POLITÉCNICA
"Ingeniamos el futuro"

CAMPUS
DE EXCELENCIA
INTERNACIONAL

CeSViMa
CENTRO DE SUPERCOMPUTACIÓN Y VISUALIZACIÓN DE MADRID

Guía del usuario

Edición junio/2016

Contenido

Prólogo

> «Sólo es posible avanzar cuando se mira lejos. Solo cabe progresar cuando se piensa en grande. »
>
> — José Ortega y Gasset

La supercomputación está en auge en estos días. La proliferación de centros de supercomputación a lo largo de la geografía española así lo atestigua.

Barcelona y Madrid fueron los pioneros que dieron los primeros pasos con las máquinas que durante casi un lustro fueron las más potentes de España: Marenostrum y Magerit. Estas instalaciones fueron el germen de la Red Española de Supercomputación (RES) que, en el momento de su puesta en marcha, aglutinaba a los siete supercomputadores más importantes del país.

Pero disponer de los recursos no es lo más importante. Lo realmente importante es el uso que se haga de ellos: la explotación y el aprovechamiento científico de los mismos es realmente su razón de ser.

Por ello, toda documentación de ayuda e información proporcionada a los investigadores cobra especial relevancia ya que facilita el camino al fin último de estas instalaciones: el progreso científico.

El equipo de administración

Sobre esta guía

A finales del año 2004, la Universidad Politécnica de Madrid aunó esfuerzos con IBM y el Centro Nacional de Supercomputación para crear el Centro de Supercomputación y Visualización de Madrid (CeSViMa). El CeSViMa, situado el Campus de Excelencia Internacional de Montegancedo una de las sedes del Parque Científico-Tecnológico de la UPM, se centra en el almacenamiento masivo de información, computación de altas prestaciones, visualización interactiva avanzada, virtualización y computación en la nube.

Dentro del área de computación de altas prestaciones oferta recursos mediante el supercomputador Magerit. La segunda versión de este supercomputador se convirtió en el más potente de España en la lista TOP500 de junio de 2011[1]. También se posicionó como el más ecológico de España y uno de los más ecológicos del planeta[2]

Ilustración 1 La lista Green500 emite un certificado a cada centro indicando su posición en la misma. Gracias a la segunda versión de Magerit, el CeSViMa alcanzó la posición 18 en Junio de 2011

Esta guía del usuario de Magerit proporciona una visión global de los sistemas y de los elementos que lo componen. También se describen los recursos y servicios disponibles destinados a supercomputación y cómo utilizarlos.

Audiencia

Esta guía está dirigida a los usuarios que tienen algún tipo de relación con el supercomputador Magerit.

Aunque se pretende minimizar los conocimientos previos necesarios, se requieren conocimientos del sistema operativo Unix/Linux.

---

[1] http://top500.org/lists/2011/06

[2] http://www.green500.org/lists/green201106&green500from=1&green500to=100(puesto 18)

Guía de lectura

Dependiendo del perfil de usuario, existen partes de este manual que tendrán mayor o menor relevancia:

- **Nuevos usuarios.**

  Tienen una información general del sistema y sus condiciones de uso en los capítulos *Visión general* y *Condiciones de uso*. También disponen de una guía de iniciación básica en el capítulo *Primeros pasos*. A los usuarios de Windows les puede interesar el anexo *SSH en Microsoft Windows* en el que se explica cómo configurar los clientes de acceso al sistema.

- **Líderes de proyecto.**

  Si se encargan únicamente de la coordinación del proyecto y publicación de resultados, se pueden centrar en los capítulos *Visión general*, *Condiciones de uso y Servicios.*

- **Investigadores y desarrolladores.**

  Disponen de información general del sistema, y sus condiciones de uso en los capítulos *Visión general y Condiciones de uso*. En el capítulo *Ejecución de trabajos* se describe el protocolo para preparar y ejecutar trabajos en la máquina y, por último, en *Preguntas frecuentes (FAQ)* tiene información acerca de los problemas y dudas más consultados.

Última versión de la documentación

Debido a las características del sistema, es muy difícil mantener la documentación constantemente actualizada. La información más completa es esta guía de usuarios cuya última versión estará disponible en la página web del CeSViMa.

Asimismo, la documentación más reciente, noticias, conferencias, seminarios y cualquier otra información de interés se encontrarán disponibles en la misma página web.

Contactar con el CeSViMa

Si tras la lectura de la documentación proporcionada existen dudas o surgen problemas relacionados con el uso del supercomputador, se puede contactar usando el Centro de atención a usuarios[3].

[3] Ver página 26

Visión general

El supercomputador Magerit es un clúster de propósito general con arquitectura dual (Intel y POWER) que permite cubrir la mayor parte de las necesidades de cómputo.

Además de la capacidad de almacenamiento local de cada nodo, el sistema dispone de servidores de disco que dan acceso a un total de 432 TB de disco compartido entre todos los nodos.

La configuración POWER es capaz de proporcionar una potencia sostenida de más de 72 TFLOPS sobre un pico teórico de casi 103'5 TFLOPS[4]. La partición Intel proporciona una potencia sostenida de más de 14'8 TFLOPS sobre un pico teórico de 15'9 TFLOPS.

Nodos

Magerit tiene dos tipos diferentes de nodos con distintas características cada uno:

- **Arquitectura POWER7**

  245 nodos BladeCenter PS702 cada uno de los cuales dispone de 2 procesadores con 8 cores cada uno (16 cores) POWER7 de 3'3 GHz (422'4 GFlops) con 32 GB de RAM y 300 GB de disco duro local.

- **Arquitectura Intel**

  41 nodos con 2 procesadores Intel Xeon E5-2670 de 8 cores a 2.6GHz (332 GFlops) y con 64 GB de RAM.

Aunque todos los nodos de una misma arquitectura tienen una configuración hardware y software idéntica, se dividen en dos funcionalidades básicas:

- **Interactivos o de login**

  Tienen habilitado el acceso al exterior y se utilizan como punto de entrada al sistema. En ellos se realizan labores de edición, compilación, gestión de trabajos e intercambio de ficheros.

  En estos nodos no se permite la ejecución de servicios o cálculos, que son cancelados de forma automática.

  Existen nodos interactivos para cada una de las dos arquitecturas de la máquina:

  - Intel: `xmagerit.cesvima.upm.es`
  - POWER: `pmagerit.cesvima.upm.es`

  Tras cada una de las dos direcciones se encuentran varias máquinas entre las que se repartirá la carga a través de un mecanismo de asignación *round-robin*.

  La conexión del CeSViMa con el exterior se realiza a través de la Universidad Politécnica de Madrid (UPM) y RedIRIS, mediante enlaces de 1Gbps a los que sólo tienen acceso estos nodos interactivos.

[4] http://top500.org/system/177311

- **Cómputo**

  Tienen como única misión ejecutar los trabajos de usuario. Estos nodos están completamente aislados del exterior y sólo son accesibles desde el gestor de trabajos.

Sistema de ficheros

Todos los nodos tienen acceso a un espacio de almacenamiento compartido y distribuido con una capacidad bruta total de 432 TB, que utiliza un sistema distribuido y tolerante a fallos denominado GPFS[5].

Cada proyecto tiene acceso a dos ubicaciones diferenciadas:

- **/home**

  Contiene toda la información de cada uno de los proyectos. Cada proyecto dispone de su propia entrada en esta ubicación.

  Dentro de cada directorio de proyecto se encuentran los directorios personales (*home*) de los usuarios que pueden utilizarse para almacenar la configuración del sistema y sus datos personales.

  Dentro de cada proyecto existen dos directorios especiales:

  - /home/<código_proyecto>/PROJECT

    Proporciona un espacio compartido por todos los miembros del proyecto, por lo que puede utilizarse para almacenar los datos o código que sean usados por múltiples usuarios del mismo proyecto.

  - /home/<código_proyecto>/SCRATCH

    Es el espacio de almacenamiento temporal. Cada usuario puede utilizar este espacio para almacenar información necesaria durante la ejecución de los trabajos.

    Los archivos con una antigüedad superior a una semana pueden ser eliminados automáticamente.

  El sistema de ficheros GPFS se encuentra controlado por un sistema de cuotas asignadas a cada grupo, es decir, se considera el total de espacio usado independientemente del miembro que lo utiliza. La coordinación del uso de este espacio de almacenamiento recae sobre el investigador responsable del proyecto.

  No se proporciona servicio de *backup* de ninguno de los sistemas de ficheros, por lo que es responsabilidad de cada usuario y/o investigador principal, realizar y gestionar sus propias copias de seguridad.

- **/sw**

  Las aplicaciones y bibliotecas que proporcione la distribución del sistema operativo se encontrarán en sus rutas originales.

[5] IBM. General Parallel File System. http://www.ibm.com/software/products/software

Por otro lado, el software adicional que se ha instalado en el sistema se encuentra disponible en el directorio /sw. Dentro del mismo, se encontrarán agrupadas las aplicaciones atendiendo a la biblioteca de paso de mensajes disponibles para ese sistema.

Las aplicaciones que no presenten limitaciones legales o contractuales, como puede ser la necesidad de una licencia de uso, estarán disponibles para todos los usuarios del sistema, mientras que las aplicaciones con estas restricciones estarán en directorios privados con control de acceso. Si un usuario desea utilizar una aplicación con restricciones deberá acreditar que está en posesión de una licencia válida.

Para solicitar la instalación de nuevas aplicaciones es necesario contactar con el Centro de Atención a Usuarios a través de support@cesvima.upm.es.

Las modificaciones especiales sobre aplicaciones no estarán disponibles, como norma general, en la zona pública y deberán instalarse en la zona personal del usuario o en la zona compartida de proyectos previa autorización expresa del CeSViMa.

Redes

Para la interconexión de todos los elementos se dispone de varios *routers* para redes Ethernet e Infiniband. Estos *routers* dan soporte a dos redes diferentes que están accesibles desde los nodos:

- **Red Infiniband:**

  Es una red de alto rendimiento y baja latencia, utilizada para las comunicaciones de las aplicaciones paralelas. Proporciona un ancho de banda de 40 Gbps con latencias inferiores a 1 microsegundo.

- **Red Ethernet:**

  Se trata de dos redes que se utilizan para cargar el sistema operativo y acceder al sistema de ficheros GPFS, respectivamente.

  Los nodos interactivos disponen de una tercera conexión de red Ethernet hacia Internet

Primeros pasos

Acceso a Magerit

El acceso al sistema se realiza mediante un conjunto de nodos denominados interactivos. Estos nodos tienen habilitado el acceso desde el exterior mediante el protocolo SSH y permiten el uso de los intérpretes más comunes (Bourne-shell ó C-shell).

El número de nodos interactivos varía en función de las necesidades del servicio[6], accediéndose a ellos mediante las direcciones **`pmagerit.cesvima.upm.es`** (arquitectura POWER) o **`xmagerit.cesvima.upm.es`** (arquitectura Intel) que redirigirán automáticamente a uno de los nodos disponibles con dicha arquitectura. Este mecanismo distribuye automáticamente a todos los usuarios entre los nodos interactivos disponibles.

Límites de uso interactivo

Las sesiones abiertas en el sistema se cancelan tras 24 horas de actividad. Se establece un límite de 10 minutos de ejecución para cada proceso en un nodo interactivo.

La ejecución de servicio o cómputo en los nodos interactivos no está permitida y será cancelada de forma automática.

Información de acceso

El acceso al servicio está controlado mediante mecanismos de autenticación. Los datos de acceso se comunican a cada usuario de forma individual por correo electrónico utilizando para ello la dirección que el propio usuario haya indicado al cumplimentar el registro de la cuenta. Esta dirección de correo electrónico es el único medio de comunicación autorizado para recibir notificaciones e información de acceso, por lo que debe mantenerse activa, debiéndose comunicar cualquier modificación de la misma con la mayor brevedad posible contactando con el Centro de atención a usuarios[7].

Una vez que se envía el identificador y clave de acceso asociada al correo registrado, el usuario será el único responsable de la utilización que se haga de su cuenta, comprometiéndose a comunicar inmediatamente cualquier sospecha que tenga sobre posibles usos no autorizados, pérdida, robo o extravío de la autenticación. Debiendo recordar que las cuentas tienen carácter personal e intransferible, no pudiéndose compartir con otras personas.

La información de acceso consiste en un identificador o nombre de usuario (*login*) y su correspondiente clave de acceso (*password*). La clave es generada automáticamente por el sistema y enviada por correo electrónico sin almacenar ninguna copia legible y sin que el personal del CeSViMa tenga conocimiento de la misma. Esta medida de seguridad

[6] La dirección `magerit.cesvima.upm.es` referencia los nodos de arquitectura POWER.

[7] Ver página 26

hace que, en caso de no recordar los datos de acceso, el personal únicamente pueda generar y enviar una nueva clave a la dirección registrada.

Cambio de contraseña

El cambio de la palabra clave se realiza utilizando el habitual comando **`passwd`** en cualquiera de los nodos interactivos.

Calidad de las contraseñas

CeSViMa se reserva el derecho de analizar de forma sistemática la calidad de las contraseñas utilizadas. Para ello se utiliza un software automático que realiza una serie de ataques básicos con el objetivo de adivinar la palabra clave.

Las contraseñas que puedan ser descifradas se consideran débiles y vulnerables. Para evitar esta potencial vulnerabilidad del sistema se notificará al propietario de la cuenta esta situación para que modifique la contraseña. Si al tercer aviso no se atiende esta petición se procederá a bloquear la cuenta.

Para la elección de una contraseña robusta es conveniente seguir la guía Recomendaciones para la elección de contraseñas[8].

Entorno

Magerit dispone de un sistema de configuración de entorno de compilación/ejecución denominado Modules. Mediante la carga de los Modules adecuados es posible personalizar de forma dinámica el entorno, eligiendo los compiladores, librerías, versiones de aplicación, etc.

Los Modules disponibles pueden ser consultados mediante la orden **`module avail`** que proporcionará un listado organizado en categorías. Para comprobar los Modules cargados basta con ejecutar **`module list`**.

Para cargar cualquier Module basta con ejecutar **`module add <app>/<version>`**, siendo `<app>/<version>` el nombre y versión de una aplicación concreta. Usando **`module load <app>`** cargará la versión por defecto de la aplicación `<app>`. Este comando modificará el entorno (las variables `PATH`, `MANPATH`, `LD_LIBRARY_PATH`...) para permitir el uso correcto de la aplicación indicada.

El comando **`module rm <app>`** permite descargar el entorno de la aplicación `<app>`, deshaciendo las modificaciones del entorno. Si se han cargado varios Modules el resto no se verán afectados.

Los usuarios tienen la posibilidad de configurar la carga de ciertos Modules en el inicio de sesión. Para ello, el fichero de configuración del intérprete utilizado debe contener la inicialización. Las cuentas de usuario incluyen en el fichero **`.bashrc`** las siguientes líneas:

[8] Ver página 36

```
##
## Load default modules
##

if [ ! -z ${MODULE_VERSION} ]
then
    module load null
fi
```

Para incluir o eliminar nuevos módulos en **.bashrc** debe usarse **module initadd <app>** o **module initrm <app>**.

Con esa configuración, se pueden añadir o eliminar módulos al inicio ejecutando **module initadd <app>** y **module initrm <app>** respectivamente.

Los modules disponibles son un entorno dinámico en continua actualización. Según se instalen y/o actualicen aplicaciones, compiladores y utilidades los modules se verán modificados pudiendo incluso desaparecer.

Si se opta por esta solución hay que tener presente que estos ficheros de configuración, así como todo el contenido del *home*, se encuentran ubicados en un sistema de ficheros compartido por todos los nodos de acceso, sea cual sea su arquitectura. Esto hace que los módulos que se configuren en **.bashrc** se intentarán cargar en el inicio de cada sesión, con independencia de la arquitectura tanto del nodo al que se esté accediendo como del nodo en el que se realizó la configuración.

Gestión de disco

Los sistemas de ficheros tienen habilitado el control de uso mediante un sistema de cuotas. Como norma general, las cuotas se establecen a nivel de grupo, es decir, se contabilizan todos los ficheros de los miembros del grupo independientemente del propietario del mismo. El investigador principal responsable del proyecto es el encargado de gestionar de forma apropiada los recursos asignados.

Por omisión, todos los grupos tienen asignada una cuota de 625 GB. Para consultar los límites de cuota asignados se puede ejecutar **quota -g <grupo>** indicando el identificador de grupo que se desea utilizar.

Si un proyecto necesita una capacidad de almacenamiento mayor que la asignada, el jefe del proyecto deberá solicitar dicha ampliación contactando con el Centro de atención a usuarios[9], justificando los motivos y la capacidad que se precisa.

[9] Ver página 26

Transferencia de ficheros

Magerit dispone de diversos servicios de transferencia de ficheros, tanto entre el sistema y los equipos de los usuarios como entre dos o más sitios de la RES.

Los métodos de transferencia seguros (SCP y SFTP) están habilitados en todos los nodos interactivos. Estos métodos cifran toda la información que se intercambia entre los dos sitios.

Un ejemplo básico de uso sería:

- **SCP:**

  ```
  ordenadorParticular> scp <origen> <destino>
  ```

  Copiar desde magerit a nuestro ordenador particular:

  ```
  u@local> scp [usuario]@magerit.cesvima.upm.es:RutaRemota RutaLocal
  ```

  Copiar desde nuestro ordenador hasta magerit:

  ```
  u@local> scp RutaLocal [usuario]@magerit.cesvima.upm.es:RutaRemota
  ```

- **SFTP:**

  ```
  u@local> sftp [usuario]@magerit.cesvima.upm.es
  sftp> put localfile
  sftp> get remotefile
  ```

Ejecución de trabajos

La ejecución de trabajos en el sistema se gestiona, única y exclusivamente, mediante el gestor SLURM. Este planificador permite ejecutar en el supercomputador tanto trabajos secuenciales como paralelos, reservando los recursos (nodos) según las necesidades de los distintos trabajos. Este proceso dependerá de los recursos solicitados por el trabajo, la disponibilidad de dichos recursos en la máquina y las políticas de ejecución que se hayan definido.

Para poder enviar un trabajo al supercomputador es necesario definir sus características en un *Job Command File* o *jobfile*, especificando los recursos que deben reservarse. El gestor de colas se encargará de buscar y reservar los recursos indicados entre los disponibles, optimizando el uso de la máquina y reduciendo el tiempo de espera de los distintos usuarios. En resumen, el gestor se encarga de maximizar la eficiencia del supercomputador.

Por lo tanto, los pasos para poder ejecutar un trabajo en la máquina se resumen en:

1. Conectarse a uno de los nodos interactivos.
2. Preparar el ejecutable y los datos que se desee enviar al supercomputador.
3. Preparar la definición del trabajo a enviar.
4. Enviar el trabajo al gestor de colas.
5. Esperar a que el sistema asigne recursos y ejecute la carga de trabajo.
6. Recuperar los resultados y enviar nuevos trabajos.

Calidades de servicio

El acceso al sistema se controla mediante calidades de servicio (*Quality of Service, QoS*) que definen la prioridad y restricciones de acceso a los recursos. Cada usuario, en función del proyecto al que se encuentre adscrito, podrá tener acceso a un subconjunto de ellas que se notificará por correo en el momento de su activación. Cada usuario tendrá asignada una *Quality of Service* (*QoS*) para el envío de trabajos determinada por el proyecto al que pertenece.

QoS	Asignado	Max WCL	Max cores	Observaciones
class_a	RES	72 horas	784	Prioridad alta
class_b	RES	36 horas	784	Prioridad media
class_c	RES	24 horas	512	Prioridad baja
premium power	UPM	72 horas	1024	Prioridad alta
premium intel	UPM	72 horas	512	Prioridad alta
standard power	UPM	72 horas	512	Prioridad estándar
standard intel	UPM	72 horas	128	Prioridad estándar

QoS	Asignado	Max WCL	Max cores	Observaciones
basic power	UPM	24 horas	512	Prioridad baja
basic intel	UPM	24 horas	64	Prioridad baja

Todos los proyectos tienen asignada su correspondiente calidad de servicio en función de sus características.

Resumen de mandatos

Desde el punto de vista del usuario, existe una interfaz muy simple que se resume en las siguientes órdenes:

- **jobcancel:**

  Cancela un trabajo que se encuentre en la cola de ejecución, esté esperando o ejecutando.

  El trabajo puede estar tanto en espera (nunca llegará a ejecutar) como en ejecución. Para los trabajos en ejecución se abortarán los procesos y se contabilizarán los recursos utilizados hasta ese momento.

- **jobq:**

  Muestra un resumen con el estado de los trabajos del usuario en el sistema desglosado según su estado.

- **jobstats:**

  Muestra información sobre un trabajo en ejecución o ya ejecutado de manera simplificada y legible incluyendo las características del trabajo y el consumo aproximado de los recursos asignados. Además muestra un pequeño análisis orientativo sobre la eficacia en el uso de los recursos.

  La información estadística únicamente se muestra para los pasos de los trabajos. Cada ejecución de `srun` se considera un paso por lo que si el *jobfile* no incluye ningún `srun`, no se dispondrá de estadísticas de uso de recursos.

- **jobsubmit:**

  Envía un trabajo al planificador para su posterior ejecución.

Definición del trabajo

La definición de un trabajo se realiza mediante un fichero de texto que se denomina *Job Command File* o *jobfile*. Este fichero es al mismo tiempo un script que se encargará de iniciar la ejecución del trabajo. A este fichero de órdenes (*script*) se le añaden directivas especiales que indican el tipo de trabajo y los recursos que precisa. El gestor de recursos sólo procesará las directivas propias, que comienzan por `#@`, mientras que el resto de las líneas se considerarán comentarios.

Los trabajos individuales son los más típicos en el sistema. En este modelo, cada trabajo solicita recursos para ejecutar un único programa que es completamente independiente:

```
#!/bin/bash
#---------------------- Start job description ----------------------
#@ arch             = (ppc64|power|intel)
#@ initialdir       = /gpfs/projects/[project_id]/[data_dir]
#@ output           = res/out-%j.log
#@ error            = res/err-%j.log
#@ total_tasks      = [number of tasks]
#@ wall_clock_limit = [hh:mm:ss]
#----------------------- End job description -----------------------

#------------------------- Start execution -------------------------

# Run our program
srun ./[myprogram]

#-------------------------- End execution --------------------------
```

La plantilla de definición de un trabajo se compone de:

- La primera línea indica el shell que debe utilizarse para interpretar la zona ejecutable del mismo
- Seguidamente se incluyen todas las directivas que definen el trabajo. Estas directivas se consideran comentarios por el shell (precedidos de #).
- Finalmente se dispone el código ejecutable que prepara, lanza la ejecución y recupera los resultados.

SLURM define un conjunto de variables que pueden utilizarse en el propio *jobfile*:

SLURM_JOBID: Identificador del trabajo que se ejecuta.

SLURM_LOCALID: Indica el identificador local de la tarea en el nodo.

SLURM_NNODES: Número de nodos asignados al trabajo.

SLURM_NODEID: Identificador relativa del nodo para el trabajo actual.

SLURM_NODELIST: Listado de nodos en los que está ejecutando el trabajo.

SLURM_NPROCS: Número total de procesos en el trabajo.

SLURM_PROCID: Identificador del proceso en el trabajo (*MPI rank*) para el proceso actual.

Directivas

Las directivas permiten indicar al planificador las características del trabajo que se desea ejecutar. Todas las directivas tienen el formato **#@ directiva [= valor]** y pueden utilizar las variables de entorno del propio sistema.

Directivas obligatorias

Todos los *jobfile* deben especificar obligatoriamente las directivas:

- *total_tasks=N* **[Directiva obligatoria]**

  Indica el número de procesos (CPUs) que se necesitan. El gestor de colas se encargará de buscar y reservar CPUs libres hasta completar este número.

  El valor máximo de esta directiva está limitado por la calidad de servicio que tenga asignada el proyecto en ese momento.

- *wall_clock_limit=hh:mm:ss* **[Directiva obligatoria]**

  Especifica el tiempo máximo que el proceso va a estar ejecutando.

  El valor máximo de esta directiva está limitado por la calidad de servicio que tenga asignada el proyecto en ese momento.

  El gestor tiene en cuenta este valor a la hora de planificar trabajos. Si el valor especificado para *wall_clock_limit* es mucho mayor que el tiempo real que consumirá la tarea, se verá penalizada en el tiempo de espera antes de empezar a ejecutar y no podrá aprovechar el sistema de *backfilling*.

  Si el proceso supera el tiempo indicado, el gestor de colas abortará su ejecución.

  Es recomendable establecer una pequeña holgura para tener cierto margen de error, aunque es una buena política y beneficia a todos los usuarios, el acotar lo máximo posible este valor.

Directivas generales

Asimismo, es posible configurar en detalle las ejecuciones mediante el uso de las directivas:

- *arch=tipo* **[Directiva recomendada para usuarios de la UPM]**

  Indica la arquitectura que se precisa para el trabajo. Los únicos valores permitidos son:

  - *ppc64*
    Equipos con procesadores Power7.
    Esta opción sólo está disponible para usuarios RES. Además es la única válida para estos usuarios.
  - *power*
    Equipos con procesadores Power7
    Opción por defecto en caso de no incluir esta directiva, para usuarios de la UPM.

  - *intel*
    Equipos con procesadores Intel (x86_64)
- *error=fichero*

  Indica dónde se redirige la salida de error del trabajo. Este fichero recogerá la salida combinada de todos los procesos que ejecutan en el trabajo, directa o indirectamente, en cualquiera de los nodos asignados al trabajo.

  Si en el nombre se añade un `%j` se reemplazará por el identificador del trabajo asignado por el planificador.

  La actualización de esta redirección es prácticamente inmediata.

- *initialdir=ruta_directorio* **[Directiva recomendada]**

  Permite establecer el directorio de trabajo del script y todas las rutas especificadas (output, error...) se consideran relativas a este directorio inicial.

  Si no se especifica se considerará el directorio de trabajo (.) en el momento de enviar el trabajo.

  Es aconsejable definir un valor absoluto para esta directiva ya que, el uso de rutas relativas al directorio de trabajo actual, puede generar confusión, errores en la ejecución o pérdida de datos.

- *requeue=yes|no*

  Activa o desactiva el re-encolado automático en el caso de un fallo hardware de un nodo de cómputo asignado. Si no se indica, los trabajos no serán re-encolados (*requeue=no*).

  - *yes*
    El trabajo será re-encolado y se planificará nuevamente como si no hubiera ejecutado.
  - *no*
    El trabajo será abortado y no se volverá a planificar.

  El trabajo re-encolado tendrá exactamente la misma configuración e identificador por lo que reemplazará las redirecciones de entrada/salida, ficheros…

  A efectos de contabilidad, nunca se borrará el consumo realizado por el trabajo que sufrió el fallo. Por lo tanto, el consumo final será la suma del realizado en el primer intento y de los sucesivos re-encolados (si se activa utilizando esta directiva).

- *output=fichero*

  Indica dónde se redirige la salida estándar del trabajo. Este fichero recogerá la salida combinada de todos los procesos que ejecutan en el trabajo, directa o indirectamente, en cualquiera de los nodos asignados al trabajo.

  Si en el nombre se añade un `%j` se reemplazará por el identificador del trabajo asignado por el planificador.

  La actualización de esta redirección no es inmediata pudiendo sufrir retardos.

- *tasks_per_node=N*

  Indica cuántas tareas deben asignarse como máximo en cada nodo. El gestor de colas asignará bloques de *tasks_per_node* tareas a cada uno de los nodos asignados.

  Utilizando esta directiva en combinación con la directiva *cpus_per_task* es posible reservar nodos en exclusiva dejando libres procesadores. Para ello el producto $tasks\_per\_node \times cpus\_per\_task$ debe dar como resultado 16.

Uso de OpenMP

El sistema permite el envío de trabajos híbridos MPI-OpenMP: se ejecutarán tantos procesos MPI como indique la directiva **total_tasks** y cada una de ellas desplegará varios hilos OpenMP.

A efectos de contabilidad, se consideran los recursos asignados a la ejecución. Cada hilo se contabiliza como una CPU ocupada, por lo que las horas consumidas por cada hilo son acumuladas.

Es decir, se contabiliza el equivalente a $total\_tasks \times cpus\_per\_task$ tareas durante el tiempo que dure la ejecución.

Para activar esta funcionalidad, es necesario añadir la directiva:

- *cpus_per_task=N*

  Indica el número de threads OpenMP que deben lanzarse por cada tarea (todos los hilos de cada tarea ejecutan en el mismo nodo). Debido a la arquitectura de cada nodo, este valor deberá ser una potencia de 2 entre 2 y 16

  El sistema se encarga automáticamente de definir la variable OMP_NUM_THREADS adecuadamente. Modificarla manualmente en el *jobfile* puede tener consecuencias imprevisibles o indeseables.

Trabajos múltiples

Asimismo, es bastante usual tener que enviar conjuntos de trabajos relacionados o, incluso, con dependencias más o menos complejas entre ellos. Según las necesidades de cómputo existen múltiples prácticas de envío de trabajos al sistema. Sin embargo, existen tres modelos básicos que son comúnmente empleados: trabajos múltiples, encadenados (*chains*) y por pasos (*steps*).

Adicionalmente, Magerit dispone de un cuarto mecanismo específico para la ejecución masiva de trabajos secuenciales.

Trabajos de ejecución múltiple

Consiste en utilizar un único trabajo que ejecute varios programas diferentes utilizando para ello el mismo conjunto de nodos. Con este fin basta con incorporar al listado básico varias llamadas ***srun*** similares que ejecuten los distintos programas.

```
#!/bin/bash
#----------------------- Start job description -----------------------
#@ arch              = (ppc64|power|intel)
#@ initialdir        = /home/<project_id>/<data_dir>
#@ output            = res/out-%j.log
#@ error             = res/err-%j.log
#@ total_tasks       = <number of tasks>
#@ wall_clock_limit = <hh:mm:ss>
##------------------------ End job description ------------------------

##-------------------------- Start execution --------------------------

## First program / First run
srun ./<myprogram1> <arg1>
## First program / Second run
srun ./<myprogram1> <arg2>

## Second program / First run
srun ./<myprogram2> <arg1>
## Second program / Second run
srun ./<myprogram2> <arg2>

# ... More executions ...

##--------------------------- End execution ---------------------------
```

Esta técnica tiene como principal ventaja la supresión del tiempo de espera en cola para la segunda y subsiguientes ejecuciones. Sin embargo, tiene como inconveniente que es necesario incrementar la solicitud de recursos (solicitar el número máximo de procesadores de todos los trabajos, sumar el tiempo de ejecución de todos ellos...) lo que incrementa el tiempo de espera para la primera ejecución.

Normalmente esta técnica está indicada para ejecuciones del mismo programa con distinto número de argumentos y de corta duración (menos de un día). Al empaquetarlos juntos, se reduce el tiempo medio de espera y no se reciben tantas penalizaciones por el envío convulsivo de trabajos.

Trabajos encadenados (*chains*)

Los trabajos encadenados permiten ejecutar un conjunto de trabajos cuando existen dependencias entre ellos. Su funcionamiento es similar al de trabajos individuales pero al finalizar la ejecución el propio trabajo envía, y en ocasiones genera, el trabajo que debe continuar la labor.

```
#!/bin/bash
##----------------------- Start job description -----------------------
#@ arch             = (ppc64|power|intel)
#@ initialdir       = /home/<project_id>/<data_dir>
#@ output           = res/out-%j.log
#@ error            = res/err-%j.log
#@ total_tasks      = <number of tasks>
#@ wall_clock_limit = <hh:mm:ss>
##------------------------ End job description ------------------------

##-------------------------- Start execution --------------------------

## Run our program
srun ./<myprogram1>

## Queue next step
jobsubmit <jobfile>

##--------------------------- End execution ---------------------------
```

Esta técnica puede ser empleada para superar las restricciones de tiempo máximo en cola: la aplicación en ejecución almacena su estado horas antes de cumplirse el plazo máximo y se reenvía, recuperando el estado en el siguiente trabajo y continuando la ejecución.

Aunque los trabajos encadenados abordan dependencias simples entre procesos, no permiten una solución sencilla a las dependencias múltiples.

Trabajos secuenciales masivos

En algunos casos es necesario ejecutar baterías de trabajos secuenciales en los que el mismo código debe ejecutarse con múltiples combinaciones de parámetros (en ocasiones miles de combinaciones) y cada una de las ejecuciones es completamente independiente de las demás. Aunque este tipo de ejecución puede realizarse utilizando alguno de los esquemas de trabajos múltiples, el sistema no está optimizado para ejecuciones de esta naturaleza.

Para mejorar el rendimiento de este tipo de trabajos se ha desarrollado un gestor de trabajos secuenciales (*seqfarmer*), cuyo uso recomendamos encarecidamente. Este mecanismo utiliza un algoritmo de cola única mediante un *pool* de nodos de cómputo (similar a lo que es habitual ver en la gestión de colas de supermercados).

El "granjero" se encarga de repartir cada uno de los trabajos secuenciales a los nodos disponibles hasta ejecutar todos los trabajos indicados. Debido a la falta de dependencias entre ellos, este programa proporciona un incremento de rendimiento casi proporcional al número de procesadores utilizados.

Para utilizar esta aplicación se precisa definir dos ficheros.

Jobfile

En el que se define la ejecución de la aplicación *seqfarmer*. Se trata de un *jobfile* estándar que lanza la ejecución de la aplicación indicando el fichero en el que se especifican qué trabajos secuenciales deben ejecutarse.

```
#!/bin/bash
##----------------------- Start job description -----------------------
#@ arch             = (ppc64|power|intel)
#@ initialdir       = /home/<project_id>/<data_dir>
#@ output           = res/out-%j.log
#@ error            = res/err-%j.log
#@ total_tasks      = <number of tasks>
#@ wall_clock_limit = <hh:mm:ss>
##------------------------ End job description ------------------------

##-------------------------- Start execution --------------------------

## Run our program
srun seqfarmer -f=<tasksfile>

##--------------------------- End execution ---------------------------
```

Listado de trabajos

Lista los trabajos secuenciales a ejecutar. Este fichero es el que se le pasara como parámetro a la aplicación *seqfarmer* en el *jobfile* definido anteriormente.

```
## Each line is a single sequential job
#@ initialdir = path/to/base/dir
#@ output     = path/to/outfile.log
#@ error      = path/to/errorfile.log
./myprogram 0 0 0
#@ output     = path/to/outfile.log
./myprogram 0 0 1
./myprogram 0 1 0
./myprogram 0 1 1
#@ output     = path/to/outfile.log
#@ error      = path/to/errorfile.log
./myprogram 1 0 0
./myprogram 1 0 1
./myprogram 1 1 0
```

```
#@ output      = path/to/outfile.log
#@ error       = path/to/errorfile.log
./myprogram 1 1 1
./myprogram 1 0 1 > path/to/outfile.log 2> path/to/errorfile.log
```

Este fichero implementa extensiones de algunas de las directivas empleadas en el *jobfile* (`initialdir`, `output` y `error`). Nótese que las directivas del *jobfile* afectan a todos los procesos secuenciales, mientras que las del *tasksfile* sólo afectan al proceso secuencial inmediatamente posterior a su definición y sustituyen a las definidas en el *jobfile*.

Las rutas relativas definidas en el *tasksfile*, lo son respecto a las rutas definidas en el *jobfile*. Las rutas absolutas se respetan en su integridad.

Las directivas implicadas son:

- #@ *initialdir:*
  - *jobfile*:

    Esta directiva es opcional. Si no se inicializa su valor por omisión es el directorio desde el que se ha enviado el *jobfile*.
  - *tasksfile*:

    Esta directiva es opcional. Por omisión, si no se inicializa su valor, se toma la ruta actual definida en el *jobfile*.

  Como resultado la ruta de cada ejecución secuencial es:

```
ruta = jobfile_path/#@ initialdir (del jobfile)/#@ initialdir (del tasksfile)
```

- #@ *output* y #@ *error*
  - *jobfile*:

    Si no se inicializan estas directivas sus valores por omisión se corresponden con las salidas por defecto:

```
salida output = jobfile_path/#@ output (del jobfile)
salida error = jobfile_path/#@ error (del jobfile)
```

  - *tasksfile*:

    La definición de estas directivas **sustituye** las definidas en el *jobfile*:

```
salida output = jobfile_path/#@ output (del tasksfile)
salida error = jobfile_path/#@ error (del tasksfile)
```

También es posible usar redirecciones estilo *bash*, tal y como se ilustra en la última línea del ejemplo del *tasksfile*. Este sistema de redirección es compatible con el resto de las directivas, siendo prioritarias las de *bash*.

El número de procesadores a solicitar en el *jobfile* (*total_taks*) es independiente del número de trabajos secuenciales que se desean ejecutar, pues lo que se define es el *pool* de procesadores a utilizar. Sin embargo, el número máximo de procesos secuenciales simultáneamente en ejecución estará limitado a tantos como procesadores se hayan solicitado menos uno (*total_tasks - 1*), pues se reserva un procesador para labores de coordinación y gestión del paralelismo. Como mínimo es necesario utilizar 3 tareas (1 gestor y 2 trabajadores).

Es aconsejable que el número de procesos secuenciales que se vayan a ejecutar (descritos en *tasksfile*) sea múltiplo del número de procesadores menos uno (*total_tasks - 1*) para aprovechar al máximo el paralelismo. También es aconsejable listar las tareas con mayor duración las primeras ya que aprovecharán mejor los recursos.

También es importante reseñar que se debe incrementar el **wall_clock_limit** para permitir la ejecución usando la fórmula:

$$wall_{seqfarmer} = wall_{sequential} \times \left\lceil \frac{num\_jobs}{total\_tasks\ -1} \right\rceil$$

Por ejemplo, si se lanzan 20 trabajos a 6 procesadores (5 ejecuciones y un coordinador), se ejecutarán unos 4 ciclos de ejecución $\left(\frac{20\ trabajos}{5\ ejecuciones}\right)$. Es decir, el **wall_clock_limit** deberá ser 4 veces el de una ejecución individual más un pequeño margen de holgura.

Asignaciones y consumos

El supercomputador Magerit dispone básicamente de dos tipos de recursos, a saber: unidades de cómputo o créditos (en función del tipo de proyecto) y almacenamiento en disco.

Unidades de cómputo o créditos

Para garantizar el correcto uso de los recursos, se delega en SLURM la responsabilidad de controlar el consumo de las asignaciones realizadas.

SLURM tiene la capacidad de limitar la cantidad máxima de unidades de cómputo o créditos que un proyecto puede consumir. El equipo de administración es el encargado de configurar adecuadamente este límite. Con cada nueva asignación se reinician los contadores de horas consumidas.

Cuando un proyecto alcanza el límite de horas establecido, no puede ejecutar más trabajos y aquellos trabajos que se encuentren en ejecución son cancelados.

Para consultar el estado de las asignaciones y consumos los usuarios tiene a su disposición un par de herramientas desarrolladas por el CeSViMa.

Mandato `projectInfo`

Aquellos usuarios con cuenta en la máquina, tienen disponible un mandato de consola desde el que pueden consultar las unidades de cómputo o créditos disponibles, así como las particiones y las QoS.

```
$ projectInfo
Recopilando información...

Información del proyecto - *******
  Cuenta de usuario                        => *******
  QoS asociada al proyecto                 => *******
  Arquitecturas disponibles para ejecutar  => *******
  Número de Unidades de Cómputo disponible => *******
  Importante:
  Las horas disponibles pueden variar dependiendo de los trabajos en ejecución.
```

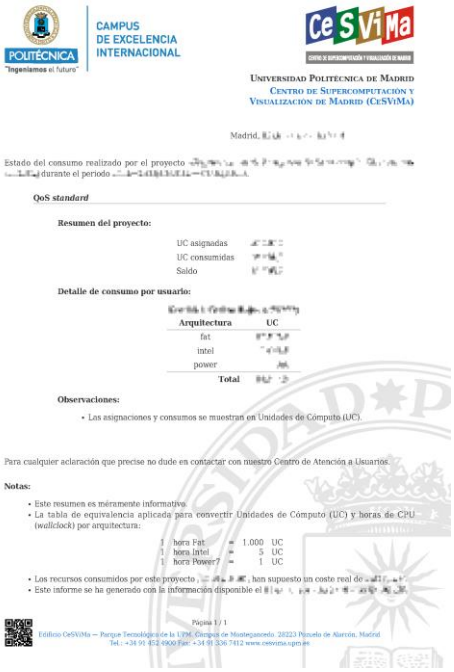
Este mandato devuelve una estimación del número de horas que le quedan al proyecto sin considerar el consumo de los trabajos que estén en ejecución en ese momento. Puede variar en función del instante de tiempo en el que se realice la consulta.

Informes semanales de consumo

Todas las semanas se envía por correo electrónico un resumen personalizado del estado en el que se encuentran todos los proyectos en los que el usuario participa junto con el consumo detallado realizado en cada uno de ellos.

Asimismo, los responsables de los distintos proyectos reciben (también de forma semanal) el detalle del consumo realizado por los proyectos que gestionan desglosado por usuario.

Los informes se generan con la información disponible hasta las 00:00:00 UTC del día en que se generan.



CAMPUS DE EXCELENCIA INTERNACIONAL

UNIVERSIDAD POLITÉCNICA DE MADRID
CENTRO DE SUPERCOMPUTACIÓN Y VISUALIZACIÓN DE MADRID (CESVIMA)

Madrid, [redacted]

Estado del consumo realizado por el proyecto [redacted] durante el periodo [redacted].

QoS *standard*

Resumen del proyecto:

UC asignadas	[redacted]
UC consumidas	[redacted]
Saldo	[redacted]

Detalle de consumo por usuario:

Arquitectura	UC
fat	[redacted]
intel	[redacted]
power	[redacted]
Total	[redacted]

Observaciones:

- Las asignaciones y consumos se muestran en Unidades de Cómputo (UC).

Para cualquier aclaración que precise no dude en contactar con nuestro Centro de Atención a Usuarios.

Notas:

- Este resumen es méramente informativo.
- La tabla de equivalencia aplicada para convertir Unidades de Cómputo (UC) y horas de CPU (*wallclock*) por arquitectura:

1	hora Fat	= 1.000	UC
1	hora Intel	= 5	UC
1	hora Power7	= 1	UC

- Los recursos consumidos por este proyecto [redacted], han supuesto un coste real de [redacted].
- Este informe se ha generado con la información disponible el [redacted].

Página 1 / 1

Edificio CeSViMa — Parque Tecnológico de la UPM, Campus de Montegancedo, 28223 Pozuelo de Alarcón, Madrid
Tel.: +34 91 452 4900 Fax: +34 91 336 7412 www.cesvima.upm.es

Ilustración 2 Detalle del informe de consumo

Almacenamiento en disco

Todos los sistemas de ficheros tienen activo un sistema de cuotas que limita su utilización a la cantidad asignada al proyecto. A efectos de cuota se considera la suma del espacio ocupado por todos los miembros del proyecto. Para evitar situaciones límite todos los proyectos incluyen un margen de seguridad.

Para consultar el estado de la cuota disponible en el sistema se debe ejecutar `quota -g [grupo]`. Aunque hay que tener en cuenta que los mandatos **du**, **quota** y **ls** generan resultados distintos, ya que cada una realiza una labor diferente:

- **ls**

  Con los parámetros adecuados (Ej.: -l), retorna el tamaño lógico de la información almacenada. Esta información no considera particularidades del sistema de ficheros (tamaño de bloque, copias...) por lo que no es válido a efectos de contabilizar la cuota.

- **quota**

  Retorna el tamaño físico de la información almacenada considerando las particularidades del sistema de ficheros. Ésta es la referencia a efectos de contabilidad.

- **du**

  Proporciona una estimación del tamaño físico ocupado en el sistema de ficheros. Su salida suele ser equivalente a la de quota, aunque puede no estar actualizada.

En caso de discrepancias, prevalece la salida proporciona por el mandato **quota**.

Servicios

Además de los recursos de computación, el CeSViMa proporciona una serie de servicios auxiliares a todos los usuarios de sus recursos.

Prácticamente la totalidad de los servicios son activados en el momento que se abre la correspondiente cuenta en el sistema y permanecerán en ese estado hasta el cierre de la misma. Algunos servicios pueden deshabilitarse bajo petición expresa del usuario.

Centro de atención a usuarios

Para solucionar cualquier duda o problema que surja durante el uso de los recursos ofertados se puede contactar con el Centro de Atención a Usuarios[10]. El servicio de atención a usuarios únicamente se presta por correo electrónico en horario de oficina.

En la consulta se debe de indicar siempre que sea posible y de la forma más clara:

- Descripción de la duda o problema surgido.
- Identificación del usuario (*login*) y proyecto asociado, sobre todo si se utiliza una cuenta de correo diferente a la utilizada en el registro.
- Día, fecha y hora aproximados en la que se detectó la incidencia.
- Condiciones en las que se produce.
- Mensajes de error o información mostrados por la máquina en el momento de la incidencia, si se dispone de ellos.
- Si la incidencia es sobre algún trabajo que ejecuta en la máquina, toda la información relativa al trabajo que ocasiona el problema: identificador de trabajo, aplicación ejecutada, mensajes de error, ficheros de log, etc.

Cada una de las comunicaciones iniciadas queda automáticamente registrada en un sistema de control de incidencias asignándoles un identificador único y conservándose su histórico.

Listas de distribución

Todas las direcciones de correo de usuarios y líderes de proyecto son añadidas de forma automática a las listas de distribución del Centro. Estas listas son utilizadas para proporcionar información a todos los usuarios: incidencias de servicio, invitaciones a conferencias o seminarios, etc.

Para evitar el correo no deseado, la lista está cerrada de tal forma que únicamente el personal de soporte del CeSViMa puede enviar correos a las listas.

Asimismo, al finalizar la relación entre el usuario y el CeSViMa la dirección del usuario es eliminada de la lista.

[10] support@cesvima.upm.es

Instalación de software

Además de las aplicaciones básicas incluidas en el sistema operativo, Magerit dispone de un repositorio de aplicaciones en `/sw`. En el repositorio existen versiones de múltiples aplicaciones adaptadas a su funcionamiento en el sistema.

Si se necesita una versión de alguna aplicación no disponible en el repositorio o una actualización a una versión más moderna se puede solicitar contactando con el Centro de Atención a Usuarios.

Asimismo, si el software a emplear precisa licencia es necesario enviar al Centro de Atención a Usuarios una copia de la misma. Este software tendrá protegido el acceso y sólo podrán acceder aquellos grupos o usuarios que acrediten tener una licencia válida.

Condiciones de uso

Todas las personas que hagan uso de los recursos proporcionados por el CeSViMa asumen las siguientes responsabilidades:

1. Todos los usuarios han leído, entendido y aceptan las políticas de uso aceptable y todos sus anexos.

2. El trabajo realizado en los recursos proporcionados por el CeSViMa (incluyendo todos los equipos personales, estaciones de trabajo, servidores y dispositivos proporcionados y/o gestionados por el CeSViMa) debe disponer de una autorización previa.

3. Cuando el trabajo realizado utilizando los recursos proporcionados por el CeSViMa derive en publicaciones, patentes, programas o similares es imprescindible incluir un agradecimiento utilizando, única y exclusivamente, una de las siguientes fórmulas:

    - **Proyectos CeSViMa:**
        - Castellano:

          El autor con agradecimiento reconoce los recursos informáticos, conocimientos técnicos y asistencia proporcionada por el Centro de Supercomputación y Visualización de Madrid (CeSViMa).
        - Inglés:

          The author thankfully acknowledges the computer resources, technical expertise and assistance provided by the Supercomputing and Visualization Center of Madrid (CeSViMa).
    - **Proyectos RES:**
        - Castellano:

          El autor con agradecimiento reconoce los recursos informáticos, conocimientos técnicos y asistencia proporcionada por el Centro de Supercomputación y Visualización de Madrid (CeSViMa) y la Red Española de Supercomputación (RES).
        - Inglés:

          The author thankfully acknowledges the computer resources, technical expertise and assistance provided by the Supercomputing and Visualization Center of Madrid (CeSViMa) and the Spanish Supercomputing Network (RES).

    La información referente a dichos artículos debe enviarse a la dirección de soporte para su inclusión en las memorias anuales del centro e informes de control internos.

4. Además de las publicaciones, el usuario debe proporcionar cualquier tipo de material publicado cuando lo solicite el personal del CeSViMa.

5. El usuario es responsable de la seguridad de sus programas y datos, y debe tomar todas las precauciones necesarias para protegerlos. En particular, el *password* y otras credenciales utilizadas para acceder a cuentas del centro debe ser protegido y **nunca**, bajo ninguna circunstancia, se debe compartir.
   Ante cualquier sospecha de un compromiso de seguridad en su sistema, *passwords*, credenciales o datos, el usuario se compromete a comunicar inmediatamente cualquier uso no autorizado, pérdida, robo o extravío de la autenticación. El usuario será responsable de cualquier daño producido al CeSViMa o cualquier daño resultante del incumplimiento de estas políticas.

6. Está prohibida la compilación e instalación de software en el sistema sin autorización previa del CeSViMa. En ningún caso se autorizará la posesión de un software protegido en el sistema si no se presenta una copia de la correspondiente licencia.

7. Se podrá recopilar información de rendimiento de los programas y trabajos en todas las ejecuciones.

Condiciones especiales para usuarios CeSViMa

Cuando el usuario accede al sistema a través del propio CeSViMa, en la propia solicitud de apertura de cuenta, firma su aceptación de las siguientes:

1. El uso de los sistemas implica respetar las condiciones de las licencias.
2. Las cuentas de usuario son personales e intransferibles. No se debe proporcionar la clave de acceso a terceras partes.
3. Si existiese sospecha de que personas no autorizadas han utilizado o intentado utilizar los recursos, debe ser notificado inmediatamente a CeSViMa.
4. Los recursos asignados deben utilizarse únicamente para las tareas propuestas.
5. Debido a condicionantes de licencias, los usuarios extranjeros y el uso de los recursos desde el extranjero deben negociarse de manera separada.
6. La cuenta de usuario debe protegerse con una clave de acceso razonablemente segura.
7. Cierto software sólo puede utilizarse en entornos académicos. Para el uso por parte de otros usuarios deberá consultarse con CeSViMa.
8. CeSViMa no realiza copias de salvaguarda de los ficheros de usuario. En cualquier caso, CeSViMa declina toda responsabilidad por la pérdida de ficheros debido a fallos del sistema.

Además de las normas generales del Centro y de las aplicables a usuarios del CeSViMa, los líderes de proyecto firman su conformidad con las normas:

1. Asegurar que los miembros del proyecto siguen las normas de usuarios y seguridad establecidas.
2. Verificar los informes de usuario y supervisar que la utilización es la adecuada.
3. Cualquier cambio en la información de contacto debe ser comunicada inmediatamente por correo electrónico al Centro de atención a usuarios[11].
4. El responsable del proyecto informará periódicamente del progreso del trabajo mediante el envío de un breve informe de actividad y resultados obtenidos cada cuatro meses[12].

   Los proyectos que no remitan dicho informe, antes del vencimiento del periodo en vigor, se entenderá que han finalizado o no tienen interés en seguir utilizando los recursos por lo que, tras un plazo de quince días, se procederá a bloquear y borrar las cuentas asignadas al mismo.

   El informe deberá incluir la referencia de los proyectos bajo los cuales se han realizado las actividades, así como los resultados obtenidos, tales como publicaciones, patentes, programas, etc. y una copia del trabajo publicado.

   En función de los resultados proporcionados en los informes se podrán asignar prioridades en el gestor de colas y, por tanto, del uso del sistema.

Condiciones especiales para usuarios RES

Si el usuario accede al sistema en virtud del acuerdo con la RES, son de aplicación las condiciones:

1. Deberá aceptar y enviar firmado el documento *User responsibilities* (disponible en el área RES[13]) que implica un compromiso expreso de aceptación de las políticas impuestas por la RES. Si no se envía dicho documento en el plazo de 15 días se suprimirá la cuenta afectada.
2. El responsable del proyecto debe informar periódicamente del progreso del trabajo realizado. Para ello es necesario enviar un informe en el área RES al menos cada 15 días. Si no se reciben los informes se deshabilitará el acceso a las colas del sistema.

[11] Ver página 26

[12] Las fechas límites son el primer día de los meses de marzo, julio y noviembre

[13] http://www.res.es/

Preguntas frecuentes (FAQ)

Esta sección está en constante actualización y algunas preguntas pueden contener información obsoleta o inválida.

Gestión de trabajos

- **No tengo acceso al sistema de colas, ¿qué sucede?**

  El acceso al sistema se bloquea debido a no cumplir las condiciones de servicio. La causa más habitual es el no envío de informes periódicos de seguimiento o la no aceptación de los términos.

- **He enviado un trabajo al sistema ¿Cómo puedo saber el estado en el que se encuentra?**

  Cada usuario puede ver sus trabajos enviados al sistema mediante la instrucción *jobq*. Es posible generar un informe más detallado junto al desempeño (si ha ejecutado) utilizando la instrucción *jobstats <job_id>*.

- **¿Puedo hacer que mi trabajo esté menos tiempo esperando?**

  La única forma de reducir el tiempo de espera es ajustando al máximo los recursos solicitados, concretamente, reduciendo al máximo el número de tareas (directivas *total_tasks* y *cpus_per_task*) así como la duración del trabajo (directiva *wall_clock_limit*). De esta forma será más fácil que el planificador pueda poner en ejecución un trabajo que precisa una ventana de tiempo más pequeña.

- **¿Cuál es la mejor forma de ejecutar varios trabajos que tienen dependencias entre ellos?**

  Se puede hacer que cada trabajo encole los trabajos que dependen de él (el *script* del *jobfile* acaba con un *jobsubmit <job_script>*)[14].

  Si los trabajos dependen entre ellos sería necesario realizar comprobaciones antes de lanzar el trabajo.

- **¿Cuál es la mejor forma de ejecutar una batería de pruebas secuenciales con múltiples combinaciones de parámetros de entrada?**

  Para una gran cantidad de combinaciones está disponible la aplicación *seqfarmer* [15]que planifica múltiples ejecuciones secuenciales en paralelo optimizando al máximo el uso de los recursos y reutilizando los mismos nodos con lo que se reduce enormemente el tiempo de espera y se evitan penalizaciones por número de trabajos en colas.

[14] Esta técnica se describe en Trabajos encadenados (*chains*) en la página 19.

[15] Ver *Trabajos secuenciales masivos* en la página 20

- **¿Cómo puedo ejecutar utilizando un nodo en exclusiva?**

  Para solicitar nodos en exclusividad hay que solicitar un múltiplo de 16 y especificar la directiva *tasks_per_node* a 16.

  Al solicitar tareas en bloques de 16 al mismo nodo, cada bloque ocupará un nodo completo con lo que se obtiene la exclusividad.

- **¿Se puede utilizar MPI con OpenMP?**

  Sí, sólo es necesario añadir la directiva *cpus_per_task* al *jobfile* tal y como se describe en la sección *Uso de OpenMP*.

Aplicaciones

- **Necesito una aplicación que no aparece instalada en el sistema**

  Las aplicaciones de terceros, para evitar que existan múltiples copias de la misma aplicación en el sistema, son instaladas por el equipo de administración. Solicite su instalación a través del *Centro de Atención a usuarios*.

  Las únicas aplicaciones que son instaladas por el usuario son los desarrollos propios del usuario.

- **Necesito una aplicación que precisa licencia**

  El software con licencia debe ser siempre controlado por el equipo de administración y el acceso al mismo estará permitido únicamente a aquellos proyectos/usuarios que hayan acreditado disponer de una licencia válida.

  Para que se habilite el acceso al software es necesario hacer llegar una copia de la misma al equipo de administración. Para ello basta con ponerse en contacto con el CeSViMa.

- **¿Puedo utilizar Magerit para desarrollar/depurar mi código?**

  Debido a que Magerit es una máquina de uso compartido, el código que ejecuta en ella debe ser suficientemente estable para no perjudicar al resto de usuarios de la máquina.

  Si se necesita ejecutar una versión de código que puede causar algún tipo de inestabilidad, debe notificarse al equipo de administración para su control y seguimiento.

Compilación

- **¿Qué opciones de compilación debería utilizar para compilar en POWER?**

  Las opciones recomendadas para la arquitectura POWER son:

  - Compiladores GNU:
    `-O[2|3] -mcpu=power7 -mtune=power7`
  - Compiladores IBM:
    `-O[2|3|5] -qstrict -qcache=auto -qarch=pwr7 -qtune=pwr7`

- **¿El binario generado debe ser de 32 ó 64 bits?**

  Magerit soporta ambos tipos de binarios.

  Para elegir el tipo de binario a generar es posible utilizar las opciones -q64 o -q32 (compiladores IBM XL) o -m64 o -m32 (compiladores GNU o Intel).

Sistema de ficheros

- **¿Dónde puedo almacenar información compartida por los usuarios de mi proyecto?**

  Cada proyecto tiene un espacio asignado en */home*. Dentro de este espacio existe una entrada especial denominada *PROJECT* (*/home/<código>/PROJECT*) que es accesible por todos los miembros de dicho proyecto y es el sitio indicado para almacenar ficheros de datos, resultados, librerías o programas que utilice todo el grupo.

  Si es necesario almacenar grandes cantidades de información es necesario ponerse en contacto con el equipo de administración.

- **¿Dónde puedo almacenar información temporal?**

  Cada proyecto dispone de un espacio en */home* en el que se puede encontrar una entrada denominada *SCRATCH* (*/home/<código>/SCRATCH*) para almacenar información temporal durante la ejecución del programa. Esta información puede eliminarse automáticamente de forma periódica. Su uso típico es almacenar el volcado de salida y error de los programas.

  Si se precisa almacenar un gran volumen de información o mantener datos durante un periodo de tiempo mayor es necesario ponerse en contacto con el equipo de administración.

- **Me he quedado sin espacio en disco**

  El sistema de ficheros */gpfs* tiene activo un sistema de cuotas que limita su utilización. En cada sistema se calcula la suma del espacio ocupado por todos los miembros.

  Para consultar el estado de la cuota disponible en el sistema se debe ejecutar la orden **quota -g [grupo]**.

- **¿Cómo afecta el uso del sistema de ficheros GPFS al rendimiento de mi aplicación?**

  El sistema GPFS puede ser un cuello de botella del sistema cuando múltiples procesos del trabajo precisan acceder a él durante la ejecución. Una posible mejora del rendimiento es utilizar en su lugar el directorio temporal local a cada nodo (*/scratch*). Este espacio es local al nodo y, por lo tanto, no es accesible desde ningún otro nodo ni cuando finalice la ejecución en curso. Asimismo su contenido se eliminará de forma automática.

  La forma habitual de utilizar este espacio consistiría en iniciar la ejecución copiando la información necesaria al directorio, leer o escribir los datos en dicho

directorio y volcar la información de los discos locales al GPFS al finalizar la ejecución.

Errores típicos

- **Al ejecutar el *job* se obtiene: *bad interpreter: No such file or directory***

  La codificación del retorno de carro es incorrecta, posiblemente usa la codificación de Windows, y el sistema no es capaz de interpretarlo.

  En este caso basta con ejecutar la orden **dos2unix** sobre el *jobfile* para subsanar el problema.

- **El trabajo no escribe nada en la salida**

  El sistema tiene un mecanismo de *buffering* activo en la salida estándar para mejorar el rendimiento reduciendo el número de operaciones de entrada/salida. Es posible que la salida quede almacenada en el *buffer* del nodo que la produce y no se actualice en el sistema de ficheros.

  Al acceder desde otro nodo, éste no verá los datos actualizados, la información se actualizará en el sistema de ficheros pasado el tiempo o al finalizar el trabajo.

- **Cada vez que intento entrar en Magerit usando** ssh **me aparece el mensaje *ssh_exchange_identification: Connection closed by remote host* ¿Qué puede estar pasando?**

  Debido a los ataques recibidos en el sistema, existe un mecanismo de bloqueo automático de las IPs que intentan realizar accesos fraudulentos. Cuando una IP queda bloqueada se recibe ese mensaje en la conexión.

  El bloqueo se produce cuando se realizan múltiples intentos de acceso (más de tres) con credenciales incorrectas (nombre de usuario inválido o *password* erróneo). Al detectar cualquiera de estas condiciones se bloquea el acceso a cualquier servicio desde esa IP en todos los nodos de login del sistema.

  Para liberar el bloqueo, debe enviarse la dirección IP desde la que se realiza la conexión al centro de atención a usuarios.

Miscelánea

- **Nunca he utilizado un sistema Unix/Linux**

  Para poder utilizar el supercomputador es preciso manejar a nivel usuario sistemas Linux. Desgraciadamente, el CeSViMa no puede proporcionar este tipo de formación.

  Existen numerosos libros introductorios que pueden obtenerse en las bibliotecas universitarias. En la mayoría de las escuelas y facultades se imparten cursos y seminarios, *installation parties* o eventos similares.

También existe documentación publicada por diversas asignaturas o por particulares, ya sean introductorios o centrados en el desarrollo[16].

Para principiantes, una buena aproximación es la colección de manuales *Aprenda... como si estuviera en primero*[17] editada por la Universidad de Navarra para dos de sus asignaturas.

- **¿Dónde se puede conseguir más información?**

  La principal fuente de información es la página web del CeSViMa o ponerse en contacto con el Centro de Atención a Usuarios.

- **Mi pregunta no aparece en este listado**

  Puede que se haya añadido recientemente y esté en la última versión del documento que puede obtenerse en la página web del CeSViMa o puede ponerse en contacto con el CeSViMa.

- **¿Cómo puedo contactar con el CeSViMa?**

  Las formas de contacto están en la sección titulada Centro de atención a usuarios en la página 26.

[16] Pablo Garaizar Sagarminaga. GNU/Linux: Programación de Sistemas. 2006.

http://www.e-ghost.deusto.es/docs/2006/ProgramacionGNULinux.pdf

[17] http://www.tecnun.es/asignaturas/Informat1/AyudaInf/

Guías y tutoriales

Recomendaciones para la elección de contraseñas

El primer eslabón en la cadena de tareas a realizar para conseguir acceso no autorizado a un sistema informático, por ser normalmente el más débil, es la obtención de la contraseña de cuentas de usuario con el fin de usurparle la identidad y de esta forma tener acceso a todos los recursos que éste tuviera. Muchas de estas contraseñas son obtenidas fácilmente al tratarse de nombres comunes o datos personales del usuario fácilmente deducibles. Además, las claves elegidas se cambian rara vez o incluso nunca. Por ello, es cuestión de tiempo y pruebas que se adivine la contraseña y se adquiera acceso al sistema.

Este problema cobra importancia cuando se analizan con detenimiento las condiciones de uso del sistema. Al aceptarlas, el usuario se hace responsable del uso fraudulento y de los daños que se hagan con su cuenta.

Existen numerosas técnicas para atacar una contraseña. Desde ataques con diccionarios de claves comunes, heurísticas basadas en datos personales (nombre) o *phishing*. Sin embargo, siguiendo una serie de recomendaciones básicas, la mayoría de las técnicas pierden gran parte de su eficacia.

Características de un buen *password*

Una buena *password* se caracteriza por:

- Es privado, esto es, conocido y usado por una única persona.
- Secreto, ya que no debe aparecer de forma no cifrada en ningún fichero, programa o papel.
- Fácilmente recordable.
- No adivinable o deducible por ningún programa en un tiempo razonable (más de una semana).

Recomendaciones

- *No utilizar contraseñas que sean palabras* en ningún idioma.

  Tampoco es aconsejable utilizar el nombre del usuario o su seudónimo, nombres de familiares o personajes de ficción, mascotas, ciudades, el nombre del propio servicio, etc.
- *No utilizar contraseñas completamente numéricas* con algún significado como el teléfono, DNI, fechas significativas (nacimiento, aniversarios), matrículas, etc.

  Tampoco es aconsejable utilizar *passwords* completamente alfabéticos.
- *Deben ser contraseñas largas* de al menos 8 caracteres.

  Las contraseñas pequeñas pueden adivinarse en un lapso pequeño de tiempo.
- *Usar siempre una mezcla de caracteres numéricos y alfabéticos*

Es aconsejable utilizar letras *mayúsculas y minúsculas* e incluso símbolos (puntos, guiones, arroba...) con lo que se incrementa el tiempo necesario para adivinarla.

  - Todo mayúsculas o todo minúsculas
  - Sólo el primer o el último carácter en mayúsculas
  - Sólo las vocales en mayúsculas
  - Sólo las consonantes en mayúsculas

- *No utilizar la misma contraseña para más de un servicio* siendo aconsejable definir un conjunto de contraseñas básicas con variaciones lógicas según el servicio.

  De esta forma se evita el acceso a múltiples servicios del mismo usuario.

- *Deben ser fáciles de recordar* para evitar tener que escribirlas.

Proteger la contraseña

Además de elegir la contraseña adecuada (no adivinable por métodos automáticos), es necesario protegerla para evitar comprometer su seguridad con ingeniería social.

- *Nunca debe proporcionarse la contraseña* bajo ningún concepto ni requerimiento.

  Esta es la base de las técnicas de *phishing*: solicitar de forma aparentemente oficial el ingreso en el sistema.

- *Nunca compartir con nadie la contraseña*.

  Si se cree que alguien puede conocer la contraseña se debe cambiar inmediatamente.

- *No mantener las contraseñas por defecto del sistema*
- *No escribir la contraseña en ningún medio* y, si se hace, nunca identificarla como contraseña, o con otra información como el nombre de usuario o el sistema.
- *No enviar su contraseña por correo ni mencionarla en una conversación*.

  Como en el caso anterior, si se hace, no hacerlo de forma implícita o proporcionando información de usuario o sistema.

- *No teclear la contraseña si alguien está observando*.

  Como norma de cortesía no se suele mirar el teclado si alguien está tecleando su contraseña.

- *No mantener la contraseña indefinidamente*.

  A pesar de realizar una buena elección es posible que se descubra: alguien puede haberlo visto al teclearlo, capturarlo mediante programas de escucha, instalar un troyano que detecte lo que se teclea, etc. En algunos casos un acceso fallido hace que se escriba el *login* del usuario en lugar de su nombre.

- *Es aconsejable renovar la contraseña* una vez al mes y con una frecuencia no inferior a los tres meses

Ejemplos

A título meramente informativo, se describen algunas técnicas básicas para la elección de contraseñas. Evidentemente, el uso de estas contraseñas no es nada aconsejable.

- Unir palabras cortas con números o símbolos:
  - mi-palabra+clave
  - soy*yo
- Usar un acrónimo de una frase sencilla de recordar:

  EuldlMdcn: En un lugar de la Mancha de cuyo nombre

  Si es una frase desconocida aún mejor:

  mPCeePqno: Mi palabra clave es el password que no olvido
- Inventar una palabra sin sentido pero pronunciable:
  - gamounitos
  - soceflos
- Añadir números o símbolos a una palabra cualquiera incrementa su seguridad:
  - 101gamounitos
  - soce10flos
- También se incrementa la seguridad reemplazando letras por números:
  - g4m0un1t0s
  - m1-p4l4br4+cl4\/3

SSH en Microsoft Windows

El protocolo SSH permite establecer conexiones seguras entre dos ordenadores conectados mediante una red no segura como Internet. Una vez establecida la conexión entre los dos puntos, todo el tráfico entre ellos es cifrado, por lo que se dificulta la posibilidad de interceptarlo. Este protocolo reemplaza a varios de los protocolos existentes como telnet, rlogin, FTP, etc.

Magerit únicamente soporta como protocolo de conexión SSH. Sin embargo, es necesario configurar adecuadamente el cliente para permitir conexiones. La mayoría de las distribuciones Unix/Linux tienen soporte de SSH incluido por lo que no es necesario realizar ningún proceso de configuración. Sin embargo, en el caso de los sistemas Windows, es necesario instalar y configurar un par de herramientas.

PuTTy: cliente SSH

El cliente recomendado para utilizar SSH en sistemas Windows es PuTTy[18], un programa libre cuyo código es la base de prácticamente la totalidad de las aplicaciones Windows que usan SSH.

Es posible descargar únicamente el ejecutable o un paquete de utilidades completo. En cualquiera de los dos casos, es necesario crear un perfil con la configuración apropiada. Para ello basta con replicar la configuración mostrada en la Ilustración 3 y almacenarla. Aunque no es necesario, se aconseja configurar la codificación en UTF-8 (categoría *Window ‣ Translation*).

[18] http://www.putty.nl/

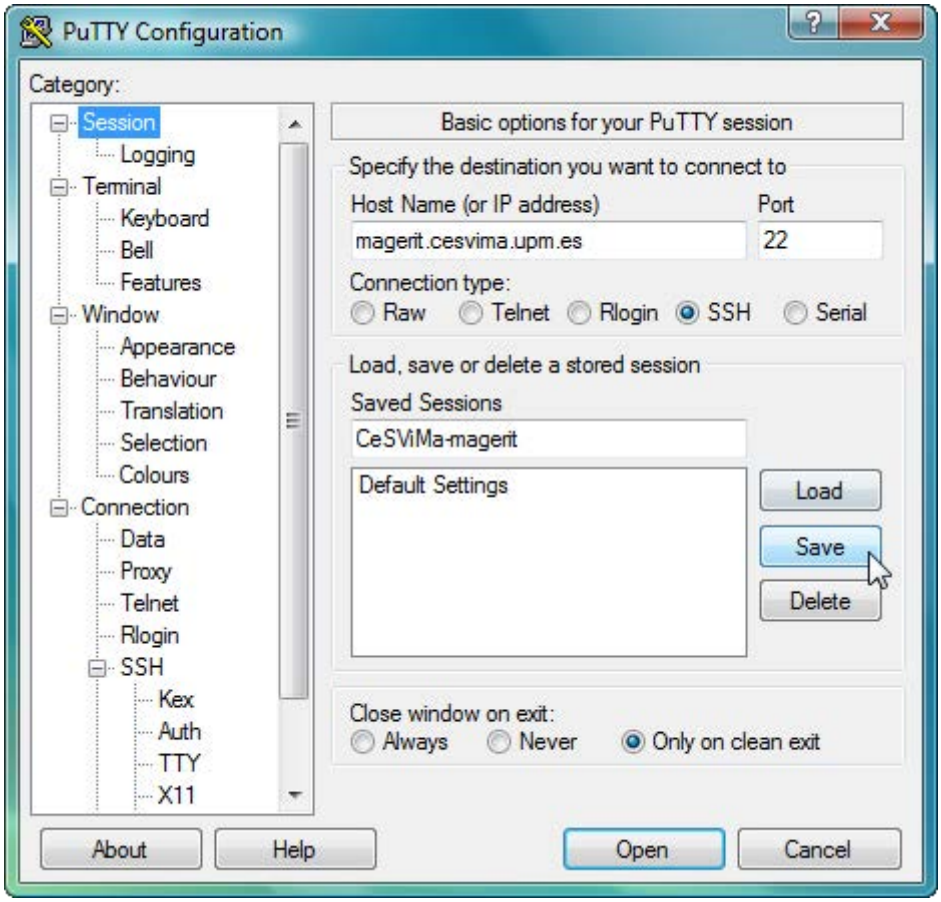


Ilustración 3 Configuración de la sesión

Intercambio de ficheros

La copia de información en Magerit también debe realizarse utilizando el protocolo SSH. Si se ha instalado el paquete completo de PuTTy se dispone de una aplicación de consola que permite realizar esta labor indicando el origen y el destino mediante la sintaxis

```
pscp usuario@magerit.cesvima.upm.es:~/fichero_remoto fichero_local
pscp fichero_local usuario@magerit.cesvima.upm.es:~/
```

También es posible utilizar la aplicación WinSCP[19], que es una interfaz gráfica que permite las transferencias entre la máquina Windows y un servidor remoto. La configuración necesaria para Magerit puede verse en la Ilustración 4.

[19] http://winscp.net/

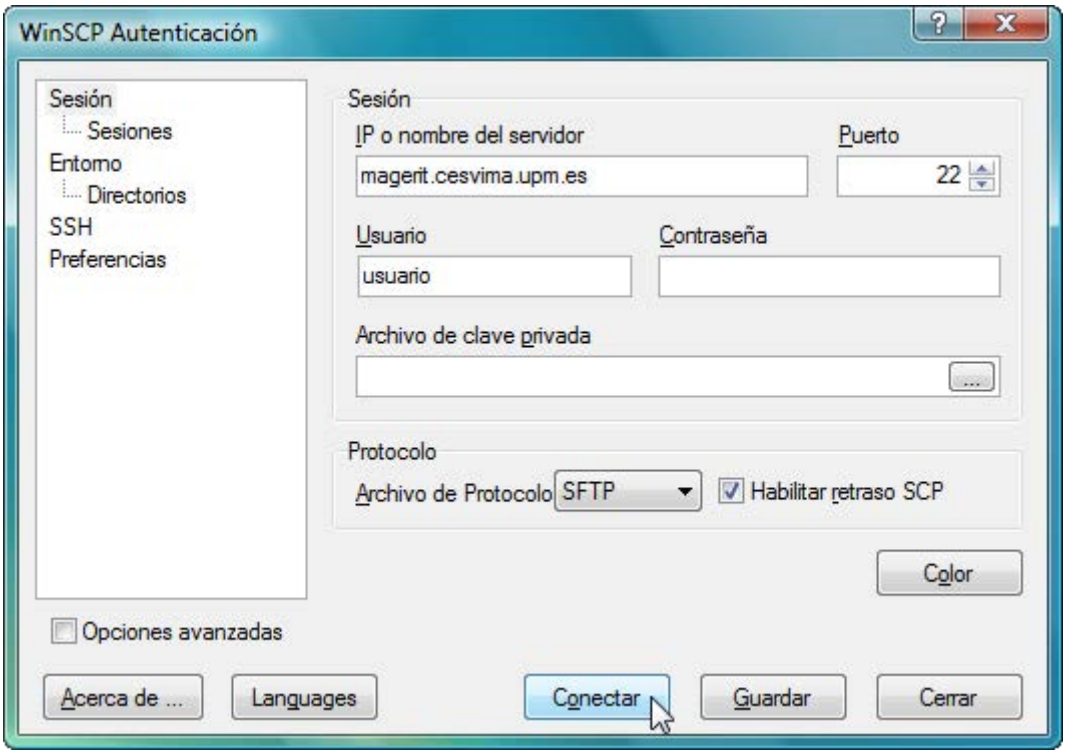


Ilustración 4 Configuración de WinSCP para intercambiar ficheros con Magerit

Conexiones gráficas

Las interfaces gráficas en los sistemas Linux utilizan una arquitectura cliente-servidor. La pantalla es gestionada por el denominado servidor X y las distintas aplicaciones son clientes de dicho servidor. Esta arquitectura permite ejecutar programas en cualquier máquina y presentar los resultados en el servidor de la máquina local.

Para hacer uso de esta funcionalidad desde Windows necesario instalar un servidor X. Existen varios servidores disponibles, siendo una opción muy recomendable Xming[20] o VcXsrv[21], ambas aplicaciones libres.

Una vez instalado se debe ejecutar el servidor antes de utilizar aplicaciones gráficas. Es recomendable añadirlo a la lista de programas que se ejecutan al iniciar la sesión.

Tras tener el servidor activo en el sistema, es necesario configurar apropiadamente el cliente SSH para que permita las conexiones X11. Si se utiliza PuTTy como cliente de SSH, se debe habilitar la opción mostrada en la Ilustración 5.

[20] http://www.straightrunning.com/XmingNotes/ o http://sourceforge.net/projects/xming

[21] http://sourceforge.net/projects/vcxsrv/

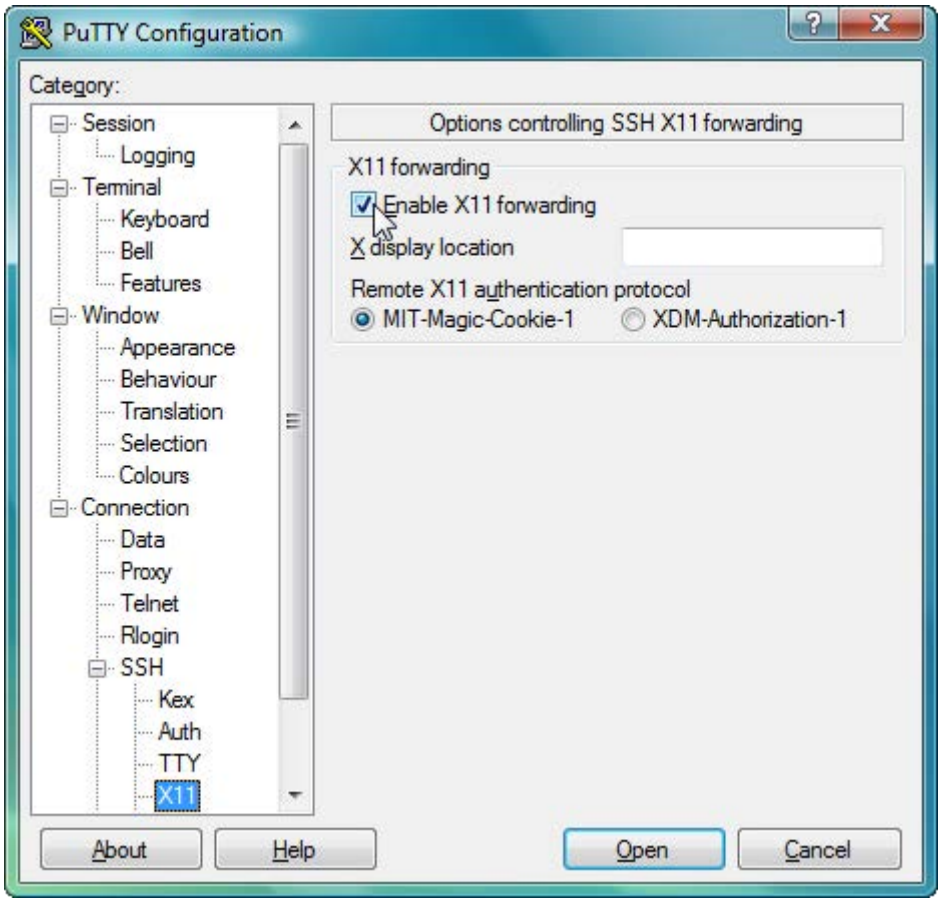


Ilustración 5 Activación del túnel para la conexión gráfica