

Manager settings

- [Settings file wcs-manager.properties](#)
- [Settings file log4j.properties](#)
- [Configuring WCS Manager database](#)
 - [Creating a database](#)
 - [Filling up the database](#)
 - [Migrating to MySQL](#)
 - [MySQL installation and setup](#)
 - [WCS configuration](#)
 - [Data import from H2 to MySQL](#)

Settings file wcs-manager.properties

Main manager options are in `usr/local/FlashphonerWebCallServer/conf/wcs-manager.properties` file.

Setting	Default value	Description
<code>spring.datasource.url</code>	<code>jdbc:h2:C:/tmp/wcs_manager;FILE_LOCK=NO</code>	JDBC URL of the database
<code>spring.datasource.driverClassName</code>	<code>org.h2.Driver</code>	Class of the database driver
<code>spring.datasource.database</code>	<code>org.hibernate.dialect.H2Dialect</code>	SQL dialect used
<code>spring.datasource.username</code>	<code>admin</code>	Username
<code>spring.datasource.password</code>	<code>admin</code>	Password
<code>spring.datasource.ddlMode</code>	<code>update</code>	Should the required tables be created in the database upon the first run create - create tables validate - do not create tables, validate integrity update - do not create tables, update the scheme
<code>spring.jpa.hibernate.ddl-auto</code>	<code>update</code>	Automatically create and fill the database upon start: create - create validate - do not create, validate integrity update - do not create, update the scheme create-drop - create, drop tables in the end of the session
<code>spring.jpa.hibernate.naming_strategy</code>	<code>org.hibernate.cfg.ImprovedNamingStrategy</code>	Table naming strategy in the database
<code>spring.jpa.database</code>	<code>H2</code>	Database dialect used
<code>spring.jpa.show-sql</code>	<code>true</code>	Show SQL queries to the database.
<code>server.port</code>	<code>9091</code>	The HTTP port, WCS Manager will respond to incoming HTTP requests to.
<code>manager.enable_https</code>	<code>true</code>	Enable HTTPS
<code>manager.https_port</code>	<code>8888</code>	HTTPS port

manager.keystore	wss.jks	SSL certificate store
manager.keystore_password	password	Password to the SSL certificate store
spring.profiles.active	dev	Active profile dev - if this profile is enabled, WCS Manager will listen for HTTP port and will not require authentication. production - if this profile is enabled, WCS Manager will listen for HTTPS port and will require Basic-authentication from incoming HTTPS connections. The profile can be redefined upon server startup as well as other settings. See the Startup options section.
node_api.port	1098	Local WCS Core port WCS Manager will connect using the RMI interface to for data exchange.
node_api.service_name	NodeApi	Name under which WCS Core is registered in the RMI register
node_api.host	localhost	Host the RMI request is addressed to.
rest_template.allow_self_signed	true	Accept self-signed certificated from the Web server when accessing the Web server via HTTPS/REST.
shell.disabled_commands	autoconfig*,beans*,cron*,dashboard*,egrep*,endpoint*,env*,filter*,java*,jmx*,jul*,jvm*,jpa*,jdbc*,jndi*,less*,mail*,metrics*,shell*,sleep*,sort*,system*,thread*,repl*	Filters out unnecessary commands
spring.main.show_banner	false	Show the banner
node.enable_stdout	false	Enable logging of WCS Server to logs/server_stdout.log. /Can be used for debugging.
sun.rmi.dgc.client.gcInterval	36000000000	Sets interval for the garbage collector for the RMI client, in milliseconds. This setting allows you to turn off forced execution of the garbage collector to increase performance of the server
sun.rmi.dgc.server.gcInterval	36000000000	Sets interval for the garbage collector for the RMI server, in milliseconds. This setting allows you to turn off forced execution of the garbage collector to increase performance of the server
stream_stats.persist_data	false	Record stream statistics to the database
server.session-timeout	10	Session timeout, in minutes
loggc	/usr/local/FlashphonerWebCallServer/logs/gc-manager-	Location and prefix of the log file of the garbage collector. The name is constructed as /usr/local/FlashphonerWebCallServer/logs/gc-manager-YYYY-MM-DD_HH-MM.log
manager.rest_redirect_http_port	8081	Port number to redirect REST API queries from server.port
manager.rest_redirect_https_port	8444	Port number to redirect REST API queries from manager.https_port

Settings file log4j.properties

This file defines logging settings for WCS Manager. By default, logs of WCS Manager are recorded to flashphoner_manager.log. See the [Logging](#) section for advanced settings and logging control.

This file is packed into wcs_manager-1.0.jar file. Use Midnight Commander to change settings - just 'enter' into wcs_manager-1.0.jar file, open log4j.properties file, edit and save changes. Restart WCS for changes to take effect.

Configuring WCS Manager database

Creating a database

By default, WCS Manager uses the embedded H2 database that is created and filled up upon first startup of the WCS server. Database files are created in the home directory of the WCS server in the 'database' directory.

If the database file does not exist, an attempt to create the database afresh will be made.
If the database file already exists, WCS Manager simply connects to this database.

Filling up the database

The base is filled up with default values (users, groups, applications etc.) only if there is no a super-group record. In all other cases the contents of the database remain intact.

Migrating to MySQL

MySQL installation and setup

Install MySQL (on CentOS 7 example) with command

```
rpm -Uvh http://dev.mysql.com/get/mysql-community-release-el7-5.noarch.rpm
yum install mysql-server
```

Launch MySQL and set root password

```
/sbin/service mysqld start
/usr/bin/mysqladmin -u root password 'rootpassword'
```

Log in to MySQL as root

```
mysql -uroot -prootpassword
```

Create database and user, grant to the user privileges to the database

```
create database wcs_db;
create user 'webcallserver'@'localhost' identified by 'password';
grant all privileges on wcs_db.* to 'webcallserver'@'localhost';
```

WCS configuration

Add the following strings to [wcs-manager.properties](#) file

```
-Dspring.datasource.url=jdbc:mysql://localhost/wcs_db
-Dspring.datasource.driverClassName=com.mysql.jdbc.Driver
-Dspring.datasource.database=org.hibernate.dialect.MySQL5Dialect
-Dspring.datasource.username=webcallserver
-Dspring.datasource.password=password
```

Where

- 'wcs_db' is the database name
- 'webcallserver' and 'password' is user name and password

Restart WCS server

```
service webcallserver restart
```

Upon first startup of WCS, set `spring.datasource.ddlMode=create` in `wcs-manager.properties`. In this case all required tables will be created automatically. Upon a subsequent run, switch this setting to `spring.datasource.ddlMode=validate`. As a result connection to the existing database is performed without creating tables.

Data import from H2 to MySQL

Migrating from H2 to MySQL, data exported from H2 database can be imported to MySQL database::

a) using SQL script

```
mysql -uroot -prootpassword wcs_db < wcs_h2_db.sql;
```

b) loading data to tables from CSV files:

stop WCS server

```
service webcallserver stop
```

log in to MySQL

```
mysql -uroot -prootpassword
```

select database

```
use wcs_db;
```

load data from CSV file to existing table, e.g.,

```
LOAD DATA INFILE "/var/lib/mysql-files/STREAM_INFO_3.csv"  
REPLACE INTO TABLE stream_info_3  
FIELDS TERMINATED BY ','  
OPTIONALLY ENCLOSED BY '"'  
LINES TERMINATED BY '\n'  
IGNORE 1 LINES;
```

However, SQL script and CSV files generated by H2 are not fully compatible with MySQL and would have to be changed for importing. E.g., for table STREAM_INFO_3 it would be required to change Boolean values.