

INSTALLATION GUIDE ON WINDOWS

Table of Contents

1. Pre-Requisites	2
1.1. Java (JVM)	2
2. Installation	3
2.1. MongoDB	3
2.1.1. Download MongoDB For Windows	3
2.1.2. Installation Of MongoDB	3
2.2. Elixir Ambience Server	4
2.2.1. Installing Elixir Ambience As A Service	4
2.3. Configure Ambience Server	5
2.3.1. Accounts and Permissions	5
2.3.2. Encryption	5
2.3.3. MongoDB Connection	5
2.3.4. HTTPS Configuration	6

Installation Guide on Windows

1. Pre-Requisites

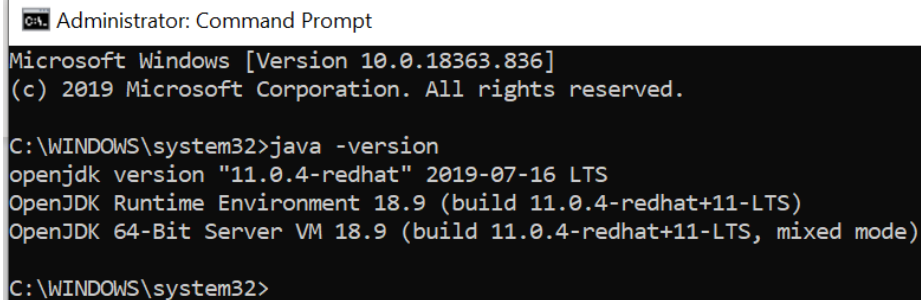
1.1. Java (JVM)

Elixir Ambience Server requires Java 11 LTS (long Term Support) . The Java Platform, standard edition 11 development lit. 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://adoptopenjdk.net/installation.html?variant=openjdk11&jvmVariant=hotspot>
- b) Oracle Java:
<https://docs.oracle.com/en/java/javase/11/install/installation-jdk-microsoft-windows-platforms.html#GUID-DAF345BA-B3E7-4CF2-B87A-B6662D691840>

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

A screenshot of a Windows Command Prompt window titled "Administrator: Command Prompt". The window shows the output of the command "java -version". The output text is: "Microsoft Windows [Version 10.0.18363.836] (c) 2019 Microsoft Corporation. All rights reserved. C:\WINDOWS\system32>java -version openjdk version "11.0.4-redhat" 2019-07-16 LTS OpenJDK Runtime Environment 18.9 (build 11.0.4-redhat+11-LTS) OpenJDK 64-Bit Server VM 18.9 (build 11.0.4-redhat+11-LTS, mixed mode) C:\WINDOWS\system32>".

```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.18363.836]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>java -version
openjdk version "11.0.4-redhat" 2019-07-16 LTS
OpenJDK Runtime Environment 18.9 (build 11.0.4-redhat+11-LTS)
OpenJDK 64-Bit Server VM 18.9 (build 11.0.4-redhat+11-LTS, mixed mode)

C:\WINDOWS\system32>
```

2. Installation

2.1. MongoDB

2.1.1. Download MongoDB For Windows

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

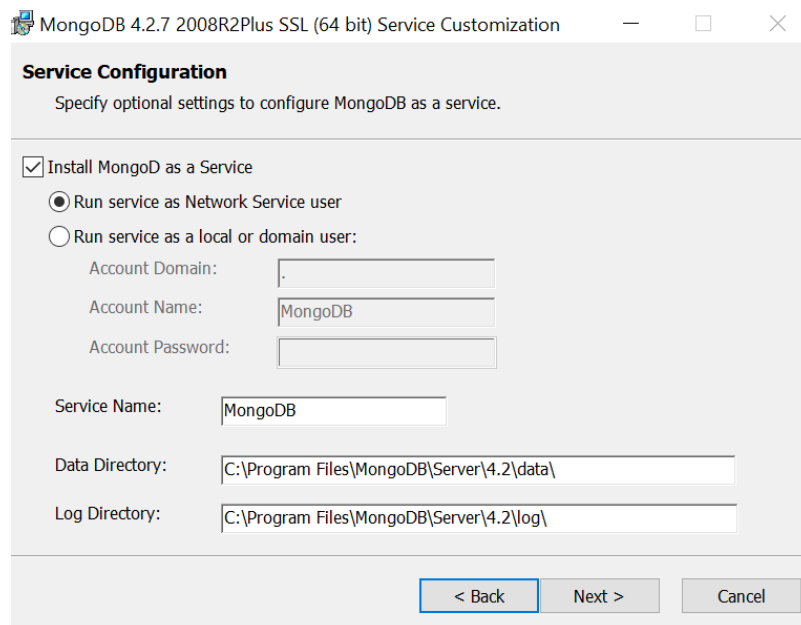
<https://www.mongodb.com/download-center/community>

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:

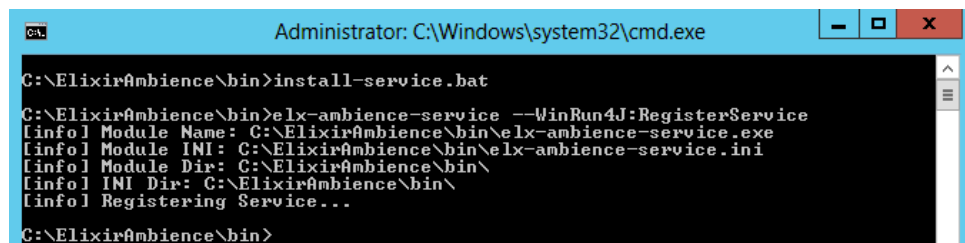
<https://docs.mongodb.com/manual/installation/#mongodb-community-edition-installation-tutorials>

2.2. Elixir Ambience Server

2.2.1. Installing Elixir Ambience As A Service

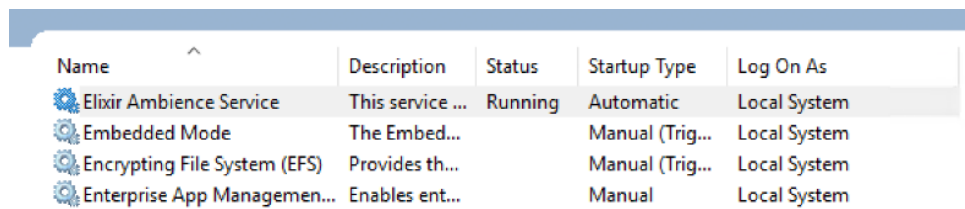
Below are two examples of how to set up a mail server.





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.



```
C:\ElixirAmbience\bin>install-service.bat
C:\ElixirAmbience\bin>elx-ambience-service --WinRun4J:RegisterService
[info] Module Name: C:\ElixirAmbience\bin\elx-ambience-service.exe
[info] Module INI: C:\ElixirAmbience\bin\elx-ambience-service.ini
[info] Module Dir: C:\ElixirAmbience\bin\
[info] INI Dir: C:\ElixirAmbience\bin\
[info] Registering Service...
```

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



Name	Description	Status	Startup Type	Log On As
 Elixir Ambience Service	This service ...	Running	Automatic	Local System
 Embedded Mode	The Embed...		Manual (Trig...	Local System
 Encrypting File System (EFS)	Provides th...		Manual (Trig...	Local System
 Enterprise App Managemen...	Enables ent...		Manual	Local System

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 Server

2.3.1.Accounts and Permissions

You need to use administrator account for Ambience 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 installation path read access	For daily operation of the application
Ambience installation path root /log folder write access	For writing logs into a file path
Ambience installation path root /data/out write access	All export/download file will be written into /data/out. You may define another location in the <i>application.conf</i> 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) itself should only be readable by the Ambience 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 (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.

```

Administrator: Command Prompt

C:\Users\He_Xi\Desktop\Ambience-5.3.0-SNAPSHOT\bin>ambience-cli encrypt hello
hello = {enc}2Y8qleTKJ2PKmJ/UEFRByw==
C:\Users\He_Xi\Desktop\Ambience-5.3.0-SNAPSHOT\bin>

```

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

2.3.3.MongoDB Connection

You may configure 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 identity the Ambience server. This certificate needs to be placed within a keystore that Ambience can use to set up the HTTPS service. To do so, perform the following:

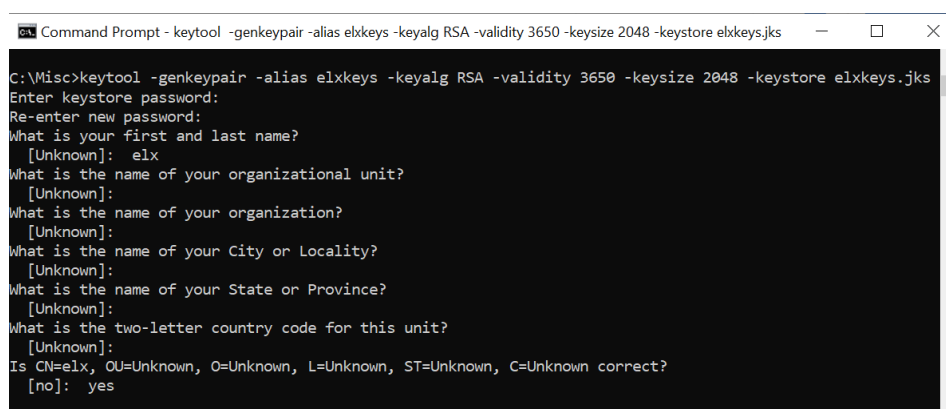
1. Generate the keystore certificate.

In the Ambience folder `letc\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.

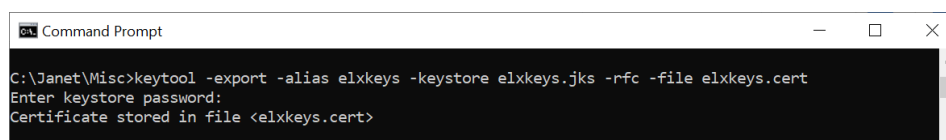


```
Command Prompt - keytool -genkeypair -alias elxkeys -keyalg RSA -validity 3650 -keysize 2048 -keystore elxkeys.jks
C:\Misc>keytool -genkeypair -alias elxkeys -keyalg RSA -validity 3650 -keysize 2048 -keystore elxkeys.jks
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: elx
What is the name of your organizational unit?
[Unknown]:
What is the name of your organization?
[Unknown]:
What is the name of your City or Locality?
[Unknown]:
What is the name of your State or Province?
[Unknown]:
What is the two-letter country code for this unit?
[Unknown]:
Is CN=elx, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown correct?
[no]: yes
```

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.



```
Command Prompt
C:\Janet\Misc>keytool -export -alias elxkeys -keystore elxkeys.jks -rfc -file elxkeys.cert
Enter keystore password:
Certificate stored in file <elxkeys.cert>
```

Now you should have two files `elxkeys.jks` and `elxkeys.cert` in the folder.

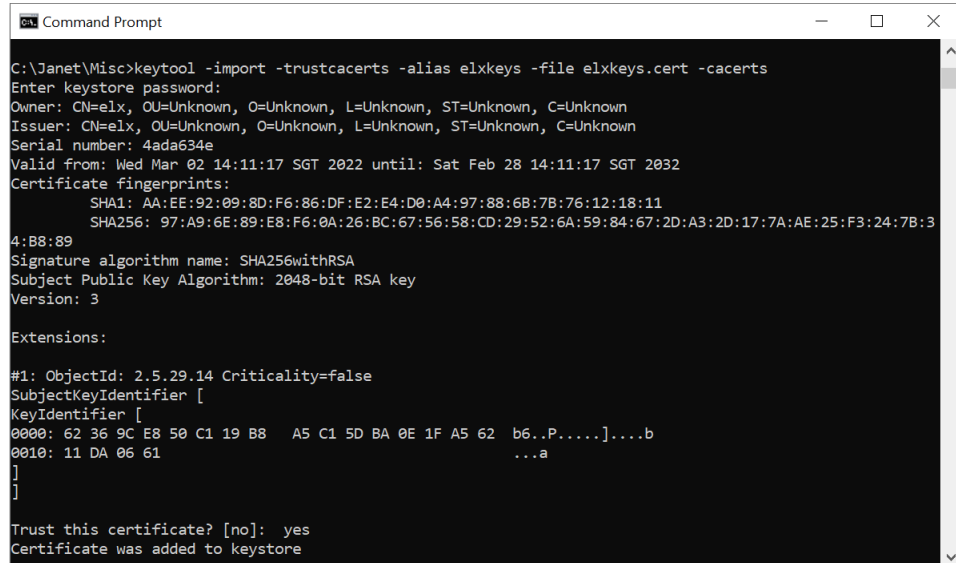
2. Add the certificate to cacerts.

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 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.



```

C:\Janet\Misc>keytool -import -trustcacerts -alias elxkeys -file elxkeys.cert -cacerts
Enter keystore password:
Owner: CN=elx, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown
Issuer: CN=elx, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown
Serial number: 4ada634e
Valid from: Wed Mar 02 14:11:17 SGT 2022 until: Sat Feb 28 14:11:17 SGT 2032
Certificate fingerprints:
    SHA1: AA:EE:92:09:8D:F6:86:DF:E2:E4:D0:A4:97:88:6B:7B:76:12:18:11
    SHA256: 97:A9:6E:89:E8:F6:0A:26:BC:67:56:58:CD:29:52:6A:59:84:67:2D:A3:2D:17:7A:AE:25:F3:24:7B:3
4:B8:89
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 2048-bit RSA key
Version: 3

Extensions:
#1: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: 62 36 9C E8 50 C1 19 B8   A5 C1 5D BA 0E 1F A5 62   b6..P.....]....b
0010: 11 DA 06 61                   ...a
]
]

Trust this certificate? [no]: yes
Certificate was added to keystore
  
```

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 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 `letclapplication.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 `letclapplication.conf` file. Otherwise, leave it as the default (false).

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

7. In the web browser, key in the URL. The Ambience 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.

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.