



UNIVERSIDAD  
POLITECNICA  
DE VALENCIA



**Universidad Politécnica de Valencia**

**Escuela Técnica Superior de Ingeniería Informática**

**PROYECTO FINAL DE CARRERA**

**“Servidor PDC con SAMBA y LDAP en  
Ubuntu 10.04 Lucid Lynx”**

**AUTOR:**

**David Sancho Baldó**

**DIRIGIDO POR:**

Dr. Lenin G. Lemus Zúñiga  
Profesor Titular  
Universidad Politécnica de  
Valencia

## INDICE

OBJETIVO .....	5
INTRODUCCIÓN.....	5
METODOLOGÍA DE TRABAJO: .....	5
BLOQUE I ANÁLISIS Y DISEÑO .....	7
CONCEPTOS BÁSICOS .....	7
BLOQUE II IMPLEMENTACIÓN Y PRUEBAS.....	9
0.  INSTALACIÓN.....	9
FIGURA 1: PASO 1 DE 7, SELECCIONANDO EL IDIOMA. ....	9
FIGURA 2: SELECCIÓN DE LA ZONA HORARIA .....	10
FIGURA 3: SELECCIÓN DE LA DISTRIBUCIÓN DEL TECLADO. ....	10
FIGURA 4: PARTICIONADO DEL DISCO DURO. ....	11
FIGURA 5: IDENTIFICACIÓN DEL USUARIO. ....	12
FIGURA 6: RESUMEN DE LA INSTALACIÓN. ....	12
FIGURA 7: REINICIAR SISTEMA. ....	13
1.  CAMBIAR LAS CONFIGURACIONES DE RED.....	15
CONFIGURACIÓN DE IP ESTÁTICA .....	15
MODIFICAR FICHEROS /ETC/HOSTS Y /ETC/HOSTNAME .....	16
2.  MONTAR UN SERVIDOR NFS EN UBUNTU LTS 10.04 LUCID LYNX.....	17
CONFIGURANDO NFS EN UBUNTU 10.04.....	17
3.  SAMBA + LDAP EN UBUNTU 10.04.....	19
INSTALAR LOS PAQUETES NECESARIOS.....	19
CONFIGURANDO LDAP .....	20
CONFIGURANDO SAMBA.....	21
FIGURA 8: LDAP SERVER UNIFORM RESOURCE IDENTIFIER.....	28
FIGURAS 9 Y 10: NOMBRE DE NUESTRO DOMINIO Y VERSIÓN DEL PROTOCOLO LDAP. ....	29
FIGURAS 11 Y 12: HACER EL USUARIO ROOT LOCAL ADMINISTRADOR DE LA BASE DE DATOS Y REQUERIMIENTO DE LOGIN EN LA BASE DE DATOS LDAP. ....	29
FIGURAS 13 Y 14: CUENTA CON PRIVILEGIOS DE LDAP Y PASSWORD. ....	29
FIGURAS 15 Y 16: MÓDULOS DE AUTENTICACIÓN PAM A ACTIVAR. ....	30
4.  INSTALACIÓN Y CONFIGURACIÓN DE LAM.....	31
FIGURAS 17 Y 18: PANTALLA DE LOGIN SERVIDOR LDAP Y ACCESO A LA CONFIGURACIÓN BÁSICA DEL LAM. ....	31
FIGURA 19: PROPIEDADES DEL SERVIDOR LDAP – CONFIGURACIÓN GENERAL. ....	32
FIGURA 20: PROPIEDADES DEL SERVIDOR LDAP – TIPOS DE CUENTA. ....	32
FIGURA 21: ÁRBOL DEL DIRECTORIO. ....	33
FIGURA 22: ADMINISTRACIÓN DE USUARIOS. ....	33

FIGURA 23: NUEVO USUARIO. ....	34
FIGURAS 24 Y 25: CONFIGURACIÓN DE CUENTA DE USUARIOS UNIX Y SAMBA. ....	34
5. INSTALACIÓN Y CONFIGURACIÓN DE WEBMIN. ....	35
FIGURA 26: INTERFAZ GRÁFICA WEBMIN. ....	35
FIGURA 27: MENU SAMBA WINDOWS FILE SHARING. ....	36
FIGURA 28: CREANDO EL RECURSO COMPARTIDO. ....	36
6. SERVIDOR DNS EN UBUNTU 10.04. ....	37
¿QUÉ ES DNS? .....	37
INSTALACIÓN DEL SERVIDOR DE DNS BIND .....	37
CONFIGURAR NUESTRA ZONA PRIMARIA DE DNS USANDO WEBMIN .....	37
CONFIGURAMOS EL SERVIDOR PARA USARSE A SÍ MISMO PARA RESOLVER DNS .....	38
7. SERVIDOR DHCP EN UBUNTU 10.04. ....	39
QUE ES DHCP ? .....	39
FIGURA 29. RED CON SERVIDOR DHCP. ....	39
INSTALAR UN SERVICIO DHCP EN UBUNTU Y DEBIAN .....	39
CONFIGURANDO EL SERVIDOR DHCP .....	40
CONFIGURAR UTILIZANDO EL MÉTODO DE RANGO DE DIRECCIONES (IP POOL) .....	40
CONFIGURAR UTILIZANDO EL MÉTODO DE DIRECCIONES MAC .....	41
CONFIGURAR EL CLIENTE DHCP EN LINUX UBUNTU .....	42
8. CONFIGURACIÓN DE LOS TERMINALES CLIENTES. ....	45
UNIENDO WINDOWS XP AL DOMINIO .....	45
FIGURA 30: PROPIEDADES DEL SISTEMA. ....	45
FIGURA 31: ASISTENTE PARA UNIR A UN DOMINIO 1/7. ....	45
FIGURA 32: ASISTENTE PARA UNIR A UN DOMINIO 2/7. ....	45
FIGURA 33: ASISTENTE PARA UNIR A UN DOMINIO 3/7. ....	45
FIGURA 34: ASISTENTE PARA UNIR A UN DOMINIO 4/7. ....	46
FIGURA 35: ASISTENTE PARA UNIR A UN DOMINIO 5/7. ....	46
FIGURA 36: ASISTENTE PARA UNIR A UN DOMINIO 6/7. ....	46
FIGURA 37: ASISTENTE PARA UNIR A UN DOMINIO 7/7. ....	46
FIGURA 38: INICIAR SESIÓN COMO USUARIO DEL DOMINIO. ....	46
FIGURA 39: SESIÓN DE "USUARIODEPRUEBA". ....	46
UNIENDO WINDOWS 7 AL DOMINIO .....	47
FIGURA 40: PROPIEDADES DEL SISTEMA. ....	47
FIGURA 41: NOMBRE DE EQUIPO. ....	47
FIGURA 42: ASISTENTE PARA UNIR A UN DOMINIO 1/7. ....	47
FIGURA 43: ASISTENTE PARA UNIR A UN DOMINIO 2/7. ....	47
FIGURA 44: ASISTENTE PARA UNIR A UN DOMINIO 3/7. ....	48
FIGURA 45: ASISTENTE PARA UNIR A UN DOMINIO 4/7. ....	48
FIGURA 46: ASISTENTE PARA UNIR A UN DOMINIO 5/7. ....	48
FIGURA 47: ASISTENTE PARA UNIR A UN DOMINIO 6/7. ....	48
FIGURA 48: ASISTENTE PARA UNIR A UN DOMINIO 7/7. ....	48
FIGURA 49: INICIAR SESIÓN COMO USUARIO DEL DOMINIO. ....	48
FIGURA 50: SESIÓN DE "USUARIODEPRUEBA". ....	49
FIGURA 51: AGREGAR NUEVO USUARIO. ....	49
FIGURA 52: AGREGAR NUEVO USUARIO. ....	49
FIGURA 53: CUENTAS DE USUARIO. ....	49
UNIENDO UBUNTU 10.04 AL DOMINIO .....	50
FIGURA 54: INICIAR SESIÓN COMO USUARIO DEL DOMINIO. ....	52

<i>FIGURA 55: SESIÓN DE "USUARIODEPRUEBA"</i> .....	52
9. VALIDACIÓN Y PRUEBAS.....	53
10. CONCLUSIONES .....	55
11. TRABAJO FUTURO.....	57
12. BIBLIOGRAFÍA.....	59

---

## Objetivo

---

Autenticar usuarios desde ordenadores con S.O. heterogéneos (**Windows** y **Linux**) contra un servidor **Linux**, y poder acceder a los recursos alojados en dicho servidor.

---

## Introducción

---

Actualmente el sistema operativo **Linux** se ha extendido bastante en cuanto a implementación de servidores se refiere. A nivel de usuario, en instalaciones desktop, no es muy aceptado, ya que en ocasiones hay problemas de compatibilidad de archivos, programas, drivers, dispositivos, etc. Pese a esto hoy en día las últimas versiones de **Linux** son casi tan fáciles de usar y personalizables como **Windows**.

Con esto no se pretende demostrar que **Windows** o **Linux** sea mejor SO, hay personas que prefieren **Windows** por su accesibilidad, facilidad de uso y potente soporte gráfico. Y en cambio hay gente que prefiere **Linux** por su estabilidad y optimización de recursos, sin olvidarnos de la gestión de la seguridad de ficheros y que es software libre.

En nuestro caso, la experiencia adquirida se utilizará para montar en el centro juvenil AMICS un servidor basado en **Linux (Ubuntu)** que permita autenticar a los usuarios del centro y compartir recursos.

Debido a que trabajo en una entidad no lucrativa, el aprovechamiento y optimización de recursos es la mejor forma de gestionar los equipos en el Centro. Por tanto tomamos la decisión de probar un servidor LDAP con **SAMBA** para autenticar clientes **Windows** y **Linux** contra él en un Intel® Celerón de 512 MB de **RAM** con la versión **Ubuntu 8.04** instalada. Debido al impecable funcionamiento durante 2 años, se decidió comprar un equipo más confiable, ya que al estar encendido las 24h del año el antiguo dejó de funcionar. Ahora el servidor en **Ubuntu 10.04** funciona perfectamente en un Intel® Core™2 Duo CPU E7400 @ 2.80GHz con 2GB de **RAM**.

### *Metodología de trabajo:*

- En el Centro Juvenil se trabaja básicamente con herramientas ofimáticas, aunque también se usan herramientas de gestión de usuarios del Centro, tesorería y alguna herramienta de diseño que otra.
- Para la mayoría de gente digamos que se podría utilizar tanto clientes **Windows** como **Linux** aunque por comodidad de los usuarios y para evitar problemas de incompatibilidad de ficheros de **Office** y **OpenOffice**, los equipos que se dedican a gestión del Centro son clientes **Windows**.

- Se propone implementar un servidor **PDC** (Primary Domain Controller) utilizando **LDAP** y **SAMBA** para la compartición de recursos con el fin de autenticar clientes **Windows-Linux** y acceder de manera permanente a recursos como impresoras compartidas, documentos y directorios personales.
  
- Puesta en marcha del centro:
  - Instalación de un S.O. **Linux (Ubuntu 10.04 Lucid Lynx)**
  - Configurar **LDAP**
  - Configurar **SAMBA**
  - Añadir los clientes al dominio
  - Compartir recursos y acceder a ellos

## BLOQUE I Análisis y diseño

### Conceptos básicos

---

**Servidor:** En informática, un servidor es una computadora que, formando parte de una red, provee servicios a otras computadoras denominadas clientes.

**PDC:** (Primary Domain Controller) Un controlador de dominio es una entidad administrativa, esto es, no es un ordenador en concreto, sino un conjunto de ordenadores agrupados que se ciñen a unas reglas de seguridad y autenticación comunes. Para regular un dominio, se precisa al menos de un equipo que sea el controlador principal, la fuente primera donde se almacenan las reglas del dominio, y donde serán consultadas esas reglas en última instancia. Un controlador primario de dominio (**PDC**) puede implementarse tanto bajo **Windows** como bajo **Linux**.

**NFS:** El Network File System (Sistema de archivos de red), o **NFS**, es un protocolo de nivel de aplicación, según el Modelo **OSI**. Es utilizado para sistemas de archivos distribuido en un entorno de red de computadoras de área local. Posibilita que distintos sistemas conectados a una misma red accedan a ficheros remotos como si se tratara de locales.

**LDAP:** ("Lightweight Directory Access Protocol", «Protocolo Ligero de Acceso a Directorios») es un protocolo de tipo cliente-servidor para acceder a un servicio de directorio. Se usó inicialmente como un front-end o interfaz final para X.500, pero también puede usarse con servidores de directorio únicos y con otros tipos de servidores de directorio.

**¿Qué es un servicio de directorio?:** Un directorio es como una base de datos, pero en general contiene información más descriptiva y más basada en atributos. La información contenida en un directorio normalmente se lee mucho más de lo que se escribe. Como consecuencia los directorios no implementan normalmente los complicados esquemas para transacciones o esquemas de reducción (rollback) que las bases de datos utilizan para llevar a cabo actualizaciones complejas de grandes volúmenes de datos. Por contra, las actualizaciones en un directorio son usualmente cambios sencillos de «todo o nada», si es que se permiten en algo.

**¿Cómo funciona LDAP?** El servicio de directorio **LDAP** se basa en un modelo cliente-servidor. Uno o más servidores **LDAP** contienen los datos que conforman el árbol del directorio **LDAP** o base de datos troncal, el cliente **LDAP** se conecta con el servidor **LDAP** y le hace una consulta. El servidor contesta con la respuesta correspondiente, o bien con una indicación de dónde puede el cliente hallar más información (normalmente otro servidor **LDAP**). No importa con qué servidor **LDAP** se conecte el cliente: siempre observará la misma vista del directorio; el nombre que se le presenta a un servidor **LDAP** hace referencia a la misma entrada a la que haría referencia en otro servidor **LDAP**. Es ésta una característica importante de un servicio de directorios universal como **LDAP**.

**Backends, objetos y atributos en LDAP:** **slapd** se suministra con tres diferentes bases de datos de backend (dorsal, o base de datos de segundo plano) entre las que elegir. Se trata de **LDBM**, una base de datos de gran rendimiento basada en disco: **SHELL**, una interfaz de base de datos

para órdenes arbitrarias de **UNIX** o guiones (scripts) del intérprete de órdenes (shell); y **PASSWD**, una sencilla base de datos de contraseñas.

**LDIF**: En general se usa para importar y exportar información de directorio entre servidores de directorios basados en **LDAP**, o para describir una serie de cambios que han de aplicarse al directorio. (siglas de "**LDAP** interchange format", «formato de intercambio de **LDAP**»). Un fichero LDIF almacena información en jerarquías de entradas orientadas a objeto. El paquete de software **LDAP** que va a utilizar incluye una utilidad para convertir ficheros **LDIF** a formato **LDBM**.

**SAMBA**: **SAMBA** es una implementación libre del protocolo de archivos compartidos de Microsoft **Windows** (antiguamente llamado **SMB**, renombrado recientemente a **CIFS**) para sistemas de tipo **UNIX**. De esta forma, es posible que ordenadores con **GNU/Linux**, **Mac OS X** o **UNIX** en general se vean como servidores o actúen como clientes en redes de **Windows**. **SAMBA** también permite validar usuarios haciendo de Controlador Principal de Dominio (**PDC**), como miembro de dominio e incluso como un dominio Active Directory para redes basadas en **Windows**; aparte de ser capaz de servir colas de impresión, directorios compartidos y autenticar con su propio archivo de usuarios.

Entre los sistemas tipo **UNIX** en los que se puede ejecutar **SAMBA**, están las distribuciones **GNU/Linux**, **Solaris** y las diferentes variantes **BSD** entre las que podemos encontrar el **Mac OS X Server** de Apple.

**PAM**: Pluggable Authentication Modules (**PAM**) es un mecanismo de autenticación flexible que permite abstraer las aplicaciones y otro software del proceso de identificación.

A la hora de identificar a un usuario para ingresar en su cuenta hay muchos sistemas, desde la clásica contraseña hasta diversos sistemas de identificación biométrica o claves de un solo uso. Muchos software necesitan identificar a los usuarios (servidores de correo, web, bases de datos y otros) antiguamente el programa debía ser modificado para usar cada sistema de identificación. Sin embargo, al igual que un programa de retoque fotográfico no necesita ser compilado para cada tableta gráfica(TG), escáner ni ningún otro dispositivo, en cambio se usa un modo estándar para usar una tableta gráfica(TG) u otro dispositivo apuntador y por otro lado se usa un modulo que permite manejar el dispositivo (Controlador de dispositivo).

Cuando una aplicación se prepara para usar **PAM**, esta se encarga de la autenticación y puede usar diversos métodos sin modificar la aplicación (contraseña, token, biometría y otros). Además, permite otras opciones como permitir el acceso sólo durante horarios preestablecidos.



## BLOQUE II Implementación y pruebas

### 0. INSTALACIÓN

Para empezar, se va a utilizar la imagen Desktop de **Ubuntu 10.04 Lucyd Lynx**, que nos dejará el sistema completamente instalado mediante el arranque gráfico. Entramos en la página web de **Ubuntu** y descargamos la imagen que necesitamos: <http://www.ubuntu.com>, pinchamos sobre *download* y elegimos la versión (en nuestro caso la 10.04 LTS Desktop Edition de 32 o 64 bits, según sea nuestro equipo). Podemos utilizar la de 64 bits sin problemas.

Una vez descargada la imagen del CD la grabamos teniendo en cuenta que no se grabará el archivo en el CD, sino que es una imagen que tendremos que grabar con alguna aplicación especializada en grabar CDs (por ejemplo **Nero**).

El CD que hemos grabado será de autoarranque, con lo que bastará introducirlo en la unidad lectora y tener adecuadamente configurado el arranque de la **BIOS** (seleccionado como primer dispositivo de arranque el CD). Simplemente con iniciar el ordenador comenzará el proceso. La primera pantalla que nos mostrará será la de selección de idioma, seleccionamos el que deseemos (español, catalán, esperanto...) y nos muestra un menú que nos permitirá comenzar la instalación directamente o arrancar en modo *live*. Podemos probar antes de instalar para comprobar que nos detecta todo bien. Desde el sistema operativo *live* también nos permitirá la opción de instalar. Si seleccionamos instalar, el proceso empezará a continuación.

En el primer paso nos pedirá que seleccionemos el idioma de la instalación. Seleccionamos el idioma que deseemos y pulsamos Adelante.



Figura 1. Paso 1 de 7, seleccionando el idioma.

Por norma general, ya aparecerá la ciudad seleccionada por defecto, Madrid, en caso contrario o que no sea el caso, seleccionamos la adecuada. Comprobamos que la zona horaria es la correcta, que la hora lo es también y pulsamos “**Adelante**”.



Figura 2: Selección de la zona horaria

El siguiente paso será seleccionar la distribución de teclado de nuestro equipo, la nuestra por lo general será España-España. En la parte inferior izquierda podremos probar que funciona correctamente pulsando sobre teclas como la ñ, la coma, punto y coma...



Figura 3: Selección de la distribución del teclado.

En este paso seleccionaremos el disco o partición donde instalar **Ubuntu**. En nuestro caso tenemos un disco exclusivamente para el servidor y seleccionamos la opción “Borrar y usar disco entero”. Si por el contrario tenemos varias particiones y queremos instalarlo en una en concreto deberemos seleccionar la opción “Especificar particiones manualmente”, en la que encontraremos una sencilla interfaz para gestionarlas.

Hay que tener en cuenta que las particiones en **GNU/Linux** no son como las particiones **Windows** que son **FAT** y **NTFS**, en **GNU/Linux** se necesitan dos particiones, una para el sistema de ficheros y otra para el espacio de intercambio (algo parecido a la memoria virtual de **Windows**). La partición de intercambio se denomina **SWAP**, y dependiendo de quien lo instale suele variar entre el mismo espacio que tengamos por memoria **RAM** hasta 2.5 veces la memoria **RAM** que tengamos. La partición para el sistema de ficheros es la que utiliza para guardar los datos, programas y el sistema operativo propiamente dicho. Hay varios sistema de ficheros que podemos utilizar: **ext2**, **ext3**, **ext4** **ResiserFS**, **XFS**... El más extendido en **GNU/Linux** es el **ext4**. Cada sistema de ficheros tiene sus propias características se puede encontrar más información en internet. En nuestro caso, hemos utilizado la opción de Borrar y usar el disco entero, y ha sido la propia aplicación la que nos ha creado las particiones de la manera más apropiada.



Figura 4: Particionado del disco duro.

El siguiente paso es la identificación del usuario. Hay que tener en cuenta que el usuario que utilizemos aquí será el usuario administrador (root) del equipo. Completamos los datos que nos solicita y pongamos nombre al equipo.



**Instalar**

**¿Quién es usted?**

¿Cómo se llama?  
Administrador del sistema ✓

¿Qué nombre desea usar para iniciar sesión?  
administrador ✓

Si este equipo va a ser usado por más de una persona, podrá configurar varias cuentas después de la instalación.

Escoja una contraseña para mantener su cuenta segura.  
●●●●●●●● ●●●●●●●● Fortaleza: normal

¿Cuál es el nombre de este equipo?  
servidor-amics ✓

Este nombre se usará si hace que el equipo sea visible a otros equipos de la red.

Iniciar sesión automáticamente

Solicitar mi contraseña para iniciar sesión

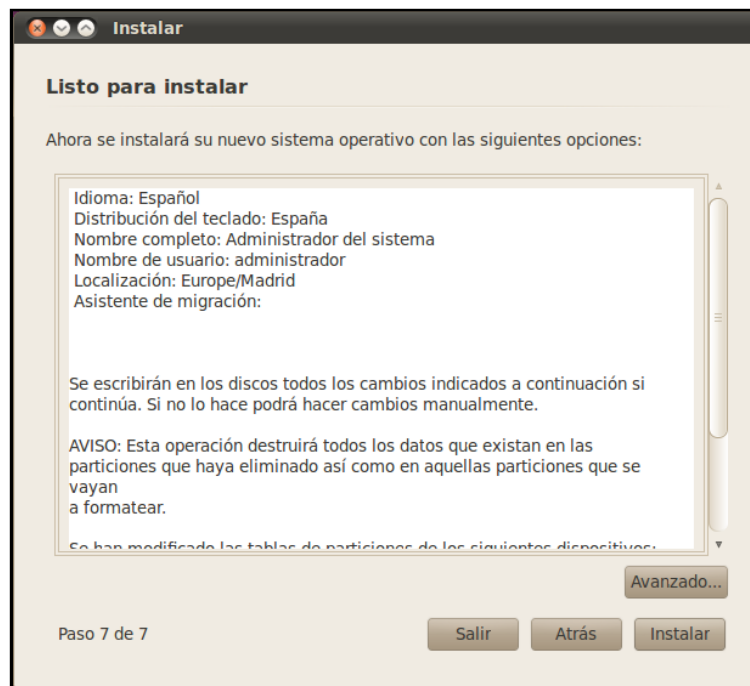
Solicitar mi contraseña para iniciar sesión y descifrar mi carpeta personal

Paso 5 de 7

Salir Atrás Adelante

Figura 5: Identificación del usuario.

Esta pantalla nos mostrará un resumen de las opciones seleccionadas. Es la última antes de la instalación. Cuando pulsemos en Adelante comenzará la instalación. Si el equipo está conectado a internet, durante la instalación actualizará los paquetes que necesite.



**Instalar**

**Listo para instalar**

Ahora se instalará su nuevo sistema operativo con las siguientes opciones:

Idioma: Español  
Distribución del teclado: España  
Nombre completo: Administrador del sistema  
Nombre de usuario: administrador  
Localización: Europe/Madrid  
Asistente de migración:

Se escribirán en los discos todos los cambios indicados a continuación si continúa. Si no lo hace podrá hacer cambios manualmente.

**AVISO:** Esta operación destruirá todos los datos que existan en las particiones que haya eliminado así como en aquellas particiones que se vayan a formatear.

Se han modificado las tablas de particiones de los siguientes dispositivos:

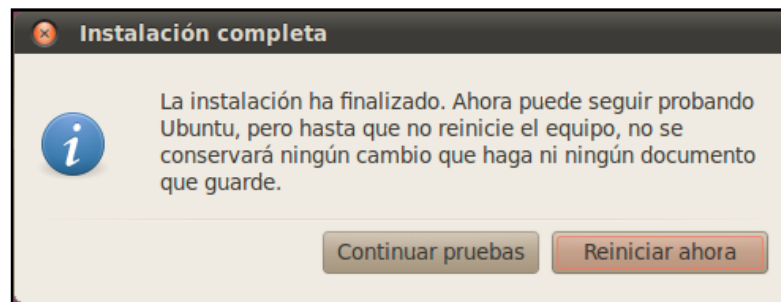
Avanzado...

Paso 7 de 7

Salir Atrás Instalar

Figura 6: Resumen de la instalación.

Una vez terminada la instalación, nos pedirá que retiremos el CD y que reiniciemos el equipo. Retiramos el CD, pulsamos “ENTER” y comenzará la primera ejecución. Tras introducir usuario y contraseña, tendremos nuestro equipo con **Ubuntu** dispuesto para ser utilizado.



*Figura 7: Reiniciar sistema.*



## 1. Cambiar las configuraciones de red

### Configuración de IP estática

Una vez finalizada la instalación de **Ubuntu 10.04** vamos a configurar nuestro servidor para que tenga una dirección IP estática.

Para ello modificaremos el fichero **interfaces** de la siguiente forma:

```
usuario@pc:~$ sudo gedit /etc/network/interfaces
```

```
auto lo eth0
iface lo inet loopback
iface eth0 inet static
address 192.168.1.250
broadcast 192.168.1.255
netmask 255.255.255.0
gateway 192.168.1.1
```

Y reiniciamos el interfaz de red para aplicar los cambios.

```
usuario@pc:~$ /etc/init.d/networking restart
```

Para comprobar que nuestra dirección IP ha sido cambiada a la deseada, ejecutamos el comando **ifconfig**:

```
usuario@pc:~$ ifconfig
```

En la salida de este comando, en propiedades de nuestra Ethernet debería aparecer la IP cambiada.

Si ahora le hacemos un ping a *google* y no hay problemas en la recepción de paquetes la operación habrá sido satisfactoria y deberíamos tener internet.

```
usuario@pc:~$ ping www.google.es
```

Si no sucede lo esperado podemos reiniciar la máquina y probablemente se solucionen el problema.

## *Modificar ficheros /etc/hosts y /etc/hostname*

En el primer fichero añadiremos a la primera y segunda líneas “127.0.0.1 servidor-amics.amics.local servidor-amics localhost” “127.0.1.1 servidor-amics.amics.local servidor-amics” y en el segundo fichero simplemente debe poner “servidor-amics.amics.local”:

*Nota: esto lo hacemos para añadir el nombre de dominio al equipo y al mismo tiempo podríamos cambiar el nombre del equipo, en nuestro caso “servidor-amics”.*

```
usuario@pc:~$ sudo gedit /etc/hosts
```

```
127.0.0.1 servidor-amics.amics.local servidor-amics localhost
127.0.1.1 servidor-amics.amics.local servidor-amics
```

```
usuario@pc:~$ sudo gedit /etc/hostname
```

```
servidor-amics.amics.local
```



## 2. Montar un Servidor NFS en Ubuntu LTS 10.04 Lucid Lynx

El Network File System (Sistema de archivos de red), o **NFS**, es un protocolo de nivel de aplicación, según el Modelo **OSI**. Es utilizado para sistemas de archivos distribuido en un entorno de red de computadoras de área local. Posibilita que distintos sistemas conectados a una misma red accedan a ficheros remotos como si se tratara de locales. Originalmente fue desarrollado en 1984 por **Sun Microsystems**, con el objetivo de que sea independiente de la máquina, el sistema operativo y el protocolo de transporte, esto fue posible gracias a que está implementado sobre los protocolos **XDR** (presentación) y **ONC RPC** (sesión). El protocolo **NFS** está incluido por defecto en los Sistemas Operativos **UNIX** y la mayoría de distribuciones **Linux**.

### Configurando NFS en Ubuntu 10.04

Para nuestro servidor **NFS** vamos a asumir que nuestra red local es 192.168.1.0/24 y que vamos a compartir nuestro directorio de usuario en el servidor “servidor-amics” (192.168.1.250).

Instalamos el **nfs-kernel-server** (los demás paquetes como: **portmap** y **nfs-common** se agregan automáticamente como dependencias del primero).

```
usuario@pc:~$ sudo apt-get install nfs-kernel-server
```

Editamos los archivos “hosts.allow” y “exports”:

```
usuario@ pc:~$ sudo gedit /etc/hosts.allow
```

Añadimos lo siguiente:

```
portmap: 192.168.1.0/24
nfs: 192.168.1.0/24
```

```
usuario@ pc:~$ sudo gedit /etc/exports
```

Añadimos lo siguiente:

```
/home/usuario 192.168.1.0/24(rw, sync, no_root_squash, no_subtree_check)
```

*Nota: El atributo “usuario” se refiere a una carpeta del home de un usuario ya creado en el sistema, también podemos exportar otras carpetas previamente creadas, o todo el “/home” que es más cómodo para una red con varios usuarios **Linux**.*

Reiniciamos los servicios correspondientes:

```
usuario@ pc:~$ sudo service portmap restart && sudo service nfs-kernel-server restart
```

Es muy probable que **portmap** no se reinicie con el comando *service*. Por lo que podemos hacerlo mediante:

```
usuario@ pc:~$ sudo /etc/init.d/portmap restart
```

De manera opcional podemos configurar para que los servicios se inicien automáticamente al inicio. Esto lo haremos así:

```
usuario@pc:~$ sudo update-rc.d -f portmap defaults && sudo update-rc.d -f nfs-kernel-server defaults
```

Finalmente para probar el correcto funcionamiento de nuestro **NFS** podemos montarlo en otra computadora de la red local, en la cual necesitamos tener instalado el paquete **nfs-common**.

Lo montaremos así:

```
usuario@pc-cliente:~$ sudo mount -t nfs 192.168.1.250:/home/usuario /media/discousuario
```

*Nota: El directorio "discousuario" debe haber sido creado antes de montarlo en la ruta especificada /media/.*

### 3. Samba + LDAP en Ubuntu 10.04

---

Vamos a configurar nuestro servidor **Ubuntu** como Controlador Primario de Dominio en el que se puedan autenticar máquinas con **Windows** o **Linux**. **OpenLDAP** es un servidor de **LDAP** (Lightweight Directory Access Protocol), que es 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** es algo similar a una base de datos, aunque no exactamente lo mismo. Una implementación de **LDAP** es **Active Directory** de **Microsoft**.

La idea es poder utilizar el servidor **Ubuntu** como Controlador de Dominio para máquinas **Windows-Linux** o poder autenticar directamente a los usuarios contra el servidor **LDAP**, teniendo un único punto de acceso para ambos sistemas.

#### PLATAFORMA:

\* **Ubuntu** Server ó Desktop 10.04 LTS (Lucid Lynx).

#### PRERREQUISITOS:

- \* Tener **Ubuntu** Server ó Desktop 10.04 LTS (Lucid Lynx) instalado en la máquina.
- \* Un servicios **NFS** de exportación de directorios de usuario **UNIX** (si queremos trabajar con maquinas **UNIX**).

#### PUNTOS DE INTERÉS PREVIOS:

- \* El nombre puesto al equipo durante la instalación **servidor-amics**
- \* El nombre de dominio completo cualificado será: **servidor-amics.amics.local**
- \* Después de la configuración, el archivo /etc/hostname se modificó a: **servidor-amics.amics.local**
- \* Tras la instalación, el archivo /etc/hosts se modificó para que la línea quedara como:  
**127.0.1.1 servidor-amics.amics.local servidor-amics** para asegurarnos que no hay problemas de resolución de nombres.
- \* El dominio **LDAP** es: **amics.local**
- \* Esto se traduce a Base DN de: dc= **amics**, dc= **local**

#### *Instalar los paquetes necesarios*

Vamos a instalar los paquetes necesarios para servidor **openLDAP** y los de **SAMBA**:

```
usuario@ pc:~$ sudo apt-get install slapd ldap-utils samba samba-doc libpam-smbpass  
smbclient smbldap-tools
```

*Nota: Por defecto el demonio **slapd** está configurado con las mínimas opciones para ejecutarse.*

## Configurando LDAP

*Nota: **OpenLDAP** utiliza un directorio independiente que contiene el `cn = config` Árbol de información de directorio (DIT). El `cn = config` (DIT) se utiliza para configurar dinámicamente el demonio **slapd**, lo que permite la modificación de las definiciones de esquema, los índices, ACLs, etc sin detener el servicio.*

El “backend = cn” sólo tiene una configuración mínima y tendrá opciones de configuración adicionales con el fin de crear el directorio *frontend*. La interfaz se rellenará con una “clásica” que será compatible con las aplicaciones de libreta de direcciones y cuentas con **UNIX** Posix. Posix permite la autenticación a varias aplicaciones, como las aplicaciones web, correo electrónico Mail Transfer Agent (MTA) de aplicaciones, etc.

Para poder autenticar aplicaciones externas con **LDAP** necesitaremos configurarlas una a una.

\* Recuerda cambiar `dc=amics`, `dc=local` los siguientes ejemplos para configurar tu servidor **LDAP**.

Primero cargaremos los ficheros de esquema necesarios. Abrimos un terminal y escribimos:

```
usuario@ pc:~$ sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/cosine.ldif
usuario@ pc:~$ sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/nis.ldif
usuario@ pc:~$ sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/inetorgperson.ldif
```

Ahora crearemos el fichero **LDIF** “`backend.ldif`” (se creará en el `/home` del administrador, si queremos podemos crearlo en una carpeta de ese home para una mayor organización pero deberéis modificar las siguientes instrucciones que utilicen esos ficheros)

```
usuario@ pc:~$ sudo gedit ~/backend.ldif
```

En el que copiaremos las siguientes líneas, no olvides cambiar las líneas con los campos `dc=amics`, `dc=local` y `olcrootPW` (contraseña del servidor `pass123`) por vuestros propios valores.

```
# Load dynamic backend modules
dn: cn=module,cn=config
objectClass: olcModuleList
cn: module
olcModulepath: /usr/lib/ldap
olcModuleload: back_hdb

# Database settings
dn: olcDatabase=hdb,cn=config
objectClass: olcDatabaseConfig
```

```
objectClass: olcHdbConfig
olcDatabase: {1}hdb
olcSuffix: dc=amics,dc=local
olcDbDirectory: /var/lib/ldap
olcRootDN: cn=admin,dc=amics,dc=local
olcRootPW: pass123
olcDbConfig: set_cachesize 0 2097152 0
olcDbConfig: set_lk_max_objects 1500
olcDbConfig: set_lk_max_locks 1500
olcDbConfig: set_lk_max_lockers 1500
olcDbIndex: objectClass eq
olcLastMod: TRUE
olcDbCheckpoint: 512 30
olcAccess: to attrs=userPassword by dn="cn=admin,dc=amics,dc=local" write by anonymous
auth by self write by * none
olcAccess: to attrs=shadowLastChange by self write by * read
olcAccess: to dn.base="" by * read
olcAccess: to * by dn="cn=admin,dc=amics,dc=local" write by * read
```

Ahora añadiremos el **LDIF** al directorio:

```
usuario@ pc:~$ sudo ldapadd -Y EXTERNAL -H ldapi:/// -f backend.ldif
```

## Configurando SAMBA

Ahora haremos una copia de seguridad del fichero de configuración de **SAMBA** **/etc/samba/smb.conf**.

```
usuario@ pc:~$ sudo cp /etc/samba/smb.conf smb.conf.old
```

Y editamos **smb.conf** para que quede de la siguiente manera:

```
usuario@ pc:~$ sudo gedit /etc/samba/smb.conf
```

Ojo debemos adecuar las líneas a la configuración particular de nuestro servidor:  
Cambiamos las líneas correspondientes a "WORKGROUP", "netbios name", "ldap suffix = dc=**amics**,dc=**local**", "ldap admin dn = cn=**admin**","dc=**amics**,dc=**local**".

```
#####
###Fichero smb.conf
#####
[global]
    # Domain name ..
    workgroup = amics
```

```
# Server name - as seen by Windows PCs ..
netbios name = samba-amics
# Be a PDC ..
domain logons = Yes
domain master = Yes
# Be a WINS server ..
wins support = true

obey pam restrictions = Yes
dns proxy = No
os level = 35
log file = /var/log/samba/log.%m
max log size = 1000
syslog = 0
panic action = /usr/share/samba/panic-action %d
pam password change = Yes

# Allows users on WinXP PCs to change their password when they press Ctrl-Alt-Del
unix password sync = no
ldap passwd sync = yes

# Printing from PCs will go via CUPS ..
load printers = yes
printing = cups
printcap name = cups

# Use LDAP for Samba user accounts and groups ..
passdb backend = ldapsam:ldap://localhost

# This must match init.ldif ..
ldap suffix = dc=amics,dc=local
# The password for cn=admin MUST be stored in /etc/samba/secrets.tdb
# This is done by running 'sudo smbpasswd -w'.
ldap admin dn = cn=admin,dc=amics,dc=local

# 4 OUs that Samba uses when creating user accounts, computer accounts, etc.
# (Because we are using smbldap-tools, call them 'Users', 'Computers', etc.)
ldap machine suffix = ou=Computers
ldap user suffix = ou=Users
ldap group suffix = ou=Groups
ldap idmap suffix = ou=Idmap
# Samba and LDAP server are on the same server in this example.
ldap ssl = no

# Scripts for Samba to use if it creates users, groups, etc.
```

```
add user script = /usr/sbin/smbldap-useradd -m '%u'
delete user script = /usr/sbin/smbldap-userdel %u
add group script = /usr/sbin/smbldap-groupadd -p '%g'
delete group script = /usr/sbin/smbldap-groupdel '%g'
add user to group script = /usr/sbin/smbldap-groupmod -m '%u' '%g'
delete user from group script = /usr/sbin/smbldap-groupmod -x '%u' '%g'
set primary group script = /usr/sbin/smbldap-usermod -g '%g' '%u'
```

```
# Script that Samba users when a PC joins the domain ..
# (when changing 'Computer Properties' on the PC)
add machine script = /usr/sbin/smbldap-useradd -w '%u'
```

```
# Values used when a new user is created ..
# (Note: '%L' does not work properly with smbldap-tools 0.9.4-1)
logon drive =
logon home =
logon path =
logon script = allusers.bat
```

```
# This is required for Windows XP client ..
server signing = auto
server schannel = Auto
```

#### [homes]

```
comment = Home Directories
valid users = %S
read only = No
browseable = No
```

#### [netlogon]

```
comment = Network Logon Service
path = /var/lib/samba/netlogon
admin users = root
guest ok = Yes
browseable = No
logon script = allusers.bat
```

#### [Profiles]

```
comment = Roaming Profile Share
# would probably change this to elsewhere in a production system ..
path = /var/lib/samba/profiles
read only = No
profile acls = Yes
browsable = No
```

```
[printers]
comment = All Printers
path = /var/spool/samba
use client driver = Yes
create mask = 0600
guest ok = Yes
printable = Yes
browseable = No
public = yes
writable = yes
admin users = root
write list = root

[print$]
comment = Printer Drivers Share
path = /var/lib/samba/printers
write list = root
create mask = 0664
directory mask = 0775
admin users = root

[shared]
writeable = yes
path = /var/lib/samba/shared
public = yes
browseable = yes

[archive]
path = /exports/archive
browseable = yes
create mask = 755
directory mask = 755
read only = no
```

```
#####
###FIN DEL FICHERO
#####
```

**SAMBA** necesita saber el password de administrador de **LDAP** (**pass123**)

```
usuario@ pc:~$ sudo smbpasswd -W
```



Y ahora reiniciamos el servicio **SAMBA**:

```
usuario@ pc:~$ sudo service smb restart
```

Ya podemos probar si funciona con el comando **smb-client** (cuando nos pida una contraseña simplemente le damos a ENTER).

```
usuario@ pc:~$ sudo smbclient -L localhost
```

Deberíamos ver algo parecido a las siguientes líneas;

```
Enter root's password:
Anonymous login successful
Domain=[amics] OS=[Unix] Server=[Samba 3.4.7]

Sharename Type Comment
-----
print$ Disk Printer Drivers Share
shared Disk
archive Disk
IPC$ IPC IPC Service (Samba 3.4.7)
Anonymous login successful
Domain=[amics] OS=[Unix] Server=[Samba 3.4.7]

Server Comment
-----
amics Samba 3.4.7

Workgroup Master
-----
amics amics
```

Ahora crearemos los directorios necesarios de perfiles y netlogon para nuestros usuarios **SAMBA**:

```
usuario@ pc:~$ sudo mkdir -v -m 777 /var/lib/samba/profiles
usuario@ pc:~$ sudo mkdir -v -p -m 777 /var/lib/samba/netlogon
```

Vamos a añadir los esquemas **SAMBA** a nuestro servidor **LDAP**. Estos esquemas se encuentran aparte, en el paquete **samba-doc** ya instalado:

```
usuario@ pc:~$ sudo cp /usr/share/doc/samba-doc/examples/LDAP/samba.schema.gz
/etc/ldap/schema/
usuario@ pc:~$ sudo gzip -d /etc/ldap/schema/samba.schema.gz
```

Ahora los convertiremos al formato "**ldif**" antes de usarlos.

Crearemos un Nuevo fichero llamado **schema\_convert.conf**:

```
usuario@ pc:~$ sudo gedit ~/schema_convert.conf
```

Y pegaremos en él las siguientes líneas:

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

Ahora usaremos la orden **slapcat** para convertir los esquemas:

```
usuario@ pc:~$ sudo slapcat -f ~/schema_convert.conf -F ~ -n0 -s
"cn={12}samba,cn=schema,cn=config" > ~/cn=samba.ldif
```

**slapcat** generara el fichero "**~/cn\samba.ldif**". Lo editamos y cambiamos los siguientes atributos:

```
usuario@ pc:~$ sudo gedit ~/cn\samba.ldif
```

Líneas a cambiar, al principio del documento:

```
dn: cn={12}samba,cn=schema,cn=config
...
cn: {12}samba
```

Por:

```
dn: cn=samba,cn=schema,cn=config
...
cn: samba
```

Y finalmente borraremos las últimas líneas del fichero donde pone:

```
structuralObjectClass: olcSchemaConfig
entryUUID: 99e797a8-07cb-102f-8c5c-739a8467e607
creatorsName: cn=config
createTimestamp: 20100609043122Z
entryCSN: 20100609043122.188753Z#000000#000#000000
modifiersName: cn=config
modifyTimestamp: 20100609043122Z
```

Añadimos el esquema al servidor:

```
usuario@ pc:~$ sudo ldapadd -Y EXTERNAL -H ldapi:/// -f ~/cn\samba.ldif
```

Esto devolverá las siguientes líneas sin errores:

```
adding new entry "cn=samba,cn=schema,cn=config"
```

Vamos a comprobar que todo funciona realizando la siguiente consulta:

```
usuario@ pc:~$ sudo ldapsearch -Y EXTERNAL -H ldapi:/// -D cn=admin,cn=config -b cn=config -W olcDatabase={1}hdb
```

Deberíamos observar estas dos líneas al final de la salida como resultado de la consulta:

```
# search result
search: 2
result: 0 Success

# numResponses: 2
# numEntries: 1
```

Si observamos una respuesta como la anterior, es que nuestro servidor **LDAP** está funcionando, pero todavía debemos configurar un par de cosas.

Descomprimos el paquete **smbldap-tools** (inicialmente descargado):

```
usuario@ pc:~$ sudo gzip -d /usr/share/doc/smbldap-tools/configure.pl.gz
```

Ahora ejecutaremos un script de **perl** que nos ayudara a configurar **SAMBA**. Para casi todas las preguntas pulsaremos ENTER dejando la opción por defecto, menos cuando pregunte el "logon home" y el "logon path" que escribiremos "." (un punto) y pulsaremos ENTER. También nos fijaremos cuando pregunte por el password (ldap master bind password y ldap slave bind password), en cuyo caso escribiremos el password (**pass123**) de administrador de nuestro

servidor ya escrito en anteriores pasos. Por favor deja las demás opciones por defecto pulsando la tecla ENTER.

Ejecución del script de configuración:

```
usuario@ pc:~$ sudo perl /usr/share/doc/smbldap-tools/configure.pl
```

Ahora que el script ha creado la configuración, podemos usarlo para crear la estructura del servidor (populate):

```
usuario@ pc:~$ sudo smbldap-populate
```

Retoques finales:

```
usuario@ pc:~$ sudo /etc/init.d/slapd stop
usuario@ pc:~$ sudo slapindex
usuario@ pc:~$ sudo chown openldap:openldap /var/lib/ldap/*
usuario@ pc:~$ sudo /etc/init.d/slapd start
```

Haz admin o tu usuario administrador, el administrador del Controlador de Dominio:

```
usuario@ pc:~$ sudo smbldap-groupmod -m 'administrador' 'Administrators'
```

Este comando debería devolver:

```
adding user administrador to group Administrators with no errors then you are looking good!
```

Ahora debemos habilitar la opción para que los clientes se autentiquen contra el servidor **LDAP**. Para ello necesitamos instalar el siguiente paquete:

```
usuario@ pc:~$ sudo apt-get --yes install ldap-auth-client
```

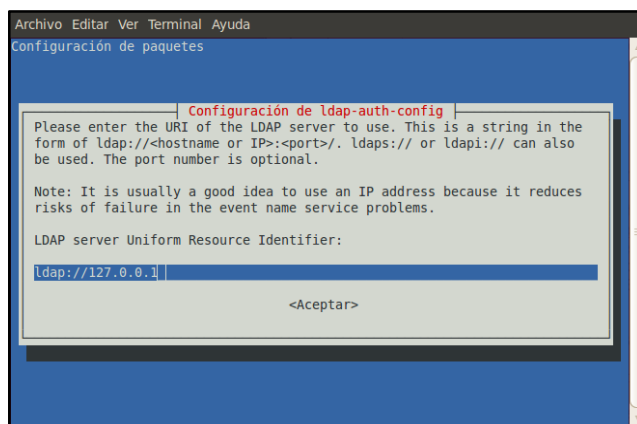
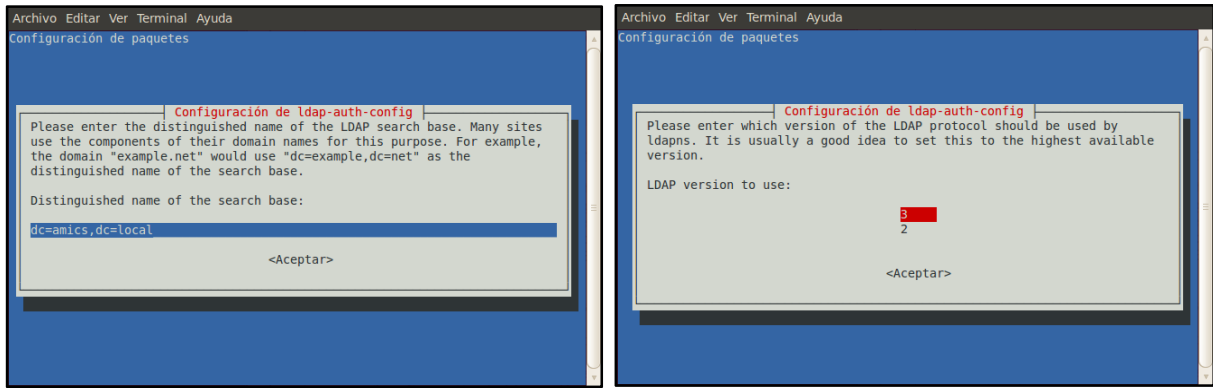
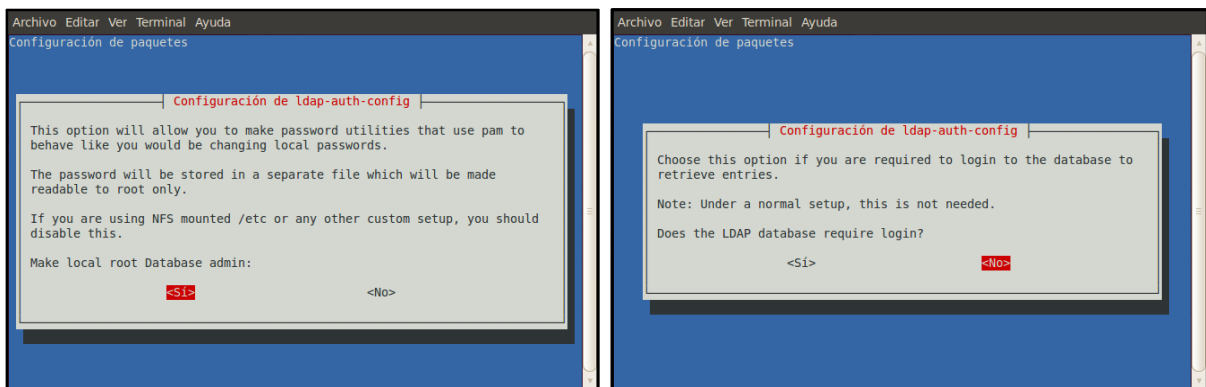


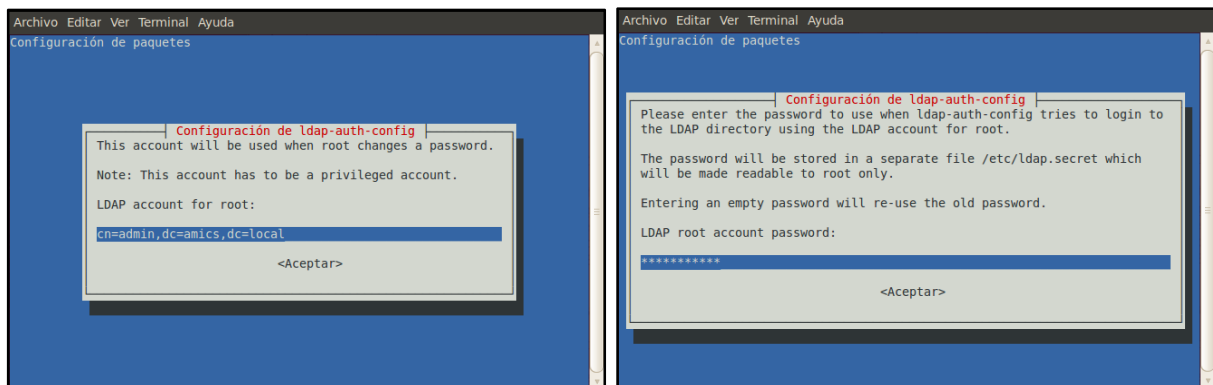
Figura 8: LDAP server Uniform Resource Identifier.



Figuras 9 y 10: Nombre de nuestro dominio y versión del protocolo LDAP.



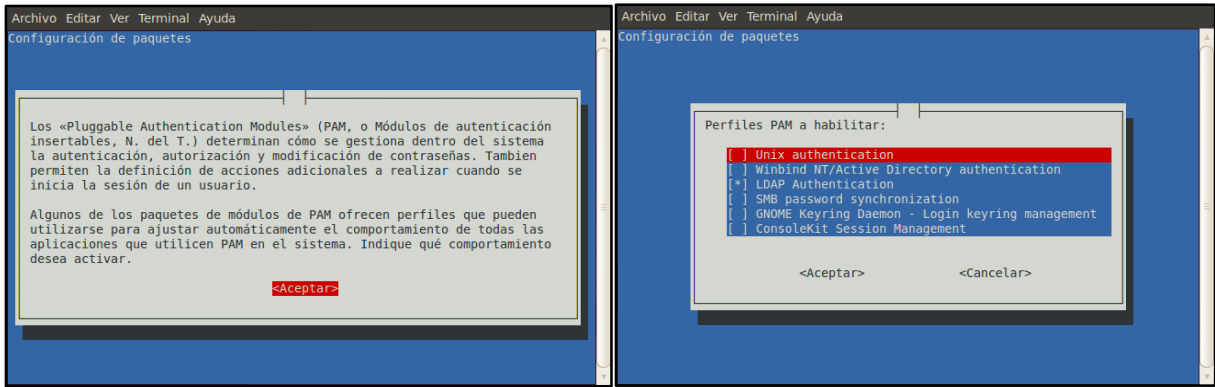
Figuras 11 y 12: Hacer el usuario root local administrador de la Base de Datos y requerimiento de login en la base de datos LDAP.



Figuras 13 y 14: Cuenta con privilegios de LDAP y password.

También debemos decirle a **PAM** y al **Name Service Switch** que use **LDAP** para la autenticación:

```
usuario@ pc:~$ sudo auth-client-config -t nss -p lac_ldap
usuario@ pc:~$ sudo pam-auth-update ldap
```



*Figuras 15 y 16: Módulos de autenticación PAM a activar.*

Si todo ha salido bien, ya puedes agregar un usuario a la base de datos:

```
usuario@ pc:~$ sudo smbldap-useradd -a -m -P usuariodeprueba
```

Podemos comprobar que se ha añadido el usuario con el siguiente comando:

```
usuario@ pc:~$ sudo ldapsearch -xLLL -b "dc=amics,dc=local" uid=usuariodeprueba
```

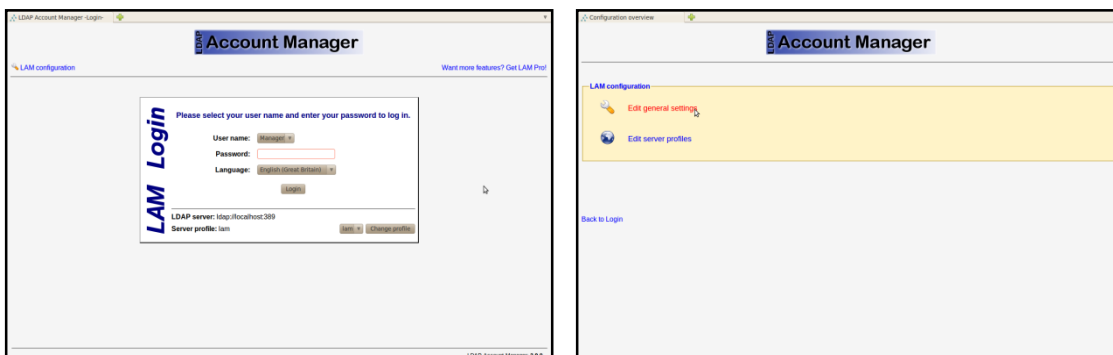
## 4. Instalación y configuración de LAM.

**LAM** es un frontend web para la administración de cuentas de usuarios **UNIX** y **SAMBA** dentro de un directorio **LDAP**. Para instalarlo teclearemos en una terminal lo siguiente:

```
usuario@ pc:~$ sudo apt-get install ldap-account-manager
```

Una vez instalado para acceder a la interfaz de configuración abriremos un webbrowser y escribiremos: <http://localhost/lam>

Nada mas abrir la página de login pulsaremos sobre la opción “**LAM Configuration**”. En la siguiente página accedemos a “**Edit general settings**”, aquí nos pedirá la contraseña maestra que por defecto es “**lam**”, la introducimos y al acceder al siguiente menu, si queremos podemos cambiarla en la opción que se encuentra en el final de la página. Una vez hecho esto vamos al menu anterior y seleccionamos la opción “**Edit server profiles**”.



Figuras 17 y 18: Pantalla de login servidor LDAP y acceso a la configuración básica del LAM.

En la pantalla de “**Edit server profiles**” debemos comprobar los datos para que coincidan con los de nuestro servidor, tanto los referentes a la configuración general como a los de tipos de cuentas. Y al final de la primera página debemos poner los datos de nuestro usuario root del **LDAP** “**cn=admin,dc=amics,dc=local**” y su contraseña.

En la segunda pestaña debemos modificar los módulos y tipos de módulos pertenecientes a nuestro servidor **LDAP**. Con la configuración básica es suficiente, pero ojo con los atributos que ya hay por defecto, nosotros tenemos los grupos Users, Groups y Computers, es posible que los del **LAM** por defecto no sean esos, habrá que cambiar tanto el nombre de dominio como el de los grupos (Users por People, Computers por Machines...). Luego aceptamos los cambios y volvamos a la ventana inicial de login.

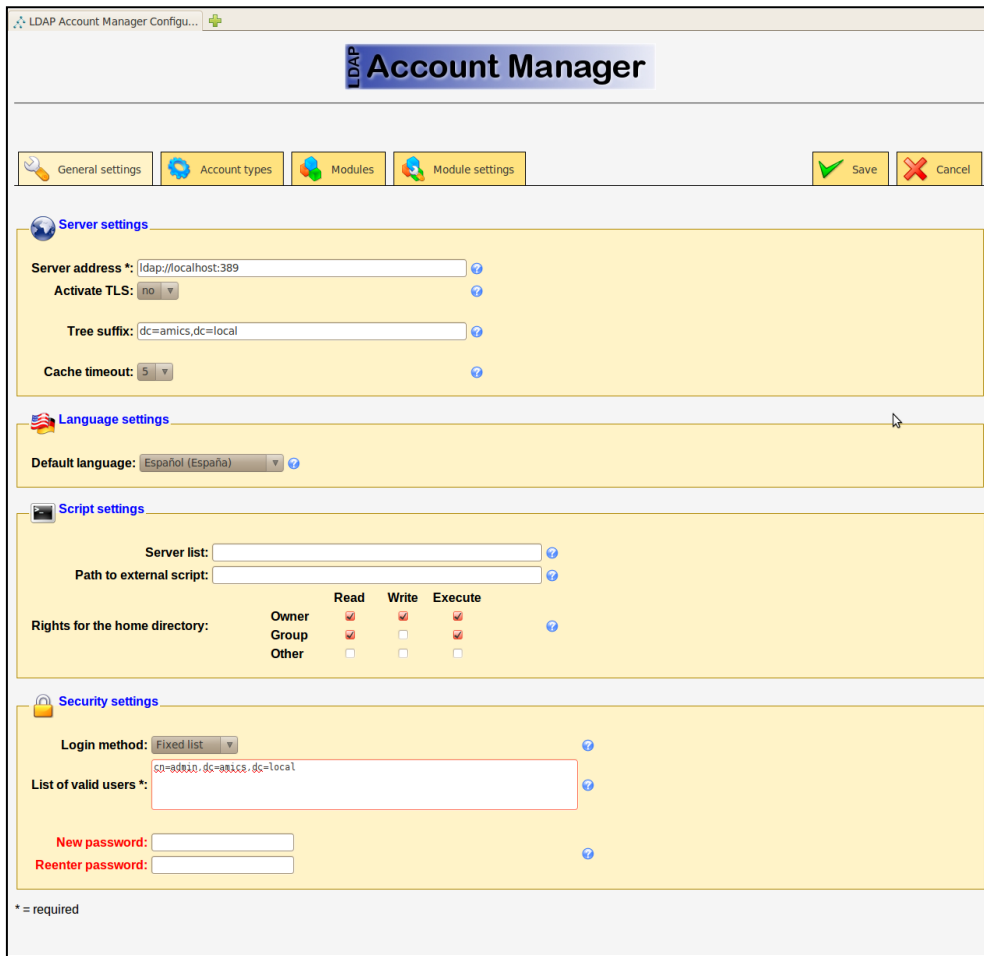


Figura 19: Propiedades del servidor LDAP – Configuración general.

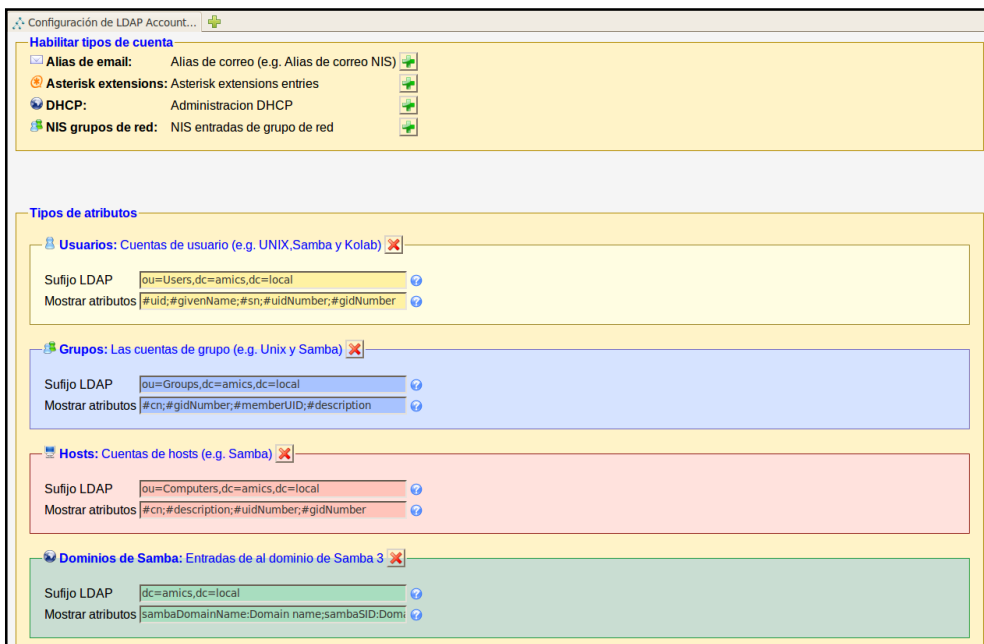


Figura 20: Propiedades del servidor LDAP – Tipos de cuenta.



Ahora iniciamos sesión con nuestro usuario admin y contraseña, que será el que le hemos dicho en la configuración anterior.

Desde este menú podemos ver el árbol de directorio, administrar los usuarios, los grupos, hosts y Dominios SAMBA.

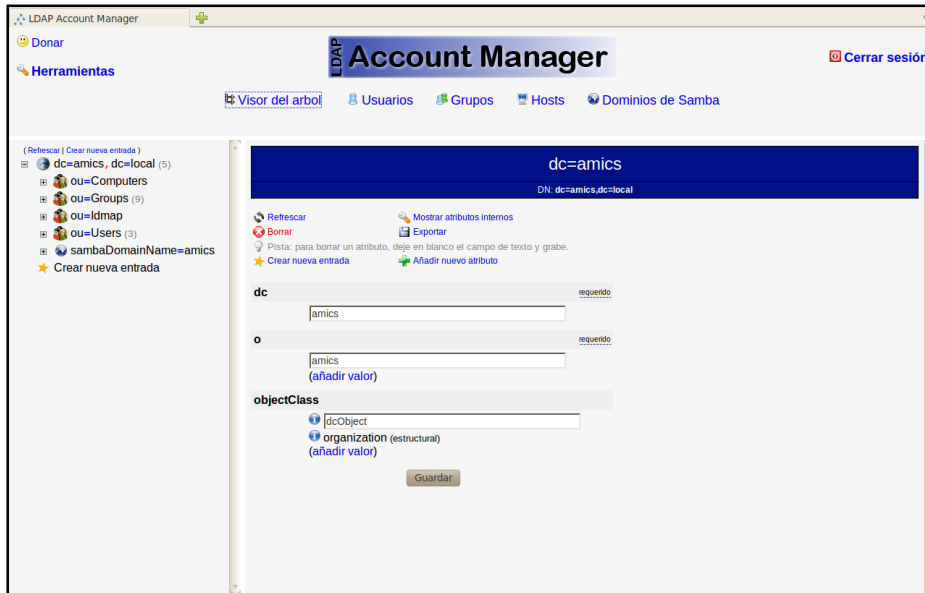


Figura 21: Árbol del directorio.

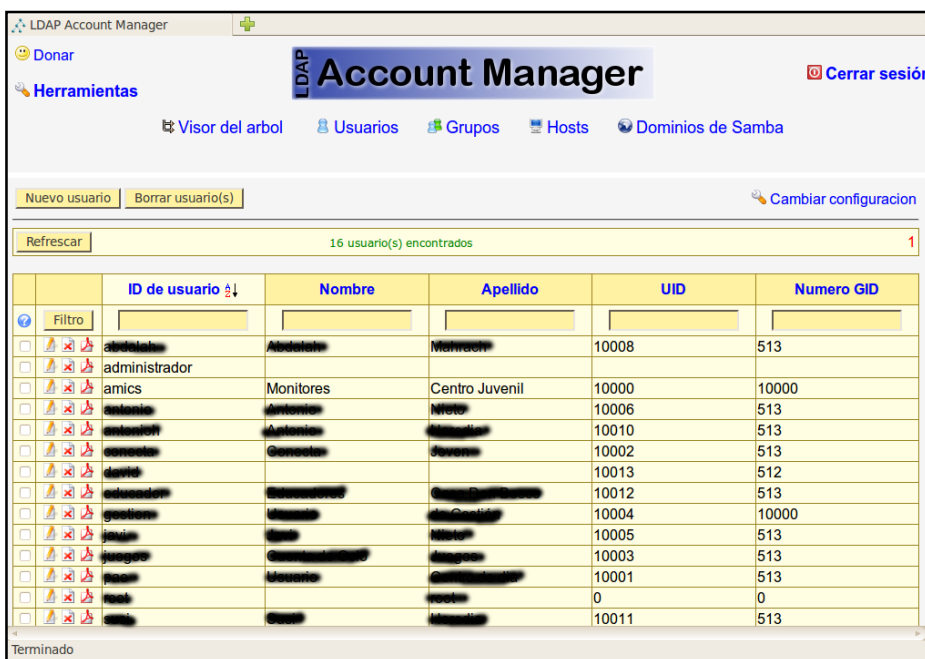


Figura 22: Administración de usuarios.

Acto seguido vamos a añadir un usuario a nuestro directorio. Debemos completar al menos las casillas que sean obligatorias \*. Tenemos tanto las opciones del usuario para **Linux** como para usuarios **Windows (SAMBA)**. No olvidemos establecer la contraseña del usuario. Si vamos a crear un usuario **Windows** como es este caso y queremos que al inicio de sesión lance un script para que monte una unidad de red por ejemplo, debemos tener en cuenta que el script será ubicado en “/var/lib/samba/netlogon”.

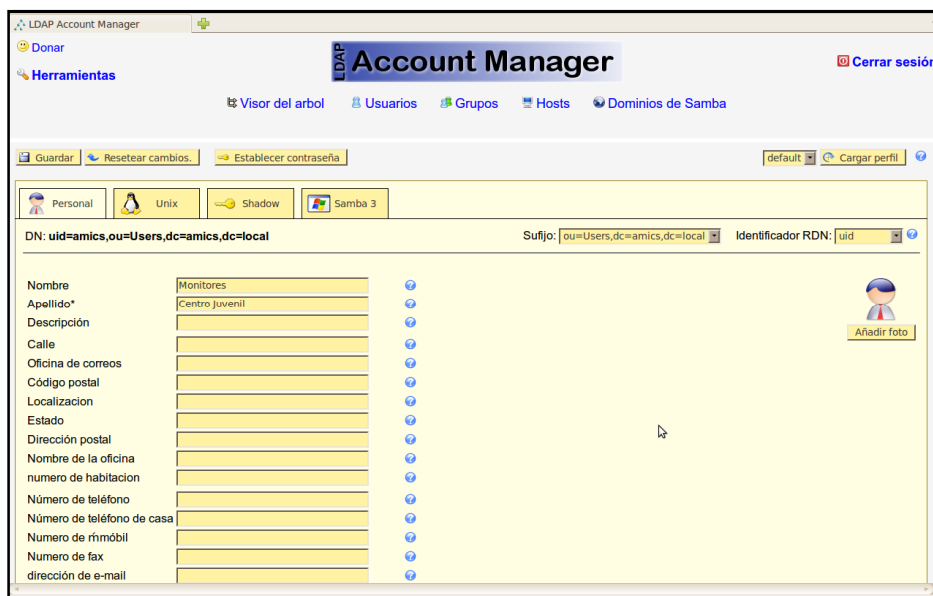
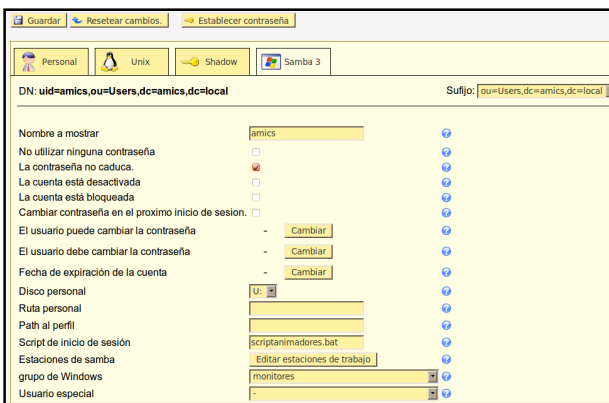
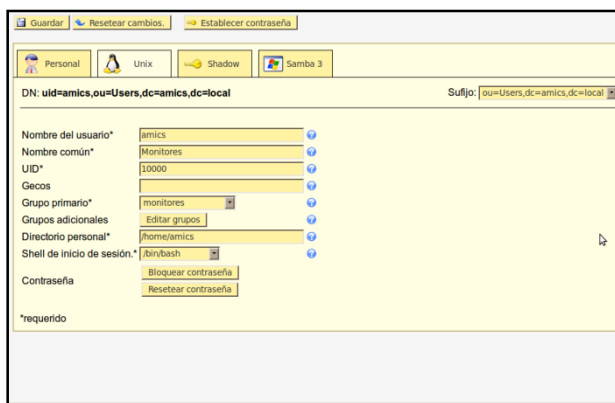


Figura 23: Nuevo usuario.



Figuras 24 y 25: Configuración de cuenta de usuarios Unix y Samba.

Con estos pasos ya podemos iniciar sesión con nuestro usuario “amics” en un cliente **Windows** unido al dominio, si configuramos la pestaña de **UNIX** también podrá autenticar desde clientes **Linux**.

## 5. Instalación y configuración de Webmin.

**Webmin** es una herramienta de configuración de sistemas accesible vía web para **OpenSolaris, GNU/Linux** y otros sistemas **UNIX**. Con él se pueden configurar aspectos internos de muchos sistemas operativos, como usuarios, cuotas de espacio, servicios, archivos de configuración, apagado del equipo, etcétera, así como modificar y controlar muchas aplicaciones libres, como el servidor web **Apache, PHP, MySQL, DNS, SAMBA, DHCP**, entre otros.

Para instalarlo en nuestro servidor **Ubuntu** basta con acceder a la página oficial y descargar el “Debian Package” que instalaremos.

<http://www.webmin.com/>

Una vez instalado para acceder a la interfaz de configuración accederemos a un webbrowser y escribiremos la dirección <https://localhost:10000> .

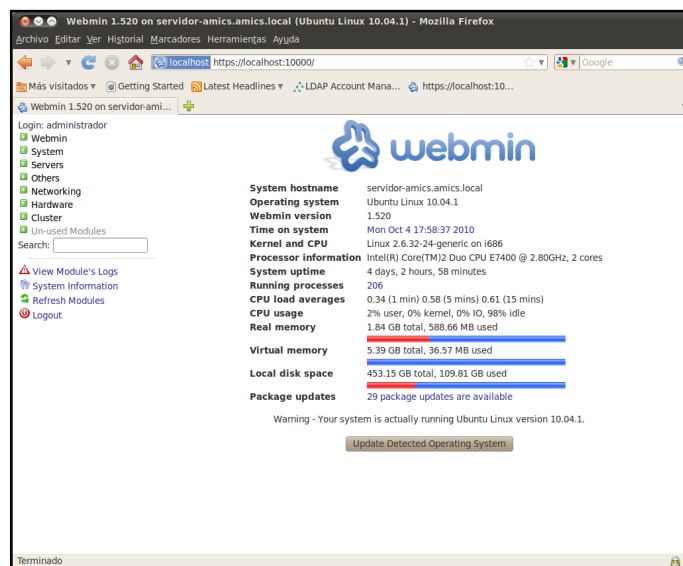


Figura 26: Interfaz gráfica Webmin.

En el apartado anterior hemos explicado cómo crear el usuario “amics” y que cargando un script de inicio de sesión montará en una unidad su recurso compartido. Ahora vamos a crear ese recurso compartido al que accederá el script del usuario.

Para ello accederemos al menú “**Servers**” --> “**Samba Windows file Sharing**” y pulsaremos en la opción “**Create a new file share**”.

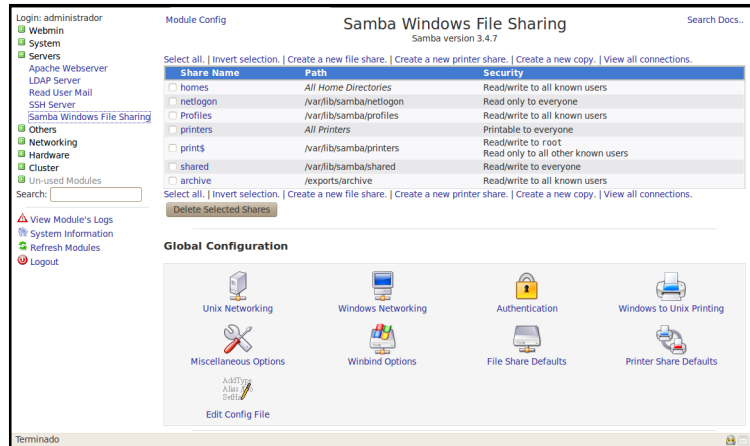


Figura 27: Menu Samba Windows file Sharing.

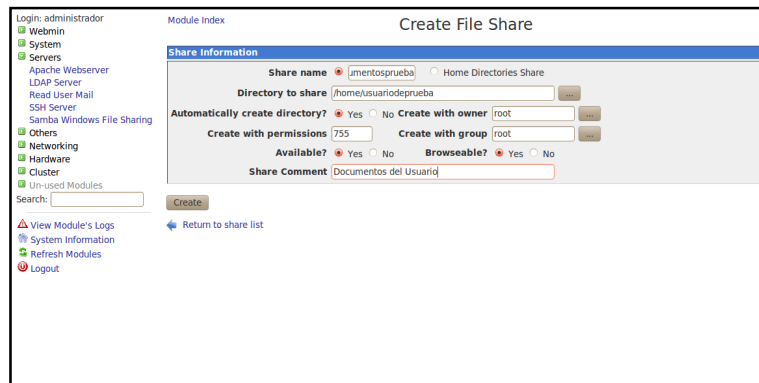


Figura 28: Creando el recurso compartido.

Desde esta sección podemos modificar la ruta del recurso compartido, la seguridad y el control del acceso al recurso, permisos, etc.

Ya para finalizar apuntaremos que en este mismo apartado **“Samba Windows file Sharing”** tenemos la opción de configurar la compartición de las impresoras y otros recursos.

## 6. Servidor DNS en Ubuntu 10.04

---

### *¿Qué es DNS?*

**Domain Name System** / Service (o **DNS**, en español: sistema de nombre de dominio) es un sistema de nomenclatura jerárquica para computadoras, servicios o cualquier recurso conectado a internet o a una red privada. Este sistema asocia información variada con nombres de dominios asignado a cada uno de los participantes. Su función más importante, es traducir (resolver) nombres inteligibles para los humanos en identificadores binarios asociados con los equipos conectados a la red, esto con el propósito de poder localizar y direccionar estos equipos mundialmente.

El **DNS** es una base de datos distribuida y jerárquica que almacena información asociada a nombres de dominio en redes como Internet. Aunque como base de datos el **DNS** es capaz de asociar diferentes tipos de información a cada nombre, los usos más comunes son la asignación de nombres de dominio a direcciones IP y la localización de los servidores de correo electrónico de cada dominio.

La asignación de nombres a direcciones IP es ciertamente la función más conocida de los protocolos **DNS**. Por ejemplo, si la dirección IP del sitio **FTP** de prox.mx es 200.64.128.4, la mayoría de la gente llega a este equipo especificando ftp.prox.mx y no la dirección IP. Además de ser más fácil de recordar, el nombre es más fiable. La dirección numérica podría cambiar por muchas razones, sin que tenga que cambiar el nombre.

### *Instalación del servidor de DNS BIND*

Como vamos a utilizar el servidor como controlador de dominio y fuente de autenticación, tiene sentido tener también algún servicio **DNS** disponible. Hay que tener en cuenta que si se disponen de varios servidores, lo recomendable es utilizar un servidor separado para las **DNS**, de tal forma que tengamos dos lugares donde resolver direcciones.

Instalar el servicio **bind9**:

```
usuario@ pc:~$ sudo apt-get install bind9
```

### *Configurar nuestra zona primaria de DNS usando WebMin*

Ahora se ha de crear la **zona DNS**. Usaremos una GUI (Graphic user interface, interfaz gráfica) ya instalada en puntos anteriores para hacer esto en lugar de hacerlo directamente modificando los archivos.

En un navegador web, abrimos la dirección: <https://localhost:10000>. Nos autenticamos con el nombre del usuario y la contraseña que pusimos al instalar el equipo (administrador, root).

“**Servidores > BIND DNS Server**” (es posible que aparezca en el menú del final “**Módulos no usados>Servidor de DNS BIND**”)

En el apartado de “**Zonas DNS Existentes**”, pulsamos sobre “**Crear zona raíz**”.

- Tipo de zona: Reenvío (nombres a direcciones) Forward (Names to addresses)
- Nombre de dominio / red: amics.local
- Archivo de registros: Automático
- Servidor maestro: servidor-amics.amics.local (aparece por defecto, poner el nombre que hayamos elegido en caso contrario)
- Dirección de correo: administrador@amics.local (dirección del administrador)

Pulsamos sobre “**Crear**”

Pulsamos sobre “**Aplicar cambios**”

Pulsamos sobre Dirección (0) en la parte superior.

Nombre: servidor-amics

Dirección: 192.158.1.250

Pulsamos “**Crear**” y después “**Regresar a Tipo de registros**”

Pulsamos sobre el botón “**Aplicar cambios**”.

### *Configuramos el servidor para usarse a sí mismo para resolver DNS*

En esta sección vamos a hacer que el archivo de configuración de resolución de nombres /etc/resolv.conf apunte a nosotros mismos, a nuestro servidor **DNS**. Es recomendable dejar un segundo servidor **DNS** para que en caso de error en nuestro **DNS** pueda seguir funcionando con normalidad.

Abrimos el archivo /etc/resolv.conf para editarlo:

```
usuario@ pc:~$ sudo gedit /etc/resolv.conf
```

Agregamos las siguientes líneas al principio del archivo:

```
search amics.local
servidor-amics 192.168.1.250
```

Reiniciar el servidor para asegurarnos que el **DNS** está funcionando correctamente.

```
usuario@ pc:~$ sudo reboot
```

## 7. Servidor DHCP en Ubuntu 10.04

### Que es DHCP ?

Un servidor Dynamic Host Configuration Protocol (**DHCP**) asigna dinámicamente las direcciones IP y otras configuraciones de una red determinada a otros ordenadores clientes que están conectados a la red. Esto simplifica la administración de la red y hace que la conexión de nuevos equipos a la red sea mucho más fácil.

Todas las direcciones IP de todos los equipos se almacenan en una base de datos que reside en un servidor.

Un servidor **DHCP** puede proporcionar los ajustes de configuración utilizando dos métodos:

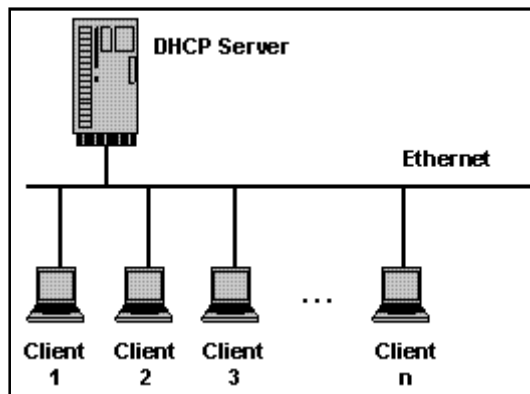


Figura 29.Red con servidor DHCP.

### RANGO DE DIRECCIONES

Este método se basa en la definición de un grupo de las direcciones IP para los clientes **DHCP** (también llamado IP address pool) que suministran sus propiedades de configuración de forma dinámica según lo soliciten los ordenadores cliente. Cuando un cliente **DHCP** ya no está en la red durante un período determinado, la configuración vence y la dirección IP del pool es puesta en libertad el uso de otros clientes **DHCP**.

### DIRECCIÓN MAC

Este método se basa en utilizar el protocolo **DHCP** para identificar la dirección de hardware única de cada tarjeta de red conectada a la red y luego es asignada una configuración constante así como la misma dirección IP cada vez que la configuración de **DHCP** del cliente realiza una petición al servidor **DHCP** desde el mismo dispositivo de red.

### *Instalar un servicio DHCP en Ubuntu y Debian*

Para instalar el servidor de asignación automática de dirección IP ejecutamos el comando:

```
usuario@ pc:~$ sudo apt-get install dhcp3-server
```

Este sencillo paso instala el servidor en nuestro **Linux**.

## Configurando el servidor DHCP

En el caso que tengan dos interfaces de red (NIC) en su servidor **Linux** tienen que seleccionar cual van a utilizar para escuchar las peticiones **DHCP**. Para configurar el servicio, editamos el archivo `/etc/default/dhcp3-server`, y cambiamos `INTERFACES="eth0"` por la tarjeta de red interna.

Es necesario hacer una copia de seguridad del archivo de configuración:

```
usuario@ pc:~$ sudo cp /etc/dhcp3/dhcpd.conf /etc/dhcp3/dhcpd.conf.back
```

## Configurar utilizando el método de rango de direcciones (IP pool)

Editamos la configuración tecleando:

```
usuario@ pc:~$ sudo vi /etc/dhcp3/dhcpd.conf
```

Y en este archivo cambiamos las siguientes secciones

```
default-lease-time 600;
max-lease-time 7200;

option subnet-mask 255.255.255.0;
option broadcast-address 192.168.1.255;
option routers 192.168.1.250;
option domain-name-servers 192.168.1.250;
option domain-name "amics.local";

subnet 192.168.1.0 netmask 255.255.255.0 {
range 192.168.1.10 192.168.1.200;
}
```

Guardamos y salimos del archivo. El texto anterior configura el servidor DHCP con los siguientes parámetros:

- Asignación a los clientes direcciones IPs del rango de 192.168.1.10 hasta 192.168.1.200
- Prestara la dirección IP por un mínimo de 600 segundos, y como máximo permitido de 7200 segundos.
- Determina la máscara de subred a 255.255.255.0
- Dirección de broadcast de 192.168.1.255
- Como **gateway/pasarela de red/router** la dirección 192.168.1.250
- Y el servidor 192.168.1.250 como su servidor DNS



## Configurar utilizando el método de direcciones MAC

Con este método se puede reservar algunas o todas las direcciones IP de nuestra red para determinadas maquinas. Como podrán ver la configuración es muy parecida a la anterior, con la salvedad que para reservar la asignación de una IP a una determinada NIC (network card interface) debemos de utilizar la etiqueta host

```
default-lease-time 600;
max-lease-time 7200;

option subnet-mask 255.255.255.0;
option broadcast-address 192.168.1.255;
option routers 192.168.1.250;
option domain-name-servers 192.168.1.250;
option domain-name "amics.local";

subnet 192.168.1.0 netmask 255.255.255.0 {
range 192.168.1.10 192.168.1.200;
}

host oracle{
hardware ethernet 00:03:47:31:e1:7f;
fixed-address 192.168.1.250;
}

host printer {
hardware ethernet 00:03:47:31:e1:b0;
fixed-address 192.168.1.21;
}
```

Ahora definiremos cual es la tarjeta de red que utilizaremos para dar el servicio **DHCP** (eth0), editando el archivo **dhcp3-server** y Completaremos INTERFACES="" " con INTERFACES="eth0"

```
usuario@ pc:~$ sudo gedit /etc/default/dhcp3-server
```

Para compartir el Internet Utilizaremos **IPTABLES**, para ello crearemos un archivo iptables.conf

```
usuario@ pc:~$ sudo gedit /etc/init.d/iptables.conf
```

Y añadimos las siguientes líneas:

```
iptables -F
iptables -X
iptables -Z
iptables -t nat -F
```

```
iptables -P INPUT ACCEPT
iptables -P OUTPUT ACCEPT
iptables -P FORWARD ACCEPT
iptables -t nat -P PREROUTING ACCEPT
iptables -t nat -P POSTROUTING ACCEPT
iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -o eth0 -j MASQUERADE (hace la magia)
iptables -A INPUT -s 192.168.1.0/24 -i eth1 -j ACCEPT

iptables -A INPUT -p udp -m udp --port 10000:20000 -j ACCEPT (aceptar algun puerto)
iptables -A INPUT -p udp -m udp --dport 2727 -j ACCEPT

echo 1 > /proc/sys/net/ipv4/ip_forward
```

Asignamos permisos al archivo

```
usuario@ pc:~$ sudo chmod 775 /etc/init.d/iptables.conf
```

Instalamos **rcconf**, que es una utilidad de **Debian** para configurar los servicios que se inician al boot-time

```
usuario@ pc:~$ sudo apt-get install rcconf
```

Ejecutamos **rcconf** y marcamos iptables.conf para que se inicie cuando carga el sistema

```
usuario@ pc:~$ sudo rcconf
```

Por último iniciamos todos los servicios

```
usuario@ pc:~$ sudo /etc/init.d/networking restart
usuario@ pc:~$ sudo /etc/init.d/dhcp3-server start
usuario@ pc:~$ sudo /etc/init.d/iptables.cf start
```

## *Configurar el cliente DHCP en Linux Ubuntu*

Si queremos configurar un escritorio o maquina con **Linux** como cliente **DHCP** seguimos los siguientes pasos:

Editamos el archivo de interfaces de red

```
usuario@ pc:~$ sudo gedit /etc/network/interfaces
```

Debemos de tener las siguientes líneas, tomando en cuenta que eth0 es un ejemplo

```
auto lo eth0
```

```
iface eth0 inet dhcp
iface lo inet loopback
```

Salvamos y salimos del archivo

Reiniciamos los servicios de red de **Linux Ubuntu**

```
usuario@ pc:~$ sudo /etc/init.d/networking restart
```

Para poder conocer las direcciones asignadas a las máquinas clientes tecleamos en un terminal:

```
usuario@ pc:~$ tail -n 15 /var/lib/dhcp3/dhclient.*.leases
```



## 8. Configuración de los terminales clientes

### Uniando Windows XP al dominio

Bien una vez en funcionamiento nuestro servidor vamos a comprobar que podemos autenticar usuarios desde máquinas **Windows XP**.

El primer paso será unirlo al dominio desde el menú **“Propiedades del sistema”**, accediendo a la pestaña **“Nombre de equipo”** pulsamos sobre el botón **“Id. De red”**. Seguimos el asistente relleno los formularios con los datos de nuestra red y cuando nos pregunte el usuario root de nuestro **LDAP**, lo tecleamos: **“administrador”** clave: **“pass123”**.

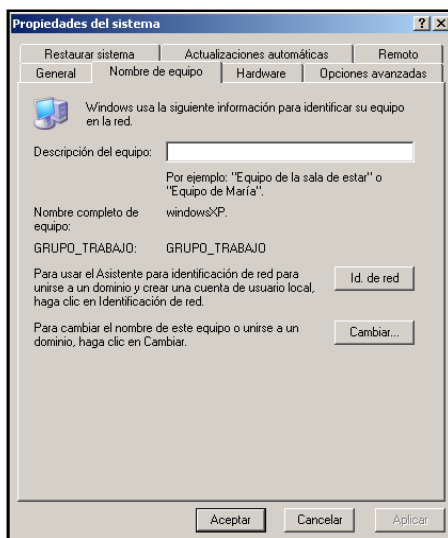


Figura 30: Propiedades del sistema.

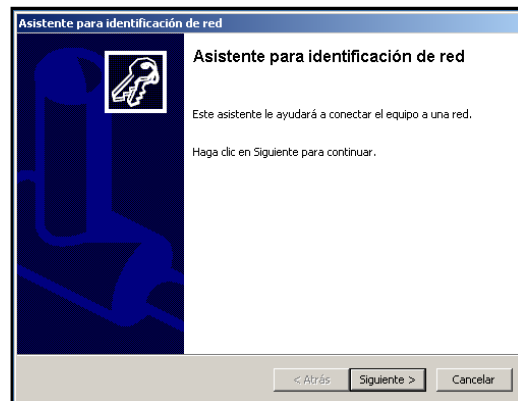


Figura 31: Asistente para unir a un dominio 1/7.

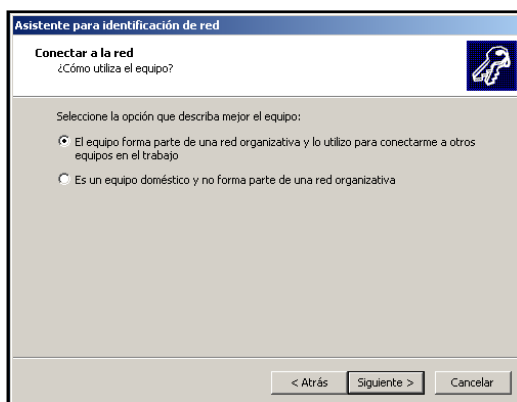


Figura 32: Asistente para unir a un dominio 2/7.

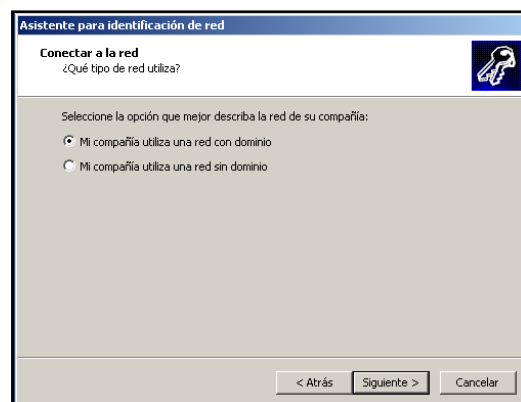


Figura 33: Asistente para unir a un dominio 3/7.

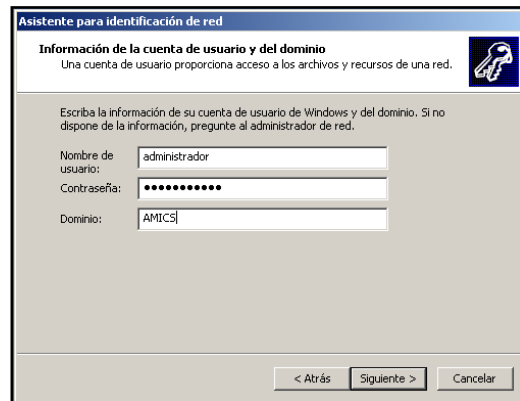
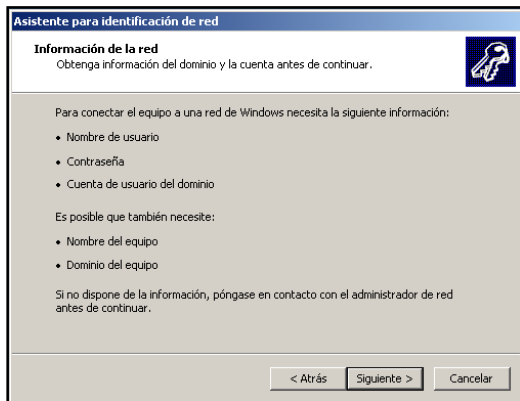


Figura 34: Asistente para unir a un dominio 4/7. Figura 35: Asistente para unir a un dominio 5/7.

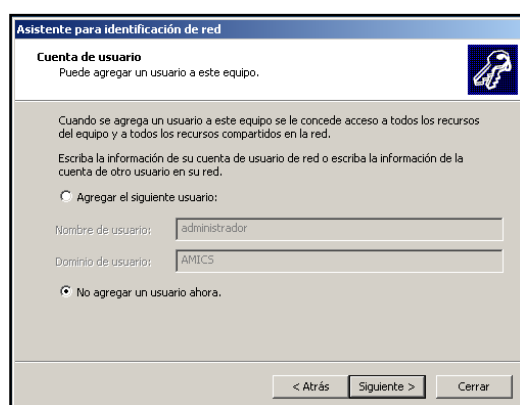
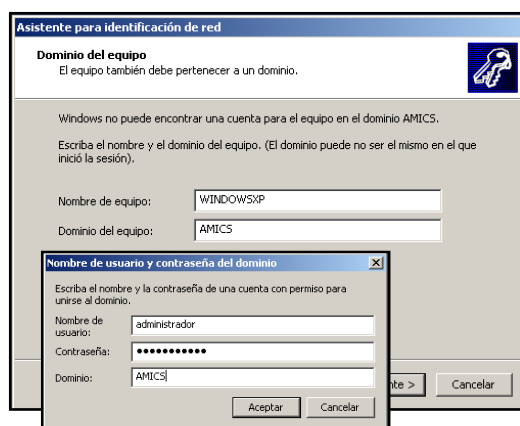


Figura 36: Asistente para unir a un dominio 6/7. Figura 37: Asistente para unir a un dominio 7/7.

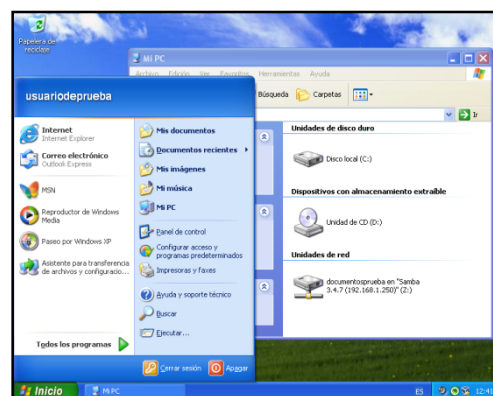


Figura 38: Iniciar sesión como usuario del dominio. Figura 39: Sesión de "usuariodeprueba".

Ahora solo falta añadir los usuarios del dominio al equipo para poder administrar sus privilegios. Para ello accederemos a cuentas de usuario, nos autenticaremos como administrador del equipo local y añadiremos los usuarios del dominio. Hay un ejemplo al final del apartado "Unir **Windows 7** al dominio".

## Uniando Windows 7 al dominio

Para unir un cliente de **Windows 7** al dominio deberemos modificar 2 campos del registro para que nos permita hacerlo. Si nos bajamos este fichero y lo ejecutamos ya podremos unir nuestro cliente al dominio sin problemas:

<https://bugzilla.samba.org/attachment.cgi?id=4988&action=view>

Una vez hecho esto accedemos a **“Propiedades del sistema”**, en el apartado **“Configuración de nombre, dominio y grupo de trabajo del equipo”** seleccionamos **“Cambiar configuración”** y como hemos hecho antes al unir un Cliente de **Windows XP** al dominio, mediante la pestaña **“Nombre de equipo”** pulsamos sobre el botón **“Id. De red”**. A lo largo del asistente nos preguntara el usuario root de nuestro **LDAP**, en nuestro caso: **“administrador”** clave: **“pass123”**.

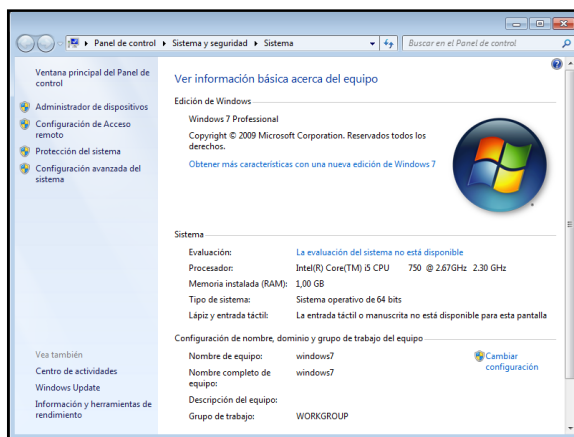


Figura 40: Propiedades del sistema.

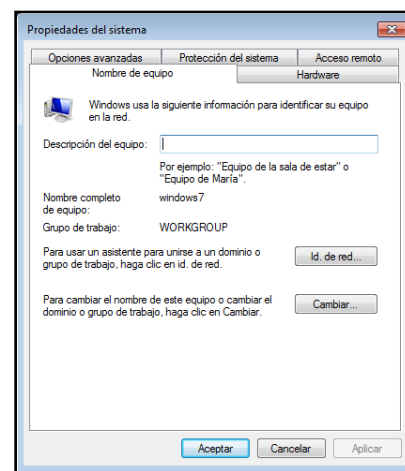


Figura 41: Nombre de equipo.

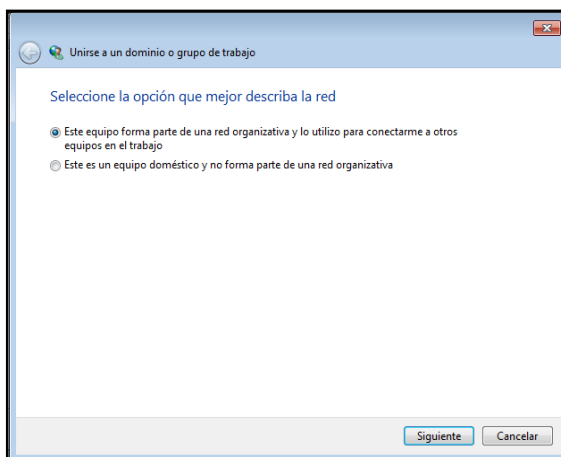


Figura 42: Asistente para unir a un dominio 1/7.

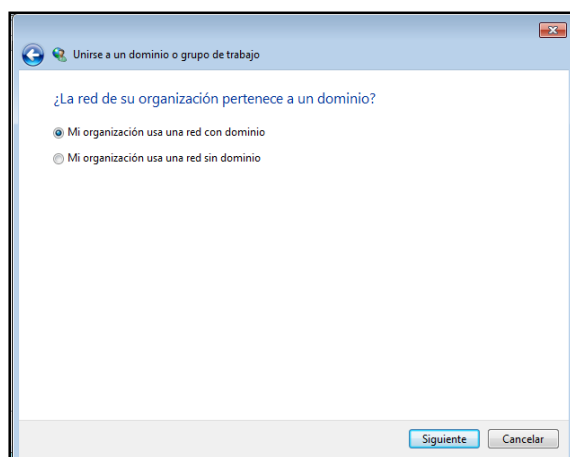


Figura 43: Asistente para unir a un dominio 2/7.

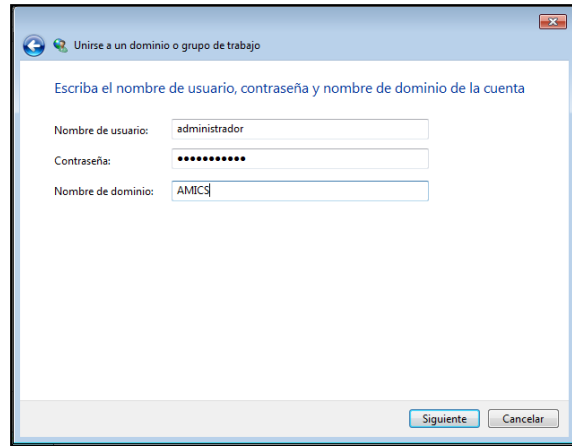
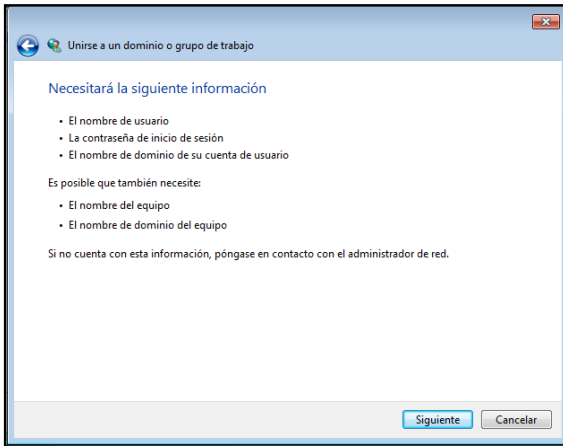


Figura 44: Asistente para unir a un dominio 3/7. Figura 45: Asistente para unir a un dominio 4/7.

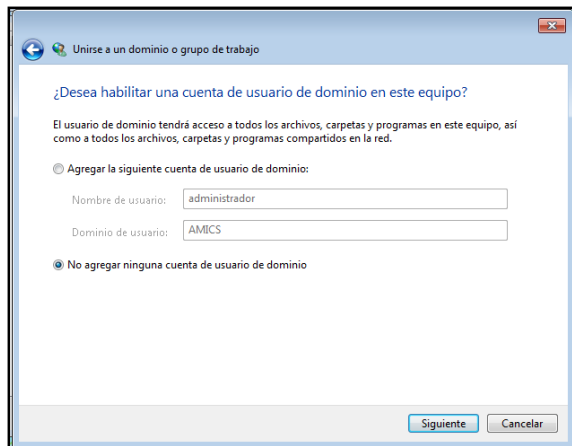
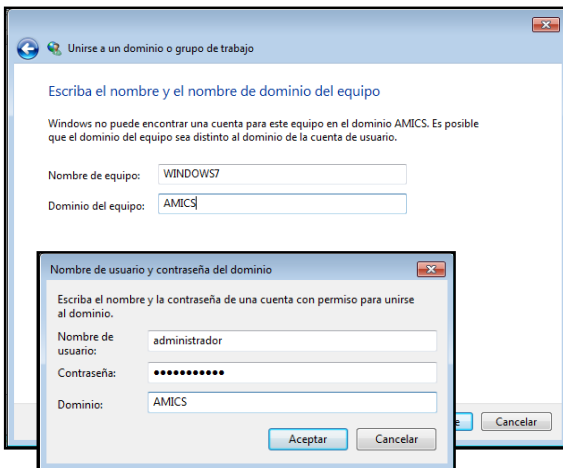


Figura 46: Asistente para unir a un dominio 5/7. Figura 47: Asistente para unir a un dominio 6/7.

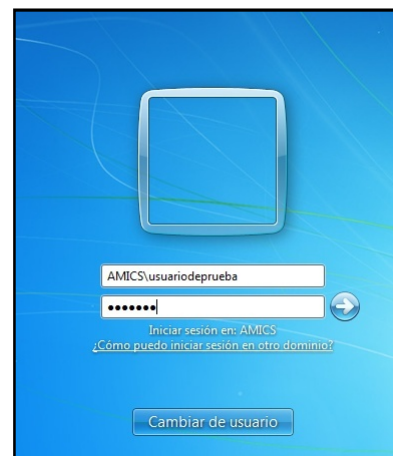
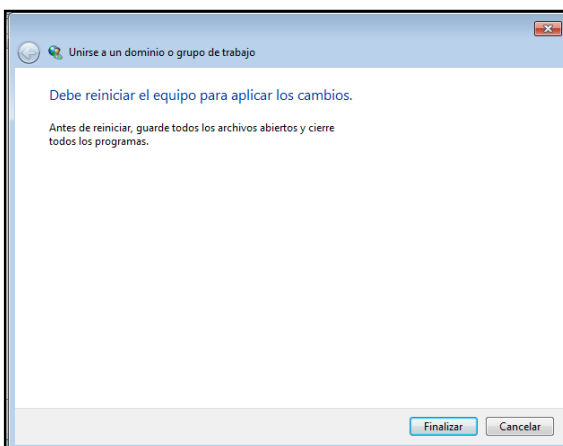


Figura 48: Asistente para unir a un dominio 7/7. Figura 49: Iniciar sesión como usuario del dominio.



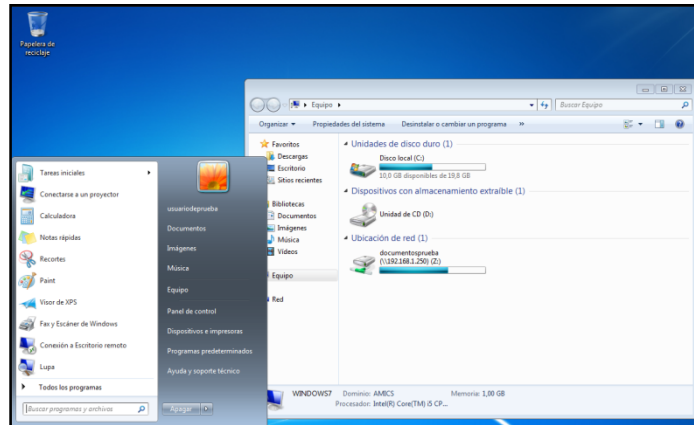


Figura 50: Sesión de "usuariodeprueba".

Para terminar si queremos administrar los permisos de algún usuario del dominio en el equipo cliente deberemos acceder al apartado de usuarios, para poder hacerlo nos pedirá el usuario y contraseña administrador del equipo local: (WINDOWS7\administrador pass: passlocal) y agregamos los usuarios otorgándole los permisos deseados.

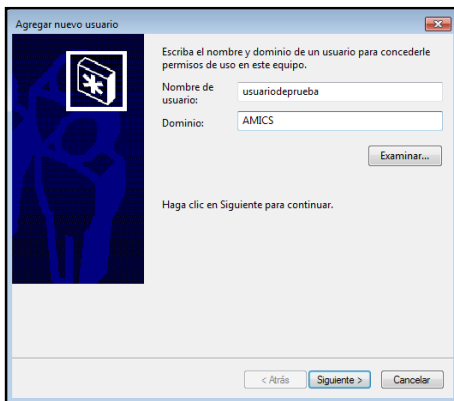


Figura 51: Agregar nuevo usuario.

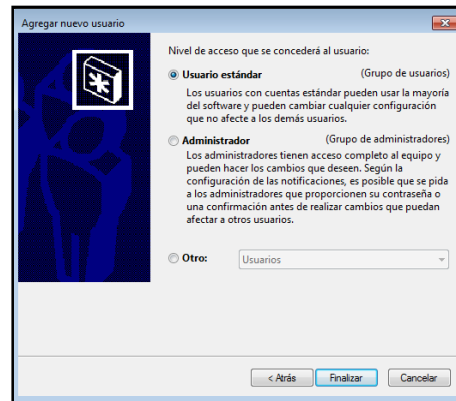


Figura 52: Agregar nuevo usuario.

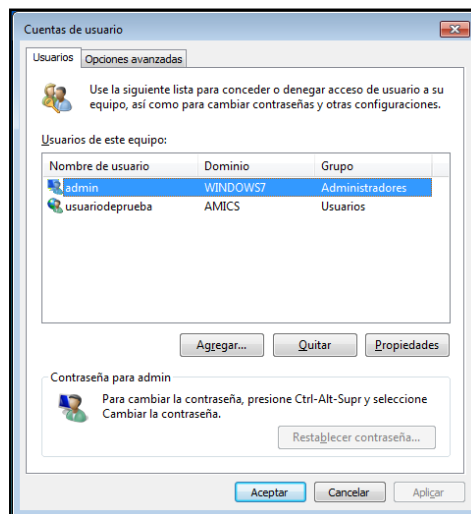


Figura 53: Cuentas de usuario.

### Uniando Ubuntu 10.04 al dominio

Lo primero instalar el software necesario para que funcione:

```
usuario@pc-cliente:~$ sudo apt-get install auth-client-config libpam-ldap libnss-ldap
```

Respondemos a las preguntas que nos realizará el proceso de instalación. (configuración de ldap-auth-config):

```
Should debconf manage LDAP configuration?: Yes (esto no lo ha preguntado en el ejemplo)
LDAP server Uniform Resource Identifier: ldap://192.168.1.250
Distinguished name of the search base: dc=amics,dc=local
LDAP version to use: 3
Make local root Database admin: Yes
Does the LDAP database require login? No
LDAP account for root: cn=admin,dc=amics,dc=local
LDAP root account password: pass123
```

Abrimos el archivo /etc/ldap.conf file para edición:

```
usuario@pc-cliente:~$ sudo gedit /etc/ldap.conf
```

Modificamos la información para que sea correcta de acuerdo con nuestra configuración:

```
host 192.168.1.250
base dc=amics,dc=local
uri ldap://192.168.1.250
rootbinddn cn=admin,dc=amics,dc=local
bind_policy soft
```

Copiamos el archivo /etc/ldap.conf a /etc/ldap/ldap.conf

```
usuario@pc-cliente:~$ sudo cp /etc/ldap.conf /etc/ldap/ldap.conf
```

Creamos un archivo nuevo /etc/auth-client-config/profile.d/open\_ldap:

```
usuario@pc-cliente:~$ sudo gedit /etc/auth-client-config/profile.d/open_ldap
```

Insertamos lo siguiente en el archivo vacío:

```
[open_ldap]
nss_passwd=passwd: compat ldap
nss_group=group: compat ldap
nss_netgroup=netgroup: compat ldap
nss_shadow=shadow: compat ldap
pam_auth=auth required pam_env.so
```

```

auth    sufficient pam_unix.so likeauth nullok
auth    sufficient pam_ldap.so use_first_pass
auth    required pam_deny.so
pam_account=account sufficient pam_unix.so
account sufficient pam_ldap.so
account required pam_deny.so
pam_password=password sufficient pam_unix.so nullok md5 shadow use_authok
password sufficient pam_ldap.so use_first_pass
password required pam_deny.so
pam_session=session required pam_limits.so
session required pam_mkhomedir.so skel=/etc/skel/
session required pam_unix.so
session optional pam_ldap.so

```

Realizamos una copia de seguridad de /etc/nsswitch.conf:

```

usuario@pc-cliente:~$ sudo cp /etc/nsswitch.conf /etc/nsswitch.conf.original

```

Realizamos una copia de seguridad de /etc/pam.d/:

```

usuario@pc-cliente:~$ cd /etc/pam.d/
usuario@pc-cliente:~$ sudo mkdir bkup
usuario@pc-cliente:~$ sudo cp * bkup/

```

Activamos el nuevo perfil de autenticación LDAP ejecutando el siguiente comando:

```

usuario@pc-cliente:~$ sudo auth-client-config -a -p open_ldap

```

Reiniciamos el servidor y probamos que podemos autenticarnos con el usuario creado (usuariodeprueba) .

```

usuario@pc-cliente:~$ sudo reboot

```

Ya podremos iniciar sesión como usuario del dominio. Si queremos que monte el home que hemos configurado para ser exportado en el servidor seguiremos los siguientes pasos:  
Instalación de los paquetes:

```

usuario@pc-cliente:~$ sudo apt-get install nfs-common

```

En la parte del cliente no se tiene que hacer nada más que montarla como un sistema de ficheros local, añadiendo la entrada correspondiente en el fichero “/etc/fstab”:

```

usuario@pc-cliente:~$ sudo gedit /etc/fstab

```

Ejemplo de entrada a añadir para montar la carpeta compartida como un fichero local:

```
192.168.1.250:/home /home nfs defaults 0 0
```

Donde indicamos:

- La dirección del servidor, seguida de dos puntos y el directorio compartido en el servidor.
- El directorio donde queremos montar la unidad.
- El tipo de sistema de ficheros. Siempre será **NFS**.
- Las opciones. Podemos añadir opciones para indicar que se monte como sólo lectura, o que no se puedan ejecutar los programas que se encuentren en esa partición. Las opciones son las mismas que los sistemas de ficheros locales.
- Dos 0

Observaciones

El home local no se borra sino que se esconde. Por este motivo, entre otros, recomendamos que en el equipo cliente sólo exista el usuario administrador.

Se recomienda que en los equipos donde se vaya a configurar este sistema sólo exista como usuario el administrador.

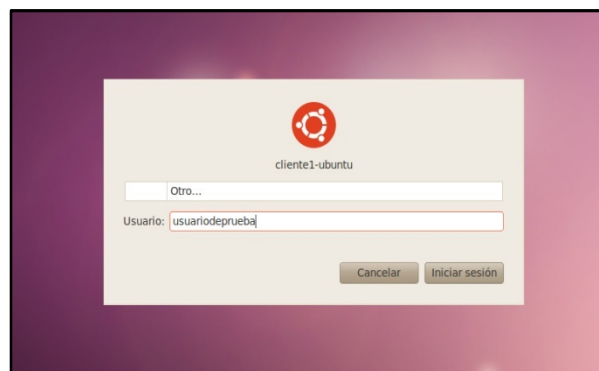


Figura 54: Iniciar sesión como usuario del dominio.

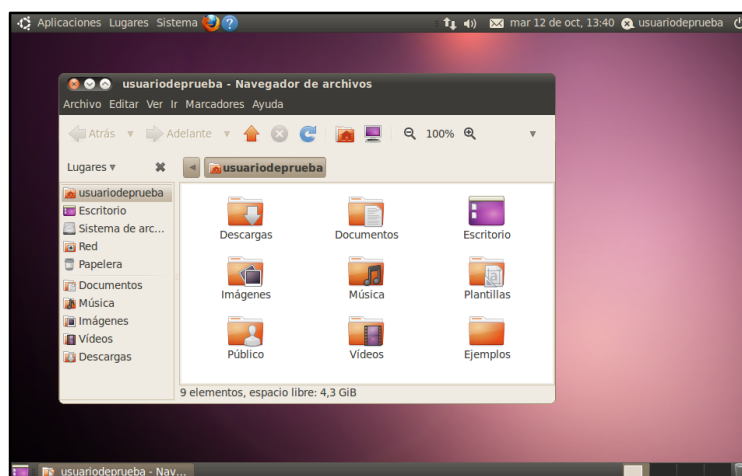


Figura 55: Sesión de “usuariodeprueba”.

## 9. Validación y Pruebas

---

En el apartado 8 se ha explicado y demostrado como unir clientes de **Windows** y **Linux** al dominio de nuestro servidor. Al final de cada apartado se ha incluido un pantallazo del momento en el que autentificamos con nuestro usuario del dominio en el terminal cliente. También en los clientes **Windows** se ha comprobado que ha conectado la unidad de recursos compartidos montada mediante **SAMBA** y gestionada por el **Webmin**.

Para poder montar los recursos compartidos como unidad de red en **Windows** simplemente se ha creado un script de inicio de sesión,--al que se ha hecho referencia en la gestión del usuario **SAMBA** con el **LAM--** y se ha guardado con permisos de ejecución en el directorio `"/var/lib/samba/netlogon/"` del servidor. El contenido del fichero utilizado como script de inicio de sesión para el usuario `usuariodeprueba` (`scriptprueba.bat`) es el siguiente:

```
NET USE Z: \\192.168.1.250\home\usuariodeprueba passusuario /USER:usuariodeprueba
```

NOTA: Debe tenerse en cuenta que a la hora de administrar el directorio compartido se deben tener permisos de lectura y escritura para la cuenta "usuariodeprueba", de lo contrario no lo montara correctamente y no se podrá leer o escribir en él.

Para los terminales **Linux** es más sencillo ya que se ha exportado el directorio `/home/` mediante NFS desde el servidor y se ha montado desde el `"fstab"` en los clientes **Linux**.



## 10. CONCLUSIONES

---

En este PFC se ha instalado y configurado:

1. Controlador de Dominio Primario en **Ubuntu 10.04 Lucid Lynx**.
2. Se ha instalado y configurado un servidor de **DNS**.
3. Se ha instalado y configurado un servidor **DHCP**.
4. Se han dado de alta usuarios, compartido recursos y accedido a éstos.

Finalmente, se ha verificado que el Controlador de Dominio funcione correctamente, que los usuarios puedan autenticarse contra éste y que los recursos compartidos sean accesibles desde clientes **Windows** y **Linux**. Estos recursos se montan sin problemas en los clientes.

Un cliente **Windows** se autentica contra el servidor y mediante un script de inicio de sesión previamente configurado, se puede acceder a una unidad de red compartida desde el servidor, podemos congelar el escritorio del usuario de modo que no se guarden documentos que se podrían perder en el escritorio, lanzar cualquier aplicación o aviso al inicio de sesión, etc.

De la misma forma el cliente **Linux** al loguear contra el servidor monta automáticamente la carpeta home previamente exportada por NFS en el servidor. Puede acceder a diversos recursos compartidos sin problemas. También puede lanzar aplicaciones en concreto mediante un script de inicio de sesión e incluso utilizar un congelador de escritorio comentado en el párrafo anterior.





## 11. Trabajo futuro

---

De cara al futuro podemos utilizar este PFC para administrar una red híbrida o pura de clientes **Windows** y/o **Linux**.

- En cuanto a cambios o mejoras a realizar podríamos añadir al servidor la autenticación segura de la capa transporte mediante **SSL**. De esta forma ciframos los datos si las transacciones u operaciones realizadas en la red lo requieren.
- Podemos instalar un servidor web instalando el paquete **apache2**.
- Conectando un punto de acceso inalámbrico podríamos servir una IP con conexión a internet y acceso al servidor de autenticación para terminales portátiles, ya que hemos instalado y configurado **DHCP** y **DNS**.
- Se puede gestionar también un servidor **FTP**.
- Configurar un servidor de Bases de Datos **MySQL**.
- Podríamos ejecutar aplicaciones **Windows** (emuladas) desde el propio servidor por terminal server.
- Podemos gestionar un **Proxy** con **squid** o un **Firewall** mediante **IPTABLES**.



## 12. Bibliografía

---

Documentación sobre implementación y pruebas:

- <http://es.wikipedia.org>
- <http://tuxnetworks.blogspot.com/2010/07/howto-samba-ldap-on-1004-lucid-short.html>
- [http://rockytophome.com/index.php?option=com\\_content&view=article&id=121:how-to-openldap-on-ubuntu-1004-updated&catid=38:computer-related&Itemid=70](http://rockytophome.com/index.php?option=com_content&view=article&id=121:how-to-openldap-on-ubuntu-1004-updated&catid=38:computer-related&Itemid=70)
- <http://es.tldp.org/COMO-INSFLUG/COMOs/LDAP-Linux-Como/>
- <http://tic.salesianos.edu>
- <http://www.guatewireless.org/os/linux/distros/debian/ubuntu/como-instalar-y-configurar-un-servidor-dhcp-en-linux-ubuntu-debian/>
- <http://cursoredlocal.wordpress.com/2010/05/12/instalacion-y-configuracion-de-servidor-dns-en-ubuntu-10-04/>
- <http://es.tldp.org/Manuales-LuCAS/doc-iptables-firewall/doc-iptables-firewall-html/>

Documentación sobre **Windows** y **Linux**:

- <http://www.muywindows.com/2010/01/18/en-servidores-windows-gana-a-linux-2-a-1>
- <http://www.muylinux.com/2010/03/03/%C2%BFquien-domina-en-servidores-windows-o-linux>
- <http://www.servidoresdedicados.com/>
- <http://www.techtear.com/2007/05/03/linux-vs-windows-uptime-y-velocidad-de-respuesta-en-servidores/>
- <http://www.kriptopolis.org/seguridad-en-servidores-windows-vs-linux>
- <http://www.tufuncion.com/windows-vs-linux>
- <http://www.entmexico.com/hosting/windows-o-linux.html>