



# Transferencia tecnológica de Geonode



# Índice

Requerimientos y requisitos	3
Identificar tipo de solución	3
Configuraciones en preinstalación	3
Configuraciones durante la instalación	4
Configuraciones post instalación	4
Preinstalación	4
Protocolo web y metodo de autenticacion	5
Credenciales y nivel de privacidad de los datos	6
Ubicación de la base de datos	7
Instalación	9
Instalar por primera vez	9
Reinstalar limpiamente	9
Reinstalar conservando los datos anteriores	9
Postinstalación	10
Vector tiles plugin	10
Cors	10
Layers manager	10

# Requerimientos y requisitos

Antes de comenzar con esta guía se requiere realizar el curso 6 “Capacitación Docker y Docker-Compose”. Todos los procesos descritos aquí están pensados para ser realizados en un sistema operativo Ubuntu Server 20.04, debido a las propiedades de docker se pueden extender a cualquier sistema operativo con nulas o mínimas variaciones.

- Se requiere tener preinstalado docker y docker compose
- Se requiere acceso irrestricto a internet en el equipo durante la instalación
- Si el equipo esta detrás de un proxy reverso, este no debe modificar las urls de ninguna petición

## Identificar tipo de solución

La versión de geonode que provee Kan está pensada para ser lo más versátil posible a las finalidades que pueda precisar el cliente, debido a esto existen varios tipos de instalación, todos con sus similitudes y diferencias. Es importante saber identificarlos para poder hacer una correcta instalación. Se deben considerar diferentes elementos de configuración en la instalación que podemos separar en tres etapas:

### Configuraciones en preinstalación

Se configuran previamente a el buildeo de geonode, por lo general, el cambio en alguna de estas implica un rebuild completo de la herramienta. También, por lo general, todas estas están preconfiguradas por parte de Kan, se incluyen en esta guía para brindar la posibilidad de cambio, téngase en cuenta que cualquier mal funcionamiento causado por cambios incorrectamente realizados en estos pasos son responsabilidad de quien los haya efectuado.

- Protocolo web y método de autenticación

- Credenciales y nivel de privacidad de los datos
- Ubicación de la base de datos

## Configuraciones durante la instalación

- Instalar por primera vez
- Reinstalar limpiamente
- Reinstalar conservando los datos anteriores

## Configuraciones post instalación

Son requerimientos extra, que se instalan o activan a posterior del buildeo de la solución, un cambio por lo general no requiere un nuevo buildeo y se pueden hacer sin inconvenientes.

- Vector tiles plugin
- Cors
- Layers manager

# Preinstalación

Ahora vamos a cubrir las instalaciones con todas sus variantes básicas, así como los procedimientos para pasar de un tipo de instalación a otra, se tiene en cuenta que para este punto ya recibió de parte de Kan la solución, ya sea como archivo comprimido, en un repositorio de git, o de alguna otra manera, también se supone que recibió una descripción de cuál es el tipo de instalación y que ya ha decidido cosas tales como el dominio, ubicación de la base de datos, y etc.

Vamos a tener en cuenta que para este momento ya se tiene descargada y/o descomprimida la solución en el directorio donde vamos a trabajar, de aquí en más vamos a llamarlo **geonode/**

## Protocolo web y metodo de autenticacion

Todas las configuraciones que necesariamente modificar a este respecto se encuentran en el archivo **geonode/.env** a continuación se analizaran las líneas pertinentes y el impacto que cada una de estas va a tener en la instalación:

Partamos del supuesto de que queremos instalar nuestra solución en el dominio <https://example.kan.com.ar/> primero debemos rellenar la siguiente línea, note que aquí solo ponemos la url, sin el protocolo

```
23. PUBLIC_PORT=80
24. NGINX_BASE_URL=example.kan.com.ar
```

Luego, en la siguiente entrada si debemos incluir el protocolo

```
48. SITEURL=https://example.kan.com.ar/
```

Ahora seguimos hasta la sección de configuración de nginx, algo a tener en cuenta en este momento es la configuración de nuestro servidor, si tenemos un proxy reverso que implanta los certificados en nuestro dominio, o si estuviéramos haciendo una instalación que no use certificados, es decir, por protocolo http, vamos a poner nuestro dominio en la sección HTTP\_HOST, en este caso, como estamos usando el protocolo https y vamos a usar certificados autofirmados, lo colocamos en la sección HTTPS\_HOST

```
65. GEONODE_LB_HOST_IP=example.kan.com.ar
66. GEONODE_LB_PORT=80
70. HTTP_HOST=
71. HTTPS_HOST=example.kan.com.ar
73. HTTP_PORT=80
74. HTTPS_PORT=443
```

En la siguiente línea podemos configurar nuestros certificados, con “**disabled**” indicamos que no vamos a usar certificados autofirmados, ya sea porque contamos con certificados emitidos por alguna entidad de confianza para el dominio, o porque estamos usando el protocolo http, como este no es el caso, la vamos a configurar en “**production**”

```
81. LETSENCRYPT_MODE=production
```

Por último, en la sección de la configuración de geoserver, tenemos que dejar explícito nuestro dominio en las siguientes dos líneas

```
90. GEOSERVER_WEB_UI_LOCATION=https://example.kan.com.ar/geoserver/  
91. GEOSERVER_PUBLIC_LOCATION=https://example.kan.com.ar/geoserver/  
92. GEOSERVER_LOCATION=http://geoserver:8080/geoserver/
```

## Credenciales y nivel de privacidad de los datos

Todas las configuraciones que necesariamente modificamos a este respecto se encuentran en el archivo `geonode/.env`. A continuación se analizarán las líneas pertinentes y el impacto que cada una de estas va a tener en la instalación.

Comenzamos con las credenciales de Geoserver que son las credenciales que nos serán solicitadas para poder ingresar como administrador a <https://example.kan.com.ar/geoserver/>. Se recomienda encarecidamente no dejar las credenciales por defecto.

```
93. GEOSERVER_ADMIN_USER=admin  
94. GEOSERVER_ADMIN_PASSWORD=geoserver
```

Seguimos con las credenciales del usuario administrador de geonode, al igual que en el caso anterior se recomienda encarecidamente no dejar las credenciales por defecto, tenga en cuenta además que este correo será el que se utilice para generar los certificados autofirmados si configuro esa opción en **“production”** por lo que debe ser un correo valido

```
117. ADMIN_USERNAME=admin  
118. ADMIN_PASSWORD=admin  
119. ADMIN_EMAIL=admin@localhost
```

Las siguientes tres opciones expresan lo siguiente:

- cualquier capa podrá ser vista por defecto por cualquier persona, aun sin loguearse;
- cualquier capa podrá ser descargada por defecto por cualquier persona, aun sin loguearse;

- cualquier persona podrá crearse un usuario para ver capas.

Se recomienda que por defecto deje todas en “**False**”

```
137.  DEFAULT_ANONYMOUS_VIEW_PERMISSION=True
138.  DEFAULT_ANONYMOUS_DOWNLOAD_PERMISSION=True
141.  ACCOUNT_OPEN_SIGNUP=True
```

## Ubicación de la base de datos

Este es un paso opcional, por defecto, la solución incluye una base de datos preconfigurada que vivirá en un contenedor de docker y guardará sus datos en un volumen. Si desea descartar esta base de datos y conectarse a una propia debe seguir los siguientes pasos:

- Abrir el archivo `./wait-for-databases.sh` y comentar todas sus líneas
- Abrir el archivo `./docker-compose.yml` y comentar las siguientes líneas

```
16.      depends_on:
17.        db:
18.          condition: service_healthy
116.     # PostGIS database.
117.     db:
118.       # use geonode official postgis 15 image
119.       image: geonode/postgis:15
120.       command: postgres -c "max_connections=..."
121.       container_name: db4${COMPOSE_PROJECT_NAME}
122.       env_file:
123.         - .env
124.       volumes:
125.         - dbdata:/var/lib/postgresql/data
126.         - dbbackups:/pg_backups
127.       restart: unless-stopped
128.       healthcheck:
129.         test: "pg_isready -d postgres -U postgres"
130.       # uncomment to enable remote connections to postgres
131.       #ports:
132.       #  - "5432:5432"
150.     dbdata:
151.       name: ${COMPOSE_PROJECT_NAME}-dbdata
152.     dbbackups:
153.       name: ${COMPOSE_PROJECT_NAME}-dbbackups
```

Para este momento, se presupone que ya tiene una base de datos o cluster de bases de datos postgres que cumple los siguientes requisitos instalada en otro servidor:

- Postgres version 13+
- Tres usuarios, por lo general: postgres (SU), geonode y geonode\_data
- Dos db con postgis instalado, por lo general: geonode y geonode\_data
- Los dueños de las dbs son los usuarios homónimos con permiso completo
- Algún puerto abierto para posibilitar la conexión: por defecto 5432

Comenzaremos por modificar algunos parametros en el archivo **geonode/.env** para conectarnos a esta base de datos, reemplace las credenciales de los tres usuarios, introduzca la dirección ip del servidor de la base de datos en DATABASE\_HOST y el puerto en DATABASE\_PORT, y en base a esos datos cree las dos url con el formato **postgis://user:pass@ip:port/dbname** y coloquelas en las secciones DATABASE\_URL y GEODATABASE\_URL

```
26. # #####
27. # backend
28. # #####
29. POSTGRES_USER=postgres
30. POSTGRES_PASSWORD=postgres
31. GEONODE_DATABASE=geonode
32. GEONODE_DATABASE_PASSWORD=geonode
33. GEONODE_GEODATABASE=geonode_data
34. GEONODE_GEODATABASE_PASSWORD=geonode_data
35. GEONODE_DATABASE_SCHEMA=public
36. GEONODE_GEODATABASE_SCHEMA=public
37. DATABASE_HOST=db
38. DATABASE_PORT=5432
39. DATABASE_URL=postgis://geonode:geonode@db:5432/geonode
40. GEODATABASE_URL=postgis://geonode_data:geonode_data@db:5432/geonode_data
```



Con esto su geonode ya está configurado para utilizar una base de datos externa en vez de utilizar la base por defecto

# Instalación

## Instalar por primera vez

Estando dentro de la carpeta **geonode/** a la misma altura que el archivo **docker-compose.yml** ejecutamos el comando **docker compose up -d**

Una vez hecho esto el proceso de instalación ya habra comenzado, docker se encargara de descargar dependencias, buildear y levantar la solución en un proceso que tarda aproximadamente unos veinte minutos en realizarse por primera vez, al finalizar podemos consultar el estado de los contenedores con **docker ps** si todos se encuentran healthy (a excepción del contenedor de letsencrypt, que se reinicia periódicamente) la solución ya está levantada y podremos acceder a ella desde la url que hayamos configurado.

## Reinstalar limpiamente

Para una reinstalación limpia procedemos a bajar los contenedores de geonode, limpiar las imágenes, el caché y los volúmenes de docker, y luego realizamos la instalación como si fuese la primera vez. Para esto, estando dentro de la carpeta **geonode/** a la misma altura que el archivo **docker-compose.yml** ejecutamos el comando **docker compose down** luego borramos todos los datos de docker que actualmente no están siendo usados por ningún contenedor con **docker system prune -fa** y vaciamos los volúmenes que no están siendo usados con **docker system prune -f --volumes** . Ahora ya podemos volver a instalar.

## Reinstalar conservando los datos anteriores

Para esto simplemente hay que realizar un backup de las dos bases de datos (por defecto geonode y geonode data), reinstalar limpiamente, detener todos los contenedores (menos el de la base de datos si es que estamos usando la

instalación por defecto) restaurar las bases de datos, y volver a levantar todos los contenedores.

# Postinstalación

Los servicios que se describen aquí son adiciones a los servicios de geonode y geoserver. Suelen ser requeridos por la mayoría de las soluciones que ofrece Kan y será Kan quien le indique si y cuales de los siguientes servicios van a ser necesarios para su caso específico. Tenga en cuenta que la omisión de un servicio necesario o la instalación de un servicio innecesario pueden causar problemas en la solución, por lo que debe instalar solo los que les sean especificados.

## Vector tiles plugin

Este servicio se encuentra en **geonode/vectortiles/** Si se le indicó que debe ser instalado, ingrese a ese directorio y ejecute el comando **bash install.sh** El proceso dura unos diez segundos, y reinicia el contenedor geoserver4geonode que tardará uno o dos minutos en estar activo nuevamente. Utilice el comando **docker ps** para verificarlo y una vez que esté iniciado, ya se puede volver a usar el geonode con normalidad.

## Cors

Este servicio se encuentra en **geonode/cors/** Si se le indico que debe ser instalado, ingrese a ese directorio y ejecute el comando **bash install.sh** El proceso dura unos diez segundos, y reinicia el contenedor geoserver4geonode que tardará uno o dos minutos en estar activo nuevamente. Utilice el comando **docker ps** para verificarlo y una vez que esté iniciado, ya se puede volver a usar el geonode con normalidad.

Si por alguna razon el archivo install.sh no se puede ejecutar, ya sea por la ausencia del mismo, o por algún problema con los permisos de ejecucion del usuario pueden reemplazar su ejecución por la de los siguientes dos comandos

```
docker cp web.xml geoserver4geonode:/usr/local/tomcat/conf/web.xml;  
docker restart geoserver4geonode;
```

## Layers manager

Si se le indico que requiere este servicio, debe saber que el mismo se encuentra preinstalado, y solo debe ser activado. Para hacer esto debe entrar al contenedor de django con el comando **docker exec -it django4geonode bash** y aplicar las migraciones con: **manage.py makemigrations layers\_manager** y **manage.py migrate layers\_manager** luego puede salir con **exit**.