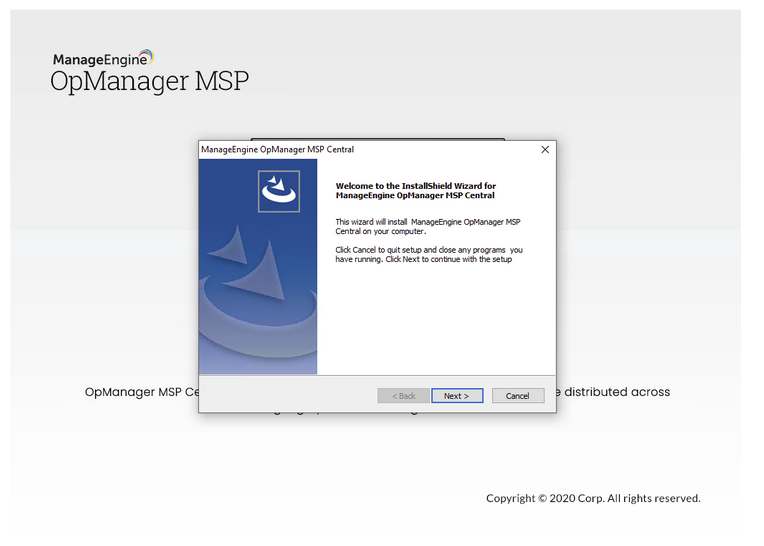
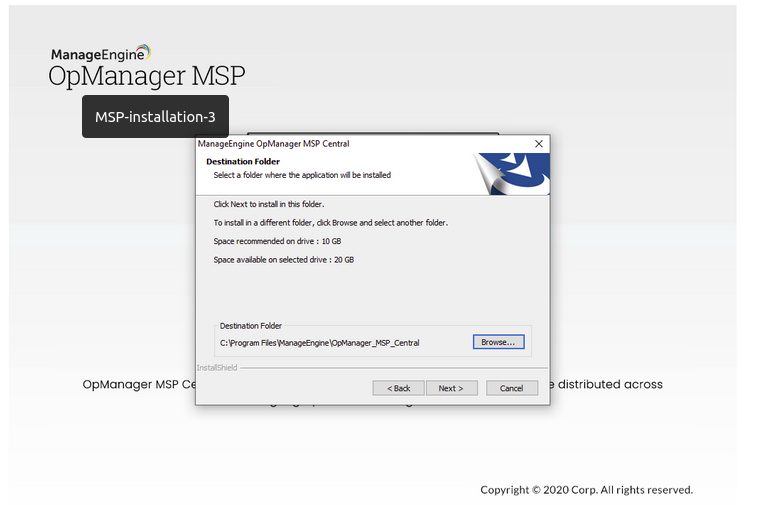
**Instalacion OpManager**

Primero instalamos una maquina de windows 10, en la que tendremos instalador xampp

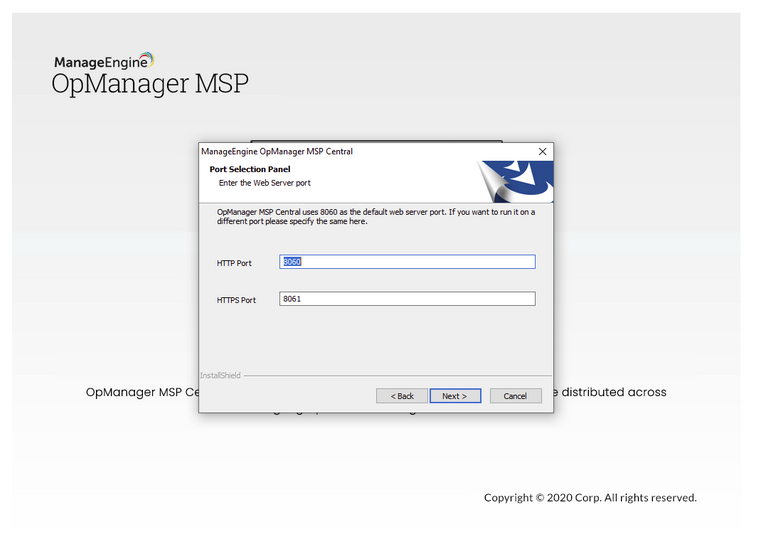
Despues de tener esto descargaremos el programa aqui <https://www.manageengine.com/es/network-monitoring/download.html>

Ejecutaremos el .exe de windows

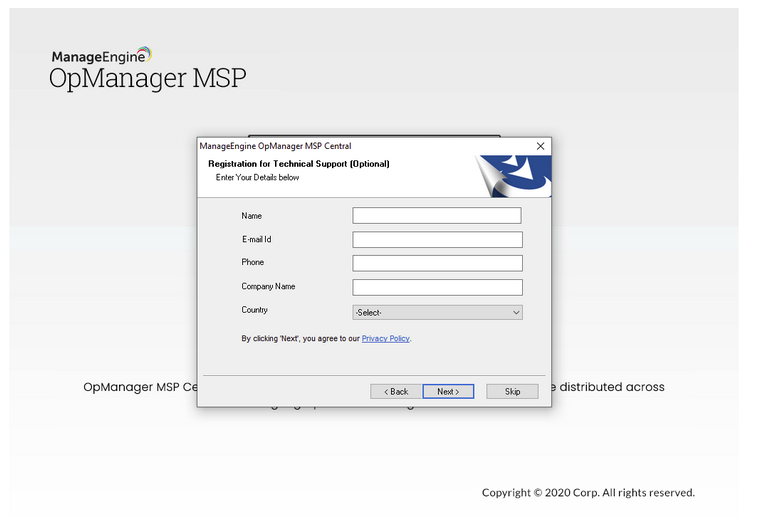


2- Aqui elegiremos la ruta donde queremos instalar el programa

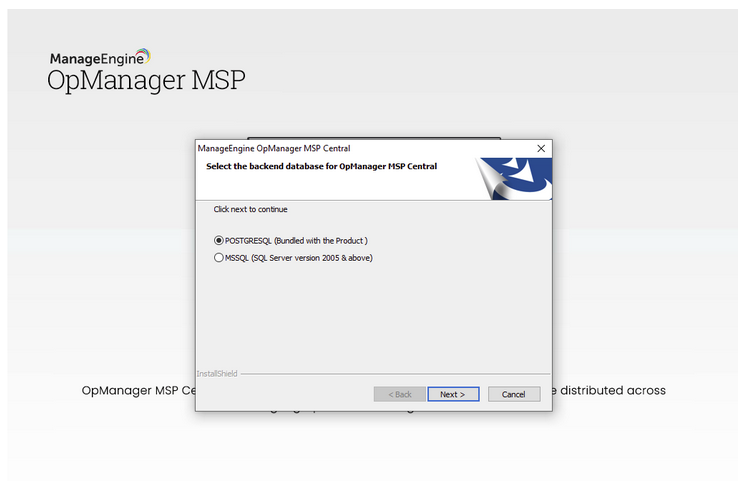
3- En este paso le diremos en el puerto en el que lo queremos lanzar



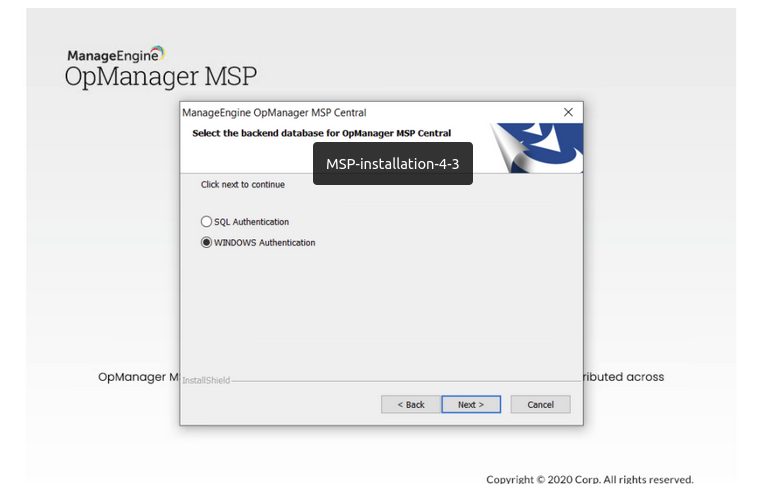
4- Seguiremos con la instalacion escribiendo el nombre el email y los datos que nos piden



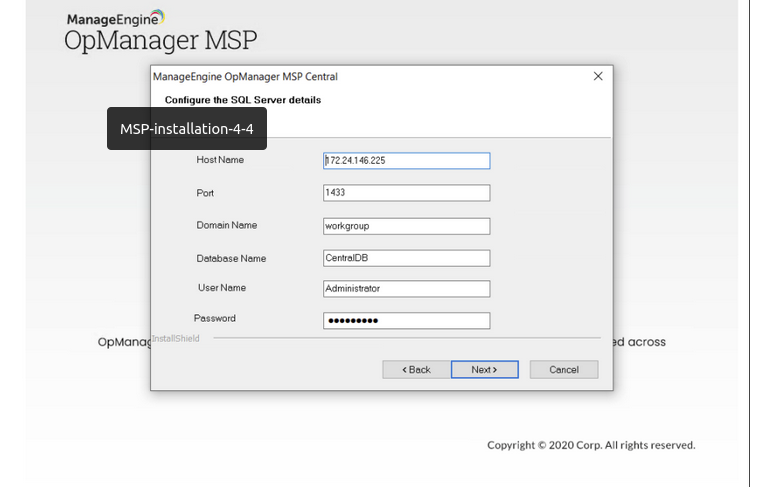
5- Aqui elegiremos la opcion de mssql ya que con postgresql a nosotros no nos ha funcionado.



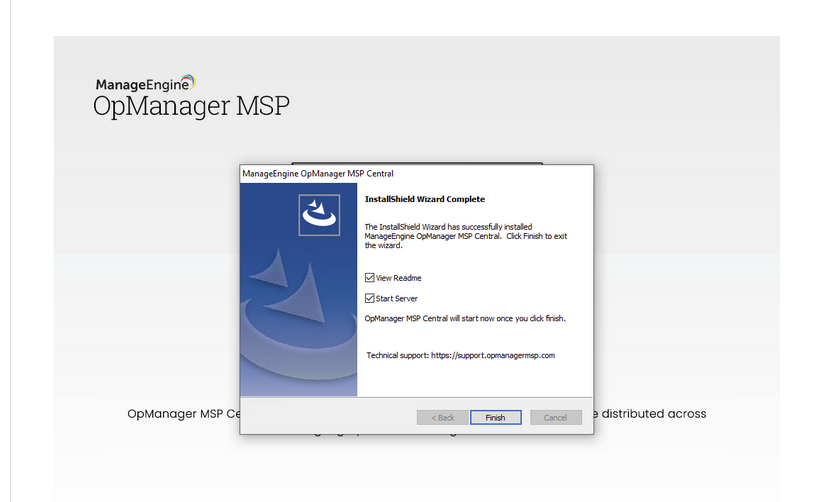
6- Elegiremos la opcion de windows authentication ya que lo estamos instalando en este sistema operativo



7- Rellenaremos los datos estos y en hostname escribiremos localhost, el puerto dejamos el que nos diga y los otros datos como queramos nombrar a la base de datos, junto al usuario y la contaseña



8- Y por ultimo ejecutaremos el programa y se nos abriria en el navegador, para entrar despues de la instalacion escribiriamos localhost(o el nombre de nuestro equipo):8060 que es el puerto que hemos habilitado.



**Configuración de Switch**

Para poder acceder al switch tenemos que reiniciar el switch introducir un clip o algo puntiagudo en el agujero donde pone reset y estar 10 segundos.

Una vez realizado el reset tenemos que configurar la red poniendo en la ip 10.90.90.x

mascara 255.255.255.0 y la puerta de enlace 10.90.90.90.

Una vez hecho eso introducimos en el buscador 10.90.90.90 y ya entraremos en el switch, nos pedira una contraseña en la que tendremos que poner admin porque es la contraseña base del switch.

Ya dentro de la configuración del switch para hacer las Vlan tenemos que ir al apartador Vlan y dentro a 802.1Q Vlan. Donde asignaremos los grupos de vlan, el nombre y el rango de puerto de cada vlan.

Luego para que no se puedan ver las vlans entre si tendremos que restringir el trafico.

Donde tendremos que ir al menu principal del switch>security>traffic segmentation.

Dentro de traffic segmentation elegimos el puerto que queremos que pase el trafico por los puertos selecionados.

**Creacion de servicio que lanza webservice**

Para crear este servicio que queremos que se pueda tambien detener ademas, de iniciar cuando el servidor se inicie, necesitaremos dos scripts .sh (ya que el webservice esta definido en una maquina ubuntu)

En el primer script que sera el de inicio escribiremos lo siguiente:

#!/bin/bash

cd /home/nombreUsuario/NombreCarpetaProyecto

git pull origin

cd /home/nombreUsuario/NombreCarpetaProyecto/WS

php bin/console server:run 192.168.4.x:8000

En estas lineas basicamente lo que estamos haciendo es acceder a la ruta donde esta el proyecto actualizarlo en el branch principal (original) y ejecutar el webservice en modo servidor en una ip especifica y en un puerto especifico como en este caso.

En el segundo script que servira para parar el servicio escribiremos lo siguiente:

#!/bin/bash

cd /home/nombreUsuario/NombreCarpetaProyecto/WS

php bin/console server:stop

Este script es mas corto y sencillo y despues de haber visto el de antes, es facil deducir que lo que hacemos es acceder a la carpeta y parar el webservice (o servidor) que esta lanzado en la ip especifica y el puerto especifico de antes.

Lo siguiente sera ubicar estos dos archivos en la siguiente ruta: /usr/local/bin. Les daremos permiso de ejecucion, es decir hacer un chmod 755 nombreDelArchivo.sh.

Por ultimo en /etc/systemd/system crearemos un archivo .service (el archivo que se ejecutara nada mas iniciar el ordenador o mas conocido como servicio como su propio nombre indica).

En este archivo lo que escribiremos sera lo siguiente:

[Unit]

Description = descripcion que quieras poner

After=networking.target

[Service]

Type=oneshot

ExecStart=/usr/local/bin/nombreArchivo.sh

ExecStop=/usr/local/bin/nombreArchivo.sh

RemainAfterExit=yes

[Install]

WantedBy=multi-user.target

De esta forma definimos que script se tiene que ejecutar dependiendo si el ordenador se esta iniciando o estemos diciendo que queremos parar el servicio etc.

Antes de darlo por finalizado haremos ciertas comprobaciones de si el servicio esta funcionando bien, las comprobaciones son las siguientes:

systemctl start nombreServicio.service

systemctl status --no-pager nombreServicio.service

systemctl stop nombreServicio.service

Si todo va bien y en las comprobaciones dice que el servicio esta activado, lo ultimo que habra que hacer es habilitar el servicio para que cuando se reinicie el servicio se inicie automaticamente, y el comando es el siguiente:

systemctl enable nombreServicio.service

**pgadmin documentacion**

Lo primero que debemos de hacer es instalar apache 2 con su correspondiente código ‘sudo apt install apache2’.

Nada más acabar la instalación de apache instalamos postgresql con el código ‘sudo apt install postgresql-10’.

Seguido de esto hacemos su postgres(para cambiar usuario) y después hacemos psql postgres.

Después de cambiar de usuario cambiamos la contraseña usando el comando

‘ALTER USER postgres PASSWORD '(la nueva contraseña)';.

Al cambiar el usuario y la cpntraseña debemos instalar pgadmin, y para instalarlo debemos introducir ‘apt install pgadmin3’.

**Hardware**

TORRE CERRADA:

gigabyte H55M-S2H LGA1150 + intel HD graphics

intel core i3 540

ram ddr3 8gb (2x4gb) 2133mhz

hdd: SEAGATE BARRACURA 7200RPM ST3500418AS

TORRE ABIERTA:

GIGABYTE GA-H81M-S2H LGA1150 + intel HD graphics

ram ddr3 8gb (2x4gb) 2133mhz

hdd SEAGATE BARRACUDA 7200RPM ST3170827AS

Intel(R) Core(TM) i5-4460 CPU@ 3.20GHz

SWITCH X2:

D-LINK DGS-1210-24