



Tutorial

Instalación de



DSpace

DSpace es un programa de software libre que sirve para almacenar colecciones digitales. Generalmente es utilizado como repositorio institucional, y administrador por bibliotecas y archivos para tal fin. Permite el autoarchivo, manejando diferentes tipos de permisos para cada grupo de usuarios, y el depósito de recursos de diferentes tipos de formato desde texto hasta video.

El programa fue creado por el MIT y HP Labs, y liberado en el 2002 bajo licencia BDS (brevemente, se permite la redistribución del código fuente o binario bajo la misma licencia). Su repositorio se encuentra almacenado en GitHub, y actualmente es sostenido por la comunidad usuaria y por Lyrisis, una organización estadounidense sin fines de lucro que da soporte a bibliotecas, archivos, museos e instituciones del patrimonio cultural. La última versión lanzada fue DSpace 8.0.

Documentación oficial

El siguiente tutorial se basó principalmente en las instrucciones de instalación descritas en la wiki oficial de Lyrisis:

<https://wiki.lyrisis.org/display/DSDOC8x/Installing+DSpace>

Algunas consideraciones sobre el hardware

Si bien la documentación mencionada más arriba no especifica requisitos de hardware, ponemos énfasis en tener, dentro de lo posible, más de 4GB de memoria RAM. En los foros hemos encontrado la siguiente lista:

- 2GB de memoria para el Frontend (UI) / Node.js
- 1GB de memoria para el Backend (REST API) / JVM / Tomcat
- 512MB de memoria para la base de datos PostgreSQL
- 512MB de memoria para Solr

pero según nuestra experiencia realizando este tutorial, para poder instalar y correr tanto el backend como el frontend se necesitará más de 4 GB. Para sitios muy activos, se necesitará más memoria RAM sin lugar a dudas.

Sistema operativo

Si bien DSpace puede instalarse en Windows, las instrucciones de Lyrisis recomiendan un sistema operativo basado en Unix, y no se dan mayores especificaciones más allá de corroborar que puedan instalarse los requisitos que detallaremos más adelante. Para la realización del presente tutorial, se eligieron dos sistemas operativos basados en GNU-Linux: Ubuntu 24.04 y Debian 12. En caso de diferencias entre alguna de las instalaciones, las mismas serán marcadas durante el desarrollo del mismo.

Algunas otras consideraciones a tener en cuenta

- Recomendamos tener buena conexión a internet durante la instalación, ya que agilizará el proceso. Algunos pasos como la ejecución de Maven pueden tardar entre 30 y 100 minutos de acuerdo a los recursos de la máquina donde se esté ejecutando.
- Realizaremos la instalación desde la terminal, por que el usuario del sistema operativo desde el cual instalemos debe tener permisos de administración (es decir, estar en la lista de *sudoers*).
- Desde la terminal también editaremos archivos de texto, por lo que se necesita un editor que en nuestro caso será *nano*. Cada vez que se nombre este programa en algún comando, reemplazar por el editor de preferencia.

- Para un ambiente de producción, para tener mayor disponibilidad de recursos, se pueden instalar el backend y el frontend en servidores separados. Sin embargo, como este no es el caso, instalaremos tanto backend como frontend en una misma máquina, por lo que este tutorial pasa por alto las instrucciones específicas para el primer caso y sigue la instalación estándar.

- No es objetivo de este tutorial hacer un recuento exhaustivo de los errores que puedan surgir durante la instalación, pero sí daremos cuenta de aquellos que surgieron en este proceso y de sus soluciones en el paso que corresponda. Para más detalle, sugerimos revisar la sección de Problemas comunes de instalación de la documentación oficial, además de los foros de DSpace y su GitHub.

Backend

Según Lyrasis, es recomendable instalar primero el backend del programa, así que comenzaremos instalando las herramientas necesarias para ello. A no ser que se den indicaciones específicas, se recomiendan las versiones más actualizadas de cada una de estas herramientas.

Herramientas para la instalación del backend

1) Java JDK

DSpace está construido sobre Spring Boot¹, el cual está escrito en Java, por lo cual Java y JDK son necesarios. Se recomienda instalar siempre versiones de soporte a largo plazo (LTS), que para la presente versión de DSpace incluyen JDK 17 y JDK 21. Puede ser tanto OpenJDK como Oracle JDK. Para este tutorial elegimos la versión libre de JDK en su versión 17.

Para instalarla, abrimos la terminal y escribimos:

```
sudo apt-get install openjdk-17-jdk
```

Podemos corroborar la instalación escribiendo en la terminal:

```
java -version
```

lo que debería devolvernos la versión de OpenJDK, junto con la JRE (Open JDK Runtime Environment) y la VM (Virtual Machine).

2) Apache Maven

Maven es utilizado aquí para el ensamblaje del paquete de instalación de DSpace. Para instalarlo, escribimos en la terminal:

```
sudo apt-get install maven
```

Podemos corroborar la instalación escribiendo en la terminal:

```
mvn -version
```

lo que devolverá la versión de Apache Maven, su ubicación en el sistema de archivos, y otras características como versión de Java instalada, sistema operativo, etc.

¹ Spring Boot es un framework para la creación de aplicaciones.



Para DSpace 8, es necesario Maven 3.8.x o superior. Por lo tanto, si el comando introducido anteriormente instala una versión más antigua, es necesario instalar Maven descargando el paquete desde el sitio oficial. Pueden seguirse las instrucciones de aquí:

<https://www.baeldung.com/install-maven-on-windows-linux-mac>

```
sudo mkdir -p /usr/local/apache-maven/apache-maven-3.9.9
sudo tar -xvf Descargas/apache-maven-3.9.9-bin.tar.gz -C
/usr/local/apache-maven/apache-maven-3.9.9
sudo nano ~/.bashrc
export M2_HOME=/usr/local/apache-maven/apache-maven-3.9.9/apache-maven-3.9.9
export M2=$M2_HOME/bin
export MAVEN_OPTS="Xms256m -Xmx512m"
export PATH=$M2:$PATH
source ~/.bashrc
```

3) Apache Ant

Ant se utiliza para realizar la instalación efectiva de DSpace a partir del instalador construido por Maven. Para instalarlo escribimos en la terminal:

```
sudo apt-get install ant
```

Nuevamente, para corroborar la instalación podemos escribir:

```
ant -version
```

4) PostgreSQL

DSpace necesita una base de datos relacional, y PostgreSQL es la recomendada (Lyrasis no brinda información sobre el uso o compatibilidad con otras bases de datos relacionales en su instructivo de instalación). Aquí debemos hacer una serie de pasos:

a) Escribimos lo siguiente para su instalación:

```
sudo apt-get install postgresql
```

Puede verificarse la instalación escribiendo:

```
psql --version
```

b) Instalamos la extensión pgcrypto de PostgreSQL escribiendo:

```
sudo apt-get install postgresql-contrib
```

Esto habilita a DSpace a crear identificadores únicos universales (es decir, no sólo únicos en relación a la propia base de datos)

c) Habilitamos las conexiones TCP/IP:

- Abrimos el archivo postgresql.conf desde la terminal escribiendo (nota: reemplazar en las siguientes direcciones el número "16" por la versión instalada):

```
sudo nano /etc/postgresql/16/main/postgresql.conf
```

y descomentamos la línea que comienza con `listen_addresses = 'localhost'`.

Guardamos y cerramos.

5) Apache Solr

Solr es el motor de búsqueda e indizador utilizado por DSpace, específicamente las versiones 8.x (no se recomiendan versiones anteriores por vulnerabilidades, ni versiones posteriores pues no son todavía completamente compatibles). La documentación recomienda instalarlo en, por ejemplo, /usr/local, así que allí iremos desde la terminal:

```
cd /usr/local
```

Allí descargamos el paquete comprimido, lo renombramos para facilitar su manipulación y lo descomprimimos:

```
sudo wget apache.org/dyn/closet.lua/lucene/8.11.4/solr-8.11.4.tgz
sudo tar fxz solr-8.11.4.tgz
```



En caso de que no funcione el enlace de descarga (en el caso del sistema Debian descargó la página en lugar del archivo), probar de la siguiente forma:

```
sudo wget https://www.apache.org/dyn/closer.lua/lucene/solr/8.11.4/solr-8.11.4.tgz?action=download
sudo mv solr-8.11.4.tgz?action=download solr-8.11.4.tgz
sudo tar fxz solr-8.11.4.tgz
```

Ahora ingresamos al directorio de instalación e instalaremos solr:

```
cd solr-8.11.4/bin
sudo ./install_solr_service.sh ../../solr-8.11.4.tgz
```

Instalación de DSpace

6) Descargar DSpace

Ahora descargamos la última versión del backend de DSpace, que como se mencionó previamente, al momento de este tutorial es la versión 8.0, y descomprimos el archivo. Previamente nos ubicamos en la carpeta `usr>local` (la locación elegida por nosotros) desde la terminal:

```
cd /usr/local
sudo wget
https://github.com/DSpace/DSpace/archive/refs/tags/dspace-8.0.tar.gz
sudo gunzip -c dspace-8.0.tar.gz | tar -xf -
```

7) Configuraciones en Postgresql

Crearemos un user y base de datos para DSpace. Ingresamos a Postgresql en la terminal de la siguiente manera:

```
sudo -i -u postgres
```

y colocamos las siguientes líneas por separado para su creación:

```
createuser --username=postgres --no-superuser --pwprompt dspace
createdb --username=postgres --owner=dspace --encoding=UNICODE
dspace
```



Si durante alguno de los pasos anteriores el resultado fue el error:

```
createdb: error: connection to server on socket
"/var/run/postgresql/.s.PGSQL.5432" failed: FATAL: Peer
authentication failed for user "postgres"
entrar a: cd etc/postgresql/14/main/pg_hba.conf
y corroborar que la primera línea descomentada sea:
```

```
local all postgres peer
```

Por último en este paso, habilitamos para la base de datos recién creada la extensión `pgcrypto` que instalamos en el paso 4:

```
psql --username=postgres dspace -c "CREATE EXTENSION pgcrypto;"
```

8) Configuraciones iniciales en DSpace

El archivo de configuraciones locales de DSpace es el archivo `local.cfg`, que crearemos a partir del archivo de ejemplo que viene en el programa y que se encuentra dentro de la carpeta que hemos descomprimido. En la terminal nos movemos hasta su ubicación y copiamos el archivo:

```
cd /usr/local/DSpace-dspace-8.0/dspace/config
sudo cp local.cfg.EXAMPLE local.cfg
```

Hay algunas configuraciones que la documentación recomienda establecer en este momento de la instalación, a saber:

- `dspace.dir`: la ubicación donde querramos instalar dspace (en nuestro caso, `/usr/local/dspace`)
- `dspace.server.url`: la URL del backend (dejaremos la opción por defecto, `http://localhost:8080/server`)
- `dspace.ui.url`: la URL del frontend (dejaremos la opción por defecto, `http://localhost:4000`)
- `dspace.name`: nombre para el sitio (por ejemplo, DSpace en la UNQ)
- `solr.server`: URL hacia el servidor de Solr, en nuestro caso `http://localhost:8983/solr`
- `db.url`: la JDBC URL completa a la base de datos (dejaremos el valor por defecto utilizado en la configuración de Postgresql, `jdbc:postgresql://localhost:5432/dspace`)
- `db.username`: user de la base de datos que agregamos en el paso anterior (`dspace`)
- `db.password`: contraseña del user de la base de datos que agregamos en el paso anterior (`dspace`)

Para hacerlas, editaremos el archivo `local.cfg` que acabamos de crear:

```
sudo nano local.cfg
```

9) Directorio de DSpace

Crearemos un directorio para el backend de DSpace. Para este paso primero cambiaremos a user root:

```
sudo su
```

y creamos la carpeta en el lugar escogido (en nuestro caso `/usr/local`):

```
cd /usr/local
mkdir dspace
```

Posteriormente le damos permisos sobre esta carpeta al user del sistema operativo desde el cual estamos trabajando con DSpace con el siguiente formato:

```
chown [user] dspace
```

10) Paquete de instalación

Salimos de root con `exit` y cambiamos al user escogido para manejar dspace. Vamos a la carpeta que hemos descomprimido y generamos el paquete de instalación de DSpace:

```
cd DSpace-dspace-8.0/
mvn package
```



Si se presenta el siguiente error:

```
[ERROR] Failed to execute goal
org.apache.maven.plugins:maven-compiler-plugin:3.13.0:compile (default
compile) on project dspace-services: Fatal error compiling: Error while
executing the external compiler javac: Error while executing process. Cannot
run program "javac": error=2, No existe el archivo o directorio
volver a instalar java - JDK debería solucionar el problema.
```



Si se presenta el siguiente error:

```
[ERROR] Failed to execute goal
org.apache.maven.plugins:maven-assembly-plugin:3.7.1:single (default) on
project dspace-parent: Failed to create assembly: Error creating assembly
archive testEnvironment: Problem creating output file for zip
/usr/local/DSpace-dspace-8.0/target,
/usr/local/DSpace-dspace-8.0/target/dspace-parent-8.0-testEnvirnment.zip
recomendamos correr el comando como root, lo que debería solucionar el
problema.
```

11) Instalar el backend

Vamos a la carpeta donde se encuentra el instalador y lo ejecutamos:

```
cd /usr/local/DSpace-dspace-8.0/dspace/target/dspace-installer
ant fresh_install
```

12) Inicializar la base de datos

Para ello, iremos a la siguiente carpeta y correremos el comando:

```
cd /usr/local/dspace
sudo ./dspace database migrate
```

Puede también correrse el siguiente comando para confirmar que la base de datos está correctamente inicializada (el estado de todas las migraciones debe decir "Success" o "Out of order"):

```
sudo ./dspace database info
```

13) Núcleos de Solr

Pasamos los núcleos creados durante la instalación de DSpace a la instancia de Solr instalada y le damos permisos a el user del sistema que posee Solr:

```
sudo cp -R /usr/local/dspace/solr/*
/usr/local/solr-8.11.4/server/solr/configsets
sudo chown -R solr:solr
/usr/local/solr-8.11.4/server/solr/configsets
```

Volvemos a iniciar Solr:

```
./usr/local/solr-8.11.4/bin/solr start
```



Es posible que los permisos dados con el comando "chown" no sean suficientes y haya que ejecutar
`"chmod -R og+rw /usr/local/solr-8.11.4/server/solr/configsets"`
para darle permisos de lectura y escritura a solr

Podemos terminar de corroborar el estado correcto de este paso yendo a <http://localhost:8983/solr/search/select> en el navegador. El resultado debe ser un json vacío.

14) Deployar DSpace

Con el siguiente comando levantaremos Spring Boot y pondremos en funcionamiento el servidor de DSpace:

```
sudo java -jar /usr/local/dspace/webapps/server-boot.jar
```

15) User con permisos de administración

Creamos una cuenta con dichos permisos con:

```
sudo /usr/local/dspace/bin/dspace create-administrator
```

16) Finalmente, en la siguiente dirección encontraremos nuestro servidor:

<http://localhost:8080/server/>

Frontend

Comenzaremos, como lo hicimos con el backend, instalando las herramientas necesarias para el front.

Herramientas para el frontend

1) Node.js

Node es un entorno de ejecución de Java utilizado por DSpace, y son compatibles tanto la versión 18.x como la versión 20.x. Tanto Node, como su manejador de paquetes NPM, son necesarios. Para instalarlos escribimos en la terminal:

```
sudo apt install nodejs  
sudo apt install npm
```

Para corroborar las versiones que hemos instalado podemos escribir:

```
node -version  
npm -version
```

2) Yarn

Yarn es un gestor de dependencias y es usado para construir el frontend de DSpace. Para instalarlo escribimos:

```
sudo npm install --global yarn
```

Para corroborar la versión instalada podemos ejecutar:

```
yarn -version
```


Instalación del frontend

4) Descargar DSpace-Angular

Desde GitHub descargaremos la última versión de DSpace-Angular, en este caso la versión 8.0. En la terminal escribimos lo siguiente para ubicarnos en la misma carpeta donde instalamos el backend, descargar el archivo comprimido, y extraer su contenido:

```
cd /usr/local
sudo wget
https://github.com/DSpace/dspace-angular/archive/refs/tags/dspace-8.0.tar.
gz
sudo gunzip dspace-8.0.tar.gz | tar -xf
```

5) Instalar dependencias

Nos ubicamos en la carpeta recién descomprimida e instalaciones con Yarn las dependencias locales:

```
cd dspace-angular-dspace-8.0
sudo yarn install
```

En este punto, en la pc con menos recursos, se tuvo un error con Yarn "violacion de segmento", es un error de C, que se debe a que no puede acceder a un segmento de memoria. Esto no se pudo resolver ya que la esta pc no cuenta con la suficiente memoria para instalar el Front End.

6) Compilar DSpace-Angular

En este punto compilaremos DSpace-Angular:

```
sudo yarn build:prod
```

7) Configuración

Creamos un archivo Yaml que contenga las configuraciones por default para la interfaz de usuario a partir del archivo de ejemplo que brinda DSpace-Angular:

```
cd dspace-angular-8.0/config
cp config.example.yml config.prod.yml
```

8) Iniciar la interfaz de usuario

Con el backend funcionando, y desde la carpeta dspace-angular-8.0, ejecutar:

```
node ./dist/server/main.js
```



Querer correr `node ./main.js` directamente dentro de la carpeta server resultará en error; debe correrse como se indica más arriba y desde la carpeta raíz dspace-angular-8.0.

Al entrar a la URL que hemos establecido para la interfaz de usuario (en nuestro caso, localhost:4000) desde el navegador, deberíamos ver DSpace funcionando, y poder logearnos con el usuario y contraseña creados en el punto 15 de la instalación del backend.



Si al ingresar a nuestra URL la respuesta es una página de “Error 500 No disponible”, y la ejecución de main.js lanza mensajes como `No _links section found at [rest-api-url]`, entrar al archivo `config.prod.yml`:
`cd dspace-angular-8.0/config`
`sudo nano config.prod.yml`
y corroborar que la URL ingresada allí para la interfaz de usuario (bajo la variable “ui”) sea la correcta. En caso de tener que modificarla, guardar el archivo, detener la ejecución de main.js y volver a levantar la interfaz con `node ./dist/server/main.js`.



Si el error 500 persiste, corroborar en el mismo archivo `config.prod.yml` que los certificados SSL estén desactivados (`ssl: false`) tanto para “ui” (la interfaz) como para “rest” (el backend). En caso de tener que efectuar modificaciones, guardar el archivo, detener la ejecución de main.js y volver a levantar la interfaz con `node ./dist/server/main.js`.



Si el error 500 persiste, corroborar que el backend “confie” en el frontend; entrar a:
`cd /usr/local/Dspace-dspace-8.0/dspace/config`
`sudo nano local.cfg`
y chequear que la siguiente línea esté descomentada:
`rest.cors.allowed-origins = ${dspace.ui.url}`
Luego guardar, cerrar y reiniciar Spring Boot:
`sudo java -jar /usr/local/dspace/webapps/server-boot.jar`



Si el error 500 persiste, corroborar que la URL del backend (variable `dspace.server.url` en el archivo `local.cfg`) sea exactamente igual a la establecida en `config.prod.yml` (bajo la variable “rest”, teniendo en cuenta tanto “host” como “namespace”). De ser diferentes, modificar en alguno de los dos archivos para que ambos archivos contengan la misma URL y reiniciar el back o el front de acuerdo al archivo que se haya modificado.

En caso de querer instalar en computadoras separadas el back y el front, se requiere activar el protocolo https en el backend, en cuyo caso se deberá instalar y configurar apache2 y algunos mods, o también, como otra opción, con Nginx. Para más información consultar la documentación oficial.