

SUPERCOMPUTADORA INKARI

UNIVERSIDAD NACIONAL DE SAN AGUSTÍN

---

## Manual de uso de Inkari

---

*Autor:*  
Ronald Apaza

*Grupo:*  
IAAPP

13 de enero de 2023

## 1 Introducción

En el ámbito de la computación nos referimos como clúster a un conjunto de ordenadores conectados entre sí y que mediante un software específico permite realizar operaciones de forma simultánea, proporcionado así, una mayor capacidad de cómputo.

## 2 Conexión al clúster

Para conectarse al clúster es necesario disponer previamente de una cuenta. Para solicitar una cuenta, se debe de escribir un correo electrónico a: `inkari.unsa@gmail.com`.

### 2.1 Acceso desde Linux

La mayoría de los ordenadores con Linux vienen dotadas de un cliente ssh que puede ser invocado directamente desde el terminal mediante el siguiente comando:

```
ssh <cuenta>@inkari.unsa.edu.pe
```

### 2.2 Acceso desde Windows

Para acceder al clúster en windows será necesario utilizar un cliente ssh. Putty es uno de los clientes más populares y ligeros, este puede ser descargado de: <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>. Una vez instalado, Putty nos ofrece la interfaz, mostrada en la figura 1:

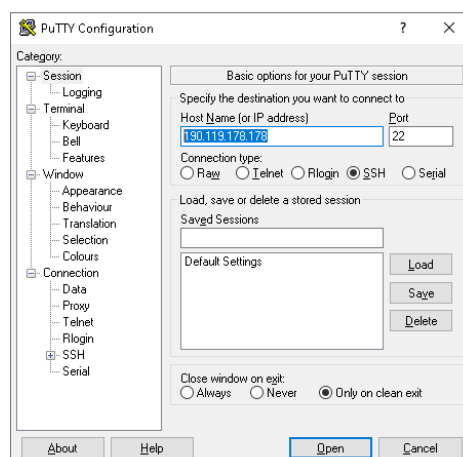


Figura 1: Interfaz de inicio del programa Putty

Tras colocar **inkari.unsa.edu.pe** (o su IP: inkari.unsa.edu.pe) en el campo Host Name, nos conectamos presionando el botón Open. Si es la primera vez que nos conectamos al clúster, se debe aceptar la alerta, mostrada en la figura 2:



Figura 2: Alerta mostrada por Putty, que solicita agregar el host como una fuente

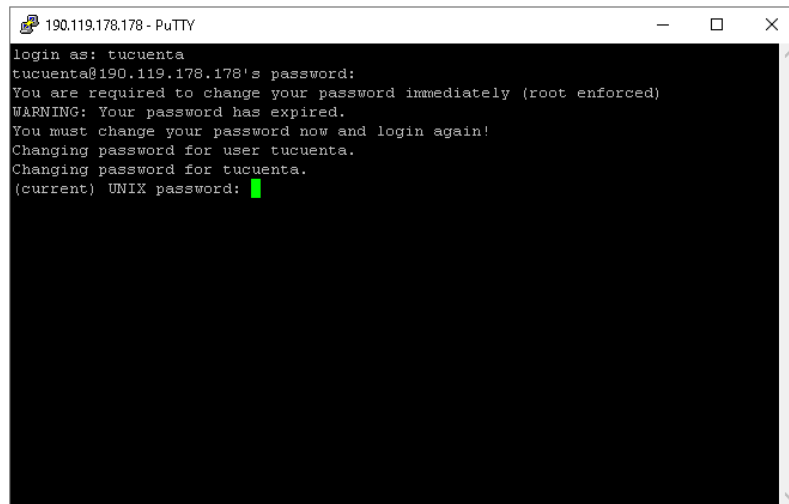
A continuación el cliente ssh, abrirá una terminal, en la cual se debe de introducir el usuario y la contraseña proporcionada por el administrador.



Figura 3: Cliente SSH, solicitando los datos de acceso

### 3 Primer acceso

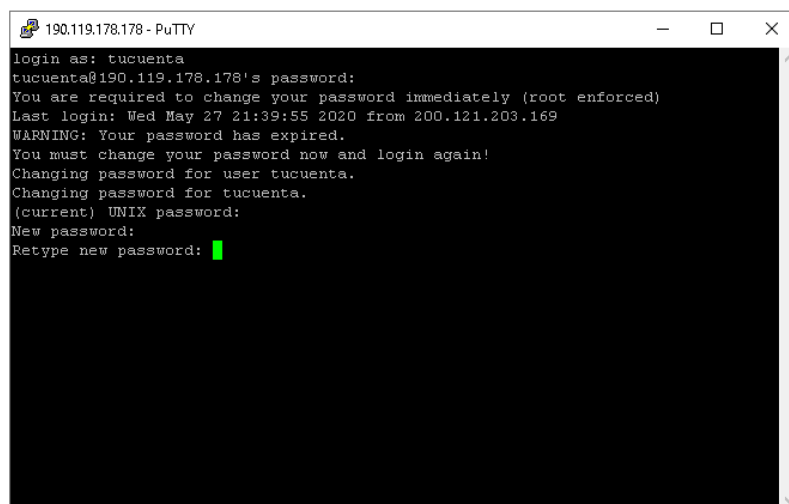
Si es la primera vez que se accede al clúster, será necesario cambiar la contraseña, para ello, tras iniciar sesión, se deberá colocar nuevamente la contraseña proporcionada por el administrador.



```
190.119.178.178 - PuTTY
login as: tucuenta
tucuenta@190.119.178.178's password:
You are required to change your password immediately (root enforced)
WARNING: Your password has expired.
You must change your password now and login again!
Changing password for user tucuenta.
Changing password for tucuenta.
(current) UNIX password: █
```

Figura 4: Cliente SSH, solicitando que se confirme la contraseña asignada por el administrador

Y a continuación se deberá asignar una nueva contraseña, la cual debe de contener 2 números y al menos una letra en mayúscula.



```
190.119.178.178 - PuTTY
login as: tucuenta
tucuenta@190.119.178.178's password:
You are required to change your password immediately (root enforced)
Last login: Wed May 27 21:39:55 2020 from 200.121.203.169
WARNING: Your password has expired.
You must change your password now and login again!
Changing password for user tucuenta.
Changing password for tucuenta.
(current) UNIX password:
New password:
Retype new password: █
```

Figura 5: Cliente SSH, solicitando una nueva contraseña

Una vez cambiada la contraseña, la sesión iniciada finalizará.

## 4 Problemas de acceso

Si tiene problemas para acceder a Inkari, una posibilidad es que ud. haya ingresado de manera **recurrente** una contraseña equivocada y ante ello Inkari haya baneado su IP al detectar dicha acción como una operación sospechosa, en caso de que su IP haya sido baneada, ud visualizará el siguiente mensaje: “Conexión rehusada” (tal y como en la figura 6).

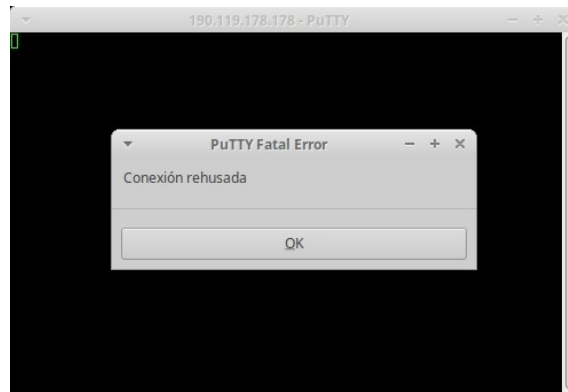


Figura 6: Conexión SSH rechazada en Putty (posiblemente a causa de una IP baneada)

En caso de que crea que su IP ue baneada, ud deberá de enviar un correo a la administración (inkari@unsa.edu.pe), indicando su dirección IP **pública**.

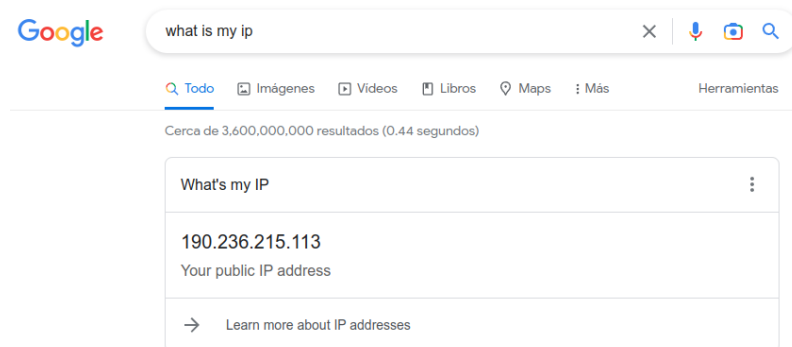


Figura 7: Conexión SSH rechazada en Putty (posiblemente a causa de una IP baneada)

Una forma de saber cuál es su IP pública es realizando una búsqueda en google con los siguientes términos: “*what is my ip*”, tal y como se muestra en la imagen 7.

## 5 Transferencia de archivos

Para la ejecución de un job en Inkari, será necesario que copie todos los ficheros necesarios al clúster, es importante resaltar que todos los archivos almacenados en el clúster no serán respaldados por la administración de Inkari, motivo por el cual se recomienda a los usuarios, tomen las medidas de precaución correspondientes para sus archivos importantes. Los usuarios solo pueden utilizar como almacenamiento de sus archivos sus respectivos espacios de trabajo, el espacio de trabajo de cada usuario puede ser accedido mediante la siguiente ruta absoluta:

/home/<cuenta>

También podrá acceder al espacio de trabajo utilizando la variable de entorno \$HOME.

### 5.1 Transferencia de archivos desde Linux

Para copiar archivos desde una computadora local al clúster, se puede utilizar el comando scp; a continuación se presenta el uso de este comando para subir un archivo al clúster.

```
scp <ruta_del_archivo> <cuenta>@inkari.unsa.edu.pe:<ruta_del_destino>
```

En caso de que se desee descargar un archivo del clúster, el comando a utilizar es el siguiente:

```
scp <cuenta>@inkari.unsa.edu.pe:<ruta_del_archivo> <ruta_del_destino>
```

Si se desea transferir carpetas en lugar de ficheros, se debe de agregar la bandera -r al comando scp. Por otra parte se recuerda al usuario que solo está permitida la transferencia de archivos menores a 200 megabytes.

### 5.2 Transferencia de archivos desde Windows

Una de las opciones para transferir archivos desde windows, es la herramienta WinSCP la cual puede ser descargada desde el siguiente enlace: <https://winscp.net/eng/download.php>. Una vez instalado, WinSCP nos ofrece la interfaz mostrada en la figura 8.

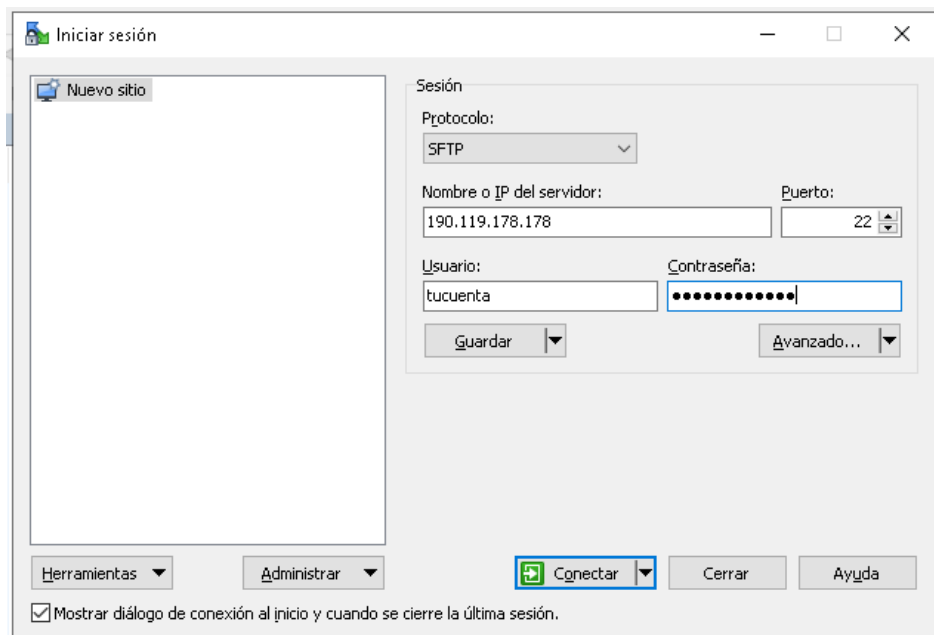


Figura 8: Interfaz inicial del programa WinSCP

Al pulsar el botón conectar (luego de rellenar los campos requeridos), será necesario que se acepte la advertencia mostrada en la figura 9.

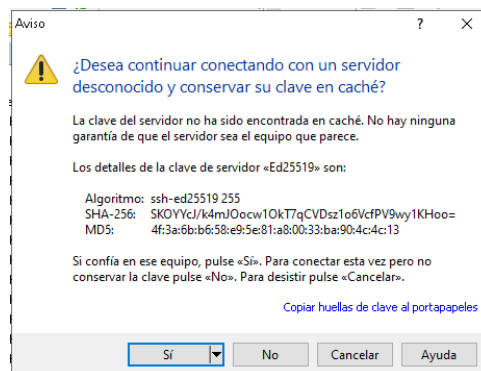


Figura 9: Alerta mostrada por WinSCP, que solicita agregar el host como una fuente confiable

Una vez aceptada dicha advertencia, WinSCP nos mostrará la siguiente interfaz, la cual nos permitirá subir y descargar archivos al clúster Inkari.

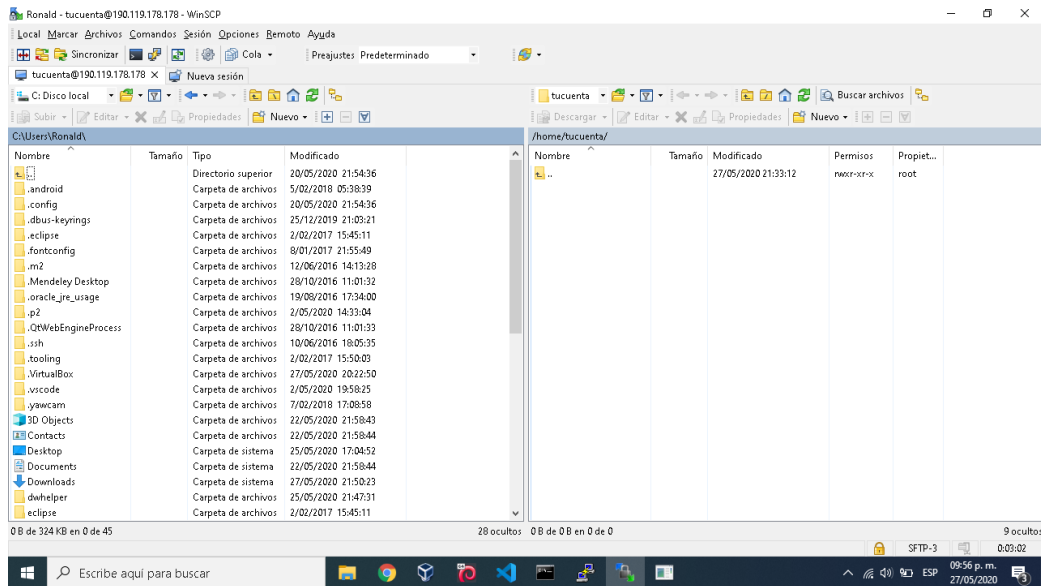


Figura 10: Interfaz de WinSCP, conectado al clúster Inkari

Si la conexión se realizó de forma exitosa, ud. deberá de visualizar la interfaz mostrada en la figura 10.

## 6 Revisión de los recursos

Para revisar si existen recursos disponibles en Inkari, deberemos de usar el comando **sinfo**, dicho comando nos permitirá visualizar el estado de cada uno de los nodos, en la figura 11 se observa un ejemplo de su salida.

```
[rapaza@inkari ~]$ sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
basic*      up          infinite    1     mix  n005
basic*      up          infinite    7     alloc n[001-004,006-007,009]
basic*      up          infinite    4     idle  n[008,010-012]
```

Figura 11: Ejemplo de la salida del comando **sinfo**

En donde, los estados mostrados significan lo siguiente:

Estado	Significado
<b>ALLOC</b>	Indica que el nodo está siendo usado en toda su capacidad.
<b>MIX</b>	Indica que el nodo está siendo usado en forma incompleta, quedando recursos para recibir nuevos jobs.
<b>IDLE</b>	Indica que el nodo está absolutamente libre y operativo para recibir nuevos jobs.

Cuadro 1: Ejemplo de la salida del comando **sinfo**



En caso de que desee visualizar que recursos disponibles en un nodo con estado MIX, se podrá hacer uso del comando **check-resources**, el cual nos mostrará un listado de todos los nodos con la cantidad de cpus y memoria disponible; y que cantidad de estos esta siendo utilizada (esto puede observarse en la figura 12).

```
[rapaza@inkari ~]$ check-resources
NodeName=n001 CPUtot=24 CPUAlloc=24 RealMemory=64300 AllocMem=64300
NodeName=n002 CPUtot=24 CPUAlloc=24 RealMemory=64300 AllocMem=64300
NodeName=n003 CPUtot=24 CPUAlloc=24 RealMemory=64300 AllocMem=64300
NodeName=n004 CPUtot=24 CPUAlloc=24 RealMemory=64300 AllocMem=64300
NodeName=n005 CPUtot=24 CPUAlloc=10 RealMemory=64300 AllocMem=10240
NodeName=n006 CPUtot=24 CPUAlloc=24 RealMemory=64300 AllocMem=64300
NodeName=n007 CPUtot=24 CPUAlloc=24 RealMemory=64300 AllocMem=64300
NodeName=n008 CPUtot=24 CPUAlloc=0 RealMemory=64300 AllocMem=0
NodeName=n009 CPUtot=24 CPUAlloc=24 RealMemory=64300 AllocMem=64300
NodeName=n010 CPUtot=24 CPUAlloc=0 RealMemory=64300 AllocMem=0
NodeName=n011 CPUtot=24 CPUAlloc=0 RealMemory=64300 AllocMem=0
NodeName=n012 CPUtot=24 CPUAlloc=0 RealMemory=64300 AllocMem=0
```

Figura 12: Ejemplo de la salida del comando **check-resources**

## 7 Uso de librerías

Para poder observar el listado de los softwares (módulos) disponibles se debe de ejecutar el siguiente comando:

```
module avail
```

Un ejemplo de la ejecución del comando es mostrado a continuación:

```
[rapaza@inkari ~]$ module avail
----- /opt/modulefiles -----
abinit/8.10.3 (L)  gcc/8.4.0  gcc/11.1.0 (D)  openmpi/2.1.6  openmpi/4.1.0  python/3.9.5 (D)
gcc/6.5.0         gcc/9.3.0  openblas/0.3.15 openmpi/3.0.6  openmpi/4.1.1 (L,D)  qe/6.7
gcc/7.5.0         gcc/10.3.0 openmpi/1.10.7  openmpi/3.1.6  python/2.7.18  stesta/4.1-b4
----- /usr/share/lmod/lmod/modulefiles/Core -----

lmod  settarg

Where:
L:  Module is loaded
D:  Default Module

Use "module spider" to find all possible modules and extensions.
Use "module keyword key1 key2 ..." to search for all possible modules matching any of the "keys".
```

Figura 13: Salida de la ejecución del comando **module avail**

**Nota:** Las librerías listadas en la figura 13, pueden variar con el paso del tiempo.

Si deseamos utilizar una librería, deberemos ejecutar el comando: *module load* y luego el nombre de la librería tal y como aparece en el listado de

*module avail*, por ejemplo, si deseamos hacer uso de la librería abinit 8.10.3, deberemos de ejecutar el siguiente comando:

```
module load abinit/8.10.3
```

**Nota:** También es posible utilizar solo el nombre de la librería (*module load abinit*), sin embargo esto cargará la versión que tiene por defecto dicha librería y en este caso será la 8.10.3 por ser la única disponible.

Finalmente se resalta que al momento de cargar una librería, esta cargará todas las dependencias necesarias, por ejemplo si cargamos la librería abinit y posteriormente ejecutamos el comando: *module list* se observa que también ya se cargo la librería OpenMPI 4.1.1.

```
[rapaza@inkari ~]$ module list
Currently Loaded Modules:
  1) openmpi/4.1.1  2) abinit/8.10.3
```

Figura 14: Listado de módulos cargados, tras cargar únicamente abinit

**Nota:** Algunos ejemplos del uso de las librerías instaladas se encuentran en el github de inkari, para acceder a ellos ingrese a: <https://github.com/unsainkari>.

## 8 Envío de trabajos

El envío de trabajos al clúster Inkari se realiza mediante el gestor de colas SLURM. Para enviar un trabajo al clúster, primero se tendrá que almacenar los archivos necesarios en la supercomputadora y luego solo se deberá de ejecutar el siguiente comando:

```
sbatch script.slm
```

El comando sbatch y el contenido del script (script.slm) permiten especificar ciertos flags (parámetros) que se tendrán en cuenta al momento de ejecutar el trabajo, por ejemplo, si se desea asignar un nombre específico a nuestro trabajo, se tendría que utilizar el siguiente comando:

```
sbatch script.slm --job-name <nombre_del_trabajo>
```

También es posible asignar dichos flags al contenido de nuestro script, para ello se utiliza la siguiente sintaxis:

```
#SLURM --job-name <nombre_del_trabajo>
```

A continuación se describen algunos de los flags usualmente utilizados:

Opción	Ejemplo de valor	Descripción
--job-name	nombre-del-job	Permite asignar un nombre al job
--odelist	n006,n009,n011	Indica una lista de nodos en los que se deberá de ejecutar el job
--nodes	1	Usado para asignar la cantidad de nodos a utilizar
--ntasks-per-node	12	Usado para asignar el número de tareas por nodo
--mem-per-cpu	2G	Indica la cantidad de memoria que usará cada cpu
--mem	24G	Indica la cantidad de memoria que usará todo el job
--output	output- %j.out	Usado para establecer el nombre del archivo de la salida estándar
--error	output- %j.err	Usado para establecer el nombre del archivo de la salida de error
--time	720:00:00	Utilizado para asignar el tiempo máximo de ejecución del job

Cuadro 2: Listado de opciones para el envío de jobs

Para profundizar en los flags que pueden ser utilizados, se recomienda la lectura de la siguiente web: <https://slurm.schedmd.com/sbatch.html>.

**Nota:** Si desea ver un script básico que hace uso de las opciones anteriormente mencionadas, ingrese a: <https://github.com/unsas-inkari/template-basic>.

## 9 Monitorización de jobs

Para monitorear los trabajos enviados se puede usar el comando *squeue*, un ejemplo de su uso es mostrado a continuación.

```
squeue -l
```

La ejecución de este comando con la bandera **l** proporciona una mayor información, tales como: el identificador del trabajo (columna 1), el nombre del trabajo (columna 3), el estado del trabajo (columna 5), entre otros, un ejemplo de esto puede verse en: 15.

```
[rapaza@inkari ~]$ squeue -l
Wed Aug 11 18:33:59 2021

```

JOBID	PARTITION	NAME	USER	STATE	TIME	TIME_LIMI	NODES	NODELIST(REASON)
695	basic	test-qe	lvillega	RUNNING	11:56:46	30-00:00:00	4	n[001-004]
531	basic	Sn02-Vsn	baparici	RUNNING	1-06:00:59	30-00:00:00	2	n[005-006]

Figura 15: Monitorización de trabajos enviados, mediante el comando *squeue*

## 10 Detención de la ejecución de jobs

Finalmente, en el caso de que se desee cancelar un trabajo enviado, se debe de utilizar el siguiente comando:

```
scancel <identificador_del_trabajo>
```