Release Notes

/ ForgeRock Access Management 7.1

Latest update: 7.1.0

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

Abstract

Notes covering new features, fixes and known issues in ForgeRock® Access Management (AM). ForgeRock Access Management provides intelligent authentication, authorization, federation, and single sign-on functionality.

Table of Contents

Overview

Read these release notes before you install ForgeRock Access Management or update your existing installation.

The information contained in these release notes cover prerequisites for installation, known issues and improvements to the software, changes and deprecated functionality, and other important information.

About ForgeRock Identity Platform™ 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.

Chapter 1
What's New

This chapter covers the new features and improvements done in the current release of ForgeRock Access Management.

Release Dates

Version	Date
AM 7.1	2021-05-12

For end of service life dates (EOSL), see the Checking Your Product Versions Are Supported article in the *ForgeRock Knowledge Base*.

New Features

AM 7.1

ForgeRock Access Management 7.1 is a minor release that introduces new features, functional enhancements, and fixes.

- OAuth 2.0 and OpenID Connect Token Exchange Support

  Following the OAuth 2.0 Token Exchange specification, AM 7.1 now lets you exchange ID tokens and access tokens in delegation and impersonation use cases.

  For more information, see "*OAuth 2.0 Token Exchange*" in the *OAuth 2.0 Guide*.

- Social Identity Provider Client Improvements

  AM 7.1 enhances the OAuth 2.0/OpenID Connect client support offered in the Social Identity Provider Service. To connect to financial-grade identity providers, AM and ForgeRock Identity Platform can now:

  - Configure `acr` values to specify a set of rules that the authorization request must satisfy when authenticating to the provider; for example, using multi-factor authentication.

    + *Learn More*

A new property, ACR Values, has been to the OpenID Connect secondary configuration of the Social Identity Provider Service.

- Accept encrypted ID tokens.

  + *Learn More*

  AM includes a new JWK URI, which the provider can use to obtain keys for verifying request object signatures, and for encrypting ID tokens.

  Two new properties have been added to the OpenID Connect secondary configuration of the Social Identity Provider Service:

  - OP Encrypts ID Tokens
  - Issuer

- Send request parameters in a JWT, or as a reference to a JWT.

  + *Learn More*

  The JWT is always signed, and optionally encrypted.

  As part of this change, the following fields have been added to the OpenID Connect secondary configuration of the Social Identity Provider Service:

  - Request Parameter JWT Option
  - Request Object Audience
  - Encrypt Request Parameter JWT
  - JWT Signing Algorithm
  - JWT Encryption Algorithm
  - JWT Encryption Method

- Authenticate using a JWT or mutual TLS (mTLS).

  + *Learn More*

  The JWT is always signed, and optionally encrypted.

As part of this change, the Use Basic Auth switch in the client has been replaced with the Client Authentication Method drop-down list, which contains the following options:

- `CLIENT_SECRET_POST`
- `CLIENT_SECRET_BASIC`
- `PRIVATE_KEY_JWT`
- `ENCRYPTED_PRIVATE_KEY_JWT`
- `TLS_CLIENT_AUTH`
- `SELF_SIGNED_TLS_CLIENT_AUTH`

Moreover, AM 7.1 also includes a new advanced server property, `openam.private.key.jwt.encryption.algorithm.whitelist` in the *Reference*, that specifies the algorithms that the client can use to encrypt both authentication JWTs and request object JWTs.

- Allow providers to return ID tokens by submitting an HTML form using the HTTP POST method, as defined in the OAuth 2.0 Form Post Response Mode specification.

  + *Learn More*

  Moreover, the Response Mode drop-down list has been added to the OpenID Connect secondary configuration of the Social Identity Provider Service.

  Moreover, the Redirect after form post URL property has been added to support the form post response mode in custom login pages.

Moreover, AM now provides a preconfigured client for Apple and itsme.

For more details, see "*Social Authentication*" in the *Authentication and Single Sign-On Guide* and "/oauth2/connect/rp/jwk_uri" in the *OpenID Connect 1.0 Guide*.

- OpenID Connect Backchannel Logout Supported

  As the OpenID provider, AM 7.1 now supports the OpenID Connect Back-Channel Logout 1.0 Draft 06. This draft lets AM send *logout tokens* to relevant relying parties when a session associated with an ID token becomes invalid.

  As part of this change, the Store OPS Tokens switch, used to enable session management at the provider, has been renamed to OIDC Session Management.

  Also, when OIDC Session Management is enabled, ID tokens will now contain a new claim, `sid`, which specifies a session ID that identifies the relying party's session with the provider. The `sid` can also be found in the logout tokens, if enabled.

For more information, see "Informing Relying Parties that a Session has Expired" in the *OpenID Connect 1.0 Guide*.

- Add Push Authentication Nodes

  AM 7.1 adds a number of new authentication nodes to assist with push authentication:

  - Opt-out Multi-Factor Authentication Node
  - Push Registration Node
  - MFA Registration Options Node
  - Get Authenticator App Node

- New Account Active Check Authentication Module

  AM 7.1 includes a new Account Active Check authentication module, which lets you determine whether an account is marked as active, or locked, without having to run through the remainder of the authentication chain.

  For more details, see "Account Active Check Module" in the *Authentication and Single Sign-On Guide*.

- New Properties Available to Claims and Access Token Scripts

  AM 7.1 adds new properties to the *OpenID Connect Claims* and *OAuth 2.0 Access Token Modification* script types, for accessing the properties of the relevant client, and the incoming request.

  For more details, see "Scripting OpenID Connect 1.0 Claims" in the *OpenID Connect 1.0 Guide* and "*Modifying the Content of Access Tokens*" in the *OAuth 2.0 Guide*.

- New Live and Ready Status Endpoints

  AM 7.1 includes new endpoints to determine if an instance is alive, and ready to process requests.

  For more details, see "*Monitoring Instances*" in the *Maintenance Guide*.

- New Access to Secrets and Credentials in Authentication Scripts

  AM 7.1 adds the ability for scripted decision nodes to access the secrets configured in AM secret stores.

  For example, a script can access credentials or secrets defined in a file system secret volume in order to make outbound calls to a third-party REST service, without hard-coding those credentials in the script.

  For more details, see "Accessing Credentials and Secrets" in the *Authentication and Single Sign-On Guide*.

- New Support for PEM-Formatted Keys and Certificates

  AM 7.1 adds support for loading the following PEM-formatted secrets:

  - Elliptic Curve and RSA private keys
    - OpenSSL format
    - PKCS#8 format
  - X.509 certificates
  - RSA public keys
  - (non-standard) AES secret keys
  - (non-standard) HMAC secret keys
  - (non-standard) Generic secrets, such as connection passwords or API keys

  ForgeRock recommends that you use PEM secrets on the secret stores that support it:

  - "The Environment and System Property Secrets Store" in the *Security Guide*
  - "File System Secret Volumes Secret Stores" in the *Security Guide*
  - "Google GSM Secret Stores" in the *Security Guide*

  For more information, see "Importing PEM-Formatted Keys" in the *Security Guide*.

- The Session Service Now Uses Secret Stores

  Client-based sessions and client-based authentication sessions now use secret stores for:

  - Signing JWTs with RSA and elliptic curve algorithms.
  - Encrypting JWTs with RSA algorithms.

  The upgrade process migrates the relevant configuration to secret stores automatically.

  HMAC signing secrets and symmetric AES keys for encryption have not been migrated yet, and are still available in the Session service configuration page.

  For more information, see "Configuring Client-Based Session Security" in the *Security Guide*.

- Loading Secrets from Google Secret Manager Supported

  AM 7.1 now lets you load secrets from Google Secret Manager (GSM).

  For more information, see "Google GSM Secret Stores" in the *Security Guide*.

Major Improvements

AM 7.1

- The SAML v2.0 Node Now Sets the `successUrl` Parameter

  The SAML v2.0 authentication node now sets the `successURL` parameter in the tree's shared state to the value of the `RelayState` parameter in the request, if any.

  If the request does not provide a value, the node uses the default `RelayState` value configured in the SP.

- The JWK URI Endpoint Can Now Return Duplicate Key IDs

  Earlier versions of AM removed the `alg` parameter from the keys returned by the `jwk_uri` endpoint.

  Removing the `alg` parameter ensures that each key ID (`kid`) exposed by the endpoint matches a unique key, as recommended by the RFC7517 specification.

  AM 7.1 includes a toggle, Include all kty and alg combinations in jwk_uri, that lets the endpoint display duplicate key IDs with their corresponding `alg` and `kty` parameters.

  The toggle property is disabled by default.

  For more information, see "Displaying Every Algorithm and Key Type Associated to a Key ID" in the *OpenID Connect 1.0 Guide*.

- Improved Workflow for Adding Servers to Existing Deployments

  A new option is available when installing an AM instance which lets you choose whether the instance is standalone, or part of an existing deployment.

  For more information, see "To Add a Server to a Site" in the *Installation Guide*.

- Improved Client Connection Handling

  AM 7.1 improves the way its ClientHandler code handles connection pools and timeouts. This affects client connections that AM opens against third parties, such as social identity providers.

  As part of this change, AM includes the following new advanced server properties:

  - `org.forgerock.openam.httpclienthandler.system.clients.connection.timeout`
  - `org.forgerock.openam.httpclienthandler.system.clients.max.connections`
  - `org.forgerock.openam.httpclienthandler.system.clients.pool.ttl`
  - `org.forgerock.openam.httpclienthandler.system.clients.response.timeout`
  - `org.forgerock.openam.httpclienthandler.system.clients.retry.failed.requests.enabled`

- `org.forgerock.openam.httpclienthandler.system.clients.reuse.connections.enabled`

For more information, see "Advanced Properties" in the *Reference*.

- Configuration Upgrade Tool Distributed With AM ZIP

  The `AM-7.1.0.zip` file now includes a configuration file upgrade tool for converting configuration files exported with the **Amster** command. The tool is provided in the `Config-Upgrader-7.1.0.zip` file, which is inside the `AM-7.1.0.zip` file.

  For more information, see What's New in the Amster documentation.

- Changes to the Retry Limit Decision Node

  The "Retry Limit Decision Node" in the *Authentication and Single Sign-On Guide* can now persist the number of failed login attempts in the identity store between successful authentications.

  To support this change, the following LDIF schema files have been updated:

  - `ad_user_schema.ldif`
  - `adam_user_schema.ldif`
  - `odsee_user_schema.ldif`
  - `opendj_remove_user_schema.ldif`
  - `opendj_user_schema.ldif`
  - `tivoli_user_schema.ldif`

  Moreover, a new file, `opendj_retry_limit_node_count.ldif`, has been added to the AM deliverables, and the DS identity setup profile has been updated.

  This new functionality is enabled by default. You must apply the new schema(s) to the identity store when upgrading to AM 7.1.

  For more information, see "*Upgrading AM Instances*" in the *Upgrade Guide*.

- Improved AES Wrap Encryption Performance

  AM 7.1 includes a new advanced server property, `org.forgerock.openam.encryption.useextractandexpand`, that specifies whether to use an improved algorithm that reduces the cost of AES Key Wrap encryption even when high iteration counts are used.

  The new algorithm is backwards-compatible; data already encrypted will be decrypted at the old performance cost, and newly-encrypted data will benefit from the improvements.

  The property is disabled by default after upgrading to AM 7.1. To enable it, configure it in your container's environment file.

For more information, see "Preparing AES Key Wrap Encryption" in the *Installation Guide*.

- The Social Provider Handler Node Can Now Be Used in Standalone AM Deployments

  Previous versions of AM required a ForgeRock Identity Platform deployment to use the "Social Provider Handler Node" in the *Authentication and Single Sign-On Guide*.

  AM 7.1 can use the node in standalone mode. The node will use the identity store configured in the realm to retrieve the user's profile, if exists. However, account claiming remains a ForgeRock Identity Platform-only feature.

  As part of this change, the following authentication nodes have been deprecated:

  - "OpenID Connect Node" in the *Authentication and Single Sign-On Guide*
  - "OAuth 2.0 Node" in the *Authentication and Single Sign-On Guide*
  - "Social Facebook Node" in the *Authentication and Single Sign-On Guide*
  - "Social Google Node" in the *Authentication and Single Sign-On Guide*
  - "Provision IDM Account Node" in the *Authentication and Single Sign-On Guide*

  The Social Authentication Implementations Service is also deprecated.

  The "*Social Authentication*" in the *Authentication and Single Sign-On Guide* documentation page has been updated with information about configuring the "Social Provider Handler Node" in the *Authentication and Single Sign-On Guide*, the "Select Identity Provider Node" in the *Authentication and Single Sign-On Guide*, and the Social Identity Provider Service.

- Web Authentication Improvements

  - You can now use the Android SafetyNet attestation format when registering and authenticating with Android devices using WebAuthn.
  - The "WebAuthn Authentication Node" in the *Authentication and Single Sign-On Guide* also now supports FacetID for mobile as per the FIDO AppID and Facet Specification.
  - Both the "WebAuthn Registration Node" in the *Authentication and Single Sign-On Guide* and the "WebAuthn Authentication Node" in the *Authentication and Single Sign-On Guide* now return a JSON as part of their metadata callbacks.

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*.

Chapter 2

Before You Install

This chapter covers software and hardware prerequisites for installing and running ForgeRock Access Management server software.

Important Information Before Installing Access Management

Files to Download	Operating Systems	Web and Java Agents
Java	Application Containers	Directory Servers
Third-Party Software	Clients and Browsers	Special Requests

Important

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.

Files to Download

Access Management software is available at https://backstage.forgerock.com. "Access Management Software" describes the files available for download.

Access Management Software

File	Description
`AM-7.1.0.zip`	Cross-platform distribution including all software components. For a list of the files in the .zip archive, see "*Downloading AM*" in the *Installation Guide*.

File	Description
`AM-7.1.0.war`	Deployable web application archive file.
`AM-SSOAdminTools-5.1.3.4.zip`	The .zip file that contains tools to manage AM from the command line.
`AM-SSOConfiguratorTools-5.1.3.4.zip`	The .zip file that contains tools to configure AM from the command line.

Operating System Requirements

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

Supported Host Operating Systems

Operating System	Versions
Red Hat Enterprise Linux, Centos	7, 8
Amazon Linux	Amazon Linux 2018.03
SuSE	12, 15
Ubuntu	18.04 LTS 20.04 LTS
Windows Server	2016, 2019

Web and Java Agents Platform Requirements

The following table summarizes the minimum required version of web and Java agents:

Minimum Agent Version Required

Agent	Versions
Web Agents	5.0.1
Java Agents	5.0.1

AM supports several versions of web agents and Java agents. For supported container versions and other platform requirements related to agents, refer to the *Web Agents Release Notes* and the *Java Agents Release Notes*.

Java Requirements

The following table lists supported Java versions:

Supported Java Versions

Vendor	Versions
OpenJDK, including OpenJDK-based distributions: • AdoptOpenJDK/Eclipse Adoptium • Amazon Corretto • Azul Zulu • Red Hat OpenJDK ForgeRock tests most extensively with AdoptOpenJDK/Eclipse Adoptium. ForgeRock recommends using the HotSpot JVM.	11
Oracle Java	11

Web Application Container Requirements

The following table summarizes supported application containers and their required versions:

Supported Web Application Containers

Container	Versions
Apache Tomcat	8.5, 9
IBM WebSphere Liberty	20.0.0.1
JBoss Enterprise Application Platform	7.3
Wildfly	15, 19

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

Caution

Java Agents and Web Agents require the WebSocket protocol to communicate with AM.

Ensure that the container where AM runs, the web server/container where the agents run, and your network infrastructure all support the WebSocket protocol.

Refer to your network infrastructure and web server/container documentation for more information about WebSocket support.

Directory Server Requirements

This section lists supported directory servers.

As described in Generic LDAPv3 Configuration Properties in the *Setup Guide*, you can configure AM to use LDAPv3-compliant directory servers as user data stores. If you have a special request to deploy AM with a user data store not mentioned in the following table, contact info@forgerock.com.

Supported Data Stores

Directory Server	Versions	Configuration	Apps / Policies	CTS	Identities	UMA
Embedded ForgeRock Directory Services [a]	7.1	✔	✔	✔	✔	✔
External ForgeRock Directory Services	Any ForgeRock-supported version	✔	✔	✔	✔	✔
File system-based	N/A	✔				
Oracle Unified Directory	11g R2				✔	
Oracle Directory Server Enterprise Edition	11g				✔	
Microsoft Active Directory	2016, 2019				✔	
IBM Tivoli Directory Server	6.4				✔	

[a]Demo and test environments only

Third-Party Software

ForgeRock provides support for using the following third-party software when logging ForgeRock Common Audit events:

Software	Version
Java Message Service (JMS)	2.0 API
MySQL JDBC Driver Connector/J	8 (at least 8.0.19)
Splunk	8.0 (at least 8.0.2)

Tip

Elasticsearch and Splunk have native or third-party tools to collect, transform, and route logs. Examples include Logstash and Fluentd.

ForgeRock recommends that you consider these alternatives. These tools have advanced, specialized features focused on getting log data into the target system. They decouple the solution from the ForgeRock Identity Platform systems and version, and provide inherent persistence and reliability. You can configure the tools to avoid losing audit messages if a ForgeRock Identity Platform service goes offline, or delivery issues occur.

These tools can work with ForgeRock Common Audit logging:

- Configure the server to log messages to standard output, and route from there.
- Configure the server to log to files, and use log collection and routing for the log files.

ForgeRock provides support for using the following third-party software when monitoring ForgeRock servers:

Software	Version
Grafana	5 (at least 5.0.2)
Graphite	1
Prometheus	2.0

For hardware security module (HSM) support, ForgeRock software requires a client library that conforms to the PKCS#11 standard v2.20 or later.

Supported Clients

The following table summarizes supported clients and their minimum required versions:

Supported Clients

Client Platform	Native Apps [a]	Chrome 62+	Internet Explorer 11+ [b]	Edge 25+	Firefox 57+	Safari 11+	Mobile Safari
Windows 8 or later	✔	✔	✔	✔	✔		
Mac OS X 10.11 or later	✔	✔			✔	✔	
Ubuntu 14.04 LTS or later	✔	✔			✔		
iOS 9 or later	✔	✔					✔
Android 6 or later	✔	✔					

[a] *Native Apps* is a placeholder to indicate the platform is not limited to browser-based technologies. An example of a native app would be something written to use ForgeRock REST APIs.

[b] Support ends June 15, 2022, in alignment with the announcement from Microsoft ending support for Internet Explorer 11

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 AM 7.1 software.

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

- If you are installing AM for the first time, see the Installation Guide.
- If you have already installed AM, see the Upgrade Guide.

  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.

Chapter 4

Changes to Existing Functionality

This chapter covers both "Critical Changes" and "Important Changes" to existing functionality.

Critical Changes

As part of planning your upgrade, you need to consider that certain changes in later AM versions may have an impact on your environment. Usually, these changes are driven by changes in specification, security policies, or performance.

When possible, the upgrade process makes the appropriate changes on AM configuration. However, sometimes you will need to perform additional configuration based on your environment needs.

In addition to mandatory upgrade steps outlined in "*Upgrading AM Instances*" in the *Upgrade Guide*, if you are using features described in the following table you will need to perform additional upgrade tasks:

Critical Changes to Existing Functionality

AM Version	Component or Feature	Change
7.1	Decompressed JWTs	By default, AM rejects any JWT that expands to more than 32 KiB (32768 bytes) when decompressed. For more information about changing this default value, see "Controlling the Maximum Size of Compressed JWTs" in the *Security Guide*.
	Request Body Size	By default, AM rejects incoming requests with a body larger than 1 MB (1048576 bytes) in size. For more information about changing this default value, see "Limiting the Size of the Request Body" in the *Security Guide*.
	Pre-Approval for Redirection URIs Enforced	This change affects AM when acting as an OAuth 2.0 and OpenID Connect client. If a redirection URI uses a scheme, host, or port that differs from that of AM, you must now add it to the global validation service to ensure that it is pre-approved. This is described in "Configuring Success and Failure Redirection URLs" in the *Authentication and Single Sign-On Guide*. Otherwise, AM rejects the URI, and redirection fails.

AM Version	Component or Feature	Change
	Subject Claim in Access and ID Tokens	The subject claim of access tokens and ID tokens has changed formats to ensure that it is locally unique. The new format is *not enforced after upgrading* to AM 7.1, but new installations default to it. The `org.forgerock.security.oauth2.enforce.sub.claim.uniqueness` in the *Reference* advanced server property controls the format of the `sub` claim. Before enabling it, ensure that your clients can use the new `sub` claim format, or a combination of the `sub` and the `subname` claims.
	The "Retry Limit Decision Node" in the *Authentication and Single Sign-On Guide*	The new Save Retry Limit to User feature in this node is enabled by default after upgrade and requires upgrading the identity store schema. Ensure you update the schema following the instructions in "*Upgrading AM Instances*" in the *Upgrade Guide*, or disable the feature. ForgeRock recommends keeping it enabled for security reasons. Failure to take any of the actions will break the authentication journey for trees using this node.
	One-Time Passwords in Authentication Nodes	One-time passwords created by the "HOTP Generator Node" in the *Authentication and Single Sign-On Guide* are now stored in the authentication tree's transient state in the *Authentication Node Development Guide*. Modify any custom authentication nodes or scripts used by the "Scripted Decision Node" in the *Authentication and Single Sign-On Guide* to retrieve the one-time passwords from the transient state after upgrading to AM 7.1.
7	User Profile Whitelist	The profile attribute whitelist controls the information returned to non-administrative users when accessing `json/user` endpoints. Common profile attributes are whitelisted by default, but you need to add any custom attribute you want your non-administrative users to see. For more information, see "Configuring the User Profile Whitelist" in the *Upgrade Guide*.
	`/json/authenticate` Endpoint	When a client makes a call to the `/json/authenticate` endpoint appending a valid SSO token, AM returns the `tokenId` field **empty** if `HttpOnly` cookies are enabled. For example: `{ "tokenId":"", "successUrl":"/openam/console", "realm":"/alpha" }`

AM Version	Component or Feature	Change
	Secure Authentication Tree State Secret ID	An AES 256-bit key called `directenctest` must be available in the environment during upgrade, but it does not need to be the same key that AM provides on the default keystore. After upgrade, ensure that the `am.authn.trees.transientstate.encryption` secret ID is always mapped to an existing, resolvable secret or key alias. Failure to do so may result in trees not working as expected.
	The Embedded DS	The embedded DS can only be used for single AM instances, for test and demo purposes. Sites are not supported. Sites using embedded DS servers must be migrated to external DS servers before upgrading.
	SAML v2.0 Secrets	AM 7 migrated SAML v2.0 to use secret stores. The upgrade process only creates the secret store files on the AM instance where you ran the upgrade process. For more information, see "Configuring Secret Stores After Upgrade" in the *Upgrade Guide*.
	`goto` and `gotoOnFail` Query Parameter Redirection	Redirection URLs for authentication services, agents, and SAML v.2.0 must be configured in the Validation Service in the *Authentication and Single Sign-On Guide* if they are not in the same scheme, FQDN, and port as AM, or are not relative to AM's URL.
	Web Agents of a Version Earlier than 5.6.3	Several properties that used to be configured as custom properties (`com.sun.identity.agents.config.freeformproperties`) have been added as regular properties. Due to this change, upgrading to AM 7 will overwrite the value of the original custom properties with the default value of the new UI properties. To work around this issue, perform one of the following actions: • Upgrade to Web Agents 5.6.3 or later before upgrading to AM 7. • After upgrading to AM 7, reconfigure the properties that you configured as custom properties in their new UI counterparts.
	Changes on the CTS Reaper Tuning Properties	AM 7 changes the way the CTS reaper searches for expired tokens. After upgrading, retune the CTS Reaper using the information in "Reaper Search Size" in the *Core Token Service Guide (CTS)*.
	OpenID Connect Clients Authenticating with JWTs	OpenID Connect clients authenticating with JWTs must include in the JWT a `jti` claim containing a unique identifier, in line with OpenID Connect Core 1.0 incorporating errata set 1.
	Cookie Filter	AM flags cookies as secure if they come through a connection marked as secure, or if they come through HTTPS. See "Managing the Secure Cookie Filter" in the *Security Guide*.

AM Version	Component or Feature	Change
6.5.0.2 // 6.5.1	OAuth 2.0 Refresh Tokens	AM only issues refresh tokens to clients that have the `refresh token` grant type configured in their client profile. After an upgrade to 6.5 or later using the UI or the **openam-upgrade-tool** .jar file, existing OAuth 2.0 clients are configured to use all grant flows, including the Refresh Token Grant flow. To configure the `refresh token` grant type manually, see "To Configure AM to Issue Refresh Tokens" in the *OAuth 2.0 Guide*.
6.5	Recovery Codes	Recovery Codes are encrypted, and existing codes are no longer displayed to the user. For more information, see "Upgrading Device Recovery Codes" in the *Upgrade Guide*.
	Secret Stores	AM 6.5 introduced secret stores for OAuth 2.0 and the persistent cookie module. The upgrade process only creates the secret store files on the AM instance where you ran the upgrade process. For more information, see "Configuring Secret Stores After Upgrade" in the *Upgrade Guide*.
	External Configuration Store	DS 6.5 introduced setup profiles, which pre-configure instances for different usages, such as CTS or configuration data. The default base DN for a DS configuration store instance (`ou=am-config`) is different than the default used by previous versions of AM (`dc=openam,dc=forgerock,dc=org`). You should not attempt to run multiple instances of AM where the configuration store base DNs do not match. Use the same configuration store base DNs when configuring external DS 6.5+ instances that will be used simultaneously alongside existing DS 6 or earlier configuration store instances. For more information, see "Preparing Configuration Stores" in the *Installation Guide*.
6	`json/` Endpoints	AM's CSRF protection filter requires that either the `X-Requested-With` or the `Accept-API-Version` headers are included on requests to endpoints under the `json` root. For more information, see "Reviewing REST API Versions Before Upgrading" in the *Upgrade Guide*.

Tip

For information on the endpoints deprecated or removed in previous versions, and their current equivalents, see the following Knowledge Base article.

Important Changes

This section lists changes done to existing functionality, services, endpoints, and others in the current release of AM.

AM 7.1

- `AM-SESSION-DESTROYED` no longer logged when a session times out

  In previous AM releases, session timeout triggered two events. This could cause AM to send two logout tokens on a timeout, if an OAuth2 client was registered for back-channel logout notifications on the session. With this change, a session is still destroyed on timeout but this is done as part of the timeout event, and the `AM-SESSION-DESTROYED` activity is not logged.

- The SAML v2.0 IdP Discovery Service Now Requires Configuring a List of Valid Redirection URLs

  The IdP discovery service application now includes a mandatory field to configure valid redirection URLs. Those are, for example, the URLs of the SPs configured in the CoT the discovery service belongs to.

  After upgrading to AM 7.1, you must:

  - Redeploy the IdP discovery application and reconfigure it to include the valid redirection URLs.
  - Configure the valid redirection URLs in the Validation Service of each of the IdPs, *in the Top Level Realm*.

  For more information, see:

  - "*Deploying the IdP Discovery Service*" in the *SAML v2.0 Guide*
  - "To Configure the Validation Service" in the *Authentication and Single Sign-On Guide*

- The Example Remote Consent Service Now Uses Secret Stores

  The Remote Consent Service example has been migrated to use AM's secret store functionality.

  As part of this change, the example Remote Consent Service signing and encryption fields have been removed in the global and realm service configurations. The following secret IDs have been created in their place:

  + *Secret ID Mappings for the OAuth 2.0 Example Remote Consent Service*

The following table shows the secret ID mappings used for the example Remote Consent Service:

Secret ID	Default Alias	Algorithms
am.services.oauth2.remote.consent.response.signing.RSA	rsajwtsigningkey	RS256 RSA (at least 2048 bits)
am.services.oauth2.remote.consent.request.encryption	selfserviceenctest	RSA-OAEP-256 RSA (at least 2048 bits)

For more information, see "To Configure the AM Example Remote Consent Service" in the *OAuth 2.0 Guide*.

If you have the Remote Consent Service example configured before upgrading, the upgrade process will migrate any secret configuration available to global or realm secret stores.

- Changes to the `sub` Claim in Access Tokens and ID Tokens

  The subject claim of access tokens and ID tokens has changed formats to ensure that it is locally unique, as required by the OpenID Connect specification. The new Backchannel logout tokens in the *OpenID Connect 1.0 Guide* also use the new format.

  The subject claim is in the format `(type!subject)`, where:

  - `subject` is the identifier of the user/identity, or the name of the OAuth 2.0/OpenID Connect client that is the subject of the token.
  - `type` can be one of the following:
    - `age`. Specifies that the *subject* is an OAuth 2.0/OpenID Connect-related user-agent or client. For example, an OAuth 2.0 client, a Remote Consent Service agent, and a Web and Java Agent internal client.
    - `usr`. Specifies that the *subject* is a user/identity.

  For example, `(usr!demo)`, or `(age!myOAuth2Client)`.

  Clients using the `sub` claim to determine the identity about which the token asserts information are impacted by this change. To make transitioning to the new format as painless as possible, AM 7.1 also includes the following:

  - A new advanced server property, `org.forgerock.security.oauth2.enforce.sub.claim.uniqueness` in the *Reference*.

    The property controls whether AM should create tokens using the new `sub` claim format or not, and *it is disabled after an upgrade to AM 7.1*, and enabled in new installations.

Tokens using the old `sub` format will still be accepted after the property is enabled. However, earlier versions of AM cannot read tokens with the new format.

- A new claim: `subname`.

  The value of the `subname` claim matches the value of the `sub` claim as it was returned in earlier versions of AM, or as returned in AM 7.1 when the new advanced server property is disabled.

  An example of the value of the `subname` claim is `demo`, or `myOauth2Client`.

  AM adds the `subname` claim to access and logout tokens regardless of the configuration of the new advanced server property. The claim is also available to ID tokens, but it is not included in the `OIDC Claims Script`. Therefore, AM does not add it to ID tokens by default.

Before enabling the advanced server property, ensure that your clients can use the new `sub` claim format, or a combination of the `sub` and the `subname` claims.

- Maximum Size of Decompressed JWTs Enforced

  A number of AM features accept JWTs to receive information. Some examples are:

  - The Remote Consent service, when it receives consent responses.
  - The OAuth 2.0/OpenID Connect authorization service, when:
    - OpenID Connect clients send `request` parameters as an encrypted JWT instead of as HTTP parameters.
    - OpenID Connect clients register dynamically using software statements.
  - The Authentication service, when configured to issue client-based sessions.

  These JWTs that AM receives can be signed and/or encrypted. Sometimes, if they are fairly large, they can also be compressed so that requests reach AM faster.

  Decompressing a JWT makes it expand in size. By default, AM 7.1 rejects any JWT that expands to more than 32 KiB (32768 bytes).

  Before upgrade, ensure that the decompressed JWTs your clients send to AM are smaller than 32 KiB before compression. If not, change the default value to a larger number after upgrade.

  For more information about changing the default value, see "Controlling the Maximum Size of Compressed JWTs" in the *Security Guide*.

- Maximum Size of Request Body Enforced

  Application servers can usually mitigate against DoS attacks that POST large amounts of form data, but AM endpoints may receive large amounts of POST data in different ways, such as in JSON, JWT, or JWK formats.

By default, AM 7.1 rejects incoming requests with a body larger than 1 MB (1048576 bytes) in size, and returns an HTTP 413 error response.

For more information about changing the default value, see "Limiting the Size of the Request Body" in the *Security Guide*.

- Changes to Web and Java Agent Profiles
  - Web Agents
    + *Added Properties*

      - AM Load Balancer Cookie Enabled (`com.forgerock.agents.config.add.amlbcookie`)

    + *Renamed Properties*

      The Agent Profile ID Whitelist property is now Agent Profile ID Allow List.

  - Java Agents
    + *Added Properties*

      - Load Balancer Cookie Enabled (`org.forgerock.agents.load.balancer.cookies.enabled`)
      - Load Balancer Cookie Name (`org.forgerock.agents.load.balancer.cookie.name`)
      - Client IP Validation Mode (`org.forgerock.agents.original.ip.check.mode.map`)
      - Client IP Validation Address Range (`org.forgerock.agents.acceptable.ip.address.map`)
      - Perform Policy Evaluation in User Authenticated Realm (`org.forgerock.agents.user.realm.overrides.policy.evaluation.realm.enabled)`
      - Accept SSO Tokens (`org.forgerock.agents.accept.sso.tokens.enabled`)
      - SSO Cookie Domain List (`org.forgerock.agents.ipdp.cookie.domain.list`)
      - Expired Session Cache Timeout (`org.forgerock.agents.sso.expired.session.cache.ttl.minutes`)
      - Expired Session Cache Max Records (`org.forgerock.agents.expired.session.cache.size`)
      - HTTP 302 Redirects Enabled (`org.forgerock.agents.302.redirects.enabled`)
      - HTTP 302 Redirect Replacement HTTP Code (`org.forgerock.agents.302.redirect.http.status.code`)

- HTTP 302 Redirect Content Type (`org.forgerock.agents.302.redirect.http.content.type`)
- HTTP 302 Redirect Data (`org.forgerock.agents.302.redirect.http.data`)
- HTTP 302 Redirect Not Enforced List (`org.forgerock.agents.302.redirect.ner.list`)
- HTTP 302 Redirect Invert Not Enforced List (`org.forgerock.agents.302.redirect.invert.enabled`)

+ *Renamed Properties*

The CDSSO Secure Enable property is now Transmit Cookies Securely.

+ *Removed Properties*

- Secure Cookies (`org.forgerock.agents.jwt.cookie.secure.enabled`)
- Session Logout Notification (`org.forgerock.agents.session.change.notifications.enabled`)
- Debug Logfile Directory (`com.iplanet.services.debug.directory`)
- Audit Logfile Path (`org.forgerock.agents.local.audit.file.path`)
- Service Resolver Class Name (`org.forgerock.agents.service.resolver.class.name`)

- The OpenID Connect Discovery Endpoint is Now Disabled by Default

  The `/.well-known/webfinger` OpenID Connect discovery endpoint is now disabled by default, and can only be enabled by realm.

  To enable the endpoint for a realm, configure the OAuth2 Provider service on the realm and next, enable the new OIDC Provider Discovery switch. Enabling the endpoint for the realm allows searches for users within that realm only.

  After upgrading to AM 7.1, the endpoint will be enabled on realms that had the OAuth2 Provider service configured. Disable the endpoint on those realms that are not using OpenID Connect discovery.

  For more information about the endpoint, see "OpenID Connect Discovery" in the *OpenID Connect 1.0 Guide*.

- Changes to the OAuth 2.0 and OpenID Connect Clients

  AM 7.1 returns an error if the administrator tries to save a client configuration containing an unsupported signing or encryption algorithm.

For example, upon saving the configuration, AM will return an error if there is a typo on an algorithm, or a symmetric signing or encryption algorithm is configured on a public client: these algorithms are derived from the client's secret, which public clients do not have.

Clients registering dynamically must also send supported algorithms as part of their configuration, or AM will reject the registration request.

Different features support different algorithms. Refer to the documentation or to the UI for more information.

The following are examples of the errors:

- `Unknown encryption algorithm configured for User info encrypted response algorithm`
- `Symmetric encryption algorithm configured for ID Token Encryption Algorithm is not allowed for a public client`

The error messages are also logged at ERROR level, and identify the client ID that the error relates to.

- One-Time Passwords Now Stored in Transient State

  One-time passwords created by the "HOTP Generator Node" in the *Authentication and Single Sign-On Guide* are now stored in the authentication tree's transient state, instead of in the shared state.

  Modify any custom authentication nodes or scripts used by the "Scripted Decision Node" in the *Authentication and Single Sign-On Guide* to retrieve the one-time passwords from the transient state after upgrading to AM 7.1.

  For more information about the transient state, see "Storing Secret Values in Transient Tree State and the Secure State" in the *Authentication Node Development Guide*.

Chapter 5

Deprecated Functionality

Functionality listed under this section has been deprecated and will be removed in a future release of AM.

AM 7.1

- Deprecated Elasticsearch and Splunk Audit Handlers

  Using the Elasticsearch and Splunk audit handlers is deprecated.

  AM 7.1 supports both file-based audit handlers and logging to standard output, which Elasticsearch and Splunk can consume.

  For more details, see "Configuring Audit Event Handlers" in the *Security Guide*.

- Deprecated isAlive JSP Page

  Using the `isAlive.jsp` to determine if an instance is alive is deprecated.

  AM 7.1 includes new endpoints to determine if an instance is alive, and ready to process requests.

  For more details, see "*Monitoring Instances*" in the *Maintenance Guide*.

- Deprecated Existing `getIDPAuthnContextInfo` Signature

  The existing signature for the `getIDPAuthnContextInfo` method of the `IDPAuthnContextMapper` interface is deprecated.

  AM 7.1 includes a new signature for the getIDPAuthnContextInfo method, which includes an additional parameter for the entity ID of the service provider (SP).

  Note that the deprecated method still works in AM 7.1, but you should update any code that uses it to the new four-parameter signature. The deprecated three-parameter signature will be removed in a future version of AM.

- Deprecated Social Authentication Nodes

  The following authentication nodes have been deprecated in favor of the "Social Provider Handler Node" in the *Authentication and Single Sign-On Guide*:

  - "OpenID Connect Node" in the *Authentication and Single Sign-On Guide*

- "OAuth 2.0 Node" in the *Authentication and Single Sign-On Guide*
- "Social Facebook Node" in the *Authentication and Single Sign-On Guide*
- "Social Google Node" in the *Authentication and Single Sign-On Guide*
- "Provision IDM Account Node" in the *Authentication and Single Sign-On Guide*
- "Create Password Node" in the *Authentication and Single Sign-On Guide*
- "Social Ignore Profile Node" in the *Authentication and Single Sign-On Guide*

As part of this change, the Social Authentication Implementations Service is also deprecated.

For more information about using the "Social Provider Handler Node" in the *Authentication and Single Sign-On Guide*, see "*Social Authentication*" in the *Authentication and Single Sign-On Guide*.

- The ssoadm, ampassword, configurator.jar and upgrade.jar Tools Remain Deprecated

  The **ssoadm** command and the `configurator.jar`, `upgrade.jar`, and `ampassword` tools remain deprecated. They will be removed in a future release of AM.

- Deprecated Direct Access to the Transient, Secure, and Shared State of Authentication Trees

  Direct access to authentication trees' transient, secure, and shared states using the TreeContext class has been deprecated.

  As part of this change:

  - Use of the `sharedState` and the `transientState` bindings for reading and updating state with the "Scripted Decision Node API Functionality" in the *Authentication and Single Sign-On Guide* are deprecated.

    Use the `nodeState` binding instead.

  - Use of the `getState` method from the TreeContext class, used to read state in authentication nodes, is deprecated.

    Use the `getStateFor` method instead.

  For more information, see "Storing Secret Values in Transient Tree State and the Secure State" in the *Authentication Node Development Guide* and "Accessing Shared State Data" in the *Authentication and Single Sign-On Guide*.

Chapter 6
Removed Functionality

Functionality listed under this section has been removed from AM.

AM 7.1

- No features or functionality have been removed in this release.

Chapter 7

Fixes, Limitations, and Known Issues

This chapter covers the status of key issues and limitations at release 7.1.

Key Fixes

AM 7.1

- OPENAM-17396: Terms of Service URI Link does not Display in Consent Page
- OPENAM-17395: SocialOpenIdConnectNode fails to recover from client's connection reset
- OPENAM-17365: Checking agent type with caller token can cause deadlock
- OPENAM-17364: Prompt login / session upgrade / OIDC ACR looping with trees
- OPENAM-17361: API Explorer Swagger Template body needs modified to include configExport, debugLogs and threadDump as per the API Documentation
- OPENAM-17357: Remote Consent Service RCS does follow RCS consented scope when authorization endpoint accessed without any scope
- OPENAM-17353: HTML pages are not picked up when placing in a theme folder
- OPENAM-17349: OIDC Refresh token - Ops token is deleted from the CTS during refresh
- OPENAM-17343: Access token call returns 500 error if password needs to be changed or has expired
- OPENAM-17322: SAML2 bearer grant returns NoUserExistsException
- OPENAM-17317: A realm without any modules can cause increased thread count and slow response.
- OPENAM-17276: AM recorder does not record anymore
- OPENAM-17271: Typo for Realm in SAML/Federation debug
- OPENAM-17260: Allow arg=newsession usage in authorize calls
- OPENAM-17242: OAuth2 Policy - Environment Condition AuthLevel >= doesn't work for ROPC grant

- OPENAM-17220: OAuthLogout.jsp compilation error isGotoUrlValid method signature not found
- OPENAM-17199: Insufficient debug logging for 'DJLDAPv3Repo.getAssignedServices'
- OPENAM-17156: Adaptive Risk checkGeoLocation null countryCode can cause module fail.
- OPENAM-17136: OAuth2 Dynamic Client Registration does not recognise recognised spec defined parameters
- OPENAM-17121: Inefficient synchronized block in OAuth2ProviderSettingsFactory
- OPENAM-17114: Save Consent check box always shown, even when not configured
- OPENAM-17097: Inconsistent scope policy evaluation between authorize and ROPC
- OPENAM-17089: Forgot password functionality broken
- OPENAM-17070: SAML2 SP intiated SSO with AM as idp Proxy, RelayState is not returned from proxy after idp authentication
- OPENAM-17060: Audit Logging "Resolve host name" is still available after OPENAM-7849
- OPENAM-17037: AM Upgrade from 6.0.0.7 to 7.0.0 causing NPE
- OPENAM-17034: In a realm if User Profile is set to Ignored the realm level Session Service quota settings is also ignored and only the Session Service setting at top level/global is evaluated
- OPENAM-17017: REST STS fails with unable get get sub-schema if cache is refreshed while updating REST config
- OPENAM-17006: Hosted SAML entity - can not remove bindings
- OPENAM-16998: Poor logging around failures "Invalid Assertion Consumer Location specified"
- OPENAM-16997: Device code grant implied consent fails if access_token request performed before user authenticates
- OPENAM-16988: Accessed endpoint including port causes verify Assertion Consumer URL to fail
- OPENAM-16955: When setCookieToAllDomains=false is used, a non matching request from other domain will fail
- OPENAM-16947: Kerberos Node in 7.0 fails to return goTo(false)
- OPENAM-16944: Regression in OPENAM-15649. LdapDecisionNodes fails if inetuserstatus does not exist
- OPENAM-16936: Tree nodes create new keystore object each time node is called.
- OPENAM-16935: Logout issue after logging into AM with 'Remember my username' selected with iOS 14.0.1

- OPENAM-16934: sm.getSchemaManager has a typo including a comma
- OPENAM-16926: Success URL node doesn't work with SAML Node for Idpinit when not using Integrated mode
- OPENAM-16910: Can not create SAML entity with entity id including a semicolon ';'
- OPENAM-16907: Kerberos Node in 7.0 does not work
- OPENAM-16904: OIDC bearer module fails with NPE when id_token does not contain kid
- OPENAM-16883: AM ignores AuthnRequestsSigned property during SSO
- OPENAM-16876: Default ACR values on OIDC client profile is not honoured in order of preference
- OPENAM-16866: AM should fail gracefully if id_token fails to generate when swapping refresh token
- OPENAM-16849: WeChat Social Auth module broken (regression)
- OPENAM-16848: Choice Collector and WDSSO node combination does not work if whitelisting is enabled
- OPENAM-16847: AM email service failing with 'Start TLS' option
- OPENAM-16838: AuthenticationApproachChecker does not handle session upgrade modules
- OPENAM-16823: IDM Nodes does not send or propagate transactionId tracking when contacting IDM
- OPENAM-16807: The dynamic values for request_uri being stored in client config does not expire and is not automatically removed
- OPENAM-16801: SAML2 SP init SSO fails after upgrade to 7.0.0
- OPENAM-16784: Upgrade to 7 fails with NullPointerException in Saml2EntitySecretsStep
- OPENAM-16769: Enabling Auto-federation when User Profile is Dynamic on SP causes SP to hang during SAML flow
- OPENAM-16758: Cannot install AM 7 on Windows
- OPENAM-16745: client_id in access token ignores what's been registered when idm cache is disabled
- OPENAM-16726: Insufficient debug logging for OAuth2 error 'invalid_client Server does not support this client's subject type'
- OPENAM-16703: OAuth2 Access token obtained from refresh token is certificate-bound regardless of "Certificate-Bound Access Tokens" configuration (when client_secret_basic used for credentials)

- OPENAM-16701: The authorize endpoint with a service parameter will cause the parameter to appear as a PAP claim in the agent's ID token
- OPENAM-16684: OIDC Dynamic Registration client_description cannot take String type
- OPENAM-16669: IdentityGateway Agent entry missing attribute required to support org.forgerock.openam.agent.TokenRestrictionResolver#getAgentInfo
- OPENAM-16617: SuccessURL session property is set to gotoURL in authentication tree
- OPENAM-16608: AM with embedded DS setup fails with permission denied for truststore
- OPENAM-16583: Crucial information is missing when encountering LDAP connections issue.
- OPENAM-16556: Radius Server's does not log IP address into AM Audit logs
- OPENAM-16555: Audit logging does not tell which policy allowed or denied a resource request
- OPENAM-16540: Issues with Social Login URLs when navigating quickly between providers
- OPENAM-16535: "JWKs URI content cache miss cache time" is not triggered when "kid" is missing from cached JWK Set
- OPENAM-16515: Social auth - insufficient debug logging for troubleshooting
- OPENAM-16485: 'Failed Login URL' is not picked up from the auth chain
- OPENAM-16472: Proxied Authentication fallback may not work when user entry lack some attributes
- OPENAM-16450: 501 when default resource version set to "oldest" and Accept-API-Version header set
- OPENAM-16418: private_key_jwt client auth fails with 500 if claim format is wrong
- OPENAM-16368: Settings of Mail and Scripting global service properties are overwritten at upgrade
- OPENAM-16367: OIDC request_uri response causes NPE while debug logging
- OPENAM-16354: Concurrency bug in OAuth2ProviderSettingsFactory
- OPENAM-16338: Failing REQUISITE module after SUFFICIENT Device Match doesn't fail chain properly
- OPENAM-16157: Session Property Whitelist Service allows case variant Property Names but DS is not case sensitive
- OPENAM-16152: After upgrade, new Identity page has duplicate 'new identity' field and email address does not save

- OPENAM-16006: Device Code Grant does not work with Implied Consent as Authorization is not approved even after consented
- OPENAM-15963: Historical retention files (csv) were not deleted
- OPENAM-15948: Update DS profiles to add VLV indexes for CTS use
- OPENAM-15743: Excessive CTS logging when Reaper is disabled (com.sun.am.ldap.connnection.idle.seconds=0)
- OPENAM-15671: LoginContext is missing debug logging for troubleshooting
- OPENAM-15663: UserInfoClaims is not part of public API
- OPENAM-14898: OTP Email Sender Authentication Node fails if no SMTP authentication credentials are specified
- OPENAM-14682: Microsoft Social Auth fails when creating an Microsoft account (Legacy OAuth2)
- OPENAM-14527: Microsoft Social Auth does not work with latest MS endpoints (Legacy OAuth2)
- OPENAM-12503: SizeBasedRotationPolicy does not delete oldest file

Limitations

The following limitations and workarounds apply to AM 7.1:

- Evaluation Installation Limitations

  In some cases, installing AM for evaluation purposes will fail with a message similar to the following if the JDK's default truststore's permissions are 444:

  ```
  $JAVA_HOME/lib/security/cacerts (Permission denied), refer to install.log under /usr/share/tomcat/
  access/var/install.log for more information.
  ```

  To work around this issue, locate the truststore that your container is using and change its permissions to 644 before installing AM:

  ```
  $ sudo chmod 644 $JAVA_HOME/lib/security/cacerts
  ```

  You can change the permissions back as they were originally after installing AM.

- Identity and Data Store Scaling Limitations

  The connection strings to the data or identity stores are static and not hot-swappable. This means that, if you expand or contract your DS affinity deployment, AM will not detect the change.

  To work around this, either:

  - Manually add or remove the instances from the connection string and restart AM or the container where it runs.

  - Configure a DS proxy in front of the DS instances to distribute data across multiple DS *shards*, and configure the proxy's URL in the connection string.

- SAML v2.0 UI Limitations

  The new UI supports SAML v2.0 IDP and SP entities only. After upgrade, entities that do not have IDP or SP roles will be listed, but cannot be inspected or edited using the UI. An error will display in the UI when trying to access these entities.

  Entities containing roles other than IDP and/or SP will only display the IDP and/or SP roles.

- Web Authentication (WebAuthn) Limitations

  AM 7.1 does not support the following functionality as described in the Web Authentication specification:

  Registration

  - Token Binding is not supported.
  - Web Authentication extensions are not supported.
  - Credential ID values are not verified against the credential IDs registered with all existing users.
  - The ECDAA signature of the Packed attestation format is not supported.

  Authentication

  - Token Binding is not supported.
  - Web Authentication extensions are not supported.
  - Signature counters are not supported.

  For more information about Web Authentication, see "*MFA: Web Authentication (WebAuthn)*" in the *Authentication and Single Sign-On Guide*.

- **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 `Realm Admin` privilege**

  In this version of AM, administrators can use the AM console as follows:

  - Delegated administrators with the `Realm Admin` privilege can access full AM console functionality within the realms they can administer. In addition, delegated administrators in the Top Level Realm who have this privilege can access AM's global configuration.

- Administrators with lesser privileges, such as the `Policy Admin` privilege, can not access the AM console.
- The top-level administrator, such as `amAdmin`, has access to full AM console functionality in all realms and can access AM's global configuration.

- Specifying Keys in JWT Headers is Not Supported

  AM ignores keys specified in JWT headers, such as `jku` and `jwe`. Configure the public keys/certificates in AM instead, as explained in the relevant sections of the documentation.

- **Different AM Versions Within a Site Are Not Supported**

  Do not run different versions of AM together in the same AM site.

- **Use of Special Characters in Policy or Application Creation is Not Supported**

  Do not use special characters within policy, application or referral names (for example, "my +referral") using the Policy Editor or REST endpoints as AM 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)

- **XACML Policy Import and Export from Different Vendors is Not Supported**

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

- **JCEKS Keystore Now Required for User Self-Services**

  In OpenAM 13.0.0, the 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 AM 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 and later solve 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.

  For specific instructions to configure the JCEKS keystore, see "Managing the AM Keystore" in the *Security Guide*.

  Note

  This procedure is not necessary for the following users:

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

Known Issues

The following important known issues remained open at the time release 7.1 became available. For details and information on other issues, see the issue tracker.

AM 7.1

- Licensing information for some third-party libraries is missing from the `legal-notices/third-party-copyrights.txt` file, available in the AM-7.1.0.zip file.

  + *More Information...*

  The following table matches the libraries with their corresponding license:

Library	License
`geronimo-jta_1.1_spec-1.1.1.jar`	Apache 2.0
`geronimo-ws-metadata_2.0_spec-1.1.3.jar`	Apache 2.0
`jacorb-omgapi-3.9.jar`	LGPL 2.1
`jakarta.activation-api-1.2.1.jar`	BSD 3
`jakarta.xml.bind-api-2.3.2.jar`	BSD 3
`javax.activation-1.2.0.jar`	CDDL 1.1
`javax.annotation-api-1.3.2.jar`	CDDL 1.1
`javax.xml.soap-api-1.4.0.jar`	CDDL 1.0
`jaxb-impl-2.3.0.jar`	CDDL 1.1
`jaxb-runtime-2.3.0.jar`	CDDL 1.1
`jboss-rmi-api_1.0_spec-1.0.6.Final.jar`	LGPL 2.1

  Find the license files in the `legal-notices/third-party-licenses` directory, available in the AM-7.1.0.zip file.

- OPENAM-16418: private_key_jwt client auth fails with 500 if claim format is wrong
- OPENAM-16449: Filter fields on the Scripts admin page don't work

- OPENAM-17045: Failing SAML2 flows on ForgeOps environments
- OPENAM-17315: Update defaults scripts with the change introduced in COMMONS-628
- OPENAM-17351: AM File based config setup cannot be used with AM recording to dump the config dump
- OPENAM-17418: OpenId account mapping fails because userInfo subject claim has value 'usr!demo'
- OPENAM-17590: OIDC login hint cookie broken
- OPENAM-17687: XUI select wrong partials if a new Partial happens to exists with same prefix
- OPENAM-17760: PEM support incorrectly decodes some EC private keys
- OPENAM-17768: Enabling whitelisting in trees causes an infinite redirect loop in the registration tree - forgeops

Chapter 8
Documentation Updates

The following table tracks changes to the documentation set following the release of AM 7.1:

Documentation Change Log

Date	Description
2021-11-15	Added a change in behavior to the logging on session timeout.
2021-05-12	Initial release of AM 7.1.

Appendix A. Release Levels and Stability Labels

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

ForgeRock Product Release Levels

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

Release Level Definitions

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

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

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.

Stability Label	Definition
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.