Replicator

JFrog Subscription Levels

CLOUD (SaaS) | SELF-HOSTED

ENTERPRISE+

Page Contents

Overview

Replication is used in a variety of use cases to synchronize repositories between remote Artifactory instances. Depending on repository size this may require the transportation of huge quantities of data. Since speed of transportation is limited by bandwidth and network latency, this is a process that may take hours and even days to complete.

Available from Artifactory version 6.0, the Replicator is a process that optimizes replication when distributing software with JFrog Distribution, dramatically reducing the load on the network and the time taken to synchronize release bundles from a source Artifactory instance to target instance or Edge node.

> The Replicator is installed out-of-the-box on every Artifactory and Artifactory Edge, but is required to be enabled. See Installation and Activation.

Supported Package Types

The replicator implements concurrent stream replication for maven, generic (zip) and docker package types supported by Artifactory.

> **Windows not supported**
>
> The Replicator does not work on Windows based systems

Default Ports

By default, the replicator uses ports **8048** and **9092**. If you are running behind a firewall, make sure these ports can be accessed.

Installation and Activation

> **Prerequisite**
>
> Make sure to configure the Custom Base URL on the Artifactory Edge Node.

The Replicator should be enabled on both Artifactory and Artifactory Edge.

To enable the Replicator, use one of the following methods:

- Set the environment variable JF_REPLICATOR_ENABLED=true before you start Artifactory (whether it is a fully featured installation or an Edge node).
  Alternatively, you can edit `$JFROG_HOME/artifactory/app/bin/artifactory.default` to include: `export JF_REPLICATOR_ENABLED=true.`
- When running with Docker, use:

  ```
  docker run ... -e JF_REPLICATOR_ENABLED=true ...
  ```

- Set the following attribute in the `system.yaml` file.

  ```
  replicator:
      enabled: true
  ```

If you install the Enterprise Plus license with Artifactory already running, you need to restart Artifactory for the license to take effect and enable the Replicator.

Configuring the Replicator

The Replicator's behavior is configured using two configuration files:

`$JFROG_HOME/artifactory/etc/system.yaml`

⚠ **Breaking changes from Artifactory version 7.0**

- All configurations have been moved from the `replicator.yaml` to the `Artifacotry system.yaml` file.
- The `tx` directory, containing temporary data, has been moved to `$JFROG_HOME/artifactory/var/bootsrtap/replicator/tx`.
- The `proflilingDisabled` parameter is deprecated. To enable/disable replicator profiling, use the `replicator.profiling.enabled` configuration in the `system.yaml` file.

replicator.internalUrl	**Optional [Default: http://localhost:[replicator.port]]** The URL through which the Replicator can be accessed within the internal network. In a default installation, using "localhost" is an appropriate setting because only the local Artifactory instance needs access to the Replicator. Since it will default to **http://localhost:[replicator.port]]** it is recommended to not set this property.
replicator.port	**Optional [Default: 8048]** The port which the replicator server will listen on for internal communication with the router.
replicator.tls.skipVerifyCertificate	**Optional [Default: false]** Specifies if the Replicator should skip the verification of self-signed certificates when it calls remote HTTPS URLs. Use with care.
logging.application.level	**Optional [Default: info]** Log level of the replicator-service.log file. Options are: warn, info, debug
replicator.profiling.allowedHost	**Optional [Default: localhost]** Allowed access host to profiling data. The default will allow access to the profiling data only to clients on the same host (recommended). If you want to allow access from every host use the reserved value 'all'
replicator.profiling.port	**Optional [Default: 8041]** Listen port for profiling data.
replicator.profiling.enabled	**Optional [Default: true]** Disable profiling. Will not listen on profiling port.
replicator.txDir.cleanUpAgeSecs	**Optional [Default: 2592000 (30 days)]** Number of seconds for a file to be considered ready for delete. The files saved here are chunks of files that are being saved during replicator work.
replicator.txDir.cleanUpIntervalSecs	**Optional [Default: cleanUpIntervalSecs (1 hour)]** Interval in which replicator will look for files in tx dir which are ready to be deleted.
replicator.size.of.bulk.sending.bytes	Optional [Default: 8,000,000 (8 MB)] Replicator will try to transfer at least this amount of bytes in every request to the target replicator. Available from Artifactory 6.12.0
replicator.max.connections.per.host.internal.client	Optional [Default: 0 (Unlimited)] Max http connections to the local Artifactory (0 means unlimited). Available from Artifactory 6.12.0

`$JFROG_HOME/artifactory/var/etc/replicator/replicator.artifactory.yaml` This file provides the Replicator with information on how it can access its associated Artifactory instance. If deleted, this file will be regenerated with its default values when you restart Artifactory. However, if the file exists, it will not be overwritten when Artifactory is restarted.

baseUrl	**Default: http://localhost:8081/artifactory** The Base URL of the associated Artifactory instance.
accessToken	An access token that the Replicator should use to access the associated Artifactory instance
artNodeId	Artifactory node id

Replicator Log Files

The Replicator's log files can be found under `$JFROG_HOME/artifactory/var/log/replicator/`.

The Replicator will rotate and compress old log files.

Working with a Proxy

The Replicator supports communication through a forward HTTP proxy.

Upon registration to Artifactory, the Replicator will request the default proxy settings and use them when needed.
Authenticated proxy servers (via basic authentication) are supported.

Working with Reverse Proxy

The replicator uses http streaming. In case you are using a reverse proxy like NGINX you should make sure the reverse proxy supports http 1.1 and does not buffer responses.
You might want to put the directive `proxy_buffering off;` only inside a location section.

```
proxy_http_version 1.1;
chunked_transfer_encoding on;
proxy_buffering off;
```

High Availability

When using the replicator between Artifactory HA clusters, there is no special configuration required on the source Artifactory cluster, however, the target Artifactory/Artifactory Edge cluster(s) must be configured with a reverse proxy with stickiness on a special HTTP header: `X-JFRPL-TXID`
The following snippet shows an example configuration with NGINX:

```
upstream replicator {
    hash $http_x_jfrpl_txid;
    server 10.1.16.31:8082;
    server 10.1.16.32:8082;
}

location /replicator/ {
      proxy_pass    http://replicator;
}
```

⚠ The port above should be the same port as the one of the Router, which is configured in the `docker-compose.yml` file.