



**OX2OX Migration Framework Scheduler Technical
Documentation for
2.1.0**

2021-07-23

Copyright notice

©2021 by OX Software GmbH. All rights reserved. Open-Xchange and the Open-Xchange logo are trademarks or registered trademarks of OX Software GmbH. All other company and/or product names may be trademarks or registered trademarks of their owners. Information contained in this document is subject to change without notice.



Contents

1	General Information	2
1.1	Warnings	2
1.2	Delivery Comment	2
1.3	Install Package Repository	2
1.4	Build Dependencies	2
2	Shipped Packages and Version	2
2.1	Package open-xchange-omf-orchestrator	2
2.1.1	Installation	2
2.1.2	Configuration	2
2.2	Package open-xchange-omf-scheduler	3
2.2.1	Installation	3
2.2.2	Configuration	3
A	Configuration Files	3

1 General Information

1.1 Warnings



Warning

This preview delivery is not for productive usage and not affected by service-level agreements.



Warning

Custom configuration or template files are potentially not updated automatically. After the update, please always check for files with a **.dpkg-new** or **.rpmnew** suffix and merge the changes manually. Configuration file changes are listed in their own respective section below but don't include changes to template files. For details about all the configuration files and templates shipped as part of this delivery, please read the relevant section of each package.

1.2 Delivery Comment

This delivery was requested with following comment:

Scheduler 2.1.0 Preview Delivery 5

1.3 Install Package Repository

This delivery is part of a restricted preview software repository:

<https://software.open-xchange.com/components/omf-scheduler/preview/2.1.0/RHEL7>
<https://software.open-xchange.com/components/omf-scheduler/preview/2.1.0/DebianStretch>
<https://software.open-xchange.com/components/omf-scheduler/preview/2.1.0/DebianBuster>

1.4 Build Dependencies

This delivery was build with following dependencies:

RedHat:RHEL-7,Debian:Stretch,Debian:Buster

2 Shipped Packages and Version

2.1 Package open-xchange-omf-orchestrator

OMF Orchestrator CLI to interoperate with the OX2OX Migration Framework.

Version: 2.1.0-5

Type: Other

2.1.1 Installation

Install on nodes with package installer **apt-get** or **yum**:

```
<package installer> install open-xchange-omf-orchestrator
```

2.1.2 Configuration

For details, please see appendix [A](#)

/opt/open-xchange/omf/orchestrator/etc/omf-orchestrator.yml (page [4](#))

2.2 Package open-xchange-omf-scheduler

OMF Scheduler OX2OX Migration Framework Scheduler.

Version: 2.1.0-5

Type: Other

2.2.1 Installation

Install on nodes with package installer **apt-get** or **yum**:

```
<package installer> install open-xchange-omf-scheduler
```

2.2.2 Configuration

For details, please see appendix [A](#)

/opt/open-xchange/omf/scheduler/etc/omf-scheduler.yml (page [7](#))

A Configuration Files

File 1 /opt/open-xchange/omf/orchestrator/etc/omf-orchestrator.yml

```

1 micronaut:
2   application:
3     name: omf
4   http:
5     services:
6       # The OMF Scheduler Source HTTP REST API service configuration
7       scheduler-admin-source:
8         url: "${omf.scheduler.url}/omf/scheduler/admin/source/"
9
10      # If the scheduler does not have a valid public certificate
11      # (e.g. uses a self-signed certificate), then its certificate can be
12      # configured here.
13      ssl:
14        enabled: true
15        trust-store:
16          path: file:/opt/open-xchange/omf/certs/scheduler.p12
17          password: secret
18          type: PKCS12
19
20      scheduler-admin-target:
21        url: "${omf.scheduler.url}/omf/scheduler/admin/target/"
22
23      # If the scheduler does not have a valid public certificate
24      # (e.g. uses a self-signed certificate), then its certificate can be
25      # configured here.
26      ssl:
27        enabled: true
28        trust-store:
29          path: file:/opt/open-xchange/omf/certs/scheduler.p12
30          password: secret
31          type: PKCS12
32
33      # The OMF Scheduler Migration HTTP REST API service configuration
34      scheduler-migration:
35        url: "${omf.scheduler.url}/omf/scheduler/migration/"
36
37      # If scheduler-admin above has a custom SSL configuration,
38      # then it needs to be repeated here.
39      ssl:
40        enabled: true
41        trust-store:
42          path: file:/opt/open-xchange/omf/certs/scheduler.p12
43          password: secret

```

```

44         type: PKCS12
45
46     # The OMF Scheduler Monitoring HTTP REST API service configuration
47     scheduler-monitoring:
48         url: "${omf.scheduler.url}/omf/scheduler/workers/monitor/"
49
50     # If scheduler-admin above has a custom SSL configuration,
51     # then it needs to be repeated here.
52     ssl:
53         enabled: true
54         trust-store:
55             path: file:/opt/open-xchange/omf/certs/scheduler.p12
56             password: secret
57             type: PKCS12
58
59 omf:
60     shell:
61         config:
62             user.dir: ${user.dir}/.omf/config
63             app.dir: /opt/open-xchange/omf/lib/scripts
64     scheduler:
65         # Credentials for the scheduler
66         # On multi-user systems, specifying the password in a configuration file
67         # with proper file system permissions is preferred to specifying it on
68         # the command line, since the command line is visible to all local users.
69         #
70         # Example:
71         # username: admin
72         # password: secret
73
74         # Location of the scheduler. Only the protocol and host name need to be
75         # specified.
76         url: "https://localhost:8443"
77     ui:
78         color: true
79         unicode: true
80         expandIds: false
81         history.file: ${user.dir}/.omf_history
82
83     logger:
84         levels:
85             # change this to TRACE to see a detailed log of the HTTP traffic between the
86             # Orchestrator and the Scheduler
87             io.micronaut.http.client: INFO

```

File 2 /opt/open-xchange/omf/scheduler/etc/omf-scheduler.yml

```

1  # https://docs.micronaut.io/latest/guide/config.html#configurationProperties
2  ---
3  micronaut:
4      # SSL configuration
5      # Required for production environments.
6      # See https://docs.micronaut.io/latest/guide/index.html#https for details.
7      ssl:
8          enabled: true
9          port: 8443
10         key-store:
11             path: file:/opt/open-xchange/omf/certs/keystore.p12
12             type: PKCS12
13             password: secret
14         server:
15             dual-protocol: true
16             port: 8080
17
18     http:
19         services:
20             # The omf-source service is use to collect health status and metrics from the Source
21             # OMF nodes. The HTTP client can be configured (ex: ssl) here by referencing

```

```

22     # https://docs.micronaut.io/latest/guide/configurationreference.html#io.micronaut.
23       http.client.ServiceHttpClientConfiguration
24     # and the subsequent sections related to micronaut.http.services.*
25     omf-source:
26       connect-timeout: 30s
27       read-timeout: 120s
28       # Example SSL configuration in case a source uses a private certificate
29       # ssl:
30       #   trust-store:
31       #     path: file:/opt/open-xchange/omf/certs/source.p12
32       #     type: PKCS12
33       #     password: secret
34     omf-target:
35       connect-timeout: 30s
36       read-timeout: 120s
37   application:
38     name: omf-scheduler
39     # Configure security including basic auth: https://micronaut-projects.github.io/
40     # micronaut-security/latest/guide/#basicAuth
41     # Must be set to true or the Source Controller is not secure
42   security:
43     enabled: true
44     # Change the security of the open api views to anonymous so that they can be viewed
45     # without credentials
46   intercept-url-map:
47     - pattern: /swagger/**
48       access:
49         - isAnonymous()
50     - pattern: /swagger-ui/**
51       access:
52         - isAnonymous()
53     - pattern: /rapidoc/**
54       access:
55         - isAnonymous()
56     - pattern: /redoc/**
57       access:
58         - isAnonymous()
59   # https://docs.micronaut.io/latest/guide/index.html#_configuring_caches
60   #caches:
61     #example:
62     #  charset: UTF-8
63     #  expire-after-access: 1h
64   metrics:
65     enabled: true
66     export:
67       # Creates an endpoint like http://host/prometheus - uses basic auth from
68       # credentials under scheduler.http.admin
69     prometheus:
70       enabled: true
71       step: PT1M
72       descriptions: true
73   router:
74     # Adds api versioning: https://docs.micronaut.io/latest/guide/index.html#apiVersioning
75     versioning:
76       enabled: true
77       parameter:
78         enabled: true
79         names: 'v'
80       header:
81         enabled: true
82         names: 'X-API-VERSION'
83     # Allows the openapi views to be seen
84   static-resources:
85     swagger:
86       paths: classpath:META-INF/swagger
87       mapping: /swagger/**
88     redoc:
89       paths: classpath:META-INF/swagger/views/redoc
90       mapping: /redoc/**
91     rapidoc:
92       paths: classpath:META-INF/swagger/views/rapidoc
93       mapping: /rapidoc/**

```

```

91     swagger-ui:
92       paths: classpath:META-INF/swagger/views/swagger-ui
93       mapping: /swagger-ui/**
94 ---
95 scheduler:
96   id: 'scheduler-0'
97   hostname: ''
98   batch:
99     presync:
100       # Max number of contexts in a batch
101       size: 10
102       # Strategy to use when creating batches.
103       # Current supported strategies:
104       #   - fill-first: create batches up to the batch size then create the next batch
105       #   - fill-equal: create batches of equal size
106       strategy: fill-equal
107     cutover:
108       size: 10
109       strategy: fill-equal
110     preprovisioning:
111       size: 10
112       strategy: fill-equal
113 kafka:
114   wait: false
115   queues:
116     batch: "omf-batch"
117     response: "omf-response"
118   resize:
119     batch: true
120     response: true
121   record.header.enhance: true
122 workers:
123   allow.shutdown: false
124 http:
125   admin:
126     # Basic auth creds
127     username: admin
128     password: secret
129     controller:
130       path: /omf/scheduler/admin
131   migration:
132     controller:
133       path: /omf/scheduler/migration
134 metrics:
135   cache:
136     windows.millis: 300000
137     batches.millis: 300000
138     migrationevents.millis: 300000
139     source.millis: 300000
140     target.millis: 300000
141 ---
142 jackson:
143   bean-introspection-module: true
144   serialization:
145     indent-output: true
146     writeDatesAsTimestamps: false
147 ---
148 datasources:
149   # Used to persist scheduling data
150   scheduler:
151     # url should use createDatabaseIfNotExist=true if the database will not
152     # already exist: https://dev.mysql.com/doc/connector-j/8.0/en/connector-j-reference-
153     # configuration-properties.html
154     url: jdbc:mysql://localhost:3306/scheduler?createDatabaseIfNotExist=true
155     username: root
156     password: my-secret-pw
157     driverClassName: org.mariadb.jdbc.Driver
158     dialect: MYSQL
159     maximumPoolSize: 10
160     maxLifetime: 180000
161   # Used to create/drop databases for sources. This is not really the "default" data
162   source

```



```

161 # but we need to use default because of bug https://github.com/micronaut-projects/
    micronaut-data/issues/598s
162 default:
163   url: jdbc:mysql://localhost:3306/
164   username: root
165   password: my-secret-pw
166   driverClassName: org.mariadb.jdbc.Driver
167   dialect: MYSQL
168   maximumPoolSize: 5
169   maxLifetime: 180000
170 ---
171 endpoints:
172   loggers:
173     enabled: true
174     sensitive: true
175   health:
176     discovery-client:
177       enabled: false
178     sources:
179       enabled: false
180     targets:
181       enabled: false
182   liquibase:
183     # fails with missing transition, might be fixed in later Micronaut releases
184     enabled: false
185   info:
186     enabled: true
187     sensitive: true
188     sourceCodeOrigin:
189       enabled: true
190       location: file:/opt/open-xchange/omf/scheduler/share/SourceCodeOrigin.txt
191 ---
192 zookeeper:
193   server: zookeeper:2181
194 ---
195 kafka:
196   bootstrap:
197     servers: kafka-1:9092, kafka-2:9092, kafka-3:9092
198   producers:
199     batch-producer:
200       enable.idempotence: true
201       # This enables transactions for the Batch Producer
202       # The value must be unique per application, but should
203       # not change for the same app after a crash, etc.
204       transactional.id: producer-1
205     #consumers:
206     #response-consumer:
207 ---
208 liquibase:
209   datasources:
210     scheduler:
211       change-log: 'classpath:liquibase/scheduler/liquibase-changelog.xml'
212 ---
213 logger:
214   levels:
215     ROOT: INFO
216     com.openxchange: INFO
217     omf: INFO
218     omf.scheduler.admin.AuthenticationProviderUserPassword: WARN
219     org.apache.kafka.clients.consumer.ConsumerConfig: WARN
220 ---

```