PRÁCTICAS
DE
ADMINISTRACIÓN
DE SISTEMAS OPERATIVOS Y
REDES DE COMPUTADORES

HITO II

INSTALACIÓN DE SERVICIOS BÁSICOS EN RED:
Acceso remoto a máquinas y servicios de recursos en red

Alejandro Reyes Albillar
45931406-S
ara65@alu.ua.es
Curso 2015 / 2016
Grado en Ingeniería Informática

Universitat d'Alacant
Universidad de Alicante

ÍNDICE

1. Licencias

Una licencia de software es un contrato entre el licenciante (autor/titular de los derechos de explotación/distribuidor) y el licenciatario (usuario consumidor /usuario profesional o empresa) del programa informático, para utilizar el software cumpliendo una serie de términos y condiciones establecidas dentro de sus cláusulas.

Las licencias de software pueden establecer entre otras cosas: la cesión de determinados derechos del propietario al usuario final sobre una o varias copias del programa informático, los límites en la responsabilidad por fallos, el plazo de cesión de los derechos, el ámbito geográfico de validez del contrato e incluso pueden establecer determinados compromisos del usuario final hacia el propietario, tales como la no cesión del programa a terceros o la no reinstalación del programa en equipos distintos al que se instaló originalmente.

Las más conocidas son la EULA (End-User License Agreement) de Microsoft y la licencia de GNU o GNU General Public License (o simplemente sus siglas del inglés GNU GPL) que es la licencia más ampliamente usada en el mundo del software y garantiza a los usuarios finales (personas, organizaciones, compañías) la libertad de usar, estudiar, compartir (copiar) y modificar el software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios. Esta licencia fue creada originalmente por Richard Stallman fundador de la Free Software Foundation (FSF) para el proyecto GNU.

2. Particionado de discos

Para hacer una partición en Windows basta con abrir el disk manager y crear las particiones con el asistente.

Sin embargo, para realizar particiones en Linux/Unix el proceso se complica un poco.

Para crear particiones usando la herramienta fdisk, primero debemos comprobar los discos del sistema disponibles y las particiones que ya haya creadas, para ello utilizamos el parámetro -l:

```
$ sudo fdisk -l

Disco /dev/sda: 160.0 GB, 160041885696 bytes
255 cabezas, 63 sectores/pista, 19457 cilindros
Unidades = cilindros de 16065 * 512 = 8225280 bytes
Identificador de disco: 0x000c3c51

Dispositivo Inicio    Comienzo   Fin     Bloques  Id  Sistema
/dev/sda2            3233        9855    53199247+  83  Linux
/dev/sda4            9856       19457    77128065   83  Linux
```

En la salida de fdisk hemos verificado que tenemos un único disco /dev/sda de 160 GB, sobre el cual ya hay dos particiones creadas con sistemas Linux, /dev/sda2 y /dev/sda4. Si nos fijamos bien, vemos que el disco tiene 19457 cilindros y que las particiones comienzan en el 3233, por lo que tenemos espacio libre para crear más si lo deseamos.

Vamos a crear entonces una partición de prueba que utilice el resto de espacio disponible en el disco, comenzamos ejecutando fdisk sobre el disco a utilizar:

```
$ sudo fdisk /dev/sda
```

Si pulsamos la m una vez dentro podremos visualizar las distintas opciones con su respectiva letra de ejecución:

```
Orden  Acción
   a   Conmuta el indicador de iniciable
   b   Modifica la etiqueta de disco bsd
   c   Conmuta el indicador de compatibilidad con DOS
   d   Suprime una partición
   l   Lista los tipos de particiones conocidos
   m   Imprime este menú
   n   Añade una nueva partición
   o   Crea una nueva tabla de particiones DOS vacía
   p   Imprime la tabla de particiones
   q   Sale sin guardar los cambios
   s   Crea una nueva etiqueta de disco Sun
   t   Cambia el identificador de sistema de una partición
   u   Cambia las unidades de visualización/entrada
   v   Verifica la tabla de particiones
   w   Escribe la tabla en el disco y sale
   x   Funciones adicionales (sólo para usuarios avanzados)
```

Vamos a crear una nueva partición, así que pulsamos “n”:

```
Orden (m para obtener ayuda): n
Acción de la orden
   e   Partición extendida
   p   Partición primaria (1-4)
```

Seleccionamos si queremos una partición extendida o primaria, en este caso podemos crearla como primaria, pulsamos “p” y dejamos que automáticamente se configure el número de la partición (se puede especificar, del 1 al 4). Después podemos seleccionar el primer y último cilindro a utilizar para la partición. Como yo voy a usar el resto de espacio disponible será del 1 al 19457 tal y como hemos visto antes, también podríamos indicar el tamaño de la partición en K, M o G. Lo dejamos por defecto en este caso:

```
p
```

```
Número de partición (1-4): 3
Primer cilindro (1-19457, valor predeterminado 1):
Último cilindro, +cilindros o +tamaño{K,M,G} (1-3232, valor predeterminado
3232):
Se está utilizando el valor predeterminado 3232
```

Ahora escribimos los cambios y salimos de fdisk:

```
Orden (m para obtener ayuda): w
¡Se ha modificado la tabla de particiones!

Llamando a ioctl() para volver a leer la tabla de particiones.

El núcleo todavía usa la tabla antigua.
La nueva tabla se usará en el próximo reinicio.
Se están sincronizando los discos.
```

Finalmente debemos ejecutar el comando partprobe para indicar al kernel que vuelva a leer la tabla de particiones:

```
$ sudo partprobe
```

Y ya tenemos nuestra nueva partición creada, ahora únicamente faltaría asignar el sistema de ficheros deseado (ext3, ext4, ntfs, etc):

```
$ sudo fdisk -l

Disco /dev/sda: 160.0 GB, 160041885696 bytes
255 cabezas, 63 sectores/pista, 19457 cilindros
Unidades = cilindros de 16065 * 512 = 8225280 bytes
Identificador de disco: 0x000c3c51

Dispositivo Inicio    Comienzo      Fin      Bloques  Id  Sistema
/dev/sda2            3233        9855    53199247+  83  Linux
/dev/sda3               1        3232    25961008+  83  Linux
/dev/sda4            9856       19457    77128065   83  Linux
```

Formateamos la partición como ext4:

```
$ mkfs.ext4  /dev/sda3
mke2fs 1.41.4 (27-Jan-2009)
mkfs.ext4: Permiso denegado mientras se intentaba determinar el tamaño del
sistema de ficheros
alex@sistemas:~$ sudo mkfs.ext4  /dev/sda3
mke2fs 1.41.4 (27-Jan-2009)
Etiqueta del sistema de ficheros=
Tipo de SO: Linux
Tamaño del bloque=4096 (bitácora=2)
Tamaño del fragmento=4096 (bitácora=2)
1623840 nodos-i, 6490252 bloques
324512 bloques (5.00%) reservados para el superusuario
Primer bloque de datos=0
```

```
Número máximo de bloques del sistema de ficheros=0
199 bloque de grupos
32768 bloques por grupo, 32768 fragmentos por grupo
8160 nodos-i por grupo
Respaldo del superbloque guardado en los bloques:
       32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632,
2654208,
       4096000

Escribiendo las tablas de nodos-i: hecho
Creating journal (32768 blocks): hecho
Escribiendo superbloques y la información contable del sistema de ficheros:
hecho

Este sistema de ficheros se revisará automáticamente cada 27 montajes o
180 días, lo que suceda primero.  Utilice tune2fs -c o -i para cambiarlo.
```

Ahora podemos montar la partición y comenzar a usarla:

```
$ sudo mount //dev/sda3
```

Vemos que está disponible con el comando df:

```
$ df -h | grep /dev/sda3
/dev/sda3              25G  172M   23G   1% /test
```

Eliminar una partición usando fdisk es más sencillo que crearla, lo primero que haremos será desmontarla del sistema con el comando umount:

```
$ sudo umount /dev/sda3
```

Accedemos de nuevo a la gestión del disco con fdisk:

```
$ sudo fdisk /dev/sda
```

Una vez dentro, eliminamos la partición con la letra “d” y seguido el número de la partición, escribimos después los cambios con “w”:

```
Orden (m para obtener ayuda): d
Número de partición (1-4): 3

Orden (m para obtener ayuda): w
¡Se ha modificado la tabla de particiones!

Llamando a ioctl() para volver a leer la tabla de particiones.
```

Ejecutamos de nuevo partproble para hacer efectivos los cambios sin reiniciar:

```
$ sudo partprobe
```

Y la partición ha sido eliminada, un fdisk -l no devolverá la partición:

```
$ sudo fdisk -l | grep sda3
```

3. Arranque y parada de servicios

Una vez instalados los diferentes servicios deberemos ponerlos en marcha utilizando las órdenes específicas de los diferentes sistemas.
En el caso de sistemas basados en Unix o Linux utilizaremos la orden "systemctl" que permite realizar, según los parámetros pasados, diferentes acciones con un proceso.

- start <nombreproceso>: Inicia el proceso en el caso de que no estuviera iniciado.
- stop <nombreproceso>: Detiene el proceso en el caso de que estuviera iniciado.
- enable <nombreproceso>: Crea una orden para que el proceso se inicie junto con el sistema.
- disable <nombreproceso>: Modifica la orden para que el proceso no se inicie junto al sistema.
- status <nombreservicio>: Comprueba el estado del servicio, es decir si está activo o no y lo muestra por pantalla.

Para poder ejecutar la orden "systemctl" debemos tener permisos de root, por lo que antes de ejecutarla deberemos dar permisos al usuario con "su -" o, en su defecto, "sudo su".

Algunos sistemas no reconocen la orden "systemctl" y utilizan en su lugar la orden "service <nombreproceso> <start|stop|restart|status>". En el caso de FreeBSD se utiliza "service" en vez de "systemctl" como lo hace CentOS.

En el caso de Windows Server se utiliza la orden "net start <nombreservicio>" para iniciar el servicio, o "net stop <nombreservicio>" para detenerlo. Estas órdenes debemos ejecutarlas desde powershell. También podemos utilizar las órdenes de powershell "Start-Service <nombreservicio>" y "Stop-Service <nombreservicio>" para este cometido.

4. Administración remota de Equipos

4.1. SSH, SFTP y SCP

El servicio SSH nos permite conectarnos remotamente a un equipo cuya ip conozcamos, de modo que, tras autenticarnos contra ese equipo de forma remota, podamos acceder al ordenador y sus elementos desde cualquier lugar.

Para ello, antes de nada debemos instalar el servicio en nuestro sistema. En el caso de Linux y Unix utilizaremos OpenSSH, mientras que en Windows instalaremos el emulador Cygwin para instalar este servicio.

En CentOS utilizaremos los comandos “yum install openssh-server” y “yum install openssh-clients” es decir, instalaremos el servidor para permitir que se conecten a nuestro equipo y el cliente para poder conectarnos remotamente a las máquinas que deseemos.

Debemos abrir el puerto 22 para que se puedan conectar equipos desde el exterior.

En FreeBSD ssh viene instalado por defecto, por lo que no hace falta instalarlo.

Según como hayamos especificado a la hora de realizar la instalación de los sistemas operativos, estos servicios vendrán instalador ya por defecto en el sistema, por lo que

Si deseamos realizar este proceso en Windows la cosa se complica, ya que no disponemos de las órdenes de instalación ni de un entorno capaz de realizar correctamente las conexiones, es por esto que debemos instalar un emulador como Cygwin e incorporarle ssh a la hora de instalarlo. En caso de realizar la instalación completa del emulador, ssh ya vendrá instalado, junto con un conjunto completo de elementos. Debemos asegurarnos de haber configurado correctamente Cygwin y haber abierto el puerto 22 en el firewall de windows para poder utilizar correctamente el servicio de ssh.

En todos los sistemas debemos asegurar que ningún usuario pueda conectarse en ningún momento como root a un equipo. Por tanto, deberemos añadir al archivo sshd_config situado en la carpeta /etc/ssh, en caso de Linux/Unix, o /etc en caso de Cygwin en Windows, la siguiente línea:

- AllowUsers <user1> <user2> …

NUNCA se debe permitir el acceso como root a un equipo, ya que compromete la seguridad del sistema de una manera impresionante. Los nombres de usuario permitidos deben tener la misma sintaxis que la que se muesta en el archivo /etc/passwd.

Además de añadir esta línea debemod descomentar las líneas:
- #PasswordAuthentication no
- #PermitEmptyPasswords no

Podemos descomentar las líneas eliminando el “#” de ellas. Además el atributo “PasswordAuthentication” debe ir a yes, ya que por defecto suele ir a no en algunos sistemas.

Finalmente y para hacer más cómo el acceso a los equipos, podemos realizar conexión mediante clave pública y privada.
Esto lo hacemos copiando una clave pública que generamos en nuestro equipo en el equipo al que queremos conectarnos, de modo que, al reconocernos, permita automáticamente la conexión.
Esta operación no se puede hacer para conectarse a Windows Server, ya que el operativo rechaza la conexión deliberadamente.

Para generar las claves pública y privada y enviarlas al equipo deseado, debemos seguir los siguientes pasos:

1. ssh-keygen -t rsa: Abre un asistente que generará la clave pública id_rsa.pub y la clave privada id_rsa que guardará en el directorio /home/root/.ssd . En este caso se ha escogido codificación RSA debido a que algunos sistemas no reconocen bien la codificación DSA.
2. cd /home/root/.ssh: Accedemos al directorio donde se han guardado las claves pública y privada.
3. ssh-copy-id -i id_rsa.pub user@ipOrdenadorDestino: Esta orden enviará al archivo /home/user/.ssh/authorized_keys la información necesaria para que el equipo se pueda conectar sin pedir la contraseña cada vez que se desee establecer la comunicación.

Una vez realizadas estas acciones, podremos realizar ssh a user@ipOrdenadorDestino sin tener que identificarnos con la contraseña cada vez que deseemos acceder a él.

Junto con la instalación de ssh nos vienen dos servicios, uno de acceso a ficheros llamado sftp y otro que nos permite copiar archivos entre equipos de forma segura llamado scp.

Para utilizar sftp únicamente necesitaremos realizar la conexión como si de ssh se tratara, simplemente debemos escribir sftp user@ipOrdenadorDestino y ya estaremos conectados al servicio SecureFTP para transferir archivos entre equipos de forma sencilla.

Para utilizar scp, utilizaremos la siguiente orden “scp <-P <puerto>> <ruta de archivo local> user@ipOrdenadorDestino:<ruta destino archivo>”.

Podemos enjaular el servidor sftp modificando el archivo sshd_conf.
Para ello debemos comentar la línea “Subsystem sftp /usr/libexec/openssh/sftp-server” y escribir las siguientes líneas:

```
Subsystem     sftp   internal-sftp
Match Group sftpusers
    ChrootDirectory %h
    ForceCommand internal-sftp
```

```
AllowTcpForwarding no
```

Donde sftp_users es un grupo que hemos creado.
A continuación creamos un usuario “sftpuser” que pertenezca a dicho grupo y lo configuramos para que tenga los permisos justos.

- usermod -G sftpusers sftpuser
- chown root:root /home/sftpuser # El grupo en FreeBSD es root:wheel
- chmod 755 /home/sftpuser
- mkdir -m 0755 /home/sftpuser/public_html
- chown sftpuser:users /home/sftpuser/public_html # El grupo en FreeBSD es sftpuser:wheel
- usermod -s /sbin/nologin sftpuser

De este modo, el usuario queda enjaulado de modo que únicamente puede conectarse por sftp al servidor.

4.2. VNC

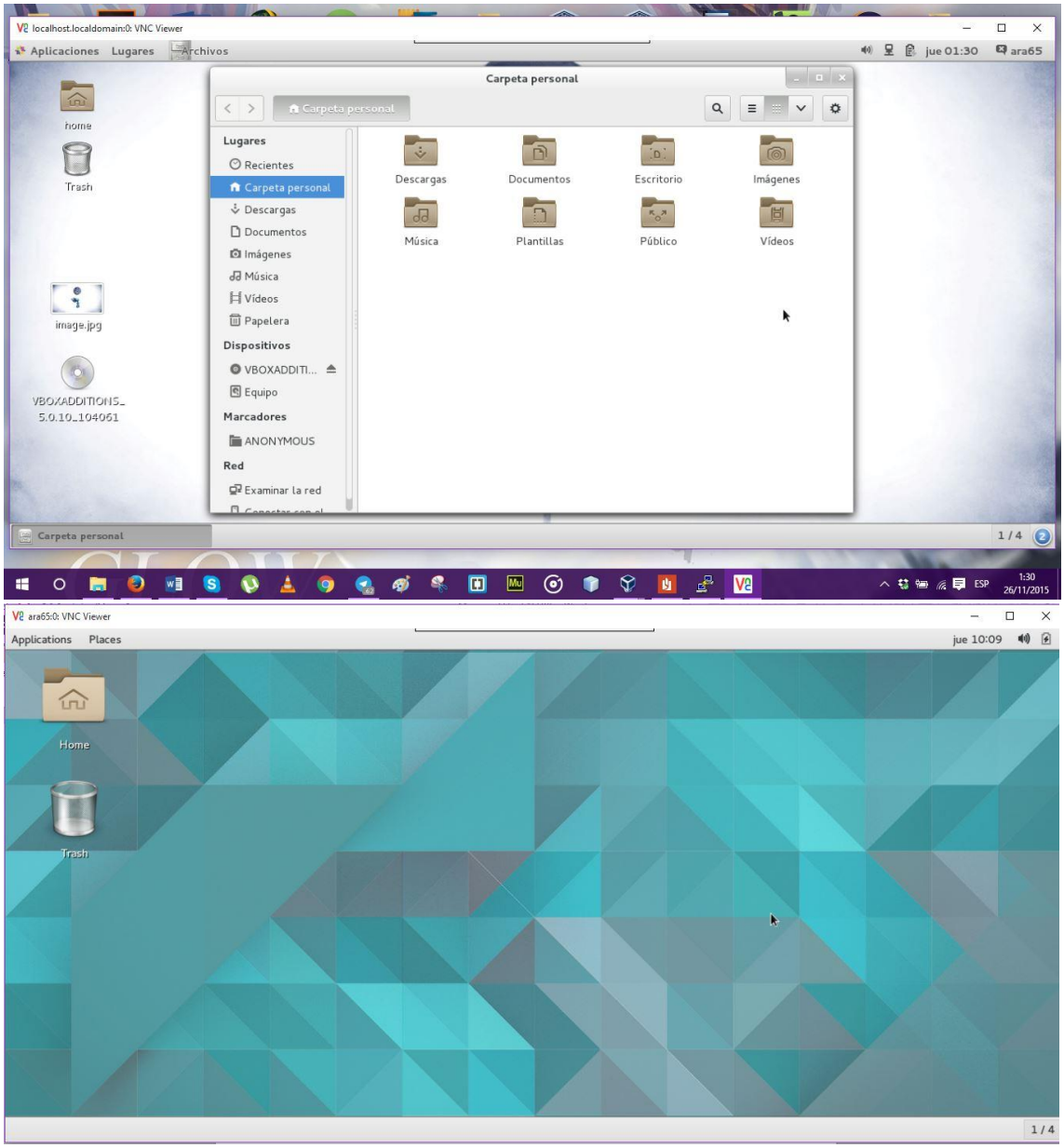
VNC, o Virtual Network Computing, es un programa de software libre basado en una estructura cliente-servidor el cual permite tomar el control del ordenador servidor remotamente a través de un ordenador cliente. Se conoce más comúnmente como escritorio remoto.

Tanto como para CentOS como para FreeBSD instalaremos x11VNC. Para ello utilizaremos las órdenes “yum install x11vnc” y “pkg install x11vnc” respectivamente. Una vez instalado x11vnc, deberemos crear una contraseña de acceso con la orden “x11vnc -storepasswd”, el cual creará un archivo con la contraseña en “/home/user/.vnc/passwd”. Ya simplemente debemos iniciar el servicio con x11vnc con la orden “x11vnc -bg -usepw -forever -rfbport <númeroPuerto>”. La opción -bg indica que el servicio se correrá ensegundo plano, la opción -usepw indica que se utilizará el archivo de contraseña creado antes, -forever se utiliza para indicar que siga aceptando conexiones después de que la primera se cierre y -rfbport se utiliza para indicar el puerto que se desea abrir para la conexión. En caso de no especificarse un puerto se abrirá por defecto el puerto 5900.

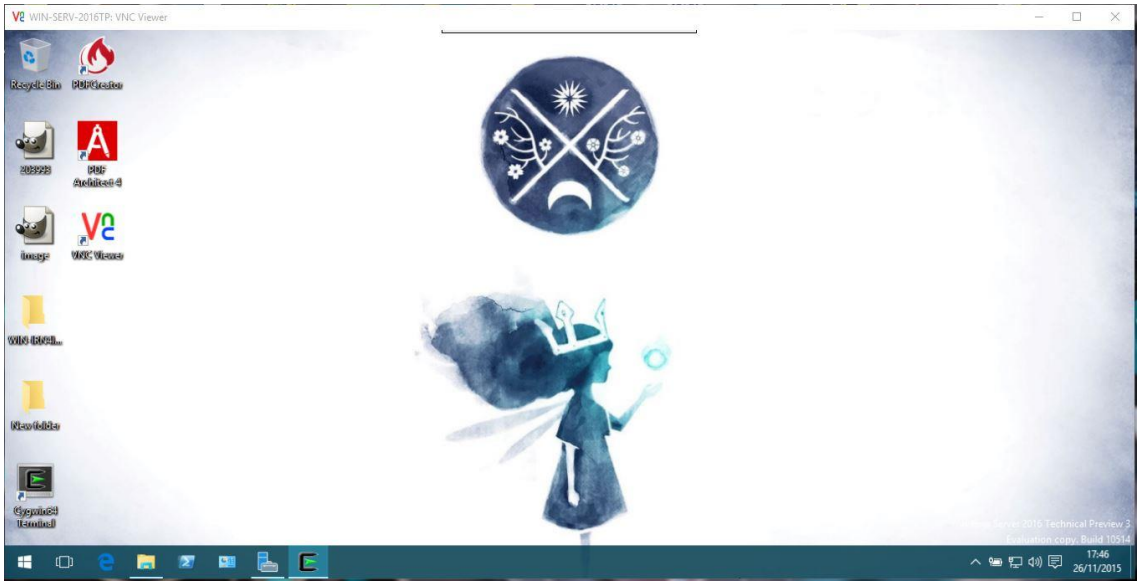
Debemos asegurarnos de que los puertos están abiertos y, en caso de no estarlo, abrirlos bien utilizando la herramienta “system-config-firewall” o añadiendo los puertos con las órdenes:

- iptables -A INPUT -m state --state NEW -m tcp -p tcp --dport 5900:5903 -j ACCEPT
- service iptables save
- service iptables restart

A continuación mostramos unas capturas de cómo se ven los sistemas desde el cliente RealVNC.



Para instalar Vnc en Windows utilizaremos el servidor RealVNC, que nos permite 30 dias de prueba gratuita de servidor vnc.



4.3. Terminal Services

Para instalar Terminal services, un asistente de escritorio remoto para Windows nos vamos a:
Server Manager > Manage > Add Roles and Features
Sin tocar nada le damos a siguiente hasta llegar a Server Roles, donde buscaremos el rol que dice Remote Desktop Services, lo marcaremos y le damos a siguiente.
Continuamos hasta llegar a donde dice “Role Services” y marcamos la opción que dice “Remote Desktop Session Host”, le damos a continuar y a instalar.

Una vez se haya instalado el rol, podremos acceder mediante el programa de conexión a escritorio remoto que nos proporciona Windows, así como cualquier otro preparado para este fin.

4.4. freenx

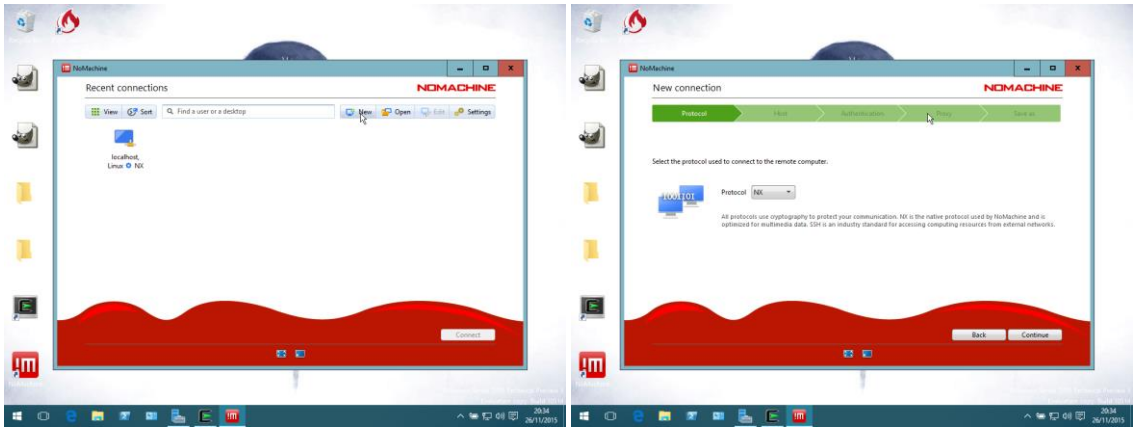
FreeNX es un software basado en el software NOMACHINE, desarrollado por una empresa italiana con el mismo nombre y cuyo objetivo es realizar una conexion remota a un escritorio utilizando el mínimo de recursos posibles.

En FreeBSD únicamente podemos instalar el cliente de x2go, otro software libre de características similares, por lo que no podremos conectarnos a él remotamente con la aplicación.

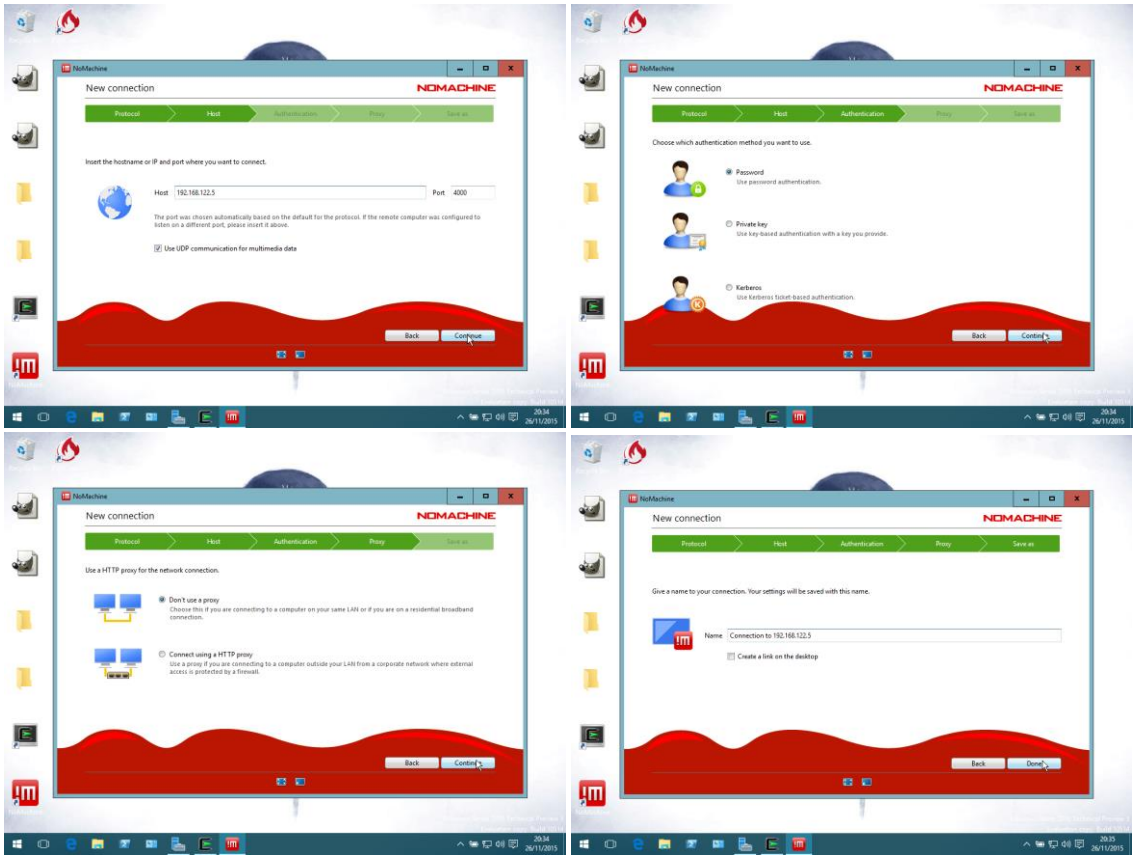
Para instalar NoMachine en CentOS debemos descargarnos el rpm desde su página oficial e instalarlo con la orden “rpm -i nomachine_5.0.47_1_x86_64.rpm”.
Tras la instalación instalaremos el cliente de NoMachine en Windows y nos conectaremos remotamente a él.

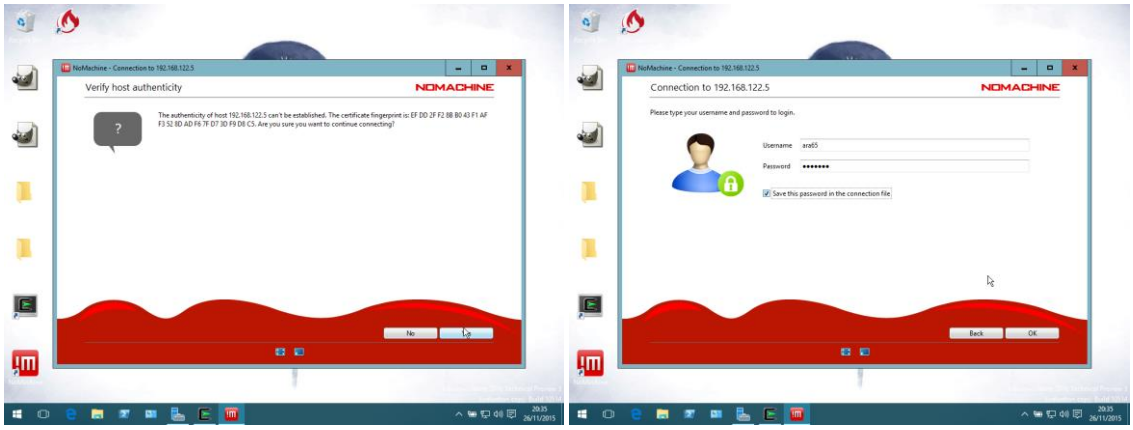
El instalador de NoMachine provee tanto de cliente como de servidor por lo que tras su instalación en Windows podemos utilizarlo como servidor para nuestro escritorio.

Abriremos nomachine y le daremos a “New…” para añadir una nueva máquina e indicamos que el protocolo a utilizar es NX.

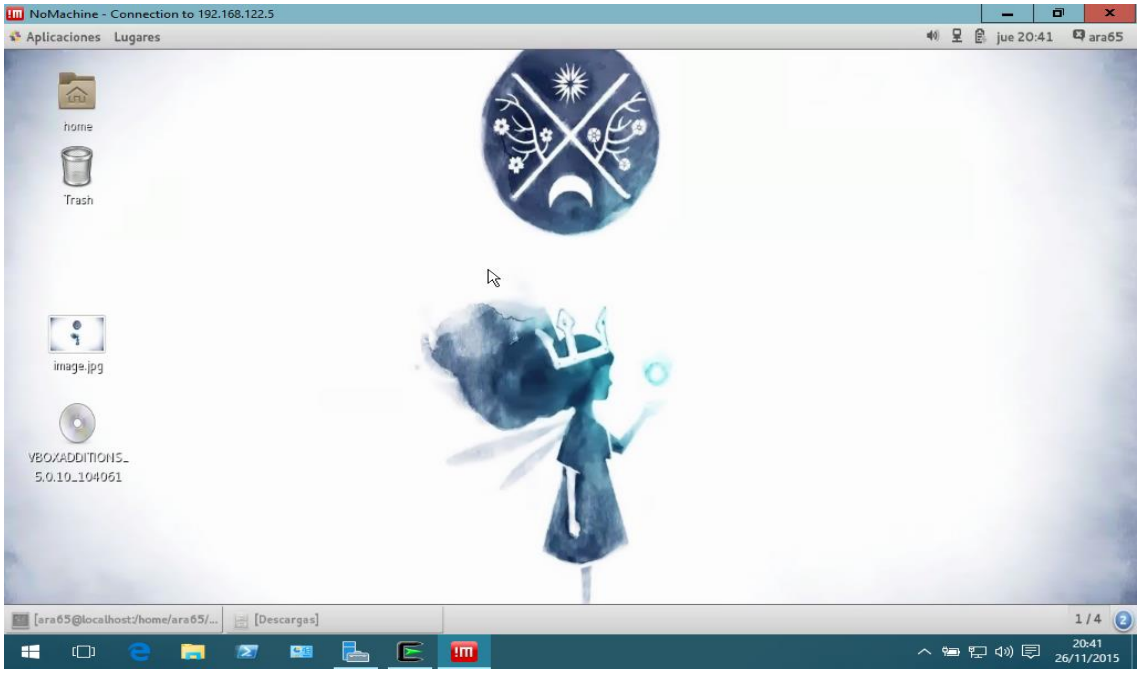


Entonces especificamos la ip del equipo al que queremos conectarnos, decimos que queremos autenticarnos por contraseña, que no queremos utilizar proxy y le damos un nombre a la conexión.





Por último especificamos el usuario y contraseña con el que nos queremos identificar en el equipo remoto y nos conectamos.



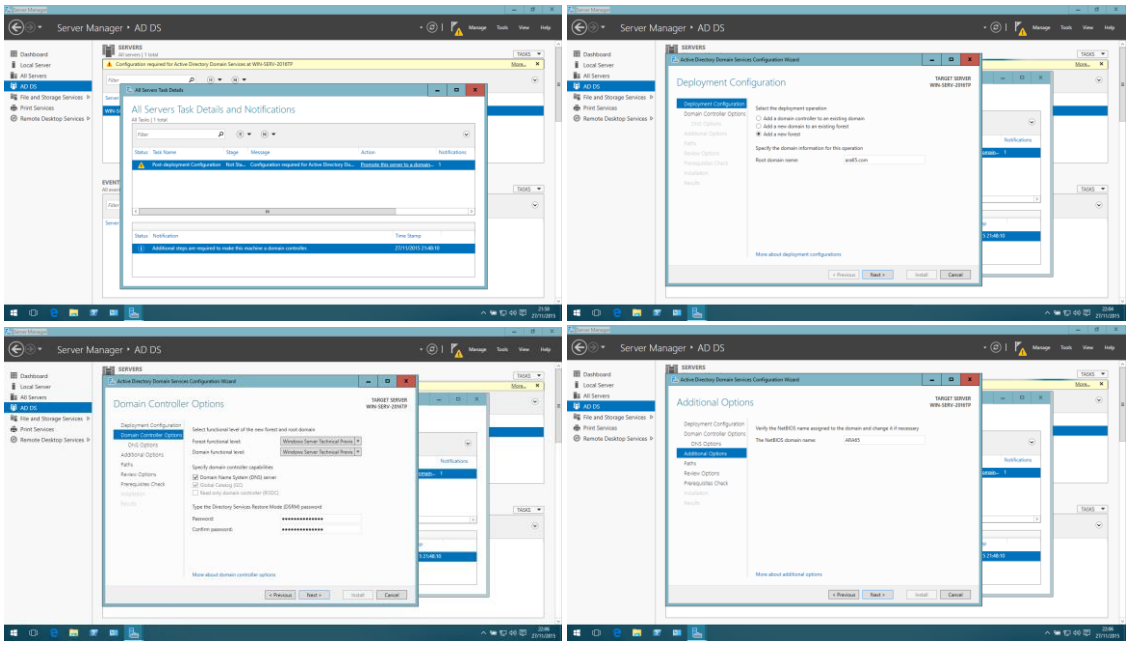
5. Servidor de directorio

5.1. Active Directory

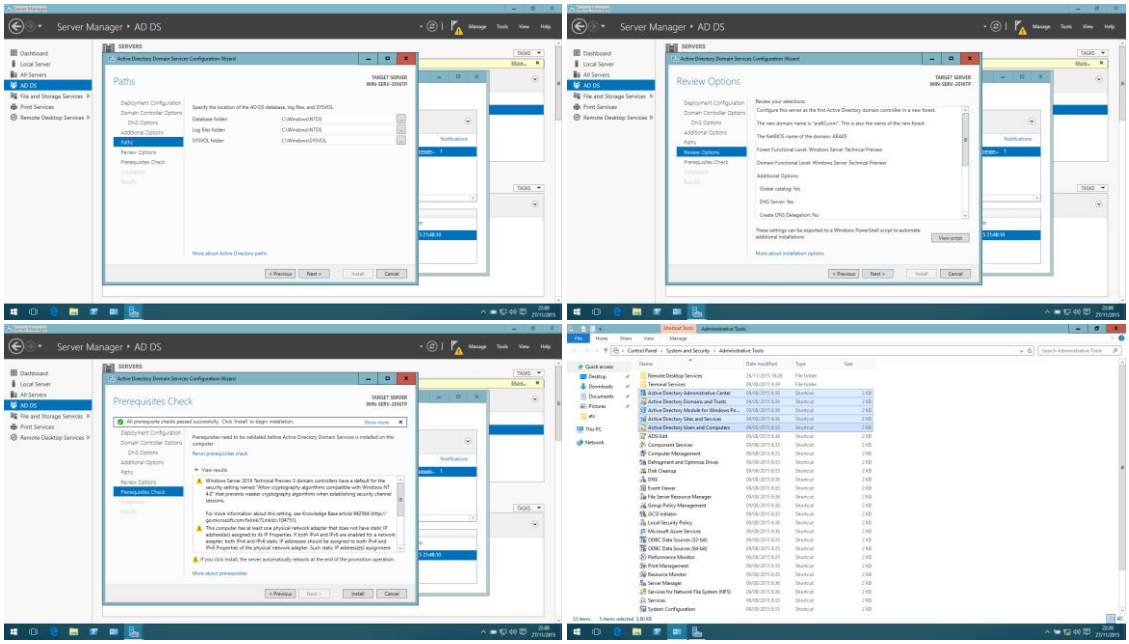
Para instalar Active Directory debemos ir al Server Manager y añadir su rol. Para ello, cuando estamos en la pestaña de selección de roles, seleccionamos la que dice “Active Directory Domain Services” y le damos a continuar y aceptar hasta que termine la instalación.

A continuación nos dirigimos a la pestaña AD DS y veremos una notificación que nos dice que hace falta configurar Active Directory.

Le damos a more para abrir la ventana que nos muestra el error, donde le daremos a “Promote this server to a domain controller”.



Esto nos abre el asistente para configurar active directory.
En el asistente seleccionamos “Add a new forest”, le asignamos un nombre al servidor y una contraseña.



Continuamos dándole a siguiente hasta instalarlo y después de reiniciar, actualizamos la contraseña del usuario y abrimos las herramientas administrativas.

5.2. OpenLDAP

Para instalar LDAP deberemos instalar los paquetes:

- openldap-clients-2.x
- openldap-servers-2.x
- authconfig
- authconfig-gtk (opcional)
- migrationtools

Los cuales podemos instalar con la orden “yum -y install openldap openldap-clients openldap-servers \
nss-pam-ldapd authconfig authconfig-gtk \
migrationtools”.

Una vez instalado todo, comenzaremos con la creación de certificados:

- Creamos el directorio /etc/openldap/cacerts con “mkdir -p /etc/openldap/cacerts”
- Nos movemos al directorio /etc/openldap/cacerts
- OpenLDAP requiere primero generar una nueva autoridad certificadora por lo que ejecutamos lo siguiente.
    - `echo "01" > ca.srl`
    - `openssl genrsa -aes128 2048 > cacert.key`
    - `openssl req -utf8 -new -key cacert.key -out cacert.csr`
    - `openssl x509 -req -in cacert.csr -out cacert.pem \`
    - `    -signkey cacert.key -days 3650`

De este juego de archivos se debe compartir cacert.pem con todos los clientes LDAP que se conectarán al servidor. Por tanto, debemos copiar este archivo dentro del directorio raíz del servidor HTTP o FTP y configure los permisos de acceso para que sea accesible desde estos servicios.

```
cp cacert.pem /var/www/html/
chmod 644 /var/www/html/cacert.pem
```

A continuación generamos el certificado y firma digital para el servidor. Cabe señalar que la firma digital que será utilizada por OpenLDAP será una tipo RSA sin contraseña en formato PEM generada a partir de una firma digital con contraseña.

```
openssl genrsa -aes128 2048 > key.pem
openssl req -utf8 -new -key key.pem -out slapd.csr
openssl x509 -req -in slapd.csr -out cert.pem \
    -CA cacert.pem -CAkey cacert.key -days 3650
openssl rsa -in key.pem -out key.pem
```

Configuramos todos los permisos necesarios para que sólo root y el grupo ldap puedan hacer uso de los certificados y firma digital con la orden “cacertdir_rehash /etc/openldap/cacerts”

Generamos los enlaces necesarios para el directorio /etc/openldap/cacerts:

```
chown -R root:ldap /etc/openldap/cacerts
chmod -R u=rwX,g=rX,o= /etc/openldap/cacerts
```

Buscamos, en el archivo /etc/sysconfig/ldap, la línea “#SLAPD_LDAPS=no”, la descomentamos y la ponemos a YES.

Con fines de organización se creará un directorio específico para este directorio y se configurará con permisos de acceso exclusivamente al usuario y grupo ldap.

```
mkdir /var/lib/ldap/autenticar
chmod 700 /var/lib/ldap/autenticar
```

Se requiere copiar el archivo DB_CONFIG.example dentro del directorio /var/lib/ldap/autenticar/, como el archivo DB_CONFIG por lo que ejecutaremos la orden “cp /usr/share/openldap-servers/DB_CONFIG.example \
  /var/lib/ldap/autenticar/DB_CONFIG”.

Todo el contenido del directorio /var/lib/ldap/autenticar debe pertenecer al usuario y grupo ldap por lo que damos permisos con la orden “chown -R ldap:ldap /var/lib/ldap/autenticar”.

Para crear la clave de acceso que se asignará en LDAP para el usuario administrador del directorio, ejecutamos la orden “slappasswd”, la cual generará un criptograma que debemos copiar, ya que lo vamos a necesitar más adelante.

Se debe crear /etc/openldap/slapd.conf como archivo nuevo con las órdenes “touch /etc/openldap/slapd.conf” o “nano /etc/openldap/slapd.conf”. El archivo creado debería contener lo siguiente:

```
include         /etc/openldap/schema/corba.schema
include         /etc/openldap/schema/core.schema
include         /etc/openldap/schema/cosine.schema
include         /etc/openldap/schema/duaconf.schema
include         /etc/openldap/schema/dyngroup.schema
include         /etc/openldap/schema/inetorgperson.schema
include         /etc/openldap/schema/java.schema
include         /etc/openldap/schema/misc.schema
include         /etc/openldap/schema/nis.schema
include         /etc/openldap/schema/openldap.schema
include         /etc/openldap/schema/ppolicy.schema
include         /etc/openldap/schema/collective.schema
```

```
include         /etc/openldap/schema/pmi.schema

TLSCACertificateFile /etc/openldap/cacerts/cacert.pem
TLSCertificateFile /etc/openldap/cacerts/cert.pem
TLSCertificateKeyFile /etc/openldap/cacerts/key.pem

allow bind_v2
pidfile         /var/run/openldap/slapd.pid
argsfile        /var/run/openldap/slapd.args

database     bdb
suffix       "dc=dominio,dc=tld"
rootdn       "cn=Manager,dc=dominio,dc=tld"
rootpw          {SSHA}LnmZLFeE1/zebp7AyEF09NlGaT1d4ckz
directory    /var/lib/ldap/autenticar

# Indices a mantener para esta base de datos
index objectClass                       eq,pres
index ou,cn,mail,surname,givenname      eq,pres,sub
index uidNumber,gidNumber,loginShell    eq,pres
index uid,memberUid                     eq,pres,sub
index nisMapName,nisMapEntry            eq,pres,sub
```

Cambiamos los permisos /etc/openldap/slapd.conf para que tenga únicamente permisos de lectura y escritura, sólo para el usuario ldap.

```
chown ldap:ldap /etc/openldap/slapd.conf
chmod 600 /etc/openldap/slapd.conf
```

Eliminamos el conjunto de archivos y directorios que componen la configuración predeterminada con la orden “rm -rf /etc/openldap/slapd.d/*”.

Es necesario crear los archivos base para el contenido del directorio /var/lib/ldap/autenticar, por lo que ejecutaremos “echo "" | slapadd -f /etc/openldap/slapd.conf” para este fin.

Convertimos el archivo /etc/openldap/slapd.conf en el nuevo subconjunto de archivos ldif que irán dentro del directorio /etc/ldap/slapd.d con la orden:

```
slaptest -f \
    /etc/openldap/slapd.conf -F \
    /etc/openldap/slapd.d.
```

Todo el contenido de los directorios “/etc/ldap/slapd.d” y “/var/lib/ldap/autenticar” deben pertenecer al usuario y grupo ldap.

```
chown -R ldap:ldap \
    /etc/openldap/slapd.d \
    /var/lib/ldap/autenticar
```

Restablecemos los contextos para los directorios /etc/ldap/slapd.d y /var/lib/ldap/autenticar ejecutando lo siguiente:

```
restorecon -R \
    /etc/openldap/slapd.d \
    /var/lib/ldap/autenticar
```

Iniciamos el servicio slapd y lo configuramos para que se inicie con el sistema

Editamos el archivo /usr/share/migrationtools/migrate_common.ph y modificamos los valores de las variables $DEFAULT_MAIL_DOMAIN y $DEFAULT_BASE a fin de que queden del siguiente modo:

```
# Default DNS domain
$DEFAULT_MAIL_DOMAIN = "dominio.tld";

# Default base
$DEFAULT_BASE = "dc=dominio,dc=tld";
```

A continuación, hay que crear el objeto que a su vez contendrá el resto de los datos en el directorio, utilizando migrate_base.pl para generar el archivo base.ldif con la orden ”/usr/share/migrationtools/migrate_base.pl > base.ldif”.

Insertamos la información generada en el directorio utilizando:

```
ldapadd -x -W -D 'cn=Manager,dc=dominio,dc=tld' \
    -h 127.0.0.1 -f base.ldif
```

Una vez hecho lo anterior, se podrá comenzar a poblar el directorio con datos. Lo primero será importar los grupos y usuarios existentes en el sistema. Importamos los usuarios creando los archivos group.ldif y passwd.ldif, utilizando migrate_group.pl y migrate_passwd.pl de modo que :

```
      /usr/share/migrationtools/migrate_group.pl \
          /etc/group group.ldif
      /usr/share/migrationtools/migrate_passwd.pl \
          /etc/passwd passwd.ldif
```

Lo anterior creará los archivos group.ldif y passwd.ldif, los cuales incluirán la información de los grupos y cuentas en el sistema, incluyendo las claves de acceso. Los datos se podrán insertar en el directorio LDAP utilizando lo siguiente:

```
ldapadd -x -W -D 'cn=Manager,dc=dominio,dc=tld' \
    -h 127.0.0.1 -f group.ldif
ldapadd -x -W -D 'cn=Manager,dc=dominio,dc=tld' \
    -h 127.0.0.1 -f passwd.ldif
```

Comprobamos qué directorios disponibles existen en el servidor 127.0.0.1 con la orden:

```
      ldapsearch -h 127.0.0.1 -x -b '' -s base \
```

```
          '(objectclass=*)' namingContexts”.
```

En el caso de obtener una salida parecida a lo siguiente:

```
# extended LDIF
#
# LDAPv3
# base <> with scope base
# filter: (objectclass=*)
# requesting: namingContexts
#

#
dn:
namingContexts: dc=dominio,dc=tld

# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

Iniciaremos los servicios y los configuraremos para que se inicien con el sistema.

Para instalar LPAP en FreeBSD utilizamos la orden “pkg install openldap-server”.

Generamos la clave SSHA con la orden “slappasswdd -s SomePassword”.
Editamos el archivo /usr/local/etc/openldap/slapd.conf:

```
#Esto ya debería estar, en el caso de no estar, debemos añadirlo
include /usr/local/etc/openldap/schema/core.schema
include /usr/local/etc/openldap/schema/cosine.schema
include /usr/local/etc/openldap/schema/inetorgperson.schema
include /usr/local/etc/openldap/schema/nis.schema
include /usr/local/etc/openldap/schema/misc.schema

password-hash {SSHA}
allow bind_v2
```

```
#Añadir
access to dn.children=”ou=Staff, ou=People, dc=example, dc=com”
       by self write
       by * auth
       by dn.children=”ou=Staff, ou=People, dc=example, dc=com” write


database     bdb
suffix       “dc=ara65,dc=com”
rootdn       “cn=ara65,dc=ara65,dc=com”

passwd       secret
directory    /var/db/openldap-data
index        objectClass  eq
```

A continuación configuramos en el cliente el archivo /use/localetc/openldap/ldap.conf:

```
BASE dc=example, dc=com
URI  ldap://ldapserver01.example.com
suffix dc=example, dc=com
binddn  cn=TestReader, ou=Roles, dc=example, dc=com
#Usar la contraseña que se ha generado anteriormente
binpw {SSHA} DF$&%adf&3Sfd6/gHS
scope sub
timelimit 5
bind_timelimit 5
bind_policy soft
```

Añadimos al archivo /etc/rc.conf la línea “slapd_enable=”YES” ” e inicimos el servicio con “service slapd start”.

6. Gestión de usuarios

6.1. Local

Para la administración local de usuarios en Windows utilizamos las opciones existentes en panel de control que nos permiten editar el tipo de cuenta que posee cada usuario y su contraseña. Además nos permite crear y eliminar usuarios a gusto. En el caso de necesitar más usuarios simplemente abrimos panel de control, vamos a cuentas de usuario y cambiar tipo de cuenta, desde este panel podemos seleccionar el usuario a modificar o si deseamos crear o eliminar un usuario.

Para la administración de usuarios en Linux/Unix tenemos las órdenes adduser, que nos permite crear un usuario, addgroup, que nos permite crear grupos y usermod, que, dependiendo de los parámetros que se le pasen, puede cambiar diferentes aspectos del usuario como la localización de su directorio home, el hecho de si puede hacer login en el equipo o no, y otras características ya descritas en la práctica anterior.

6.2. NIS

Para instalar nis en CentOS utilizaremos la siguiente orden “yum -y install ypbind yp-tools ypserv”. Tras haber instalado lo necesario para que funcione debemos con figurar el archivo /etc/yp.conf al que añadiremos la línea “domain dominio.net server 192.168.122.5”.

También editaremos el fichero /etc/ypserv.conf, al que añadiremos las siguientes líneas en caso de que no existan ya:

```
dns: no
files: 30
xfr_check_port: yes
* : * : shadow.byname : port
* : * : passwd.adjunct.byname : port
```

Una vez terminada con la edición de ypserv.conf, editaremos el archivo /etc/sysconfig/network, añadiendo la siguiente línea “NISDOMAIN="dominio.net" “.

Ahora, para integrarse al dominio recién configurado ejecutamos las órdenes “domainname dominio.net” y “ypdomainname dominio.net”.

A continuación creamos el archivo /var/yp/securenets y le introducimos lo siguiente:

```
host 127.0.0.1
255.255.255.0 192.168.122.5
```

Hemos supuesto una red 192.168.122.5 con máscara de 24 bits para configurar lo anterior.

Una vez realizado todo lo anterior solo resta iniciar los servicios con “service portmap start”, “service ypserv start” y “service ypserv enable”.
Podemos comproar que el servicio está activo mediante la orden “rpcinfo -u localhost ypserv”.

Ahora creamos los mapas de NIS con la orden “/usr/lib/yp/ypinit -m” con el que se nos abre un asistente en el que vamos añadiendo hosts. Tras añadir al último host, pulsamos Ctrl+D para cerrar el asistente.

Tras esto, iniciamos los servicios ypbind, yppasswdd e ypxfrd y los configuramos para que se inicien con el sistema.

En el cliente deberemos instalar “ypbind” e “yp-tools” y configurar los archivos necesarios para que funcione correctamente.

Debemos editar el archivo /etc/sysconfig/network y añadir la línea “NISDOMAIN=internal”.

A continuación editaremos el archivo vi /etc/yp.conf considerando que el dominio a utilizar es dominio.net y que la dirección IP del servidor es 192.168.0.254.

- domain dominio.net server 192.168.0.54

Editamos el archivo /etc/hosts:

Debemos asegurarnos que esté definido un registro que asocie la dirección IP principal del sistema con el nombre de anfitrión del sistema. Considerando que la IP del servidor es 192.168.0.254 y que el nombre de anfitrión es servidor.dominio.net, deberá encontrar o añadir un registro similar al siguiente:

192.168.0.254servidor.dominio.net servidor

A continuación estableceremos el dominio NIS. Para ello utilizaremos las siguientes dos órdenes:

- domainname dominio.net
- ypdomainname dominio.net

Ajustes en los archivos /etc/nsswitch.conf, /etc/hosts.allow y /etc/hosts.deny.

Añadimos al archivo /etc/nsswitch.conf las siguientes líneas al final:

```
passwd: files nis
shadow: files nis
group: files nis
```

A fin de establecer una seguridad apropiada, es necesario denegar el acceso a todo en el archivo /etc/hosts.deny, por lo que añadiremos la línea “portmap:ALL”.

En el archivo /etc/hosts.allow, definiremos los anfitriones y redes que tendrán permitido acceder a los servicios configurados:

portmap:127.0.0.1
portmap:192.168.0.0/255.255.255.0

Iniciamos servicio ypbind y lo configuramos para que se inicie con el sistema.

Podemos asegurarnos de que todo funciona correctamente, utilizando el siguiente mandato que realizará una solicitud RPC para solicitar información del servicio ypbind:

rpcinfo -u localhost ypbind

El mandato anterior deberá regresar una salida similar a la siguiente. Si acaso regresa algo distinto o conexión rehusada, deben revisarse todos los procedimientos realizados hasta este punto.

```
el programa 100007 versión 1 está listo y a la espera
el programa 100007 versión 2 está listo y a la espera
```

Utilice el siguiente mandato para consultar todos los datos que están siendo distribuídos por el servicio ypserv del servidor NIS.

ypcat passwd

Para su instalación en Windows, abrimos Server Manager y añadimos el rol “Identity Management for UNIX”. Una vez completada la instalación el servicio estará instalado en el equipo.

6.3. LDAP

LDAP son las siglas de Lightweight Directory Access Protocol (en español Protocolo Ligero/Simplificado de Acceso a Directorios) que hacen referencia a un protocolo a nivel de aplicación que permite el acceso a un servicio de directorio ordenado y distribuido para buscar diversa información en un entorno de red. LDAP también se considera una base de datos (aunque su sistema de almacenamiento puede ser diferente) a la que pueden realizarse consultas.

Un directorio es un conjunto de objetos con atributos organizados en una manera lógica y jerárquica. El ejemplo más común es el directorio telefónico, que consiste en una serie de nombres (personas u organizaciones) que están ordenados alfabéticamente, con cada nombre teniendo una dirección y un número de teléfono adjuntos. Para entender mejor, es un libro o carpeta, en la cual se escriben nombres de personas, teléfonos y direcciones, y se ordena alfabéticamente.

Un árbol de directorio LDAP a veces refleja varios límites políticos, geográficos u organizacionales, dependiendo del modelo elegido. Los despliegues actuales de LDAP tienden a usar nombres de Sistema de Nombres de Dominio (DNS por sus siglas en inglés) para estructurar los niveles más altos de la jerarquía. Conforme se desciende en el directorio pueden aparecer entradas que representan personas, unidades organizacionales, impresoras, documentos, grupos de personas o cualquier cosa que representa una entrada dada en el árbol (o múltiples entradas).

Habitualmente, almacena la información de autenticación (usuario y contraseña) y es utilizado para autenticarse aunque es posible almacenar otra información (datos de contacto del usuario, ubicación de diversos recursos de la red, permisos, certificados, etc). A manera de síntesis, LDAP es un protocolo de acceso unificado a un conjunto de información sobre una red.

Debido a esto, este sistema, el cual hemos explicado antes como instalar, nos permitirá administrar remotamente los usuarios.

7. Servicio DNS

Para instalar el servicio DNS, debemos instalar antes la herramienta bind, así como bind-utils para poder configurar correctamente el servidor. Para ello utilizaremos la orden “yum -y install bind bind-chroot bind-utils” en CentOS y “pkg install bind bind-chroot bind-utils”.

Generaremos una firma digital para mejorar la seguridad. Para ello ejecutamos la orden “rndc-confgen -a -r /dev/urandom -b 512 -c /etc/rndc.key”.

Cambiamos los permisos del archivo con “chown root:named /etc/rndc.key” y “chmod 640 /etc/rndc.key” para que pertenezca al usuario “root”, al grupo “named” y posea permisos de lectura y escritura.

A continuación actualizamos el archivo caché con los servidores del DNS raíz, que podemos obtener con la orden

```
wget -N http://www.internic.net/domain/named.root \
    -O /var/named/named.ca.
```

Concederemos, al igual que al archivo rnfc.key, los permisos necesarios con las órdenes:

- chown root:named /var/named/named.ca
- chmod 640 /var/named/named.ca

Editamos el archivo /etc/named.conf partir de la configuración mínima que permitirá utilizar el servicio para todo tipo de uso. Dicha configuración básica es la siguiente:

```
options {
       directory "/var/named";
       dump-file "/var/named/data/cache_dump.db";
       statistics-file "/var/named/data/named_stats.txt";
       memstatistics-file "/var/named/data/named_mem_stats.txt";
       managed-keys-directory "/var/named/dynamic";
       version "BIND";
       forwarders {
              8.8.8.8;
              8.8.4.4;
       };
       forward first;
// Solo habilitar lo siguiente si se va a utilizar DNSSEC y si los
// servidores DNS del proveedor tienen soporte para DNSSEC.
       dnssec-enable yes;
       dnssec-validation yes;
       dnssec-lookaside auto;
       bindkeys-file "/etc/named.iscdlv.key";
};

include "/etc/rndc.key";
include "/etc/named.root.key";
```

```
controls {
       inet 127.0.0.1 allow { 127.0.0.1; } keys { "rndc-key"; };
};

logging {
       channel default_debug {
              file "data/named.run";
              severity dynamic;
       };
       category lame-servers { null; };
};

view "local" {
       match-clients {
              127.0.0.0/8;
              10.0.0.0/8;
              172.16.0.0/12;
              192.168.0.0/16;
       };
       recursion yes;
       include "/etc/named.rfc1912.zones";
       zone "." IN {
              type hint;
              file "named.ca";
       };
};

view "public" {
        match-clients { any; };
        recursion no;
        zone "." IN {
                type hint;
                file "named.ca";
        };
};
```

Lo anterior define como opciones que el directorio predeterminado será /var/named, se define un archivo donde se almacena la información del caché en /var/named/data/cache_dump.db; un archivo de estadísticas en /var/named/data/named_stats.txt, un archivo de estadísticas específicas en lo concerniente al uso de la memoria en /var/named/data/named_mem_stats.txt; consultas recursivas permitidas solamente a 127.0.0.1 y 192.168.1.0/24, se definen como ejemplos de servidores DNS para reenviar consultas a 8.8.8.8 y 8.8.4.4, que corresponden a los servidores DNS públicos de Google, los cuales puede reemplazar por los servidores DNS del proveedor de acceso a Internet utilizado); se define que la primera opción al realizar una consulta será reenviar a los DNS que se acaban de definir; se incluyen los archivos de configuración /etc/named.rfc1912.zones, que corresponde a las zonas del RFC 1912 y la firma digital única que se generó automáticamente tras instalar el paquete bind; Se define que los controles se realizan solamente desde 127.0.0.1, hacia 127.0.0.1, utilizando la firma digital única.

Conviene asegurarse que el archivo /etc/named.conf tenga los contextos correspondientes a fin de evitar potenciales problemas de seguridad.

```
chcon -u system_u -r object_r -t named_conf_t /etc/named.conf
```

Para añadir dominios, idealmente se deberían definir primero los siguientes datos:

1. Dominio a resolver.
2. Servidor de nombres principal (SOA). Éste debe ser un nombre que ya esté plenamente resuelto, y debe ser un FQDN (Fully Qualified Domain Name).
3. Lista de todos los servidores de nombres (NS) que se utilizarán para efectos de redundancia. Éstos deben ser nombres que ya estén plenamente resueltos y deben ser además FQDN (Fully Qualified Domain Name).
4. Cuenta de correo del administrador responsable de esta zona. Dicha cuenta debe existir y debe ser independiente de la misma zona que se está tratando de resolver.
5. Al menos un servidor de correo (MX), con un registro A, nunca CNAME.
6. IP predeterminada del dominio.
7. Sub-dominios dentro del dominio (www, mail, ftp, ns, etc.) y las direcciones IP que estarán asociadas a éstos.

Es importante tener bien en claro que los puntos 2, 3 y 4, involucran datos que deben existir previamente y estar plenamente resueltos por otro servidor DNS; Lo anterior quiere decir que jamás se deben utilizar datos que sean parte o dependan, del mismo dominio que se pretende resolver. De igual modo, el servidor donde se implementará el DNS deberá contar con un nombre FQDN y que esté previa y plenamente, resuelto en otro DNS.

Se debe crear una zona de reenvío por cada dominio sobre el cual se tenga autoridad plena y absoluta y se creará una zona de resolución inversa por cada red sobre la cual se tenga plena y absoluta autoridad. Es decir, si usted es el propietario del dominio «cualquiercosa.com», debe crear el archivo de zona correspondiente con el fin de resolver dicho dominio. Por cada red con direcciones IP privadas, sobre la cual se tenga control y absoluta autoridad, se debe crear un archivo de zona de resolución inversa a fin de resolver inversamente las direcciones IP de dicha zona.

Regularmente la resolución inversa de las direcciones IP públicas es responsabilidad de los proveedores de servicio ya que son éstos quienes tienen el control sobre éstas.

Todos los archivos de zona deben pertenecer al usuario «named» a fin de que el servicio named pueda acceder a éstos o bien modificar éstos en el caso de tratarse de zonas esclavas.

- Creación de los archivos de zona.

Los siguientes corresponderían a los contenidos para los archivos de zona requeridos para la red local y por el NIC con el que se haya registrado el dominio. Cabe señalar que en las zonas de reenvío siempre se especifica al menos un registro SOA y un registro NS. De manera opcional y en caso de que exista un servicio de correo electrónico, añada al menos un registro MX (Mail Exchanger o intercambiador de correo). Solamente necesitará sustituir nombres y direcciones IP y quizá añadir nuevos registros para complementar su red local.
Configuración mínima para /etc/named.conf en CentOS 5 y Red Hat™ Enterprise Linux 5.

La configuración mínima del archivo /chroot/etc/named.conf y que permitirá utilizar el servicio para todo tipo de uso, es la siguiente:

```
options {
       directory "/var/named";
       dump-file "/var/named/data/cache_dump.db";
       statistics-file "/var/named/data/named_stats.txt";
       memstatistics-file "/var/named/data/named_mem_stats.txt";
       managed-keys-directory "/var/named/dynamic";
       version "BIND";
       forwarders {
              8.8.8.8;
              8.8.4.4;
       };
       forward first;
// Solo habilitar lo siguiente si se va a utilizar DNSSEC y si los
// servidores DNS del proveedor tienen soporte para DNSSEC.
        dnssec-enable yes;
        dnssec-validation yes;
        dnssec-lookaside auto;
        bindkeys-file "/etc/named.iscdlv.key";
};

include "/etc/rndc.key";
include "/etc/named.root.key";

logging {
        channel default_debug {
                file "data/named.run";
                severity dynamic;
        };
        category lame-servers { null; };
};

controls {
       inet 127.0.0.1 allow { 127.0.0.1; } keys { "rndckey"; };
};

view "local" {
       match-clients {
              127.0.0.0/8;
              10.0.0.0/8;
              172.16.0.0/12;
```

```
            192.168.0.0/16;
      };
      recursion yes;
      include "/etc/named.rfc1912.zones";
};
```

Lo anterior define como opciones que el directorio predeterminado será /var/named, se define un archivo donde se almacena la información del caché en /var/named/data/cache_dump.db; un archivo de estadísticas en /var/named/data/named_stats.txt, un archivo de estadísticas específicas en lo concerniente al uso de la memoria en /var/named/data/named_mem_stats.txt; consultas recursivas permitidas solamente a 127.0.0.1 y 192.168.1.0/24; se definen como ejemplos de servidores DNS para reenviar consultas a 8.8.8.8 y 8.8.4.4, que corresponden a servidores DNS públicos de Google, los cuales puede reemplazar por los servidores DNS del proveedor de acceso a Internet utilizado; se define que la primera opción al realizar una consulta será reenviar a los DNS que se acaban de definir; se incluyen los archivos de configuración /etc/named.rfc1912.zones, que corresponde a las zonas del RFC 1912 y la firma digital única que se generó automáticamente tras instalar el paquete bind; Se define también que los controles se realizan solamente desde 127.0.0.1, hacia 127.0.0.1, utilizando la firma digital única.

Conviene asegurarse que el archivo /etc/named.conf tenga los contextos correspondientes para SELinux a fin de evitar potenciales problemas de seguridad.

```
chcon -u system_u -r object_r -t named_conf_t /etc/named.conf
```

Ejemplo de Zona de reenvío red local /var/named/data/red-local.zone.

```
$TTL 3600
@            IN    SOA   dns.red-local.     alguien.gmail.com. (
             2015090901; número de serie
             1800 ; tiempo de refresco
             900 ; tiempo entre reintentos de consulta
             604800 ; tiempo tras el cual expira la zona
             3600 ; tiempo total de vida
             )
@            IN    NS    dns.red-local.net.
@            IN    MX    10    mail
@            IN    TXT   "v=spf1 a mx -all"
@            IN    A     192.168.1.1
intranet     IN    A     192.168.1.1
maquina2     IN    A     192.168.1.2
maquina3     IN    A     192.168.1.3
maquina4     IN    A     192.168.1.4
www          IN    A     192.168.1.1
mail         IN    A     192.168.1.1
ftp          IN    CNAME intranet
dns          IN    CNAME intranet
```

Zona de resolución inversa red local /var/named/data/1.168.192.in-addr.arpa.zone

```
$TTL 3600
@            IN    SOA    dns.red-local.     alguien.gmail.com. (
             2015090901 ; número de serie
             1800 ; tiempo de refresco
             900 ; tiempo entre reintentos de consulta
             604800 ; tiempo tras el cual expira la zona
             3600 ; tiempo total de vida
             )
@            IN    NS     dns.red-local.
1      IN    PTR    intranet.red-local.
2      IN    PTR    maquina2.red-local.
3      IN    PTR    maquina3.red-local.
4      IN    PTR    maquina4.red-local.
```

Zona de reenvío del dominio /var/named/data/dominio.com.zone

Suponiendo que hipotéticamente se es la autoridad para el dominio «dominio.com», se puede crear una Zona de Reenvío con un contenido similar al siguiente:

```
$TTL 3600
@            IN    SOA    fqdn.dominio.tld.   alguien.gmail.com. (
             2015090901; número de serie
             1800 ; tiempo de refresco
             900 ; tiempo entre reintentos de consulta
             604800 ; tiempo tras el cual expira la zona
             3600 ; tiempo total de vida
             )
@            IN    NS     fqdn.dominio.tld.
@            IN    MX     10     mail
@            IN    TXT    "v=spf1 a mx -all"
@            IN    A      201.161.1.226
servidor     IN    A      201.161.1.226
www          IN    A      201.161.1.226
mail         IN    A      201.161.1.226
ftp          IN    CNAME  servidor
dns          IN    CNAME  servidor
```

Zona de resolución inversa del dominio /var/named/data/1.161.201.in-addr.arpa.zone

Suponiendo que hipotéticamente se es la autoridad para el segmento de red 201.161.1.0/24 (regularmente lo debe de hacer el proveedor de servicio de acceso hacia Internet), se puede crear una Zona de Resolución Inversa con un contenido similar al siguiente:

```
$TTL 3600
@            IN    SOA    fqdn.dominio.tld.   alguien.gmail.com. (
             2015090901 ; número de serie
             1800 ; tiempo de refresco
             900 ; tiempo entre reintentos de consulta
             604800 ; tiempo tras el cual expira la zona
```

```
            3600 ; tiempo total de vida
            )
@           IN    NS    fqdn.dominio.tld.
1     IN    PTR   servidor.dominio.com.
2     IN    PTR   maquina2.dominio.com.
3     IN    PTR   maquina3.dominio.com.
4     IN    PTR   maquina4.dominio.com.
```

Cada vez que haga algún cambio en algún archivo de zona, deberá cambiar el número de serie a fin de que tomen efecto los cambios de inmediato cuando se reinicie el servicio named, ya que de otro modo tendría que reiniciar el equipo, algo poco conveniente.

Las zonas de resolución inversa que involucran direcciones IP públicas son responsabilidad de los ISP (proveedores de servicio de acceso hacia Internet). Crear una zona de resolución inversa sin ser la autoridad de dicha zona tiene efecto sólo para quien use el servidor DNS recién configurado como único DNS.
Configuración de opciones del archivo /etc/named.conf

```
options {
      directory "/var/named";
      dump-file "/var/named/data/cache_dump.db";
      statistics-file "/var/named/data/named_stats.txt";
      memstatistics-file "/var/named/data/named_mem_stats.txt";
      managed-keys-directory "/var/named/dynamic";
      version "BIND";
      forwarders {
            8.8.8.8;
            8.8.4.4;
      };
      forward first;
// Solo habilitar lo siguiente si se va a utilizar DNSSEC y si los
// servidores DNS del proveedor tienen soporte para DNSSEC.
      dnssec-enable yes;
      dnssec-validation yes;
      dnssec-lookaside auto;
      bindkeys-file "/etc/named.iscdlv.key";
};

include "/etc/rndc.key";
include "/etc/named.root.key";

logging {
      channel default_debug {
            file "data/named.run";
            severity dynamic;
      };
      category lame-servers { null; };
};

controls {
      inet 127.0.0.1 allow { 127.0.0.1; } keys { "rndc-key"; };
};
```

```
view "local" {
	match-clients {
		127.0.0.0/8;
		10.0.0.0/8;
		172.16.0.0/12;
		192.168.0.0/16;
	};
	recursion yes;
	include "/etc/named.rfc1912.zones";
	zone "." IN {
		type hint;
		file "named.ca";
	};
	zone "red-local" {
		type master;
		file "data/red-local.zone";
		allow-update { none; };
	};
	zone "1.168.192.in-addr.arpa" {
		type master;
		file "data/1.168.192.in-addr.arpa.zone";
		allow-update { none; };
	};
};
```

8. DHCP

Primero y antes de nada, debemos instalar el servicio de DHCP en el equipo que corresponda, ya sea en CentOS con “yum -y install dhcp” o en FreeBSD con “pkg install dhcp”. Debemos recordar de abrir los puertos 67 y 68, necesarios para el uso de este servicio.
Una vez habilitado e iniciado el servicio “service dhcpd enable” y “service dhcpd start”, editaremos el archivo /etc/sysconfig/dhcpd, al cual añadiremos la línea “DHCPDARGS=enp0s1” para indicar que dicha interfaz es la correspondiente a la LAN.

A continuaciónañadimos las siguientes líneas al archivo /etc/dhcp/dhcp.conf:

```
# Si se tienen problemas con equipos con Windows Vista/7/8 omita la opción
# server-identifier. Ésto aunque rompe con el protocolo DHCP, permite a los
# clientes Windows Vista/7/8 poder comunicarse con el servidor DHCP y aceptar
# la dirección IP proporcionada.
# server-identifier 172.16.1.1;
ddns-update-style interim;
ignore client-updates;
authoritative;
default-lease-time 900;
max-lease-time 7200;
option ip-forwarding off;
option domain-name "ara65.CentOS.net";
```

```
option ntp-servers 0.pool.ntp.org, 1.pool.ntp.org, 2.pool.ntp.org,
3.pool.ntp.org;

shared-network ara65CentOS {
       #Dirección de red con mascara de 28(255.255.255.240)
       subnet 172.16.1.0 netmask 255.255.255.240 {
              option routers 172.16.1.1;
              option subnet-mask 255.255.255.240;
              option broadcast-address 172.16.1.15; #dirección difusión
              option domain-name-servers 172.16.1.1; #servidor DNS
              option netbios-name-servers 172.16.1.1;
              range 172.16.1.2 172.16.1.14; #Rango direcciones de la 2 a la 14
       }
}
#Asignación de IP’s estáticas
host printer{
       option host-name “centosprinter.ara65.CentOS.net”;
       hardware Ethernet 00:24:2B:65:54:84; #Se le asigna una mac
       fixed-address 172.16.1.13;
}
host pc1 {
       option host-name "pc1.ara65.CentOS.net";
       hardware ethernet 00:50:BF:27:1C:1C;
       fixed-address 172.16.1.14;
}

# Lista de direcciones MAC que tendrán permitido utilizar el servidor
# DHCP.
# deny unknown-clients impide que equipos fuera de esta lista puedan
# utilizar el servicio.
deny unknown-clients;
host printer {
       hardware ethernet 00:24:2B:65:54:84;
}
host pc1 {
       hardware ethernet 00:50:BF:27:1C:1C;
}
host pc2 {
       hardware ethernet F4:C7:14:70:FA:AC;
}
host laptop1 {
       hardware ethernet 44:87:FC:AA:DD:2D;
}
host laptop2 {
       hardware ethernet 70:F1:A1:9F:70:3B;
}
```

Para comprobar que funciona ejecutamos el comando “ifdown eth0
dhclient -d -I nombre-equipo -H nombre-equipo eth0” y nos debería mostrar la información para el interfaz en concreto.

9. Servidores de Archivos

9.1.NFS

Para configurar un sistema de archivos NFS debemos instalar su paquete con “yum -y install nfs-utils” o, si se prefiere una herramienta más gráfica “yum -y install system-config-nfs”.

Una vez instalado, editamos el archivo /etc/sysconfig/nfs y habilitamos las siguientes variables para que los puertos necesarios sean abiertos.

- RQUOTAD_PORT=875
- LOCKD_TCPPORT=32803 #Esta línea ya existe, se puede simplemente descomentar
- LOCKD_UDPPORT=32769 # Esta línea ya existe, se puede simplemente descomentar
- MOUNTD_PORT=892
- STATD_PORT=662

En el caso de que los servicios “rpcbind” y “nfs-lock” no estén activos, debemos iniciarlos.

Ahora iniciamos el servicio con “service nfs start” y leemos la configuración con “service nfs reload”.

Añadimos al archivo /etc/hosts.deny las siguientes líneas:

- portmap: ALL
- lockd: ALL
- mountd: ALL
- rquotad: ALL
- statd: ALL

Y al archivo /etc/hosts.allow les añadimos estas:

- portmap: 192.168.122.0/24
- lockd: 192.168.122.0/24
- mountd: 192.168.122.0/24
- rquotad: 192.168.122.0/24
- statd: 192.168.122.0/24

Abrimos los puertos 111, 2049, 32769, 662, 875 y 892 tanto en TCP como en UDP en el firewall para que funcione correctamente el servicio.

Por último modificamos, o creamos en el caso de que no exista, el fichero /etc/exports añadiendo el directorio o volumen que se dese compartir, en nuestro caso /datosSamba/nfs, con la sintaxis “<carpetaACompartir> ip(opciones)”.

Para, por ejemplo, compartir un directorio un equipo remoto añadiremos al archivo exports la línea “/datosSamba/nfs 192.168.122.0/24(rw,no_root_squash)”. En el cliente ejecutaremos la orden “-mount -o hard,intr,ro [ip]:/carpetaACompartir /rutadestino”, en este ejemplo sería “mount -o hard,intr,ro 192.168.122.4:/datosSamba/nfs /datosSamba/nfs”.

Para habilitar NFS en FreeBSD debemos modificar el archivo /etc/rc.conf y añadirle las siguientes líneas:

- portmap_enable="YES"
- nfs_server_enable="YES"
- mountd_flags="-r" #Se ejecuta automáticamente cuando se monta NFS
- mountd_enable="YES"

Para habilitar la configuración de cliente deberemos añadir también al archivo la línea:

- nfs_client_enable="YES"
- rpcbind_enable="YES"
- rpc_lockd_enable="YES"
- rpc_statd_enable="YES"

De igual manera que en CentOS, se modificará el archivo /etc/exports para indicar la carpeta o volumen que se desea compartir.

Una vez terminado iniciamos el servicio con "service nfsd start".

Para instalar este servicio en Windows simplemente debemos añadir el rol correspondiente en el server manager.
En la pestaña Server Roles seleccionamos:

- File and Storage Services > File and iSCSI Services > File Server Resource Manager y Server for NFS.

En la pestaña Features seleccionamos "Client for NFS", instalamos y reiniciamos si es necesario.

En reiniciar abrimos Server Manager y vamos a la pestaña donde dice File and Storage Services >Shares, y ahí haces doble click derecho "New Share".
sobre el recurso que deseas compartir, seleccionas NFS Share - Quick, le das un nombre al recurso compartido, confirmas la localización del servicio y el nombre que se utilizará para acceder a él y seleccionas el tipo de autenticación a utilizar.

Por último das los permisos necesarios para cada uno de los usuarios y terminamos con la configuración.

9.2. SAMBA/SMB

Para instalar Samba en CentOS utilizamos la orden "yum -y install samba samba-client samba-common". Para que el servicio funcione correctamente debemos acordarnos de abrir los puertos 135 - 139 en tcp y udp, además del 145 en tcp.

Una vez instalados los servicios y abiertos los puertos, debemos iniciar los procesos necesarios, por lo que ejecutaremos las opciones "service nmb start" y "service smb start". Para que se inicien con el sistema ejecutaremos las órdenes "chkconfig nmb on" y "chkconfig smb on".

Una vez realizado este proceso, daremos de alta a los diferentes usuarios, pero primero debemos darle una contraseña al usuario root con “smbpasswd -a root”.

Una vez realizado esto, crearemos el nuevo usuario con los siguientes comandos:

- useradd -s /sbin/nologin sambauser
- smbpasswd -a sambauser

Editamos el archivo de configuración /etc/samba/lmhosts donde añadiremos las ips y un nombre descriptivo de dicha ip de cada uno de los equipos que pueden conectarse a este servidor. Dicho archivo tendrá una sintaxis parecida a la mostrada a continuación:

```
127.0.0.1          localhost
192.168.122.5            servidorCentos
192.168.122.4            servidorFree
192.168.122.2            servidorWindows
```

Para finalizar editamos el archivo /etc/samba/smb.conf:

```
workgroup=MYGROUP
server string =Samba Server version %v
netbios name =MYSERVER
interfaces = lo eth0 enp0s3 enp0s8
hosts allow = 127. 192.168.122.
max protocol=SMB2
```

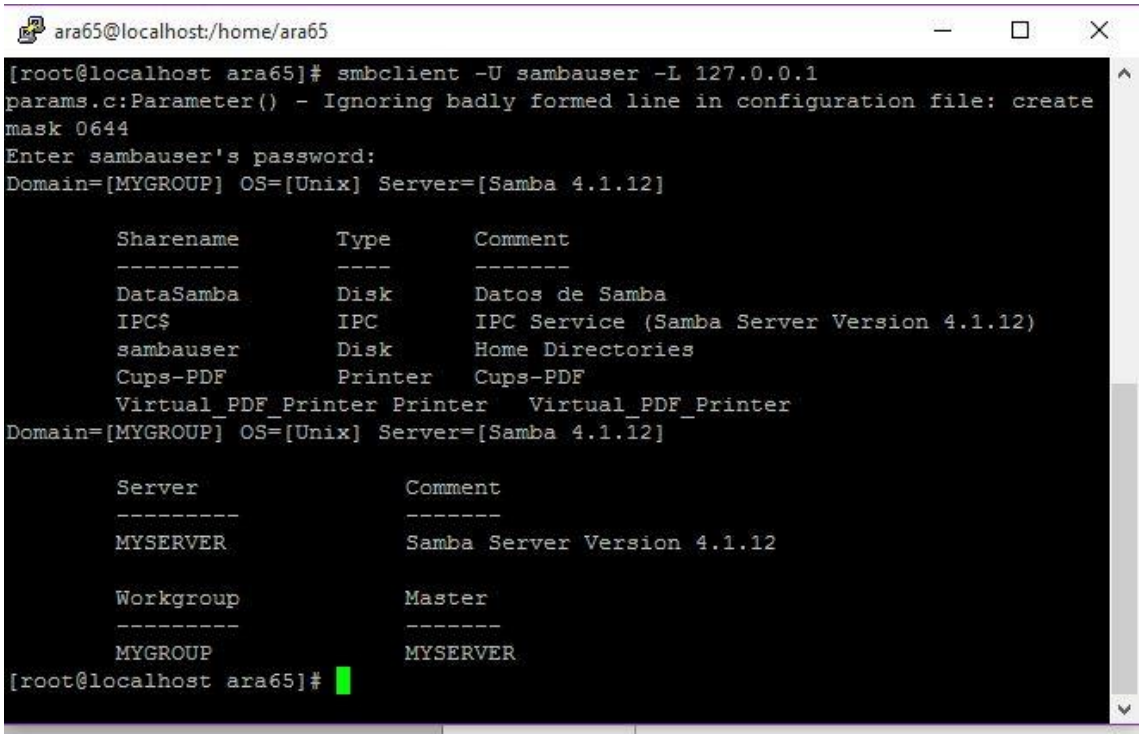
Reiniciamos los servicios smb y nmb.

Creamos un directorio para compartir por SAMBA y añadimos lo siguiente al final de /etc/samba/smb.conf

```
[DataSamba]
       comment= Datos de Samba
       path = /datosSamba
       guest ok = YES
       writable = YES
       directory mask = 0755
       create mask 0644
```

Finalmente reiniciamos el servicio smb.

Podemos probar el cliente con la orden “smbclient -U sambauser -L 127.0.0.1” y obtendremos una salida parecida a esta:



Para instalar SAMBA en FreeBSD utilizaremos la orden “pkg install samba36” y añadiremos al archivo /etc/rc.conf la línea “samba_enable=”YES” “.

Como no existe el archivo smb.conf pero sí que existe un archivo que contiene el simple, lo copiamos con la orden “cp /usr/local/etc/smb.conf.sample /usr/local/etc/smb.conf”.

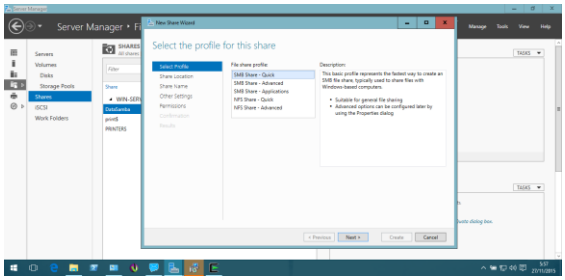
Descomentamos la línea “hosts allow” del archivo /usr/local/etc/smb.conf y añadimos al final del archivo las líneas para el nuevo directorio.

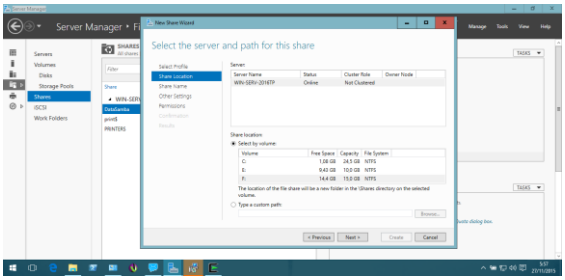
Debemos crear un usuario para samba y asignarle una contraseña con las órdenes:

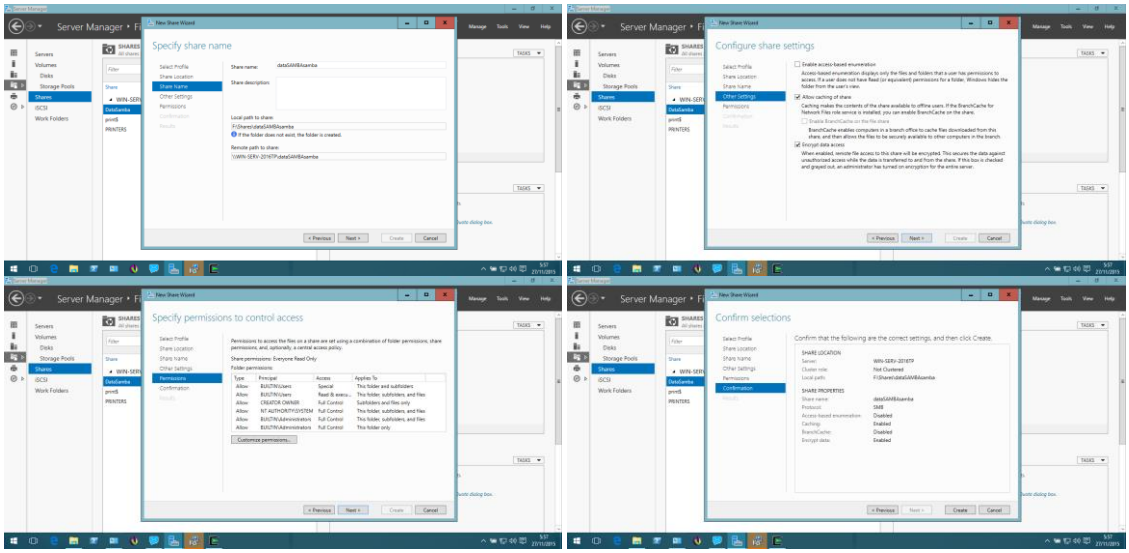
- pw groupadd smbprivate -M ara65
- smbpasswd -a ara65

Ya solo nos falta iniciar el servicio y probarlo como hemos hecho anteriormente.

En Windows, Samba se instala al mismo tiempo que NFS por lo que no es necesario hacer nada más aparte de compartir la carpeta deseada del mismo modo que NFS pero eligiendo SMB Quick en vez de NFS Quick.







10. Servidor de Impresión

Para el servicio de impresión hemos decidido utilizar impresoras a pdf en los diferentes servidores. En el caso de Unix/Linux instalaremos Cups con su complemento Cups-PDF, mientras que en Windows instalaremos Bullzip PDF printer.

Comenzaremos instalando las impresoras con “yum install cups cups-pdf” y “pkg install cups cups-pdf”. Esto instalará el servicio y un asistente web al que podemos acceder mediante la dirección “localhost:631” en nuestro navegador web.

Para poder acceder remotamente a esta interfaz y a las impresoras en cuestión debemos abrir el puerto 631 en los servidores Unix/Linux. Además de esto, en la interfaz web de CUPS, accediendo de manera local, debemos ir a la pestaña Administración y marcar las casillas donde dice:

- Compartir impresoras conectadas a este sistema
- Permitir la impresión desde Internet

Debemos recordar de darle al botón “Cambiar Configuración” para guardar los cambios. En algunas ocasiones nos pedirá un usuario y una contraseña, que serán el usuario y la contraseña del usuario root de nuestro servidor correspondiente.

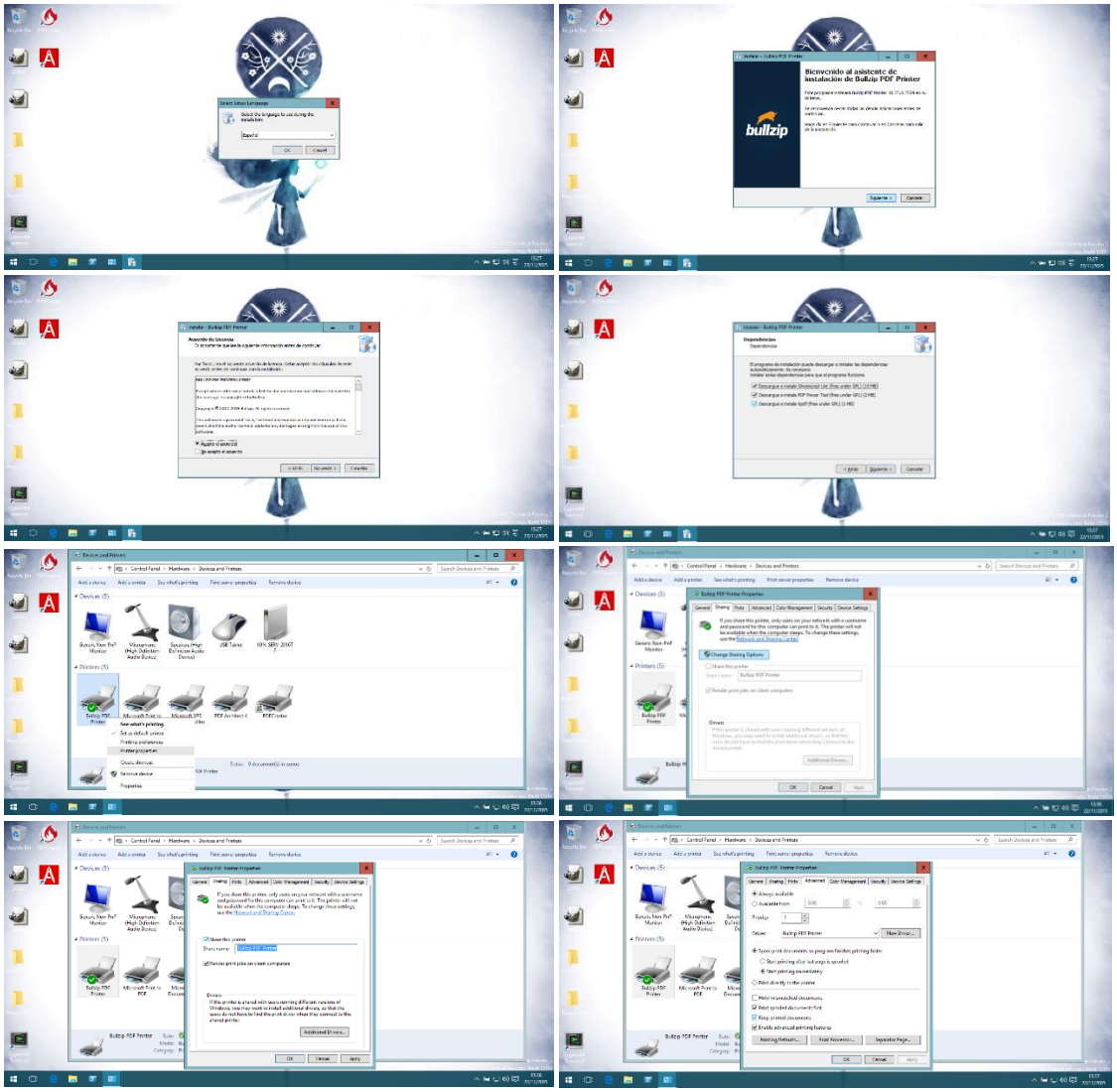
Para cambiar el directorio donde se guardan los archivos que imprimimos desde remoto debemos ir al archivo “cups-pdf.conf”, situado en “/usr/local/etc/cups” en FreeBSD y en “/etc/cups/” en CentOS y editar la línea donde pone “Out /var/spool/cups-psf/${USER}” por “Out <rutaDeGuardado>”.
En el caso de imprimirse desde un usuario anónimo, como es el caso de Windows, los archivos se guardarán en “/var/spool/cups-pdf/ANONYMOUS”, por lo que, para ver los archivos deberás navegar mediante tablero de comandos y entonces abrir los archivos mediante tablero de comandos con una herramienta visor de .pdf como evince.

En FreeBSD debemos añadir las siguientes líneas al archivo rc.conf para que el servicio se inicie junto con el sistema y se pueda utilizar correctamente:

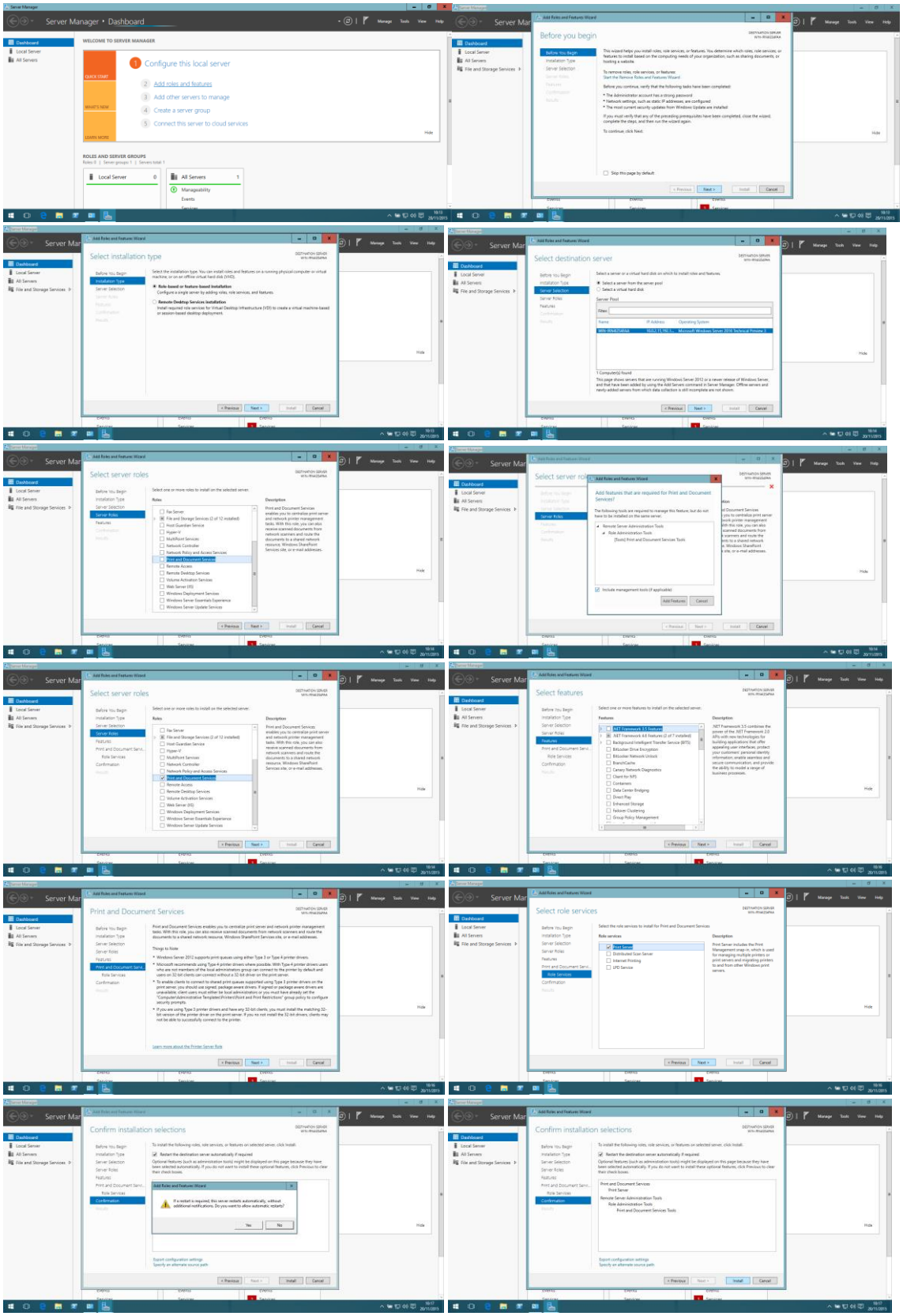
- cupsd_enable="YES"
- devfs_system_ruleset="system"

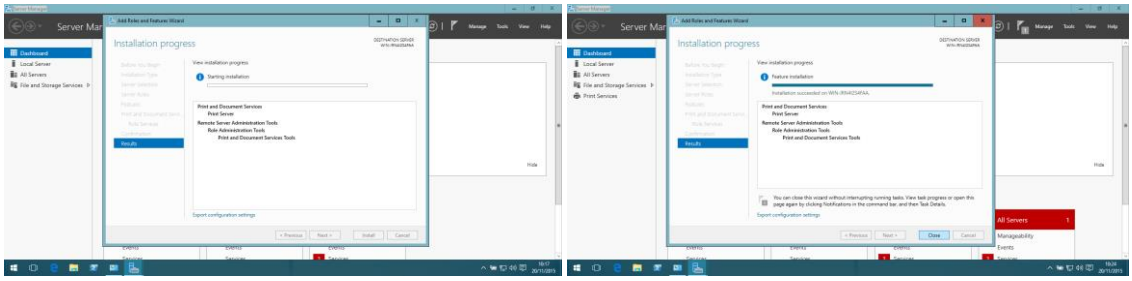
En el caso de Windows hemos de instalar Bullzip PDF Printer con su instalador.

Una vez instalada la impresora, debemos compartirla en Printer Properties > Sharing> Change Sharing options. Añadiremos también, en la pestaña “Advanced”, la opción para que conserve los documentos impresos.

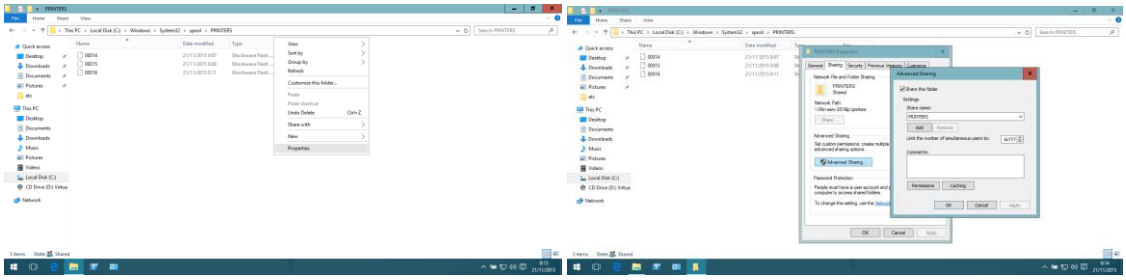


Ahora debemos abrir Windows Server Manager y añadir el rol del servidor de impresión.


Server Manager • Dashboard
Configure this local server
Add roles and features
Add other servers to manage
Create a server group
Connect this server to cloud services
Before you begin
Select installation type
Select destination server
Select server roles
Add features that are required for Print and Document Services?
Select features
Print and Document Services
Select role services
Confirm installation selections



Finalmente, para que la impresora sea visible desde fuera del equipo, debemos compartir la carpeta “C:\Windows\System32\spool\PRINTERS”. Una vez realizado esto, nuestra impresora será visible fuera de nuestro equipo accediendo por el nombre del servidor y la impresora.



Ahora las impresoras estarán disponibles para ser agregadas desde cualquier ordenador.

11. Servidor FTP

Para instalar el servidor FTP en FreeBSD utilizaremos la orden “pkg install proftpd” y para CentOS utilizaremos “yum -y install vsftpd”, según estemos en CentOS o en FreeBSD. Para la utilización de FTP utilizaremos una partición en un disco aparte.

Una vez creada la partición, configuraremos el servidor ftp modificando el archivo /etc/vsftpd/vsftpd.conf:

- Cambiar la línea “anonymous_enable=YES” a NO
- Descomentar las líneas :
    - ascii_upload_enable=YES
    - ascii_download_enable=YES
    - chroot_local_user=YES #Habilitar usuarios chroot locales
    - chroot_list_enable=YES #Habilitar lista chroot
    - chroot_list_file=/etc/vsftpd/chroot_list #Para especificar la ruta de la lista chroot
    - ls_recurse_enable=YES
- Cambiar la línea “listen=NO” a YES en caso de utilizar IPv4

- Cambiar la línea “listen_ipv6=YES” a NO en caso de no necesitar IPv6, como es nuestro caso.
- Añadir las siguientes líneas al final:
    - local_root=<carpeta> #En el caso de no escribir esta línea el directorio raíz será el directorio /home de cada usuario
    - use_localtime=YES #Utilizar el tiempo local
    - seccomp_sandbox=NO #En el caso de no poder acceder, añadir esta línea

Por último añadiremos al archivo “/etc/vsftpd/chroot_list” los usuarios a los que permitiremos acceder a directorios fuera de su directorio /home.

Iniciamos e habilitamos el servicio para que se inicie con el sistema con “systemctl start vsftpd” y “systemctl enable vsftpd”.

Podemos añadir un nivel más de seguridad con certificados TLS/SSL. Para ello nos dirigimos a la carpeta /etc/pki/tls y desde allí y como usuario root ejecutamos lo siguiente:

```
openssl req -sha256 -x509 -nodes -days 1825 -newkey rsa:4096 \
    -keyout private/vsftpd.key \
    -out certs/vsftpd.crt
```

Esto generará una clave de cifrado RSA en la que se incluirán los datos que se van pidiendo.

Ahora le damos permisos al archivo generado con la orden “chmod 400 certs/vsftpd.crt private/vscftpd.key”.
Ahora editamos el archivo /etc/vsftpd/vsftpd.conf y añadimos todo lo que viene a continuación al final del mismo:

```
# Habilita el soporte de TLS/SSL
ssl_enable=YES
# Deshabilita o habilita utilizar TLS/SSL con usuarios anónimos
allow_anon_ssl=NO
# Obliga a utilizar TLS/SSL para todas las operaciones, es decir,
# transferencia de datos y autenticación de usuarios locales.
# Establecer el valor NO, hace que sea opcional utilizar TLS/SSL.
force_local_data_ssl=YES
force_local_logins_ssl=YES
# Se prefiere TLSv1 sobre SSLv2 y SSLv3
ssl_tlsv1=YES
ssl_sslv2=NO
ssl_sslv3=NO
# Rutas del certificado y firma digital
rsa_cert_file=/etc/pki/tls/certs/vsftpd.crt
rsa_private_key_file=/etc/pki/tls/private/vsftpd.key
# Los desarrolladores de FileZilla decidieron con la versión 3.5.3 que
# eliminarían el soporte para el algoritmo de cifrado 3DES-CBC-SHA,
# con el argumento de que este algoritmo es una de los más lentos.
# Sin embargo con ésto rompieron compatibilidad con miles de
# servidores FTP que utilizan FTPES. La solución temporal, mientras
```

```
# los desarrolladores de FileZilla razonan lo absurdo de su
# decisión, es utilizar la siguiente opción:
ssl_ciphers=HIGH
# Filezilla además requiere desactivar la siguiente opción que puede
# romper compatibilidad con otros clientes. Cabe señalar que Filezilla
# se ha convertido en un desarrollo políticamente incorrecto por dejar
# de respetar los estándares.
require_ssl_reuse=NO
```

Para configurar proftpd en FreeBSD, debemos añadir primero la siguiente línea al archivo /etc/rc.conf:

- proftpd_enable="YES"

A continuación creamos el archivo proftpd.scoreboard necesario para iniciar el servicio con el comando “touch /var/run/proftpd.scoreboard”.

Una vez creado el scoreboard, creamos el grupo al que pertenecerá el usuario ftp. Para ello utilizamos la orden “pw groupadd -n ftp”.

Deberemos crear el directorio home para ftp. Para ello utilizamos las siguientes instrucciones:

- `mkdir /datosFTP`
- `cd /datosFTP`
- `mkdir in pub`
- `chown nobody in`
- `chmod 5777 in`
- `mkdir /datosFTP/pub/media`
- `mount_nullfs /media /datosFTP/pub/media`

La carpeta home será “/datosFTP”, la carpeta “in” será para los archivos recibidos y la carpeta “pub” será para los archivos que quieras que se puedan descargar.

Modificamos el archivo /etc/fstab, añadiendo la siguiente línea:

- /media /home/ftp/pub/media nullfs rw 0 0

Ahora es tiempo de configurar el servidor en si mismo, lo que haremos editando el archivo /usr/local/etc/proftpd.conf. A este archivo añadiremos lo siguiente:

```
ServerName     "FreeBSDFTP"
ServerType     standalone
DefaultServer  on
DefaultRoot    /datosFTP
Umask          022
MaxInstances   30
User           nobody
Group          nogroup
AllowOverwrite off
Port                21
AllowForeignAddress on
#Esto impide que se sobreescriban datos
```

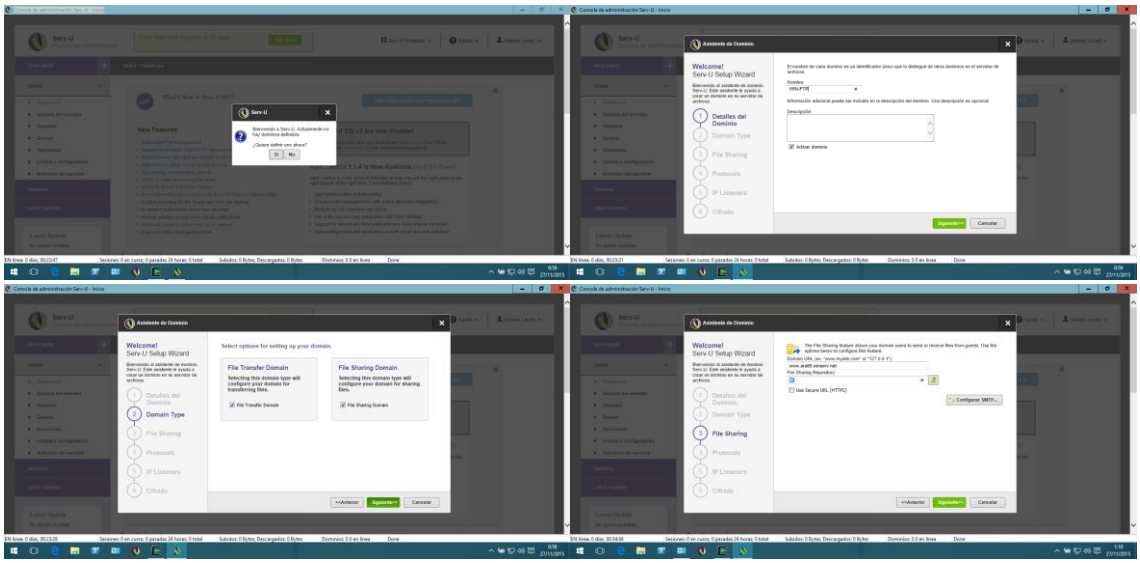
```
AllowOverwrite off
AllowRetrieveRestart on
#Las siguientes líneas tienen que ver con el log
ScoreboardFile /var/run/proftpd.scoreboard
SystemLog /var/log/proftpd.sys
TransferLog /var/log/proftpd.xfer
ServerLog /var/log/proftpd.serv

<Limit SITE_CHMOD>
  DenyAll
</Limit>

<Limit LOGIN>
  DenyAll
  AllowUser ftpuser
</Limit>

<Limit ALL>
  DenyAll
</Limit>

<Limit CDUP CWD LIST PWD>
  AllowAll
</Limit>

<Directory /datosFTP/in>
  <Limit STOR STOU>
    Allowall
  </Limit>
</Directory>

<Directory /datosFTP/pub>
  <Limit READ>
    AllowAll
  </Limit>
</Directory>

MaxClients 10

<Anonymous /datosFTP>
  <Limit LOGIN>
    AllowAll
  </Limit>

  User ftp
  Group ftp
  UserAlias anonymous ftp
  RequireValidShell off
</Anonymous>
```

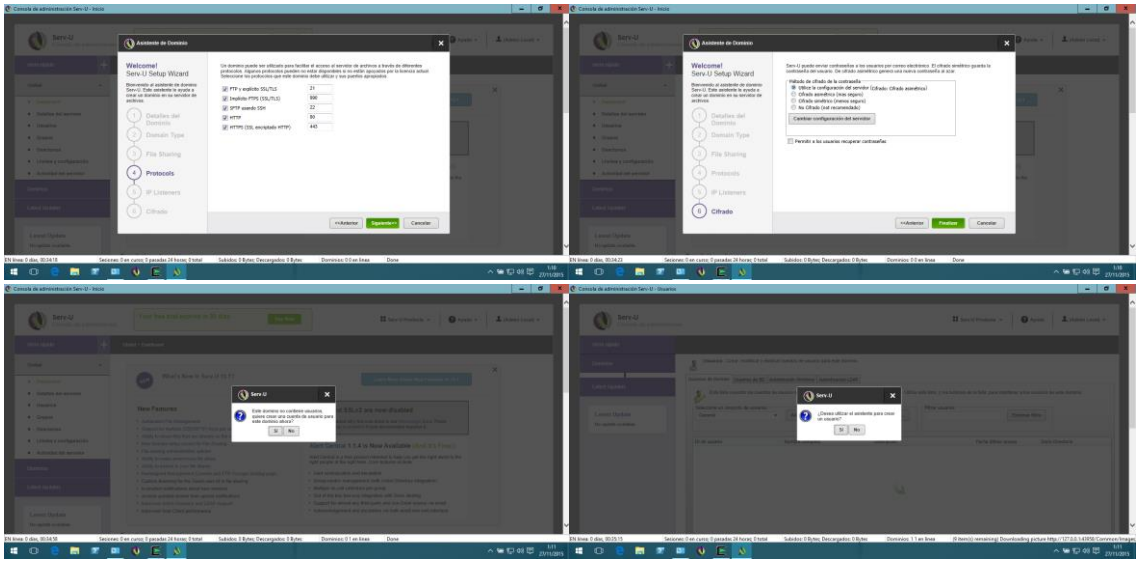
Añadimos la siguiente línea al archivo /etc/hosts:

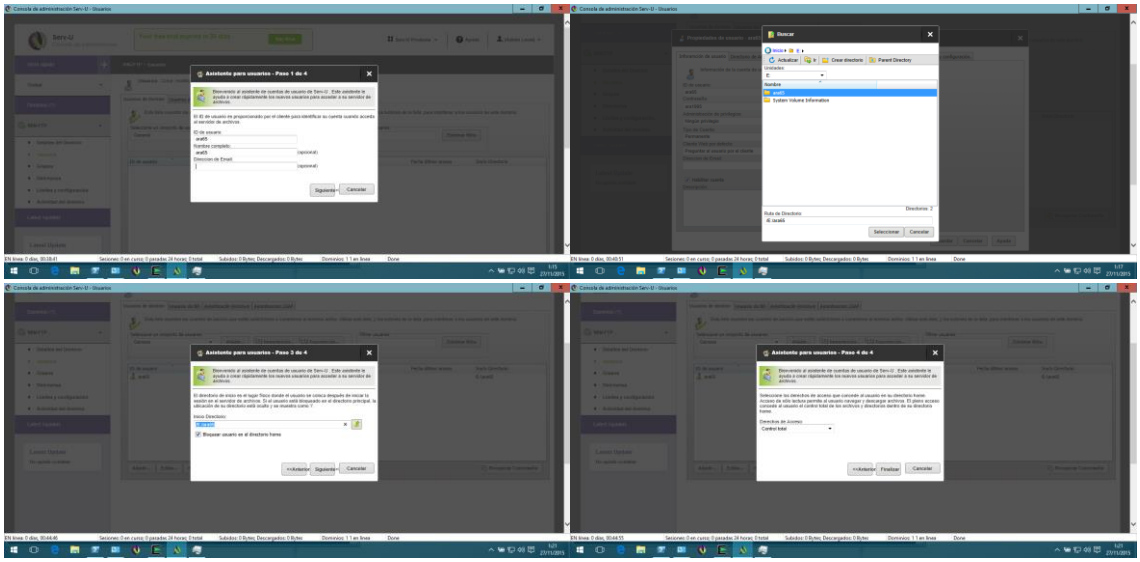
- 192.168.122.4 ara65

Iniciamos el servidor con “service proftpd start”.

Para Windows instalaremos Serv-U, el cual podemos descargar desde su página oficial. Una vez instalado, lo abrimos. En el caso de que nos diga que hace falta que habilitemos las cookies, abriremos internet explorer, no Microsoft Edge, iremos a las opciones de internet, a la pestaña Privacy y haremos click en Sites. Ahí escribiremos la dirección local 127.0.0.1 y le daremos a “Allow”. A continuación abriremos de nuevo Serv-U y comenzaremos con la configuración de nuestro servidor y la creación de nuestro usuario.







Una vez completados todos los pasos correspondientes podemos acceder al servicio con cualquier cliente ftp, como filezilla o lftp.

12. Emulación de otro Sistema Operativo

12.1. CYGWIN

Cygwin es un Software desarrollado para Windows que nos permite emular un sistema Linux casi a la perfección. Su instalación permite instalar multitud de entornos, como GNOME, MATE o XFCE, y herramientas como editores de texto, servidor de ssh, etc.

Para instalarlo vamos a la página oficial del proyecto y nos descargamos la versión correspondiente a nuestro sistema.

Una vez abierto el instalador seleccionamos el servidor más cercano a nosotros y, una vez realizado esto seleccionamos los paquetes a instalar. El proceso de descarga e instalación de paquetes, dependiendo de cuantos paquetes se intenten instalar al mismo tiempo, puede tardar entre 2 y 5 horas.

Una vez terminada la instalación, debemos crear los usuarios y los grupos que van a existir en nuestra emulación. Para ello, ejecutamos como administrador Cygwin y, una vez dentro de él escribimos uno a uno los siguientes comandos habiendo ejecutado el programa como admin :

```
- export CYGWIN='ntsec tty' #Exporta la terminal de Cygwin
- chmod 0755 /var
- ssh-host-config    #Asistente de configuración de ssh
- #Crea el archivo group, exportando los existentes en el sistema
- mkgroup > /etc/group
- #Crea el archivo passwd para usuarios normales
- mkpasswd -cl > /etc/passwd
```

- `#Crea el archivo passwd para usuarios de un dominio`
- `#mkpasswd -d > /etc/passwd`
- `chmod +rw /etc/group`
- `chmod +rw /etc/passwd`

Una vez configurado esto, ya podremos utilizar Cygwin naturalmente

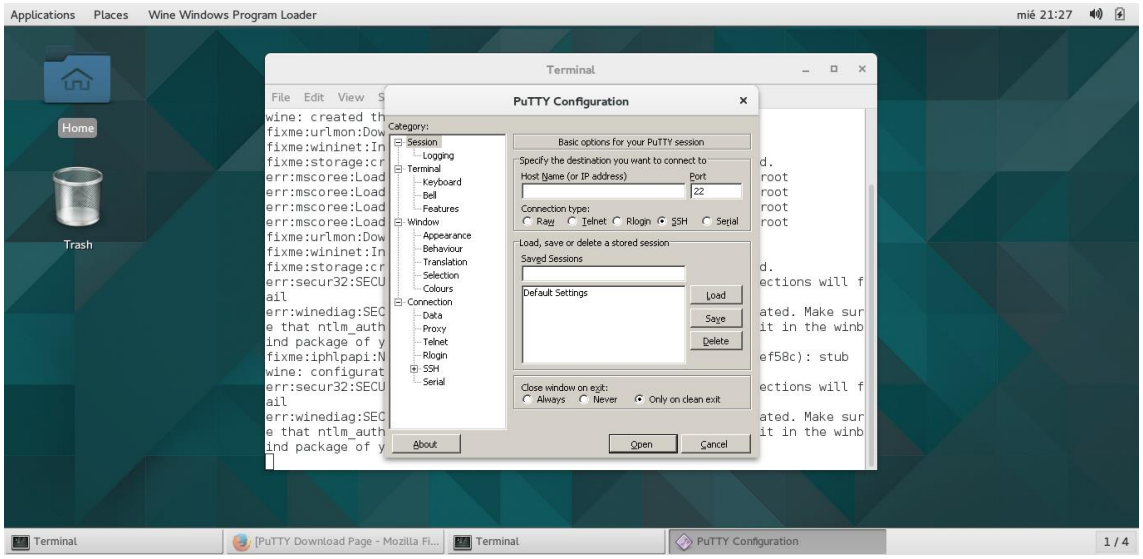
12.2. WINE

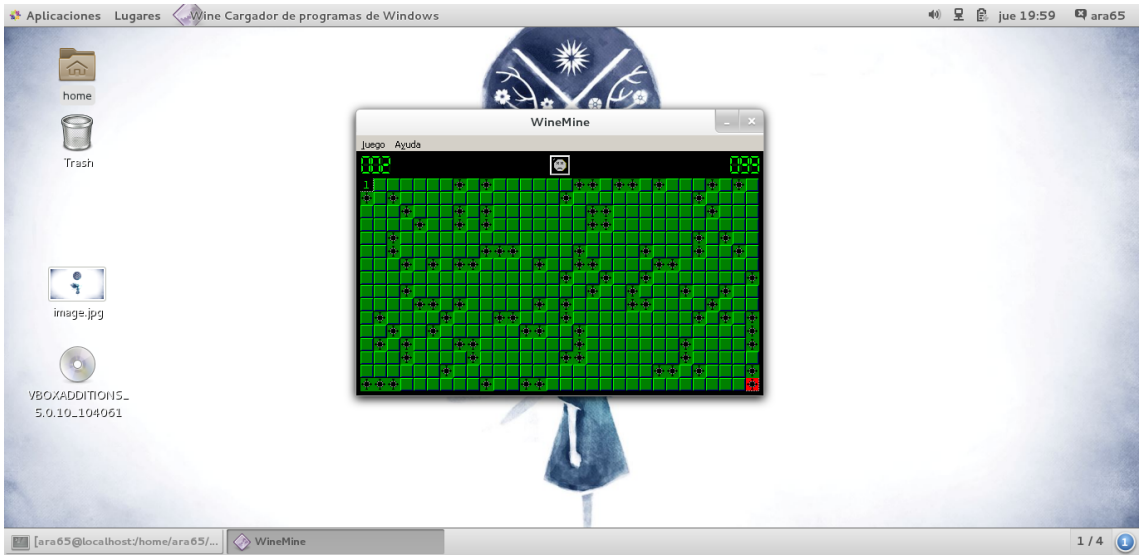
Wine es un emulador para entornos basados en Linux/Unix que permite emular ejecutables preparados para Windows, es decir, nos permite ejecutar archivos .exe dentro de un sistema basado en Linux/Unix.

Para instalarlo en FreeBSD utilizaremos la orden “pkg install i386-wine” y para hacerlo en CentOS utilizaremos “yum -y install wine”.
Una vez instalado y teniendo ya descargado el binario .exe que se desee ejecutar, iremos a la carpeta que contiene el binario y lo ejecutaremos con la orden “wine <nombre.exe>”.

La primera vez que se ejecute el programa nos dirá que necesita de unos ciertos programas, que son de las librerías necesarias para poder ejecutarlos correctamente.

Nos descargamos, por ejemplo, Putty y lo ejecutamos con la orden “wine Putty.exe”. Aquí mostramos la ejecución en FreeBSD y en CentOS, respectivamente.





13. Virtualización

La virtualización es un recurso muy utilizado por los usuarios para emular diferentes sistemas operativos sin tener que instalarlos directamente sobre el ordenador, cosa que puede derivar en problemas como fallos constantes de disco, perdida de datos de vital importancia …

Es por esto que existen herramientas que, mediante una imagen del sistema pueden, en un disco duro virtual, “instalar” dichos sistemas operativos en nuestro ordendor. En nuestro caso utilizamos Oracle Vitualbox para esta tarea, aunque existen otros com Vmware, Qemu, y alguno que otro más.

Además de esto, algunos de estos programas permiten insertar discos “en caliente”, así como redimensionar un disco que se ha quedado pequeño con una herramienta en concreto.