

Instalación de la iso Alpine Linux con mpi4py

Librerías y dependencias instaladas:

- python3
- pip
- make cmake gcc g++ gfortran
- numpy
- openmpi
- mpi4py
- openjdk8

Crear y acceder a la máquina virtual con el usuario *root*:



Ejecutar:

```
sd-GCDA:~# alpine-install
```

Elige `sys` cuando se pregunte sobre el modo de disco (esto instalará permanentemente Alpine en el disco).

Elige el disco predeterminado montado por VMware y reinicia una vez que se complete la instalación.

Después de la instalación, actualizamos los repositorios para hacer la instalación de las librerías:

```
sd-GCDA:~# apk update
```

Instalar Python 3.9 y PIP.

```
sd-GCDA:~# apk add python3
```

```
sd-GCDA:~# python3 -m ensurepip
```

```
sd-GCDA:~# /usr/bin/python3 -m pip install --upgrade pip
```

Instalación de la librería *numpy*. Primero resolvemos sus dependencias:

```
sd-GCDA:~# pip install cython
```

```
sd-GCDA:~# apk add --update make cmake gcc g++ gfortran
```

```
sd-GCDA:~# apk --no-cache add musl-dev linux-headers g++
```

Instalación de *numpy*:

```
sd-GCDA:~# pip3.9 install numpy
```

Instalación de MPI. Es necesario habilitar repositorios:

```
sd-GCDA:~# vi /etc/apk/repositories
```

```
sd-GCDA:~# enable [http://dl-cdn.alpinelinux.org/alpine/v3.14/community]
```

Instalación de OpenMPI:

```
sd-GCDA:~# apk add --upgrade openmpi
```

Si queremos desarrollar aplicaciones MPI, hay que instalar los *wrappers* de compilación (mpicc, mpic++, mpif90, mpif77):

```
sd-GCDA:~# apk add openmpi-dev
```

Instalación de mpi4py, para dar soporte MPI a Python:

```
sd-GCDA:~# env MPICC=mpicc python3.9 -m pip install mpi4py
```

Instalación de Java:

```
sd-GCDA:~# apk add openjdk8
```

Con estas librerías tenemos el entorno de ejecución preparado.

Extras y resolución de problemas:

En caso de acceder vía *ssh* y ejecutar como *root* es necesario hacer las siguientes modificaciones:

Como root, edita el fichero *sshd_config* en */etc/ssh/sshd_config*:

```
sd-GCDA:~# nano /etc/ssh/sshd_config
```

Agrega una línea en la sección Autenticación del archivo que diga *PermitRootLogin yes*. Es posible que esta línea ya exista y esté comentada con un "#". En este caso, elimina el "#".

PermitRootLogin yes

Restaura el servicio *ssh* para actualizar los cambios:

```
sd-GCDA:~# service sshd restart
```

Para OpenMPI, no es aconsejable ejecutar aplicaciones MPI desde *root*.

```
localhost:~# mpirun
-----
mpirun has detected an attempt to run as root.

Running as root is *strongly* discouraged as any mistake (e.g., in
defining TMPDIR) or bug can result in catastrophic damage to the OS
file system, leaving your system in an unusable state.

We strongly suggest that you run mpirun as a non-root user.

You can override this protection by adding the --allow-run-as-root option
to the cmd line or by setting two environment variables in the following way:
the variable OMPI_ALLOW_RUN_AS_ROOT=1 to indicate the desire to override this
protection, and OMPI_ALLOW_RUN_AS_ROOT_CONFIRM=1 to confirm the choice and
add one more layer of certainty that you want to do so.
We reiterate our advice against doing so - please proceed at your own risk.
-----
localhost:~#
```

Para habilitar la ejecución de MPI como *root*, se exportan las siguientes variables de entorno:

```
sd-GCDA:~# nano /etc/profile
```

Y se añaden las siguientes variables:

```
export OMPI_ALLOW_RUN_AS_ROOT=1
export OMPI_ALLOW_RUN_AS_ROOT_CONFIRM=1
```