OpenAM Release Notes

Version 13.5

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

Abstract

Notes covering OpenAM prerequisites, fixes, known issues. OpenAM provides open source authentication, authorization, entitlement, and federation software.

Table of Contents

Chapter 1

What's New in OpenAM 13.5

Before you install OpenAM or update your existing OpenAM installation, read these release notes. Then update or install OpenAM.

1.1 Major New Features in OpenAM 13.5.0

OpenAM 13.5.0 is a minor release that introduces new features, functional enhancements, and fixes to OpenAM.

This release introduces the following product enhancements:

- Smarter Security Features
- OAuth v2.0/OpenID Connect 1.0 Enhancements
- Performance Enhancements
- User Experience Enhancements
- Dev Ops Features

1.1.1 Smarter Security Features

- **Push Authentication for Passwordless Login and Easy Multi-Factor**

  OpenAM 13.5 introduces a new authentication option that uses push notifications alongside an updated ForgeRock Authenticator app.

The ForgeRock Authenticator app can now respond to push notifications for multi-factor authentication, including a passwordless login mechanism.

For more information, see Section 2.9.1.2, "About Push Authentication" in the *OpenAM Administration Guide*.

- **New Elliptic Curve Digital Signature Algorithms**. OpenAM 13.5 introduces Elliptic Curve Digital Signature Algorithm (ECDSA) support for signing OpenID Connect `id_tokens` and stateless session tokens.

  The following ECDSA algorithms are now supported:

  - ES256 - ECDSA using SHA-256 hashes and the NIST standard P-256 elliptic curve.
  - ES384 - ECDSA using SHA-384 hashes and NIST standard P-384 curve.
  - ES512 - ECDSA using SHA-512 hashes and NIST standard P-521 curve.

  You can generate public and private keys for these algorithms using **keytool**.

  For more information, see Section 9.8.3, "Configuring Elliptic Curve Digital Signature Algorithms" in the *OpenAM Administration Guide*.

- **Default JCEKS Keystore**. OpenAM now uses a JCEKS keystore as its default keystore. User self service requires two key aliases: one for signing and one for encryption. OpenAM's JCEKS keystore comes with two default test aliases that can be used for out-of-the-box configuration that should be used for demo purposes only.

  For upgrades to OpenAM 13.5, the keystore remains the same as previously configured. For example, if you had a JKS keystore configured, the keystore configuration remains as JKS after upgrade to OpenAM 13.5. You will need to reconfigure the keystore to JCEKS on OpenAM 13.5.

  For more information, see Section 8.3.1, "Configuring the Signing and Encryption Key Aliases" in the *OpenAM Administration Guide*.

- **New Trust Transaction Header System Property**. OpenAM supports the propagation of the transaction ID across the ForgeRock platform, such as from OpenDJ or OpenIDM to OpenAM, using the HTTP header `X-ForgeRock-TransactionId`.

  You can set a new property `org.forgerock.http.TrustTransactionHeader` to `true`, which will trust any incoming `X-ForgeRock-TransactionId` headers. By default, the `org.forgerock.http.TrustTransactionHeader` is set to `false`, so that a malicious actor cannot flood the system with requests using the same transaction ID header to hide their tracks.

For more information, see Section 6.6, "Configuring the Trust Transaction Header System Property" in the *OpenAM Administration Guide*.

1.1.2 OAuth v2.0/OpenID Connect 1.0 Enhancements

OpenAM 13.5 introduces a number of enhancements for its OAuth 2.0 and OpenID Connect 1.0 components:

- **New Stateless `idtokeninfo` Endpoint for OIDC Token Validation**

  OpenAM 13.5 now supports a new `/oauth2/idtokeninfo` endpoint for OpenID Connect 1.0 (OIDC) `id_token` validation. The feature allows clients to offload validation of an OIDC token to the endpoint and to retrieve the claims contained within the `id_token` directly without accessing the datastore.

  For more information, see Section 2.1.3.2.5, "Endpoint for Validating OpenID Connect 1.0 ID Tokens" in the *OpenAM Developer's Guide*.

- **OAuth v2.0 Stateless Token Blacklisting**

  OpenAM 13.5 now supports stateless OAuth v2.0 token blacklisting.

  For more information, see Section 13.6, "Configuring Stateless OAuth 2.0 Token Blacklisting" in the *OpenAM Administration Guide*.

- **Stateless OAuth 2.0 Access and Refresh Tokens**

  OpenAM 13.5 now supports stateless OAuth 2.0 access and refresh tokens that can be quickly validated

  For more information, see Section 14.9, "Stateless OpenID Connect 1.0 Access and Refresh Tokens" in the *OpenAM Administration Guide*.

- **New Field in OAuth2/OIDC Agent Settings: `com.forgerock.openam.oauth2provider.jwks`.**

  OpenAM 13.5 has been updated with a new field in the OAuth v2.0/OIDC agent setting to specify a static JWKSet value: `com.forgerock.openam.oauth2provider.jwks`.

  This setting will allow the Public Key Selector to accept `JWKS` rather than `JWKs_URL`.

  For more information, see Section 5.8, "Configuring OAuth 2.0 and OpenID Connect 1.0 Clients" in the *OpenAM Administration Guide*.

- **OpenID Connect ID Token Encryption**

OpenAM 13.5 now supports the ability to encrypt OIDC ID tokens. Administrators can now enable encryption in the OpenAM console.

For more information, see Section 14.11, "Encrypting OpenID Connect ID Tokens" in the *OpenAM Administration Guide*.

- **Expose OAuth v2.0 Access/Refresh Token Signing Public Key via HTTP**

  OpenAM 13.5 supports the ability for an application to obtain the public key used to digitally sign the access and refresh tokens from OpenAM via HTTP.

  Applications can use the `/oauth2/connect/jwk_uri` endpoint to obtain the public key to sign the access and refresh tokens from OpenAM. The application can then validate an OAuth v2.0 stateless token without contacting OpenAM.

  For more information, see Procedure 13.9, "To Obtain the OAuth 2.0/OpenID Connect 1.0 Public Signing Key" in the *OpenAM Administration Guide*.

- **OAuth 2.0 User Consent Page Can Be Optional**

  OpenAM 13.5 can now make the OAuth 2.0 user consent page optional. You can set this up by configuring two new settings:

  - On the OAuth2 Provider settings, enable the `Allow clients to skip consent` option.
  - On the OAuth 2.0 Client (agent) settings, enable `Implied consent`.

  When both settings are configured, OpenAM treats the requests as if the client has already saved consent and will suppress any user consent pages to the client.

  For more information, see Section 13.4.4, "Allowing Clients To Skip Consent" in the *OpenAM Administration Guide*.

1.1.3 Performance Enhancements

OpenAM 13.5 has made a number of fixes that improves OpenAM's performance. Some of the fixes are as follows:

- **Option to Generate or Disable Sign Out Tokens**

  OpenAM 13.5 now provides a `Store Ops Token` option to generate and store a sign out token in the CTS store for an OIDC provider.

  When the property is disabled, OAuth 2.0 performance may be improved by not storing the sign out tokens in the CTS store.

For more information, see Section 1.4.11, "OAuth2 Provider" in the *OpenAM Reference*.

- **Same Call to /oauth2/authorize and /oauth2/access_token Optimized**

  OpenAM 13.5 has optimized the sequence to generate an /oauth2/OIDC token (for example, SSO login, oauth2/authorize, access_token with authz code).

- **CTS Uses Replace Instead Of Delete/Add for Single Valued Attributes**

  OpenAM 13.5 now uses LDIF `replace` instead of a `delete/add` combination for single valued attributes in OpenDJ. This feature improves performance, requiring less processing in OpenDJ.

1.1.4 User Experience Enhancements

- **Continued Migration of JATO Objects to XUI**

  In OpenAM 13.5, the Services and Global configuration screens in the OpenAM console have migrated to the new XUI interface.

1.1.5 Dev Ops Features

- **New Policy Export/Import ssoadm Command**

  OpenAM 13.5 has enhanced its **ssoadm** command to support policy export and import to JSON.

  For more information, see Procedure 3.6, "To Export Policies in JSON Format (Command Line)" in the *OpenAM Administration Guide*.

1.1.6 Platform Enhancements

- **New Elasticsearch and JMS Audit Event Handlers**

  Enhancements to the ForgeRock Common Audit Framework allow OpenAM 13.5 to log user and administrative activity to Elasticsearch and JMS.

  For more information about the Elasticsearch audit event handler, see Section 6.4.4, "Implementing Elasticsearch Audit Event Handlers" in the *OpenAM Administration Guide*.

  For more information about the JMS audit event handler, see Section 6.4.5, "Configuring JMS Audit Event Handlers" in the *OpenAM Administration Guide*.

1.2 Improvements in OpenAM 13.5.0

The following improvements and additional features were added in this release:

- OPENAM-5093: OAuth2 user consent confirmation can be optional
- OPENAM-5131: FederationConfig.properties in unconfigured Fedlet should have com.sun.identity.common.serverMode=false by default
- OPENAM-5213: OAuth2 tokeninfo endpoint is not returning client_id info
- OPENAM-5938: Cert Auth module should not read cert from HTTP request when 'iplanet-am-auth-cert-gw-cert-auth-enabled' is set
- OPENAM-6315: Proxying SAML2 Second level status code
- OPENAM-7146: Revoke access tokens while revoking refresh tokens
- OPENAM-7294: Support for WS-Federation active requestor profile
- OPENAM-7320: Consider using JDK JAXP/XML instead of Xerces/Xalan to keep up with JDK fixes
- OPENAM-7702: Give the ability to disable creation of sign out tokens
- OPENAM-7778: XML Signature DigestMethod should be configurable when using SAML2
- OPENAM-7820: Additional delete/revoke token endpoints for Oauth2
- OPENAM-7914: Make the attribute com.sun.identity.server.fqdnMap hot-swappable
- OPENAM-7996: Self-registration destination after registration
- OPENAM-8194: The default WS-Fed IDP attribute mapper should provide a way to Base64 encode binary attributes
- OPENAM-8387: OpenAM should provide more detailed log messages when KeyUtil.getDecryptionKey does not find the requested key
- OPENAM-8423: Introduce "audience URL" attribute in OAuth2 client for Saml2GrantTypeHandler
- OPENAM-8578: The default WS-Fed and SAML2 IDP attribute mapper should provide a way to Base64 binary encoding of NameID
- OPENAM-8580: OpenAM should allow to use objectGUID value from AD when working with persistent NameID

- OPENAM-8932: WS-Federation should support attribute mapping with custom namespaces
- OPENAM-9124: FaceBook authentication module & documentation should be updated to reflect changes to FaceBook API
- OPENAM-9279: User registration should return authn success addition properties inline with the authn endpoint

1.3 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 more information on ForgeRock's security advisory policy, click the following link: http://www.forgerock.com/services/security-policy/

Chapter 2

Before You Install OpenAM Software

This chapter covers software and hardware prerequisites for installing and running OpenAM server software.

ForgeRock supports customers using the versions specified here. Other versions and alternative environments might work as well. When opening a support ticket for an issue, however, make sure you can also reproduce the problem on a combination covered here.

2.1 OpenAM Operating System Requirements

ForgeRock supports customers using OpenAM server software on the following operating system versions:

Table 2.1. Supported Operating Systems

Operating System	Version
Red Hat Enterprise Linux, Centos	6, 7
SuSE	11
Ubuntu	12.04 LTS, 14.04 LTS
Solaris x64	10, 11
Solaris Sparc	10, 11

Operating System	Version
Windows Server	2008, 2008 R2, 2012, 2012 R2

2.2 Java Requirements

Table 2.2. JDK Requirements

Vendor	Version
Oracle JDK	7, 8
IBM SDK, Java Technology Edition (Websphere only)	7

2.3 OpenAM Web Application Container Requirements

Table 2.3. Web Containers

Web Container	Version
Apache Tomcat	7, 8
Oracle WebLogic Server	12c
JBoss Enterprise Application Platform	6.1+
JBoss Application Server	7.2+
WildFly AS	9
IBM WebSphere	8.0, 8.5.5.8+

The web application container must be able to write to its own home directory, where OpenAM stores configuration files.

2.4 Data Store Requirements

Table 2.4. Supported Data Stores

Data Store	Version	CTS Datastore	Config Datastore	User Datastore	UMA Datastore
Embedded OpenDJ	3.5	✓	✓	✓	✓
External OpenDJ	2.6, 2.6.4, 3.0, 3.5	✓	✓	✓	✓
Oracle Unified Directory	11g			✓	
Oracle Directory Server Enterprise Edition	11g			✓	
Microsoft Active Directory	2008, 2008 R2, 2012, 2012 R2			✓	
IBM Tivoli Directory Server	6.3			✓	

2.5 Supported Clients

The following table summarizes supported clients:

Table 2.5. Supported Clients

Client Platform	Native Apps[a]	Chrome 16+[b]	IE 9+, Microsoft Edge	Firefox 3.6+	Safari 5+
Windows 7 or later	✓	✓	✓	✓	✓
Mac OS X 10.8 or later	✓	✓		✓	
Ubuntu 12.04 LTS or later	✓	✓		✓	✓
iOS 7 or later	✓	✓			✓
Android 4.3 or later	✓	✓			

[a] *Native Apps* is a placeholder to indicate OpenAM is not just a browser-based technology product. An example of a native app would be something written to use our REST APIs, such as the sample OAuth 2.0 Token Demo app.

[b] Chrome, Firefox, and Safari are configured to update automatically, so customers will typically running the latest versions. The versions listed in the table are the minimum required versions.

2.6 Supported Upgrade Paths

The following table contains information about the supported upgrade paths to OpenAM 13.5:

Table 2.6. Upgrade Paths

Version	Upgrade Supported?
OpenAM 9.0.x	No
OpenAM 9.5.x	No
OpenAM 10.0.x	No
OpenAM 11.0.x	Yes

Version	Upgrade Supported?
OpenAM 12.0.x	Yes
OpenAM 13.0.x	Yes

Note

Upgrading between OpenAM Enterprise and OpenAM OEM versions is not supported.

For more information, see Checking your product versions are supported in the *ForgeRock Knowledge Base*.

2.7 Special Requests

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

Chapter 3

Installing or Upgrading

This chapter covers installing and upgrading OpenAM 13.5 software.

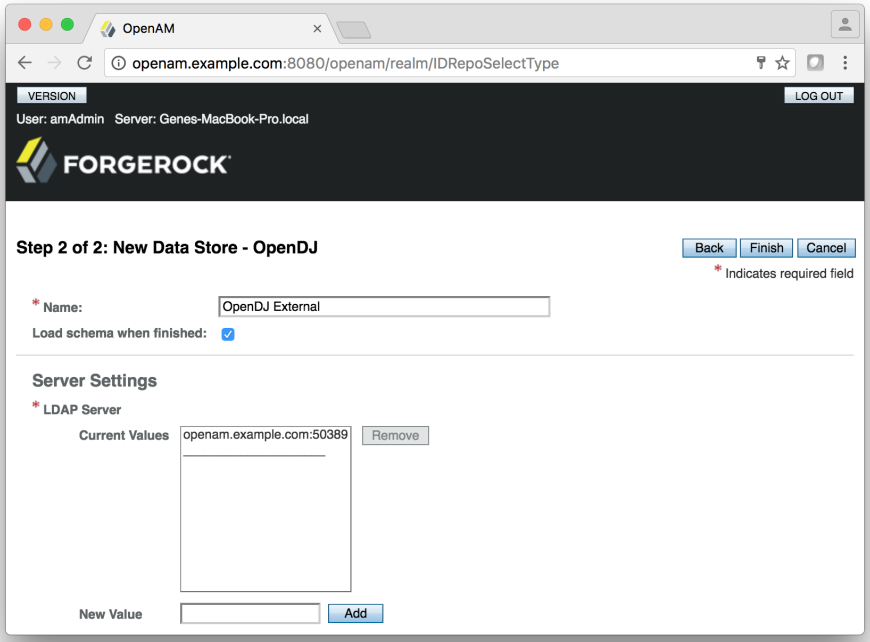
Note

Do *not* perform an upgrade by deploying the new version and then importing an existing configuration by running the **ssoadm import-svc-config** command. Importing an outdated configuration can result in a corrupted installation.

Before you install OpenAM or upgrade your existing OpenAM installation, read these release notes. Then, install or upgrade OpenAM.

- If you are installing OpenAM for the first time, see the OpenAM Installation Guide.
- If you are upgrading from OpenAM 13.0 to OpenAM 13.5 and want to configure Push Authentication with your external data store, you must manually apply the schema update to the data store by enabling the `load schema when finished` in the OpenAM console. You can set the property on the OpenAM console by navigating to Top Level Realm > Data Stores > New > data store type.

Figure 3.1. Load Schema When Finished



- In a clean OpenAM 13.5 installation, you must manually update the schema using the `opendj_pushdevices.ldif` if you want to use Push Notifications. The `opendj_pushdevices.ldif` is located in `/path/to/tomcat/webapps/openam/WEB-INF/template/ldif/opendj` [1] folder. To manually update the schema, see Updating Directory Schema in the *OpenDJ Administration Guide* for instructions.

For additional information about upgrading OpenAM, see the OpenAM Upgrade Guide.

[1] There are analogous `pushdevices.ldif` files for Active Directory in the `/path/to/tomcat/webapps/openam/WEB-INF/template/ldif/ad` folder; Oracle DSEE in the `/path/to/tomcat/webapps/openam/WEB-INF/template/ldif/odsee` folder; Tivoli in the `/path/to/tomcat/webapps/openam/WEB-INF/template/ldif/tivoli` folder. For instructions to update the schema, see the respective directory server documentation.

Chapter 4

Changes and Deprecated Functionality

This chapter covers both major changes to existing functionality, and also deprecated and removed functionality.

4.1 Important Changes to Existing Functionality

The following changes are new in OpenAM 13.5:

- **Cookie Domain Defaults to FQDN**. When adding another server to an existing OpenAM 13.5.0 deployment using the GUI configurator, the cookie domain in the OpenAM setup wizard sets the cookie domain to be the full URL that was used to access the configurator, such as `example.com`.

  For more information, see OPENAM-9369.

- **Some CTS OIDs Use the Custom `Float2dp` Data Type**. The following CTS OIDs now use the new, custom `Float2dp` data type:

  - enterprises.36733.1.2.3.3.1.2.*
  - enterprises.36733.1.2.3.3.1.6.*
  - enterprises.36733.1.2.3.4.1.2.*.*
  - enterprises.36733.1.2.3.6.0
  - enterprises.36733.1.2.3.7.1.2.0
  - enterprises.36733.1.2.3.7.2.2.0

The `Float2dp` data type is a floating point number with the value `d-2` in the `DISPLAY-HINT` clause. SNMP clients that handle the `DISPLAY-HINT` clause will correctly display the value as a floating point number with two decimal places. Other types of clients that do not handle the `DISPLAY-HINT` clause will incorrectly display the value as an integer that is one hundred times larger than the correct value.

All other CTS OIDs use the `Counter64` data type, a standard data type returned by SNMP OIDs.

For more information, see Chapter 8, "Core Token Service (CTS) Object Identifiers" in the *OpenAM Reference*.

- **SAML 2.0 NameID Persistence Extended**. OpenAM's SAML 2.0 account management and NameID persistence logic was updated to work better with non-persistent NameID formats in OpenAM 13.0.0. It has now been extended to have persistence completely controlled by the hosted IdP flag (`idpDisableNameIDPersistence`) and the hosted/remote SP flag (`spDoNotWriteFederationInfo`). These flags now also control whether the persistent NameID details should be stored in the datastore.

  This change allows deployments that have a read-only user store shared by the SP and IdP to not store persistent federation information.

  The NameID persistence logic can now summarized as follows:

  ```
  Persistent NameID         -> NameID RECOMMENDED be stored
  Transient NameID          -> NameID MUST NOT be stored
  Ignored user profile mode -> NameID CANNOT be stored (fails if used in
                               combination with persistent NameID-Format)
  For any other case        -> NameID MAY be stored based on customizable logic
  ```

  The following changes were made on the identity provider side:

  - **Setting: idpDisableNameIDPersistence**. OpenAM provides a setting, `idpDisableNameIDPersistence`, which disables the storage of the NameID values for all NameIDs issued by that IdP instance.

  - **SP's spDoNotWriteFederationInfo Repurposed**. The SP's `spDoNotWriteFederationInfo` setting has been repurposed. It no longer applies to unspecified NameID-Formats and now allows persistence to be set to NOT store federation info.

- **NameID Lookup Changes**. The NameID lookup mechanism has been modified, so that it only tries to look up existing NameID values for the user if the NameID is actually persisted for the corresponding NameID-Format.

- **Method in the IDPAccountMapper Interface**. The `IDPAccountMapper` interface has been extended with a new `shouldPersistNameIDFormat` method.

  The default implementation of `shouldPersistNameIDFormat` in `DefaultIDPAccountMapper` first checks whether `idpDisableNameIDPersistence` is enabled in the hosted IdP configuration. If `idpDisableNameIDPersistence` is disabled, the logic advances and accesses the remote SP's `spDoNotWriteFederationInfo` flag.

  For more information, see shouldPersistNameIDFormat in the *OpenAM API Javadoc*.

The following changes have been made on the service provider side:

- **Changes to SPAccountMapper**. The `SPAccountMapper` implementations now no longer need to perform reverse lookups using the received NameID value. The `SPACSUtils` now performs the reverse lookup if the NameID-Format should be persisted. This change was made to ensure that NameID values are only persisted in the data store if they have not been stored there previously.

- **SP's spDoNotWriteFederationInfo Repurposed**. The SP's `spDoNotWriteFederationInfo` setting has been repurposed. It no longer is specific to unspecified NameID-Formats. It affects all non-persistent NameID-Formats.

- **Method in the SPAccountMapper Interface**. The `SPAccountMapper` interface has been extended with the following new method:

```
/**
* Tells whether the provided NameID-Format should be persisted in the user data
*       store or not.
*
* @param realm The hosted SP's realm.
* @param hostEntityID The hosted SP's entityID.
* @param remoteEntityID The remote IdP's entityID.
* @param nameIDFormat The non-transient, non-persistent NameID-Format in question.
* @return true if the provided NameID-Format should be persisted
* in the user data store, false otherwise.
**/
public boolean shouldPersistNameIDFormat(String realm, String hostEntityID,
       String remoteEntityID, String nameIDFormat);
```

  This implementation first checks whether NameID persistence has been completely disabled at the IdP level (idpDisableNameIDPersistence

setting), and if not, it will look at the SP configuration as well (spDoNotWriteFederationInfo setting).

For more information, see OPENAM-8580.

These changes are new in OpenAM 13.0.0:

- **New Attribute Required in Authentication Service Definition**. OpenAM 13 requires that schemas in the definition of an authentication service contain `resourceName` attributes.

  The attributes are not added to custom authentication service definitions when upgrading from a previous version, so must be added manually.

  The specific changes required in the service definition schema are:

  - The `Schema` element in the service definition must contain a `resourceName` attribute. This value is used to refer to the service when managing the service using REST.

    For example:

    ```
    <Schema
     serviceHierarchy="/DSAMEConfig/authentication/iPlanetAMAuthSampleAuthService"
     i18nFileName="amAuthSampleAuth"
     revisionNumber="10"
     i18nKey="sampleauth-service-description"
     resourceName="mySampleAuthService">
    ```

  - Any `SubSchema` elements in the service definition must contain a `resourceName` attribute, with a value of `USE-PARENT`.

    For example:

    ```
    <SubSchema
     name="serverconfig"
     inheritance="multiple"
     resourceName="USE-PARENT">
    ```

  An example of a service definition compatible with OpenAM 13 can be found in Section 3.3.6, "The Sample Auth Service Configuration" in the *OpenAM Developer's Guide*.

Procedure 4.1. To Add Required Attributes to Custom Service Definition Schemas

You can add the required attributes either before or after upgrading to OpenAM 13. The steps in this procedure cover adding the attributes before upgrading.

1. If you have not already done so, install and configure a tool for altering the contents of the OpenDJ configuration store, for example the OpenDJ Control Panel or Apache Directory Studio.

2. Connect to the embedded configuration store using the same bind DN credentials as configured in OpenAM. The default is `cn=Directory Manager`.

3. In the directory tree of the configuration store, locate the `sunServiceSchema` attribute for your custom service definition under `ou=services`.

   For example, on a default install the definition for the data store service is located here: `ou=1.0,ou=sunAMAuthDataStoreService,ou=services,dc=openam,dc=forgerock,dc=org`

4. Edit the XML stored within the `sunServiceSchema` attribute, adding the required `resourceName` attribute to `Schema` and `SubSchema` elements.

5. Commit the changes to the configuration store, and proceed to upgrade OpenAM.

Failure to add the required attributes will result in the OpenAM 13 user interface being unable to view or edit custom services, or create or edit authentication modules based on them after upgrade. You may also see a `Not found error` message displayed in the administration console when creating or editing authentication modules.

- **AD/LDAP/RADIUS Authentication Modules Allow More Than One Primary/Secondary Server**. The Active Directory, LDAP, and RADIUS authentication modules now allow one or more servers to be designated as primary or secondary servers.

  When authenticating users from a directory server that is remote to OpenAM, set the primary server values, and optionally, the secondary server values. Primary servers have priority over secondary servers.

  ssoadm attributes are: primary is `iplanet-am-auth-ldap-server`; secondary is `iplanet-am-auth-ldap-server2`.

  Both properties take more than one value; thus, allowing more than one primary or secondary remote server, respectively. Assuming a multi-data center environment, OpenAM determines priority within the primary and secondary remote servers, respectively, as follows:

  - Every LDAP server that is mapped to the current OpenAM instance has highest priority.

For example, if you are connected to `openam1.example.com` and `ldap1.example.com` is mapped to that OpenAM instance, then OpenAM uses `ldap1.example.com`.

- Every LDAP server that was not specifically mapped to a given OpenAM instance has the next highest priority.

  For example, if you have another LDAP server, `ldap2.example.com`, that is not connected to a specific OpenAM server and if `ldap1.example.com` is unavailable, OpenAM connects to the next highest priority LDAP server, `ldap2.example.com`.

- LDAP servers that are mapped to different OpenAM instances have the lowest priority.

  For example, if `ldap3.example.com` is connected to `openam3.example.com` and `ldap1.example.com` and `ldap2.example.com` are unavailable, then `openam1.example.com` connects to `ldap3.example.com`.

For more information, see OPENAM-3575.

- **Legacy User Self Service Endpoints Disabled by Default**.

  The REST endpoints used by the legacy user self service features, such as registering for an account or resetting a forgotten password, are now disabled by default.

  Legacy deployments should migrate to the new user self-service features in OpenAM 13.5, see Chapter 8, "Configuring User Self-Service Features" in the *OpenAM Administration Guide*.

  To restore the legacy endpoints, enable the Configure > Global Services > Legacy User Self Service > Legacy Self-Service REST Endpoint option.

  Warning

  Restoring the legacy self service endpoints allows REST requests crafted such that the body of the self-service email contains a malicious URL that end users may visit, hiding the correct OpenAM URL that is appended to the end of the email body.

- **REST Endpoint Changes**

Version 3.0 of the `/users` endpoint is provided in this release of OpenAM. The response differs from version 2.0 of the endpoint, which remains available for backwards compatibility.

The new version of the endpoint returns details about all users. The previous version only returned a list of usernames.

Version 3.0 of the `/users` endpoint does not support the following `_action` values:

```
https://openam.example.com:8443/openam/json/users/?_action=register
https://openam.example.com:8443/openam/json/users/?_action=confirm
https://openam.example.com:8443/openam/json/users/?_action=anonymousCreate
https://openam.example.com:8443/openam/json/users/?_action=forgotPassword
https://openam.example.com:8443/openam/json/users/?_action=forgotPasswordReset
```

Responses to Different Versions of the /users Endpoint

In this section, long URLs are wrapped to fit the printed page, and some of the output is formatted or truncated for easier reading.

Version 3.0 of the `/users` endpoint:

Version 2.0 of the `/users` endpoint:

```
$ curl \
--header "iPlanetDirectoryPro: AQIC5w...2NzEz*" \
--header Accept-API-Version: protocol=1.0,resource=2.0 \
"https://openam.example.com:8443/openam/json/users?_queryId=*"
{
    "result": [
        "amAdmin",
        "demo"
    ],
    "resultCount": 2,
    "pagedResultsCookie": null,
    "totalPagedResultsPolicy": "NONE",
    "totalPagedResults": -1,
    "remainingPagedResults": -1
}
```

- **Workaround for java.lang.VerifyError in WebSphere**.

  When loading classes from OpenAM within WebSphere Application Server using the IBM Technology for JVM and Apache Axis2 framework, a `java.lang.VerifyError JVMVRFY013 class loading constraint violated` error may occur. For more information on the `java.lang.VerifyError` error, see "java.lang.VerifyError: JVMVRFY013 class loading constraint violated" Error.

  Procedure 4.2. Fixing a WebSphere java.lang.VerifyError Error

  1. Remove the following JARs from the `WEB-INF/lib` directory in the `openam.war` file:

     - `jaxp-api-1.4.2.jar`
     - `xercesImpl-2.11.0.jar`
     - `xml-apis-2.11.0.jar`
     - `xml-resolver-2.11.0.jar`
     - `xml-serializer-2.11.0.jar`

     For instructions on how to expand the `openam.war` file, make changes to `bootstrap.properties` file, and then rebuild the `openam.war` file, see Procedure 1.6, "To Prepare OpenAM for JBoss and WildFly" in the *OpenAM Installation Guide*.

  2. Set the following custom JVM properties on the WebSphere server:

```
-Djavax.xml.soap.MessageFactory=com.sun.xml.internal.messaging.saaj.soap.ver1_1.SOAPMessageFactory1
-Djavax.xml.soap.SOAPFactory=com.sun.xml.internal.messaging.saaj.soap.ver1_1.SOAPFactory1_1Impl
-Djavax.xml.soap.SOAPConnectionFactory=com.sun.xml.internal.messaging.saaj.client.p2p.HttpSOAPConne
-Djavax.xml.soap.MetaFactory=com.sun.xml.internal.messaging.saaj.soap.SAAJMetaFactoryImpl
-Dcom.ibm.websphere.webservices.DisableIBMJAXWSEngine=true
```

3. Restart the WebSphere server.

- **Different return type for GetUserInfo method of ScopeValidator interface**.

  The return type for the `getUserInfo` method of the `org.forgerock.oauth2.core.ScopeValidator` interface, formerly `Map<String, Object>`, is now `org.forgerock.oauth2.core.UserInfoClaims`. The new return type lets callers of the `getUserInfo` method see values of users' claims.

  This change affects OAuth v2.0 scope validator plugins. For more information, see Section 3.2, "Customizing OAuth 2.0 Scope Handling" in the *OpenAM Developer's Guide*.

- **Oracle Directory Server Enterprise Edition no longer supported for the OpenAM configuration store**.

  In previous versions, it was possible to deploy the OpenAM configuration store in an external Oracle Directory Server Enterprise Edition instance.

  In OpenAM 13.5, this is no longer possible. You must deploy the OpenAM configuration store in an OpenDJ server instance: either the embedded OpenDJ directory server instance that is installed together with OpenAM, or in an external server instance.

- **The steps to install and configure a Java Fedlet have changed**.

  In previous versions, the Create Fedlet wizard included the federation configuration in the `fedlet.war` file, and added the `fedlet.war` file to the `Fedlet.zip` file. This is no longer the case, and as a result, the steps you perform to install and configure a Java Fedlet have changed.

  For updated steps to install and configure a Java Fedlet, see Section 6.1.1, "Creating and Installing a Java Fedlet" in the *OpenAM Developer's Guide*.

- **Persistent cookies are now encrypted and signed**.

  In previous versions, persistent cookies were encrypted with OpenAM's public RSA key. OpenAM 13.5 now signs the persistent cookie with a user-specified HMAC signing key in addition to encrypting it.

  For information about the new HMAC Signing Key property in the Persistent Cookie authentication module, see Section 2.5.20, "Hints for the Persistent Cookie Module" in the *OpenAM Administration Guide*.

4.2 Deprecated Functionality

No features have been deprecated in OpenAM 13.5.

The following features are deprecated in OpenAM 13.0.0:

- The OpenAM Logging, User Self Service, and Password Reset Services are deprecated. The User Self Service has been renamed to Legacy User Self Service. New audit logging and user self-service capabilities are available in OpenAM 13.0.0.
- The classic JATO-based UI is deprecated for the end-user pages and replaced in OpenAM with the JavaScript-based XUI as a replacement. The classic UI for end user pages is likely to be removed in a future release.
- Listing tokens with the `/frrest/oauth2/token/?_queryId` method is deprecated. Improved `_queryFilter` support will be added to replace the `_queryId` method.
- The Device Print Service is deprecated. For information on replacement device identification features, see Section 2.5.6, "Hints for the Device ID (Match) Authentication Module" in the *OpenAM Administration Guide*.

4.3 Removed Functionality

The following functionality has been removed from OpenAM 13.5:

- **Server configuration property removed**. The following server configuration property has been removed from OpenAM:
  - `com.sun.am.event.connection.idle.timeout`
- Network Security Services for Java (JSS) has been removed from OpenAM. As a result, OpenAM no longer provides native support for Federal Information Processing Standard (FIPS) mode.

  FIPS mode should instead be provided by your JRE. Use the following links for more information about enabling FIPS support:

  - Oracle JRE
  - IBM JRE
- OpenAM's implementation of OAuth v1.0 has been removed.
- The Administration Service has been removed.

The following functionality has been removed from OpenAM 13.0.0:

- The `/identity` endpoints that were not previously deprecated are no longer in OpenAM:
  - `/log`
  - `/getCookieNamesForToken`
  - `/getCookieNamesToForward`
- Use of the legacy Netscape LDAP SDK is replaced by the OpenDJ SDK.
- The `sun-idrepo-ldapv3-config-connection-mode` property replaces `sun-idrepo-ldapv3-config-ssl-enabled`, which has been removed from the configuration schema (`sunIdentityRepositoryService`).

  For more information, see OPENAM-3714.
- The `openam-auth-ldap-connection-mode` property replaces `iplanet-am-auth-ldap-ssl-enabled`, which has been removed from the configuration schema (`sunAMAuthADService` and `iPlanetAMAuthLDAPService`).

  For more information, see OPENAM-5097.
- **New `openam.deserialisation.classes.whitelist` Property**. OpenAM uses the JATO framework for some console pages and for legacy login pages. The JATO framework uses serialized Java objects to maintain state during the console session.

  To ensure that the serialized objects have not been exploited by a malicious user, OpenAM now provides a new `openam.deserialisation.classes.whitelist` property that lists valid classes when OpenAM performs object deserialization. The default should work for most deployments.

  To access and update the property on the OpenAM console, navigate to Configure > Server Defaults, click Security, and then click the Object Deserialisation Class Whitelist tab.

  For more information, see OPENAM-5925.
- REST services relying on the following endpoints have been removed from OpenAM.
  - `/identity/attributes`
  - `/identity/authenticate`
  - `/identity/authorize`
  - `/identity/create`

  - `/identity/delete`
  - `/identity/isTokenValid`
  - `/identity/logout`
  - `/identity/read`
  - `/identity/search`
  - `/identity/update`
  - `/json/[realm/]referrals`
  - `/ws/1/entitlement/decision`
  - `/ws/1/entitlement/decisions`
  - `/ws/1/entitlement/entitlement`
  - `/ws/1/entitlement/entitlements`
  - `/ws/1/entitlement/listener`
  - `/ws/1/entitlement/privilege`
  - `/ws/1/token`
- The Persistent Cookie (Legacy) settings in the Core Authentication module have been removed. For information on how to configure persistent cookies in this release, see Section 2.5.20, "Hints for the Persistent Cookie Module" in the *OpenAM Administration Guide*.
- The server-only WAR file has been removed from the OpenAM distribution. For information about how to remove console access in OpenAM 13, see How do I remove console access in OpenAM 13? in the *ForgeRock Knowledge Base*.
- The Distributed Authentication Service (DAS) has been removed.
- Referral policies are no longer available in OpenAM. If you are upgrading from a previous version of OpenAM and currently use referral policies, please refer to the OpenAM Upgrade Guide for migration information.
- Specifying a realm in POST data is no longer supported. A number of other methods are supported, such as specifying the realm as a query parameter.

Chapter 5

Fixes, Limitations, and Known Issues

OpenAM issues are tracked at https://bugster.forgerock.org/jira/browse/OPENAM. This chapter covers the status of key issues and limitations at release 13.5.

5.1 Key Fixes

The following bugs were fixed in release 13.5. For details, see the OpenAM issue tracker.

5.1.1 Key Fixes in OpenAM 13.5.0

The following important issues were fixed in this release:

- OPENAM-1945: Default Configuration create invalid domain cookie
- OPENAM-3095: When a SP sends an unsigned Authn Request using SAML ECP OpenAM sees it as a wrong message
- OPENAM-5264: Can't login to OpenAM with no cookies set in the platform service
- OPENAM-6362: HOTP and OATH auth-modules do not set 'failureUserID' when throwing InvalidPasswordException, this breaks OpenAM account lockout
- OPENAM-6878: OpenAM forgot password search hard coded for UID

- OPENAM-7002: The email attribute property defined in the email service is not used when sending e-mail in forgotten password flow
- OPENAM-7298: Custom response attributes are not visible in the policy editor UI and are erased when editing policies through the UI
- OPENAM-7320: Consider using JDK JAXP/XML instead of Xerces/Xalan to keep up with JDK fixes
- OPENAM-7778: XML Signature DigestMethod should be configurable when using SAML2
- OPENAM-7820: Additional delete/revoke token endpoints for Oauth2
- OPENAM-7864: Failure to connect to syslog server can cause OpenAM to hang
- OPENAM-8074: Changing an user password with the same value returns 400 with ldap errorcode=20
- OPENAM-8091: OpenAM cannot connect to a DataStore which accepts only TLSv1.2
- OPENAM-8108: Radius auth module not usable in auth-chain with 'shared-state' enabled
- OPENAM-8125: IE 9/10: can't create policy resource
- OPENAM-8142: OAuth2 Access Tokens are inaccessible if the OAuth2 Client contains a space in their name
- OPENAM-8174: OpenAM gives an Internal Server Error when the user tries to reset their password before the minimum password age
- OPENAM-8194: The default WS-Fed IDP attribute mapper should provide a way to Base64 encode binary attributes
- OPENAM-8225: Reading binary attributes, for example objectGUID, from the IdRepo cache not always returning valid values
- OPENAM-8282: Password Reset questions are not randomly chosen when resetting password
- OPENAM-9370: Configuration dialog stuck after successful configuration on weblogic

5.2 Limitations

The following limitations and workarounds are for OpenAM 13.5.0:

- **JCEKS Keystore Support for User Self-Services**. In OpenAM 13.0.0, OpenAM's user self-service feature is stateless, which means that the end-user is tracked and replayed by an encrypted and signed JWT token on each OpenAM instance. It also generates key pairs and caches its keys locally on the server instance.

  In a multi-instance deployment behind a load balancer, one server instance with the user self-services enabled will not be able to decrypt the JWT token from the other instance due to the encryption keys being stored locally to its server.

  OpenAM 13.5.0 solves this issue by providing a JCEKS keystore that supports asymmetric keys for encryption and symmetric keys for signing. Users who have installed OpenAM 13.0.0 and enabled the user self-service feature will need to run additional steps to configure a JCEKS keystore to get the user self-service feature operating after an upgrade to OpenAM 13.5.0.

  Note that for users of the IBM JDK on Websphere must generate their own JCEKS keystore to replace the default installation. Currently, the IBM JDK cannot load a JCEKS keystore created from a Sun/Oracle JDK on Websphere. This issue will be fixed in OpenAM 14.0.0.

  For specific instructions to configure the JCEKS keystore, see Section 8.3.1, "Configuring the Signing and Encryption Key Aliases" in the *OpenAM Administration Guide*.

  Note

  This procedure is not necessary for the following users:

  - Users upgrading from versions prior to OpenAM 13.0.0 to OpenAM 13.5.0 are not impacted.
  - Users who upgrade from OpenAM 13.0.0 to OpenAM 13.5.0 and do not enable the user self-services feature are not impacted.
  - Users who do a clean install of OpenAM 13.5.0 are not impacted.

The following limitations and workarounds are for OpenAM 13.0.0:

- **Cached JavaScript Files from OpenAM 12.0.0 May Cause Redirect to undefined:8080**. If you configure an OpenAM 12.0.0 instance with long-lived cache times for the `/XUI/index.html` file, you may experience unexpected redirects to `undefined:8080` after upgrading to OpenAM 13.

To work around this issue, in your chosen web container, or proxy server, reconfigure the cache time for the `/XUI/index.html` file to be short-lived, for example, 5 minutes. Allow enough time that cached files with the long-lived cache time will have expired before upgrading to OpenAM 13.

Note

This issue does not affect upgrades from OpenAM 12.0.1 or later. OpenAM 12.0.1 and later set a short-lived `cache-control` header on UI files to work around the problem of having stale files cached locally.

- **RADIUS Service Only Supports Commons Audit Logging**. The new RADIUS service only supports the new Commons Audit Logging, available in this release. The RADIUS service cannot use the older Logging Service, available in releases prior to OpenAM 13.0.0.
- **Administration Console Access Requires the `RealmAdmin` privilege** . In OpenAM 13.5, administrators can use the OpenAM administration console as follows:
  - Delegated administrators with the `RealmAdmin` privilege can access full administration console functionality within the realms they can administer. In addition, delegated administrators in the Top Level Realm who have this privilege can access OpenAM's global configuration.
  - Administrators with lesser privileges, such as the `PolicyAdmin` privilege, can not access the OpenAM administration console.
  - The top-level administrator, such as `amadmin`, has access to full console functionality in all realms and can access OpenAM's global configuration.
- **Do Not End Policy Names with a "/" Character**. Do not use a "/" character at the end of a policy name as it will cause OpenAM to not read, edit, or delete the policy.

  After upgrade, users who have policies with a trailing slash "/" character at the end of a policy name should remove the slash (OPENAM-5400). ways:

  To remove slashes in the policy names, remove them as recommended in: OPENAM-5187.
- **Upgrade Incorrectly Sets Default Value for the REST APIs Service**. The workaround is to manually set the default version setting in the REST APIs service to the preferred value:

```
$ openam/bin/ssoadm set-attr-defs -s RestApisService -t Global \
     -a openam-rest-apis-default-version=Latest -u amadmin -f .pass
```

For background information, see OPENAM-6302.

- **OAuth2 Scopes Behavior Affected by Upgrade**. After an upgrade, OAuth v2.0 scope behavior uses a deprecated implementation class, `org.forgerock.openam.oauth2.provider.impl.ScopeImpl`.

  The workaround is to manually update the OAuth v2.0 providers to use the `org.forgerock.openam.oauth2.OpenAMScopeValidator`.

  For background information, see OPENAM-6319.

- **Different OpenAM Version within a Site**. Do not run different versions of OpenAM together in the same OpenAM site.

- **Avoid Use of Special Characters in Policy or Application creation**. Do not use special characters within policy, application or referral names (for example, "my+referral") using the Policy Editor or REST endpoints as OpenAM returns a 400 Bad Request error. The special characters are: double quotes ("), plus sign (+), command (,), less than (<), equals (=), greater than (>), backslash (\), and null (\u0000). (OPENAM-5262)

- **Avoid Using REST Endpoint Names for Realm Names**. Do not use the names of OpenAM REST endpoints as the name of a realm. The OpenAM REST endpoint names that should not be used includes: "users", "groups", "realms", "policies" and "applications". (OPENAM-5314)

- **Deploying OpenAM on Windows in an IPv6 Network**. When deploying OpenAM components on Microsoft Windows in an IPv6 environment, you must use the Java 7 Development Kit on Windows (due to JDK-6230761, which is fixed only in Java 7).

- **Database Repository Type is Experimental**. The Database Repository type of data store is experimental and not supported for production use.

- **Enforcing Session Quotas with Session Failover**. By default OpenAM does not enforce session quotas when running in Site mode without session failover. To work around this behavior, set the server configuration property `openam.session.useLocalSessionsInMultiServerMode` to `true`. You can set this property in OpenAM console under Deployment > Servers > *`Server Name`* > Advanced.

- **XACML Policy Import and Export**. OpenAM can only import XACML 3.0 files that were either created by an OpenAM instance, or that have had minor manual modifications, due to the reuse of some XACML 3.0 parameters for non-standard information.

- **Custom Profile Attributes Are Visible in the User Profile Only With the Classic UI**. The ability to view and edit custom profile attributes is limited to the classic UI. Custom profile attributes do not appear in the user profile when users log in to OpenAM using the XUI.

- **"Unknown Error. Please contact your administrator" Message Appears When Accessing Classic UI Pages in the OpenAM Console** . In certain deployments, administrators are unable to access Classic UI pages in OpenAM console from XUI pages, instead receiving the message, "Unknown Error. Please contact your administrator." For example, selecting Federation from the top-level menu can cause this problem to occur.

  The problem is the result of an incompatibility between older versions of Tomcat 7 and JDK 8. If you encounter this problem, upgrade the version of Tomcat 7 in which you run OpenAM to work around it.

5.3 Known Issues

The following important known issues remained open at the time release 13.5 became available. For details and information on other issues, see the OpenAM issue tracker.

5.3.1 Known Issues in OpenAM 13.5.0

The following important issues remained open when OpenAM 13.5.0 became available:

- OPENAM-71: SAML2 error handling in HTTP POST and Redirect bindings
- OPENAM-1105: Init properties sometimes don't honor final settings
- OPENAM-1194: Unable to get AuthnRequest error in multiserver setup
- OPENAM-1323: Unable to create session service when no datastore is available
- OPENAM-1660: Read-access to SubjectEvaluationCache is not synchronized
- OPENAM-2168: Authentication Success Rate and Authentication Failure Rate are always 0
- OPENAM-2911: IdP initiated SSO with persistent identifier causes URLNotFoundException: Invalid service host name.
- OPENAM-9307: Significant number of internal errors while generating access_token load

- OPENAM-9357: Upgrading to 13.x does not populate Subject Type in OAuth2Client config, causing an NPE
- OPENAM-9358: Default Microsoft Social Auth login configuration fails

Chapter 6

How to Report Problems or Provide Feedback

If you have questions regarding OpenAM which are not answered by the documentation, there is a mailing list which can be found at https://lists.forgerock.org/mailman/listinfo/openam where you are likely to find an answer.

If you have found issues or reproducible bugs within OpenAM 13.5, report them in https://bugster.forgerock.org.

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
  - Web server or container and version
  - Java version
  - OpenAM 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

The following table tracks changes to the documentation set following the release of OpenAM 13.5:

Table 7.1. Documentation Change Log

Date	Description
2016-07-20	Initial release of OpenAM 13.5.0.
2016-11-21	OpenAM 13.5.0 documentation refresh 1, which includes the following updates: • Revised and clarified Procedure 3.1, "To Set up Administration Tools" in the *OpenAM Installation Guide*. • Corrected the example Elasticsearch index in Procedure 6.5, "To Prepare for Elasticsearch Audit Logging" in the *OpenAM Administration Guide*. Elasticsearch indexes must be all lower case, and the example had an index with some upper case characters. • Corrected the description of the Mobile Carrier Attribute Name in Section 2.5.12, "Hints for the HOTP Authentication Module" in the *OpenAM Administration Guide*.

Date	Description
	• Added a new section, Section 22.2, "Managing Scripts With the ssoadm Command" in the *OpenAM Administration Guide*, which provides **ssoadm** command examples of script management. • Revised Section 1.11, "Preparing Oracle WebLogic" in the *OpenAM Installation Guide*, to provide additional guidance for deploying OpenAM into WebLogic and removed outdated information. There is now a new section with steps to perform before deploying OpenAM in WebLogic. • Modified example paths in Procedure 1.2, "To Prepare an External OpenDJ Identity Repository With Manual Schema Updates" in the *OpenAM Installation Guide* that could lead to confusion. • Corrected the example SSL connector in Procedure 23.1, "To Set Up OpenAM With HTTPS and Self-Signed Certificates" in the *OpenAM Administration Guide* to add the `keystoreType` property and to update the `protocol` property to the default used by Apache Tomcat8.x. • Corrected and clarified the steps to change the `amadmin` user's password in Section 27.5, "Administering the amadmin Account" in the *OpenAM Administration Guide*. The section now contains a procedure to change the `amadmin` user's password when the configuration store is in the embedded OpenDJ server, and another procedure to change the password when the configuration store is in an external OpenDJ server. • Revised Chapter 23, "Managing Certificates and Keystores" in the *OpenAM Administration Guide* to provide more information about key aliases and keystores in OpenAM. Added new procedures to configure the keystore and to change the user self-service key aliases. The procedure to change the signing key is also updated. • Updated the documentation set to reflect that OpenAM 13.5 defaults to JCEKS keystore instead of to JKS keystore.

Date	Description
	• Revised and clarified Chapter 21, "Backing Up and Restoring OpenAM Configurations" in the *OpenAM Administration Guide*. • Updated Chapter 4, "Localization" in the *OpenAM Reference* to include XUI localization support information. • Updated the file descriptor section with some additional instructions for daemon processes. See Section 1.3, "Setting Maximum File Descriptors" in the *OpenAM Installation Guide*. • Updated the Configuring the Core Token Service with small fixes to the instructions in Chapter 6, "Configuring the Core Token Service" in the *OpenAM Installation Guide*. • Updated the list of distribution files in the OpenAM Deployment Planning Guide in Section 1.3, "OpenAM Server Overview" in the *OpenAM Deployment Planning Guide*. • Removed the `com.sun.am.event.connection.idle.timeout` from the OpenAM Reference Guide. • Added a link to create a configuration store backend in Procedure 2.4, "To Custom Configure OpenAM" in the *OpenAM Installation Guide*. • Added ability to use UTF-8 encoded user names and passwords during REST authentication. See Section 2.1.1.4, "Authentication and Logout" in the *OpenAM Developer's Guide*. • Added Microsoft Edge to the list of supported browsers, and changed references in the documentation from "Internet Explorer" to "Internet Explorer and Microsoft Edge." • Revised the chapter on CTS OIDs to include the OIDs' data types, and corrected several entries in the CTS OIDs diagram. See Chapter 8, "Core Token Service (CTS) Object Identifiers" in the *OpenAM Reference*.

Date	Description
	• Updated Active Directory Hints section to indicate that `cn` gets its value from the `uid` or `username` and `sn` gets its value from `givenName` in Section 4.3.1, "Hints for Configuring Active Directory Data Stores" in the *OpenAM Administration Guide*. • Updated the HMAC One-Time-Password (HOTP) hints with a note about configuring login page session timeouts in Section 2.5.12, "Hints for the HOTP Authentication Module" in the *OpenAM Administration Guide*. • Updated the information on cookie domain values, which can now be empty strings for host-only cookies or any non-top level domain in Section 1.1, "Preparing a Fully Qualified Domain Name" in the *OpenAM Installation Guide*.

Chapter 8

Support

You can purchase OpenAM support subscriptions and training courses from ForgeRock and from consulting partners around the world and in your area. To contact ForgeRock, send mail to info@forgerock.com. To find a partner in your area, see http://forgerock.com/partners/find-a-partner/.