IBM

IBM Architecture Room LIVE! Installation

Contents

Chapter 1. Installing

IBM Architecture Room LIVE! supports installation with and without Internet connectivity (offline). IBM Architecture Room LIVE! provides installation scripts which can be used for installation and deployment of the web server. These are Node JS scripts and work on both Windows and Linux. The following sections cover the different procedures for each of the scenarios.

To update an existing version of the application, go directly to Chapter 3, "Updating IBM Architecture Room LIVE!," on page 19.

IBM Architecture Room LIVE! system requirements

Your environment must meet or have these prerequisites, operating systems, and other software versions for installing and running IBM Architecture Room LIVE!

Prerequisites
- Mongo database version 4.0.1
- Node JS version 8.11.3
- PM2 version 3.2.2
- Ensure Java version 8 or later is installed on the server where the licensing agent server is running.
- Before you start the license server agent, ensure that Rational Common Licensing server is configured with the IBM Architecture Room LIVE! license features.

Note: Microsoft Visual C++ Redistributable Package is available by default, but you might have to reinstall it, if you find any errors related to this package while installing IBM Architecture Room LIVE!

Operating systems

The following operating systems are supported by the IBM Architecture Room LIVE! Server:
- Red Hat Enterprise Linux (RHEL) Server x86-64 version 6.8.x, 7.x, or later
- Windows Server 2016 (all editions) 64-bit
- Windows Server 2012 Standard Edition 64-bit
- Windows Server 2012 R2 Standard Edition 64-bit
- Windows Server 2012 Datacenter Edition 64-bit
- Windows Server 2012 R2 Datacenter Edition 64-bit
- Windows Server 2008 R2 Enterprise Edition 64-bit
- Windows Server 2008 R2 Standard Edition 64-bit
- Windows 10 64-bit
- Windows 7 64-bit

Browser support

The IBM Architecture Room LIVE! Web application works best with the recent versions of three web browsers. Use the following versions of these web browsers:

- Chrome version 67 or later
- Firefox version 57 or later

Screen resolution

The display of IBM Architecture Room LIVE! on your monitor depends on the monitor, screen resolution, and browser zoom settings. The recommended screen resolution for a sharp quality display is 1920 x 1080 pixels. If your screen resolution is not in accordance with the required specification, then during login, you are prompted to check your resolution. If your screen resolution is low, then you can adjust the zoom level of your browser by using the `Ctrl` and `+` keys or `Ctrl` and `-` keys for optimal visual experience.

About the installation scripts

Use the **node** command for running the IBM Architecture Room LIVE! installation and deployment scripts.

List of installation scripts

The following table summarizes the scripts that are used, what the scripts do, and what they assume from the environment:

preinstall-dr.js	Downloads the Mongo database and the PM2 process management application from the internet. The license agreements of these applications prevent us from bundling them with IBM Architecture Room LIVE!. This script must be run on a computer with internet access.
install-dr.js	Installs the Mongo database and the PM2 process management application. Run this script on the computer where you want to install the IBM Architecture Room LIVE! web server. No internet access is required.
deploy-dr.js	Starts the Mongo database and launches the IBM Architecture Room LIVE! web server. Run this script on the same computer where the **install-dr.js** script was run.

Note: On Linux, the Mongo database is not downloaded and installed automatically by the scripts. For this reason, the best way to install Mongo on Linux is to use the package management system that is provided by each Linux distribution. See Mongo documentation for more information.

Installing on Windows with Internet

When you have Internet access, installing IBM Architecture Room LIVE! on Windows is a quick procedure.

Procedure

1. Extract the file IBM_Architecture_Room_LIVE_<version>_setup.zip to an empty folder <live-install-folder>. For example, the name of the ZIP file might be IBM_Architecture_Room_LIVE_V9.7_setup.zip.

Note: The built-in "Extract All..." utility in Windows has a limitation on the length of the path names of the extracted files. You can reduce the risk of running into this limitation by extracting the compressed file into a folder where the full path name is short. If this doesn't help, use a better extraction utility, such as 7-zip.

2. Install Node JS. You must have the administrative access on the machine to perform the installation.

   a. Double-click `<live-install-folder>\server-sw\win32_x86_x64\node-v8.11.3-x64.msi`.

   b. Accept the default installation options. The Windows installer automatically updates the PATH variable. Before you proceed, test that the node command is available from the command prompt by typing the following command:

      `node --version`

      The version of Node JS should be printed.

      Note: If you do not have administrative access to your machine, then you may get errors when running the installation and deployment scripts. To get the administrative rights, right-click Command Prompt program in the **Start** menu and select `Run as administrator`.

3. Open a new command prompt in the administrative mode and go to `<live-install-folder>\DR_Install` folder. Run the following command:

   `node preinstall-dr.js`

   The **preinstall-dr.js** script downloads the Mongo database and the PM2 process management application from the Internet. The downloaded applications are placed in the following path: `<live-install-folder>\DR_Install\downloaded`.

   The next script to run is `install-dr.js` which installs the Mongo database and the PM2 process management application. The installation is controlled by the settings in the file `<live-install-folder>\DR_Install\dr-config.json`. Review these settings and update them if needed before running the script. Settings marked with the *// user option comment* are those that you may want to change before installing.

4. Perform the installation by running the following command:

   `node install-dr.js`

   The installation of the IBM Architecture Room LIVE! web server is now complete, and it is ready to be deployed. The deployment settings are found in `<live-install-folder>\DR_Install\dr-config.json` and you can customize the settings as described earlier.

5. Stop the **MongoDB service** in Windows Services, and change its **Startup Type** to `Manual`. This step is important for the deploy script to create the `data` folder in the path specified in the `dr-config.json` file.

6. Configuring license server.

   Prerequisites:
   - Ensure Java version 8 or later is installed on the server where the licensing agent server is running.
   - Before you start the license server agent, ensure that Rational Common Licensing server is configured with the IBM Architecture Room LIVE! license features.

a. **Start the licensing agent on Windows:** Run the following command:

```
java -cp . -jar DRLiveLicenseAgent.jar [server_port@license_server_host_address] [agent_port]
```

For example: `java -cp . -jar DRLiveLicenseAgent.jar` *27000@1.2.3.4 6666*, where *27000* is the license server port number, *1.2.3.4* is the host address, and *6666* is the license agent port number.

b. **Configure IBM Architecture Room LIVE!**

1) Update `licenseAgentURL` value in the `<live-install-folder>\ liveHomeServer\user_config.js` file to include the license server agent details. For example: `licenseAgentURL: http://localhost:6666`

2) Restart IBM Architecture Room LIVE!

Note: If licensing is not configured, then IBM Architecture Room LIVE! will run on trial license mode for 60 days by default. Trial license will expire after 60 days, and users are warned about the same from two-weeks before the trial license expiration date and are reminded to configure the product license. To use the product after the trial license expires, you must activate a valid IBM Architecture Room LIVE! license.

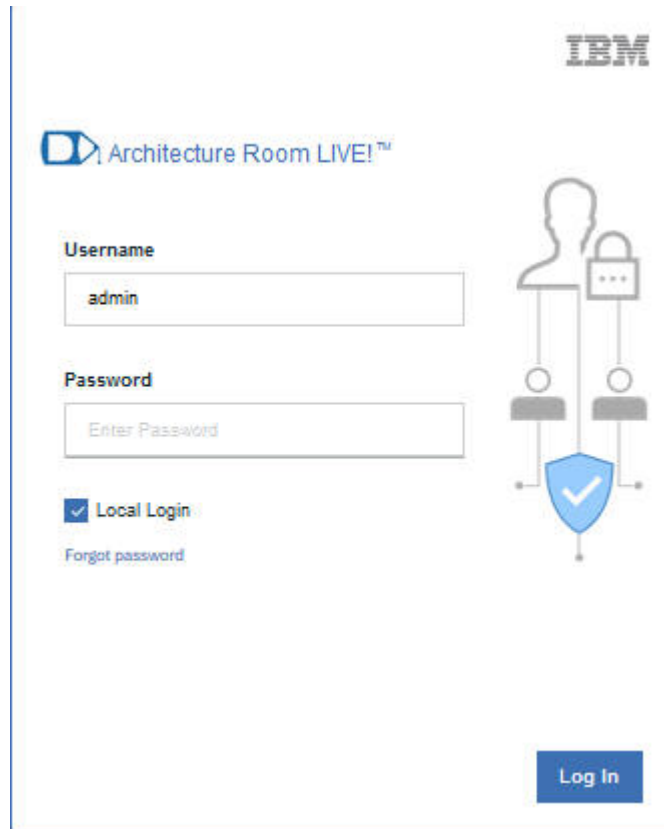7. Deploy the server by running the following command:

```
node deploy-dr.js
```

This script starts up the Mongo database (unless it is already running). It then starts the IBM Architecture Room LIVE! web server by using the PM2 process management application. The settings for how to deploy the server are stored in the PM2 configuration file `<live-install-folder>\DR_Install\ pm2.config.js`. You can edit this file as needed.

The deployment script finishes by printing the URL of the IBM Architecture Room LIVE! Web server. Test whether the installation and deployment was successful by opening the URL in a web browser. For example, the default settings in `pm2.config.js` yields the following URL:

```
https://localhost/index.html
```

where *localhost* is the server name where the application is running.

Notes: You might face the security certificate related issues while logging in from the browsers.

- If the Live! server does not have a valid SSL certificate and uses a self-signed certificate (by default it does), you might see the browser security warning. Add a browser security exception to the web page and continue.
- If the authentication fails because the Live! authentication server does not have a valid security certificate, you will see the *Authentication server is down* error when you login. To resolve the issue, see the *Invalid security certificate error during authentication* section in the Invalid security certificate errors topic.

Installing on Windows without Internet

Installing on Windows without Internet involves a few manual steps in the procedure.

Procedure

1. Download the IBM_Architecture_Room_LIVE_<version>_setup.zip file in a system that has Internet access.
2. Extract the file IBM_Architecture_Room_LIVE_<version>_setup.zip to an empty folder <live-install-folder>. For example, the name of the ZIP file might be IBM_Architecture_Room_LIVE_V9.7_setup.zip.

   Note: The built-in "Extract All..." utility in Windows has a limitation on the length of the path names of the extracted files. You can reduce the risk of

running into this limitation by extracting the compressed file into a folder where the full path name is short. If this doesn't help, use a better extraction utility, such as 7-zip.

3. Install Node JS. You must have the administrative access on the machine to perform the installation.

   a. Double-click `<live-install-folder>\server-sw\win32_x86_x64\node-v8.11.3-x64.msi`.

   b. Accept the default installation options. The Windows installer automatically updates the PATH variable. Before you proceed, test that the node command is available from the command prompt by typing the following command:

   ```
   node --version
   ```

   The version of Node JS should be printed.

   Note: If you do not have administrative access to your machine, then you may get errors when running the installation and deployment scripts. To get the administrative rights, right-click Command Prompt program in the **Start** menu and select `Run as administrator`.

4. Open a new command prompt in the administrative mode and go to `<live-install-folder>\DR_Install` folder. Run the following command:

   ```
   node preinstall-dr.js
   ```

   The **preinstall-dr.js** script downloads the Mongo database and the PM2 process management application from the Internet. The downloaded applications are placed in the following path: `<live-install-folder>\DR_Install\downloaded`.

5. Go to the machine that does not have access to internet, where you want to install the IBM Architecture Room LIVE! web server, and copy the `<live-install-folder>` to the installation machine. Continue to run the remaining steps on that machine. As a prerequisite, you must have installed Node JS on that machine (see step 3).

   The next script to run is `install-dr.js` which installs the Mongo database and the PM2 process management application. The installation is controlled by the settings in the file `<live-install-folder>\DR_Install\dr-config.json`. Review these settings and update them if needed before running the script. Settings marked with the *// user option comment* are those that you may want to change before installing.

6. Perform the installation by running the following command:

   ```
   node install-dr.js
   ```

   The installation of the IBM Architecture Room LIVE! web server is now complete, and it is ready to be deployed. The deployment settings are found in `<live-install-folder>\DR_Install\dr-config.json` and you can customize the settings as described earlier.

7. Stop the **MongoDB service** in Windows Services, and change its **Startup Type** to `Manual`. This step is important for the deploy script to create the `data` folder in the path specified in the `dr-config.json` file.

8. Configuring license server.

   Prerequisites:

   • Ensure Java version 8 or later is installed on the server where the licensing agent server is running.

- Before you start the license server agent, ensure that Rational Common Licensing server is configured with the IBM Architecture Room LIVE! license features.

a. **Start the licensing agent on Windows:** Run the following command:

```
java -cp . -jar DRLiveLicenseAgent.jar [server_port@license_server_host_address] [agent_por
```

For example: java -cp . -jar DRLiveLicenseAgent.jar *27000@1.2.3.4 6666*, where *27000* is the license server port number, *1.2.3.4* is the host address, and *6666* is the license agent port number.

b. **Configure IBM Architecture Room LIVE!**

1) Update licenseAgentURL value in the <live-install-folder>\ liveHomeServer\user_config.js file to include the license server agent details. For example: licenseAgentURL: http://localhost:6666

2) Restart IBM Architecture Room LIVE!

Note: If licensing is not configured, then IBM Architecture Room LIVE! will run on trial license mode for 60 days by default. Trial license will expire after 60 days, and users are warned about the same from two-weeks before the trial license expiration date and are reminded to configure the product license. To use the product after the trial license expires, you must activate a valid IBM Architecture Room LIVE! license.

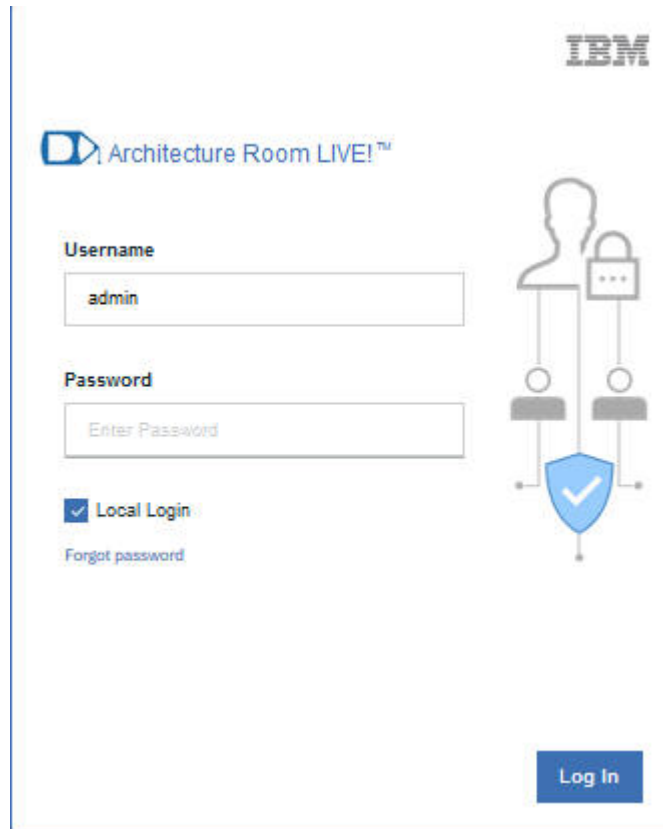9. Deploy the server by running the following command:

```
node deploy-dr.js
```

This script starts up the Mongo database (unless it is already running). It then starts the IBM Architecture Room LIVE! web server by using the PM2 process management application. The settings for how to deploy the server are stored in the PM2 configuration file <live-install-folder>\DR_Install\ pm2.config.js. You can edit this file as needed.

The deployment script finishes by printing the URL of the IBM Architecture Room LIVE! web server. Test whether the installation and deployment was successful by opening the URL in a web browser. For example, the default settings in pm2.config.js yields the following URL:

```
https://localhost/index.html
```

where *localhost* is the server name where the application is running.

Notes: You might face the security certificate related issues while logging in from the browsers.

- If the Live! server does not have a valid SSL certificate and uses a self-signed certificate (by default it does), you might see the browser security warning. Add a browser security exception to the web page and continue.
- If the authentication fails because the Live! authentication server does not have a valid security certificate, you will see the *Authentication server is down* error when you login. To resolve the issue, see the *Invalid security certificate error during authentication* section in the Invalid security certificate errors topic.

Installing on Linux with Internet

Use the following steps to install IBM Architecture Room LIVE! on a Linux machine that has the Internet access.

Procedure

1. Extract the file IBM_Architecture_Room_LIVE_<version>_setup.zip to an empty folder <live-install-folder>. For example, the name of the ZIP file might be IBM_Architecture_Room_LIVE_V9.7_setup.zip.
2. Download Mongo DB and install in the system. See the Mongo documentation for more information.
3. Install Node JS. You must have the administrative access on the machine to perform the installation.
   a. Run the following command in a shell:

```
tar -Jxf <dr-install-folder>/server-sw/linux_x86_x64/
node-v8.11.3-linux-x64.tar.xz
```

Note: You must manually update the PATH variable so that it contains the bin folder of the Node JS installation.
Before you proceed, ensure that the node command is available from the command line:

```
node --version
```

The version of Node JS should be printed.

Note: If you do not have root access to your machine, then you may get errors when you run the installation and deployment scripts.

4. Go to `<live-install-folder>/DR_Install` folder and run the following command:

```
sudo node preinstall-dr.js
```

The `preinstall-dr.js` script downloads the PM2 process management application from the Internet. The downloaded applications are placed in the following path: `<live-install-folder>/DR_Install/downloaded`.

The next script to run is `install-dr.js` which installs the PM2 process management application. The installation is controlled by the settings in the file `<live-install-folder>/DR_Install/dr-config.json`. Review these settings and update them if needed before running the script. Settings marked with the *// user option* comment are those that you might want to change before installing. You must pay attention to the setting in the `mongo-linux-install-folder` and ensure that it is set to the path where `bin/mongod` is located.

Review and update the installation settings by directly editing the `dr-config.json` file as needed.

5. Perform the installation by running the following command:

```
sudo node install-dr.js
```

The installation of the IBM Architecture Room LIVE! web server is now complete, and it is ready to be deployed. Once again, the deployment settings are found in `<live-install-folder>/DR_Install/dr-config.json` and you can customize the settings as needed.

6. Configuring license server.

   Prerequisites:

   - Ensure Java version 8 or later is installed on the server where the licensing agent server is running.
   - Before you start the license server agent, ensure that Rational Common Licensing server is configured with the IBM Architecture Room LIVE! license features.

   a. **Start the licensing agent on Linux:** Run the following command:

   ```
   java -Djava.library.path="." -jar DRLiveLicenseAgent.jar [server_port@license_server_host_a
   ```

   For example: java -Djava.library.path="." -jar DRLiveLicenseAgent.jar *27000@1.2.3.4 6666,* where *27000* is the license server port number, *1.2.3.4* is the host address, and *6666* is the license agent port number.

   b. **Configure IBM Architecture Room LIVE!**

1) Update `licenseAgentURL` value in the `<live-install-folder>/user_config.js` file to include the license server agent details. For example: `licenseAgentURL: http://localhost:6666`

2) Restart IBM Architecture Room LIVE!

Note: If licensing is not configured, then IBM Architecture Room LIVE! will run on trial license mode for 60 days by default. Trial license will expire after 60 days, and users are warned about the same from two-weeks before the trial license expiration date and are reminded to configure the product license. To use the product after the trial license expires, you must activate a valid IBM Architecture Room LIVE! license.
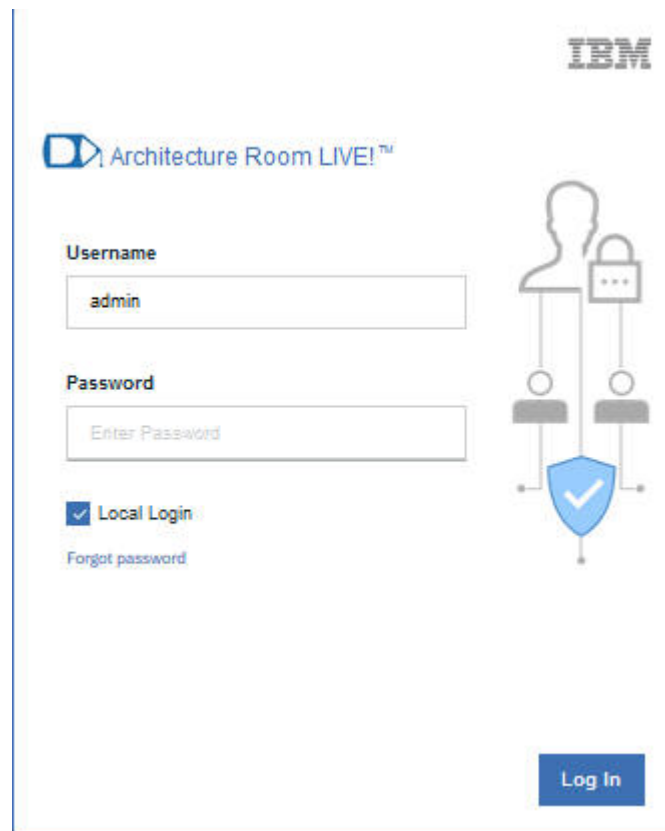
7. Deploy the server by running the following command:

`sudo node deploy-dr.js`

The deployment script starts up the Mongo database (unless it is already running). It then starts the IBM Architecture Room LIVE! web server by using the PM2 process management application. The settings for how to deploy the server are stored in the PM2 configuration file `<live-install-folder>/DR_Install/pm2.config.js`. You can edit this file as needed.

The deployment script finishes by printing the URL of the IBM Architecture Room LIVE! web server. Test whether the installation and deployment was successful by opening the URL in a web browser. For example, the default settings in the `pm2.config.js` file yield the following URL:

`https://localhost/index.html`

where *localhost* is the server name where the application is running.

Notes: You might face the security certificate related issues while logging in from the browsers.

- If the Live! server does not have a valid SSL certificate and uses a self-signed certificate (by default it does), you might see the browser security warning. Add a browser security exception to the web page and continue.

- If the authentication fails because the Live! authentication server does not have a valid security certificate, you will see the *Authentication server is down* error when you login. To resolve the issue, see the *Invalid security certificate error during authentication* section in the Invalid security certificate errors topic.

Installing on Linux without Internet

Use the following steps to install IBM Architecture Room LIVE! on a Linux machine that does not have Internet access.

Procedure

1. Download the `IBM_Architecture_Room_LIVE_<version>_setup.zip` file in a system that has Internet access.

2. Extract the file `IBM_Architecture_Room_LIVE_<version>_setup.zip` to an empty folder `<live-install-folder>`. For example, the name of the ZIP file might be `IBM_Architecture_Room_LIVE_V9.7_setup.zip`.

3. Install Node JS. You must have the administrative access on the machine to perform the installation.

   a. Run the following command in a shell:

   ```
   tar -Jxf <dr-install-folder>/server-sw/linux_x86_x64/
   node-v8.11.3-linux-x64.tar.xz
   ```

   Note: You must manually update the PATH variable so that it contains the bin folder of the Node JS installation.
   Before you proceed, ensure that the node command is available from the command line:

   ```
   node --version
   ```

   The version of Node JS should be printed.

   Note: If you do not have root access to your machine, then you may get errors when you run the installation and deployment scripts.

4. Go to `<live-install-folder>/DR_Install` folder and run the following command:

   ```
   sudo node preinstall-dr.js
   ```

   The `preinstall-dr.js` script downloads the PM2 process management application from the Internet. The downloaded applications are placed in the following path: `<live-install-folder>/DR_Install/downloaded`.

5. Go to the machine that has no Internet access, where you want to install the IBM Architecture Room LIVE! web server, and copy the `<live-install-folder>` to the installation machine. Continue to run the remaining steps on that machine. As a prerequisite, you must have installed Node JS on that machine (see step 3).

6. Download Mongo DB and install in the system. See the Mongo documentation for more information.

   The next script to run is `install-dr.js` which installs the PM2 process management application. The installation is controlled by the settings in the file `<live-install-folder>/DR_Install/dr-config.json`. Review these settings and

update them if needed before running the script. Settings marked with the *// user option* comment are those that you might want to change before installing. You must pay attention to the setting in the `mongo-linux-install-folder` and ensure that it is set to the path where `bin/mongod` is located.

Review and update the installation settings by directly editing the `dr-config.json` file as needed.

7. Perform the installation by running the following command:

```
sudo node install-dr.js
```

The installation of the IBM Architecture Room LIVE! web server is now complete, and it is ready to be deployed. Once again, the deployment settings are found in `<live-install-folder>/DR_Install/dr-config.json` and you can customize the settings as needed.

8. Configuring license server.

   Prerequisites:
   - Ensure Java version 8 or later is installed on the server where the licensing agent server is running.
   - Before you start the license server agent, ensure that Rational Common Licensing server is configured with the IBM Architecture Room LIVE! license features.

   a. **Start the licensing agent on Linux:** Run the following command:

   ```
   java -Djava.library.path="." -jar DRLiveLicenseAgent.jar [server_port@license_server_host_add
   ```

   For example: `java -Djava.library.path="." -jar DRLiveLicenseAgent.jar` *27000@1.2.3.4 6666*, where *27000* is the license server port number, *1.2.3.4* is the host address, and *6666* is the license agent port number.

   b. **Configure IBM Architecture Room LIVE!**

   1) Update `licenseAgentURL` value in the `<live-install-folder>/user_config.js` file to include the license server agent details. For example: `licenseAgentURL: http://localhost:6666`

   2) Restart IBM Architecture Room LIVE!

      Note: If licensing is not configured, then IBM Architecture Room LIVE! will run on trial license mode for 60 days by default. Trial license will expire after 60 days, and users are warned about the same from two-weeks before the trial license expiration date and are reminded to configure the product license. To use the product after the trial license expires, you must activate a valid IBM Architecture Room LIVE! license.

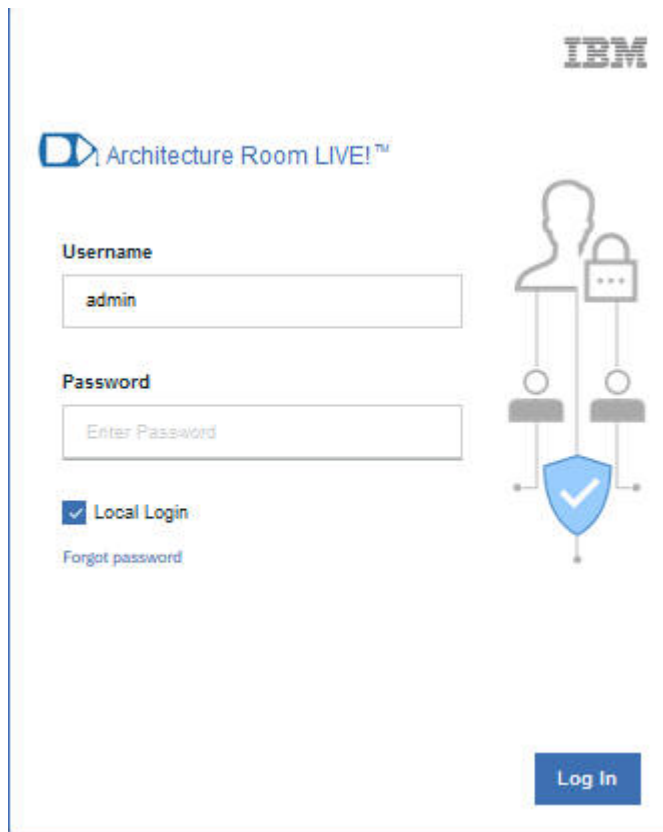9. Deploy the server by running the following command:

```
sudo node deploy-dr.js
```

The deployment script starts up the Mongo database (unless it is already running). It then starts the IBM Architecture Room LIVE! web server by using the PM2 process management application. The settings for how to deploy the server are stored in the PM2 configuration file `<live-install-folder>/DR_Install/pm2.config.js`. You can edit this file as needed.

The deployment script finishes by printing the URL of the IBM Architecture Room LIVE! web server. Test whether the installation and deployment was successful by opening the URL in a web browser. For example, the default settings in the `pm2.config.js` file yield the following URL:

```
https://localhost/index.html
```

where *localhost* is the server name where the application is running.

Notes: You might face the security certificate related issues while logging in from the browsers.

- If the Live! server does not have a valid SSL certificate and uses a self-signed certificate (by default it does), you might see the browser security warning. Add a browser security exception to the web page and continue.
- If the authentication fails because the Live! authentication server does not have a valid security certificate, you will see the *Authentication server is down* error when you login. To resolve the issue, see the *Invalid security certificate error during authentication* section in the Invalid security certificate errors topic.

Chapter 2. Migrating Live! 2019.04.x and earlier builds to the latest build

Live! was providing a single-room modeling capability in each installation until the 2019.04.x builds. If someone wanted to organize and maintain different projects, they had to install multiple instances of Live! for each project and had to download and move the data between the different projects manually. This was a very tedious process and error-prone. To overcome the problems and to support various projects in a single instance, the latest versions of Live! comes with multiple projects support that allows you to maintain multiple projects in different rooms. The multiple-projects support enables the administrator to manage and migrate users and whiteboards data easily across projects.

Before you begin

Note: This topic is intended for the Home and Room administrators who have the access to export and import data across Live! rooms.

Since Live! server runs on port 443 by default, you might end up having both the new and old versions of the product running on the same port, unless you changed them in both or in one of the versions. If both versions of the product run on the same port, you will lose the old instance of Live! That way you will not be able to compare and verify data of the new instance with the old data if want to. Hence for verification purposes, or if you want to retain both instances, you must ensure that you change the default authorization and audio server ports in either the old or new instance or both.

Important: If both the old and new instances of Live! are run on the same port (for example, port 443), the data on the old instance will be lost as it will be overwritten with the data of the new instance.

About this task

When you want to upgrade from Live! 2019.04.x and earlier builds to 2019.07 or later builds, you must ensure that the user and whiteboard data from the older version is migrated to the latest version.

Remember:
- If the administrator decides to retain the old version of Live! for comparing data and decides to modify its default server port, then ensure that the old instance of Live! is restarted after the port is changed.
- The administrator in the old instance of Live! must export the data of users and whiteboards before starting with the migration procedure.
- The administrator must also take a copy of the `<old-live-install-folder>\DR_Install\dr-config.json` file, `<old-live-install-folder>\lib_server\config` folders that contains the license keys and the certificates, and the `<old-live-install-folder>\user_config.js` file, so that the configuration information can be reused in the new installation.

Procedure
1. Download the `IBM_Architecture_Room_LIVE_<version>_setup.zip` file.

2. Extract the file to an empty folder `<new-live-install-folder>`. For example, the name of the ZIP file might be `IBM_Architecture_Room_LIVE_V9.7_setup.zip`.

   Notes:
   - The built-in `"Extract All..."` utility in Windows has a limitation on the length of the path names of the extracted files. You can reduce the risk of running into this limitation by extracting the compressed file into a folder where the full path name is short. If this doesn't help, use a better extraction utility, such as 7-zip.
   - Make sure to extract the files into an empty folder. Do not overwrite an existing IBM Architecture Room LIVE! installation, especially when the application is running.
   - Since this is an update to an existing IBM Architecture Room LIVE! installation, Node JS from the previous installation must be available. If not, perform step 2 in the Installation on Windows with internet access to install Node JS.

3. Copy the `drlive.crt` and `drlive.key` files from `<old-live-install-folder>\lib_server\config` folder to the following folders:
   - `<new-live-install-folder>\liveAuthServer\libsever\Config\`
   - `<new-live-install-folder>\liveHomeServer\libsever\Config\`
   - `<new-live-install-folder>\liveRoomServer\libsever\Config\`

4. Review the `<old-live-install-folder>\user_config.js` file and update the `<new-live-install-folder>\liveHomeServer\user_config.js` as required.

5. Goto the command prompt and run the following command as an administrator:

   `node deploy-dr.js`

   The new instance of Live! server is started.

6. Goto a web browser and type the following URL to start the application: `https://<hostname>/home.html`, where `<hostname>` is the name or IP address of the Live! home server.

7. Browse to the location of the exported data of the old Live! instance and select the folder that you want to import at `<old-install folder>\ExportImportScripts\data\<tagname.zip>`, where *<tagname.zip>* is the file that you want to import. Extract the *<tagname.zip>* file.

8. Copy the content of the *<tagname.zip>* file and paste it into the new Live! instance at `<new-live-install-folder>\liveRoomServer\ExportImportScripts\data\home`.

9. Log in as a home administrator.

10. To import the user data from the old Live! instance, goto the administration menu in the Home dashboard, and import the *<tagname.zip>* folder. For more information about how to import, see Importing groups and users.

11. To import the whiteboards and whiteboard templates from the old Live! instance into a specific room, click the room link or log in to the room as an administrator.

12. In the Room page, import the *<tagname.zip>* where *<tagname>* is the exported folder, to import the whiteboards and whiteboard templates data. For more information about how to import whiteboard data into rooms, see Importing whiteboards.

After the users and whiteboards are imported into the new Live! instance, you can compare the new user data on the Home dashboard and the new whiteboards data in the Rooms dashboard data with the old Live! instances.

Chapter 3. Updating IBM Architecture Room LIVE!

To update an existing IBM Architecture Room LIVE! installation to the latest version, perform the following steps:

Procedure

Note: When you plan to install in a new location or overwriting an existing version, ensure that you take a copy of the `<live-install-folder>\DR_Install\dr-config.json` file, `lib_server\config` folder that contains the license keys and the certificates, and `<live-install-folder>\user_config.js` file, so that you can reuse the configuration in the new installation.

1. Extract the file `IBM_Architecture_Room_LIVE_<version>_setup.zip` to an empty folder `<live-install-folder>`. For example, the name of the ZIP file might be `IBM_Architecture_Room_LIVE_V9.7_setup.zip`.

   Note:
   - The built-in "`Extract All...`" utility in Windows has a limitation on the length of the path names of the extracted files. You can reduce the risk of running into this limitation by extracting the compressed file into a folder where the full path name is short. If this doesn't help, use a better extraction utility, such as 7-zip.
   - Make sure to extract the files into an empty folder. Do not overwrite an existing IBM Architecture Room LIVE! installation, especially when the application is running.
   - Since this is an update to an existing IBM Architecture Room LIVE! installation, Node JS from the previous installation must be available. If not, perform step 2 in the Installation on Windows with internet access to install Node JS.

2. If you want to delete the old installation folder, you need to perform this step to ensure that there is no process running, which can prevent the folder from being deleted.

   a. Open an administrative command shell and run the following command:
      ```
      pm2 kill
      ```

      Note: Sometimes, running the `pm2 kill` command stops the Mongo database.

   b. If you find Mongo server still running after the previous step, then enter the Mongo shell by running the following command: `<mongo-install>\bin\mongo`, where *<mongo-install>* is specified either by `mongo-win-install-folder` or `mongo-linux-install-folder` in the file `<live-install-folder>\DR_Install\dr-config.json`.

   c. Run the following commands in the Mongo shell for stopping the Mongo database:
      ```
      use admin
      db.shutdownServer()
      ```

   d. Ensure that the version of Node JS is the same in the new version of the application that you are updating to. Go to a command prompt and run the following command:
      ```
      node --version
      ```

Compare the output of this command with the version of Node JS that is in the new installation. On Windows, the Node JS installer is located in `<live-install-folder>\server-sw\win32_x86_x64\`, and on Linux, it is located in `<dr-install-folder>/server-sw/linux_x86_x64/`. If the version is not the same, you should uninstall Node JS (using the normal procedure for uninstalling installed programs) and then install the new Node JS version.

e. Copy the following files from the old installation or from the backup location to the new installation folder:

- `dr-config.json` file from the old installation (`<live-install-folder>\DR_Install\`) to the new installation (`<live-install-folder-new>\DR_Install\`)
- `config` folder from (`<live-install-folder>\lib_server\`) to (`<live-install-folder-new>\lib_server\`)
- `user_config.js` file from (`<live-install-folder>\`) to (`<live-install-folder-new>\`)

f. Open an administrative command shell and navigate to `cd <live-install-folder>\DR_Install\`, and run the following command:

`node deploy-dr.js`

g. You can delete the old install folder (`<live-install-folder>`) at this point.

The deployment script first stops the currently running IBM Architecture Room LIVE! web server. However, it does not stop the PM2 daemon process or the Mongo database. That is the reason you need to perform step 2 if you want to avoid lingering processes that prevent the old installation folder from being deleted.

The deployment script then starts the new version of the server. The version of Node JS, Mongo, and PM2 remains the same.

Chapter 4. Uninstalling IBM Architecture Room LIVE!

There is no script for uninstalling the IBM Architecture Room LIVE! web server, so you need to perform the following steps manually (from an administrative command prompt):

Procedure

Run the following commands from the command prompt from the `<live-install-folder>\DR_Install` folder:

1. `pm2 delete pm2.config.js`

   Stops the IBM Architecture Room LIVE! web server and deletes it from PM2 list of managed processes.

2. `pm2 kill`

   Stops the PM2 daemon process.

3. `npm uninstall -g pm2`

   Uninstalls PM2. If it is still available in the PATH after this command, you must manually remove the PM2 files. Use where pm2 (Windows) or which pm2 (Linux) to find out where it is located.

4. Enter the Mongo shell by running `<mongo-install>\bin\mongo`, where `<mongo-install>` is specified either by `mongo-win-install-folder` or `mongo-linux-install-folder` in the file `<live-install-folder>\DR_Install\dr-config.json` folder.

   Run the following commands in the Mongo shell for stopping the Mongo database:

   ```
   use admin
   db.shutdownServer()
   ```

5. On Windows, you can uninstall Mongo using the following command:

   `msiexec /uninstall <live-install-folder>\DR_Install\downloaded\mongo-win.msi`

   On Linux, the commands for uninstalling Mongo depend on your Linux distribution; see the Mongo documentation.

6. Now you can remove the IBM Architecture Room LIVE! server by deleting the `<live-install-folder>` folder.

Chapter 5. Advanced configurations

User can customize the following configurations by editing the `dr-config.json` file that is available in the `<live-install-folder>\DR_Install` folder.

- Mongo installation folder can be updated as follows: `mongo_win_install_folder` OR `mongo_linux_install_folder` as needed based on the operating system.
- Mongo data folder can be updated as follows: `mongo_win_data_folder` OR `mongo_linux_data_folder` as needed based on the operating system.

SSL Certification

The IBM Architecture Room LIVE! web server uses HTTPS, and you need to decide the SSL certificate that must be used. You can choose one of the following two options.

Certificate configuration

- Use your own SSL certificate. This is the recommended approach for a IBM Architecture Room LIVE! installation that is intended for a production environment. The connection to the server will be secure and users will not get any warnings from the browser.

  In some instances, you might have to extract the `.crt` and `.key` files from the `.pfx` file, so that the certificate and the decrypted key files are available separately in a plain text format. For more information. see "Extracting the certificate and keys from a `.pfx` file" on page 24.

- Use a self-signed SSL certificate. You can use this approach if you don't yet have your own SSL certificate to use. However, in this case users will get warnings from the web browser about the use of a self-signed certificate as the server will not be considered secure.

The IBM Architecture Room LIVE! installation contains the self-signed certificates in the following locations:

- home server: `<live-install-folder>\liveHomeServer\lib_server\config`
- authentication server: `<live-install-folder>\liveAuthServer\lib_server\config`
- room server: `<live-install-folder>\liveRoomServer\lib_server\config`

Replace the certificate `drlive.crt` and `drlive.key` files with your self-signed certificates and keys for all three servers. The web servers will then use the self-signed certificate by default.

Port configuration

- SSL port:

  By default, IBM Architecture Room LIVE! uses the port *443*. To change the default port, update the SSL port parameter **sslPort:<portnumber>** in the `<live-install-folder>\liveHomeServer\user_config.js` file.

- Audio server port:

  By default, the audio server uses the port *452*. To change the default port, update the audio port parameter **audioPort:<portnumber>** in the `<live-install-folder>\liveHomeServer\user_config.js` file.

Extracting the certificate and keys from a `.pfx` file

The `.pfx` file, which is in a PKCS#12 format, contains the SSL certificate (public keys) and the corresponding private keys. Sometimes, you might have to import the certificate and private keys separately in an unencrypted plain text format to use it on another system. This topic provides instructions on how to convert the `.pfx` file to `.crt` and `.key` files.

Extract `.crt` and `.key` files from `.pfx` file

PREREQUISITE: Ensure OpenSSL is installed in the server that contains the SSL certificate.

1. Start OpenSSL from the `OpenSSL\bin` folder.
2. Open the command prompt and go to the folder that contains your `.pfx` file.
3. Run the following command to extract the private key:

   ```
   openssl pkcs12 -in [yourfile.pfx] -nocerts -out [drlive.key]
   ```

   You will be prompted to type the import password. Type the password that you used to protect your keypair when you created the `.pfx` file. You will be prompted again to provide a new password to protect the `.key` file that you are creating. Store the password to your key file in a secure place to avoid misuse.
4. Run the following command to extract the certificate:

   ```
   openssl pkcs12 -in [yourfile.pfx] -clcerts -nokeys -out [drlive.crt]
   ```
5. Run the following command to decrypt the private key:

   ```
   openssl rsa -in [drlive.key] -out [drlive-decrypted.key]
   ```

   Type the password that you created to protect the private key file in the previous step.

   The `.crt` file and the decrypted and encrypted `.key` files are available in the path, where you started OpenSSL.

Convert `.pfx` file to `.pem` format

There might be instances where you might have to convert the `.pfx` file into `.pem` format. Run the following command to convert it into PEM format.

```
openssl rsa -in [keyfile-encrypted.key] -outform PEM -out [keyfile-encrypted-pem.key]
```

Note: Ensure that the name of the certificate file is `drlive.crt` and the private key file is named `drlive.key`. The renamed `drlive.crt` and `drlive.key` files must be updated for all three servers:

- liveAuthServer: `<live-install-folder>\liveAuthServer\lib_server\config`
- liveHomeServer: `<live-install-folder>\liveHomeServer\lib_server\config`
- liveRoomServer: `<live-install-folder>\liveRoomServer\lib_server\config`

Live! server configurations

The `<live-install-folder>\liveAuthServer\user_config.js` and `<live-install-folder>\liveHomeServer\user_config.js` files list the default values of the configurable parameters for the Live! authentication and home servers. You can change the values for some of the parameters as requirement.

Configurations in the `<live-install-folder>\liveAuthServer\`
`user_config.js` file

The following table shows the configurable parameters and their default values:

Parameter name	Default value
authPort	51201 **Limitation:** In this release, Live! home browser client uses the default port *51201* and cannot be modified.
mongodbAuthURL	`mongodb://127.0.0.1:27017/LiveAuth`
passwordRecoveryPassphrase	*recover live! auth password @live*

Configurations in the `<live-install-folder>\liveHomeServer\`
`user_config.js` file

The following table shows the configurable parameters and their default values:

Parameter name	Default value
mongodbURL	*mongodb://127.0.0.1:27017/Live*
authServerURL	`https://localhost:51201`, where *51201* is the default port
homeServerURL	https://localhost
sslPort	443
audioPort	452
licenseAgentURL	"", // e.g. 'http://localhost:XXXX'
maxNumberOfRooms	16
maxNumberOfSessionsPerRoom	50
enableTracking	true
strictSSL	true

Log files

You can view the log messages of the IBM Architecture Room LIVE! web servers such as the home, authentication, and audio servers in their respective log files. For example, `livehome-err.log` and `livehome-out.log` are two log files of the home web server. Besides the web server logs, the Mongo database related log files are available in the following path: `<live-install-folder>\DR_Install\mongoinstall.log`.

- By default, the log messages are directed to the `<live-install-folder>\logs\` folder. The logs folder contains the logs of the Mongo database and the web servers such as authentication, home, and audio servers. Each web server will have two log files; one that contains the output and the other that contains the error messages. For example, `<live-install-folder>\logs\liveauth-out.log` and `<live-install-folder>\logs\liveauth-err.log`. You can provide custom log file names by editing the properties `out_file` and `error_file` in the `<live-install-folder>\DR_Install\pm2.config.js` file.

- Log messages are also produced by the Mongo database. By default, these messages are printed to `<live-install-folder>\DR_Install\mongoinstall.log`.

You can use a different log file by editing the property `mongo_log_file` in the `<live-install-folder>\DR_Install\dr-config.json` file.

Chapter 6. Troubleshooting

This topic provides solutions or workarounds for some of the issues that you might encounter while working with Live!

Security certificate errors

Live! installation must use a valid SSL certificate. In the absence of a valid certificate, you might be using a self-signed security certificate, which some of the web browsers such as Firefox might not recognize. In such cases, the web browser displays the "Authentication server is down" error message and will not allow you to login. To resolve the authentication problem, you have to add a security exception to the site and accept the invalid certificate to proceed to login.

Invalid security certificate error during authentication:

While you log in, you might come across the "Authentication server is down" error if the server does not have a valid security certificate. Until you obtain a valid certificate, you can add a security exception to the server so that you can proceed to log in to Live!

1. Goto the web browser window and type the URL with Live! authentication server name and port: `https://<server>:<port>/`, where *<server>* is the name or IP address and *<port>* is the port number of the authentication server. For example: `https://<server>:51201/`, where 51201 is the default authentication port number. For the default configurations and their values, see, "Live! server configurations" on page 24.
2. Add a security exception to the site and confirm the security exception.
3. Open another web browser and start Live! by typing the URL: `https://<server>/home.html`, where *<server>* is the name or IP address of the Live! server.
4. Live! home page is displayed.

Invalid security certificate error while starting Live! audio:

Live! audio server uses port 452 by default for audio. If the audio server does not have a valid security certificate, you might encounter audio connection issue. Until you obtain a valid certificate, you can add a security exception to the audio server so that you can use Live! audio feature.
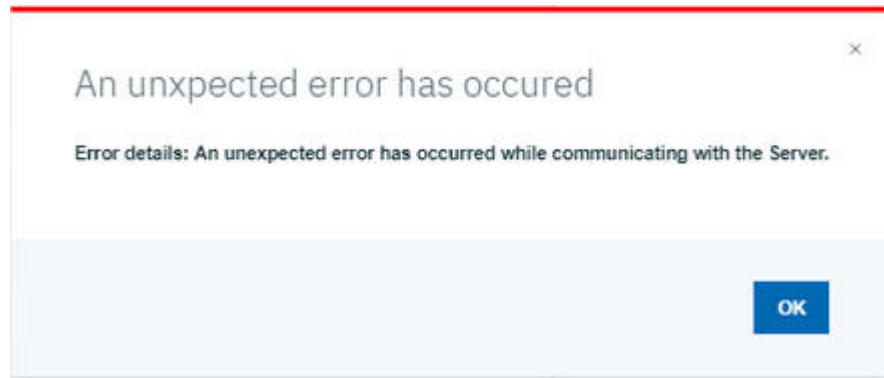
1. Goto the web browser window and type the URL with Live! audio server name and port: `https://<server>:<port>/`, where *<server>* is the name or IP address and *<port>* is the port number of the Live! audio server. For example: `https://<server>:452/`, where 452 is the default audio port number. For the default configurations and their values, see, "Live! server configurations" on page 24.
2. Add a security exception to the site and confirm the security exception.
3. Click Call icon to start or join the call. Audio server will be connected.

Active directory server connection error

When the connection to the active directory server is not established, you will see a server connection error when you access the rooms, groups, and users pages on the Live! Home dashboard.

Connection to the active directory server fails:

You will see the following error message when the connection to the active directory server fails:

To resolve the connection issue, ensure that the active directory server is connected and reachable.