Release Notes

/ Directory Services 6.5

Latest update: 6.5.6

Mark Craig

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

Abstract

Notes covering ForgeRock® Directory Services features, fixes, and known issues.

Table of Contents

About Directory Services 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.

Directory Services software provides an LDAPv3-compliant directory service, developed for the Java platform, delivering a high-performance, highly available, and secure store for the identities managed by your organization. *Read these notes before you install or upgrade Directory Services software.*

The easy installation process, combined with the power of the Java platform, makes this the simplest and fastest directory service to deploy and manage. Directory Services software comes with plenty of tools. Directory Services software also offers REST access to directory data over HTTP.

Directory Services software is free to download, evaluate, and use for developing your applications and solutions. ForgeRock offers training and support subscriptions to help you get the most out of your deployment.

These release notes cover the following topics:

- Hardware and software prerequisites for installing and upgrading Directory Services 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
- Documentation updates
- Definitions for release levels and interface stability
- Getting support

See the Installation Guide after you read these *Release Notes*. The *Installation Guide* also covers upgrade for Directory Services software.

Chapter 1
What's New

This chapter covers new capabilities in Directory Services 6.5.

Maintenance Releases

ForgeRock maintenance 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.

- **DS 6.5.6 Maintenance Release**

  DS 6.5.6 is the latest release targeted for DS 6.5.x deployments, and can be downloaded from the *ForgeRock Backstage* website. To view the list of fixes in this release, see DS 6.5.6.

  The release can be deployed as an initial deployment or updated from an existing DS 6.5.x deployment.

What's New in 6.5

DS 6.5.6

- The **supportextract** command now collects environment variables used in configuration expressions.

DS 6.5.5

- DS servers now more effectively calculate reservable memory when using G1 garbage collection, and reduce the risk of long fsync pauses.

  This change introduces a `ds-mon-db-cache-size-total` metric to track the maximum size of the database cache. It also changes the `ds-mon-db-log-size-active` metric to reflect only live data.

- The **supportextract** command now uses the **jcmd** command, if available, for heap dumps. Otherwise, it uses the **jmap** command.

DS 6.5.4

- There are no new features introduced in DS 6.5.4, only bug fixes.

DS 6.5.3

This release of Directory Services software includes the following new feature:

- ADS-Certificate Key Pair and Peer DS Servers' Trusted Public Keys Support in PKCS#11 Modules

  DS servers now support storing the `ads-certificate` key pair and peer DS servers' trusted public keys in a PKCS#11 module, such as a hardware security module (HSM). This means you can store the keys used to secure replication traffic and to protect symmetric keys in an HSM. Previously, DS servers supported use of a PKCS#11 module only to store the keys used to secure other communications.

  Note

  DS servers support PKCS#11 modules through the JVM. How to configure the JVM to allow DS servers to use your module, how to generate keys in the module, and how to export and import public key certificates all depend on your specific HSM/PKCS#11 module.

  For details on how to perform such actions, see your PKCS#11 module's documentation.

  Before trying this feature, perform these tests:

  - Test that the PKCS#11 module supports multiple aliases for the same certificate.
  - Test that the PKCS#11 module supports generating a key pair with an RSA self-signed certificate and a key size of 2048 bits.
  - Test the replication topology using the default JKS `ads-truststore` implementation.

    Verify that replication functions properly *before* using the PKCS#11 module.

  After validating the results of the tests, use the new feature by performing these steps for each server:

  1. Configure JVM security to enable use of the PKCS#11 module for the server.
  2. Generate a key pair using an RSA self-signed certificate and a key size of 2048 bits with the alias `ads-certificate` on the PKCS#11 module.
  3. After generating the key pair:

     a. Export the `ads-certificate` certificate, and reimport it on the PKCS#11 module using the MD5 fingerprint as the certificate alias.

        For example, if the `ads-certificate` certificate MD5 fingerprint is `07:35:80:D8:F3:CE:E1:39:9C:D0:73:DB:6C:FA:CC:1C`, reimport the certificate with the alias `073580D8F3CEE1399CD073DB6CFACC1C`.

b. Prepare LDIF to update `cn=admin data` for the new certificate.

The LDIF adds the new certificate as an instance key, and sets the key ID for the current server to use the key. In the following example LDIF, the certificate alias is `073580D8F3CEE1399CD073DB6CFACC1C`, and the server *`hostname:admin-port`* combination is `opendj.example.com:4444`:

```
dn: ds-cfg-key-id=073580D8F3CEE1399CD073DB6CFACC1C,cn=instance keys, cn=admin data
changetype: add
ds-cfg-key-id: 073580D8F3CEE1399CD073DB6CFACC1C
ds-cfg-public-key-certificate;binary:: MIIB6zCCAVSgAwIBAgIEDKSUFjANBgkqhkiG9w0BA
QUFADA6MRswGQYDVQQKExJPcGVuREogQ2VydGlmaWNhdGUxGzAZBgNVBAMTEm9wZW5hbS5leGFtcGxl
LmNvbTAeFw0xMzAyMDcxMDMwMzNaFw0zMzAyMDIxMDMwMzNaMDoxGzAZBgNVBAoTEk9wZW5ESiBDZXJ
0aWZpY2F0ZTEbMBkGA1UEAxMSb3BlbmFtLmV4YW1wbGUuY29tMIGfMA0GCSqGSIb3DQEBAQUAA4GNAD
CBiQKBgQCfGLAiUOz4sC8CM9T5DPTk9V9ErNC8N59XwBt1aN7UjhQl4/JZZsetubtUrZBLS9cRrnYdZ
cpFgLQNEmXifS+PdZ0DJkaLNFmd8ZX0spX8++fb4SkkggkmNRmi1fccDQ/DHMlwl7kk884lXummrzcD
GbZ7p4vnY7y7GmD1vZSP+wIDAQABMA0GCSqGSIb3DQEBBQUAA4GBAJciUzUP8T8A9VV6dQB0SYCNG1o
7IvpE7jGVZh6KvM0m5sBNX3wPbTVJQNij3TDm8nx6yhi6DUkpiAZfz/OBL5k+WSw80TjpIZ2+klhP1s
srsST4Um4fHzDZXOXHR6NM83XxZBsR6MazYecL8CiGwnYW2AeBapzbAnGn1J831q1q
objectClass: top
objectClass: ds-cfg-instance-key

dn: cn=opendj.example.com:4444,cn=Servers,cn=admin data
changetype: modify
replace: ds-cfg-key-id
ds-cfg-key-id: 073580D8F3CEE1399CD073DB6CFACC1C
```

Do not yet update `cn=admin data`. You must not change the server key ID until the server is ready to use the PKCS#11 module.

4. Edit the `ads-truststore` trust manager provider configuration to access the PKCS#11 module with the following properties:

trust-store-file

Leave this unchanged.

trust-store-pin

Set this to the PIN for the PKCS#11 module.

You can avoid exposing secrets in the configuration by using expressions. For details, see "Using Configuration Property Value Substitution" in the *Administration Guide*.

trust-store-type

Set this to `PKCS11`.

5. Stop the DS server.

6. Using the **ldifmodify** command while the server is stopped, update `cn=admin data` for the server with the LDIF you prepared.

This step changes the key ID for the server, letting it use its keys held in the PKCS#11 module.

7. *On another, running server replica*, use the **ldapmodify** command to update `cn=admin data` for the server with the LDIF you prepared.

   Replication propagates the change to the other running server replicas.

   This step changes the key ID for the server, letting other servers trust its new certificate once it restarts.

8. Restart the DS server.

DS 6.5.2

- There are no new features introduced in DS 6.5.2, only bug fixes.

DS 6.5.1

- There are no new features introduced in DS 6.5.1, only bug fixes.

DS 6.5.0

This release of Directory Services software includes the following new features:

- **Connection Limiting**

  DS servers now allow you to limit the number of concurrent connections per client.

  For details, see "Limiting and Restricting Client Connections" in the *Administration Guide*.

- **Data Distribution**

  DS proxy servers now support simple, non-elastic data distribution.

  You can configure a proxy server to equitably distribute LDAP write requests across multiple replication partitions to scale the directory service horizontally. As the present implementation does not permit elastic scaling or data redistribution, make sure that you understand the documented constraints of the present implementation before deploying it in production.

  For an example, see "Scaling Out Using Data Distribution" in the *Deployment Guide*.

  The mechanism is described in "Routing Requests to Remote Directory Servers" in the *Administration Guide*.

- **Database Caching**

A new DS directory server uses shared cache by default for all JE database backends. As a result, you are no longer required to set the database cache size using the `db-cache-percent` or `db-cache-size` setting for each backend.

It remains possible to use these settings if necessary by configuring them appropriately. For details, see "Database Cache Settings" in the *Administration Guide*.

- **Logging**
  - DS servers now support sending access log messages to standard output.

    For details, see "Sending JSON Access Logs to Standard Output" in the *Administration Guide*.

    When the new handler is used, standard output from the server includes a mix of JSON for access messages and non-JSON DS-format server event messages.
  - Common Audit logging now supports blacklisting log message fields to prevent them from showing up in log messages.

    For an example, see "Blacklisting Log Message Fields" in the *Administration Guide*.
  - Common Audit logging now supports writing multiple file-based logs to the same directory by setting a different `log-file-name-prefix` for each file-based log.
- **Monitoring**

  DS servers now provide health status checks for anonymous requests over HTTP and LDAP. This allows a remote application to check that a server is "alive" and "healthy".

  Anonymous HTTP requests can retrieve "alive" and "healthy" status codes. Anonymous LDAP requests can retrieve "alive" and "healthy" boolean values.

  The "alive" and "healthy" status indicates that the server has passed its own internal tests. It is not, however, a guarantee that the server is free from other errors. If a server is *not* "alive," it requires administrative intervention. If a server is *not* "healthy," temporarily route requests to another server.

  When a server is not "alive" or "healthy," a user with privileges to read monitoring information receives health status error messages in the body of the HTTP response, and can obtain health status error messages over LDAP as described below. No error messages are returned in response to anonymous requests.

  For examples demonstrating how to use this feature, see the following documentation:

  - "Monitoring Liveness over HTTP" in the *Administration Guide*
  - "Monitoring Ability to Handle Requests Over HTTP" in the *Administration Guide*
  - "Monitoring Health Status With Prometheus" in the *Administration Guide*

- "Monitoring Health Status Anonymously Over LDAP" in the *Administration Guide*
- "Monitoring Health Status Details Over LDAP" in the *Administration Guide*

When you upgrade DS servers to this release, the anonymously accessible HTTP endpoints are not configured. To add the endpoints on an upgraded server that lacks them, use the **dsconfig** command:

```
$ dsconfig \
create-http-endpoint \
--endpoint-name /alive \
--type alive-endpoint \
--set enabled:true \
--set authorization-mechanism:HTTP Anonymous \
--set authorization-mechanism:HTTP Basic \
--hostname opendj.example.com \
--port 4444 \
--bindDN "cn=Directory Manager" \
--bindPassword password \
--trustAll \
--no-prompt
$ dsconfig \
create-http-endpoint \
--endpoint-name /healthy \
--type healthy-endpoint \
--set enabled:true \
--set authorization-mechanism:HTTP Anonymous \
--set authorization-mechanism:HTTP Basic \
--hostname opendj.example.com \
--port 4444 \
--bindDN "cn=Directory Manager" \
--bindPassword password \
--trustAll \
--no-prompt
```

- **Platform Integration**

  When setting up a directory server for use with other ForgeRock Identity Platform™ component products, you can use available setup profiles to greatly simplify initial configuration.

  For details, see "*Using Directory Server Setup Profiles*" in the *Installation Guide*.

  In addition, you can use profiles in embedded servers. For details, see "To Set Up an Embedded Server" in the *Developer's Guide*.

- **Replication**

  When configuring a replication server on a multi-homed system with multiple IP addresses, you can now specify which listen addresses to use.

  Set the property, `listen-address`, as shown in "Choosing the Listen Address for Replication" in the *Administration Guide*.

- **Support**

  The DS support extract command, **supportextract**, now ships with DS server software, making it easier to capture troubleshooting information.

  The command works on all supported platforms.

  For details, see "*supportextract — extract support data*" in the *Reference*.

Product Improvements

DS 6.5.6

- No additional improvements in this release.

DS 6.5.5

- OPENDJ-7073: PKCS#11 key managers and trust managers now support non-default types. The default type is `PKCS11`.

DS 6.5.4

- There were no product improvements in this release, only bug fixes.

DS 6.5.3

- OPENDJ-5600: The supportextract tool should capture stack traces with jcmd
- OPENDJ-5960: The supportextract tool should gather basic changelogDb information
- OPENDJ-6163: The supportextract tool needs to gather archived-configs
- OPENDJ-6422: Make the supportextract tool compliant with JVM unified logging framework
- OPENDJ-6830: The supportextract tool should capture stack traces in server.out with SIGQUIT
- OPENDJ-6929: Support storing ads-certificate key-pair and other instance public keys in an HSM
- OPENDJ-6930: Increase interoperability with HSMs when protecting and distributing symmetric keys

DS 6.5.2

- OPENDJ-6125: The supportextract tool needs to gather the rootUser and monitorUser ldif files

- OPENDJ-6128: The supportextract tool needs to gather Profile and Data Information

DS 6.5.1

This release of Directory Services software includes the following enhancement:

- **Tools Improvements**

  All **setup** command profiles now allow you to set the domain or the base DN. For details, see "*Using Directory Server Setup Profiles*" in the *Installation Guide*.

DS 6.5.0

This release of Directory Services software includes the following enhancements:

- **DevOps**

  Administrative tasks affecting directory server backends no longer modify the server configuration.

  This supports DevOps and other deployments that require an immutable configuration.

- **Embedded Server**

  Embedded directory servers now support both online and offline modes for importing LDIF and rebuilding indexes.

- **Examples**

  The server software now includes an example password storage scheme extension.

  For details, unzip the file `/path/to/opendj/example-pwdscheme.zip` and see the `README` file inside.

- **Indexing**

  The `debugsearchindex` attribute, used for investigating how searches are indexed, now holds JSON values that are significantly easier to read.

  For examples, see "Clarifying Which Indexes Are Used by a Search" in the *Administration Guide*.

- **Java Support**

  DS software now supports Java 8 and Java 11.

- **LDAP Support**

  DS servers now support the experimental LDAP Relax Rules control, as described in The LDAP Relax Rules Control (Internet-Draft).

- **Logging**

- DS servers now maintain a filtered JSON LDAP access log in addition to the primary (unfiltered) access log.

  This log includes messages of interest to developers and administrators. For details, see "To Configure Filtered JSON LDAP Access Logs" in the *Administration Guide*.

- Common Audit logging now supports logging to a PostgreSQL.

  For details, see "Configuring JDBC Access Logs" in the *Administration Guide*.

- **Monitoring**

  The monitoring dashboard samples now show the number of available DS servers.

- **Password Management**

  DS servers now support using proxied authorization with the LDAP password modify extended operation.

- **Replication**

  - DS servers now support strings as replication group IDs.

    For an example, see "Replication Groups" in the *Administration Guide*.

  - Standalone replication servers now support querying `cn=changelog`.

  - DS servers now provide a command to debug issues with the file-based changelog database, "*changelogstat — debug changelog and changenumber files*" in the *Reference*.

- **REST to LDAP**

  REST to LDAP now supports returning `null`-value fields in JSON resources.

  To enable this feature, set `return-null-for-missing-properties:true` for a Rest2ldap endpoint or `"returnNullForMissingProperties": true` in the REST to LDAP Gateway configuration file, `rest2ldap.json`.

  By default, a REST to LDAP mapping omits JSON fields for LDAP attributes that have no values. For example, the following entry is missing a value for the optional `description`:

  ```
  dn: uid=bjensen,ou=people,dc=example,dc=com
  objectClass: person
  uid: bjensen
  cn: Babs Jensen
  sn: Jensen
  ```

  A REST to LDAP mapping which maps the `uid`, `cn`, `sn`, and `description` attributes could return the following JSON:

```
{
     "id": "bjensen",
     "fullName": "Babs Jensen",
     "familyName": "Jensen"
     }
```

With the setting enabled, the mapping returns a JSON resource with a `"description"` field:

```
{
     "id": "bjensen",
     "fullName": "Babs Jensen",
     "familyName": "Jensen",
     "description": null
     }
```

- **Security**
  - DS servers no longer overwrite a destination server's secret keys when configuring replication.

    As a result, you no longer need to configure replication before configuring data confidentiality, or before configuring an encrypted password storage scheme such as 3DES, AES, or Blowfish.
  - DS fingerprint certificate mappers now support SHA-256 as a fingerprint algorithm.
- **Tools**
  - The **status** command has been rewritten.

    For details, see "Important Changes to Existing Functionality".
  - The **manage-account** command now supports the following additional subcommands:
    - **clear-account-expiration-time**
    - **clear-authentication-failure-times**
    - **clear-grace-login-use-times**
    - **clear-last-login-time**
    - **clear-password-changed-by-required-time**
    - **clear-password-changed-time**
    - **clear-password-expiration-warned-time**
    - **clear-password-history**
    - **clear-password-is-reset**
    - **set-account-expiration-time**

- **set-authentication-failure-times**
- **set-grace-login-use-times**
- **set-last-login-time**
- **set-password-changed-by-required-time**
- **set-password-changed-time**
- **set-password-expiration-warned-time**
- **set-password-is-reset**

In addition, the actions can be performed over LDAP independently of the command by using the LDAP Password Policy State extended operation with OID `1.3.6.1.4.1.26027.1.6.1`. DS servers implement a class for this operation. See the Javadoc for PasswordPolicyStateExtendedOperation.

- DS command-line tools now have options to specify the keystore or truststore type when securing connections and using client authentication or authenticating the server.

  For details about the options available with each command, see "Tools Reference" in the *Reference*.

- The DS **ldapmodify** and **ldappasswordmodify** commands now report password policy error codes when applicable.
- Templates for the **makeldif** command now support a `<DateTime>` tag to generate timestamp values in LDIF attributes.

  For details, see "*makeldif.template — template file for the makeldif command*" in the *Reference*.

- **Upgrade**

  You can now upgrade mutable DS server data after upgrading the immutable binary files and configuration. Use the `--dataOnly` option with the **upgrade** command.

  This improvement is available when upgrading from DS 6.0.0 or later releases.

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 for running Directory Services software in production. It covers the following topics:

- Downloading Directory Services software
- Choosing hardware
- Choosing an operating system
- Preparing the Java environment
- Choosing an application server when using the DSML or REST to LDAP gateway
- Assigning FQDNs when using replication
- Synchronizing System Clocks For Replication
- Using appropriately signed digital certificates

Downloading Directory Services Software

The ForgeRock BackStage download site provides access to ForgeRock releases. ForgeRock releases are thoroughly validated for ForgeRock customers who run the software in production deployments, and for those who want to try or test a given release.

"Directory Services Software" describes the available software.

Directory Services Software

File	Description
DS-6.5.6.zip	Cross-platform distribution of the server software. Pure Java, high-performance server that can be configured as: • An LDAPv3 directory server with the additional capability to serve directory data to REST applications over HTTP. • An LDAPv3 directory proxy server providing a single point of access to underlying directory servers.

File	Description
	• A replication server handling replication traffic with directory servers and with other replication servers, receiving and sending changes to directory data. Server distributions include command-line tools for installing, configuring, and managing servers. The tools make it possible to script all operations. By default, this file unpacks into an `opendj/` directory.
`DS-6.5.6.msi`	Microsoft Windows native installer for the server software. By default, this installs files into a `C:\Program Files (x86)\OpenDJ\` directory.
`DS_6.5.6-1_all.deb`	Server software native packages for Debian and related Linux distributions. By default, this installs files into an `/opt/opendj/` directory.
`DS-6.5.6-1.noarch.rpm`	Server software native packages for Red Hat and related Linux distributions. By default, this installs files into an `/opt/opendj/` directory.
`DS-dsml-servlet-6.5.6.war`	Cross-platform DSML gateway web archive.
`DS-rest2ldap-servlet-6.5.6.war`	Cross-platform REST to LDAP gateway web archive.
`DS-monitoring-dashboard-samples-6.5.6.zip`	Sample Grafana dashboard demonstrating how to graph DS server metrics stored in a Prometheus database. You are responsible for adapting the sample to suit your production requirements. These resources are provided for *demonstration purposes only*. Commercial support for the ForgeRock DevOps Examples is not available from ForgeRock. For details on how to try the sample dashboard, see the `README.md` file delivered inside the .zip file.

Choosing Hardware

Thanks to the underlying Java platform, Directory Services software runs well on a variety of processor architectures. Many directory service deployments meet their service-level agreements without the very latest or very fastest hardware.

Fulfilling Memory Requirements

When installing a directory server for evaluation, you need 256 MB memory (32-bit) or 1 GB memory (64-bit) available.

For installation in production, read the rest of this section. You need at least 2 GB memory for a directory server and four times the disk space needed for initial production data in LDIF format. A replicated directory server stores data, indexes for the data, operational attribute data, and historical information for replication. The server configuration trades disk space for performance and resilience, compacting and purging data for good performance and for protection against temporary outages. In addition, leave space for growth in database size as client applications modify and add entries over time.

For a more accurate estimate of the disk space needed, import a known fraction of the initial LDIF with the server configured for production. Run tests to estimate change and growth in directory data, and extrapolate from the actual space occupied in testing to estimate the disk space required in production.

Directory servers almost always benefit from caching all directory database files in system memory. Reading from and writing to memory is much faster than reading from and writing to disk storage.

For large directories with millions of user directory entries, there might not be room to install enough memory to cache everything. To improve performance in such cases, use quality solid state drives either for all directory data, or as an intermediate cache between memory and disk storage.

Fulfilling Minimum Disk Space Requirements

To evaluate DS software, make sure you have 10 GB free disk space for the software and for sample data.

The more data you have, the more disk space you need. Before deploying production systems, make sure you have enough space. For details, see "Planning for High Scale" in the *Deployment Guide*.

Choosing a Processor Architecture

Processor architectures that provide fast single thread execution tend to help Directory Services software deliver the lowest response times. For top-end performance in terms of sub-millisecond response times and of throughput ranging from tens of thousands to hundreds of thousands of operations per second, the latest x86/x64 architecture chips tend to perform better than others.

When deploying DS servers with replication enabled, allow at minimum two CPU cores per server. Allow more CPU cores per server, especially in high-volume deployments or when using CPU-intensive features such as encryption. Single CPU systems seriously limit server performance.

Chip multi-threading (CMT) processors can work well for directory servers providing pure search throughput, though response times are higher. However, CMT processors are slow to absorb hundreds or thousands of write operations per second. Their slower threads get blocked waiting on resources, and thus are not optimal for deployments with high write throughput requirements.

Fulfilling Network Requirements

On systems with fast processors and enough memory to cache directory data completely, the network can become a bottleneck. Even if a single 1 Gb Ethernet interface offers plenty of bandwidth to handle your average traffic load, it can be too small for peak traffic loads. Consider using separate interfaces for administrative traffic and for application traffic.

To estimate the network hardware required, calculate the size of the data returned to applications during peak load. For example, if you expect to have a peak load of 100,000 searches per second, each returning a full 8 KB entry, you require a network that can handle 800 MB/sec (3.2 Gb/sec) throughput, not counting other operations, such as replication traffic.

Fulfilling Storage Requirements

Warning

The directory server does not currently support network file systems such as NFS for database storage. Provide sufficient disk space on local storage such as internal disk or an attached disk array.

For a directory server, storage hardware must house both directory data, including historical data for replication, and server logs. On a heavily used server, you might improve performance by putting access logs on dedicated storage.

Storage must keep pace with throughput for write operations. Write throughput can arise from modify, modify DN, add, and delete operations, and from bind operations when a login timestamp is recorded, and when account lockout is configured, for example.

In a replicated topology, a directory server writes entries to disk when they are changed, and a replication server writes changelog entries. The server also records historical information to resolve potential replication conflicts.

As for network throughput, base storage throughput required on peak loads rather than average loads.

Choosing an Operating System

Directory Services 6.5 software is supported on the following operating systems:

- Linux 2.6 and later
- Microsoft Windows Server 2008, 2008 R2, 2012, 2012 R2, and 2016
- Oracle Solaris 10, 11 (SPARC, x64)

In order to avoid directory database file corruption after crashes or power failures on Linux systems, enable file system write barriers and make sure that the file system journaling mode is ordered. For details on how to enable write barriers and how to set the journaling mode for data, see the options for your file system in the **mount** command manual page.

Setting Maximum Open Files

DS servers need to be able to open many file descriptors, especially when handling thousands of client connections. Linux systems in particular often set a limit of 1024 per user, which is too low to handle many client connections to the DS server.

When setting up your DS server for production use, make sure the server can use at least 64K (65536) file descriptors. For example, when running the server as user `opendj` on a Linux system that uses `/etc/security/limits.conf` to set user level limits, you can set soft and hard limits by adding these lines to the file:

```
opendj soft nofile 65536
opendj hard nofile 131072
```

The example above assumes the system has enough file descriptors available overall. You can check the Linux system overall maximum as follows:

```
$ cat /proc/sys/fs/file-max
204252
```

Setting Maximum Inotify Watches

A directory server backend database monitors file events. On Linux systems, backend databases use the inotify API for this purpose. The kernel tunable `fs.inotify.max_user_watches` indicates the maximum number of files a user can watch with the inotify API. Make sure this tunable is set to at least 512K:

```
$ sysctl fs.inotify.max_user_watches
fs.inotify.max_user_watches = 524288
```

If this tunable is set lower than that, change it as shown in the following example:

```
$ sudo sysctl --write fs.inotify.max_user_watches=524288
[sudo] password for admin:
fs.inotify.max_user_watches = 524288
```

Preventing Interference With Antivirus Software

Prevent antivirus and intrusion detection systems from interfering with DS software.

Before using DS software with antivirus or intrusion detection software, consider the following potential problems:

Interference with normal file access

Antivirus and intrusion detection systems that perform virus scanning, sweep scanning, or deep file inspection are not compatible with DS file access, particularly database file access.

Antivirus and intrusion detection software can interfere with the normal process of opening and closing database working files. They may incorrectly mark such files as suspect to infection due to

normal database processing, which involves opening and closing files in line with the database's internal logic.

Prevent antivirus and intrusion detection systems from scanning database and changelog database files.

At minimum, configure antivirus software to whitelist the DS server database files. By default, exclude the following file system directories from virus scanning:

- `/path/to/opendj/changelogDb/` (if replication is enabled)

  Prevent the antivirus software from scanning these changelog database files.

- `/path/to/opendj/db/`

  Prevent the antivirus software from scanning database files, especially `*.jdb` files.

Port blocking

Antivirus and intrusion detection software can block ports that DS uses to provide directory services.

Make sure that your software does not block the ports that DS software uses. For details, see "Limiting System and Administrative Access" in the *Security Guide*.

Negative performance impact

Antivirus software consumes system resources, reducing resources available to other services including DS servers.

Running antivirus software can therefore have a significant negative impact on DS server performance. Make sure that you test and account for the performance impact of running antivirus software before deploying DS software on the same systems.

Preparing the Java Environment

Directory Services software consists of pure Java applications. Directory Services servers and clients run on any system with full Java support. Directory Services is tested on a variety of operating systems, and supported on those listed in "Choosing an Operating System".

Directory Services 6.5 software requires Java 8 or 11, specifically at least the Java Standard Edition runtime environment, or the corresponding Java Development Kit to compile Java plugins and applications.

Note

ForgeRock validates Directory Services software with OpenJDK and Oracle JDK, and does occasionally run sanity tests with other JDKs such as the IBM JDK and Azul's Zulu. 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

Directory server JE database backends can require additional JVM options. When running a directory server with a 64-bit JVM and less than 32 GB maximum heap size, you must use the Java option, `-XX:+UseCompressedOops`. To use the option, edit the `config/java.properties` file. The following example settings include the option with the arguments for offline LDIF import, for rebuilding backend indexes, and for starting the directory server:

```
import-ldif.offline.java-args=-server -XX:+UseCompressedOops
rebuild-index.offline.java-args=-server -XX:+UseCompressedOops
start-ds.java-args=-server -XX:+UseCompressedOops
```

Make sure you have a required Java environment installed on the system. If your default Java environment is not appropriate, set `OPENDJ_JAVA_HOME` to the path to the correct Java environment, or set `OPENDJ_JAVA_BIN` to the absolute path of the **java** command. The `OPENDJ_JAVA_BIN` environment variable is useful if you have both 32-bit and 64-bit versions of the Java environment installed, and want to make sure you use the 64-bit version.

Running in a Container

For some settings, DS servers depend on system information reported by the JVM to determine defaults. When running DS servers in containers such as Docker, the Java 8 JVM returns information about the operating system that does not reflect container constraints and limits. When using Java 8, manually adjust the settings described below.

Note

Java 11 supports gathering container information, as described in JDK-8146115. This fix was backported to Java 8 update 191, as mentioned in the JDK 8u191 Update Release Notes.

Skip this section when using Java 8 update 191 or later, or Java 11.

If necessary, override automatic CPU detection by specifying the number of CPUs the JVM uses with `-XX:ActiveProcessorCount=`*`count`* in `config/java.properties`.

Before adjusting settings, determine the following container constraints:

- The number of CPU core hardware threads dedicated to the containerized system, which is usually twice the number of CPU cores

- The amount of RAM dedicated to the containerized system

When running DS servers in containers such as Docker, adjust the following settings:

- `num-request-handlers`

  Recommendation: Set this either to 2 or to 1/4 of the number of core hardware threads, whichever is larger.

- `num-worker-threads`

  Recommendation: Set this either to 4 or to 5/8 of the number of core hardware threads, whichever is larger.

- `db-num-cleaner-threads`

  Recommendation: Set this either to 2 or to 1/4 of the number of core hardware threads, whichever is larger.

- `num-update-replay-threads`

  Recommendation: Set this either to 4 or to 1/2 of the number of core hardware threads, whichever is larger.

- `-Xmx` (Java setting limiting maximum heap size)

  To use the option, edit the `config/java.properties` file and restart the server.

  For example, consider a container limited to 8 GB RAM. The following setting limits the maximum heap size to 8 GB when starting the directory server:

  ```
  start-ds.java-args=-server -Xmx8G
  ```

Choosing an Application Server

DS servers run as standalone Java services, and do not depend on an application server.

The REST to LDAP and DSML gateway applications run on Apache Tomcat (Tomcat) and Jetty.

ForgeRock supports only stable application container releases. See the Tomcat and Jetty documentation for details about the right container to use with your Java environment.

Assigning FQDNs For Replication

Directory Services replication requires use of fully qualified domain names (FQDNs), such as `opendj.example.com`.

Host names like `my-laptop.local` are acceptable for evaluation. In production, and when using replication across systems, you must either ensure DNS is set up correctly to provide FQDNs, or update the hosts file (`/etc/hosts` or `C:\Windows\System32\drivers\etc\hosts`) to supply unique, FQDNs.

Synchronizing System Clocks For Replication

When using DS replication, keep server system clocks synchronized.

To keep the system clocks synchronized, use a tool that always moves the clock forwards. For example, `ntpd` adjusts the size of a second so that time always moves forwards to eventual clock consistency.

Never move the system clock *backwards*. Never use tools such as **ntpdate** that may move the clock backwards.

Getting Digital Certificates Signed

If you plan to configure SSL or TLS to secure network communications between the server and client applications, install a properly signed digital certificate that your client applications recognize, such as one that works with your organization's PKI or one signed by a recognized certificate authority.

To use the certificate during installation, the certificate must be located in a file-based keystore supported by the JVM (JKS, JCEKS, PKCS#12), or on a PKCS#11 token. To import a signed certificate into a keystore, use the Java **keytool** command.

For details, see "Preparing For Secure Communications" in the *Administration Guide*.

Special Requests

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

Chapter 3
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.

Important Changes to Existing Functionality

DS 6.5.6

- No changes in this release.

DS 6.5.5

- No changes in this release.

DS 6.5.4

- No changes in this release.

DS 6.5.3

- No changes in this release.

DS 6.5.2

- No changes in this release.

DS 6.5.1

- **Property `je-backend-shared-cache-enabled` Changes Default Value After Upgrade from 6.5.0 to 6.5.1**

  There is an issue when running an upgrade from DS 6.5.0 to 6.5.1. If you did not set the `je-backend-shared-cache-enabled` property and accepted the default value of `TRUE` prior to the upgrade, the value changes AFTER upgrade to `FALSE`. You may have to reset this value to `TRUE` for your deployments.

  If you set the `je-backend-shared-cache-enabled` property prior to upgrade to either `TRUE` or `FALSE`, the value does not change after upgrade.

DS 6.5.0

Take the following changes into account when upgrading to Directory Services 6.5. These changes will have an impact on existing deployments. Adjust existing scripts and clients accordingly:

- The **status** command has been rewritten, with the following notable changes:
  - The command is no longer interactive.

    You must supply the required options when invoking the **status** command.
  - The command now has an `--offline` option.

    When you run **status --offline** on a running server, the command only displays a portion of the available information.
  - You can now run the command against a remote DS server version 6 or later.
  - The output shows more information than before.
- The **dsreplication status** command no longer shows metrics for M.C. (missing changes) and A.O.M.C. (age of oldest missing change). Instead, it shows the replication delay.

  For DS 6 and later servers that expose a replication delay metric, the command shows the delay value. For DS 5.5 and earlier servers, the command shows `N/A`.
- The `db/admin` backend has been renamed `db/adminRoot`.
- The global server configuration property, `reject-unauthenticated-requests`, a boolean, has been removed and replaced with the property, `unauthenticated-requests-policy`.

  The new property can be set to the following values:

  `reject`

  Same behavior as `reject-unauthenticated-requests:true`

  `allow`

  Same behavior as `reject-unauthenticated-requests:false`

  `allow-discovery`

  Like `reject`, but allows unauthenticated base object searches of the root DSE
- The server-side (plugin) Java API is continuing to evolve, as noted in "*Release Levels and Interface Stability*".

  Server plugins written against this API will have to be adapted and recompiled to work with this version. For Java API reference documentation, see the Javadoc.

- The example server plugin described in "*Writing a Server Plugin*" in the *Developer's Guide* unpacks LDAP schema files into a `db/schema` directory. This is the correct schema file location for new DS servers since the 6.0 release.

  When you upgrade a server from version 5.5.x or earlier, however, schema files remain in the `config/schema` directory. As a result, when using the example plugin with the upgraded server, you must manually move schema files from the `db/schema` directory to the `config/schema` directory.

- The proxy backend configuration property `service-discovery-mechanism` has been renamed `shard`.

- The **encode-password** command now displays the encoded password without additional characters.

  In other words, the output is now `{scheme}encoded-password` rather than `Encoded Password: "{scheme}encoded-password"`.

Deprecated Functionality

This section lists deprecated functionality. Deprecation is defined in "ForgeRock Product Stability Labels".

DS 6.5.6

- No changes in this release.

DS 6.5.5

- No changes in this release.

DS 6.5.4

- No changes in this release.

DS 6.5.3

- Support for the Solaris operating system is deprecated and will be discontinued in a future release.

- The **backup**, **restore**, and **dsreplication** commands are deprecated in this release and will be replaced with equivalent functionality in a future release.

DS 6.5.2

The following monitoring metrics depending on the JVM implementation are not stable interfaces. They are deprecated and expected to be removed in the next major release:

Garbage collection statistics

Affected metrics have names like `ds-mon-jvm-garbage-collector-*` under `cn=monitor`, and `ds_jvm_garbage_collector_*` in Prometheus output.

Memory pool use

Affected metrics have names like `ds-mon-jvm-memory-pools-*` under `cn=monitor`, and `ds_jvm_memory_pools_*` in Prometheus output.

DS 6.5.1

- There are no new deprecated features in DS 6.5.1, other than those identified in DS 6.5.0.

DS 6.5.0

- All **dsreplication** subcommands are deprecated. Their names, meanings, and outputs are likely to change in a future release. This includes the following subcommands:
  - **dsreplication configure**
  - **dsreplication initialize**
  - **dsreplication initialize-all**
  - **dsreplication post-external-initialization**
  - **dsreplication pre-external-initialization**
  - **dsreplication purge-historical**
  - **dsreplication reset-change-number**
  - **dsreplication resume**
  - **dsreplication status**
  - **dsreplication suspend**
  - **dsreplication unconfigure**
- The **setup** command may be replaced by a simpler implementation in a future release.
- The HTTP monitoring endpoint, `/admin/monitor`, has been deprecated.

  Use `/metrics/api` or `/metrics/prometheus` instead.
- The following metrics are deprecated:
  - `ds-mon-approx-oldest-change-not-synchronized` (LDAP)

- `ds-mon-approximate-delay` (LDAP)
- `ds-mon-missing-changes` (LDAP)
- `ds_replication_changelog_connected_replicas_approx_oldest_change_not_synchronized_seconds` (Prometheus)
- `ds_replication_changelog_connected_replicas_approximate_delay_seconds` (Prometheus)
- `ds_replication_changelog_connected_replicas_missing_changes` (Prometheus)

Note

In mixed topologies, a directory server version 6 or earlier connected to a replication server version 6.5 or later cannot consume messages about other servers going offline. The monitoring framework reflects this as a delay on the directory server that could not consume the message.

The delay is reflected in the **dsreplication status** output and the values of the deprecated metrics described above.

The delay is calculated correctly again once all servers in the topology are upgraded to at least version 6.5, or when the offline the server comes back online and has seen a change to directory data.

Monitor replication delay instead of using the deprecated metrics. For details, see "Monitoring Replication Delay Over LDAP" in the *Administration Guide* or "Monitoring Replication Delay Over HTTP" in the *Administration Guide*.

Removed Functionality

DS 6.5.6

- No changes in this release.

DS 6.5.5

- No changes in this release.

DS 6.5.4

- There are no removed features in DS 6.5.4, other than those identified in DS 6.5.0.

DS 6.5.3

- There are no removed features in DS 6.5.3, other than those identified in DS 6.5.0.

DS 6.5.2

- There are no removed features in DS 6.5.2, other than those identified in DS 6.5.0.

DS 6.5.1

- There are no removed features in DS 6.5.1, other than those identified in DS 6.5.0.

DS 6.5.0

- The **manage-account get-password-history** subcommand has been removed due to security concerns.

Chapter 4

Fixes, Limitations, and Known Issues

This chapter covers the status of key issues and limitations for DS 6.5.

Key Fixes

The following important bugs were fixed in this release:

DS 6.5.6

- OPENDJ-8698: DS should write config archive files in a crash consistent way
- OPENDJ-8845: Bad encoding of PersistentSearch's changeType of the EntryChangeNotificationResponseControl
- OPENDJ-7970: Ensure that DS is crash resilient for all runtime file changes
- OPENDJ-7761: DS sporadically hangs while reconnecting to an RS
- OPENDJ-7653: replication issue in the cloud after ldapadd
- OPENDJ-6349: "RuntimeException: Should never happen" in HttpClientConnection

DS 6.5.5

- OPENDJ-5927: Server stuck on a DS trying to reconnect to an RS
- OPENDJ-6992: Persistent search from IDM is blocking worker threads.
- OPENDJ-7450: The startswith (sw) operator on indexed JSON attribute is slow
- OPENDJ-7481: JSON logs do not contain proxy auth DN
- OPENDJ-7655: Replaying multiple MODIFYDN operations is very slow
- OPENDJ-7699: Supportextract throws NoSuchElementException when the server.pid file is empty
- OPENDJ-7737: ConfigurationFramework#initialize0 changes the class loader without clearing the map of registered jar files
- OPENDJ-7818: Package based upgrade does not support instances running as non-root

- OPENDJ-7851: Supportextract tool: clobbers the server.out filehandle when kill -3 is used.
- OPENDJ-8028: Prometheus monitoring doesn't work with Telegraf
- CMON-109: Prometheus metrics contains more than one HELP metric line for the same metric

DS 6.5.4

- OPENDJ-4058: IDM Account Status notification handler doesn't look for certificates correctly
- OPENDJ-5439: LeastRequestsStrategy should distribute load randomly when idle
- OPENDJ-5851: ACI: getEffectiveRights with authz do not print out acl rights
- OPENDJ-6309: Search operation on whole tree skips nodes if there are DNs without backends in the directory information tree (DIT)
- OPENDJ-6377: Replication replay: issues with ReplaySynchronizer
- OPENDJ-6498: Profile creation stores AM cts and config global aci's in base64 format
- OPENDJ-6711: Replication status reports The provided value "5277383431" could not be parsed as an integer.
- OPENDJ-6812: Client tools fail in offline mode when Account Status Notification Handlers are used
- OPENDJ-6910: Supportextract --maxLogFiles gathers logs but not the latest logs
- OPENDJ-6970: Tamil locales cause illegal matchingRules values
- OPENDJ-6994: Strict-format-country-string does not affect the server
- OPENDJ-7014: Some operational attributes are not replicated when a restore --dry-run is used against an online server
- OPENDJ-7016: Status command outputs malformed JSON in script friendly mode
- OPENDJ-7020: Rebuild-index offline ignores rebuild-index.offline.java-args
- OPENDJ-7031: VLVIndex are incorrectly rebuilt by rebuild-index
- OPENDJ-7115: DS does not start when deployed with ISTIO side car container in the GCP K8s cloud
- OPENDJ-7176: Filters with malformed attribute descriptions cannot be parsed
- OPENDJ-7232: StackOverflowError in Tomcat logs when using external DS
- OPENDJ-7286: Changelog searches can start with incorrect cursors
- OPENDJ-7414: AM: Persistent search with changesOnly gets cancelled by a request timeout

DS 6.5.3

- OPENDJ-5600: The supportextract tool should capture stack traces with jcmd
- OPENDJ-5895: Unable to rebuild indexes when the Error Log Handler is assigned to a password policy
- OPENDJ-5960: The supportextract tool should gather basic changelogDb information
- OPENDJ-6163: The supportextract tool needs to gather archived-configs
- OPENDJ-6240: DS not honoring per user resource limits when processing RESTful operation requests
- OPENDJ-6371: The supportextract tool generates data but returns 1 instead of 0 on Windows
- OPENDJ-6394: Update forgerock-commons for 6.5.3
- OPENDJ-6422: Make the supportextract tool compliant with JVM unified logging framework
- OPENDJ-6464: IsMemberOfVirtualAttributeProvider does not process subordinate nested groups
- OPENDJ-6474: REST: some requests fails when stressing embedded http endpoint with Gatling
- OPENDJ-6512: Problems when work queue fills
- OPENDJ-6521: setup checks admin port despite options --skipPortCheck --doNotStart
- OPENDJ-6527: server does not return password policy responses with only warnings
- OPENDJ-6540: The supportextract tool hangs when loggers are configured to use /dev/stdout
- OPENDJ-6557: IDM Password Sync plugin induces 100% CPU in Apache Http Components when used with JDK 11
- OPENDJ-6675: The supportextract tool cannot collect gc files when there are dots in the path
- OPENDJ-6695: Heap slowly fills with DomainDBCursors
- OPENDJ-6708: The supportextract tool fails with an error parsing json
- OPENDJ-6733: SMTP handler sends incorrect email when account status is modified by manually updating ds-pwp-account-disabled attribute
- OPENDJ-6781: example-plugin fails to build on 6.5 branch
- OPENDJ-6778: Proxy server mishandles abandon requests
- OPENDJ-6787: Changelog searches are extremely slow if any cursors are exhausted
- OPENDJ-6820: dsconfig "-w -" option doesn't prompt for password

- OPENDJ-6822: Reduce number of expensive seeks in BlockLogReader
- OPENDJ-6830: The supportextract tool should capture stack traces in server.out with SIGQUIT
- OPENDJ-6892: Incorrect units for two updates metrics
- OPENDJ-6929: Support storing ads-certificate key-pair and other instance public keys in an HSM
- OPENDJ-6930: Increase interoperability with HSMs when protecting and distributing symmetric keys

DS 6.5.2

- OPENDJ-5972: bin/status command fails when using a french locale
- OPENDJ-6125: supportextract tool needs to gather the rootUser and monitorUser ldif files
- OPENDJ-6128: supportextract tool needs to gather Profile and Data Information
- OPENDJ-6170: supportextract tool misses rotated or non-standard GC log files
- OPENDJ-6173: cn=monitor memory pool stats do not get updated properly over time
- OPENDJ-6196: HTTP connection handler continues to listen to 0.0.0.0 after setting listen-address
- OPENDJ-6217: NPE when running supportextract tool on upgraded instance
- OPENDJ-6222: SMTP messages are sometimes not encoded with the correct charset
- OPENDJ-6248: NPE when running supportextract without monitoring user configured
- OPENDJ-6235: Stale ds-sync-hist attribute values reappear in the entry after replication is unconfigured

DS 6.5.1

- OPENDJ-5423: Incorrectly reported missing parent entries cause import-ldif and index rebuilds to fail
- OPENDJ-5584: Server does not validate sum of memory used by JE backend caches in all cases
- OPENDJ-5611: Change number indexing can lag behind replication under extreme load
- OPENDJ-5675: JDK11: supportextract tool cannot find jstack command
- OPENDJ-5726: Proxy distribution has trouble scaling writes to 3 shards
- OPENDJ-5727: Add optional base DN for each profile
- OPENDJ-5793: Replication on windows: ChangelogException while adding entries
- OPENDJ-5794: JE db-cache-size settings conflicts with shared cache

- OPENDJ-5801: ldap operation fails with "49 Invalid Credentials" when bindDN of 'cn=Directory Manager' is supplied in a properties file
- OPENDJ-5843: Rebuild-index failed with ConfigException on db-cache-size
- OPENDJ-5955: Missing version fallback feature for profiles
- OPENDJ-5977: Can not use custom base dn with cts profile because organization unit is forced
- OPENDJ-5979: Server does not validate sum of memory used by JE backend caches after upgrade
- OPENDJ-6039: AM Config Store Profile doesn't have enough access in ProductionMode when upgrading AM
- OPENDJ-6089: TelephoneNumber syntax in DN creates an incorrect entry DN value

DS 6.5.0

- OPENDJ-5406: Duplicate entry DNs if entry is deleted and then added during export-ldif or dsreplication initialize
- OPENDJ-5140: PersistentSearch heap usage grows
- OPENDJ-5553: Rest2Ldap cannot connect to TLSv1.2 servers
- OPENDJ-5496: DS fails to reconnect to an RS, disconnecting in handshake phase, after system restart
- OPENDJ-5594: StackOverflowError with groupOfURLs when isMemberOf is requested
- OPENDJ-4589: dsconfig --offline is not case-insensitive
- OPENDJ-4325: Changelog searches requesting changelogCookie are very slow
- OPENDJ-3341: REST to LDAP gateway: HTTP response for API description is empty
- OPENDJ-5210: Possible memory-leak if request received while bind in progress
- OPENDJ-3153: REST to LDAP gateway: changing password fails when using proxied authorization
- OPENDJ-5272: "idle-time-limit" global configuration property has no effect
- OPENDJ-5137: Reading compressed or encrypted entries fails to close the InflaterInputStream
- OPENDJ-5606: Upgrade to DS 6.0 fails if multiple filesystems are involved
- OPENDJ-4625: Changelog range searches miss entries
- OPENDJ-2356: verify-index displays an inappropriate error message when run in online mode
- OPENDJ-4229: status command with keystore options throws NullPointerException

- OPENDJ-4852: Backup with --backupAll misses a few backends
- OPENDJ-4881: Updates via Rest2ldap fail if record does not contain the necessary object class
- OPENDJ-5115: ldappasswordmodify fails, NPE in PasswordPolicyState updatePasswordHistory
- OPENDJ-4967: Rest2ldap UndeliverableException occurs when a referenced entity cannot be fetched
- OPENDJ-5558: SDK: LdapUrl is not IPv6 clean
- OPENDJ-5481: ERR_OPERATION_NOT_FOUND_IN_PENDING message used twice in different contexts
- OPENDJ-3343: Invalid Conflict resolution on Add sequence when Parent & Child are added on different replica
- OPENDJ-4947: SASL DIGEST-MD5: bind request failed with protocol error
- OPENDJ-5582: LdapClientSocket connection leaked when handshake fails
- OPENDJ-5293: Proxy: Replication Service Discovery Mechanism logs WARNING
- OPENDJ-3480: Updating schema backend properties while it is enabled leaves schema backend in broken state

Limitations

DS 6.5.6

- No changes in this release.

DS 6.5.5

- No changes in this release.

DS 6.5.4

- There are no limitations in DS 6.5.4, other than those identified in DS 6.5.0.

DS 6.5.3

- There are no limitations in DS 6.5.3, other than those identified in DS 6.5.0.

DS 6.5.2

- There are no limitations in DS 6.5.2, other than those identified in DS 6.5.0.

DS 6.5.1

- There are no limitations in DS 6.5.1, other than those identified in DS 6.5.0.

DS 6.5.0

DS 6.5.0 has the following limitations:

- Configuring a server with both local backends and proxy backends is not supported.

  As described in "*Configuring Privileges and Access Control*" in the *Administration Guide*, access control models for directory servers and proxy servers cannot function at the same time in the same server.

- DS servers provide full LDAP v3 support, except for alias dereferencing, and limited support for LDAPv2.

- Directory servers store passwords prefixed with the storage scheme in braces, as in `{scheme}`. For details, see "Configuring Password Storage" in the *Administration Guide*.

  To prevent users from effectively attempting to choose their own password storage scheme, directory servers do not support passwords that strictly match this format. Specifically, directory servers do not support passwords that match `{string}*`.

  Requests to update `userPassword` values with such passwords fail with result code 19 (Constraint Violation) and an additional message indicating that passwords may not be provided in pre-encoded form.

- When you configure account lockout as part of password policy, DS directory servers lock an account after the specified number of consecutive authentication failures. Account lockout is not transactional across a replication topology, however. Global account lockout occurs as soon as the authentication failure times have been replicated.

- When configuring replication between servers with different versions, use the appropriate **dsreplication** command:

  - When adding a new server to a replication topology with 2.6.x servers, use the **dsreplication** command installed with a 2.6 server.

  - When adding a new server to a replication topology with 3.x and later servers, use the **dsreplication** command installed with a new server.

  - After adding servers, use the **dsreplication** command installed with a new server.

- The policy-based access control handler used in proxy servers:

  - Does not support the Get Effective Rights control.

  - Does not check the `modify-acl` privilege when global access control policies are changed. The `config-write` privilege is sufficient to change global access control policies.
  - Does not send alert notifications when global access control policies change.
- When using ACIs or collective attributes with the proxy server data distribution feature, the ACI and entries having collective attribute values must be located at or above the `partition-base-dn`. When changing this data, make the change behind the proxy to one directory server replica in each shard. Your changes are not replicated outside the shard.

  The proxy server data distribution feature does not currently support the following:

  - Importing distributed data with the **import-ldif** command.
  - Changes to the number of partitions after data has been deployed.
  - Modify DN operations to distributed entries.
  - Updates to entries at or above the `partition-base-dn`.
  - Virtual static groups.
  - Data distribution does not support these virtual attributes:

    ```
    member
    uniqueMember
    ```

    The `isMemberOf` virtual attribute works as expected as long as you replicate the group entries on every shard.
  - Data distribution does not support these LDAP controls:

    Server-Side Sort controls: `1.2.840.113556.1.4.473`, `1.2.840.113556.1.4.474`
    Simple Paged Results control: `1.2.840.113556.1.4.319`
    Virtual List View controls: `2.16.840.1.113730.3.4.9`, `2.16.840.1.113730.3.4.10`
- The Password Policy control (OID: `1.3.6.1.4.1.42.2.27.8.5.1`) is supported for add, bind, and modify operations. It is not supported for compare, delete, search and modify DN operations.
- REST to LDAP does not support modify RDN operations.
- Prevent antivirus and intrusion detection systems from interfering with DS software.

  Before using DS software with antivirus or intrusion detection software, consider the following potential problems:

  Interference with normal file access

  Antivirus and intrusion detection systems that perform virus scanning, sweep scanning, or deep file inspection are not compatible with DS file access, particularly database file access.

Antivirus and intrusion detection software can interfere with the normal process of opening and closing database working files. They may incorrectly mark such files as suspect to infection due to normal database processing, which involves opening and closing files in line with the database's internal logic.

Prevent antivirus and intrusion detection systems from scanning database and changelog database files.

At minimum, configure antivirus software to whitelist the DS server database files. By default, exclude the following file system directories from virus scanning:

- `/path/to/opendj/changelogDb/` (if replication is enabled)

  Prevent the antivirus software from scanning these changelog database files.

- `/path/to/opendj/db/`

  Prevent the antivirus software from scanning database files, especially `*.jdb` files.

Port blocking

Antivirus and intrusion detection software can block ports that DS uses to provide directory services.

Make sure that your software does not block the ports that DS software uses. For details, see "Limiting System and Administrative Access" in the *Security Guide*.

Negative performance impact

Antivirus software consumes system resources, reducing resources available to other services including DS servers.

Running antivirus software can therefore have a significant negative impact on DS server performance. Make sure that you test and account for the performance impact of running antivirus software before deploying DS software on the same systems.

- REST to LDAP query filters do not work with properties of subtypes.

  For example, the default example configuration describes a user type, and a POSIX user type that inherits from the user type. If your query filter is based on a POSIX user type property that is not a property of the user type, such as `loginShell` or `gidNumber`, the filter always evaluates to false, and the query returns nothing.

- When applying a Common REST patch operation, described in "Patching Resources" in the *Developer's Guide*, to a `Json` syntax attribute, you cannot patch individual fields of the JSON object. You must change the entire JSON object instead.

  As a workaround, you can perform an update of the entire object, changing only the desired fields in your copy.

- When the global server property `invalid-attribute-syntax-behavior` is set to `accept` or `warn`, a search on group membership using a value with invalid syntax returns nothing.
- Due to a Java issue on Windows systems (JDK-8057894), when configuring DS directory servers with data confidentiality enabled you might see an error message containing the following text:

  ```
  Unexpected CryptoAPI failure generating seed
  ```

  If this happens, try running the command again.

Known Issues

Tip

When deploying DS servers in production, make sure that you follow the installation instructions. Allow DS servers to use at least 64K (65536) file descriptors. Also, tune the JVM appropriately.

- OPENDJ-8842: Proxy DS does not cancel psearch to Backend DS if psearch is cancelled
- OPENDJ-8460: Deploying DS6.5.5+JDK11 causes continuous hostname resolution errors in pods with DS6.5.5+JDK8
- OPENDJ-8137: LDIF backend silently rejects entries that fail schema validation
- OPENDJ-8089: rest2ldap gateway returns string instead of boolean
- OPENDJ-7942: The server ignores critical VLV request controls when falling back to an unindexed search
- OPENDJ-7919: A search for modifyTimestamp>=00000101000000Z results in a YEAR error and disconnect
- OPENDJ-7654: DS is sometimes unable to connect to RS after full gc
- OPENDJ-7643: Log that is supposedly generated from dsreplication operation is empty or does not exist
- OPENDJ-7640: Supportextract doesn't collect all security store info if both key manager and trust manager use the same store file
- OPENDJ-7516: External cn=changelog is not updated while replication initialization is in progress
- OPENDJ-7286: Changelog searches can start with incorrect cursors
- OPENDJ-7219: PreParseAddOperation cannot remove attributes
- OPENDJ-7197: Health Status returns OK however server is having JVM issues and cannot serve requests

- OPENDJ-6992: Persistent search from IDM is blocking worker threads.
- OPENDJ-6787: Changelog searches are extremely slow if any cursors are exhausted
- OPENDJ-6579: Schema is not populated to remote instances if added before enabling replication
- OPENDJ-6468: Some prometheus metrics are labeled as gauge, but seem to be counters
- OPENDJ-6465: same value for 'baseDn' profile property can be specified during setup
- OPENDJ-6380: Warning message for duplicate objectlass schema definition is misleading
- OPENDJ-6149: The Global Access Control Policy option within the dsconfig tool is misleading as is the error message returned
- OPENDJ-6022: PTA to Active Directory returns more than one entry when only one exists
- OPENDJ-5964: Replication: DS fails to fail-back to its local RS
- OPENDJ-5956: Data discrepancy between servers if the same attribute has extra spaces in RDN
- OPENDJ-5650: supportextract: Some Linux distros need htop instead of top -H
- OPENDJ-5201: Tools may prompt to trust certificate multiple times for different reasons

Chapter 5
Documentation Updates

Warning

Many examples in the documentation trust server certificates with the `--trustAll` option.

Examples using the `--trustAll` option are insecure except within a trusted network segment.

In production deployments, use appropriate trust options. For details, see the "Tools Reference" in the *Reference*.

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

Documentation Change Log

Date	Description
2022-08-03	Release of DS 6.5.6.
2021-11-10	Corrected a link in "Product Improvements".
2021-08-16	Initial release of DS 6.5.5. Added "Linux Page Caching" in the *Administration Guide* to explain how to avoid long pauses when the kernel flushes dirty pages to disk.
2021-06-29	• Added a note to "To Initialize All Servers From the Same LDIF" in the *Administration Guide* that clarifies when to follow the steps in "To Restore All Replicas to a Known State" in the *Administration Guide* instead. • Corrected "Data Replication and Data Sovereignty" in the *Deployment Guide* to indicate that the top entries in the example domain should *not* be replicated. • Updated "HTTP-Based Monitoring" in the *Administration Guide* and "LDAP-Based Monitoring" in the *Administration Guide* to clarify that the current replication delay metric is only meaningful once the server has received replication updates. • Updated "Rebuilding Indexes" in the *Administration Guide* to note that, when a server rebuilds an index online, the data in the affected database backend is temporarily unavailable to client applications.
2020-11-24	Updated "Troubleshooting TLS/SSL Connections" in the *Administration Guide* and "How Certificates are Used" in the *Security Guide* to add an example command that demonstrates how to list trusted certificates with the Java 8 **keytool** command.

Date	Description
2020-11-06	Fixed the list of known issues for 6.5.0.
2020-09-23	• Initial release of DS 6.5.4. The following documentation updates were made: • Updated "*Monitoring Metrics*" in the *Reference* to add notes that some JVM metrics, particularly GC-related metrics, depend on the JVM version and configuration. • Updated "Fulfilling Storage Requirements" to highlight that DS servers do not support NFS for directory data. • Fixed the port number for the first server in "Trying Replication" in the *Getting Started*.
2020-02-27	Initial release of DS 6.5.3. The following documentation changes occurred: • Updated "To Enable the External Change Log" in the *Administration Guide* to clarify which configuration is required, and to cover standalone directory servers. The separate procedure, *To Enable the External Change Log (Standalone Server)*, has been removed. • Added an explanation for Crypto Manager support for PKCS#11 modules. See DS 6.5.3. • Changed **dsreplication enable** to **dsreplication configure** in section 8 of the Admin Guide. • Removed obsolete options in **ldifdiff** examples. See "Examples" in the *Reference*.
2020-01-22	Added OPENDJ-5423 to the Key Fixes list for OpenDJ 6.5.1. See DS 6.5.1.
2019-12-20	• Updated "*Changing Server Certificates*" in the *Administration Guide* to clarify that, when using a keystore to hold DS server keys, the password for the keystore and any private keys it contains must be the same. • Updated "To Allow a User to Read the Change Log" in the *Administration Guide* to include access to changelog attributes on the root DSE entry. The IDM liveSync feature requires access to these attributes. • Updated "LDAP Metrics by Name" in the *Reference* to fix the types of the `ds-mon-updates-inbound-queue` and `ds-mon-updates-outbound-queue` metrics, which are integers.
2019-12-20	• Added "Replication Network Use and Operations" in the *Administration Guide* to describe how replication uses DS server ports that must remain open to remote clients. • Updated "Troubleshooting LDIF Import" in the *Administration Guide* to reference the correct configuration property for relaxing syntax checking.

Date	Description
2019-11-12	• Updated "Running in a Container" in the *Deployment Guide* as the section is no longer relevant to Java 8 update 191 and later releases. • Updated "Preventing Interference With Antivirus Software" in the *Installation Guide* to clarify how to prevent interference. • Added "To Disable Change Number Indexing" in the *Administration Guide* to explain how to disable change number indexing when not needed. For example, disable change number indexing when using DS as a CTS store for AM. • Updated "Metric Types" in the *Reference* to clarify the definitions of monitoring metrics.
2019-09-17	• Updated "Deprecated Functionality" to indicate deprecated JVM monitoring metrics. • Added "Upgrading System and Server Tuning" in the *Installation Guide* to underline the importance of revisiting tuning settings during major version upgrades. • Updated "To Set Up a Directory Proxy Server" in the *Installation Guide* to clarify the requirements when setting up a directory proxy server in production mode. • Added "*On Using a Load Balancer*" in the *Administration Guide* with recommendations for your deployment. • Updated "Resetting the Directory Manager's Password" in the *Administration Guide* to indicate the default file that holds the directory superuser entry. • Added "To Transform a Directory Server/Replication Server Into a Standalone Directory Server" in the *Administration Guide*. • Updated "Choosing the Listen Address for Replication" in the *Administration Guide* to fix a broken link.
2019-06-20	Initial release of DS 6.5.2. • Updated "Java Settings" in the *Administration Guide* to remove mention of the `-XX:+UseAES -XX:+UseAESIntrinsics` options. Recent JVM versions enable these options automatically. If you enable them manually, you may also need the `-XX:+UnlockDiagnosticVMOptions` option. • Updated "*Managing Data Replication*" in the *Administration Guide* to clarify that you should not run **dsreplication configure** and **dsreplication initialize** commands at the same time, nor should you run multiple **dsreplication configure** commands at the same time.
2019-04-10	Initial release of DS 6.5.1. • Updated the documentations that all **setup** command profiles now allow you to set the domain or the base DN. For details, see "*Using Directory Server Setup Profiles*" in the *Installation Guide*.

Date	Description
2018-11-28	Initial release of DS 6.5.0. In addition to the changes described in "*What's New*" and "*Compatibility*", the following important changes were made to the documentation: • Added a new guide, *Getting Started*, that provides a quick, hands-on looks at what Directory Services software can do. • Added a new chapter, "*Configuring REST APIs*" in the *Administration Guide*, focused on administrative work to build REST APIs. The chapter includes a new section, "Mapping JSON Profiles to LDAP" in the *Administration Guide*, as an example for administrators creating new REST APIs starting from JSON resources rather than LDAP entries. • Added a new chapter, "*Deploying for DevOps and SaaS*" in the *Deployment Guide*, that focuses on use of DS software in DevOps and SaaS deployments. • Added a section, "Setting Disk Space Thresholds For Replication Changelog Databases" in the *Administration Guide*, showing how to set low and full disk thresholds for replication server changelog databases. • Updated "Limitations" and "To Add a New Replica to an Existing Topology" in the *Installation Guide* to reflect the need to use the **dsreplication** command installed with a 2.6 server when adding a new server to a replication topology with 2.6.x servers. • Rewrote the section, "Choosing Load Balancing Settings" in the *Administration Guide*, to clarify how to choose among the alternatives offered by directory proxy servers. • "*Attribute Types*" in the *LDAP Schema Reference* now includes a *Used By* list in each table of attribute properties. The list consists of links to sections describing the object classes that require or allow the the attribute. • A new section shows how to back up and restore configuration files. See "Backing Up and Restoring Configuration Files" in the *Administration Guide*. • Added a step to "To Upgrade Replicated Servers" in the *Installation Guide* showing how to add missing privileges to the global administrator account. These privileges are required when using the **dsreplication status** command. • Added an example of the proxy user and ACI required on DS directory servers. See "To Set Up a Directory Proxy Server" in the *Installation Guide*, or "Configuring a Proxy Backend" in the *Administration Guide*. • Corrected the synopsis for `targattrfilters` in "ACI Targets" in the *Administration Guide*.

Date	Description
	The documentation incorrectly suggested `(targattrfilters != "expression")` as a legal ACI target. In an ACI target, `targattrfilters` must be set equal to an expression, as in `(targattrfilters = "expression")`. The documentation also incorrectly indicated that you separate `targattrfilters` expressions with semicolons. Instead, use commas to separate multiple `targattrfilters` expressions. • Clarified in "Updating Resources" in the *Developer's Guide* that an update operation requires a JSON payload including *all* the writable fields of the resource that you want to retain. The update replaces the writable fields of the resource with the values in your JSON payload. If you want to change only part of a JSON resource, see "Patching Resources" in the *Developer's Guide* instead. • Updated "To Set Up JMX Access" in the *Administration Guide* to explain how to avoid periodic full garbage collection events when using JMX.

Appendix A. Release Levels and Interface Stability

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

Some interfaces are labelled as Evolving in the body of the documentation. In addition, the following rules apply:

- All Java APIs are Evolving, except `com.*` packages, which are Internal/Undocumented.
- The class `org.forgerock.opendj.ldap.CoreMessages` is Internal.
- Text in log messages should be considered Internal. Log message IDs are Evolving.
- Monitoring metrics available over LDAP (`cn=monitor`), HTTP, and JMX are Evolving.
- The default content of `cn=schema` (LDAP schema) is Evolving.
- The interface of the "*changelogstat — debug changelog and changenumber files*" command is Evolving.
- Newly Deprecated and Removed interfaces are identified in "*Compatibility*".
- Interfaces that are not described in released product documentation should be considered Internal/Undocumented. For example, the LDIF representation of the server configuration, `config.ldif`, should be considered Internal.

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 • 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

ForgeRock Product Stability Labels

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

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

ForgeRock Stability Label Definitions

Stability Label	Definition
Stable	This documented feature or 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 feature or 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.
Legacy	This feature or interface has been replaced with an improved version, and is no longer receiving development effort from ForgeRock. You should migrate to the newer version, however the existing functionality will remain. Legacy features or interfaces will be marked as *Deprecated* if they are scheduled to be removed from the product.
Deprecated	This feature or interface is deprecated and likely to be removed in a future release. For previously stable features or interfaces, the change was likely announced in a previous release. Deprecated features or interfaces will be removed from ForgeRock products.
Removed	This feature or 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 considered as new technology that is not yet supported. Technology preview features may be functionally incomplete and the function as implemented is subject to 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 features or interfaces can change without notice. If you depend on one of these features or interfaces, contact ForgeRock support or email info@forgerock.com to discuss your needs.

Appendix B. Getting Support

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.

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.