How to Install Liferay on CentOS 7

24th May 2017  6,312k

Liferay is a free and open source enterprise portal software. It is written in Java and uses MySQL to store its data. Liferay is a suite of different business related applications. It consists of many different features such as Alerts and Announcements, document library integration, Asset Publishing, Blogs, Calendar, Chat, Document and Image management, Knowledge Base, LDAP Integration, Mail, Message Boards, Polls, Software Catalog, Themes, User Directory, Web Content, Web Form Builder, WebDAV Integration, Wiki etc. It can be used in different sectors of industries like banking, healthcare, government, retail or manufacturing etc. This application is very popular and used by many prominent companies like Allianz, HP, Domino's, BOSCH, US Bank etc.

In this tutorial, we will install Liferay on CentOS 7 server.

Requirements

Liferay requires at least 2GB of RAM to run Java. All the required dependencies will be installed throughout the tutorial. You will need a minimal installation of CentOS 7 with root access on it. If you are logged in as a non-root user, you can run sudo -i to switch to root user.

Installing Liferay

Before installing any package it is recommended that you update the packages and repository using the following command.

yum -y update

Liferay runs on both OpenJDK and Oracle JAVA. In this tutorial, we will install the latest version of Oracle Java into the server. Run the following commands for same.

wget --no-cookies --no-check-certificate --header "Cookie:oraclelicense=accept-securebackup-cookie" "http://download.oracle.com/otn-pub/java/jdk/8u131-b11/d54c1d3a095b4ff2b6607d096fa80163/jdk-8u131-linux-x64.rpm"

If you do not have wget installed, you can run the yum -y install wget to install wget. Now install the downloaded RPM using the following command.

yum -y localinstall jdk-8u131-linux-x64.rpm

You can now check the java version using the following command.

java -version

You will get the following output.

[root@liptan-pc ~]# java -version

java version "1.8.0\_131"

Java(TM) SE Runtime Environment (build 1.8.0\_131-b11)

Java HotSpot(TM) 64-Bit Server VM (build 25.131-b11, mixed mode)

Once you install Java, you can download the Liferay application bundled with Apache Tomcat. Apache Tomcat is an application server used to serve java application. Run the following command to download Liferay.

cd /var wget https://downloads.sourceforge.net/project/lportal/Liferay%20Portal/7.0.2%20GA3/liferay-ce-portal-tomcat-7.0-ga3-20160804222206210.zip -O liferay.zip

You can always look for the latest link to the application on [Liferay download page](https://www.liferay.com/downloads).

Extract the archive using the following command.

unzip liferay.zip

If you don't have unzip installed, you can run yum -y install unzip. The above command will extract the content of the archive to liferay-ce-portal-7.0-ga3/ directory. Rename the directory to liferay using the following command.

mv liferay-ce-portal-\* liferay

To use MySQL database with the application, you will need to install MySQL first. Run the following command to install MySQL/MariaDB on your system.

yum -y install mariadb mariadb-server

To start MariaDB and enable it to start at boot time using the following commands.

systemctl start mariadb systemctl enable mariadb

Now run the following commands to secure your MySQL or MariaDB installation.

mysql\_secure\_installation

It will run a small script which asks you to provide the root password for MariaDB. As we have just installed MariaDB, the root password is not set, just press enter to proceed further. It will ask you if you want to set a root password for your MariaDB installation, choose y and set a strong password for the installation. It will also ask you for removing test databases and anonymous users. Most of the questions are self-explanatory and you should answer yes or y to all the questions.

Now you will need to create a new file to enter MySQL details. Run the following command to create a new portal-ext.properties.

nano /var/liferay/tomcat-8.0.32/webapps/ROOT/WEB-INF/classes/portal-ext.properties

If you do not have nano installed, you can run yum -y install nano. Now paste the following text into the nano editor. Now paste the following lines into the file. Change your MySQL root password with the password you set earlier.

jdbc.default.driverClassName=org.mariadb.jdbc.Driver

jdbc.default.url=jdbc:mariadb://localhost/lportal?useUnicode=true&characterEncoding=UTF-8&useFastDateParsing=false

jdbc.default.username=root

jdbc.default.password=StrongPassword

schema.run.enabled=true

schema.run.minimal=true

Now that the database server is ready, you can set up a systemd service so that it can be managed as a service and can be automatically started on failures and boot time. Run the following command to create a systemd service file.

nano /etc/systemd/system/liferay.service

Copy and paste the following code into the editor.

[Unit]

Description=Liferay Tomcat service

After=syslog.target network.target[Service]

Type=forking

ExecStart=/var/liferay/tomcat-8.0.32/bin/startup.sh

ExecStop=/var/liferay/tomcat-8.0.32/bin/shutdown.sh

Restart=always

User=root

Group=root

StandardOutput=syslog

StandardError=syslog

SyslogIdentifier=liferay

[Install] WantedBy=multi-user.target

Now you can start Liferay server and configure it to automatically start at boot time using following commands.

systemctl start liferay systemctl enable liferay

To check if Liferay service is started correctly, you can run the following command to check the status of the Liferay service.

systemctl status liferay

You should get output similar to shown below.

[root@liptan-pc ~]# systemctl status liferay

● liferay.service - Liferay Tomcat service

Loaded: loaded (/etc/systemd/system/liferay.service; enabled; vendor preset: disabled)

Active: active (running) since Mon 2017-05-22 19:27:22 UTC; 10s ago

Process: 16141 ExecStart=/var/liferay/tomcat-8.0.32/bin/startup.sh (code=exited, status=0/SUCCESS)

Main PID: 16155 (java)

CGroup: /system.slice/liferay.service

└─16155 /usr/bin/java -Djava.util.logging.config.file=/var/liferay/tomcat-8.0.32/conf/logging.properties -Dja...

May 22 19:27:22 liptan-pc systemd[1]: Starting Liferay Tomcat service... May 22 19:27:22 liptan-pc liferay[16141]: Tomcat started. May 22 19:27:22 liptan-pc systemd[1]: Started Liferay Tomcat service.

You can now access your web application on the following URL through your favorite web browser.

http://Server-IP-address:8080

Instead of using the application on Port 8080, you can use Apache to run on port 80 or 443 in the case of using SSL. This way the main application will run on port 8080, and Apache will work as a reverse proxy. Install Apache web server and mod\_ssl using the following command.

yum -y install httpd mod\_ssl

Now start Apache web server and enable it to automatically start at boot time using the following commands.

systemctl start httpd systemctl enable httpd

Now we will need to generate SSL certificates from Let's Encrypt client. If you can want to use commercial SSL certificates instead, you can purchase [SSL certificates from HostPresto](https://hostpresto.com/ssl-certificates/).

To install Let's Encrypt client also called Certbot, You will need to add EPEL repository also. Run the following command for same.

yum -y install epel-release yum -y update

Now install Certbot using the following command.

yum -y install python-certbot-apache

Once the installation finishes, run the following command to obtain the SSL certificates from Let's Encrypt. Make sure that your domain is pointed to the server, the Let's Encrypt will check the domain authority before providing the certificates.

certbot certonly --apache -d yourdomain.com

This command will run Let's Encrypt client to obtain the certificates only but not to install it. It may ask you which SSL configuration to use during authentication, choose ssl.conf. Finally, provide your email address and accept the terms and condition. Once the certificates are generated, they are likely to be stored in the following directory.

/etc/letsencrypt/live/yourdomain.com

Where yourdomain.com is your actual domain. In the directory, you will find cert.pem which is your domains certificate and privkey.pem which is your certificate's private key.

Let's Encrypt SSL expires in 90 days, so it is recommended to set an automatic renewal for your certificates. Run the following command to open your crontab file.

crontab -e Enter the following line into the crontab file.

30 1 \* \* 1 /usr/bin/certbot renew >> /var/log/le-renew.log

The above cron job will automatically run every Monday at 1:30 AM and if your certificates are due for expiry, it will automatically renew them.

As the SSL certificates are now generated, we can proceed further to configure httpd configuration file. Create a new virtual host file using the following command.

nano /etc/httpd/conf.d/yourdomain.com.conf Replace yourdomain.com with your actual domain. Now copy and paste the following lines into the file.

<VirtualHost \*:80>

ServerName yourdomain.com

Redirect permanent / https://yourdomain.com/

TransferLog /var/log/httpd/yourdomain.com\_access.log

ErrorLog /var/log/httpd/yourdomain.com\_error.log

</VirtualHost>

<VirtualHost \*:443> ServerName yourdomain.com ServerAdmin me@liptanbiswas.com SSLEngine On SSLCertificateFile /etc/letsencrypt/live/yourdomain.com/cert.pem SSLCertificateKeyFile /etc/letsencrypt/live/yourdomain.com/privkey.pem ProxyPreserveHost On ProxyPass / http://localhost:8080/ ProxyPassReverse / http://localhost:8080/ TransferLog /var/log/httpd/yourdomain.com\_ssl\_access.log ErrorLog /var/log/httpd/yourdomain.com\_ssl\_error.log </VirtualHost>

In the above configuration change yourdomain.com to your actual domain. Also, make sure that the path to your SSL certificate and private key are correct. Once done, you will have to restart your Apache server so that the changes made can take effect.

systemctl restart httpd

Now you will need to disable your SELinux because reverse proxy creates errors with SELinux policies. To temporary disable SELinux without restarting the server, run the following command.

setenforce 0

To completely disable the SELinux you will need to edit /etc/selinux/config file.

nano /etc/selinux/config

Find the following line:

SELINUX=enforcing

Change it to:

SELINUX=disabled

Now you will need to reboot your server so that the new configuration can take effect. Now you can start the Liferay service if not started already using the following command.

systemctl start liferay

You can now browse the following URL in your favorite browser.

https://yourdomain.com

You will see the following page to enter basic configuration.

Enter your portal name, choose a language and provide Administrator details. Select the database type as MariaDB and provide the root username and password again. Click **Finish Configuration** button once done. You will be shown the terms and conditions to use the software, click *I Agree* to proceed. Now you will need to provide the password and password hint for the admin account. Finally, you will be taken to your portal dashboard which will look like shown below.

Conclusion

In this tutorial, we have learned how to install Liferay on your CentOS 7 server instance. You can now use the portal to increase the productivity of your business.