

Release notes

IG integrates web applications, APIs, and microservices with the ForgeRock Identity Platform, without modifying the application or the container where they run. Based on reverse proxy architecture, IG enforces security and access control in conjunction with Access Management modules.



What's New

[Discover new features.](#)



Requirements

[Check IG prerequisites.](#)



Compatibility

[Review incompatible changes.](#)



Fixes

[Review bug fixes, limitations, known issues.](#)



Documentation

[Track documentation changes.](#)



Support

[Get support and training.](#)

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

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

What's new

Token exchange

▼ [Token exchange filter](#)

[OAuth2TokenExchangeFilter](#) is a new filter to exchange a client's access token or ID token for a new token with increased or reduced scopes, while preserving the original token subject

Connectivity with OAuth 2.0-protected third-party services

▼ [OAuth2ClientFilter renamed as AuthorizationCodeOAuth2ClientFilter.](#)

IG provides several client authentication filters, which protect resources by using different types of information and credentials. To make it easier to differentiate between these filters, the [OAuth2ClientFilter](#) has been renamed as [AuthorizationCodeOAuth2ClientFilter](#). For backward compatibility, the name [OAuth2ClientFilter](#) can still be used in routes.

The following client authentication filters are available to authenticate clients:

- [AuthorizationCodeOAuth2ClientFilter](#), using OAuth 2.0 delegated authorization
- [ClientCredentialsOAuth2ClientFilter](#), using the client's OAuth 2.0 credentials
- [ResourceOwnerOAuth2ClientFilter](#), using the resource owner's password credentials

▼ [ClientCredentialsOAuth2ClientFilter uses `client_secret_basic` or `client_secret_post`](#)

The [ClientCredentialsOAuth2ClientFilter](#) can now obtain a client's access token, using the token endpoint authentication method `client_secret_post`. In previous releases, it could use only `client_secret_basic`.

Client authentication is now provided by the `endpointHandler` property of [ClientCredentialsOAuth2ClientFilter](#), which uses [ClientSecretBasicAuthenticationFilter](#) or [ClientSecretPostAuthenticationFilter](#). In previous releases, it was provided by the now deprecated properties `clientId` and `clientSecretId`.

▼ [ResourceOwnerOAuth2ClientFilter for services to access resources protected by OAuth 2.0.](#)

A new filter [ResourceOwnerOAuth2ClientFilter](#) is available for services to access resources protected by OAuth 2.0, using the *Resource Owner Password Credentials* grant type. For an example of use, see [Using OAuth 2.0 resource owner password credentials](#).

▼ [Filters to support OAuth 2.0 client authentication](#)

When processing requests or responses, IG can require access to systems such as the Identity Cloud to query user information. The following filters have been added to facilitate OAuth 2.0 client authentication to these systems, where IG is the client:

- [ClientSecretBasicAuthenticationFilter](#)
- [ClientSecretPostAuthenticationFilter](#)
- [EncryptedPrivateKeyJwtClientAuthenticationFilter](#)
- [PrivateKeyJwtClientAuthenticationFilter](#)

Use these filters with the following objects:

- [ClientRegistration](#)
- [AuthorizationCodeOAuth2ClientFilter](#)
- [OAuth2TokenExchangeFilter](#)
- [ClientCredentialsOAuth2ClientFilter](#)
- [ResourceOwnerOAuth2ClientFilter](#)

▼ [OAuth 2.0 session sharing across routes](#)

The property `oAuth2SessionKey` has been added to [AuthorizationCodeOAuth2ClientFilter](#) to allow multiple applications to share the same OAuth 2.0 session.

After a resource owner gives one application protected by IG consent to use its data, they don't need to give consent for another application protected by IG.

In previous releases, the OAuth 2.0 session was bound to the full URI of the client callback, containing the IG hostname. So it was not possible to use the same OAuth 2.0 session to access different applications.

Circuit breaking

▼ [CircuitBreakerFilter](#)

[CircuitBreakerFilter](#) is a new filter to monitor for failures. When the failures reach a specified threshold, the [CircuitBreakerFilter](#) prevents further calls to downstream filters and returns a runtime exception.

▼ [Circuit breaker in ClientHandler and ReverseProxyHandler](#)

OPENIG-6517 A new property `circuitBreaker` has been added to [ClientHandler](#) and [ReverseProxyHandler](#) to provide a circuit breaker service when the number of failures reaches a configured threshold.

Stability

▼ [JwtBuilderFilter produces encrypted JWT](#)

The `JwtBuilderFilter` now produces encrypted JWTs, in addition to unsigned JWTs, signed JWTs, and signed then encrypted JWTs.

▼ [JwtSession cookie compression](#)

The property `useCompression` has been added to [JwtSession](#). When a session stores large items, such as tokens, use the default value `true` to reduce size of the cookie that stores the JWT.

Other

▼ [Windows start script for IG in standalone mode](#)

A script is now provided to start IG in standalone mode on Windows. For information, see [Install IG in standalone mode](#).

▼ [Stop scripts for IG in standalone mode](#)

Scripts are now provided to stop IG in standalone mode, on Unix/OS X and Windows. For information, see [Install IG in standalone mode](#).

▼ [IG_OPTS environment variables for startup](#)

`IG_OPTS` is a new environment variable to separate Java runtime options for IG startup and stop scripts with IG in standalone mode. Use `IG_OPTS` instead of `JAVA_OPTS` for all options that are not shared with the stop script.

For more information, see [Define environment variables for startup, runtime, and stop](#).

▼ [Support for samesite cookies in standalone mode](#)

`sameSite` is a new subproperty of `session` in [admin.json](#), to manage the circumstances in which a cookie is sent to the server. Use this property to reduce the risk of cross-site request forgery (CSRF) attacks when IG is in standalone mode.

▼ [SNI to serve different certificates for TLS Connections to different server names](#)

In [ServerTlsOptions](#), `sni` is a new property to serve different secret key and certificate pairs for TLS connections to different server names in the deployment. In previous releases, only the `keyManager` property was available to serve the same secret key and certificate pair for TLS connections to all server names.

Use this property when IG is acting server-side, to front multiple services or websites on the same port of a machine.

For an example, see [Serve different certificates for TLS connections to different server names](#).

▼ [Vert.x metrics](#)

Vert.x metrics are now available by default for IG in standalone mode, to provide metrics for HTTP, TCP, and the internal component pool. The metrics provide low-level information about requests and responses, such as the number of bytes, duration, the number of concurrent requests, and so on.

Metrics are provided at the Prometheus Scrape Endpoint and Common REST Monitoring Endpoint endpoints.

For more information, see the `vertx` object in [AdminHttpApplication\(`admin.json` \)](#), and [Monitoring Vert.x metrics](#).

▼ [IG proxies all WebSocket subprotocols by default](#)

In previous releases, for IG in standalone mode it was necessary to list the WebSocket subprotocols that were proxied by IG, with the `vertx` property of [admin.json](#).

From this release, IG proxies all WebSocket subprotocols by default; it is not necessary to specify protocols. If you do specify protocols, IG supports only those protocols and no others.

▼ [Configurable conditions for retries in ClientHandler and ReverseProxyHandler](#)

`condition` is a new property in the `retries` configuration of `ClientHandler` and `ReverseProxyHandler`. Use this property to configure a condition on which to trigger a retry. In previous releases, a retry could be triggered only for runtime exceptions.

▼ [User ID in audit logs](#)

Audit logs can now include a user ID. Example scripts and setup information is provided in [Recording user ID in audit events](#).

▼ [Tracking ID logged in access audit events](#)

In routes containing an `OAuth2ResourceServerFilter`, OAuth 2.0 token tracking IDs are now logged in access audit events.

▼ [Transformation from string to placeholder string](#)

The `$string` transformation has been added to facilitate the transformation from a string to a placeholder string, which is not encoded. Use this transformation for placeholder strings that must not be encrypted, for example, when they reference a secret value.

For more information, see [string](#) in *Token Transformation*.

▼ [Use expressions to configure paths in UriPathRewriteFilter](#)

The `mapping` object in `UriPathRewriteFilter` now uses configuration expressions to define the `fromPath` and `toPath`. In previous releases, the `mapping` object was a static JSON map.

For more information, see [UriPathRewriteFilter](#).

▼ [New EL functions for better pattern matching](#)

The functions `find` and `matchesWithRegex` are added to use as replacements for the deprecated function `matches`.

The function `findGroups` is added to use as a replacement for the deprecated function `matchingGroups`.

For more information, see [Functions](#).

▼ [Additional logging for a BadRequestException during policy evaluation requests](#)

To help with troubleshooting, a debug message is now also logged when a `BadRequestException` occurs during policy evaluation requests. In previous releases, the original error was not logged, IG just returned an HTTP 401 Unauthorized.

▼ [PolicyDecisionContext includes actions from the policy decision response](#)

Actions from the AM policy decision response are now available in the `PolicyDecisionContext`, and available for use.

The resource value that was used when making the policy request is now available in `PolicyDecisionContext`.

[PolicyDecisionContext](#).

▼ [AmService detects AM version](#)

[AmService](#) now reads the AM version from the AM endpoint, and uses the discovered version instead of the value configured in the `AmService` property `version`.

The property `version` is used only if `AmService` cannot discover the AM version.

▼ [Certificate issued by a trusted CA for any hostname or domain is accepted for a connection to any domain](#)

When IG is acting as a WebSocket proxy, and the downstream application is on HTTPS, the WebSocket configuration host can now allow a certificate issued by a trusted CA for any hostname or domain to be accepted for a connection to any domain. For information, see the `hostnameVerifier` property of [ClientTlsOptions](#).

▼ [Product information in startup logs](#)

Key product information, such as the product version and build number, is now included in the startup logs.

▼ [Improved error handling in ScriptableFilter and ScriptableHandler](#)

The ScriptableFilter and ScriptableHandler now propagate script exceptions as runtime exceptions in the promise flow. In previous releases, they replaced the exception with a response, with HTTP status 500. Users didn't know if the response was from the requested endpoint or caused by an exception in the chain.

▼ [AmService Websocket connections protected from timeout](#)

A heartbeat can be configured on the [AmService](#) WebSocket notification service to prevent Websocket connections from being closed for timeout.

▼ [Timeout of idle AM sessions](#)

A new filter [AmSessionIdleTimeoutFilter](#) is available to force the revocation of AM sessions that have been idle for a specified timeouot.

Use this filter in front of a SingleSignOnFilter or CrossDomainSingleSignOnFilter, to manage idle timeout for client sessions in AM.

▼ [Proxy configuration can be created in the heap and used for AM notifications](#)

A new [ProxyOptions](#) heaplet is available to define a proxy to which a [ClientHandler](#) or [ReverseProxyHandler](#) can submit requests, and an [AmService](#) can submit Websocket notifications.


A new global ProxyOption heap object is provided.

Requirements

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.

Downloads

Download the following product software from the [ForgeRock BackStage download site](#) :

- IG-7.2.0.zip : For deployment in standalone mode
- IG-7.2.0.war : For deployment in web container mode
- IG-sample-application-7.2.0.jar : Web application for testing IG configurations

For information about using the Docker image provided with the product software, see the [Deployment guide](#).

Operating systems

IG is tested on Windows Server 2016, Windows Server 2019, and Linux operating systems.

Web application containers

In web container mode, IG runs in the following containers:

- Apache Tomcat 9

NOTE

Apache Tomcat 10 is not supported. Use IG in standalone mode as an alternative.

- Jetty 9, 10

NOTE

Jetty 11 is not supported. Use IG in standalone mode as an alternative.

- JBoss EAP 7.3.2, 7.4

Deploy IG to the root context of a container. Deployment in other contexts causes unexpected results, and is not supported.

Java

IG supports the following Java environments:

Supported Java versions

Vendor	Versions
--------	----------

Vendor	Versions
<p>OpenJDK, including OpenJDK-based distributions:</p> <ul style="list-style-type: none"> • AdoptOpenJDK/Eclipse Temurin Java Development Kit (Adoptium) • Amazon Corretto • Azul Zulu • Red Hat OpenJDK <p>ForgeRock tests most extensively with AdoptOpenJDK/Eclipse Temurin.</p> <p>ForgeRock recommends using the HotSpot JVM.</p>	11
Oracle Java	11

Always use a JVM with the latest security fixes.

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

Java 11 is the only long-term supported (LTS) Java version for most ForgeRock products. Earlier versions of Java do not contain required cryptography fixes. If you are using an earlier version of Java, secure your installation.

HTTP protocol

IG supports HTTP/1.1 and HTTP/2.0.

HTTP/1.0 is not supported.

FQDNs

IG replication requires use of fully qualified domain names (FQDNs), such as `ig.example.com`.


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

Certificates

For secure network communications with client applications that you do not control, install a properly signed digital certificate that your client applications recognize, such as one that works with your organization's PKI, or one signed by a recognized CA.

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

Third-party software for encryption



Bouncy Castle is required for signature encryption with RSASSA-PSS or Deterministic ECDSA. For information, see [The Legion of the Bouncy Castle](#) .

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.

Studio browser

ForgeRock has tested many browsers with Studio, including:

- Chrome, latest stable version
- Firefox, latest stable version

Features requiring later versions of ForgeRock Access Management

Feature	Requires
OAuth2TokenExchangeFilter	From AM 7.1
Support for refresh of idle sessions when the <code>SingleSignOnFilter</code> is used for authentication with AM. For more information, see the <code>sessionIdleRefresh</code> property of AmService .	From AM 6.5.3
Eviction of revoked OAuth 2.0 access tokens from the cache. For more information, see CacheAccessTokenResolver , and the <code>cache</code> property of OAuth2ResourceServerFilter .	From AM 6.5.3

Feature	Requires
Support for OAuth 2.0 Mutual TLS (mTLS). For more information, see ConfirmationKeyVerifierAccessTokenResolver , and Validate Certificate-Bound Access Tokens .	From AM 6.5.1

Incompatible changes

The following changes introduced in IG 7.2 can impact your migration from IG 7.1:

▼ [ScriptableResourceUriProvider accepts returned values only as a String](#)

`ScriptableResourceUriProvider` accepts returned values only as a `String`. In previous releases, it accepted returned values as a `String` or `Promise<String>`. For more information, see `ScriptableResourceUriProvider` in [PolicyEnforcementFilter](#).

▼ [Logback upgrade](#)

IG has upgraded the version of Logback, used for the logging framework. The Logback update introduces changes that can affect your existing deployment. For more information about changes in Logback, see the [Logback website](#).

▼ [AM 5.x.x EOL](#)

AM 5.x.x has reached Product End of Life and is no longer supported. The default value of the `AmService` property `version` has changed to `6`. For more information, refer to [Product Support Lifecycle Policy | PingGateway and Agents](#).

▼ [keyType for CapturedUserPasswordFilter is required](#)

For better security, the `keyType` for `CapturedUserPasswordFilter` is now required, and the use of `DES` is deprecated.

▼ [JWT classes relocated to new packages](#)

Classes related to JWT stateless sessions have moved from the package `org.forgerock.openig.jwt` to `org.forgerock.openig.session.jwt`.

Classes and functions used to validate a JWT, used with a `JwtValidatorCustomizer` in a `JwtValidationFilter`, have moved from the package `org.forgerock.openig.tools.jwt` to `org.forgerock.openig.tools.jwt.validation`.

The IG scripting engine has been updated to incorporate the changes automatically.

▼ [CDSSO requires session cookies with SameSite=None, Secure=True](#)

To improve privacy, browsers have recently changed third-party cookie policies to require the following settings for session cookies: `SameSite=None` , `Secure=True` .

Depending on your deployment and route configuration, configure session cookies as follows:

- For stateful sessions in standalone mode, by [admin.json](#)
- For stateful sessions in web container mode, by the web container:
 - For Tomcat, see [Configure SameSite for HTTP session cookies in Tomcat](#), and [Configure IG for HTTPS \(server-side\) in Tomcat](#).
 - For Jetty, see [Configure SameSite for HTTP session cookies in Jetty](#) and [Configure IG for HTTPS \(server-side\) in Jetty](#).
 - For JBoss, see [Configure SameSite for HTTP session cookies in JBoss](#) and [Configure IG for HTTPS \(server-side\) in JBoss](#).
- For stateless sessions in standalone mode and web container mode, by [JwtSession](#).

Deprecation

The following features and properties are deprecated, as defined in [ForgeRock Product Stability Labels](#), and likely to be removed in a future release:

▼ [Delivery of IG war file](#)

The delivery of a .war file was deprecated in IG 7.

▼ [Methods to read or set query and form parameters](#)

The `request.form` method used in scripts to read or set query and form parameters is deprecated. Use the following methods instead:

- `Request.getQueryParams()` to read query parameters.
- `Entity.getForm()` to read form parameters.
- `Entity.setForm()` to set form parameters.

▼ [LdapClient class and 'ldap' script binding](#)

The `LdapClient` class and the `ldap` script binding were deprecated in IG 7.1.

▼ [JwtBuilderFilter with unsigned, unencrypted JWTs](#)

The use of `JwtBuilderFilter` with unsigned, unencrypted JWTs was deprecated in IG 7.

Object	Deprecat ed in IG	Deprecated settings	Replacement settings
AmService	6.5	password	Replaced by passwordSecretId . If the deprecated and replacement properties are both provided, the replacement property takes precedence.
AuditService	7	event-handlers	Replaced by eventHandlers .
OAuth2ClientFilter	7.2	Filter name	Replaced by AuthorizationCodeOAuth2ClientFilter. For backward compatibility, the name OAuth2ClientFilter can still be used in routes.
ClientCredentialsOAuth2ClientFilter	7.2	clientId , clientSecretId , handler If you use the deprecated properties, provide clientId , clientSecretId to obtain the client secret, which authenticates using the client_secret_basic method.	Replaced by endpointHandler , which uses ClientSecretBasicAuthenticationFilter or ClientSecretPostAuthenticationFilter.
Captured UserPasswordFilter	6.5	key	Replaced by keySecretId . If the deprecated and replacement properties are both provided, the replacement property takes precedence.
	7.2	keyType value DES	Replaced by AES .

Object	Deprecat ed in IG	Deprecated settings	Replacement settings
ClientHan dler	7.2	proxy and systemProxy	Replaced by proxyOptions .
	6.5	proxy subproperty password	Replaced by passwordSecretId . If the deprecated and replacement properties are both provided, the replacement property takes precedence.
	6.5	<ul style="list-style-type: none"> • keyManager • sslCipherSuite s • sslContextAlgo rithm • sslEnabledProt ocols • trustManager 	Replaced by the ClientTlsOptions object. For more information, see ClientTlsOptions .
	6.5	websocket subproperties: <ul style="list-style-type: none"> • keyManager • sslCipherSuite s • sslContextAlgo rithm • sslEnabledProt ocols • trustManager 	Replaced by the ClientTlsOptions object. For more information, see ClientTlsOptions .
	7.2	hostnameVerifier	Replaced by hostnameVerifier in ClientTlsOptions . If a ClientHandler includes the deprecated "hostnameVerifier": "ALLOW_ALL" configuration, it takes precedence, and deprecation warnings are written to the logs.

Object	Deprecat ed in IG	Deprecated settings	Replacement settings
ClientRegi stration	7	<ul style="list-style-type: none"> • keystore • privateKeyJwtA lias • privateKeyJwtP assword 	<p>Replaced by privateKeyJwtSecretId .</p> <p>If the deprecated and replacement properties are both provided, the replacement property takes precedence.</p>
	7	name , when used to identify a registration	Replaced by clientId . For information, see ClientRegistration , and the example route in Use Multiple OpenID Connect Providers .
	6.5	clientSecret	<p>Replaced by clientSecretId .</p> <p>If the deprecated and replacement properties are both provided, the replacement property takes precedence.</p>
	7.2	<ul style="list-style-type: none"> • clientSecretId • tokenEndpointA uthMethod • tokenEndpointA uthSigningAlg • privateKeyJwtS ecretId • jwtExpirationT imeout • secretsProvide r . 	Replaced by authenticatedRegistrationHan dler .
CorsFilter	7.1	origins	Replaced by acceptedOrigins . For information, see CorsFilter .
CryptoHe aderFilter	7	Whole object	Not replaced. For information, see CryptoHeaderFilter .
DesKeyGe nHandle	7	Whole object	Not replaced. For information, see DesKeyGenHandler .

Object	Deprecat ed in IG	Deprecated settings	Replacement settings
ElasticsearchAuditEventHan dler	7.1	Whole object	Not replaced.
JwtBuilder Filter	6.5	signature subproperties: <ul style="list-style-type: none"> • keystore • alias • password 	Replaced by signature property secretId . If the deprecated and replacement properties are both provided, the replacement property takes precedence.

Object	Deprecat ed in IG	Deprecated settings	Replacement settings
JwtSessio n	7	encryptionSecretId and signatureSecretId	Replaced by authenticatedEncryptionSecretId and encryptionMethod.
	7	cookieName and cookieDomain	Replaced by cookie, and its subproperties name, domain, path, secure, httpOnly. If the deprecated and replacement properties are both provided, the replacement property takes precedence.
	6.5	password	Replaced by passwordSecretId. If the deprecated and replacement properties are both provided, the replacement property takes precedence.
	6.5	Combination of password, alias, and keystore Combination of passwordSecretId, alias, and keystore	Replaced by encryptionSecretId. If the deprecated and replacement properties are both provided, the replacement property takes precedence.
	6.5	sharedSecret	Replaced by signatureSecretId. If the deprecated and replacement properties are both provided, the replacement property takes precedence.
KeyMana ger	6.5	password	Replaced by passwordSecretId. If the deprecated and replacement properties are both provided, the replacement property takes precedence.

Object	Deprecat ed in IG	Deprecated settings	Replacement settings
KeyStore	6.5	password	Replaced by passwordSecretId . If the deprecated and replacement properties are both provided, the replacement property takes precedence.
OpenAm AccessTo kenResolv er	7	Whole object	Not replaced. For information, see OpenAmAccessTokenResolver .
Password ReplayFilt er	7	headerDecryption subproperties key and keyType	Replaced by keySecretId and secretsProvider .

Object	Deprecat ed in IG	Deprecated settings	Replacement settings
ReversePr oxyHandl er	7.2	proxy and systemProxy	Replaced by proxyOptions .
	7.1	proxy subproperty password	Replaced by passwordSecretId .
	7	<ul style="list-style-type: none"> • keyManager • sslCipherSuite s • sslContextAlgo rithm • sslEnabledProt ocols • trustManager 	Replaced by the ClientTlsOptions object. For more information, see ClientTlsOptions .
	7	websocket subproperties: <ul style="list-style-type: none"> • keyManager • sslCipherSuite s • sslContextAlgo rithm • sslEnabledProt ocols • trustManager 	Replaced by the ClientTlsOptions object. For more information, see ClientTlsOptions .
	7.2	hostnameVerifier	Replaced by hostnameVerifier in ClientTlsOptions . If a ReverseProxyHandler includes the deprecated "hostnameVerifier": "ALLOW_ALL " configuration, it takes precedence, and deprecation warnings are written to the logs.

Object	Deprecat ed in IG	Deprecated settings	Replacement settings
Route	6.5	monitor	Replaced by the Prometheus Scrape Endpoint and Common REST Monitoring Endpoint. For information, see Monitoring Endpoints .
SingleSign OnFilter	7	logoutEndpoint	Replaced by logoutExpression.
SplunkAu ditEventH andler	7.1	Whole object	Not replaced.
SqlAttribu tesFilter	7	dataSource as a JNDI lookup name	Replaced by dataSource as a JdbcDataSource configuration object.
Stateless AccessTo kenResolv er	6.5.1	signatureSecretId	Replaced by verificationSecretId.
	6.5.1	encryptionSecretI d	Replaced by decryptionSecretId.
UserProfil eFilter	6.5	ssoToken	Replaced by username in UserProfileFilter.
	6.5	amService and profileAttributes	Replaced amService and profileAttributes, as sub- properties of userProfileService

Object	Deprecat ed in IG	Deprecated settings	Replacement settings
The environm ent variable and system property that define the file system directory for configura tion files.	6.5	OPENIG_BASE and openig.base	Replaced by IG_INSTANCE_DIR and ig.instance.dir . If neither the deprecated setting nor the replacement setting are provided, configuration files are in the default directory \$HOME/.openig (on Windows, %appdata%\OpenIG). If the deprecated setting and the replacement setting are both provided, the replacement setting is used.
<u>Functions</u>	7.1.2	matches	Replaced by matchesWithRegex or find .
	7.1.2	matchingGroups	Replaced by findGroups .
<u>ClientTlsO ptions</u>	7.1.2	sslEnabledProtocol s with SSL 3 and SSL 2	<ul style="list-style-type: none"> • Use TLS 1.3 when it is supported by available libraries, otherwise use TLS 1.2. • If TLS 1.1 or TLS 1.0 is required for backwards compatibility, use it only with express approval from enterprise security.

Removed

The following features and properties have been removed, as defined in [ForgeRock product stability labels](#):

Object	Removed setting	Replacement setting
StaticResponseHandler	version	None

Fixes


The following important issues were fixed in this release:

- OPENIG-4956: Inbound WebSocket connection is not closed when outbound connection is closed abruptly
- OPENIG-5425: JwkSetHandler: No error displayed when using an invalid configuration such as a public key exported -as jwk- for decryption usage
- OPENIG-5539: The ForwardedRequestFilter should not change original URI parameter values when rebasing
- OPENIG-5540: PEM secret format fails to decode some EC private keys
- OPENIG-5610: Null Pointer Exception when using ForwardedRequestFilter with ResourceHandler
- OPENIG-5683: HTTP/2 : set max connections
- OPENIG-5725: Add SNI configuration
- OPENIG-5743: Standalone: Possible OOME for large requests
- OPENIG-5778: sessionInfo requests can lead to a build up of agent tokens being created
- OPENIG-5793: Unexpected behaviour of EL function matches
- OPENIG-5805: The notification service should attempt to refresh the caller token when receiving a 401 on WebSocket connections
- OPENIG-5868: WebSocketClientHandshakeException: Invalid subprotocol seen when using IG standalone to proxy WebSocket requests
- OPENIG-5872: Stop Tyrus WebSocket connection retry when Websocket Client is closed
- OPENIG-6206: When checking for peer certificates in a request, validate that the SSLSession is available
- OPENIG-6394: Stack traces are printed twice in the log files

Security advisories

ForgeRock issues security advisories in collaboration with our customers and the open source community to address any security vulnerabilities transparently and rapidly.

ForgeRock's security advisory policy governs the process on how security issues are submitted, received, and evaluated as well as the timeline for the issuance of security advisories and patches.

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

Limitations

Limitations are inherent to the design, not bugs to be fixed.

▼ [Multiple spaces in unquoted cookie values are changed to a single space in JBoss](#)

[OPENIG-4395](#) 

In JBoss, multiple spaces in unquoted cookie values are reduced to one space. For example:

```
testCookieName=cookie    value
```

is changed to

```
testCookieName=cookie value
```

▼ [No access to common time related functions in expressions](#)

[OPENIG-4201](#) 

The value of `System.currentTimeMillis()` cannot currently be used in filters, such as `JwtBuilderFilter`, for claims such as `exp` and `iat`.

▼ [Scripts can access anything in their environment](#)

[OPENIG-3274](#) 

IG scripts are not sandboxed, but instead have access to anything in their environment. You must make sure that the scripts that IG loads are safe.

▼ [Persist UMA shares](#)

[OPENIG-3273](#) 

Shared resources cannot be persisted when IG restarts. They must be shared each time that IG restarts. For more information, see [Support UMA resource servers](#).

▼ [Proxy WebSocket traffic when running in Jetty](#)

[OPENIG-3248](#) 

When IG is running in the Jetty application container, it cannot proxy WebSocket traffic.

For more information, see [Proxy WebSocket traffic](#), and the `websocket` property of `ClientHandler` or `ReverseProxyHandler`.

▼ [Blocked ClientHandler with asynchronous HTTP clients](#)

OPENIG-2417

IG processes responses from asynchronous HTTP clients by using two thread pools of the same size:

- The first thread pool receive the response headers.
- The second thread pool completes the promise by to executing the callback and writing the response content to the stream. Reading and writing to the stream are synchronous, blocking operations.

Synchronous operation can cause routes to declare a blocked ClientHandler.

To recover from blocking, restart the route, or, if the route is `config.json`, restart the server. To prevent blocking, increase the number of worker threads.

▼ [Cannot use custom config.json in Studio](#)

OPENIG-1557

When a customized `config.json` is configured in Studio, Studio cannot deploy routes.

▼ [Log file of audit events can be overwritten](#)

OPENIG-813

The log file of audit events can be overwritten when the log file is rotated.

When `CsvAuditEventHandler` is used to log audit events, the log file is overwritten if it is rotated before the file suffix, `rotationFileSuffix`, changes. By default, `rotationFileSuffix` is defined as a date in the format `_yyyy-MM-dd`.

Log files are rotated when one of the following limits is reached: `maxFileSize`, `rotationInterval`, or `rotationTimes`.

Set the log rotation parameters so that the log is not likely to rotate before `rotationFileSuffix` changes.

▼ [Cannot use SAML with AM policy agent](#)

OPENIG-291

When SAML is used with an AM policy agent, class cast exceptions occur.

▼ [SAML fails with incorrect user-defined mapping](#)

OPENIG-234

When the user defined mapping is incorrectly set, missing SAML assertions produce an infinite loop during authentication attempts.

▼ [For mutual authentication in HTTPS cannot specify which certificate to present](#)

[OPENIG-221](#) 

IG can check server certificates for HTTPS. However, for mutual authentication, the client certificate must be the first certificate in the KeyStore.

Known issues


The following important issues remain open in this release:

- OPENIG-5913: (UI) Route configuration lost sometime after un-deploy from route list
- OPENIG-4817: Can't specify any host information for HTTP/2 request

Documentation

Date	Description
June 2022	<p>Initial release of Identity Gateway 7.2 software.</p> <p>In addition to the changes described elsewhere in these notes, the following important changes were made to the documentation:</p> <p><i>New Sections</i></p> <ul style="list-style-type: none"> • Guidance on internationalization has been added to Internationalization. • A description of IG as a microgateway has been added to IG as a microgateway. • An example of how to pass runtime information in a JWT, with Identity Cloud as an identity provider, has been added to the Identity Cloud guide. • Information about how to configure SameSite for HTTP session cookies has been added to Install IG in Apache Tomcat, Install IG in Jetty, and Install IG in JBoss EAP. <p><i>Best practices</i></p> <ul style="list-style-type: none"> • A section on upgrade has been added to the Installation guide, and information about migrating from web container mode to standalone mode has been taken from the Release notes. • Information about how to increase the security of cookies in your deployment has been added to Manage cookies. • Information about how to change the session configuration has been added to Managing sessions. <p><i>Clarifications</i></p> <p>For completeness, information about features and properties that were deprecated in previous releases but not yet removed has been added back into the guides. Where possible use replacements instead of deprecated features or properties. For more information, refer to the Deprecated section of the <i>Release Notes</i>.</p>

Appendix A: Release levels and interface stability

For information about release levels, refer to [Product Support Lifecycle Policy](#) | [PingGateway and Agents](#) .

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	<p>This documented feature or interface is expected to undergo backwards-compatible changes only for major releases.</p> <p>Changes may be announced at least one minor release before they take effect.</p>
Evolving	<p>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.</p> <p>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.</p>
Legacy	<p>This feature or interface has been replaced with an improved version, and is no longer receiving development effort from ForgeRock.</p> <p>You should migrate to the newer version, however the existing functionality will remain.</p> <p>Legacy features or interfaces will be marked as <i>Deprecated</i> if they are scheduled to be removed from the product.</p>


Stability Label	Definition
Deprecated	<p>This feature or interface is deprecated, and likely to be removed in a future release.</p> <p>For previously stable features or interfaces, the change was likely announced in a previous release.</p> <p>Deprecated features or interfaces will be removed from ForgeRock products.</p>
Removed	<p>This feature or interface was deprecated in a previous release, and has now been removed from the product.</p>
Technology Preview	<p>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.</p> <p><i>DO NOT DEPLOY A TECHNOLOGY PREVIEW INTO A PRODUCTION ENVIRONMENT.</i></p> <p>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.</p> <p>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.</p> <p>Technology previews are provided on an “AS-IS” basis for evaluation purposes only, and ForgeRock accepts no liability or obligations for the use thereof.</p>
Internal/Undocumented	<p>Internal and undocumented features or interfaces can change without notice.</p> <p>If you depend on one of these features or interfaces, contact ForgeRock support or email info@forgerock.com to discuss your needs.</p>

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 everyone, 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.

Was this helpful?  