

Índice

Tarea 5.1: Verificación y ajuste de la configuración de Odoo y PostgreSQL usando Docker.....	2
1. Verificación del sistema operativo y gestor de base de datos.....	3
Paso 1: Identificar el sistema operativo dentro del contenedor de Odoo.....	3
Comando para listar los contenedores activos y obtener el nombre/ID del contenedor de Odoo.....	3
Comando para acceder al contenedor de Odoo.....	3
Comando para verificar el sistema operativo dentro del contenedor:.....	3
Preguntas a responder:.....	4
Paso 2: verificar la versión de PostgreSQL.....	4
Comando para acceder al contenedor de PostgreSQL:.....	4
Conéctate a PostgreSQL y especifica la base de datos.....	4
Verificar la versión de PostgreSQL dentro de la base de datos correcta.....	4
Listar las bases de datos disponibles:.....	5
Listar todos los usuarios(roles) creados en PostgreSQL.....	5
Preguntas:.....	5
2. Verificación de configuraciones y cálculo de recursos.....	6
Paso 3: Verificar la cantidad de memoria RAM asignada al contenedor de Odoo.....	6
Preguntas para los alumnos:.....	7
3. Documentación de operaciones e incidencias.....	7
Preguntas para los alumnos:.....	7

Tarea 5.1: Verificación y ajuste de la configuración de Odoo y PostgreSQL usando Docker

1. Verificación del sistema operativo y gestor de base de datos

Paso 1: Identificar el sistema operativo dentro del contenedor de Odoo

Los alumnos deben verificar en qué sistema operativo está corriendo Odoo dentro del contenedor Docker.

- Comando para listar los contenedores activos y obtener el nombre/ID del contenedor de Odoo:

```
docker ps
```

Este comando les mostrará una lista de los contenedores en ejecución. Deben identificar el contenedor de Odoo (puede tener un nombre como "odoo" o un ID de contenedor).

- Comando para acceder al contenedor de Odoo:

```
docker exec -it <nombre_o_id_del_contenedor_de_odoo> /bin/bash
```

Esto abrirá una terminal interactiva dentro del contenedor de Odoo.

- Comando para verificar el sistema operativo dentro del contenedor:

```
uname -a
```

Los alumnos podrán ver detalles del sistema operativo (por ejemplo, si es Ubuntu, Debian, etc.).

Preguntas a responder:

1. ¿Cuál es el sistema operativo que se está usando para ejecutar Odoo dentro del contenedor?
2. ¿Cumple con los requisitos mínimos de Odoo para su instalación?

Paso 2: Verificar la versión de PostgreSQL

Ahora, los alumnos deben verificar que la versión de PostgreSQL instalada cumple con los requisitos para Odoo.

- Comando para acceder al contenedor de PostgreSQL:

```
docker exec -it <nombre_o_id_del_contenedor_de_postgresql> /bin/bash
```

- Conéctate a PostgreSQL y especifica la base de datos

```
psql -U odoo -d <nombre_de_tu_bd>
```

- Verificar la versión de PostgreSQL dentro de la base de datos correcta

```
SELECT version();
```

Esto mostrará la versión de PostgreSQL.

- Listar las bases de datos disponibles: Una vez dentro de `psql`, puedes listar las bases de datos disponibles con el siguiente comando:

```
\l
```

Listar todos los usuarios(roles) creados en PostgreSQL

```
\du
```

```
}
```

Preguntas:

1. ¿Qué versión de PostgreSQL está instalada?
2. ¿Es compatible con Odoo según los requisitos técnicos?

2. Verificación de configuraciones y cálculo de recursos

Paso 3: Verificar la cantidad de memoria RAM asignada al contenedor de Odoo

Los alumnos deben verificar cuánta memoria RAM está utilizando el contenedor de Odoo para garantizar que es suficiente para **60 usuarios concurrentes** que usarán el sistema.

Preguntas para los alumnos:

1. ¿Cuánta memoria RAM está utilizando Odoo en el contenedor?
2. Según el cálculo de trabajadores concurrentes, ¿es suficiente la memoria RAM asignada? Explica el proceso.

3. Documentación de operaciones e incidencias

Paso 4: Documentación de operaciones

Los alumnos deben **documentar todos los comandos que han utilizado** para verificar las configuraciones y ajustar los recursos. Deben incluir capturas de pantalla de los resultados y una breve explicación de lo que han hecho en cada paso.

Paso 5: Documentación de incidencias

Si durante el proceso encontraron problemas, como configuraciones erróneas o falta de recursos, deben **documentar esas incidencias** y cómo las resolvieron.

Preguntas para los alumnos:

1. ¿Qué incidencias encontraste durante la verificación de las configuraciones?
2. ¿Cómo solucionaste los problemas?

1. Verificación del sistema operativo y gestor de base de datos

Paso 1: Identificar el sistema operativo dentro del contenedor de Odoo

Comando para listar los contenedores activos y obtener el nombre/ID del contenedor de Odoo

Docker ps

Este comando lista los contenedores que tenemos en ejecución y los identificamos por el ID.

en este caso el mio es: bea26cb4a254 (lo usaremos en los siguientes comandos)

```
pc16@pc16:~/odoo-docker$ sudo docker ps
[sudo] password for pc16:
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
cb85e1ed9985   odoo:14       "/entrypoint.sh odoo"   2 weeks ago   Up About an hour   0.0.0.0:8069-
>8069/tcp, :::8069->8069/tcp, 8071-8072/tcp   odoo-docker_odoo_1
bea26cb4a254   postgres:13   "docker-entrypoint.s..." 2 weeks ago   Up About an hour   5432/tcp
                                odoo-docker_db_1
pc16@pc16:~/odoo-docker$ _
```

Comando para acceder al contenedor de Odoo

Para poder ejecutar este comando tenemos que ejecutar antes

sudo docker-compose up -d

se utiliza para iniciar los servicios definidos en el archivo docker-compose.yml

Luego ejecutamos:

docker exec -it <nombre_o_id_del_contenedor_de_odoo> /bin/bash

nos permite acceder a la línea de comandos del contenedor de Odoo

```
root@pc16:/home/pc16/odoo-docker# sudo docker-compose up -d
odoo-docker_db_1 is up-to-date
odoo-docker_odoo_1 is up-to-date
root@pc16:/home/pc16/odoo-docker# docker exec -it bea26cb4a254 /bin/bash
root@bea26cb4a254:/#
```

Comando para verificar el sistema operativo dentro del contenedor:

Uname -a

```
root@bea26cb4a254:/# uname -a
Linux bea26cb4a254 5.15.0-122-generic #132-Ubuntu SMP Thu Aug 29 13:45:52 UTC 2024 x86_64 GNU/Linux
```

Con este comando podemos ver los detalles del sistema operativo del servidor (nuestra máquina virtual de linux server)

Preguntas a responder:

1. ¿Cuál es el sistema operativo que se está usando para ejecutar Odoo dentro del contenedor?

Linux, Ubuntu

2. ¿Cumple con los requisitos mínimos de Odoo para su instalación?

Si porque el kernel 5.15-0-122-generic es una versión moderna y la arquitectura x86_64 es adecuada para odoo

Paso 2: verificar la versión de PostgreSQL

Comando para acceder al contenedor de PostgreSQL:

```
Docker exec -it <nombre_o_id_del_contenedor_de_postgresql> /bin/bash
```

(Al hacerlo ahora nos dará error porque ya lo hemos realizado en el paso anterior).

Conéctate a PostgreSQL y especifica la base de datos

```
psql -U odoo -d <nombre_de_tu_bd>
```

comando para conectarnos a la base de datos

```
root@bea26cb4a254:/# psql -U odoo -d db_pcana
psql (13.16 (Debian 13.16-1.pgdg120+1))
Type "help" for help.

db_pcana=# _
```

Verificar la versión de PostgreSQL dentro de la base de datos correcta

```
SELECT version();
```

Este comando solo funciona si el anterior de la conexión a la base de datos ha funcionado y estamos dentro de la línea de comandos podemos comprobarlo viendo **db_pcana=**

Sirve para ver la versión del sistema operativo, la arquitectura y el sistema operativo

```
root@bea26cb4a254:/# psql -U odoo -d db_pcana
psql (13.16 (Debian 13.16-1.pgdg120+1))
Type "help" for help.

db_pcana=# SELECT version();
               version
-----
 PostgreSQL 13.16 (Debian 13.16-1.pgdg120+1) on x86_64-pc-linux-gnu, compiled by gcc (Debian 12.2.0-14) 12.2.0, 64-bit
(1 row)

[1]+  Stopped                  psql -U odoo -d db_pcana
root@bea26cb4a254:/#
```

Listar las bases de datos disponibles:

N

comando para ver las base de datos que tenemos

```
db_pcana=# \l
               List of databases
  Name      | Owner  | Encoding | Collate  | Ctype    | Access privileges
-----+-----+-----+-----+-----+-----
db_pcana    | odoo   | UTF8     | C        | en_US.utf8 |
postgres    | odoo   | UTF8     | en_US.utf8 | en_US.utf8 |
template0   | odoo   | UTF8     | en_US.utf8 | en_US.utf8 | =c/odoo          +
template1   | odoo   | UTF8     | en_US.utf8 | en_US.utf8 | =c/odoo          +
(4 rows)
db_pcana=#
```

Listar todos los usuarios(roles) creados en PostgreSQL

\du

para listar los roles que tiene un usuario, es decir para ver que cosas puede hacer ese usuario en la base de datos donde se esta ejecutando el comando

```
db_pcana=# \du
               List of roles
Role name | Attributes                                     | Member of
-----+-----+-----
odoo      | Superuser, Create role, Create DB, Replication, Bypass RLS | {}
db_pcana=# _
```

Preguntas:

1. ¿Qué versión de PostgreSQL está instalada?

PostgreSQL 13.16 (se puede ver con el comando `SELECT version();`); al acceder a la base de datos que tenemos creada en Odoo)

2. ¿Es compatible con Odoo según los requisitos técnicos?

Si Odoo requiere PostgreSQL 10 o superior y la versión 13.16 cumple con este requisito

2. Verificación de configuraciones y cálculo de recursos

Paso 3: Verificar la cantidad de memoria RAM asignada al contenedor de Odoo

Los alumnos deben verificar cuánta memoria RAM está utilizando el contenedor de Odoo para garantizar que es suficiente para 60 usuarios concurrentes que usarán el sistema.

Regla de oro: (#CPU * 2) + 1

Los trabajadores de Cron necesitan CPU

1 trabajador ~ = 6 usuarios concurrentes

1. Calcular el número de trabajadores necesarios

Usuarios concurrentes: 60

Usuario por trabajador: 6 usuarios por trabajador

$60/6 = 10$ trabajadores

2. Aplicar la regla de oro para la cantidad de CPU

CPU necesaria = (cpu x 2) +1

se puede despejar: CPU = (trabajadores -1) / 2

$CPU = (10-1) / 2 = 4.5$

Necesitaríamos al menos 5CPUs.

3. Calcular la RAM

Memoria RAM recomendada = x trabajadores x 1GB

Memoria RAM recomendada = $10 \times 1GB = 10GB$

Resultado:

Trabajadores necesarios: 10

CPU necesarias: 5

Memoria RAM recomendada: 10GB

Preguntas para los alumnos:

1. ¿Cuánta memoria RAM está utilizando Odoo en el contenedor?

docker stats <nombre_del_contenedor>

```
CONTAINER ID   NAME          CPU %     MEM USAGE / LIMIT   MEM %     NET I/O       BLOCK
I/O          PIDS
bea26cb4a254   odoo-docker_db_1  0.10%    35.21MiB / 1.918GiB  1.79%    4.56kB / 3.09kB  69.7MB
/ 590kB      7
```

Esta usando 36.62MiB

2. Según el cálculo de trabajadores concurrentes, ¿es suficiente la memoria RAM asignada? Explica el proceso.

Ver la memoria RAM del servidor se usa el comando **free -m**

No sería suficiente porque no tenemos en la máquina virtual 10GB

```
root@pc16:/home/pc16/odoo-docker# free -m
              total        used        free      shared    buff/cache   available
Mem:            1963         290        1164         12         508        1503
Swap:           2047           0         2047
```

3. Documentación de operaciones e incidencias

Preguntas para los alumnos:

1. ¿Qué incidencias encontraste durante la verificación de las configuraciones?

-**sudo docker-compose up -d**

-**Docker exec -it <nombre_o_id_del_contenedor_de_postgresql> /bin/bash**

-poner el puerto :3306

2. ¿Cómo solucionaste los problemas?

1. Que sin iniciar los servicios con **sudo docker-compose up -d** no podemos conectarnos a PostgreSQL ni ver las bases de datos ni hacer consultas, no podremos hacer nada.

2. Si hacemos toda la tarea seguida una vez ejecutado este comando ejecutarlo de nuevo para la parte dos nos dará error **Docker exec -it <nombre_o_id_del_contenedor_de_postgresql> /bin/bash**

3. poner el puerto al conectarnos desde otra máquina. ip y :3306