SSLHandshakeException: PKIX path validation failed: java.security.cert.CertPathValidatorException: Could not determine revocation status.` The certificates are validated by checking the Certificate Revocation Lists (CRLs) specified by the CRL Distribution Point (CDP) field to determine whether the certificate has been revoked or not. The CRLDPs are available in both the root certificate and the intermediate certificates present in the keystore files `tomcat.ks` and `idm.jks`. Certificate revocation checking, however, is disabled by default. As a result, the PKIX trust manager is unable to determine the revocation status of the certificates.	To fix this issue, enable CRL distribution point checking by setting the `-Dcom.sun.security.enableCRLDP` property to `true`. To set the property, perform the following actions: 1. Stop Tomcat. 2. Go to the `setenv.sh` file located in the Tomcat's `bin` directory. For example, `/opt/netiq/idm/apps/tomcat/bin/setenv.sh`. 3. Add the property `-Dcom.sun.security.enableCRLDP=true` in CATALINA_OPTS as: `export CATALINA_OPTS="-Dcom.sun.security.enableCRLDP=true"` 4. Start Tomcat.
After upgrading Identity Manager, logging in to Identity Manager Dashboard is extremely slow for non-admin users. There is a significant delay in loading the Applications and the Dashboard pages. This issue occurs due to the nested group search, which is enabled by default. The application will look for the permissions inherited by the logged-in user via the nested group membership, regardless of whether there are any nested groups in the environment.	(Conditional) The following steps apply to Identity Manager 4.8.5 and later. 1. Log in to the server where Identity Applications is upgraded to 4.8.5 version. 2. Navigate to the `/opt/netiq/idm/apps/tomcat/conf/` location. 3. Open the `ism-configuration.properties` file in a text editor. 4. At the end of the file, add the following property: `DirectoryService/realms/jndi/params/USE_NESTED_GROUPS=false` 5. Save the file and restart Tomcat.

Issue	Suggested Actions
After upgrading Identity Applications to 4.8 version from a prior version, the Form Renderer does not work as expected. This issue is observed when the default IDMProv deployment context is modified to a custom context.	To work around this issue, perform the following steps: 1. Log in to the server where Identity Applications is upgraded to 4.8 version. 2. Navigate to the `/opt/netiq/idm/apps/sites` directory. 3. Edit the `ServiceRegistry.json` file. 4. Modify the deployment context from `IDMProv` to the custom context that was specified prior to upgrade. 5. Save the `ServiceRegistry.json` file. 6. Navigate to the `/opt/netiq/idm/apps/sites/forms/` directory. 7. Edit the `main.<version>.js` file, where *<version>* is the randomly generated alphanumeric value. 8. Modify the deployment context from `IDMProv` to the custom context that was specified prior to upgrade. 9. Save the `main.<version>.js` file. 10. Restart Tomcat.

After you upgrade Identity Manager in a distributed environment to 4.8.1 version, login to the Identity Applications fails. The following error message is displayed:
`Your login process did not complete successfully.`
Logging to the Identity Applications requires trust anchor certificates for establishing a secure connection between the Identity Applications and the OSP. A trust anchor certificate must include the Basic Constraints extension with the Subject Type set to CA. Identity Manager makes use of the property `jdk.security.allowNonCaAnchor` to validate the trust anchors in the certificate. By default, this property is set to `false`. Therefore, when the trust anchors are not found in the certificates, the connection between Identity Applications and OSP cannot be established and the login fails. You will notice the following exception in the `idm-osp.log` file:
`sun.security.validator.ValidatorException: TrustAnchor with subject "CN=***, L=***, O=***" is not a CA certificate`
	To resolve this issue, you must satisfy either of the following conditions:
	◆ Ensure that the certificates used to establish a secure connection between the Identity Applications and the OSP are trusted CA certificates with proper Basic Constraints extension.
	◆ In case of self signed certificates and custom certificates that are trusted by the clients, you can change the property `jdk.security.allowNonCaAnchor` to allow non CA certificates without Basic Constraints extension. Perform the following actions to modify the Java security settings:
	1. Navigate to the `/opt/netiq/common/jre/lib/security/java.security` directory.
	2. Set the value of the property `jdk.security.allowNonCaAnchor=true`.
	3. Save the file.

Issue	Suggested Actions
After upgrading to Identity Applications 4.8.1 version, you are not able to open forms while requesting for permissions in the Identity Applications Dashboard.	To resolve this issue, manually restart the NGNIX and Golang services using the following commands: ◆ **NGNIX:** `/opt/netiq/common/ngnix/ngnix` ◆ **Golang:** `/etc/init.d/netiq-golang.sh` The Identity Applications uses the NGNIX service for rendering forms in the Identity Applications Dashboard.
After you upgrade Identity Reporting in a standard edition, the **is_prov** parameter in the `configupdate.sh.properties` is set to `true`. Since Identity Applications is not available in a standard edition, the value of this parameter must be set to `false`.	Manually set the **is_prov** parameter to `false` in the `configupdate.sh.properties` file.
Unable to re-run the Identity Manager engine installer if the prior upgrade of Identity Manager Engine fails. For example, if the 4.8 upgrade for Identity Manager Engine fails on the first attempt and you try upgrading Identity Manager Engine again, the upgrade process cannot be triggered.	Perform the following steps: 1. Downgrade the Identity Manager engine to the previous version using the `novell-DXMLengnx` RPM. 2. Upgrade Identity Manager engine.
After you upgrade Identity Manager, the following property is added to the `ism-configuration.properties` file: `com.netiq.idm.osp.ldap.admin-dn = cn=admin,ou=sa,o=system`	Comment out the property in the `ism-configuration.properties` file and restart Tomcat. It does not cause any functionality loss.
After you upgrade Identity Manager, the following SSPR property is added to the `ism-configuration.properties` file, even if you do not have SSPR in your deployment: `com.netiq.sspr.redirect.url = https://___SSPR_IP___:___SSPR_TOMCAT_HTTPS_PORT___/sspr/public/oauth`	Comment out the property in the `ism-configuration.properties` file and restart Tomcat. It does not cause any functionality loss.
After you upgrade Identity Manager, the `ism-configuration.properties` file populates some duplicate values of `java.protocol.handler.pkgs` property.	There is no loss of functionality. To resolve this issue, perform the following actions: 1. Navigate to the `ism-configuration.properties` file located at `/opt/netiq/idm/apps/tomcat/conf/`directory. 2. Modify the `ism-configuration.properties` file and remove the duplicate values of the `java.protocol.handler.pkgs` property. 3. Save the file and restart Tomcat.

Issue	Suggested Actions
Unable to start Tomcat after Identity Manager upgrade. You will notice few exceptions in tomcat logs and a communication failure between the workflow engine and the Identity Vault.	1. Log in to iManager. 2. Navigate to **Roles and Tasks** > **NetIQ Certificate Access** > **Server Certificates**. 3. Select the **SSL CertificateDNS** check box and click **Export**. 4. In the **Certificates** drop-down list, select the *SSL CertificateDNS*. 5. Clear the **Export private key** check box. Ensure that the **Export format** is set to *DER*. 6. Click **Next** > **Save the exported certificate** to download the certificate in your system. 7. Log in to the Identity Applications server. 8. Stop Tomcat. 9. Navigate to `opt/netiq/common/jre/bin` directory and import the certificate to `idm.jks` file using the following command: `opt/netiq/common/jre/bin/keytool -import -trustcacerts -alias <certificate_alias_name> -keystore /opt/netiq/idm/apps/tomcat/conf/idm.jks -file <certificate_file_downloaded>` 10. Restart Tomcat.