Memoria del Trabajo de Fundamentos de Redes

Sergio Quijano Rey, Daniel Gonzálvez

Motivación

La motivación que hay detrás de este trabajo es la de estudiar cómo funcionan los productos de almacenamiento en la nube, a través de la creación de un servidor bastante básico de almacenamiento en red sobre nuestro propio servidor.

Para ello hemos usado una *Raspberry Pi 3* en la que hemos instalado el programa de software libre *Nextcloud*. Sobre este servidor muy básico hemos tenido que realizar ciertas labores de administración, más propias de una asignatura como *Ingeniería de Servidores*, y otras labores centradas en aspectos de redes.

Esto nos ha permitido conocer y aprender hasta cierto punto los aspectos a tener en cuenta para crear infraestructuras de este tipo, centrándonos especialmente en aspectos específicos de las redes.

Especificación del hardware usado

El hardware que hemos usado es el siguiente:

- Raspberry Pi 3 Model B:
  - CPU Quad Core 1,2GHz Broadcom BCM2837 64bit
  - 1GB RAM
  - Tarjeta SD *Samsumg Evo*, 64 GB (actúa como disco duro del ordenador)
- Cargador de móvil *MicroUSB*: dará corriente a la *Raspberry Pi 3*

Raspberry Pi 3 Model B

Instalación básica del servidor

Instalación del sistema operativo

- Ubuntu Server: sin interfaz gráfica para ahorrar recursos
- Descargamos el `.img` del siguiente link

- Vemos dónde está localizada la tarjeta SD con `lsblk`
- Quemamos la imagen con el comando: `sudo dd if="ubuntu-raspberry.img" of="/dev/sdb" bs=4M status=progress`
  - `if`: archivo de entrada
  - `of`: archivo de salida
  - `bs`: tamaño del bloque
  - `status=progress`: para ver la barra de progreso
- Se podría usar una herramienta con `GUI` como `etcher`

Configuración inicial del sistema

Montaje de la Raspberry Pi

- Insertamos la tarjeta SD en la *Raspberry Pi*
- Conectamos el cable ethernet RJ45 entre el router de nuestra casa y la *Raspberry Pi*
- Conectamos un cargador de móvil *micro-usb* a la toma de la *Raspberry Pi*

Primera conexión

- Gracias a que estamos usando *Ubuntu Server*, tenemos `ssh` habilitado por defecto
- Abrimos el administrador del router en nuestro navegador (`192.168.1.1`) para localizar la ip local de la raspberry
- Hacemos ssh: `ssh ubuntu@192.168.1.8`

Localizando la raspberry en el administrador del router

- El usuario `ubuntu` está en la lista de `sudoers`
- El usuario `root` no tiene contraseña. Solo se puede alcanzar:
  1. Accediendo a `ubuntu`
  2. `sudo su -`
- Lo primero que vamos a hacer es actualizar el sistema:
  - `sudo apt update; sudo apt upgrade`
  - Instalamos paquetes básicos como `vim` o `make` para empezar a trabajar

Creación del usuario de administración

Crearemos un usuario `administrator` para las labores de administración del servidor. Para ello ejecutamos `useradd -m administrator`. La opción `-m` crea el directorio `home` del usuario según lo especificado por el `skel`.

Añadimos al administrador al grupo `sudo` con el comando `usermod -aG sudo administrator`. La opción `-a` indica que estamos añadiendo al grupo, en vez de cambiar al usuario de grupo, y la opción `-G` es necesario para el anterior *flag*, indicando los grupos suplementarios a los que se va a añadir.

Para cambiar la contraseña al usuario, usamos el comando `passwd administrator`, que nos pide por el *prompt* introducir la antigua contraseña y, a continuación, la nueva contraseña.

Ahora borramos al usuario que venía por defecto con el comando `userdel -r ubuntu`, con la opción `-r` para que borre su `home`. Es muy importante que antes de hacer esto nos aseguremos que el usuario `administrator` puede hacer sudo. En otro caso, al darse que `root` no tiene contraseña, perderíamos todo modo de hacer `sudo`, pues no podemos cambiar de usuario desde `administrator` a `root` ni logearnos directamente a `root`

Aspectos de Redes en la instalación del servidor

Protección básica del servidor y SSH

En lo que sigue, vamos a dar conexión a nuestro servidor para que sea alcanzable desde fuera de la red local. Por ello, hay que dar una seguridad básica para evitar ciertos ataques que comprometan nuestro servidor. Ahora mismo, el mayor problema que tenemos es que se pueden hacer ataques de fuerza bruta en `ssh` para logearse, probando todas las posibles contraseñas. Y esto es aún más peligroso teniendo en cuenta de que se puede hacer directamente sobre el usuario `root`. Aunque ahora esto último no se puede hacer, porque `root` no tiene contraseña, si en algún momento se le asigna una contraseña, esto sería posible. Por tanto, estos son los problemas más básicos (que ni remotamente son los únicos) que vamos a tratar de solventar.

Para ello vamos a usar `shh`. `ssh` puede usar claves de cifrado asimétrico. Para generar estas claves, desde nuestros ordenadores con los que accedemos al servidor, lanzamos el comando `ssh-keygen`, que genera los archivos `~/.ssh/id_rsa.pub` (clave pública) y `~/.ssh/id_rsa` (clave privada). La encriptación que se

Instalación de Nextcloud

Uso básico de Nextcloud

Análisis del sistema

Referencias