PRÁCTICA E1: Instalar un Servidor SSH en Windows / Linux **MF0486_3 Seguridad en equipos informáticos**		**Fecha**	04 / 03 / 2022
Curso	**7.1. MF0486_3 Seguridad en equipos informáticos**	**Plan de Formación**	FC-2021.1/II.000/1914256

Nombre y Apellidos:	JAVIER PRUNELL ACOSTA	**Firma del Alumno:**	
DNI:	54056005W	**Firma del Profesor:**	

Apto: ☐ **No Apto:** ☐ **Calificación:** ☐

Instrucciones Generales

La puntuación máxima será de 10 puntos.
Esta prueba tendrá una duración máxima de 540 minutos
(Temporalizados durante la Unidad de Aprendizaje 1)

El alumno/a deberá acatar las siguientes normas durante la duración de la actividad:

- Rellene el encabezado con su nombre, apellidos y D.N.I.
- Firme en todas y cada una de las hojas entregadas, incluidas las que estén en blanco.
- Usar exclusivamente bolígrafo azul o negro
- Guardar los ficheros generados en una carpeta con nombre **MF0486_3_E1**
- El docente le indicará al final como entregar el contenido de dicha carpeta
- Al finalizar el ejercicio y antes de entregarlo **comprueba tus respuestas**, en caso de duda consulta al docente.

Equipo y material

- Bolígrafo azul.
- Folios.
- Ordenadores.
- Conexión a Internet. (Para buscar información a modo de ayuda)

Instrucciones específicas

Nombre: En esta práctica se pretende que el alumnado se familiarice con el servicio SSH y aprenda a configurarlo adecuadamente de acuerdo a los requisitos deseados mediante los archivos de configuración. Al finalizar la práctica se deberá entregar un guión que explique el proceso y los pasos seguidos para realizar todos y cada uno de los puntos.

Agrupamiento: Individual

Descripción: Los pasos que habrá que seguir para realizar la práctica son los siguientes:

1. Instalar e iniciar un servidor SSH en una máquina Linux (Preferiblemente Ubuntu)
2. Iniciar el Servidor SSH y probar a conectarte desde un cliente SSH de Windows (ejemplo: Putty) al servidor de SSH de GNU/Linux y comprueba si es posible conectarse correctamente desde cualquier equipo.
3. Cambia los protocolos SSH en cliente y servidor y comprueba si se permite la conexión. Por ejemplo, que el cliente use solo la versión 1 del protocolo y el servidor la 2.
4. Por defecto, al autenticarnos correctamente en el servidor SSH, éste nos muestra la fecha y hora de la última vez que nos conectamos. Encuentra la opción que se encarga de modificar este aspecto.
5. El servicio SSH por defecto escucha en el puerto 22. Modifícalo para que arranque en el puerto 10022 (u otro) y averigua cuál sería el comando utilizado para poder acceder al servidor. (ayuda: man ssh).
6. Configurar el servicio SSH para que no admita hacer login como root.
7. Configura el servidor SSH para que solo permita la autenticación de los usuarios que nosotros indiquemos.
8. Configura el servidor SSH para validarnos sin introducir contraseña.
9. Configura el servidor SSH de forma adecuada para que acepte la redirección X11, de tal forma que se puedan ejecutar aplicaciones gráficas de forma remota. Haz pruebas y comprueba su funcionamiento.
10. Utiliza los enlaces proporcionados para completar el trabajo, investigando qué otras opciones podrían ser útiles para la configuración de nuestro servidor SSH.

Instrucciones específicas

Recursos: https://www.openssh.com/manual.html

https://help.ubuntu.com/community/SSH/OpenSSH/Configuring?_ga=2.74147152.312157118.1647942508-1909757974.1644235693

https://manpages.ubuntu.com/manpages/xenial/man1/ssh.1.html?_ga=2.82993364.312157118.1647942508-1909757974.1644235693

https://www.redeszone.net/tutoriales/servidores/servidor-openssh-linux-configuracion-maxima-seguridad/

Duración: 540 minutos

Condiciones de realización:

La actividad se llevará a cabo en el aula y el alumnado contará en todo momento supervisión del docente.

El alumno podrá hacer uso de internet para su realización.

En ella se valorará la utilización de herramientas para la gestión del tiempo y secuenciación del uso de las aplicaciones necesarias. Y se observará especialmente la autonomía del alumnado a la hora de ejecutar y tomar decisiones. Como también la estructuración del ejercicio en donde se solicitará, orden, coherencia y limpieza.

Una vez terminado la práctica se le notificará al docente y pasará a su evaluación.

INSTALACIÓN SSH EN RASPIAN (RASPBERRY PI 4)

Para la instalación de SSH, primeramente, vamos a instalar tanto el cliente como el servidor. Para ello debemos acceder al root del sistema e introducir:

apt-get install openssh-client
apt-get install openssh-server

Para comprobar que el servidor esta en funcionamiento, aplicamos el siguiente comando:

Sudo systemctl status ssh

Posteriormente, vamos a iniciar el servidor SSH y posteriormente, vamos directamente a configurar el archivo sshd_config.

Sudo systemctl start ssh // sudo systemctl restart ssh // sudo systemctl stop ssh
Sudo gedit /etc/ssh/sshd_config

Para ello, modificaremos algunas de las líneas de la documentación del sshd;

- Modificaremos el puerto para evitar que sea por defecto, en nuestro caso: 22445
- Vamos a crear usuarios, para poder darle "allow o deny" dentro del archivo:
  Sudo useradd -a -d /home/username -s /bin/bash sergio
  Passwd sergio
  Sergio 2
- Al crear el usuario, ahora podemos, en el documento: **AllowUsers sergio sergio2**
- Podemos también denegar un supuesto usuario: **DenyUsers adrian adrian2**
- También modificamos el tiempo para introducir la pass: **LoginGraceTime 1m**
- Procedemos a bloquear la fuerza bruta: **PermitRootLogin no**
- Intentos máximos para acceder por contraseña: **MaxAuthTries 5**
- Habilitamos la autentificación con usuario y contraseña: **PasswordAuthentication yes**
- Habilitar la interfaz gráfica por parte del cliente: **X11Forwarding yes**

Guardamos el documento y reiniciamos el servidor:

Sudo systemctl ssh restart

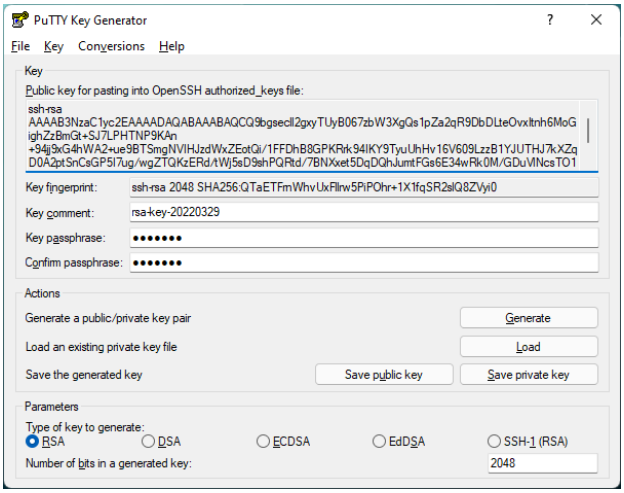
Para comprobar que todo está correcto, debemos de ir a un cliente que se encuentre en otra red, por ejemplo, un móvil, descargarse una APP cliente. Nos pedirá la contraseña y podemos entrar directamente al servidor.

sergio@direccionip:22445

Básicamente la instalación por SSH y su configuración se puede realizar en poco tiempo, pero la idea es generar un acceso más seguro a través de Latch: https://latch.elevenpaths.com/www/
También tendremos que tener en cuenta dos programas, KNOCKD y el FAIL2BAN. En estos casos, uno oculta el SSH y el otro evita sacar contraseñas por fuerza bruta.

CONFIGURACIÓN DEL SSH CON CLAVE PRIVADA Y CLAVE PUBLICA

Primeramente tenemos que generar tanto la clave privada como la clave pública. Para ello, tenemos que acceder al PuTTy Key Generator. Generamos las dos claves y guardamos los dos archivos. La clave pública tenemos que guardarla en el servidor, y la clave privada, siempre tiene que estar en el cliente.



Posteriormente, nos tenemos que conectar al servidor desde un cliente para generar los directorios necesarios y copiar la clave publica dentro del servidor. En este caso, hemos creado el usuario sergio con la contraseña sergio1.

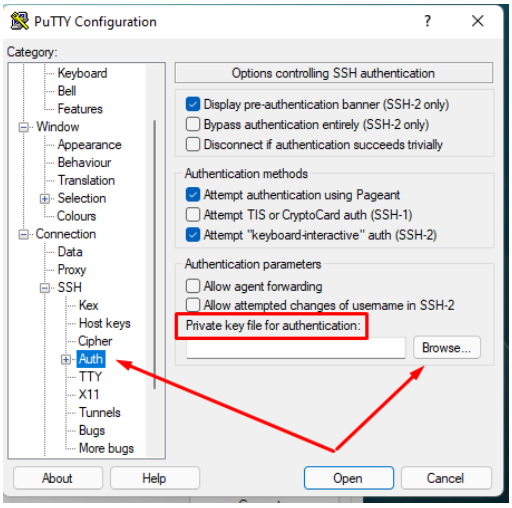
Ssh sergio@192.168.1.44
Mkdir .ssh/
Chmod 777 .ssh
Cd .ssh
Nano authorized_keys

En este caso, hemos creado el archivo authorized_keys. Dentro de este "archivo" vamos a pegar la clave pública:

```
sergio@javier-VirtualBox:~$ ls
sergio@javier-VirtualBox:~$ ls -a
.   .bash_history  .bashrc  .config     .gtkrc-xfce  .profile
..  .bash_logout   .cache   .gtkrc-2.0  .local       .ssh
sergio@javier-VirtualBox:~$ cd .ssh
sergio@javier-VirtualBox:~/.ssh$ ls -a
.  ..  authorized_keys
sergio@javier-VirtualBox:~/.ssh$ cat authorized_keys

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCQ9bgsecll2gxyTUyB067zbW3XgQs1pZa2qR9DbDLt
eOvxItnh6MoGighZzBmGt+SJ7LPHTNP9KAn+94jj9xG4hWA2+ue9BTSmgNVIHJzdWxZEotQi/1FFDhB8
GPKRrk94IKY9TyuUhHv16V609LzzB1YJUTHJ7kXZqD0A2ptSnCsGP5I7ug/wgZTQKzERd/tWj5sD9shP
QRtd/7BNXxet5DqDQhJumtFGs6E34wRk0M/GDuVlNcsTOlEUxzFW/Pv26kIU61WVxsNuC+n8hlEKzFeY
GypELf4J/QZmlTfClccXobTlwyc/hlTf0vJlDqWVI4njcyLWVfy7w7Dc2T2H rsa-key-20220329
```

Posteriormente, salimos del servidor, y volvemos a entrar, pero esta vez, accedemos con nuestra clave privada. Lo haremos desde el Putty seleccionando el archivo de la clave privada:



De esta forma, al acceder, tendremos que ponder la "passphrase" y accedemos con un sistema de seguridad mucho mayor que con una clave principal, que, basicamente se podria sacar por fuerza bruta. Al utilizar la clave privada y clave publica, el nivel de seguridad es tremendamente más alto.

CONFIGURACION DEL FAIL2BAN

Esta herramienta es indispensable a la hora de tener un nivel de seguridad óptico en un SSH, de esta forma, después de la instalación, procedemos a la configuración del archivo.

Apt-get install fail2ban
Nano /etc/fail2ban/jail.conf

Aquí vamos a modificar algunas de las características para mejorar la seguridad.

- BANTIME: es una característica del archivo para bloquear una IP, por defecto está en 10 minutos, pero podemos ponerle el tiempo que consideremos.
- MAXRETRY: es la característica de intentos por IP para después añadirla a la lista negra y bloquearla indefinidamente.
- IGNOREIP: son la lista de las IP ignoradas por fail2ban

Para comprobar el estado del servicio, y visualizar las opciones, tenemos el comando fail2ban-cliente status

Los programas cliente pueden ser de diferente tipo, tanto desde smartphone, como por línea de comandos y por interfaz gráfica para conectarnos al servicio.

INSTALACIÓN DE SSH EN WINDOWS 10

Dentro del sistema operativo de Windows, tenemos que confirmar que tenemos preinstalado OPENSSH. Accedemos a programas y características y verificamos:

Para la configuración, instalación y puesta en marcha del servidor SSH, tenemos que acceder por Power Shell.

```
PS C:\Windows\system32> Get-WindowsCapability -Online | Where-Object Name -like 'OpenSSH*'

Name  : OpenSSH.Client~~~~0.0.1.0
State : Installed

Name  : OpenSSH.Server~~~~0.0.1.0
State : Installed
```

Ahora vamos a proceder a la instalación tanto del servidor como del cliente, para ello:

```
PS C:\Windows\system32> Add-WindowsCapability -Online -Name OpenSSH.Server~~~~0.0.1.0


Path          :
Online        : True
RestartNeeded : False
```

```
PS C:\Windows\system32> Add-WindowsCapability -Online -Name OpenSSH.Client~~~~0.0.1.0


Path          :
Online        : True
RestartNeeded : False
```

Ahora iniciamos el SSH con los siguientes comandos. Aquí tendremos la instalación completa, ahora solamente nos falta acceder por usuario@maquina:puerto.

```
PS C:\Windows\system32> Start-Service sshd
ADVERTENCIA: Esperando a que se inicie el servicio 'OpenSSH SSH Server (sshd)'...
ADVERTENCIA: Esperando a que se inicie el servicio 'OpenSSH SSH Server (sshd)'...
PS C:\Windows\system32> Set-Service -Name sshd -StartupType 'Automatic'
PS C:\Windows\system32> if (!(Get-NetFirewallRule -Name "OpenSSH-Server-In-TCP" -ErrorAction S
ilentlyContinue | Select-Object Name, Enabled)) {
>>     Write-Output "Firewall Rule 'OpenSSH-Server-In-TCP' does not exist, creating it..."
>>     New-NetFirewallRule -Name 'OpenSSH-Server-In-TCP' -DisplayName 'OpenSSH Server (sshd)'
-Enabled True -Direction Inbound -Protocol TCP -Action Allow -LocalPort 22
>> } else {
>>     Write-Output "Firewall rule 'OpenSSH-Server-In-TCP' has been created and exists."
>> }
Firewall rule 'OpenSSH-Server-In-TCP' has been created and exists.
PS C:\Windows\system32>
```