

FPI - Acceso remoto al laboratorio

Escuela Técnica Superior de Ingeniería de Telecomunicación
Universidad Rey Juan Carlos

gsyc-profes (arroba) gsync.urjc.es

Octubre de 2020



©2020 GSyC
Algunos derechos reservados.
Este trabajo se distribuye bajo la licencia

Creative Commons Attribution Share-Alike 4.0

- 1 Introducción
- 2 Selección del host
- 3 Sesión desde Windows
- 4 Sesión desde macOS

Las prácticas de la asignatura

- Tienen que funcionar en el laboratorio linux de la ETSIT
- Tendrás que entregarlas en el laboratorio

Pero normalmente trabajarás en el ordenador de tu casa, que resulta más cómodo. En la asignatura aprenderás

- A entrar desde casa en tu cuenta del laboratorio linux
- A sincronizar tu ordenador de casa con tu cuenta del laboratorio

Para abrir tu cuenta linux, sigue estas instrucciones

<https://labs.etsit.urjc.es>

Recuerda que esta cuenta es distinta a la cuenta de dominio único de la URJC (que usas por ejemplo para los ordenadores Windows).

Para trabajar en el laboratorio desde casa necesitas

- 1 Elegir una máquina (también llamada *host*) que esté activa
- 2 Abrir una sesión en la máquina usando algún cliente del protocolo ssh (*secure shell*)

Elegir una máquina activa

En esta página web encontrarás el listado de máquinas del laboratorio

<https://labs.etsit.urjc.es/index.php/parte-de-guerra/>

Sugerencia: guarda esta página en tus marcadores, la usarás mucho

- También puedes encontrarla buscando en google *parte de guerra etsit*
- Puedes usar cualquier máquina del campus de Fuenlabrada, ya sea física o virtual
- No importa si hay varias personas en la misma máquina, pero si todos usáis la misma, tal vez podría tener problemas de rendimiento. Por tanto, elige una al azar
- El *parte de guerra* no es completamente fiable. Si alguna máquina no te permite entra, prueba con otra

Ejemplos de nombre de máquina:

f-13202-pc05

f-1-vm04

- Para acceder a un *host* desde cualquier lugar de internet (que no sea el propio laboratorio), es necesario indicar su nombre completo (*FQDN*, *fully qualified domain name*)
- En nuestro caso el FQDN es el nombre de máquina, añadiendo el sufijo `aulas.etsit.urjc.es`

p.e.

`f-13202-pc05.aulas.etsit.urjc.es`

`f-1-vm04.aulas.etsit.urjc.es`

La primera vez que abras sesión en una máquina, el cliente ssh mostrará un mensaje parecido a este

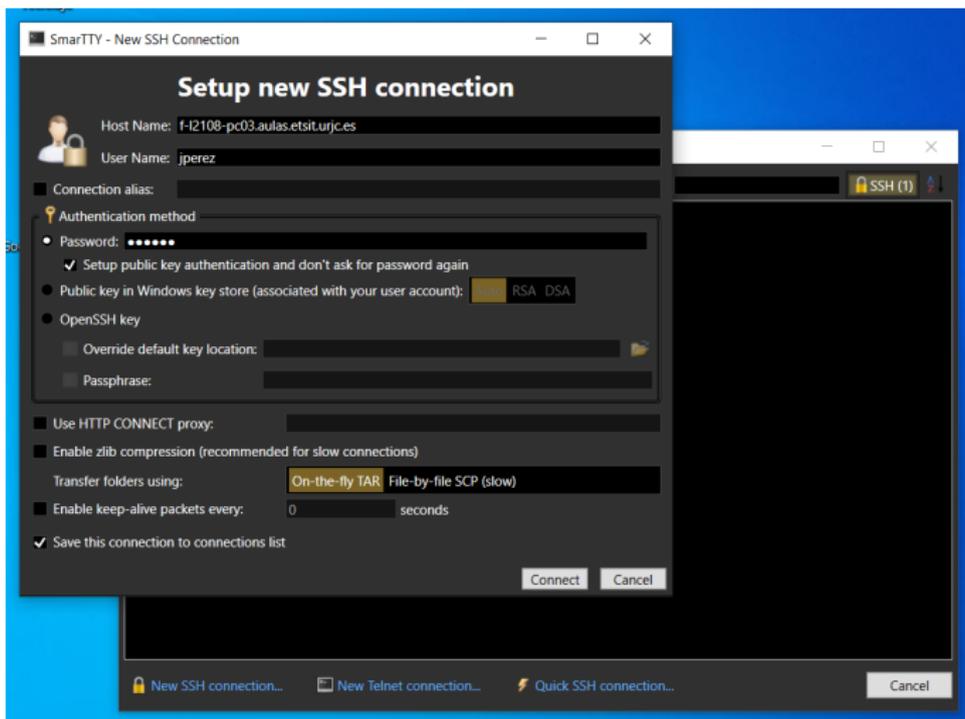
```
The authenticity of host 'f-13202-pc05.aulas.etsit.urjc.es'
can't be established. ECDSA key fingerprint is
SHA256:ucEkxgpIobKhgw0b979NY97fmuaTtWwewdLa//SxVtk.
Are you sure you want to continue connecting (yes/no)?
```

- Esto significa que para estar 100 % seguros de que ningún atacante suplanta la identidad de la máquina deberíamos revisar esta huella digital
- Como no estamos en un entorno especialmente peligroso, podemos contestar yes sin comprobar nada. Esto guarda la huella digital, y no volverá a mencionarla a menos que
 - Suframos un verdadero ataque
 - El administrador reinstale la máquina sin conservar la huella

Sesión ssh desde Windows

Para Microsoft Windows hay muchos clientes ssh disponibles

- Posiblemente el más usado es Putty, pero es un poco antiguo. No permite el uso de pestañas, y la configuración para evitar teclear contraseñas en cada sesión es un poco complicada
- Aquí recomendamos SmarTTY. Es gratuito, sencillo y potente. Cualquier buscador te indicará que puedes descargarlo desde <https://sysprogs.com/SmarTTY>

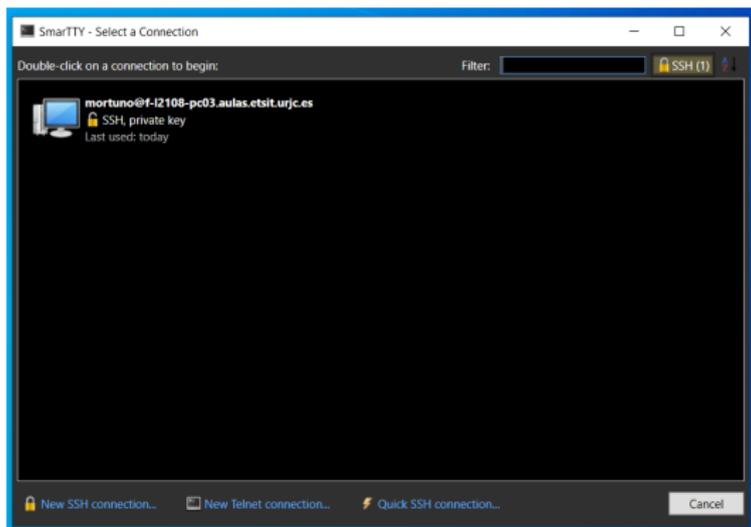


SmarTTY: nueva conexión

- 1 Al iniciar SmarTTY, nos pedirá que seleccionemos una conexión. Elegimos *New SSH connection*
- 2 En el campo *host name* escribimos el nombre completo de la máquina. P.e. *f-12108-pc03.aulas.etsit.urjc.es*
- 3 En el campo *user name* escribimos nuestro nombre de usuario en el laboratorio (*jperez, mgarcia,...*)
- 4 En el campo *password* escribimos nuestra contraseña
- 5 Dejamos el resto de parámetros en su valor por omisión y pulsamos *connect*
- 6 La primera vez que nos conectemos a una máquina nos aparecerá una ventana titulada *Save host key* donde nos mostrará su huella digital. Como no estamos en un entorno especialmente sensible, la guardamos sin más como nos recomienda.

- Si todo ha ido bien, SmarTTY nos preguntará si preferimos un *terminal inteligente (Start with a smart terminal)* o un terminal normal (*Start with a regular terminal*)
- Elegimos el terminal normal y marcamos la opción *remember the choice* para que no pregunte de nuevo. Hecho esto, ya podemos trabajar en la sesión
- Pulsando el icono que representa un signo de más de color verde, podemos añadir una nueva sesión normal en otra pestaña

En este vídeo puedes verlo <https://youtu.be/pV2E5Tfr1aI>



Una vez que hayamos creado la conexión, si en otro momento queremos entrar en la misma máquina no es necesario repetir todos los pasos, los datos de la conexión se habrán guardado y basta con hacer doble click sobre su icono

Sesión ssh desde macOS

En macOS no es necesario instalar ningún programa

- 1 Ejecutamos *Terminal*
- 2 En el menú *Shell* elegimos la opción *Nueva conexión remota*
- 3 En el panel izquierdo (*Servicio*) debe estar seleccionada la opción *Shell segura (ssh)*
- 4 En el panel derecho (*Servidor*), pulsamos el botón con el signo más para añadir la dirección de la máquina a la que nos queremos conectar. P.e. *f-12108-pc03.aulas.etsit.urjc.es*
- 5 En el campo *Usuario* escribimos nuestro nombre de usuario en el laboratorio Linux. (*jperez, mgarcia,...*)
- 6 Pulsamos conectar

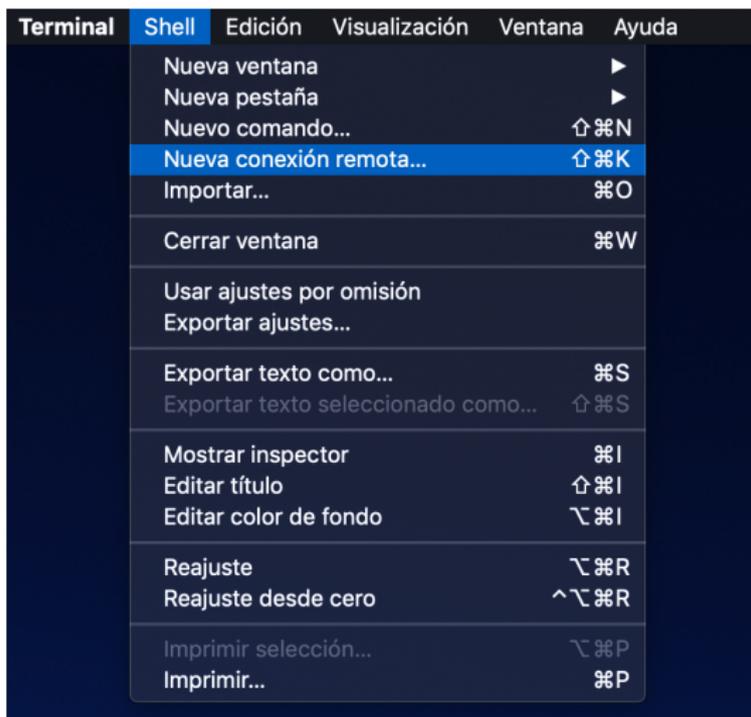
- Para abrir varios terminales, basta pulsar de nuevo *Conectar*
- Normalmente, cuando escribes una contraseña, el cliente no la muestra en pantalla, pero aparece un asterisco o similar cada vez que pulsas una letra

contraseña: *****

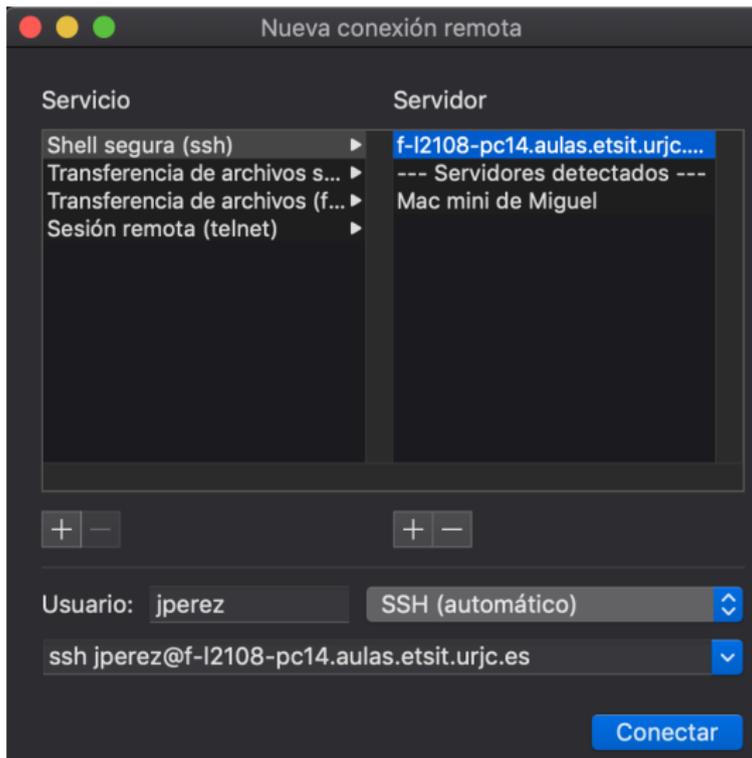
En este caso no, la escritura de la contraseña es completamente invisible. Que esto no te confunda

En este vídeo puedes ver todo el proceso

<https://youtu.be/f8CJINHulgs>



Terminal de macOS



Parámetros de la conexión