Release Notes

/ ForgeRock Identity Management 6.5

Latest update: 6.5.2.0

Mark Craig
Lana Frost
Mike Jang
Andi Egloff

ForgeRock AS
201 Mission St., Suite 2900
San Francisco, CA 94105, USA
+1 415-599-1100 (US)
www.forgerock.com

Abstract

Notes covering ForgeRock® Identity Management software requirements, fixes, and known issues. This software offers flexible services for automating management of the identity life cycle.

Table of Contents

About ForgeRock Identity Management Software

ForgeRock Identity Platform™ serves as the basis for our simple and comprehensive Identity and Access Management solution. We help our customers deepen their relationships with their customers, and improve the productivity and connectivity of their employees and partners. For more information about ForgeRock and about the platform, see https://www.forgerock.com.

The ForgeRock Common REST API works across the platform to provide common ways to access web resources and collections of resources.

ForgeRock Identity Management software provides centralized, simple management and synchronization of identities for users, devices and things.

ForgeRock Identity Management software is highly flexible and therefore able to fit almost any use case and workflow.

These release notes are written for anyone using the ForgeRock Identity Management 6.5 release. Read these notes before you install or upgrade ForgeRock Identity Management software.

These release notes cover the following topics:

- A list of the major new features and functionality provided with this release
- Hardware and software prerequisites for installing and upgrading ForgeRock Identity Management software
- Compatibility with previous releases
- Potential upcoming deprecation and removals that affect scripts and applications
- Issues fixed since the previous release
- Known issues open at the time of release

For installation instructions, see "*Preparing to Install and Run Servers*" in the *Installation Guide*.

Several samples are provided to familiarize you with the IDM features. For more information, see "*Overview of the Samples*" in the *Samples Guide*.

For an architectural overview and a high-level presentation of IDM, see "*Architectural Overview*" in the *Integrator's Guide*.

Chapter 1 What's New

This chapter covers new capabilities in the current release of ForgeRock Identity Management.

1.1. Maintenance Releases

IDM 6.5.2.0

IDM 6.5.2.0 is the latest maintenance release targeted for IDM 6.5 deployments. Download this release from the ForgeRock Download Center. For a list of fixes in this release, see Key Fixes in IDM 6.5.2.0.

You can deploy IDM 6.5.2.0 as an initial deployment, or update from an existing 6.5 deployment. Download IDM 6.5 from the ForgeRock Download Center.

1.2. Patch Bundle Releases

ForgeRock patch bundle releases contain a collection of fixes and minor RFEs that have been grouped together and released as part of our commitment to support our customers. For general information on ForgeRock's maintenance and patch releases, see Maintenance and Patch Availability Policy.

1.3. New Features

This release of ForgeRock Identity Management software includes the following new features:

IDM 6.5.2.0

There are no new features in this release, only bug fixes.

IDM 6.5.1.0

Support for *Sign in with Apple*

IDM 6.5.1.0 supports social registration through Apple. For more information, see "Set Up Apple as an IDM Social Identity Provider" in the *Integrator's Guide*.

IDM 6.5.0.4

There are no new features in this release, only bug fixes.

IDM 6.5.0.3

There are no new features in this release, only bug fixes.

IDM 6.5.0.2

There are no new features in this release, only bug fixes.

IDM 6.5.0.1

There are no new features in this release, only bug fixes.

IDM 6.5.0

- Delegated Administration Capabilities

  IDM 6.5.2 supports delegated administration, through a privilege model. For more information, see "Privileges and Delegation" in the *Integrator's Guide*.

- New End User UI

  IDM 6.5.2 includes an End User UI based on the Vue JavaScript framework. To facilitate customization, ForgeRock has made the End User UI code available in the following public Git repository: *Identity Management (End User) - UI* .

  You can customize the default End User UI, and create your own End User UIs, based on the code in this Git repository.

  Only one registration flow is provided by default. You can set up separate registration flows for different sets of users (such as employees and contractors), as described in "Configuring Multiple User Self-Registration Flows" in the *Integrator's Guide*.

  Important

  The default workflows provided with IDM have been rewritten to use the Vue JS framework. Previously, these workflows used JQuery and Handlebars. If your deployment includes existing workflows, you *must* rewrite these to use Vue JS if you want to view them in the new End User UI. The new UI does not support older workflow templates that use JQuery and Handlebars.

  To rewrite existing workflows for the new UI, you must have a basic understanding of the Vue JS framework and how to create components. For more information, see the Vue documentation. For an example of a workflow template written for the new UI, see `/path/to/samples/provisioning-with-workflow/workflow/contactorOnboarding.bar`. This archive file includes the workflow definition (`contactorOnboarding.bpmn20.xml`) and the corresponding JavaScript template (`contractorForm.js`) to render the workflow in the new UI.

> If you previously generated your workflows with a bpmn file (and never created custom JavaScript files), the new UI will just generate these as before and you will not have to convert them.

- Keystores and Truststores now configured through the Secrets Service

  The configuration keystores and truststores are now managed by a new IDM secrets service. You can modify secrets through the `secrets.json` file in your project's `conf/` subdirectory. The secrets service also supports key rotation, which means the active key may not be what's used to decrypt information.

  In addition, each alias in `secrets.json` now has a dedicated capability and function. For more information, see "Accessing IDM Keys and Certificates" in the *Integrator's Guide*.

- Oracle Database Universal Connection Pool (Oracle UCP)

  IDM now supports Oracle UCP as an alternative to the default HikariCP connection pool library, solely for an Oracle DB. For more information, see "Setting Up an Oracle DB Repository" in the *Installation Guide*.

- JSON Standard Output Audit Event Handler

  IDM now supports sending log messages to standard output in the OSGi console.

  For details, see "JSON Standard Output Audit Event Handler" in the *Integrator's Guide*.

- New Notification Service

  IDM now includes a dedicated customizable notification service that sends messages as configured. Notifications are no longer configured in the `onUpdateUser.js` script, but are shown in dedicated `notification-*.json` files. For more information, see "Configuring Notifications" in the *Integrator's Guide*.

- New HubSpot Connector and Sample

  IDM 6.5.2 supports a new HubSpot connector, available from the ForgeRock BackStage download site:

  For more information, see "*HubSpot Connector*" in the *Connector Reference*. To help you get started with this connector, see "*Synchronizing Data Between IDM and HubSpot*" in the *Samples Guide*.

1.4. Product Enhancements

IDM 6.5.2.0

Add support for CLOUDHSM as a keystore type

The CLOUDHSM keystore type is now available as a possible keystore type. See "Configuring IDM to Support an HSM Provider" in the *Integrator's Guide* for more information on configuring IDM to work with an HSM.

Add support for boolean column types in explicit mappings

boolean is now a supported column type when configuring explicit mappings.

IDM 6.5.1.0

Removal of policy validation on hashed passwords

OPENIDM-11456 : Policy validation is no longer applied to hashed values because it is not possible to inspect and apply validation to the clear text value.

Improved security in the default log message format

OPENIDM-15100 : The default log message formatter has changed from ThreadIdLogFormatter to SanitizedThreadIdLogFormatter. The new default encodes control characters (such as newline characters) using URL-encoding, to protect against log forgery. Control characters in stack traces are not encoded. For more information, see "Set the Log Message Format" in the *Integrator's Guide*.

Change in how boolean values are assessed

OPENIDM-15517 : Properties stored in the repository with boolean (true/false) values are processed differently from IDM 6.5.1.0. A property value is now considered false if its value is false or null. The value is considered true only if it is true, not if it is null. If you are migrating from a previous IDM release, you might need to adjust your scripts to take this change into account

IDM 6.5.0.4

New Workflow Logging Capabilities

IDM has added new logging workflow capabilities to indicate who approved an action and when.

For example, a user1 is logged as the person who onboarded a contractor:

```
{
   "_id":"f24ac83b-200c-449d-b017-d12b9c6c9091-3871",
   "timestamp":"2020-05-06T17:39:52.021Z",
   "eventName":"workflow-create_process",
   "transactionId":"f24ac83b-200c-449d-b017-d12b9c6c9091-3865",
   "userId":"user1",
   "runAs":"user1",
   "objectId":"workflow/processinstance/6",
   "operation":"CREATE",
   "changedFields":[

   ],
   "revision":null,
   "status":"SUCCESS",
   "message":"Process created. processDefinitionId = contractorOnboarding:1:5, processDefinitionKey =
 null, businessKey = null",
   "passwordChanged":false
}
```

`manager1` assigns the task to herself. This event is recorded in the `"changedFields":["/assignee"]` field.

```
{
   "_id":"f24ac83b-200c-449d-b017-d12b9c6c9091-5748",
   "timestamp":"2020-05-06T17:43:18.058Z",
   "eventName":"workflow-update_task",
   "transactionId":"f24ac83b-200c-449d-b017-d12b9c6c9091-5744",
   "userId":"manager1",
   "runAs":"manager1",
   "objectId":"workflow/taskinstance/36",
   "operation":"UPDATE",
   "changedFields":[
      "/assignee"
   ],
   "revision":null,
   "status":"SUCCESS",
   "message":"Task updated",
   "passwordChanged":false
}
```

When the `manager1` completes the task. The `transactionID` is correlated with all `manager/user` operations.

```
{
   "_id":"f24ac83b-200c-449d-b017-d12b9c6c9091-5868",
   "timestamp":"2020-05-06T17:43:22.138Z",
   "eventName":"activity",
   "transactionId":"f24ac83b-200c-449d-b017-d12b9c6c9091-5838",
   "userId":"manager1",
   "runAs":"manager1",
   "objectId":"managed/user/d736487d-c146-4a0e-b677-ebfd6805b1d2",
   "operation":"CREATE",
   "changedFields":[

   ],
   "revision":"000000001edd9dc2",
   "status":"SUCCESS",
   "message":"create",
   "passwordChanged":false
}
```

The `Task completed` event is logged:

```
{
   "_id":"f24ac83b-200c-449d-b017-d12b9c6c9091-5926",
   "timestamp":"2020-05-06T17:43:22.827Z",
   "eventName":"workflow-complete_task",
   "transactionId":"f24ac83b-200c-449d-b017-d12b9c6c9091-5838",
   "userId":"manager1",
   "runAs":"manager1",
   "objectId":"workflow/taskinstance/36",
   "operation":"complete",
   "changedFields":[

   ],
   "revision":null,
   "status":"SUCCESS",
   "message":"Task completed",
   "passwordChanged":false
}
```

IDM 6.5.0.3

In Windows deployments, the IDM code has been fixed to look for `jvm.dll` files to support either Java 8 or Java 11. However, switching between Java 8 and Java 11 can break the Windows service. Therefore, if you are using Java 8 and want to move to Java 11, uninstall the Windows service using **server.bat /uninstall openidm** on the Java 8 installation, and reinstall using **server.bat /install openidm** on the Java 11 installation. For more information, see "Installing as a Windows Service" in the *Installation Guide*.

IDM 6.5.0.2

Signout Works Properly When Access Token has Expired

IDM 6.5.0.2 has improved the signout process to work properly when an AM access token has become invalid or expired.

IDM 6.5.0.1

There are no product enhancements in this release, other than bug fixes.

IDM 6.5.0

.NET Connector Server Now Uses WCF by Default

On Windows 10, 2012, and 2016, the .NET connector server now uses Windows Communication Foundation (WCF) as the default WebSockets library, instead of Vtortola. Vtortola is still the default library on Windows 2008.

Synchronization Performance Improvements

IDM now supports asynchronous (queued) synchronization for implicit synchronization operations. For more information, see "Queued Synchronization" in the *Integrator's Guide*.

Improved Connectors and Samples

The Salesforce Connector has been rewritten as a standard ICF connector, rather than a separate IDM module. For more information, see "*Salesforce Connector*" in the *Connector Reference*.

The SCIM sample (`samples/sync-with-scim/`) has been revised. For more information, see "*Synchronizing Data Between IDM and a SCIM Provider*" in the *Samples Guide*.

Java Support

IDM software now supports Java 8 and Java 11.

Ability to Encrypt/Decrypt over REST

You can now use the `?_action=eval` option on the `script` endpoint. For more information, see "Encrypting and Decrypting Information" in the *Integrator's Guide*.

1.5. Security Advisories

ForgeRock issues security advisories in collaboration with our customers and the open source community to address any security vulnerabilities transparently and rapidly. ForgeRock's security advisory policy governs the process on how security issues are submitted, received, and evaluated as well as the timeline for the issuance of security advisories and patches.

For details of all the security advisories across ForgeRock products, see Security Advisories in the *Knowledge Base library*.

Chapter 2
Before You Install

This chapter covers requirements to consider before you run ForgeRock Identity Management software, especially before you run the software in your production environment.

If you have a special request to support a component or combination not listed here, contact ForgeRock at info@forgerock.com.

2.1. Supported Repositories

The following repositories are supported for use in production:

- ForgeRock Directory Services (DS) 6.5

  By default, IDM uses an *embedded* DS instance for testing purposes. The embedded instance is not supported in production. If you want to use DS as a repository in production, you must set up an external instance.

- MySQL version 5.6 and 5.7 with MySQL JDBC Driver Connector/J 5.1.18 or later
- MariaDB version 10.0, 10.1, and 10.2 with MySQL JDBC Driver Connector/J 5.1.18 or later
- Microsoft SQL Server 2012, 2014, and 2016

  > **Warning**
  >
  > For deployments using Microsoft JDBC Driver 7.x for SQL Server with Java 11, see Known Issues IDM 6.5.0.3.

- Oracle Database 11gR2, 12c, 12c Release 1 (12.1), 12c Release 2 (12.2), and 19c
- PostgreSQL 9.3.10, 9.4.5, 9.5, 9.6, and 10.x
- IBM DB2, 10.1, 10.5, 11

ForgeRock supports repositories in cloud hosted environments, such as AWS and GKE Cloud, as long as the underlying repository is supported. In other words, the repositories listed above are supported, regardless of how they are hosted.

Note

These repositories may not be supported on all operating system platforms. See documentation from repository owners for more information.

Do not mix and match versions. For example, if you're running Oracle Database 11gR2, and want to take advantage of the new support for Oracle UCP, download driver and companion JARs for Oracle version 11gR2.

2.2. Containers

You must install IDM as a stand-alone service, using Apache Felix and Jetty, as provided. Alternate containers are not supported.

IDM bundles Jetty version 9.2.

2.3. Supported Connector Versions

IDM 6.5.1.0 bundles version 1.5.20.5 of the connectors. For a list of all supported connectors, see "*Connector Overview*" in the *Connector Reference*.

Windows versions 2008, 2012 R2, and 2016 are supported as the remote systems for connectors and password synchronization plugins.

You must use the supported versions of the .NET Connector Server, or the Java Connector Server. The 1.5.x Java Connector Server is backward compatible with the version 1.1.x connectors. The 1.5.x .NET Connector Server is compatible only with the 1.4.x and 1.5.x connectors. For more information, see "IDM / ICF Compatibility Matrix".

The Java connector server requires Java 8 or Java 11 and is supported on any platform on which Java runs.

The .NET connector server requires the .NET framework (version 4.5 or later) and is supported on Windows Server versions 2008 R2, 2012 R2 and 2016.

Important

Although the scripted connector toolkits are supported, connectors that you build with these toolkits are not supported. You can find examples of how to build connectors with these toolkits in the Samples Guide.

The following table lists the connector and connector server versions that are supported across IDM versions. For a list of connectors supported with this IDM release, see "*Connector Overview*" in the *Connector Reference*. For a list of connector releases associated with this version of IDM, see "*Connector Release Notes Overview*" in the *Connector Release Notes*

IDM / ICF Compatibility Matrix

IDM Version	ICF Framework	Supported Java Connectors	Supported .NET Connectors
3.x	1.4.x, 1.5.x	Java connectors version 1.1.x - 1.5.x	Active Directory Connector 1.4.0.0, PowerShell Connector 1.4.x Note that the Active Directory connector is deprecated. For more information, see "*Active Directory Connector*" in the *Connector Reference*
4.x	1.4.x, 1.5.x	Java connectors version 1.1.x - 1.5.x	Active Directory Connector 1.4.0.0, PowerShell Connector 1.4.x Note that the Active Directory connector is deprecated. For more information, see "*Active Directory Connector*" in the *Connector Reference*
5.x	1.4.x, 1.5.x	Java connectors version 1.1.x - 1.5.x	Active Directory Connector 1.4.0.0, PowerShell Connector 1.4.x Note that the Active Directory connector is deprecated. For more information, see "*Active Directory Connector*" in the *Connector Reference*
6.x	1.4.x, 1.5.x	Java connectors version 1.1.x - 1.5.x	PowerShell Connector 1.4.x

The following table lists the supported password synchronization plugins:

Supported Password Synchronization Plugins

Plugin	Supported Version
DS Password Synchronization Plugin	6.5.0, supported with DS 6.5.x and IDM 6.5.x 6.0, supported with DS 6.0.x and IDM 6.0.x

Plugin	Supported Version
	5.5.0, supported with DS 5.5.x and IDM 5.5.x 5.0, supported with DS 5.0.x and IDM 5.0.x 3.5, supported with OpenDJ 3.5 and OpenIDM 4.x DS Password Sync plugins are not supported with DS OEM
Active Directory Password Synchronization Plugin	1.7.0, 1.5.0, 1.4.0, 1.3.0, 1.2.0, and 1.1.0 supported on Windows 2008 R2, Windows 2012 R2, and Windows 2016 **Note** Because version 1.4.0 can fail to make a secure connection with certain Windows versions, ForgeRock recommends using a later version.

2.4. Choosing a Browser

ForgeRock has tested many browsers with the IDM UI, including the following browsers:

- Chrome and Chromium, latest stable version
- Firefox, latest stable version
- Safari, latest stable version
- Internet Explorer 11 and later

2.5. Choosing an Operating System

IDM is supported on the following operating systems:

- Red Hat Enterprise Linux (and CentOS Linux) 6.5 and later, 7.x
- Ubuntu Linux 16.04, 18.04
- Windows 2008 R2, 2012 R2, 2016

2.6. Preparing the Java Environment

IDM requires Java 8 or Java 11, specifically at least the Java Standard Edition runtime environment.

ForgeRock validates IDM software with Oracle JDK and OpenJDK, and does occasionally run sanity tests with other JDKs. Support for very specific Java and hardware combinations is best-effort. This means that if you encounter an issue when using a particular JVM/hardware combination, you must also demonstrate the problem on a system that is widespread and easily tested by any member of the community.

ForgeRock recommends that you keep your Java installation up to date with the latest security fixes.

Important

- The clock implementation in JDK 8 is based on `System.currentTimeMillis()` and supports time resolution up to the millisecond only. JDK 11 has an enhanced system clock implementation that provides at least the same precision as the underlying system clock.

  Precise time resolution is important for features such as queued synchronization that rely on precise time for ordering of operations. It is therefore recommended that you use JDK 11 for optimum performance of these features.

- If you are using Oracle JDK 8 and you use 2048-bit SSL certificates, you *must* install the Unlimited JCE policy to enable IDM to use those certificates.

  Download and install the Unlimited JCE Policy for Java 8 from the Oracle Technetwork site. Unzip the JCE zip file and install the JCE policy JAR files in the `/lib/security` folder of the JRE.

2.7. Fulfilling Memory and Disk Space Requirements

When you install IDM for evaluation, with the embedded DS repository, you need 256 MB memory (32-bit) or 1 GB memory (64-bit) available.

You also need 10 GB free disk space for the software and for sample data.

Important

A DS repository (whether embedded or external) requires free disk space of 5% of the filesystem size, plus 1 GB by default. To change this requirement, set the `disk-full-threshold` in the DS configuration. For more information, see Setting Disk Space Thresholds For Database Backends in the *DS Administration Guide*.

In the case of an embedded DS instance, you can manage the configuration using the `dsconfig` command in `/path/to/openidm/db/openidm/opendj/bin`.

In production, disk space and memory requirements will depend on the size of your external repository, as well as the size of the audit and service log files that IDM creates.

The amount of memory that IDM consumes is highly dependent on the data that it holds. Queries that return large data sets will have a significant impact on heap requirements, particularly if they are run in parallel with other large data requests. To avoid out of memory errors, analyze your data requirements, set the heap configuration appropriately, and modify access controls to restrict requests on large data sets.

Chapter 3
Fixes, Limitations, and Known Issues

This chapter covers the status of key issues, limitations, and known issues for this release of ForgeRock Identity Management. For details and information on other issues, see the IDM issue tracker.

3.1. Fixed Issues

Key Fixes in IDM 6.5.2.0

- OPENIDM-17053: Registration form is not loading
- OPENIDM-15283: Unable to login to Admin console after setting enableDynamicRoles==true
- OPENIDM-17195: Change password button disabled state is inverted
- OPENIDM-17634: sustaining/6.5.x - Links for account creation and password reset are not being shown
- OPENIDM-13745: Add the ability to configure the failover variable for the LDAP Connector to the Admin UI
- OPENIDM-17126: Changing schedule to 15 min intervals breaks the admin UI
- OPENIDM-14791: 401 is returned 30 minutes after authentication in fullstack
- OPENIDM-16674: Need to check for presence of OriginResourceContext before trying to use it
- OPENIDM-16394: IDM 6.5.0.3 end-user UI blank in IE11
- OPENIDM-15805: End User UI doesn't format page correctly within Delegated Admin's view of managed/users with very long details
- OPENIDM-17638: sustaining/6.5.x - missing locale in enduser UI
- OPENIDM-15988: Remove Manager does not work with external DS repo (using the UI or PATCH via curl)
- OPENIDM-17600: entryExpireSeconds for UUID token longer than 1 hour expires early
- OPENIDM-17779: Update npm to match version in pom.xml

- OPENIDM-17867: sustaining/6.5.x - Forgot username/password and registration - Access denied error

- OPENIDM-17826: Upgrade Webpack to version 4 for to facilitate security fixes

- OPENIDM-17748: sustaining/6.5.x - Remove obsolete files for 6.5.2.0 upgrade

- OPENIDM-15931: IDM Startup issues (Java 11) during upgrade 6.5.0.4-->6.5.1.0 and also OOTB with some Java 11.x

- OPENIDM-16771: Updating managed/user property from the EndUserUI fails with policy validation error if there are Required relationships

- OPENIDM-17591: NPE when creating object with null value for singleton relationship

- OPENIDM-17792: 7.1 doesn't start on M1 mac

- OPENIDM-14459: Unable to remove relationship on origin via PATCH with link-expanded field

- OPENIDM-12326: JWT dynamic role calculation configuration is invalid for auth module role properties

- OPENIDM-16931: SynchronizationException caught on clustered recon node not propagated to other nodes

- OPENIDM-17204: Improve IDM REST API query performance

- OPENIDM-15507: Paging controls in connector data tab are disabled and should not be

- OPENIDM-15718: triggerSyncProperties does not work properly when using roles

- OPENIDM-13845: Sorting by default leads to extreme slowness in Admin UI

- OPENIDM-16379: Removing values from a multi-valued managed/user property fails with policy validation error if the property is set to Required

- OPENIDM-16259: Unable to add situational event scripts to mappings via the Admin UI

- OPENIDM-16297: Add support for boolean column types in explicit mappings

- OPENIDM-16249: cURL PATCH remove request does not delete relationship when _fields are specified

- OPENIDM-16037: UI does not reflect the default sync failure handler if none is specified

- OPENIDM-10087: Merge JDBC and DJ retry logic

- OPENIDM-16091: 'length' attribute in managed object causes 'Uncaught TypeError' error in web console

- OPENIDM-17071: NullPointerException with augmentSecurityContext
- OPENIDM-16774: Provide full details of schedules in the IDM admin UI

Key Fixes in IDM 6.5.1.0

- OPENIDM-15150: IE11 script error in End-User UI
- OPENIDM-15103: UI: New version of workflow breaks UI forms
- OPENIDM-14046: Duplicates of the same workflow process show within the end user UI
- OPENIDM-14583: using ForgeRock github end-user-ui does not allow you to use "|\/" in the password
- OPENIDM-15024: Settings userEditable: false for mail disables changes in end UI profile page
- OPENIDM-14205: Exception caught marshalling a SynchronizationEvent for requests made with CLIENT_CERT authentication
- OPENIDM-14489: PKCS12 keystore in IDM
- OPENIDM-14025: Deadlock during concurrent generic object update operations with MySQL repository
- OPENIDM-12372: A managed object is not capable of handling simultaneous requests from an edge
- OPENIDM-12681: Admin GUI: Role condition with attribute type boolean are treated as string
- OPENIDM-13265: reconById fails with sourceQueryFullEntry true on an external source
- OPENIDM-10660: User metadata is logged in the audit log when an object is changed
- OPENIDM-15135: sustaining/6.5.x - Changed fields are incorrect in audit file
- OPENIDM-15391: Inconsistent results in enduser UI with delegated admin
- OPENIDM-15705: sustaining/6.5.x - unable to add widget in admin UI
- OPENIDM-12330: Notification create date no longer stored by default
- OPENIDM-15584: Using SalesForce connector and changing the updated context URL is not picked up
- OPENIDM-13633: Enabling password history causes error for existing users when they log into the enduser UI and edit their profile
- OPENIDM-15650: UI: Misalignment of managed Object Attributes

- OPENIDM-15598: Notification Time and Date incorrect in End User UI
- OPENIDM-15776: UI: Maven build does not fail on Eslint Errors
- OPENIDM-15196: Fullstack with social IDP provisioning - arbitrary redirect_uri value is not respected
- OPENIDM-15320: Changing connectionTimeout in datasource.jdbc makes no difference in behavior
- OPENIDM-14832: triggerSyncProperties does not work when using an encrypted password
- OPENIDM-15223: Base Connector Details not changing when updated context URL
- OPENIDM-15446: Missing indexes on relationship table
- OPENIDM-15861: sustaining/6.5.x - scriptedcrest 1.5.1.0 not compatible with groovy connector 1.5.19.1
- OPENIDM-15859: sustaining/6.5.x - update example sample provisioner for databasetable 1.5.19.1
- OPENIDM-15875: sustaining/6.5.x - multiple password example is not working anymore
- OPENIDM-15862: sustaining/6.5.x - sample for scriptedrest is not working with scriptedrest connector 1.5.19.1
- OPENIDM-14125: Synchronization fails for mappings with names longer than 50
- OPENIDM-12632: queryFilter on recon audit fails using MSSQL as repo

Key Fixes in IDM 6.5.0.4

- OPENIDM-9962: Exclude unmodified attributes for UPDATE operations against ICF targets
- OPENIDM-12207: UI login fails with non-ASCII username or password
- OPENIDM-12334: UI: IDM Recon result failure summary doesn't respond to click on "View Entries"
- OPENIDM-12591: authzMembers can have duplicate entries when added using openidm.create() in scripts
- OPENIDM-13213: Editing the members property of the managed role object schema breaks conditional provisioning role members
- OPENIDM-13238: Using runAs for a user with delegated administration priviledges doesn't seem to return the correct results
- OPENIDM-13821: Queued sync event getting stuck in state PENDING
- OPENIDM-13854: REST - Deleting user with a non existent relationship object returns 404

- OPENIDM-13900: Allow exceptions to be thrown from workflow scripts
- OPENIDM-13966: Modifying the Display Properties of a relationship within the admin UI causes the notify attribute to be lost
- OPENIDM-13983: Unable to delete attribute when it has "scope": "private"
- OPENIDM-14051: NullPointerException on jdbc explicit tables with explicitMapping type NUMBER
- OPENIDM-14066: Recon status report showed extra recon was done
- OPENIDM-14099: queued sync doesn't work for mappings with names longer than 38 characters in JDBC repo
- OPENIDM-14287: cli.sh keytool export and import causes IDM startup failure with 'Invalid AES key length' error
- OPENIDM-14322: Unable to delete private properties via openidm.update()
- OPENIDM-14324: We need to be able to run Jetty.xml from a Project directory
- OPENIDM-14340: Workflow callActivity not working with Cron Expression
- OPENIDM-14349: Relationship properties not in source object when returnByDefault is true
- OPENIDM-14432: Restarting IDM cluster generates error message on first node: Scheduled service "scheduler-service-group.liveSync" invocation reported failure:
- OPENIDM-14462: Trailing spaces stripped from input after " in Admin UI
- OPENIDM-14468: Delegated Admin access on array attributes
- OPENIDM-14520: Admin UI: IDM Recon result failure summary "View Entries" does not display entries
- OPENIDM-14534: Fix exception in delegated admin code
- OPENIDM-14548: External REST: Calling endpoints which return a JSON array throws error
- OPENIDM-14692: Workflow: Need to show who approved what when
- OPENIDM-14771: Managed user property that is userEditable and nullable isn't visible on Enduser UI.
- OPENIDM-14911: Self-registration with email validation enabled disables field validation on registration form
- OPENIDM-15025: Managed user property that is userEditable and nullable isn't visible in the admin UI under privileges

Key Fixes in IDM 6.5.0.3

- OPENIDM-12208: Clustered reconciliation fails due to paging cookie from ldap AD
- OPENIDM-12498: UI: Schedule Task Scanner with empty Object Property Field gets unexpected value added
- OPENIDM-12710: API descriptor not available after setting minLength property via admin UI
- OPENIDM-12969: Assignment of workflow to candidate user/group fails
- OPENIDM-13041: Workflow approval displaying all attributes
- OPENIDM-13415: managed/user is duplicated in UI Authentication Client Cert Query
- OPENIDM-13421: Unable to sort by _id in ScriptedSQL Sample
- OPENIDM-13721: NULL not set correctly when adding users. It is set to string of 'null'
- OPENIDM-13737: Self-service registration fails in multi-node cluster scenario when configured for full-stack
- OPENIDM-13740: Explicit repo table: validate mapping before CREATE
- OPENIDM-13763: Admin UI: Japanese input not working for managed user and role
- OPENIDM-13807: Reset Button in Edit Role Immediately Following New Creation Secretly Allowed to be Clicked
- OPENIDM-13811: Windows Service does not start up IDM with JDK 11
- OPENIDM-13814: Salesforce Provider - "User Info Endpoint" doesn't work in UI - typo
- OPENIDM-13847: Workflow task asignee doesn't display username in the UI
- OPENIDM-13882: Admin UI sends multiple REST requests with opposite values in the payload when disabling a connector
- OPENIDM-14163: Workflow: Groovy classpath problem
- OPENIDM-14184: Selfservice password reset gives no warning/explanation for passwords failing CANNOT_CONTAIN_OTHERS policy
- OPENIDM-14253: Admin UI: Tab key to move to next textbox does nothing after selecting Japanese input
- OPENIDM-14266: Remove security/realm.properties

Key Fixes in IDM 6.5.0.2

- OPENIDM-12152: IDM needs a openidm encrypt script binding that allows specification of a purpose to use for encryption
- OPENIDM-12190: Router authz fails in multiple-passwords sample
- OPENIDM-12248: Data races in state shared across threads in recon
- OPENIDM-12312: UNIQUE policy on properties other than userName not correctly check during self-registration
- OPENIDM-12318: Unable to create new contacts because reCaptcha load failure
- OPENIDM-12353: Processing an array attribute containing null element results in null pointer exception
- OPENIDM-12376: Error retrieving scheduler jobs and firing triggers after upgrading to 6.5
- OPENIDM-12529: IDM 6.5.0 Encrypt / Decrypt section includes behaviour which only works in 7.0.0+
- OPENIDM-12613: UI Bug (a missing Admin in the user profile drop down menu) for managed object user
- OPENIDM-12664: Target phase run when reconById dispatched on mapping configured for clustered recon
- OPENIDM-12680: Reconciliation stuck in ACTIVE_QUERY_ENTRIES (or other ACTIVE_ state) and cannot be cancelled
- OPENIDM-12755: Editing of task in admin console throws validatorErrors in handlebars-4.0.5.js
- OPENIDM-12804: uuid token expiry doesn't work with jdbc repo
- OPENIDM-12813: Admin UI login requires auto-reload of End-User interface
- OPENIDM-12802: API Explorer getting 401 Unauthorized after Full-Stack
- OPENIDM-12886: Registration "Sign In" link does nothing
- OPENIDM-12897: Large integers not handled correctly in JavaScript
- OPENIDM-12904: Sending mail with null "to" field causes IDM to hang
- OPENIDM-12954: Ensure signout works properly when access token has expired
- OPENIDM-12964: 'Try resetting your password again' link is not working after entering KBA incorrectly.

- OPENIDM-13064: End User admin link broken when Self-Service relative URL is not "/"
- OPENIDM-13086: Do not cache Managed Roles and Assignments within ReconContext during reconciliation
- OPENIDM-13111: !== in mergeWithTarget.js (and possibly other scripts) doesn't check if value is undefined only if value is null
- OPENIDM-13119: UI does not correctly display validation for Password History
- OPENIDM-13160: PATCH may succeed although If-Match does not match _rev
- OPENIDM-13162: ManagedObject UPSERT contract creates orphan meta object on update via PUT
- OPENIDM-13229: 'Sign in' in the registration interface has a broken link due to trailing "/"
- OPENIDM-13241: Sample password history policy results in 500 error when used with SelfService registration/reset
- OPENIDM-13242: Updating relationship with the same object in a different relationship will not delete reverse references of the updated relationship.
- OPENIDM-13261: Fix exception in PendingLinkAction.getPendingActionContext
- OPENIDM-13411: identityServer.getProperty() returns null pointer if property isn't set rather than being handled gracefully
- OPENIDM-13457: UI broken for social auth registration
- OPENIDM-13721: NULL not set correctly when adding users. It is set to string of 'null'

Key Fixes in IDM 6.5.0.1

- OPENIDM-12017: OPENIDM-12017: IDM CAUD syslog product name (APP-NAME) is null
- OPENIDM-12192: OPENIDM-12192: Modifying virtual property corrupts managed.json
- OPENIDM-12200: OPENIDM-12200: Uncaught TypeError in JavaScript console when saving reverse relationship
- OPENIDM-12228: OPENIDM-12228: remove the INFO message for ScriptedFilter
- OPENIDM-12254: OPENIDM-12254: IDM UI doesn't render linked view for SAP R3
- OPENIDM-12309: OPENIDM-12309: "require" javascript changes are not picked up by IDM 6.5
- OPENIDM-12370: OPENIDM-12370: enable HSM data decryption from IDM 3.1.0 instances
- OPENIDM-12383: OPENIDM-12383: API descriptor not available after setting relationship-type property to nullable

- OPENIDM-12413: OPENIDM-12413: Multi-nodes clustered recon may fail with wrong situation
- OPENIDM-12517: OPENIDM-12517: Adding the triggerSyncProperties in sync.json stops pushing a newly created managed object implicitly to the end resource
- OPENIDM-12796: OPENIDM-12796: jsonstorage "local" self-service with "uuid" option fails in multi-node cluster scenario
- OPENIDM-12865: OPENIDM-12865: jwt token fails in multi-node cluster scenario

Key Fixes in IDM 6.5.0

The following important bugs were fixed in ForgeRock Identity Management 6.5.2:

- OPENIDM-10542: IDM decryption fails with AES 256-bit key
- OPENIDM-11292: Registration autologin with full-stack not working
- OPENIDM-9665: Startup of OpenIDM with MySQL repo ends in ACTIVE_READY state even if repo-jdbc bundle fails to initialize
- OPENIDM-11602: Recons failing due to memory issues via the scripted sql connector
- OPENIDM-11480: With Oracle repo, Create or Update Managed user via UI results in 500 error
- OPENIDM-6514: JDBC repo errors on startup when using mysql
- OPENIDM-10132: IDM does not start, when configured with HSM and Embedded DS
- OPENIDM-9446: Random startup failures when using DB2 as a repo
- OPENIDM-9520: Update via REST with PUT removes private fields which are not included in the request
- OPENIDM-9331: Enabling CSV tamper prevention through the Admin UI may fail with a keystore password error
- OPENIDM-10600: Internal error "no deployed process definition found" after deleting process definition
- OPENIDM-5465: Performance Issue updating conditional role memberships
- OPENIDM-7665: Admin UI mapping view returns HTTP 400 error
- OPENIDM-10653: Password reset fails using explicit tables
- OPENIDM-9576: Records with missing _sortKeys are not returned in query results
- OPENIDM-8043: Unable to initialize keystore and truststore when passwords are different

- OPENIDM-10720: If a user does not exist in the workflow identity service there will be an NPE when trying to retrieve that user
- OPENIDM-10793: Problems with propvalue column size in properties tables
- OPENIDM-11052: Admin UI Mappings page load delay on system?_action=test REST call
- OPENIDM-11597: IllegalArgumentException updating external account if trace is enabled
- OPENIDM-10948: OpenerHandler require does not work with Internet Explorer
- OPENIDM-10919: JavaScript in Internet Explorer does not support the "includes" method of String
- OPENIDM-11863: Default configuration for jsonstore.json is incorrect
- OPENIDM-10603: Unexpected "manager" property in the "before" of activity audit records when patching manager on a user
- OPENIDM-11055: In some Full Stack configurations, you might need to increase the default header size
- OPENIDM-11237: The `openidm.workflow.enabled` property does not affect workflows
- OPENIDM-10749: Require modules appear to be reloaded with every script reference
- OPENIDM-10321: Salesforce provisioner fails to activate and throws NPEs at runtime
- OPENIDM-10787: The javascript.recompile.minimumInterval config values incorrect for common-js modules
- OPENIDM-10974: openidm.objecttypes.objecttype definition not consistent across DBs
- OPENIDM-11510: UI: Can't edit properties of newly added object type in connector configuration
- OPENIDM-11024: NPE can be thrown if the authentication service comes up before the identityService
- OPENIDM-10263: Salesforce connector error while accessing data from User and Profile objects
- OPENIDM-11822: migrateRepoRelationshipsData.js script does not set relationships correctly for >1000 relations
- OPENIDM-10828: MongoDB Connector UI configuration has an incorrect documentation link
- OPENIDM-11215: IDM hangs using IE11 with error "Promise is undefined" in ResourceQueryFilterEditor.js
- OPENIDM-10823: UI intermittently doesn't work with a changed REST context when using Firefox
- OPENIDM-11862: Setting a timeout on a uuid token via jsonstore.json has no effect

- OPENIDM-11739: Concurrent recons could cause exception deleting interim state instance deleteInterimStateInstance
- OPENIDM-11737: Missing relationship references when link expanding on missing resourceCollection in schema
- OPENIDM-11810: When generating full config, "id": "FIX_ME" is returned under operationOptions
- OPENIDM-10833: Cluster widget doesn't show shutdown time for killed node correctly
- OPENIDM-11235: Recon shows error "Target does not support attribute lastSync"
- OPENIDM-11174: Unable to resume scheduler jobs after successful pause
- OPENIDM-11852: Clustered recon in a multi-node environment may never complete
- OPENIDM-10740: Sharing and Activity (UMA) sections in the Self-Service UI do not display thumbnails
- OPENIDM-10400: When configuring a new LDAP Connector config for AD using the Admin UI, the groupMembership, groupType, and groupScope attributes in the user schema are not set up properly
- OPENIDM-10867: Email password string property substitution is not displayed in UI
- OPENIDM-11554: Health service does not identify ds repo bundle correctly
- OPENIDM-11231: IDM logs has suspicious INFO message in clustered recon
- OPENIDM-10578: Unable to specify the authenticationId within augmentSecurityContext script
- OPENIDM-11511: Changing the name of an object type in connector config creates erroneous entries
- OPENIDM-11667: If Salesforce is unavailable, testing the Salesforce Connector throws a 500 error
- OPENIDM-11704: UI: Can't edit validation policy without specifying a parameter
- OPENIDM-10758: openidm.read() returns different content if called from managed.json action or a custom endpoint
- OPENIDM-10829: PUT modifications to workflow/taskInstance/[_id] return 'Task updated' even when no changes occur
- OPENIDM-11393: assigning a userTask to openidm-admin could cause null pointer exception
- OPENIDM-10537: Deleting a previously set field during profile completion does not work
- OPENIDM-11640: null exception in defaultMappings.json

3.2. Limitations

IDM 6.5.2.0

- There are no limitations in functionality in this release, other than what is listed in IDM 6.5.0.

IDM 6.5.1.0

- There are no limitations in functionality in this release, other than what is listed in IDM 6.5.0.

IDM 6.5.0.4

- There are no limitations in functionality in this release, other than what is listed in IDM 6.5.0.

IDM 6.5.0.3

- There are no limitations in functionality in this release, other than what is listed in IDM 6.5.0.

IDM 6.5.0.2

- There are no limitations in functionality in this release, other than what is listed in IDM 6.5.0.

IDM 6.5.0.1

- There are no limitations in functionality in this release, other than what is listed in IDM 6.5.0.

IDM 6.5.0

ForgeRock Identity Management 6.5 has the following known limitations:

- When you add or edit a connector through the Admin UI, the list of required `Base Connector Details` is not necessarily accurate for your deployment. Some of these details might be required for specific deployment scenarios only. If you need a connector configuration where not all the Base Connector Details are required, you must create your connector configuration file over REST or by editing the provisioner file. For more information, see "Configuring Connectors" in the *Integrator's Guide*.
- For OracleDB repositories, queries that use the `queryFilter` syntax do not work on CLOB columns in explicit tables.
- A conditional GET request, with the `If-Match` request header, is not currently supported.
- IDM provides an embedded workflow and business process engine based on Activiti and the Business Process Model and Notation (BPMN) 2.0 standard. As an embedded system, local integration is supported. Remote integration is not currently supported.
- When using privileges, relationships are not returned in queries. This means information that is handled as a relationship to another object (such as roles for a managed user) will not be available.

- Support for running remote connector servers with the legacy communication protocol has been removed. Connections to remote connector servers must use the `websocket` protocol.

3.3. Known Issues

IDM 6.5.2.0

- There are no new known issues in this release, other than those issues listed in IDM 6.5.0, IDM 6.5.0.3, and IDM 6.5.1.0.

IDM 6.5.1.0

- OPENIDM-15931: IDM startup issues on Java 11

  On certain OS variants, running Java 11, errors are seen from the logging service when the server starts up.

  Workaround:

  - If you are upgrading to IDM 6.5.1 from a previous 6.5.x version, copy the `openidm/bundle/javax.annotation-api-1.2.jar` file from your old instance to the `bundle` directory of your new 6.5.1.0 instance.

  - If you are not upgrading to IDM 6.5.1 from a previous 6.5.x version, download the javax.annotation-api-1.2.jar file from the Maven repository, and copy it to the `bundle` directory of your 6.5.1.0 instance.

IDM 6.5.0.4

- There are no new known issues in this release, other than those issues listed in IDM 6.5.0 and IDM 6.5.0.3.

IDM 6.5.0.3

- OPENIDM-15650: UI: Misalignment of managed Object Attributes

- Microsoft JDBC Driver 7.x for Java 11 Does Not Work with IDM 6.5.x

  ForgeRock has found that the Java 11 version of the Microsoft JDBC Driver 7.x for SQL Server (`mssql-jdbc-7.2.2.jre11.jar`, `mssql-jdbc-7.4.1.jre11.jar`) does not work with IDM 6.5.x due to a class loading problem.

  One possible workaround is to use the Java 8 version of the driver (`mssql-jdbc-7.2.2.jre8.jar` and `mssql-jdbc-7.4.1.jre8.jar`), which we have found to work with Java 11. Note that Microsoft does not recommend this configuration and may not support it.

If you are using Java 11 and must use the Java 11 version of the driver (`mssql-jdbc-7.2.2.jre11.jar`, `mssql-jdbc-7.4.1.jre11.jar`), the only workaround is to update your IDM version from 6.5.x to an upcoming major release, which fixes this issue.

IDM 6.5.0.2

- There are no known issues in this release, other than those issues listed in IDM 6.5.0.

IDM 6.5.0.1

- There are no known issues in this release, other than those issues listed in IDM 6.5.0.

IDM 6.5.0

The following important issues remained open at the time of this release:

- OPENIDM-14099: Queued sync does not work for mappings with names longer than 38 characters (JDBC repo)

  Workaround: Queued synchronization creates locks when it acquires the mappings to process on a particular IDM node. The length of the `objectid` column in the `locks` table is 38 characters by default. Because the lock `_id` is set to the mapping name, it can easily exceed 38 characters. You should increase the length of this column to 255 characters.

- OPENIDM-12170: Delete on managed or internal object does not return the included relationship fields that were included in the request
- OPENIDM-12177: Notifications service does not work with relationship fields
- OPENIDM-12109: Able to add managed object property with illegal character via Admin UI
- OPENIDM-12106: Delegated Admin query filter and fields requests does not work properly with object type
- OPENIDM-12105: Delegated Admin UI Should Only Display Supported Fields in grid
- OPENIDM-12100: An existing privilege should default new schema fields to READ
- OPENIDM-12078: You cannot customize the aliases of the default keys added to the IDM keystore and truststore

  Workaround: To generate the default keys and certificates with custom aliases, see "To Generate Keys and Certificates With Custom Aliases (Workaround for OPENIDM-12078)".

- OPENIDM-12077: UI has JSON type pulldown for _rev for internal users
- OPENIDM-12074: Authentication Provider does not work after restarting IDM and AM

Workaround: If you have shut down IDM and AM, start AM first. When you can log in, start IDM then navigate to the IDM Admin UI.

- OPENIDM-12063: Repo init service fails in audit-jdbc sample
- OPENIDM-12060: Sync triggers can get stuck after nodes are recycled
- OPENIDM-12017: IDM CAUD syslog product name (APP-NAME) is null
- OPENIDM-11960: Complex query expressions are not correctly parsed to SOQL for Salesforce
- OPENIDM-11950: Infinite loop possible for Managed PATCH operations
- OPENIDM-11921: Errors logged when password-reset email URL is expired and clicked
- OPENIDM-11879: Workflow time zone handling is not consistent and leads to unexpected results
- OPENIDM-11765: Warnings on startup when using embedded DS repo with Java 11
- OPENIDM-11714: Full Stack: /admin endpoint redirects to self-service page
- OPENIDM-11536: Cannot set user password for user created through full-stack social registration
- OPENIDM-11408: Paging is not working in 'Association/Data Association Management for mapping detail'.

  Workaround: The JSON audit handler does not support paging. If you use an audit audit handler that supports paging (such as the repository or elasticsearch handlers), you will not encounter this issue.

- OPENIDM-11370: Activiti workflow mail task goes to default localhost:25

  Workaround: Use the external email service described in "*Configuring Outbound Email*" in the *Integrator's Guide*.

- OPENIDM-10761: Progressive Profiling scripted condition does not include user fields within "object" map
- OPENIDM-10660: User metadata is logged in the audit log when an object is changed
- OPENIDM-10455: Query and non-read operations not authorised for openidm-admin role with OAuth
- OPENIDM-10072: Scheduler service registered too early by OSGi
- OPENIDM-9791: Error while generating process diagram, image will not be stored in repository
- OPENIDM-9554: Workflow Processes Completed have "Not Found Error" for managed/user
- OPENIDM-9353: IDM does not audit the http response headers in the access audit log

- OPENIDM-9081: WARNING about extensions directory not existing appears in felix console upon restart of IDM
- OPENIDM-8518: Not Found error when accessing a process instance via Admin UI
- OPENIDM-8295: Non-required single relationship properties should be nullable
- OPENIDM-8122: OpenIDM Cluster incorrectly shows ready and running
- OPENIDM-8052: Cannot create a remote (.NET) connector through the UI
- OPENIDM-6467: syslog audit event handler created although required property not set
- OPENIDM-4149: availableConnectors are not updated after remote ICF shut down
- OPENIDM-4068: Config Changes made in config files should get logged by the Config Audit Logger.

To Generate Keys and Certificates With Custom Aliases (Workaround for OPENIDM-12078)

1. Generate each default key with the custom alias, for example:

```
keytool -genseckey \
 -alias openidm-sym-default-custom \
 -keyalg AES \
 -keysize 128 \
 -keystore security/keystore.jceks \
 -storetype JCEKS

 keytool -genseckey \
  -alias openidm-selfservice-key-custom \
  -keyalg AES \
  -keysize 128 \
  -keystore security/keystore.jceks \
  -storetype JCEKS

  keytool -genseckey \
 -alias openidm-jwtsessionhmac-key-custom \
 -keyalg HmacSHA256 \
 -keysize 2048 \
 -keystore security/keystore.jceks \
 -storetype JCEKS

 keytool -genkey \
 -alias openidm-localhost-custom \
 -keyalg RSA \
 -keysize 2048 \
 -keystore security/keystore.jceks \
 -storetype JCEKS

 keytool -genkey \
 -alias server-cert \
 -keyalg RSA \
 -keysize 2048 \
 -keystore security/keystore.jceks \
 -storetype JCEKS
 keytool -genkey \
 -alias selfservice-custom \
 -keyalg RSA \
 -keysize 2048 \
 -keystore security/keystore.jceks \
 -storetype JCEKS

 keytool -export \
 -alias openidm-localhost-custom \
 -file exportedCert \
 -keystore security/keystore.jceks \
 -storetype JCEKS

 keytool -import \
 -alias openidm-localhost-custom \
 -file exportedCert \
 -keystore security/truststore
 \
-storetype JKS
```

Note that these commands do not change the alias of the default `server-cert`. To customize the `server-cert` alias for an embedded DS repository, define the custom alias in the `resolver/boot.properties` file, for example `"openidm.config.crypto.opendj.localhost.cert=my-custom-alias"`.

2. Edit the aliases that are defined in `conf/secrets.json`. For example, with the aliases specified previously:

```
{
  "stores": [
    {
      "name": "mainKeyStore",
      "class": "org.forgerock.openidm.secrets.config.FileBasedStore",
      "config": {
        "file": "&{openidm.keystore.location|&{idm.install.dir}/security/keystore.jceks}",
        "storetype": "&{openidm.keystore.type|JCEKS}",
        "providerName": "&{openidm.keystore.provider|SunJCE}",
        "storePassword": "&{openidm.keystore.password|changeit}",
        "mappings": [
          {
            "secretId" : "idm.default",
            "types": [ "ENCRYPT", "DECRYPT" ],
            "aliases": [ "openidm-sym-default-custom" ]
          },
          {
            "secretId" : "idm.config.encryption",
            "types": [ "ENCRYPT", "DECRYPT" ],
            "aliases": [ "openidm-sym-default-custom" ]
          },
          {
            "secretId" : "idm.password.encryption",
            "types": [ "ENCRYPT", "DECRYPT" ],
            "aliases": [ "openidm-sym-default-custom" ]
          },
          {
            "secretId" : "idm.jwt.session.module.encryption",
            "types": [ "ENCRYPT", "DECRYPT" ],
            "aliases": [ "openidm-localhost-custom" ]
          },
          {
            "secretId" : "idm.jwt.session.module.signing",
            "types": [ "SIGN", "VERIFY" ],
            "aliases": [ "openidm-jwtsessionhmac-key-custom" ]
          },
          {
            "secretId" : "idm.selfservice.encryption",
            "types": [ "ENCRYPT", "DECRYPT" ],
            "aliases": [ "selfservice-custom" ]
          },
          {
            "secretId" : "idm.selfservice.signing",
            "types": [ "SIGN", "VERIFY" ],
            "aliases": [ "openidm-selfservice-key-custom" ]
          }
        ]
      }
    },
    {
      "name": "mainTrustStore",
      "class": "org.forgerock.openidm.secrets.config.FileBasedStore",
      "config": {
        "file": "&{openidm.truststore.location|&{idm.install.dir}/security/truststore}",
        "storetype": "&{openidm.truststore.type|JKS}",
        "providerName": "&{openidm.truststore.provider|SUN}",
        "storePassword": "&{openidm.truststore.password|changeit}",
        "mappings": [
        ]
      }
    }
  ],
  "populateDefaults": false
}
```

Chapter 4
Compatibility

This chapter covers major and minor changes to existing functionality, as well as deprecated and removed functionality. You must read this chapter before you start a migration from a previous release.

4.1. Important Changes to Existing Functionality

Take the following changes into account when you update to IDM 6.5.2. These changes will have an impact on existing deployments. Adjust existing scripts and clients accordingly:

IDM 6.5.2.0

- Embedded Workflow Database

  Previously, you could use the Activiti workflow engine's embedded H2 database for demo and testing purposes. IDM no longer includes this database. Before you use workflow, you must install a JDBC repository.

  For more information, see "Enabling Workflows" in the *Integrator's Guide*.

IDM 6.5.1.0

- New bundled connector versions

  All connectors bundled with IDM 6.5.1 have been upgraded to version 1.5.19.1. See Connector Changes in IDM 6.5.1.0.

IDM 6.5.0.4

- IDM 6.5.0.4 has upgraded its Jetty library from version 9.4.15 to 9.4.27. The `SslContextFactory` class has changed to `SslContextFactory.(Server|Client)`. Users who have custom Jetty configurations and are upgrading from versions 6.5.0.x to 6.5.0.4 may encounter an error due to this class change.

  You can do one of two workarounds:

  1. Use the Jetty library 9.4.27 as-is, instead of your custom Jetty configuration.
  2. Manually update the `sslContextFactory` to `sslContextFactory$Server`

For example, if you are using IDM 6.5.0.2, the Jetty class would be:

```
<New id="sslContextFactory" class="org.eclipse.jetty.util.ssl.SslContextFactory">
```

After updating to IDM 6.5.0.4, update the new Jetty class to the following:

```
<New id="sslContextFactory" class="org.eclipse.jetty.util.ssl.SslContextFactory$Server">
```

IDM 6.5.0.3

- There are no important changes or enhancements in functionality in this release.

IDM 6.5.0.2

- There are no important changes or enhancements in functionality in this release.

IDM 6.5.0.1

- There are no important changes or enhancements in functionality in this release.

IDM 6.5.0

Changes to `openidm.encrypt()`

The output of `openidm.encrypt()` has changed when using `ECB` as your cipher mode (such as `AES/ECB/PKCS5Padding`). This means the resulting encrypted hash will change each time `openidm.encrypt()` is run on a value. Even though the encryption result may differ each time, `openidm.decrypt()` will still work.

No automated update process

The automated update process available with previous IDM versions is no longer supported. Updating servers is now a manual process and is described in detail in "*Updating Servers*" in the *Installation Guide*.

Existing workflows must be rewritten

If your deployment includes existing workflows, you *must* update these to use Vue JS if you want to view them in the new End User UI. For more information, see "Using Custom Templates for Activiti Workflows" in the *Integrator's Guide*.

Endpoint change from `repo/internal` to `internal`

Internal objects previously accessible at the `repo/internal/` endpoint are now accessible at the `internal/` endpoint. For example, internal user objects are now accessible at `internal/user` rather than `repo/internal/user`.

> **Note**
>
> Because this is a breaking change, additional steps are necessary when upgrading from previous versions of IDM. For more information, see "Changes to `repo/internal`".

Roles are now referred to by full path

Internal and managed roles are now referenced by their full path (for example, `openidm-authorized` is now `internal/role/openidm-authorized`). Support for using role names without a full path is deprecated, and may be removed in a later release.

DS repositories now return a null value for missing properties

Previously, embedded and external `repo.ds.json` files defaulted to not returning empty properties. They now return the empty properties with a value of `null`. This aligns more closely with the behavior seen in JDBC repositories that use explicit mappings.

If you wish to revert this behavior, change `returnNullForMissingProperties` to false in the `rest2LdapOptions` property in your `repo.ds.json` file. For more information about the `returnNullForMissingProperties` property, see Gateway REST2LDAP Configuration File in the *DS Reference*.

End user notification configuration files have changed

End user notifications are now configured in `notification-*.json` files. To review the defaults, see "Notification Configuration Files" in the *Integrator's Guide*.

Notification configuration options have been removed from `onUpdateUser.js`.

In addition, the following files have been removed for IDM 6.5:

- `userNotifications.js`
- `onDelete-user-cleanup.js`

Change to proxy configuration for external REST service

In previous releases, configuring a proxy for the external REST service was achieved by setting the `proxySystem` property in the `external.rest.json` configuration file. There is now a system-wide HTTP client configuration that includes proxy settings. For more information, see "Configuring HTTP Clients" in the *Integrator's Guide*.

4.2. ICF and Connector Changes

The following ICF and connector changes will have an impact on existing IDM deployments that use those connectors:

Connector Changes in IDM 6.5.2.0

- All connectors that are bundled with IDM 6.5.2.0 have been upgraded to version 1.5.20.8.

  For a list of changes to the connectors in version 1.5.20.8 and earlier, see "*Connector Release Notes Overview*" in the *Connector Release Notes*.

Connector Changes in IDM 6.5.1.0

- All connectors that are bundled with IDM 6.5.1.0 have been upgraded to version 1.5.19.1.

  The main changes with these upgraded connectors are as follows:

  - Connector dependencies are now bundled with the connectors. This means that you do not have to download the dependencies separately. Because the dependencies are included in the connector, and not in the IDM `lib` directory, the bundled connector dependency files will have no impact on existing dependency files in that directory.

  - Several connectors are now bundled with the remote connector server (RCS). If you are running connectors remotely, through RCS 1.5.19.1, the following connectors are in the `openicf/connectors` directory, and do not need to be copied to the remote server:

    - CSV File Connector

    - Database Table Connector

    - Groovy Connector

    - Kerberos Connector

    - LDAP Connector

    - SCIM Connector

    - Scripted REST Connector

    - Scripted SQL Connector

    - SSH Connector

  This list shows the main issues fixed in version 1.5.19.1 of the connectors:

  - OPENICF-1445: SSH connector: Stale or disconnected SSH sessions are not detected when borrowing from the pool

  - OPENICF-1433: SSH connector: Kerberos username prompt for public key and password auth

  - OPENICF-1414: Scripted Groovy (v3) based connectors fail to load with IDM releases prior to 7.0

- OPENICF-1408: Java RCS: NPE when we set proxyHost for client mode
- OPENICF-1407: Java RCS: Incorrect url in Debug message of HttpRequestPacket header for non-SSL
- OPENICF-1404: Java connector server proxy config for port is incorrect
- OPENICF-1400: Java Connector Server: Property name usessl should match docs and code
- OPENICF-1399: restarting IDM with active RCS causes RCS to decrement websocket connection count
- OPENICF-1396: OPENIDM-15448 changes seemingly broke querying ldap via the data tab
- OPENICF-1395: Investigate and clean up the following start up error message
- OPENICF-1394: missing connectorserver.scope in connectorserver property file
- OPENICF-1388: LDAP Connector 1.5.5.0 throws java.lang.NoSuchMethodError on Java 8
- OPENICF-1373: Java RCS: default connectorserver.connectionTtl breaks the connection housekeeping
- OPENICF-1371: Java Connector server does not always reestablish closed websockets
- OPENICF-1352: Salesforce connector: pagination and cookies not working properly

IDM 6.5.0.4

- There are no new ICF and connector changes in this release.

IDM 6.5.0.3

- There are no new ICF and connector changes in this release.

IDM 6.5.0.2

- There are no new ICF and connector changes in this release.

IDM 6.5.0.1

- There are no new ICF and connector changes in this release.

IDM 6.5.0

Improvements to the Scripted Groovy Connectors

Connectors based on the Groovy Connector toolkit now use the `CachingSimpleTemplateEngine` utility class, instead of the `SimpleTemplateEngine` class.

The `SimpleTemplateEngine` class is prone to memory leaks. If you have existing Groovy search scripts that use templates, you should update them to use the new class. For example, change:

```
import groovy.text.SimpleTemplateEngine
```

to

```
import org.forgerock.openicf.connectors.groovy.text.CachingSimpleTemplateEngine
```

in your `SearchScript.groovy` scripts.

Removed Azure AD Sample Scripts

The PowerShell Azure AD sample scripts and corresponding sample have been removed from the IDM product. These scripts used a deprecated Powershell Module and may be revised in a future IDM release.

4.3. Deprecated Functionality

This section lists functionality that has been deprecated in the IDM 6.5 releases. Deprecation is defined in "ForgeRock Product Interface Stability".

IDM 6.5.2.0

No functionality has been deprecated in this release.

IDM 6.5.1.0

No functionality has been deprecated in this release.

IDM 6.5.0.4

No functionality has been deprecated in this release.

IDM 6.5.0.3

No functionality has been deprecated in this release.

IDM 6.5.0.2

No functionality has been deprecated in this release.

IDM 6.5.0.1

No functionality has been deprecated in this release.

IDM 6.5.0

- The Office 365 connector is deprecated and support for its use with IDM will be removed in a future release.

  Instead of the Office 365 connector, use the PowerShell Connector Toolkit with the Azure AD scripts, available from the ForgeRock BackStage download site.

- The ability to configure keystores, truststores, obfuscation, and encryption in the IDM 6 version of `openidm/resolver/boot.properties` file is deprecated and will be removed in a future release.

  The ability to set up encryption with a key alias in the `managed.json` file has also been deprecated.

  This functionality has been replaced by the secrets service in IDM 6.5. For more information, see "Configuring the Keystore and Truststore" in the *Integrator's Guide*.

- Support for the `TLSv1.1` protocol has been deprecated and will be removed in a future release. For more information, on the potential vulnerability, see *CVE-2011-3389* from the *National Vulnerability Database* from the US National Institute of Standards and Technology.

  The default security protocol for IDM is `TLSv1.2`. Do not downgrade this protocol to `TLSv1.1` unless necessary. For more information, see "Setting the TLS Version" in the *Integrator's Guide*.

- Support for `oauthReturn` as an endpoint for OAuth2 and OpenID Connect standards has been deprecated for interactions with AM and will be removed in a future release. Support has been removed for interactions with social identity providers, as discussed in "Removed Functionality".

  Default versions of relevant configuration files no longer include `oauthReturn` in the `redirectUri` setting. However, for IDM 6.5, these configuration files should still work both with and without `oauthReturn` in the endpoint.

  This change affects any configuration where IDM interacts as a Relying Party with AM as an OpenID Provider. For related documentation, see "*Integrating IDM With the ForgeRock Identity Platform*" in the *Samples Guide*

- In schedule configurations, setting a time zone using the `timeZone` field is deprecated. To specify a time zone for schedules, use the `startTime` and `endTime` fields, as described in "Configuring Schedules" in the *Integrator's Guide*.

- Support for the `MD5` and `SHA-1` hash algorithms is deprecated and will be removed in a future release. You should use more secure algorithms in a production environment. For a list of supported hash algorithms, see "Encoding Attribute Values by Using Salted Hash Algorithms" in the *Integrator's Guide*.

- The Active Directory (AD) .NET Connector is deprecated and support for its use in IDM will be removed in a future release.

  For simple Active Directory (and Active Directory LDS) deployments, the Generic LDAP Connector works better than the Active Directory connector, in most circumstances. For more information, see "*Generic LDAP Connector*" in the *Connector Reference*.

  For more complex Active Directory deployments, use the PowerShell Connector Toolkit, as described in "*PowerShell Connector Toolkit*" in the *Connector Reference*.

  Note that deprecating the AD Connector has no impact on the PowerShell connector, or on the .NET Connector Server.

- When configuring connectors, (see "Configuring Connectors" in the *Integrator's Guide*), you can set up `nativeType` property level extensions. The `JAVA_TYPE_DATE` extension is deprecated.

- Support for a POST request with `?_action=patch` is deprecated, when patching a specific resource. Support for a POST request with `?_action=patch` is retained, when patching by query on a collection.

  Clients that do not support the regular PATCH verb should use the `X-HTTP-Method-Override` header instead.

  For example, the following POST request uses the `X-HTTP-Method-Override` header to patch user jdoe's entry:

  ```
  $ curl \
  --header "X-OpenIDM-Username: openidm-admin"
   \
  --header "X-OpenIDM-Password: openidm-admin"
   \
  --header "Content-Type: application/json"
   \
  --request POST
   \
  --header "X-HTTP-Method-Override: PATCH" \
     --data '[
      {
      "operation":"replace",
      "field":"/description",
      "value":"The new description for Jdoe"
      }
      ]' \
      "http://localhost:8080/openidm/managed/user/jdoe"
  ```

4.4. Removed Functionality

This section lists functionality that has been removed in the IDM 6.5 releases.

IDM 6.5.2.0

No functionality has been removed in this release.

IDM 6.5.1.0

> **Important**
>
> The Scripted CREST connector, and the corresponding sample have been removed in this release.
>
> This connector is still supported for IDM 6.5 deployments up to 6.5.0.4. From IDM 6.5.1 onwards, you should migrate any deployments that use this connector to the "*Scripted REST Connector*".

IDM 6.5.0.4

No functionality has been removed in this release.

IDM 6.5.0.3

The `security/realm.properties` file has been removed from the installation.

IDM 6.5.0.2

No functionality has been removed in this release.

IDM 6.5.0.1

No functionality has been removed in this release.

IDM 6.5.0

- Support for `oauthReturn` as an endpoint for OAuth2 and OpenID Connect standards has been removed for interactions with social identity providers. It is still available for interactions with AM, as discussed in "Deprecated Functionality".

  Default versions of relevant configuration files no longer include `oauthReturn` in the `redirectUri` setting.

  This change affects any configuration where IDM interacts as a Relying Party with a social identity provider as an OAuth2 or an OpenID Connect Provider. For related documentation, see "*Configuring Social Identity Providers*" in the *Integrator's Guide*

- The automated update facility has been removed. For information on updating servers, see "*Updating to IDM 6.5*".

- Support for the BoneCP Java database connection (JDBC) pool library has been removed. HikariCP has been the default IDM JDBC pool library since version 5. This affects deployments that use JDBC repositories.

  For more information on the configuration of HikariCP, see "Understanding the JDBC Connection Configuration File" in the *Integrator's Guide*.

- Support for running remote connector servers with the legacy communication protocol has been removed. Connections to remote connector servers must use the `websocket` protocol.

- Support for the `TLSv1.0` protocol has been removed. For more information, see the following PDF: *Migrating from SSL and Early TLS* from the *PCI Security Standards Council*.

  The default security protocol for IDM is `TLSv1.2`. Do not downgrade this protocol unless you have a specific need.

Chapter 5

Updating to IDM 6.5

IDM 6.5 provides a number of new features that require changes to an existing configuration. These changes can be broken into two categories: changes that are required for IDM to function, and changes that are only required if you wish to make use of these new features. Before performing the changes laid out in this chapter, review the instructions in "*Updating Servers*" in the *Installation Guide*.

5.1. Required Changes to IDM

The following changes are required when updating from a previous IDM release:

5.1.1. Database Changes

There have been several changes to the database structure for IDM repositories. Run the following scripts to upgrade your database, which can be found in `bin/update/scripts/database-type/`:

`alter_internalrole.sql` or `alter_internalrole.ldif`

This updates the `internalrole` table to include several new columns.

`alter_objecttypes.sql`

Previous MySQL, Oracle, and PostgreSQL database configurations had set the `objecttype` column of IDM's `objecttypes` table to `NULL`. This should be changed to `NOT NULL`.

Microsoft SQL and DB2 were already configured to be `NOT NULL` and need no further changes. DS also needs no changes.

`alter_relationships.sql`

> **Caution**
>
> This script removes a column from the `relationships` table. We recommend making a backup of your repository prior to running this file.

This removes the `properties` column from the `relationships` table. IDM gets relationship properties from the `fullobject` column, making the `properties` column unnecessary.

`alter_uinotification.sql`

This updates the `uinotification` table to adjust the column length for `createDate`.

`create_indices.sql`

(PostgreSQL only) This creates an index for `reconid` in the `genericobjects` table, and adds indices for several fields in the `clusterobjects` table.

`migrate_metaobjects.sql`

> **Caution**
>
> This script deletes meta data from the `genericobjects` table after migrating that data to new tables. We recommend making a backup of your repository prior to running this file.

This creates two new tables, `metaobjects` and `metaobjectproperties`, then moves user meta data from `genericobjects` into these two tables.

The number of scripts found in this directory may vary depending on the database you are using. Scripts not listed above are optional, and relate to enabling or configuring specific features in IDM. These will be referenced in the steps for enabling that particular feature in "Enabling New Features in IDM".

5.1.1.1. Removal of Property Tables in PostgreSQL

> **Note**
>
> This section only applies if you are using PostgreSQL for your repository, and is optional. It should not harm anything to leave these tables in your repository, but it is recommended to remove them for the sake of keeping your database clean.

If you are using PostgreSQL, the following tables previously used to store property data are no longer needed, and may be removed:

- `openidm.genericobjectproperties`
- `openidm.managedobjectproperties`
- `openidm.configobjectproperties`
- `openidm.relationshipproperties`
- `openidm.schedulerobjectproperties`
- `openidm.clusterobjectproperties`
- `openidm.updateobjectproperties`

Since dropping tables from your database is destructive, it is strongly recommended that you back up your database before performing this action.

If you are using your old `repo.jdbc.json` configuration, references to these tables will need to be removed. For example, the updated resource mapping for the config object table removes the `propertiesTable` property and would now be:

```
"config" : {
    "mainTable" : "configobjects"
},
```

5.1.2. Configuration Changes

The following changes to your configuration are required:

5.1.2.1. Changes for the New Secrets Service

The IDM 6 version of `boot.properties` may not be supported in the next release. Therefore, you should review the differences as described in "Configuration Options in `secrets.json`" in the *Integrator's Guide* as soon as possible.

5.1.2.1.1. Secrets Service Updates to `boot.properties`

When comparing the `boot.properties` files from IDM 6 and IDM 6.5, you'll note differences based on the new secrets service:

- Keystore and Truststore information (such as `openidm.truststore.type` or `openidm.keystore.password`) are no longer stored in `boot.properties`. This information has been moved to `conf/secrets.json`.
- Cryptographic settings such as `openidm.config.crypto.alias` have been moved to `conf/secrets.json`.

5.1.2.1.2. Secrets Service Updates to `managed.json`

In the IDM 6 version of `managed.json` file, you'll see the following entry related to user password encryption:

```
"key" : "openidm-sym-default"
```

For the IDM 6.5 version of `managed.json`, this entry has changed to:

```
"purpose" : "idm.password.encryption"
```

You can now define `idm.password.encryption` in the new `secrets.json` file.

5.1.2.2. Changes to `repo/internal`

Internal objects are no longer stored in `repo/internal`, and are now accessed via the `internal` endpoint. If you are updating from a previous release of IDM, you must update existing references to `repo/internal` to the new endpoint.

1. References to `repo/internal` in existing configuration files need to be changed to `internal`. The following files must be updated:

`authentication.json`

The `authModules` of `STATIC_USER` and `INTERNAL_USER` need to update their `queryOnResource` value from `repo/internal/user` to `internal/user`.

`managed.json`

The managed user's `authzRoles` "Internal Role" resource collection should change its `path` from `repo/internal/role` to `internal/role`.

`policy.json`

The `resource` of `repo/internal/user/*` should change to `internal/user/*`.

`router.json`

One filter pattern needs to be updated: `(managed|system|repo/internal)($|(/.))` should change to `(managed|system|internal)($|(/.)`.

One filter pattern needs to be deleted: the `repo/internal/user((/.)|$)` pattern is no longer required and should be deleted from `router.json`.

2. Run the `removeRepoPathFromRelationships` endpoint. This will update any existing relationships to remove `repo/` from internal roles:

```
$ curl \
 --header "X-OpenIDM-Username: openidm-admin" \
 --header "X-OpenIDM-Password: openidm-admin" \
 --request GET \
"http://localhost:8080/openidm/endpoint/removeRepoPathFromRelationships"
```

> **Note**
>
> Prior to running this endpoint, you may need to temporarily adjust `access.js` to include extra access to endpoints:
>
> ```
> {
>     "pattern"    : "endpoint/*",
>     "roles"      : "*",
>     "methods"    : "read",
>     "actions"    : "*"
> },
> ```

5.1.2.3. Changes to Internal Roles and Internal Users

There have been updates to the internal schema for internal roles and users, which require updating existing entries in your repository. To update these internal roles and internal users, run the `updateInternalUserAndInternalRoleEntries` endpoint:

```
$ curl \
 --header "X-OpenIDM-Username: openidm-admin" \
 --header "X-OpenIDM-Password: openidm-admin" \
 --request GET \
"http://localhost:8080/openidm/endpoint/updateInternalUserAndInternalRoleEntries"
```

> **Note**
>
> Prior to running this endpoint, you may need to temporarily adjust access.js to include extra access to endpoints:
>
> ```
> {
>    "pattern"    : "endpoint/*",
>    "roles"      : "*",
>    "methods"    : "read",
>    "actions"    : "*"
> },
> ```

5.1.2.4. Changes to Conditional Roles

The way in which conditional roles are granted to new users has changed. Previously, conditional roles were granted as part of the onCreate script. This functionality was achieved with the following configuration of the user object (in conf/managed.json):

```
"onCreate" : {
    "type" : "text/javascript",
    "source" : "require('onCreateUser').setDefaultFields(object);require('roles/
conditionalRoles').createConditionalGrantsForUser(object,
 'roles');require('onCreateUser').emailUser(object);"
},
```

Conditional role grants are now achieved internally within the IDM backend. To ensure that an updated deployment continues to work as designed, remove the following from all onCreate scripts in your existing managed.json file:

```
require('roles/conditionalRoles').createConditionalGrantsForUser(object, 'roles');
```

5.1.2.5. Changes to repo.jdbc.json

The following fields can be removed from existing repo.jdbc.json files when upgrading from a previous version:

- The properties field of the relationships object has been removed when using generic resource mappings. The object path to this field is /resourceMapping/genericMapping/relationships/properties.

5.1.2.6. Enabling HikariCP

HikariCP is the new default IDM Java database connection (JDBC) pool library. If you are using a JDBC repository, adjust datasource.jdbc-default.json to use hikari instead of boneCP for the connectionPool type:

```
"connectionPool" : {
    "type" : "hikari",
    ...
}
```

For more information on configuring HikariCP, see "Understanding the JDBC Connection Configuration File" in the *Integrator's Guide*.

5.1.2.7. Changes to `router.json`

5.1.2.7.1. Changes to onRequest Filter

The call to `router-authz.js` has been modified in the `onRequest` filter in `router.json`:

```
"onRequest" : {
    "type" : "text/javascript",
    "source" : "require('router-authz').testAccess()"
}
```

5.1.2.7.2. Addition of Relationship Filter

An `onResponse` filter has been added in `router.json`, adding filtering around relationships:

```
{
    "pattern" : "^(managed|internal)($|(/.+))",
    "condition" : {
        "type" : "text/javascript",
        "source" : "context.caller.external === true || context.current.name === 'selfservice'"
    },
    "onResponse" : {
        "type" : "text/javascript",
        "source" : "require('relationshipFilter').filterResponse()"
    }
}
```

5.1.2.7.3. Changes to Internal User Password Encryption

It is no longer necessary to separately encrypt internal user passwords through an `onRequest` script in `router.json`. Internal users will now use the encryption key alias defined in `boot.properties`.

The following entry in `router.json` can be safely removed:

```
{
    "pattern" : "internal/user((/.+)|$)",
    "onRequest" : {
        "type" : "text/javascript",
        "source" : "request.content.password = require('crypto').hash(request.content.password);"
    },
    "methods" : [
        "create",
        "update"
    ]
}
```

5.1.2.8. Changes to the `redirectUri` for Social Identity Providers

The value of `redirectUri` for social identity providers, as configured per "*Configuring Social Identity Providers*" in the *Integrator's Guide* has changed.

If you've configured a social identity provider for a previous version of IDM, you'll need to update the `redirectUri` for the provider, by removing the `oauthReturn/` from the URL, in two locations:

`identityProvider-*name*.json`

> In the configuration file named for the identity provider, such as `identityProvider-google.json`.

When configuring your identity provider

> When you configure your identity provider, look for an entry such as `Redirect` or `Return` URL. You'll need to update the value corresponding to the IDM `redirectUri` on the social identity provider developer (or similar) page.

For example, for IDM 6, you'll have a `redirectUri` such as:

```
http://idm.example.com:8080/oauthReturn/
```

In this case, you'd change the value of `redirectUri` to:

```
http://idm.example.com:8080/
```

5.1.2.9. Updating `logging.properties`

Recent security fixes prevent Jetty from logging sensitive data, such as passwords. Verify that your `conf/logging.properties` file includes the following excerpt (and add the excerpt if necessary) to prevent unnecessary data from being logged:

```
# Logs the output from Jetty
 # Sets the following Jetty classes to INFO level by default because if logging is set to FINE or higher,
 # sensitive information can be leaked into the logs
 org.eclipse.jetty.server.HttpChannel.level=INFO
 org.eclipse.jetty.server.HttpConnection.level=INFO
 org.eclipse.jetty.server.HttpInput.level=INFO
 org.eclipse.jetty.http.HttpParser.level=INFO
 org.eclipse.jetty.io.ssl.SslConnection.level=INFO
```

This configuration logs request data at `INFO` level, preventing data such as password changes from being logged. In situations where you *need* to log all data (for example, if you are debugging an issue in a test environment) change the settings here to `FINE` or `FINEST`. For example:

```
org.eclipse.jetty.server.HttpConnection.level=FINE
```

5.1.2.10. Changes When Interacting With AM and DS

If you've integrated IDM with AM and DS, as described in "*Integrating IDM With the ForgeRock Identity Platform*" in the *Samples Guide*, note the `redirectUri` in your project's `authentication.json` file.

For IDM 6.5, the `redirectUri` will have a value like `http://idm.example.com:8080/`.

For IDM 6, the corresponding `redirectUri` has a corresponding value of `http://idm.example.com:8080/oauthReturn/`.

For IDM 6.5, you can use either endpoint, as long as you're consistent with the corresponding value described in *OAuth 2.0 and OpenID Connect 1.0 Client Settings*

5.2. Enabling New Features in IDM

If you are updating from a previous IDM release, read this section and follow the steps required for each feature that you want to enable in the updated deployment.

5.2.1. Enabling Queued Synchronization

IDM now supports queued synchronization, which allows you to queue implicit synchronization activity on actions that would otherwise trigger an immediate implicit synchronization. Several changes are necessary to turn this feature on when updating from a previous version of IDM:

Updating Databases and Configurations for Queued Synchronization

1. Update your IDM database to add the new `syncqueue` and `locks` tables by running either `create_syncqueue.sql` or `create_syncqueue.ldif` (depending on your database type), which can be found in `bin/update/scripts/`*`database-type`*`/`.

2. Update your repository configuration files to include the new `locks` and `sync/queue` mappings in the `explicitMapping` resource map. For `repo.jdbc.json`, add:

```
"locks" : {
    "table" : "locks",
    "objectToColumn" : {
        "_id" : "objectid",
        "_rev" : "rev",
        "nodeId" : "nodeid"
    }
},
"sync/queue" : {
    "table" : "syncqueue",
    "objectToColumn" : {
        "_id" : "objectid",
        "_rev" : "rev",
        "syncAction" : "syncAction",
        "resourceCollection" : "resourceCollection",
        "resourceId" : "resourceId",
        "mapping" : "mapping",
        "objectRev" : "objectRev",
        "oldObject" : {"column" : "oldObject", "type" : "JSON_MAP"},
        "newObject" : {"column" : "newObject", "type" : "JSON_MAP"},
        "context" : {"column" : "context", "type" : "JSON_MAP"},
        "state" : "state",
        "nodeId" : "nodeId",
        "remainingRetries" : {"column" : "remainingRetries", "type" : "NUMBER"},
        "createDate" : "createDate"
    }
},
```

For `repo.ds-external.json`, add:

```
"locks" : {
  "dnTemplate": "ou=locks,dc=openidm,dc=forgerock,dc=com",
  "objectClasses": [ "uidObject", "fr-idm-lock" ],
  "properties": {
    "_id": {
      "type": "simple", "ldapAttribute": "uid", "isRequired": true, "writability": "createOnly"
    },
    "nodeId": {
      "type": "simple", "ldapAttribute": "fr-idm-lock-nodeid"
    }
  }
},
"sync/queue" : {
  "dnTemplate": "ou=queue,ou=sync,dc=openidm,dc=forgerock,dc=com",
  "objectClasses": [ "uidObject", "fr-idm-syncqueue" ],
  "properties": {
    "_id": {
      "type": "simple", "ldapAttribute": "uid", "isRequired": true, "writability": "createOnly"
    },
    "syncAction": {
      "type": "simple", "ldapAttribute": "fr-idm-syncqueue-syncaction"
    },
    "resourceCollection": {
      "type": "simple", "ldapAttribute": "fr-idm-syncqueue-resourcecollection"
    },
    "resourceId": {
      "type": "simple", "ldapAttribute": "fr-idm-syncqueue-resourceid"
    },
    "mapping": {
      "type": "simple", "ldapAttribute": "fr-idm-syncqueue-mapping"
    },
    "objectRev": {
      "type": "simple", "ldapAttribute": "fr-idm-syncqueue-objectRev"
    },
    "oldObject": {
      "type": "json", "ldapAttribute": "fr-idm-syncqueue-oldobject"
    },
    "newObject": {
      "type": "json", "ldapAttribute": "fr-idm-syncqueue-newobject"
    },
    "context": {
      "type": "json", "ldapAttribute": "fr-idm-syncqueue-context"
    },
    "state": {
      "type": "simple", "ldapAttribute": "fr-idm-syncqueue-state"
    },
    "nodeId": {
      "type": "simple", "ldapAttribute": "fr-idm-syncqueue-nodeid"
    },
    "remainingRetries": {
      "type": "simple", "ldapAttribute": "fr-idm-syncqueue-remainingretries"
    },
    "createDate": {
      "type": "simple", "ldapAttribute": "fr-idm-syncqueue-createdate"
    }
  }
}
```

3. Update the sync mappings you wish to enable queued synchronization on to include the new `queuedSync` property:

```
"queuedSync" : {
    "enabled" : true,
    "pageSize" : 100,
    "pollingInterval" : 1000,
    "maxQueueSize" : 20000
},
```

For more information about this feature, see "Queued Synchronization" in the *Integrator's Guide*.

5.2.2. Enabling Privileges

Privileges are a new feature of internal roles, which allow for delegating certain administrative privileges to users, without needing to assign a full administrator role. An example where this may be useful is for support personnel who may need the ability to manage users, but shouldn't be able to manage other aspects of IDM:

1. Before proceeding further, ensure you have run the required database scripts referenced in "Database Changes".
2. Update your `repo.jdbc.json` or `repo.ds.json` files to include temporal constraints and privileges for internal roles:

```
"internal/role" : {
    "table" : "internalrole",
    "objectToColumn" : {
        "_id" : "objectid",
        "_rev" : "rev",
        "name" : "name",
        "description" : "description",
        "temporalConstraints" : { "column" : "temporalConstraints", "type" : "JSON_LIST" },
        "condition" : "conditional",
        "privileges" : { "column" : "privs", "type" : "JSON_LIST" }
    }
},
```

If you are using DS, update your `repo.ds.json` file to include temporal constraints and privileges for internal roles:

```
"internal/role": {
  "dnTemplate": "ou=roles,ou=internal,dc=openidm,dc=forgerock,dc=com",
  "objectClasses": [ "fr-idm-internal-role" ],
  "properties": {
    "_id": {
      "type": "simple", "ldapAttribute": "cn", "isRequired": true, "writability": "createOnly"
    },
    "name": {
      "type": "simple", "ldapAttribute": "fr-idm-name"
    },
    "description": {
      "type": "simple", "ldapAttribute": "description"
    },
    "temporalConstraints": {
      "type": "json", "ldapAttribute": "fr-idm-temporal-constraints", "isMultiValued": true
    },
    "condition": {
      "type": "simple", "ldapAttribute": "fr-idm-condition"
    },
    "privileges" : {
      "type": "json", "ldapAttribute": "fr-idm-privilege", "isMultiValued": true
    }
  }
},
```

3. Update `policy.json` to add privileges-related policies (`policy.js` in `/bin/defaults/script/` has been updated with these new policies):

```
{
    "resource" : "internal/role/*",
    "properties" : [
        {
            "name" : "name",
            "policies" : [
                {
                    "policyId" : "required"
                },
                {
                    "policyId" : "not-empty"
                },
                {
                    "policyId" : "cannot-contain-characters",
                    "params" : {
                        "forbiddenChars" : [
                            "/*"
                        ]
                    }
                }
            ]
        },
        {
            "name" : "privileges",
            "policies" : [
                {
                    "policyId" : "valid-type",
                    "params" : {
                        "types" : [
                            "array"
                        ]
                    }
                },
                {
                    "policyId" : "valid-array-items",
                    "params" : {
                        "properties" : [
                            {
                                "name" : "name",
                                "policies" : [
                                    {
                                        "policyId" : "required"
                                    },
                                    {
                                        "policyId" : "not-empty"
                                    },
                                    {
                                        "policyId" : "valid-type",
                                        "params" : {
                                            "types" : [
                                                "string"
                                            ]
                                        }
                                    }
                                ]
                            },
                            {
                                "name" : "path",
                                "policies" : [
                                    {
                                        "policyId" : "required"
                                    },
                                    {
                                        "policyId" : "not-empty"
                                    },
                                    {
                                        "policyId" : "cannot-contain-characters",
                                        "params" : {
                                            "forbiddenChars" : [
                                                "/*"
                                            ]
                                        }
                                    },
                                    {
                                        "policyId" : "valid-privilege-path"
                                    }
                                ]
                            },
                            {
                                "name" : "accessFlags",
                                "policies" : [
                                    {
                                        "policyId" : "required"
                                    },
                                    {
                                        "policyId" : "not-empty"
                                    },
                                    {
                                        "policyId" : "valid-type",
                                        "params" : {
                                            "types" : [
                                                "array"
                                            ]
                                        }
                                    },
                                    {
                                        "policyId" : "valid-accessFlags-object"
                                    }
                                ]
                            },
                            {
                                "name" : "actions",
                                "policies" : [
                                    {
                                        "policyId" : "required"
                                    },
                                    {
                                        "policyId" : "valid-type",
                                        "params" : {
                                            "types" : [
                                                "array"
                                            ]
                                        }
                                    }
                                ]
                            },
                            {
                                "name" : "permissions",
                                "policies" : [
                                    {
                                        "policyId" : "required"
                                    },
                                    {
                                        "policyId" : "not-empty"
                                    },
                                    {
                                        "policyId" : "valid-type",
                                        "params" : {
                                            "types" : [
                                                "array"
                                            ]
                                        }
                                    },
                                    {
                                        "policyId" : "valid-permissions"
                                    }
                                ]
                            },
                            {
                                "name" : "filter",
                                "policies" : [
                                    {
                                        "policyId" : "valid-type",
                                        "params" : {
                                            "types" : [
                                                "string",
                                                "null"
                                            ]
                                        }
                                    }
                                ]
                            }
                        ]
                    }
                }
            ]
        }
    ]
}
```

4. **Warning**

   Because this step involves deleting data (the `Roles` column), we strongly recommend making a backup of your repository prior to making this change.

   (Optional) You can drop the `Roles` column from the `internaluser` table by running `alter_internaluser.sql` or `alter_internaluser.ldif` in your database.

   Once this is run, you can remove `roles` from the `internal/user` resource in your `repo.jdbc.json` or `repo.ds.json` file.

   The `INTERNAL_USER` object in `authentication.json` should also be updated to use `authzRoles` instead of roles:

```
{
    "name" : "INTERNAL_USER",
    "properties" : {
        "queryId" : "credential-internaluser-query",
        "queryOnResource" : "internal/user",
        "propertyMapping" : {
            "authenticationId" : "username",
            "userCredential" : "password",
            "userRoles" : "authzRoles"
        },
        "defaultUserRoles" : [ ]
    },
    "enabled" : true
},
```

5. To allow the new privilege endpoint to be called, the following patterns need to be added to `access.js`:

```
{
    "pattern"    : "privilege",
    "roles"      : "*",
    "methods"    : "action",
    "actions"    : "listPrivileges"
},
{
    "pattern"    : "privilege/*",
    "roles"      : "*",
    "methods"    : "read",
    "actions"    : "*"
},
```

Note

If you already have custom access rules, take a moment to assess these rules before trying to apply new privileges. Any custom access rules created in `access.js` will be applied before privileges are considered, which may prevent the new privileges from being correctly applied.

Note

The `ownIDP()` `customAuthz` script referenced is broad by default, to accommodate any social identity providers you may use. For a production deployment, this should be replaced with `ownRelationship()` `customAuthz` scripts, applied to each of the specific social identity providers you intend to use. For example, if you wish to enable Google and Facebook as social identity providers, the `managed/*` pattern calling `ownIDP()` should be changed to:

```
{
    "pattern"    : "managed/google",
    "roles"      : "internal/role/openidm-authorized",
    "methods"    : "read",
    "actions"    : "*",
    "customAuthz" : "ownRelationship()"
},
{
    "pattern"    : "managed/facebook",
    "roles"      : "internal/role/openidm-authorized",
    "methods"    : "read",
    "actions"    : "*",
    "customAuthz" : "ownRelationship()"
}
```

For more information about social identity providers, see "*Configuring Social Identity Providers*" in the *Integrator's Guide*.

6. Finally, add and enable `enableDynamicRoles` in the `JWT_SESSION` session module in `authentication.json`:

```
"enableDynamicRoles" : true,
```

For more information about privileges, see "Privileges and Delegation" in the *Integrator's Guide*.

5.2.3. Enabling Dynamic Role Calculation

To enable dynamically recalculating role assignments without requiring the user to log out and back in, open `authentication.json`, and enable the `enableDynamicRoles` property in the `JWT_SESSION` session module:

```
"enableDynamicRoles" : true
```

This will also enable privileges on internal roles, but can be used as its own feature even if you do not plan to use privileges.

> **Note**
>
> If your IDM instance has a large number of role assignments, performance may be impacted by enabling this feature.

5.2.4. Adding Thread IDs to Log Messages

IDM can now include the thread ID for the thread generating a log message, which can help when debugging. To enable this feature, open `logging.properties` and adjust the `ConsoleHandler` and `FileHandler` formatters to use `ThreadIdLogFormatter`:

```
java.util.logging.ConsoleHandler.formatter = org.forgerock.openidm.logger.ThreadIdLogFormatter
java.util.logging.FileHandler.formatter = org.forgerock.openidm.logger.ThreadIdLogFormatter
```

5.2.5. Access to Notifications

To enable access to the new notifications service used for the end user UI, make the following changes:

1. Run the database update script (either `create_notifications.sql` or `create_notifications.ldif`) provided in `bin/update/scripts/database-type/`.

2. Update your `repo.jdbc.json` file, adding the `internal/notification` mapping to your `genericMapping` object:

   ```
   "internal/notification" : {
       "mainTable" : "notificationobjects",
       "propertiesTable" : "notificationobjectproperties",
       "searchableDefault" : false,
       "properties" : {
           "/createDate" : {
               "searchable" : true
           },
           "/notificationType" : {
               "searchable" : true
           }
       }
   }
   ```

   If you are using DS, update the `genericMapping` resource collection in your `repo.ds.json` file to include the new `internal/notification` mapping:

   ```
   "internal/notification" : {
     "dnTemplate": "ou=notification,ou=internal,dc=openidm,dc=forgerock,dc=com"
   }
   ```

3. To allow users to see their own information regardless of privileges, add the following patterns to `access.js`:

   ```
   // Grant users access to their own user metadata
   {
       "pattern"   : "internal/usermeta/*",
       "roles"     : "internal/role/openidm-authorized",
       "methods"   : "read",
       "actions"   : "*",
       "customAuthz" : "ownRelationship()"
   },
   // Grant users access to their own notifications
   {
       "pattern"   : "internal/notification/*",
       "roles"     : "internal/role/openidm-authorized",
       "methods"   : "read,delete",
       "actions"   : "*",
       "customAuthz" : "ownRelationship()"
   },
   {
       "pattern"   : "managed/user/*",
       "roles"     : "internal/role/openidm-authorized",
       "methods"   : "read,query",
       "actions"   : "*",
       "customAuthz" : "ownRelationshipCollection(['idps','_meta','_notifications'])"
   ```

```
},
{
    "pattern"   : "notification",
    "roles"     : "internal/role/openidm-authorized",
    "methods"   : "action",
    "actions"   : "deleteNotificationsForTarget",
    "customAuthz" : "request.additionalParameters.target === (context.security.authorization.component
 + '/' + context.security.authorization.id)"
},
{
    "pattern"   : "managed/*",
    "roles"     : "internal/role/openidm-authorized",
    "methods"   : "read",
    "actions"   : "*",
    "customAuthz" : "ownIDP()"
}
```

4. Remove the `access.js` code blocks for `endpoint/usernotifications` and `endpoint/usernotifications/*`.

5. Remove the `exports.createNotification` code block from `onUpdateUser.js`.

6. Replace `onDelete-user-cleanup.js` with `postDelete-notification-cleanup.js`.

7. Remove the following file: `userNotifications.js`.

For more information on the new notification service, see "Configuring Notifications" in the *Integrator's Guide*.

5.2.6. Enabling File Uploads

IDM provides a generic file upload service, that lets you upload and save files either to the filesystem or to the repository. To enable this feature, take the following steps:

1. Run the database update script (either `create_files.sql` or `create_files.ldif`) provided in `bin/update/scripts/`*`database-type`*`/`.

2. Update the `explicitMapping` resource collection in your `repo.jdbc.json` file to include the new file mapping:

```
"file" : {
    "table" : "files",
    "objectToColumn" : {
        "_id" : "objectid",
        "_rev" : "rev",
        "content" : "content"
    }
}
```

If you are using DS, update the `genericMapping` resource collection in your `repo.ds.json` file to include the new file mapping:

```
"file" : {
  "dnTemplate": "ou=file,dc=openidm,dc=forgerock,dc=com"
}
```

For more information about the file upload service, see "Uploading Files to the Server" in the *Integrator's Guide*.

5.2.7. Enabling Oracle UCP

Oracle UCP is a connection pool designed to cache JDBC connections. For IDM 6.5, it is an alternative to HikariCP for Oracle DB, as described in "Setting Up an Oracle DB Repository" in the *Installation Guide*. If you want to use Oracle UCP for IDM 6.5 instead of HikariCP, take the following steps:

1. Find any custom settings that you created in your current `datasource.jdbc-default.json` file.
2. Find the `datasource.jdbc-ucp-oracle.json` file in the `/path/to/openidm/db/oracle/conf` directory, and modify that file as needed for compatibility.
3. Replace the `datasource.jdbc-default.json` file with the newly customized `datasource-jdbc-ucp-oracle.json` file in your project's `conf/` subdirectory.

Chapter 6
How to Report Problems and Provide Feedback

If you have questions regarding ForgeRock Identity Management software that are not answered by the documentation, you can ask questions on the forum at https://forgerock.org/forum/fr-projects/openidm/.

When requesting help with a problem, include the following information:

- Description of the problem, including when the problem occurs and its impact on your operation
- Description of the environment, including the following information:
  - Machine type
  - Operating system and version
  - Repository type and version
  - Java version
  - IDM release version
  - Any patches or other software that might be affecting the problem
- Steps to reproduce the problem
- Any relevant access and error logs, stack traces, or core dumps

Chapter 7
Documentation Updates

"Documentation Change Log" tracks important changes to the documentation:

Documentation Change Log

Date	Description
2022-08-16	- Release of IDM 6.5.2.0.
2021-10-26	- The latest version of the Active Directory password synchronization plugin uses UTC timestamps for logs.
2021-05-10	- Added a caution regarding the BouncyCastle `dump` utility. - Added instructions to "Delete Orphaned Meta Entries" in the *Installation Guide*. - Updated the default `datasource.jdbc-default.json` file configuration for workflow in the *Integrator's Guide*. - `integer` is a supported managed object type. - The latest version of the Active Directory Password Synchronization Plugin supports a new registry key to configure the maximum retry attempts for password changes. For more information, see the registry key, `maxFileRetry` in the *Password Synchronization Plugin Guide*. - The latest version of the Active Directory Password Synchronization Plugin supports a new registry key to configure a search filter to omit users/groups from password syncing. For more information, see the registry key, `userSearchFilterStrict` in the *Password Synchronization Plugin Guide*.
2021-03-05	- Added Ubuntu Linux 18.04 as a supported OS and PostgreSQL 10.x as a supported repository. - Added a note about some configurations not using property substitution in the *Integrator's Guide*. - Added a restriction warning on the production use of the embedded workflow H2 database in the *Integrator's Guide*.
2021-02-04	- Corrected the documentation on the impact of Daylight Savings Time on schedules in the *Integrator's Guide*. - The latest version of the Active Directory Password Synchronization Plugin supports a new registry key that helps prevent infinite password update loops.

Date	Description
	For more information, see the registry key, pwdChangeInterval in the *Password Synchronization Plugin Guide*.
2020-12-07	• Release of IDM 6.5.1.0. • Improved the logging documentation in the *Integrator's Guide*. • Removed the restriction on support for the SSH connector in the *Connector Reference*. This connector is fully supported. • Fixed a documentation bug related to the storage location of metadata in the *Integrator's Guide*. • Added information about validating properties of unknown resources in the *Integrator's Guide*.
2020-10-16	• Updated the path to the custom self-service stage sample project in the *Self-Service REST API Reference*. • Added OPENIDM-15650 to the list of "Known Issues", present in IDM 6.5.0.3 and 6.5.0.4. • Added note regarding user DELETE requests for Salesforce connector.
2020-08-19	Added Oracle 19c to the list of "Supported Repositories".
2020-07-18	Added a description for the maxTokenSize property of the IWA authentication module.
2020-06-29	Initial release of IDM 6.5.0.4. The following items were added: • Added a note to manually update the conf/ui-themeconfig.json after updating IDM. For information, see "Updating to Version 6.5.2.0" in the *Installation Guide*.
2020-05-27	• Added missing TOKEN authentication method and miscellaneous improvements to "*SCIM Connector*" in the *Connector Reference*. • Indicated that the automated update process (for updates to patch bundle releases) is not supported on Windows.
2020-05-12	Fixed some errors in the procedure on configuring a secure connection to a JDBC repository.
2020-04-08	Update PostgreSQL repository instructions to indicate that index tuning is required.
2020-04-02	Clarified the changes required to managed.json during an update to ensure that conditional role grants work correctly.
2020-03-24	• Fixed outdated Bootstrap version references in the *Integrator's Guide* • Added SocketTimeout to migration properties in the *Installation Guide* • Corrected openidm.patch scripting reference in the *Integrator's Guide*

Date	Description
2020-03-12	Add SCIM Connector 1.5.2.0 in the *Connector Release Notes*
2020-03-04	OPENIDM-14055 - Fixed incorrect default port number in Oracle Repository docs OPENIDM-14343 - Clarified update process for customized shell scripts OPENIDM-14353 - Fixed an issue related to bnd file creation in Oracle Repository docs OPENIDM-14383 - Noted that the DS Password Sync Plug-in version must match both the DS and IDM versions OPENIDM-14260 - Remove 'name' property from provisioner file examples OPENIDM-13766 - Clarified that removing managed object core objects and their nested properties requires additional UI customization for the UI to work
2020-02-25	Initial release of IDM 6.5.0.3.
2019-10-18	Added a workaround for the issue related to queued sync mapping names (OPENIDM-14099). See IDM 6.5.0.
2019-10-15	Initial release of IDM 6.5.0.2.
2019-10-11	Revised the password synchronization documentation for a bug related to the configuration of IDM secrets. See "*Synchronizing Passwords With ForgeRock Directory Services (DS)*" in the *Password Synchronization Plugin Guide*.
2019-10-04	Fixed the examples for encryption and decryption using the `openidm.encrypt` and `openidm.decrypt` functions. See "Encrypting and Decrypting Information" in the *Integrator's Guide*.
2019-09-10	Revised the logging documentation to include security advice on logging levels. See "Set the Logging Level" in the *Integrator's Guide* and "Updating `logging.properties`".
2019-08-28	Revised the documentation on setting up an Oracle DB repository to clarify issues related to the OracleUCP connection pool. See "Setting Up an Oracle DB Repository" in the *Installation Guide*.
2019-08-22	Fixed an error in the documentation on authenticating as a different user (`runAs` authentication). See "Authenticating as a Different User" in the *Integrator's Guide*.
2019-08-21	Revised the documentation on authenticating with client certificates ("Authenticating With Client Certificates" in the *Integrator's Guide*).
2019-08-19	Added information on restricting the maximum payload size in HTTP requests ("Restricting the HTTP Payload Size" in the *Integrator's Guide*).
2019-07-23	Corrected error in the Self-Service Reference regarding storage of the JWT token ("The Self-Service Process Flow" in the *Self-Service REST API Reference*).
2019-07-12	Added information on enabling HTTP Strict-Transport-Security. See "Enabling HTTP Strict-Transport-Security" in the *Integrator's Guide*. Removed erroneous reference to JavaScript in Terms & Conditions ("Configuring Terms & Conditions in the Admin UI" in the *Integrator's Guide*).
2019-05-14	Updated the REST API reference to indicate that `_totalPagedResultsPolicy=ESTIMATE` is not implemented in IDM. See "Common REST and IDM" in the *Integrator's Guide*.

Date	Description
	Updated "Transforming Data Types" in the *Integrator's Guide* to include the `object` type.
2019-05-13	Added a note for support of repositories in cloud hosted environments. See "Supported Repositories".
2019-05-02	Republication to fix links to Quartz documentation.
2019-09-10	Initial release of IDM 6.5.0.1.
2019-03-22	The DS Password Synchronization Plugin, version 6.5.0, is supported with DS 6.5.0 and DS 6.5.1. The corresponding compatibility matrix has been updated.
2019-03-05	The restriction on disabling persistent configuration in a production environment has been removed. See "Making Configuration Changes" in the *Integrator's Guide*.
2019-02-21	A section has been added to the release notes, regarding removing `objectproperties` tables if you are using PostgreSQL (see "Removal of Property Tables in PostgreSQL").
2019-02-13	The section describing writing custom UI templates for Activiti Workflows has been changed ("Using Custom Templates for Activiti Workflows" in the *Integrator's Guide*) to reflect the process required for the new End User UI.
2019-02-06	The section describing the configuration of workflows has been changed ("Enabling Workflows" in the *Integrator's Guide*). The `mail` parameter of the Activiti engine is currently not supported (see OPENIDM-11370).
2019-02-01	Fixed a documentation issue in "Encrypting and Decrypting Information" in the *Integrator's Guide*. The process did not work as documented and has been revised.
2019-01-29	Fixed the following documentation issues: • Added a scripting step to clear the `reconprogressstate` column from the `genericobjects` table in the repository after the update process. For more information, see "Upgrade Your Existing Repository" in the *Installation Guide*.
2018-12-13	Fixed the following documentation issues: • OPENIDM-12269 Incorrect migration endpoint in Update section ("Running Your Migration" in the *Installation Guide*). • OPENIDM-12277 - Missing image in Trusted Devices section ("Configuring Trusted Devices on IDM" in the *Integrator's Guide*). • OPENIDM-12267 - Erroneous metric in API metrics reference ("API Metrics available in IDM" in the *Integrator's Guide*). • OPENIDM-12234 - Describe ability to encrypt/decrypt via REST ("Encryption and JSON Blob Code Blocks" in the *Integrator's Guide*).

Appendix A. Release Levels and Interface Stability

This appendix includes ForgeRock definitions for product release levels and interface stability.

A.1. ForgeRock Product Release Levels

ForgeRock defines Major, Minor, Maintenance, and Patch product release levels. The release level is reflected in the version number. The release level tells you what sort of compatibility changes to expect.

Release Level Definitions

Release Label	Version Numbers	Characteristics
Major	Version: x[.0.0] (trailing 0s are optional)	• Bring major new features, minor features, and bug fixes • Can include changes even to Stable interfaces • Can remove previously Deprecated functionality, and in rare cases remove Evolving functionality that has not been explicitly Deprecated • Include changes present in previous Minor and Maintenance releases
Minor	Version: x.y[.0] (trailing 0s are optional)	• Bring minor features, and bug fixes

Release Label	Version Numbers	Characteristics
		• Can include backwards-compatible changes to Stable interfaces in the same Major release, and incompatible changes to Evolving interfaces • Can remove previously Deprecated functionality • Include changes present in previous Minor and Maintenance releases
Maintenance, Patch	Version: x.y.z[.p] The optional `.p` reflects a Patch version.	• Bring bug fixes • Are intended to be fully compatible with previous versions from the same Minor release

A.2. ForgeRock Product Interface Stability

ForgeRock products support many protocols, APIs, GUIs, and command-line interfaces. Some of these interfaces are standard and very stable. Others offer new functionality that is continuing to evolve.

ForgeRock acknowledges that you invest in these interfaces, and therefore must know when and how ForgeRock expects them to change. For that reason, ForgeRock defines interface stability labels and uses these definitions in ForgeRock products.

Interface Stability Definitions

Stability Label	Definition
Stable	This documented interface is expected to undergo backwards-compatible changes only for major releases. Changes may be announced at least one minor release before they take effect.
Evolving	This documented interface is continuing to evolve and so is expected to change, potentially in backwards-incompatible ways even in a minor release. Changes are documented at the time of product release. While new protocols and APIs are still in the process of standardization, they are Evolving. This applies for example to recent Internet-Draft implementations, and also to newly developed functionality.
Deprecated	This interface is deprecated and likely to be removed in a future release. For previously stable interfaces, the change was likely announced in a previous release. Deprecated interfaces will be removed from ForgeRock products.
Removed	This interface was deprecated in a previous release and has now been removed from the product.
Technology Preview	Technology previews provide access to new features that are evolving new technology that are not yet supported. Technology preview features may be functionally incomplete and the function as implemented is subject to

Stability Label	Definition
	change without notice. DO NOT DEPLOY A TECHNOLOGY PREVIEW INTO A PRODUCTION ENVIRONMENT. Customers are encouraged to test drive the technology preview features in a non-production environment and are welcome to make comments and suggestions about the features in the associated forums. ForgeRock does not guarantee that a technology preview feature will be present in future releases, the final complete version of the feature is liable to change between preview and the final version. Once a technology preview moves into the completed version, said feature will become part of the ForgeRock platform. Technology previews are provided on an “AS-IS” basis for evaluation purposes only and ForgeRock accepts no liability or obligations for the use thereof.
Internal/Undocumented	Internal and undocumented interfaces can change without notice. If you depend on one of these interfaces, contact ForgeRock support or email info@forgerock.com to discuss your needs.

Appendix B. Getting Support

For more information and resources about IDM and ForgeRock support, see the following sections:

B.1. Accessing Documentation Online

ForgeRock publishes comprehensive documentation online:

- The ForgeRock Knowledge Base offers a large and increasing number of up-to-date, practical articles that help you deploy and manage ForgeRock software.

  While many articles are visible to community members, ForgeRock customers have access to much more, including advanced information for customers using ForgeRock software in a mission-critical capacity.

- ForgeRock product documentation, such as this document, aims to be technically accurate and complete with respect to the software documented. It is visible to everyone and covers all product features and examples of how to use them.

B.2. Using the ForgeRock.org Site

The ForgeRock.org site has links to source code for ForgeRock open source software, as well as links to the ForgeRock forums and technical blogs.

If you are a *ForgeRock customer*, raise a support ticket instead of using the forums. ForgeRock support professionals will get in touch to help you.

B.3. Getting Support and Contacting ForgeRock

ForgeRock provides support services, professional services, training through ForgeRock University, and partner services to assist you in setting up and maintaining your deployments. For a general overview of these services, see https://www.forgerock.com.

ForgeRock has staff members around the globe who support our international customers and partners. For details on ForgeRock's support offering, including support plans and service level agreements (SLAs), visit https://www.forgerock.com/support.