

Configuración nuevo server AWS

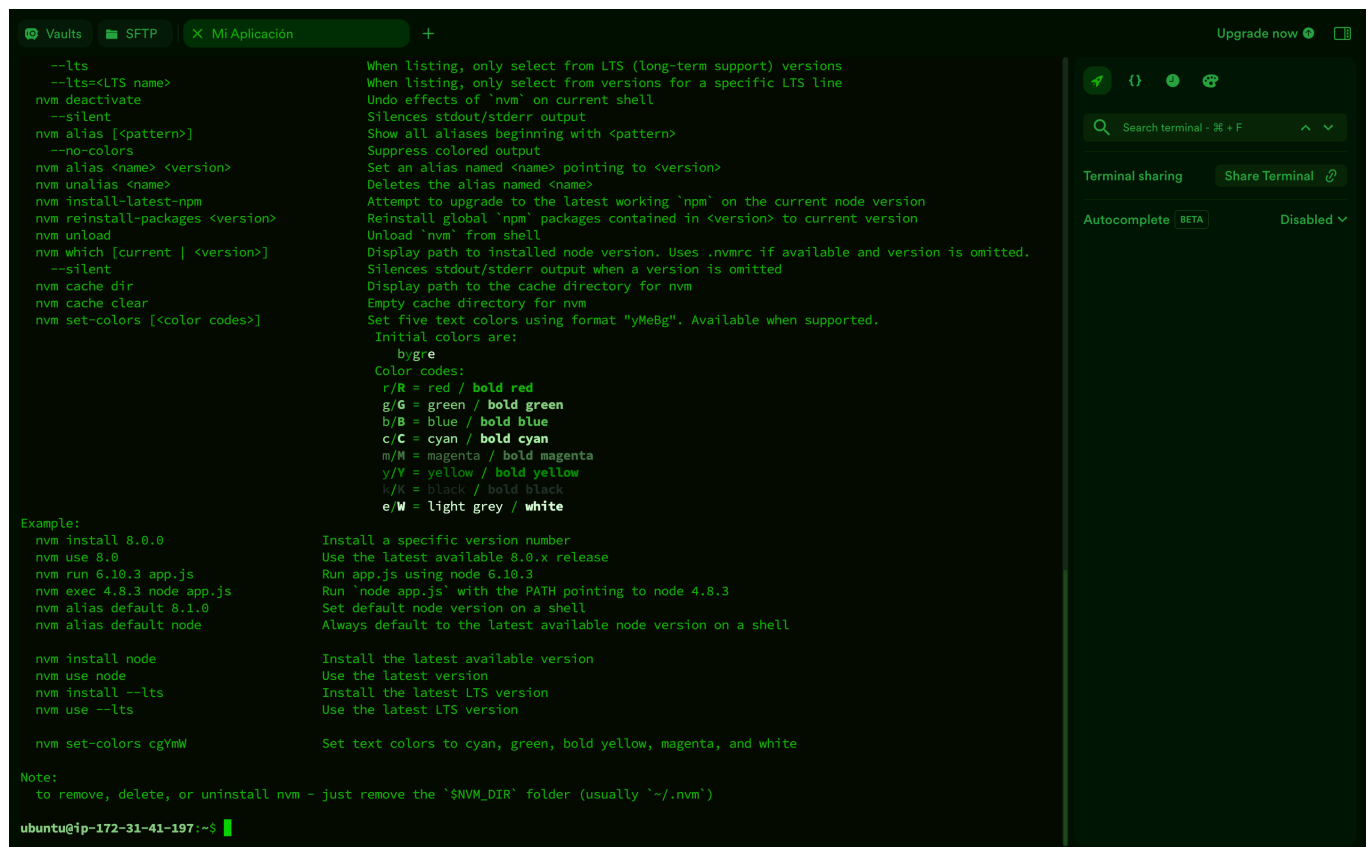
Documentación de los pasos esenciales para levantar un servidor desde cero en AWS, incluyendo instalación de herramientas, configuración del entorno de desarrollo y despliegue con NGINX.

(Ya loggeado) hace falta instalar NVM. Para instalar NVM son estos dos comandos.

```
1 curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.40.3/install.sh | bash
```

```
1 export NVM_DIR="$([ -z "${XDG_CONFIG_HOME-}" ] && printf %s "${HOME}/.nvm" ||  
printf %s "${XDG_CONFIG_HOME}/nvm")"  
2 [ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh" # This loads nvm
```

Si le damos `nvm` y nos sale esto es que ya está:



```
--lts
--lts<LTS name>
nvm deactivate
--silent
nvm alias [<pattern>]
--no-colors
nvm alias <name> <version>
nvm unalias <name>
nvm install-latest-npm
nvm reinstall-packages <version>
nvm unload
nvm which [<current | <version>]
--silent
nvm cache dir
nvm cache clear
nvm set-colors [<color codes>]

When listing, only select from LTS (long-term support) versions
When listing, only select from versions for a specific LTS line
Undo effects of 'nvm' on current shell
Silences stdout/stderr output
Show all aliases beginning with <pattern>
Suppress colored output
Set an alias named <name> pointing to <version>
Deletes the alias named <name>
Attempt to upgrade to the latest working 'npm' on the current node version
Reinstall global 'npm' packages contained in <version> to current version
Unload 'nvm' from shell
Display path to installed node version. Uses .nvmrc if available and version is omitted.
Silences stdout/stderr output when a version is omitted
Display path to the cache directory for nvm
Empty cache directory for nvm
Set five text colors using format "yMeBg". Available when supported.
Initial colors are:
  byg/e
Color codes:
  r/R = red / bold red
  g/G = green / bold green
  b/B = blue / bold blue
  c/C = cyan / bold cyan
  m/M = magenta / bold magenta
  y/Y = yellow / bold yellow
  k/K = black / bold black
  e/W = light grey / white

Example:
nvm install 8.0.0          Install a specific version number
nvm use 8.0                Use the latest available 8.0.x release
nvm run 6.10.3 app.js       Run app.js using node 6.10.3
nvm exec 4.8.3 node app.js  Run 'node app.js' with the PATH pointing to node 4.8.3
nvm alias default 8.1.0    Set default node version on a shell
nvm alias default node     Always default to the latest available node version on a shell

nvm install node           Install the latest available version
nvm use node               Use the latest version
nvm install --lts          Install the latest LTS version
nvm use --lts              Use the latest LTS version

nvm set-colors cgYmW      Set text colors to cyan, green, bold yellow, magenta, and white

Note:
to remove, delete, or uninstall nvm - just remove the '$NVM_DIR' folder (usually '~/.nvm')

ubuntu@ip-172-31-41-197:~$
```

Luego, ocupamos instalar node.js en el servidor:

```
1 npm install v22.14.0
```

```

ubuntu@ip-172-31-41-197:~$ nvm install v22.14.0
Downloading and installing node v22.14.0...
Downloading https://nodejs.org/dist/v22.14.0/node-v22.14.0-linux-x64.tar.xz...
##### 100.0%
Computing checksum with sha256sum
Checksums matched!
Now using node v22.14.0 (npm v10.9.2)
Creating default alias: default -> v22.14.0
ubuntu@ip-172-31-41-197:~$ node -v
v22.14.0
ubuntu@ip-172-31-41-197:~$ █

```

Cuando sea un servidor (ya sea AWS u otros) es mejor utilizar esa línea de comandos (Termius) y usar esos comandos.

Como ya tenemos node, ya tenemos npm también. El siguiente paso es clonar el repositorio (en el ejemplo, es el repositorio que nos pasó el profe).

```

ubuntu@ip-172-31-41-197:~$ npm -v
10.9.2
ubuntu@ip-172-31-41-197:~$ ls
ubuntu@ip-172-31-41-197:~$ cd
ubuntu@ip-172-31-41-197:~$ git clone https://github.com/black4ninja/deploy-react-node-2025.git
Cloning into 'deploy-react-node-2025'...
remote: Enumerating objects: 28, done.
remote: Counting objects: 100% (28/28), done.
remote: Compressing objects: 100% (24/24), done.
remote: Total 28 (delta 1), reused 25 (delta 1), pack-reused 0 (from 0)
Receiving objects: 100% (28/28), 39.53 KiB | 3.04 MiB/s, done.
Resolving deltas: 100% (1/1), done.
ubuntu@ip-172-31-41-197:~$ █

```

```

ubuntu@ip-172-31-41-197:~$ ls
deploy-react-node-2025
ubuntu@ip-172-31-41-197:~$ cd deploy-react-node-2025/
ubuntu@ip-172-31-41-197:~/deploy-react-node-2025$ cd mi-aplicacion-react/
ubuntu@ip-172-31-41-197:~/deploy-react-node-2025/mi-aplicacion-react$ npm install

added 150 packages, and audited 151 packages in 5s

30 packages are looking for funding
  run `npm fund` for details

1 moderate severity vulnerability

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
npm notice
npm notice New major version of npm available! 10.9.2 -> 11.4.1
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.4.1
npm notice To update run: npm install -g npm@11.4.1
npm notice
ubuntu@ip-172-31-41-197:~/deploy-react-node-2025/mi-aplicacion-react$ cd s
server/ src/
ubuntu@ip-172-31-41-197:~/deploy-react-node-2025/mi-aplicacion-react$ cd s
server/ src/
ubuntu@ip-172-31-41-197:~/deploy-react-node-2025/mi-aplicacion-react$ cd server
ubuntu@ip-172-31-41-197:~/deploy-react-node-2025/mi-aplicacion-react/server$ npm install

added 66 packages, and audited 67 packages in 2s

14 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
ubuntu@ip-172-31-41-197:~/deploy-react-node-2025/mi-aplicacion-react/server$ █

```

Para iniciar el server con `pm2` es necesario instalarlo:

```
1 npm install pm2 -g
```

[illegible]

Falta editar una regla para poder entrar. Clickea en la instancia en el panel de AWS, en seguridad, dale en en grupos de seguridad, dale en editar reglas de entrada, y crea una nueva regla de entrada.

Buscar Instancia por atributo o etiqueta (case-sensitive)

Todos los ...

< 1 >

✓

Name

▼

ID de la instancia

▼

Estado de la i...

▼

Tipo de inst...

Comprobación de

Estado de la al:

▼

Zona de dispon...

DNS

✓

MiAplicación

i-0a70c830f5b060107

✓ En ejecución

t2.micro

✓ 2/2 comprobaci...

Ver alarmas +

us-east-2c

ec2-

i-0a70c830f5b060107 (MiAplicación)

⚙️

▼

Detalles

Estado y alarmas

Monitoreo

Seguridad

Redes

Almacenamiento

Etiquetas

▼ Detalles de seguridad

Rol de IAM

-

ID del propietario

268462546339

Hora de lanzamiento

Thu May 22 2025 11:26:33 GMT-0600 (hora estándar central)

Grupos de seguridad

sg-06bbcf8b6d2048a84 (launch-wizard-1)

▼ Reglas de entrada

Q Filtrar reglas

< 1 >

Nombre	ID de la regla del grupo d...	Intervalo de pu...	Protocolo	Origen	Grupos
-	sgr-03a3ce601d04ce32a	22	TCP	0.0.0.0/0	launch-

▼ Reglas de salida

Q Filtrar reglas

< 1 >

© 2025, Amazon Web Services, Inc. o sus filiales.

Privacidad

Términos

Preferencias de cookies

aws

Buscar

[Opción+S]

📧

🔔

🕒

⚙️

Estados Unidos (Ohio)

tuzobus

☰

EC2

>

Grupos de seguridad

>

sg-06bbcf8b6d2048a84 - launch-wizard-1

>

Editar reglas de entrada

📄

🔄

🖨️

Editar reglas de entrada

Las reglas de entrada controlan el tráfico entrante que puede llegar a la instancia.

Reglas de entrada

ID de la regla del grupo de seguridad

Tipo

Protocolo

Intervalo de puertos

Origen

Descripción: opcional

sgr-03a3ce601d04ce32a

SSH

TCP

22

Pers...

Q

Eliminar

-

TCP personalizado

TCP

3000

Any...

Q

0.0.0.0/0

Eliminar

Agregar regla

⚠️ Las reglas cuyo origen es 0.0.0.0/0 o :::/0 permiten a todas las direcciones IP acceder a la instancia. Recomendamos configurar reglas de grupo de seguridad para permitir el acceso únicamente desde direcciones IP conocidas.

Cancelar

Previsualizar los cambios

Guardar reglas

CloudShell Comentarios

© 2025, Amazon Web Services, Inc. o sus filiales. Privacidad Términos Preferencias de cookies

Ocupamos NGINX.

```
1 sudo apt update
2 sudo apt install nginx
```

Para saber si NGINX está corriendo, checamos si active: active (running) con el siguiente comando

```
1 ubuntu@ip-172-31-41-197:~/deploy-react-node-2025/mi-aplicacion-react/server$
  sudo service nginx status
2 ● nginx.service - A high performance web server and a reverse proxy server
3     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset:
  enabled)
4     Active: active (running) since Thu 2025-05-22 18:16:15 UTC; 31s ago
5     Docs: man:nginx(8)
6     Process: 5410 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on;
  master_process on; (code=exited, status=0/SUCCESS)
7     Process: 5411 ExecStart=/usr/sbin/nginx -g daemon on; master_process on;
  (code=exited, status=0/SUCCESS)
8     Main PID: 5504 (nginx)
9     Tasks: 2 (limit: 1129)
10    Memory: 4.6M
11    CPU: 24ms
12    CGroup: /system.slice/nginx.service
13            └─5504 "nginx: master process /usr/sbin/nginx -g daemon on;
  master_process on;"
14            └─5507 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" ""
  "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""
15
16 May 22 18:16:15 ip-172-31-41-197 systemd[1]: Starting A high performance web
  server and a reverse proxy server...
17 May 22 18:16:15 ip-172-31-41-197 systemd[1]: Started A high performance web
  server and a reverse proxy server.
```

Añadimos estas dos reglas a seguridad. (PUERTOS DEFAULT)

-	HTTP	TCP	80	Any...	0.0.0.0		Eliminar
-	HTTPS	TCP	443	Any...	0.0.0.0		Eliminar
Agregar regla							

Y PARA EL DE LA BASE DE DATOS:

-	TCP personalizado	TCP	3306	Any...	0.0.0.0		Eliminar
---	-------------------	-----	------	--------	---------	--	----------

Configuración NGINX

Config default de NGINX:

```
1  cd /etc/nginx/sites-available/
```

```
1  sudo vim salesnest.com.mx
2
3  (PARA SALIR)
4  :q!
```

```
1  server {
2      listen 80;
3      server_name 3.131.217.131; # Cambia esto por tu dominio o IP pública
4
5      location / {
6          proxy_pass http://localhost:3000;
7          proxy_http_version 1.1;
8          proxy_set_header Upgrade $http_upgrade;
9          proxy_set_header Connection 'upgrade';
10         proxy_set_header Host $host;
11         proxy_cache_bypass $http_upgrade;
12     }
13 }
```

Luego,

```
1  :wq!
```

para guardar y salir de VIM.

```
ubuntu@ip-172-31-41-197:/etc/nginx/sites-available$ sudo vim salesnest.com.mx
ubuntu@ip-172-31-41-197:/etc/nginx/sites-available$ ls
default  salesnest.com.mx
ubuntu@ip-172-31-41-197:/etc/nginx/sites-available$
```

Después elimina lo de sites-enabled:

```
ubuntu@ip-172-31-41-197:/etc/nginx/sites-available$ cd ..
ubuntu@ip-172-31-41-197:/etc/nginx$ ls
conf.d      fastcgi_params  koi-win        modules-available  nginx.conf  scgi_params    sites-enabled  uwsgi_params
fastcgi.conf  koi-utf        mime.types     modules-enabled    proxy_params  sites-available  snippets      win-utf
ubuntu@ip-172-31-41-197:/etc/nginx$ cd sites-enabled/
ubuntu@ip-172-31-41-197:/etc/nginx/sites-enabled$ ls
default
ubuntu@ip-172-31-41-197:/etc/nginx/sites-enabled$ sudo rm default
ubuntu@ip-172-31-41-197:/etc/nginx/sites-enabled$ ls
ubuntu@ip-172-31-41-197:/etc/nginx/sites-enabled$
```

Y ejecuta esto:

```
1  sudo ln -s /etc/nginx/sites-available/salesnest.com.mx /etc/nginx/sites-enabled/
```

Si le das ls, debería salirte (ahí mismo en sites-enabled) un archivo llamado salesnest.com.mx

Para probar la configuración de NGINX. Si me sale ok es que puedo reiniciar y todo estará bien.

```
1  ubuntu@ip-172-31-41-197:/etc/nginx/sites-enabled$ sudo nginx -t
2  nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
3  nginx: configuration file /etc/nginx/nginx.conf test is successful
```

Después de hacer esto, ya debería jalar la config que le pusimos:

```
1  sudo service nginx restart
```

```
1  sudo nginx -t
2  sudo service nginx restart
```