INSTALLATION GUIDE ON WINDOWS

Table of Contents

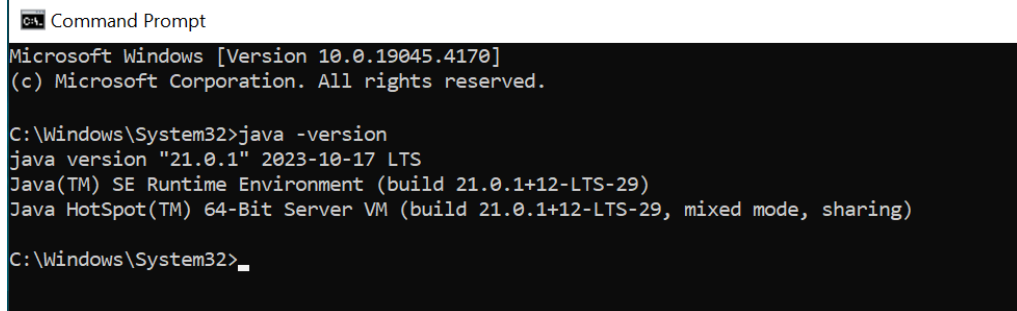# Installation Guide on Windows

1. Pre-Requisites

1.1. Java (JVM)

Elixir Ambience/Repertoire Server requires Java 17 LTS (Long Term Support) or higher. The Java Platform, standard edition 17 development kit. If you do not have Java installed, you may obtain a copy from OpenJDK or Oracle.

You may follow installation instruction from your chosen vendor:

a) OpenJDK:
   https://openjdk.org/install/

b) Oracle Java:
   https://www.oracle.com/sg/java/technologies/downloads/

To check if the Java version is installed correctly, run the following command using a Windows Command Line tool, with the following output observed.

2. Installation

2.1. MongoDB

MongoDB installation is for Ambience software only.

2.1.1. Download MongoDB For Windows

Download the release 4.2.7 of MongoDB (*MongoDB.msi*) or higher from

https://www.mongodb.com/download-center/community

2.1.2. Installation Of MongoDB

Double-click the *MongoDB.msi* file. A set of the screens will appear to guide you through the installation process.

Install the MongoDB as a service and fill up your data and log directories accordingly.

You may proceed with the default option.

You will receive a success message upon completing the installation.

Alternatively, you can follow the full installation guide here:
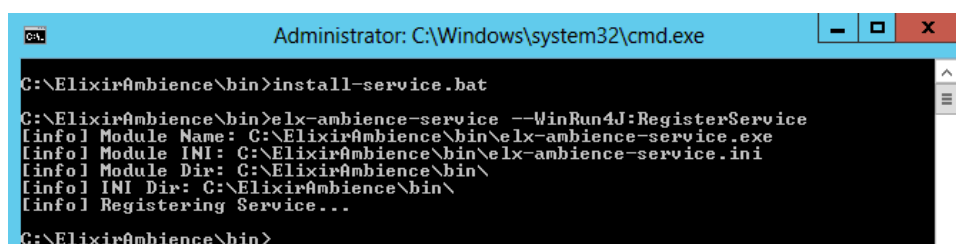
https://docs.mongodb.com/manual/installation/#mongodb-community-edition-installation-tutorials

2.2. Elixir Ambience/Repertoire Server

2.2.1. Installing Elixir Ambience/Repertoire As A Service

Below are two examples of how to set up a mail server. In these examples, Ambience is used.

1. Login to Windows Server as an administrator.

2. Unzip the *elx-ambience-windows-4.4.0.zip* file into *C:* or your selected file path. An *ElixirAmbience* folder would be created under the C drive or to your selected file path.

3. Navigate to "ElixirAmbience\bin" folder using the Windows Explorer and open the *elx-ambience-service.xml* file with Notepad. Change the *name* value to Elixir-Ambience. Edit the *description* value as desired and save the file.

4. With the Windows Explorer still in the "ElixirAmbience\bin" folder, hold the "Shift" key and right click in the folder to select *Open command window here*. The command window would be opened.

5. Type the command *install-service.bat* in the command line tool and the ElixirAmbience Service would be installed.
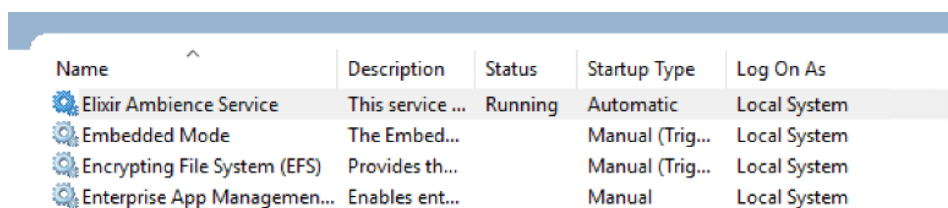

6. Open the System Services panel and the Elixir Ambience Service would appear.

7. Double-click on the *Elixir Ambience Service* to open its properties, under the *General* screen. Change the *Startup* type to *Automatic* and click *Apply* to set the service auto start while the system restarts.

2.3. Configure Ambience/Repertoire Server

2.3.1. Accounts and Permissions

You need to use administrator account for Ambience/Repertoire installation. You may use service account for daily operations.

The service account must have the following file path permission and access:

Permission and access	Reason
Java installation read access	For daily operation of the application
Ambience/Repertoire installation path read access	For daily operation of the application
Ambience/Repertoire installation path root */log* folder write access	For writing logs into a file path
Ambience/Repertoire installation path root */data/out* write access	All export/download file will be written into */data/out*. You may define another location in the *application.conf* file.

2.3.2. Encryption

All plain text passwords can be encoded by using *{enc}XXXX* style logic. This is intended to obfuscate the text to prevent casual/accidental viewing. The config file (and indeed the whole of Ambience/Repertoire) itself should only be readable by the user, so nobody else should be able to see the contents in the first place.

Any time a password changes, the server needs to be restarted.

Ambience/Repertoire (all services) supports encrypting sensitive data like passwords by using the prefix *{enc}* in the configuration. You can then use an encryption mechanism we supply to generate the encrypted string.

You need to replace the encrypted string into *etc\application.conf* file.

2.3.3. MongoDB Connection (Ambience Only)

You may configurate the connection between MongoDB and Elixir Ambience by modifying the following section under *\etc\application.conf* file.

```
elixir.data.mongodb {
  default {
    connectionString = "mongodb://"${mongodb}":27017"
    database {
      eno = "eno"
    }
  }
}
```

Edit default (you can change the name) to match your MongoDB server connection.

The connection string can be encrypted using *{enc}* syntax.

Add a new key like "default" to add multiple MongoDB servers - the name is purely descriptive. Database **key=value**, keys must be unique throughout all MongoDB connections, value is an actual (mongodb) database name.

Lookup is by key (unique) which is an alias to the physical database name, allowing you to easily switch to point to a different physical database (perhaps on a different server) without changing every caller.

2.3.4. HTTPS Configuration

To enable HTTPS connection, a SSL certificate is required to identify the Ambience/Repertoire server. This certificate needs to be placed within a keystore that Ambience/Repertoire can use to set up the HTTPS service.
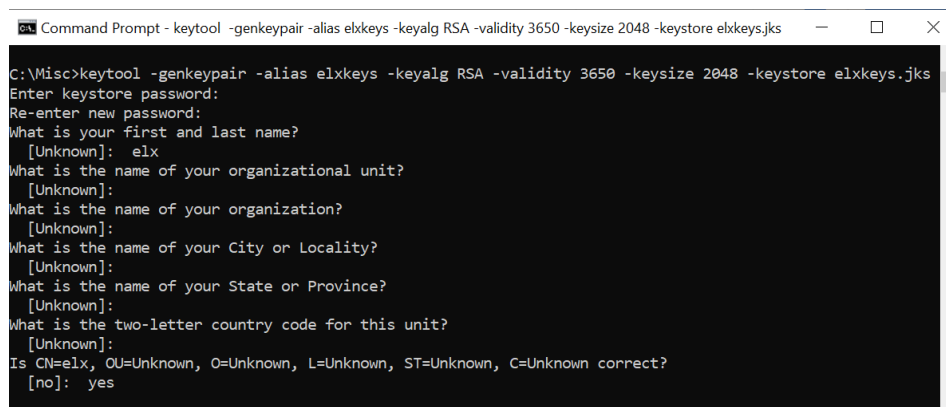
To do so, perform the following:

1. Generate the keystore certificate.

    In the Ambience/Repertoire folder *\etc\https* or any folder, open a command prompt. Key in the following command:

    ```
    keytool -genkeypair -alias elxkeys -keyalg RSA -validity 3650 -
    keysize 2048 -keystore elxkeys.jks
    ```

    When prompted for the keystore password, key in `changeit`. Do note that the password will not appear. Key in the other information as desired when prompted.

    Do note that `elxkeys` is used in this example. You can use any name.
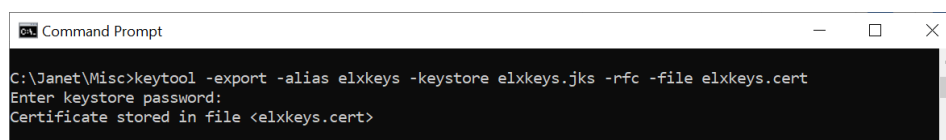

    Extract the certificate using the following command:

    ```
    keytool -export -alias elxkeys -keystore elxkeys.jks -rfc -file
    elxkeys.cert
    ```

    Key in the password `changeit` when prompted.

    Now you should have two files *elxkeys.jks* and *elxkeys.cert* in the folder.
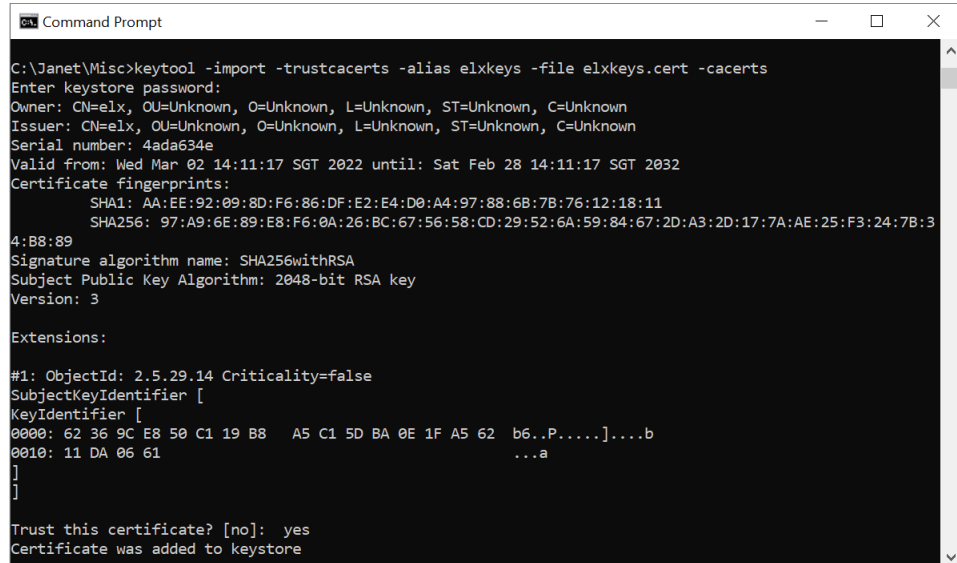
2. Add the certificate to cacerts.

Go to the folder where the **cacerts** file is located. In Windows, this file is located in the Java home directory, under the \lib\security folder. Make a backup copy of the file.

Go back to the Ambience/Repertoire folder *\etc\https* and key in the following command in the command prompt:

```
keytool -import -trustcacerts -alias elxkeys -file elxkeys.cert -cacerts
```

Key in the password `changeit` when prompted. Answer `yes` to trust the certificate.

The **cacerts** has been updated now and Java will trust the certificate when the authentication client connects to the SSO server via HTTPS.

3. Copy the keystore file (**elxkeys.jks**) in the Ambience/Repertoire folder *\etc\https*.

Ignore this step if the keystore file is created in this folder.

4. Enable HTTPS connection in the */etc/application.conf* file.

```
ambience {
  bindAddress: "0.0.0.0"
  bindPort: ${internal-port}
  https {
    enabled = true
    keystore = "https/elxkeys.jks"
    keystore-type = "JKS"
    keystore-password = "changeit"
  }
}
```

You may want to use another port number for HTTPS connection. See note below.

5. Change the url to HTTPS in the *\etc\application.conf* file.

```
ambience.web {
  url = "https://"${host}":"${port}
  response-headers {
    #any of these can be overridden either by supplying an
alternate string or else "" to turn the header off
    X-XSS-Protection = "1; mode=block"
    X-Frame-Options = "SAMEORIGIN"
    X-Content-Type-Options = "nosniff"
  }
}
```

6. If the SSO service is chosen as the preferred service, change the ***requiresPKCE = true*** into the SSO client section in the *\etc\application.conf* file. Otherwise, leave it as the default (false).

```
elixir.sso.client {
  ...
  service-definition {
    elxsso {
      ...
      logout = ${sso-server-based}"/simple-sso/logout"
      requiresPKCE = true
      debug = true
      ...
    }
  }
}
```

7. In the web browser, key in the URL. The Ambience/Repertoire software will be launched in HTTPS mode.

Note:

By default, Ambience uses port 1740, which is used by the HTTP mode. IT is recommended to change to another port, say 1741, for HTTPS mode.

For Repertoire, port 1730 is used. The same method can be used for Repertoire for HTTPS mode.

Depending on the browser, an error may occur when switching from HTTPS back to HTTP mode, as the HTTP cookie may not be able to replace the HTTPS cookie.

If you do not wish to change the port number, you can delete the cookie in the browser before using HTTP. For example, in the Chrome browser, go to *Chrome Settings -> Privacy and Security* option, locate the cookie and delete it. Once the HTTPS cookie was deleted, you could log in again via HTTP using the same host and port.

Therefore, it is advised to change both the internal and external port in the *application.conf* file to ensure that the HTTPS cookie do not prevent you from reverting to HTTP, because HTTP will be on a separate port with a distinct set of cookies.