

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
Gestor de contenidos (Alfresco)
Aplicaciones híbridas

Tareas programadas (Quartz)
Gestor documental (Alfresco)
Inversión de control (Spring)

Control de autenticación y
acceso (Spring Security)
UDDI
Web Services
Rest Services
Social SSO
SSO (Cas)

JPA-Hibernate, MyBatis
Motor de búsqueda empresarial (Solr)
ETL (Talend)

Dirección de Proyectos Informáticos.
Metodologías ágiles
Patrones de diseño
TDD

BPM (jBPM o Bonita)
Generación de informes (JasperReport)
ESB (Open ESB)



[Home](#) | [Quienes Somos](#) | [Empleo](#) | [Foros](#) | [Tutoriales](#) | [Servicios Gratuitos](#) | [Contacte](#)

Tutorial desarrollado por: Cesar Crespo Martín 2003-2005.

Si te gusta lo que ves, **puedes contratarnos** para impartir **cursos presenciales** en tu empresa o para ayudarte en proyectos (Madrid).

Contacta: cesar@autentia.com.



Descargar este documento en formato PDF [linuxprint.pdf](#)

CCS

Centro de Cálculo de Sabadell 1ª empresa
española de software
www.ccs.biz

Impresoras tarjetas mayor

Impresoras de tarjetas y tarjetas precios
únicos, plazos inmediatos.
www.inditar.com

Cartucho de tinta por 0€

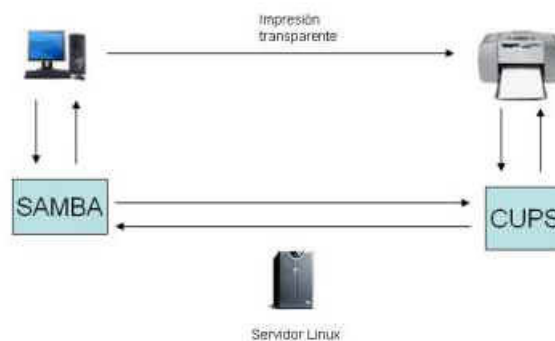
Tinta y toner - ahorra hasta 75% Originales y
compatibles
www.conzumio.com

Anuncios Goooooogle

Anunciarse en este sitio

Compartición de impresoras y ficheros con Linux, CUPS y SAMBA con clientes Windows.

Una de los principales atractivos de Linux es su uso como servidor de ficheros e impresoras mediante Samba para un entorno en el que los clientes son equipos Windows. Samba hace de medio de comunicación entre los clientes Windows y el servidor de impresión Linux que en este caso ha sido CUPS.



El objetivo de la compartición de impresoras remotas es hacer totalmente transparente al usuario el uso de una impresora como si fuera una impresora local, independientemente de donde se encuentre dicha impresora.

Configuración de CUPS:

Cups, actualmente es uno de los mejores servidores de impresión que existen para servidores Linux habiendo desbancado al clásico lpd.

Para conectar una impresora físicamente al servidor debemos tener en cuenta sobre que puerto se va a instalar, si será puerto paralelo o USB. La impresora HP LaserJet 1005 se conecta al servidor mediante un puerto USB. Si conectamos al primero puerto USB del servidor el dispositivo asociado será /dev/usb/lp0, si es al segundo el dispositivo sera /dev/usb/lp1 etc... Esta información la tendremos que tener en cuenta para la posterior configuración de la impresora.

Los drivers para esta impresora y otros modelos de impresoras HP se encuentran en la URL:

<http://foo2zjs.rkkda.com/>

Para bajar los drivers usamos el comando wget.

\$wget <http://foo2zjs.rkkda.com/foo2zjs.tar.gz>

A continuación descomprimos el paquete:

\$ tar xzf foo2zjs.tar.gz

\$ cd foo2zjs

Una vez bajados y descomprimidos leemos la información que se nos indica en el directorio INSTALL, donde se nos indican los siguientes pasos:

Compilamos el driver:

```
$ make
```

Obtener de internet los ficheros necesarios según la impresora que tengamos:

```
$ ./getweb 2300      # Get Minolta 2300 DL .ICM file
$ ./getweb 2200      # Get Minolta 2200 DL .ICM files
$ ./getweb cpwl      # Get Minolta Color PageWorks/Pro L .ICM files
$ ./getweb 1005      # Get HP LJ1005 firmware file
$ ./getweb 1000      # Get HP LJ1000 firmware file
$ ./getweb update    # Get latest version of this software.
```

A continuación instalamos el driver en el sistema. Para ello necesitamos entrar como root:

```
$ su
# make install
```

Para que los nuevos drivers sean detectados reiniciamos el servidor de impresión:

```
# /etc/init.d/cups restart
```

Para probar que la instalación ha sido correcta podemos realizar un test:

```
# make test
```

Una vez instalados los drivers dependiendo de la distribución de Linux con la que estemos trabajando ejecutamos la herramienta de configuración de impresoras.

En el caso de Redhat 7.2/7.3/8.0/9.0, Fedora Core 1:

```
# printconf-gui
```

En Mandrake:

```
# printerdrake
```

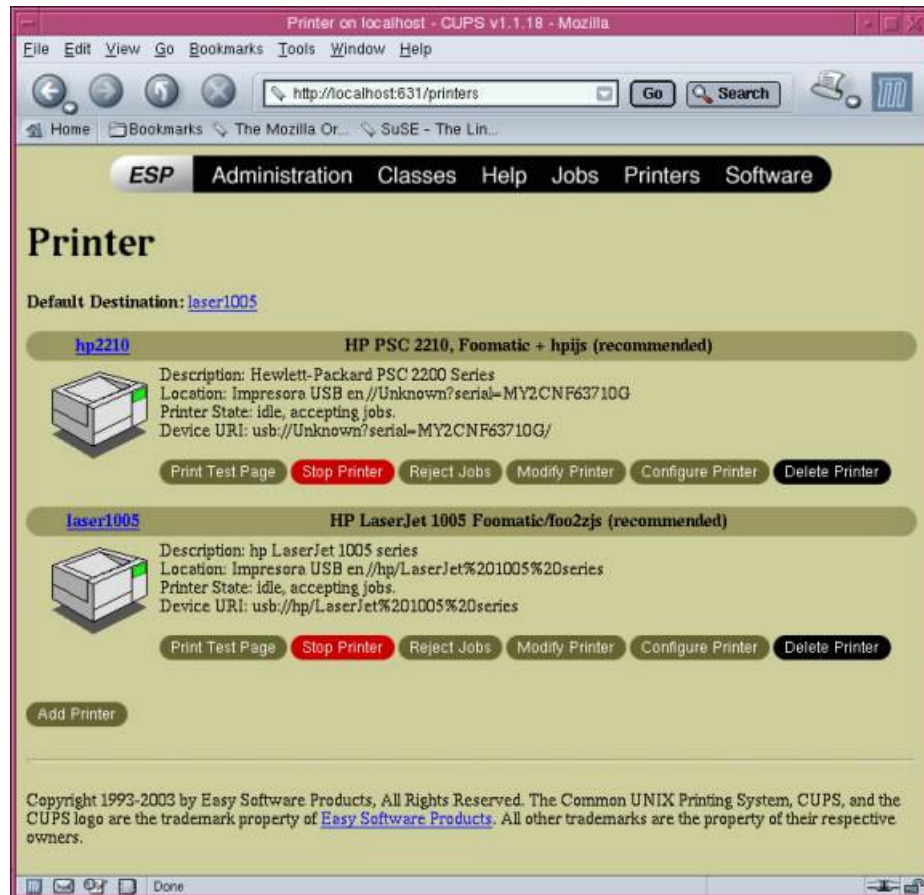
En SuSE 8.2/9.0 para la versión de consola

```
# yast
```

y para usar con las XWindow

```
# yast2
```

O bien si preferimos hacerlo manualmente, Cups puede ser configurado por interfaz Web accediendo al puerto 631 del servidor. Esta opción es la ideal en caso de preferir una configuración manual y en el caso de otras distribuciones que no tengan una herramienta de configuración personalizada como por ejemplo Debian.



A su vez esta utilidad de CUPS nos permite controlar totalmente el servicio de impresión remotamente, así como los trabajos activos etc.

Algunos foros y manuales se indica que es necesario actualizar el firmware de la impresora mediante el comando "cat firmware.img /dev/usb/lp0", y que esta operación hay que realizarla cada vez que se reinicia la impresora. En nuestro caso esto no ha sido necesario, dejando incluso de funcionar si se realizaba esta operación.

Más información acerca de impresoras y sus drivers para Linux se encuentran en:

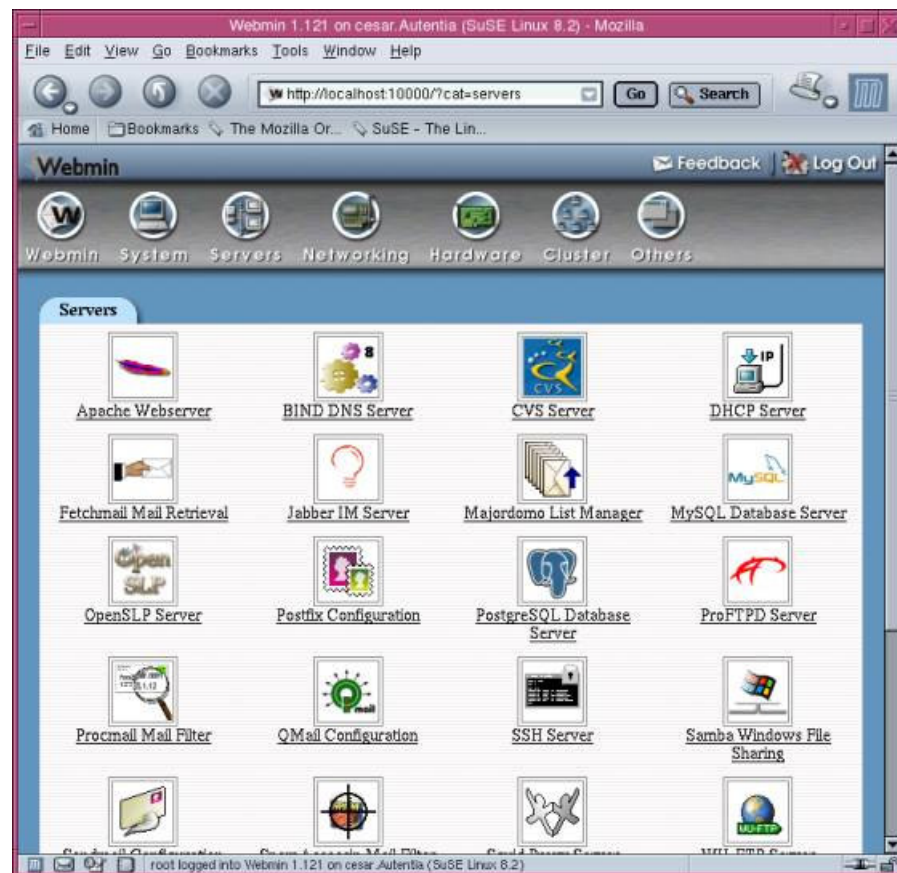
<http://www.linuxprinting.org/>

Configuración de Samba:

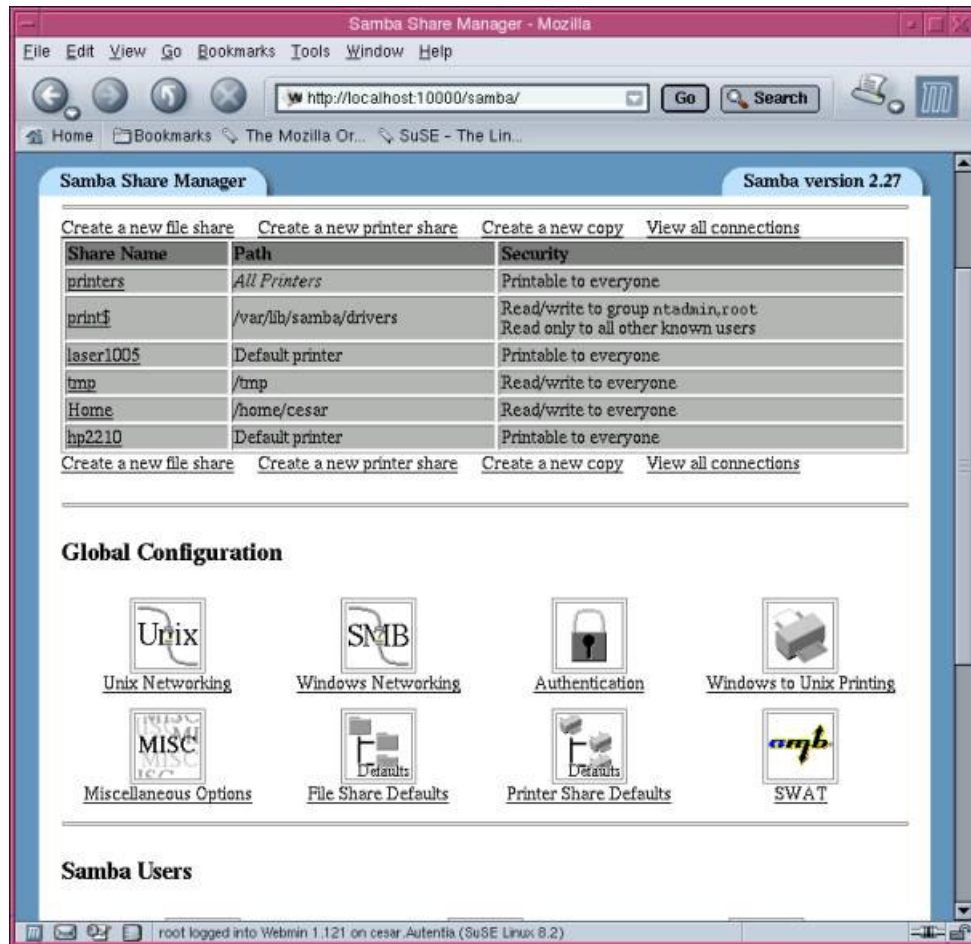
Una buena herramienta de administración por Web para un servidor Linux es sin duda Webmin:



Para acceder a este portal de administración una vez instalado Webmin y arrancado accedemos a él mediante HTTP por el puerto 10000. Por lo que tenemos que dirigirnos a esta URL: <http://localhost:10000/>. Para administrar el servidor se nos pide que nos autentiquemos, para lo cual utilizamos la cuenta de root del servidor Linux. A continuación nos dirigimos a la opción de configuración de servidores pulsando en Servers:



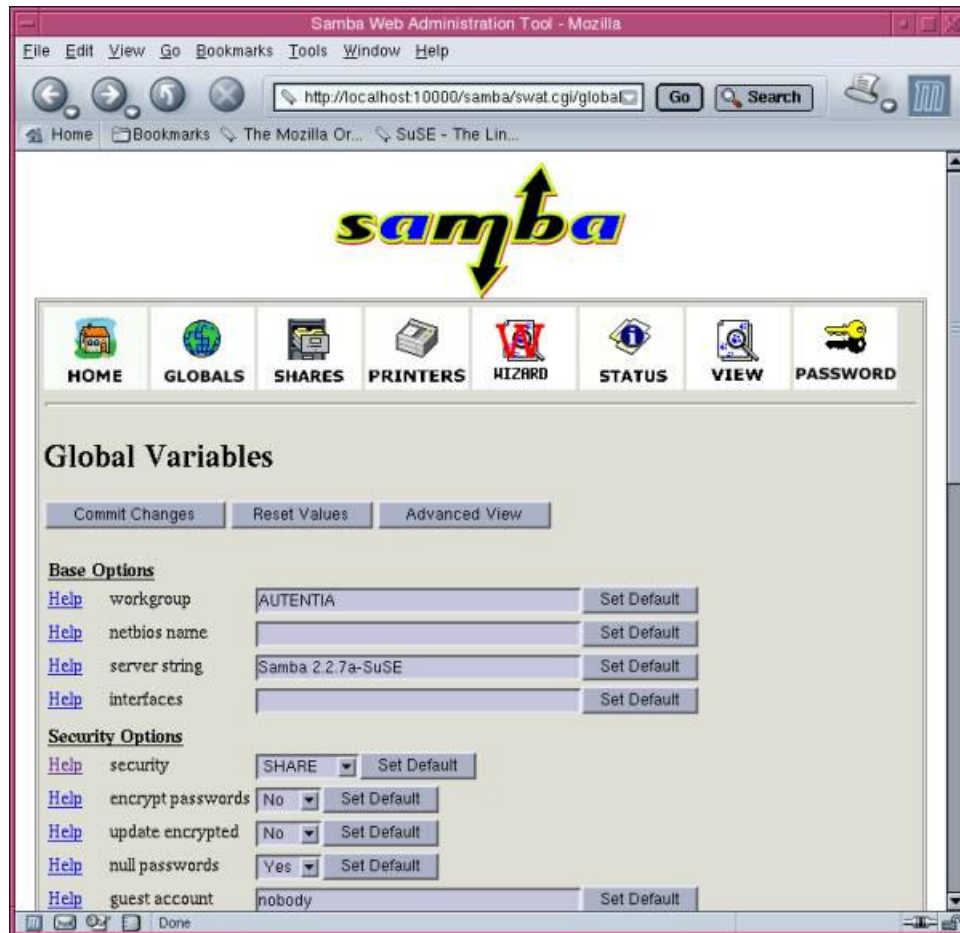
Para configurar Samba hacemos link en "Samba Windows File Sharing", donde se nos dirige a <http://localhost:10000/samba/>:



Una vez aquí utilizamos la herramienta SWAT que es la herramienta de configuración Web de Samba:



Lo primero que tenemos que configurar para el uso de Samba son las propiedades globales, para lo cual pulsamos el link "Globals":



Aquí se nos presentan las propiedades globales de compartición de recursos de Samba.

Por cada propiedad configurable existe un link asociado "Help" en el que se nos muestra la descripción de esa propiedad.

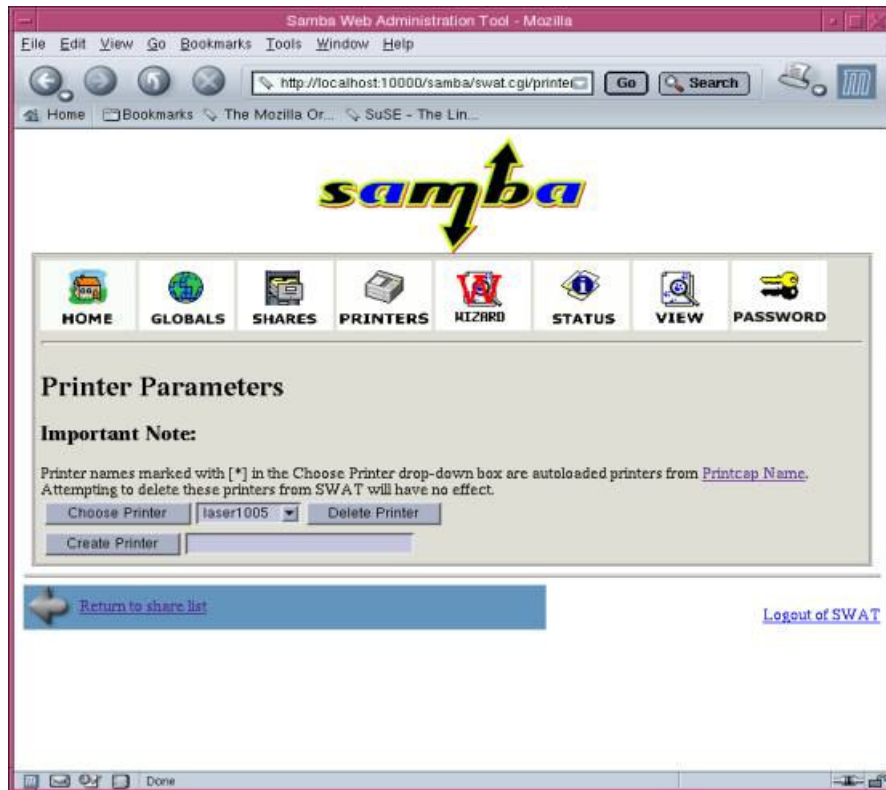
Para un uso sencillo de Samba usamos el modo de seguridad SHARE . Por lo que en Security Options la opción "security" la ponemos a SHARE.

Para tener un uso publico de ficheros transparente a los clientes Windows hacemos que la comunicación entre clientes Windows y Samba se realice mediante la cuenta *Guest/Invitado* (en la versión en inglés de Windows la cuenta de invitado se llama *Guest* y en la versión en español *Invitado*). Para esto debemos configurar las propiedades "guest ok" y "null passwords". Se habilitan la opción null passwords ya que por defecto la cuenta de invitado no tiene password.

En el servidor Linux tiene que existir una cuenta asociada al usuario *Guest/Invitado* sabiendo que todos los ficheros creados en el sistema de ficheros Linux se crearán con este usuario. Además hay que tener en cuenta que si en Linux no hay permisos para este usuario sobre un recurso como un directorio o fichero en el servidor, no podrá ser compartido en samba.

Para esto existe la propiedad "guest account" donde se le indica el usuario que usará Linux para la comunicación con los clientes Windows. En este ejemplo hemos utilizado la cuenta local del servidor Linux "nobody" para compartir ficheros por Samba. Una buena técnica también podría ser crear un usuario llamado samba o smb para identificar el usuario con el servicio que va a prestar.

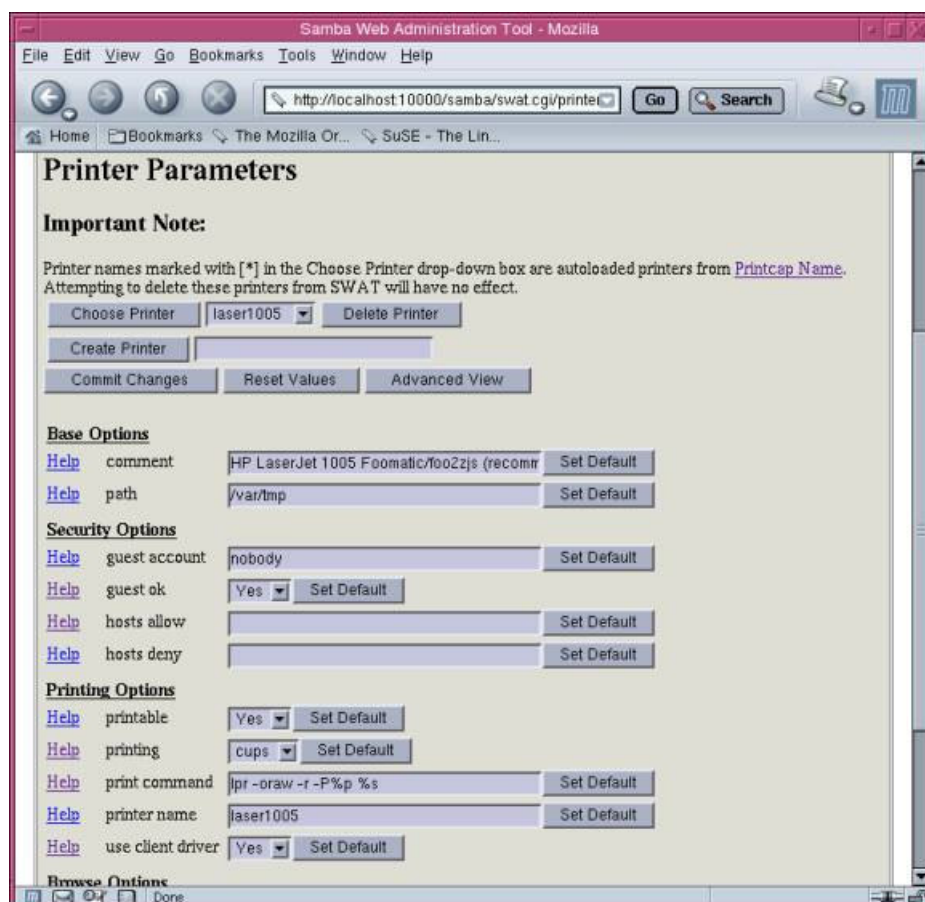
Una vez configuradas las opciones globales de samba pulsamos el botón "Printers" para configurar las impresoras a compartir:



Aquí se nos indica que elijamos una de las impresoras que tenemos presentes en el sistema.

Si en el fichero smb.conf se ha indicado que cargue las impresoras presentes en el fichero printcap aparecerán como importadas con una configuración por defecto. En caso contrario habrá que añadirlas manualmente.

Pulsamos "Choose Printer" con la impresora seleccionada y se nos presenta la configuración de las propiedades Samba para dicha impresora:



Una vez elegida la impresora a configurar pulsamos el botón **Advanced view** ya que utilizaremos propiedades avanzadas para la configuración de la misma.

Al igual que antes la opción *"guest ok"* tiene que estar habilitada.

También tenemos que habilitar las propiedades *browseable*, *avaible* y *printable* para dar permisos para imprimir y ver los trabajos de impresión.

Finalmente configuramos dos de los parámetros más importantes para la compartición de impresoras con clientes Windows.

Estos parámetros son *"printing"* y *"print command"*.

El parámetro *"printing"* controla como se comunicará Samba con el servidor de impresión instalado, que en nuestro caso es CUPS, por lo tanto este parámetro debe ser configurado como CUPS.

Finalmente el parámetro *"print command"* indica las opciones de impresión que con que se mandará el trabajo de impresión al servidor CUPS una vez ha sido recibido por Samba.

La opción por defecto para este parámetro es: `lpr -r -P%p %s`.

A estos parámetros hay que añadirles la opción `-oraw` para que el trabajo que nos llega desde un servidor Windows sea enviado a la impresora sin formatear, ya que Windows mediante su driver de la impresora nos envía el trabajo de impresión en el lenguaje de la impresora.

Por lo tanto la opción *print command* debe quedar como sigue:

```
lpr -oraw -r -P%p %s
```

De esta forma conseguimos un doble objetivo, primero que desde los clientes Windows puedan usar esta impresora como si fuera una impresora local, ya que le envían documentos en un formato comprensible por la impresora; y segundo esa misma impresora puede ser usada desde Linux como impresora local pudiendo enviar trabajos en diferentes formatos:

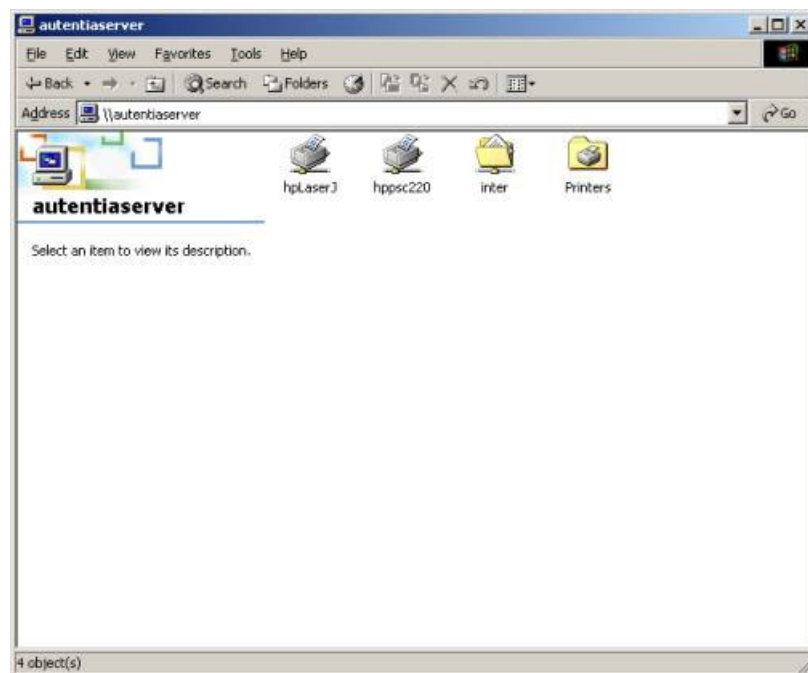
```
#lpr -Plaser1005 mozilla.ps
```

```
#lpr -Plaser1005 mail.txt
```

Finalmente para proporcionar una mayor transparencia al cliente Samba permite que los drivers necesarios por esta impresora sean descargados directamente del servidor, sin necesidad de mostrar al usuario el menú de selección de driver. Para ello debemos configurar las opciones *printer driver* de la configuración de Samba indicando donde se encuentran los ficheros correspondientes al driver.

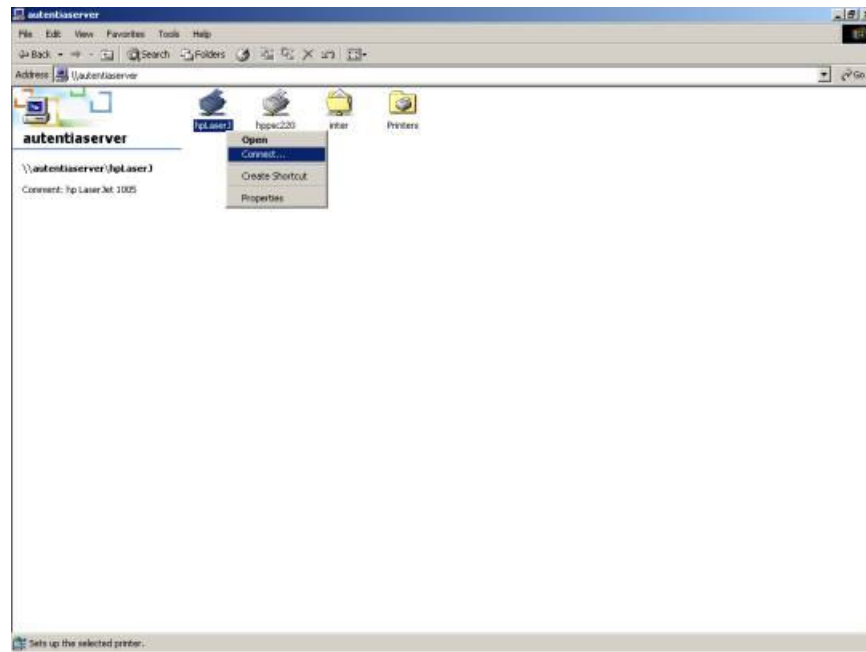
Configuración de los clientes Windows

Una vez configurado el servidor debemos configurar correctamente el cliente para la utilización de este recurso compartido:

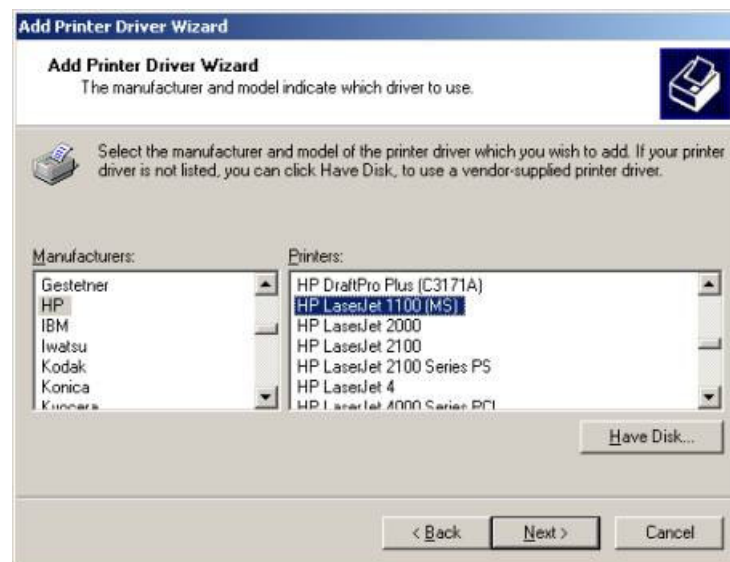


Para poder usar esta impresora debemos pulsar el botón derecho sobre ella y seleccionar **Connect ...**

Si en el servidor Samba hemos configurado la impresora para que nos instale los drivers automáticamente la instalación de la impresora será totalmente transparente y se nos añadirá automáticamente a nuestra lista de impresoras.



En caso contrarió tendremos que instalar el driver apropiado en el cliente Windows localmente previamente a conectar la impresora. Una vez conectada se nos indicará que seleccionemos el driver apropiado para la impresora.

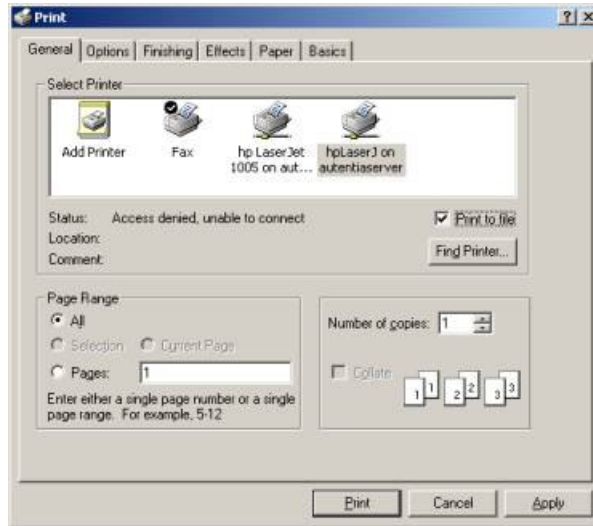


Finalmente tenemos una impresora remota en nuestro servidor Linux que podemos usar de forma totalmente transparente. Además esta impresora es totalmente configurable ya que podremos modificar las propiedades de impresión con las que se envía este documento.

Posibles problemas y soluciones

Cuando en un equipo Windows se instala una impresora, es necesario tener instalados los drivers particulares para la impresora que estamos utilizando. Estos drivers serán los responsables de comunicarse con la impresora formateando los trabajos de impresión que se envían en un lenguaje que entienda la impresora. En el caso de la impresora HP LaserJet 2210 siguiendo los pasos anteriormente indicados tenemos una impresora totalmente funcional en Linux, pero al imprimir remotamente mediante Samba desde un cliente Windows nos encontramos que los trabajos no llegan a imprimirse.

Una técnica muy útil cuando aparecen problemas en estos casos, consiste en que desde el cliente Windows en las propiedades avanzadas de impresión imprimamos nuestro documento en un fichero.



Este fichero contiene la información "cruda" que se envía a la impresora, es decir en el lenguaje o formato que entiende la impresora físicamente.

Debemos copiar este fichero al servidor Linux. Con este fichero es con el que realizaremos las pruebas de impresión desde el servidor emulando así el proceso de impresión remota, ya que Samba lo único que hace es encolar este fichero en la cola de impresión.

Teniendo este fichero en el servidor podemos comprobar si se produce algún error al imprimirlo directamente mediante el comando lpr:

```
$ lpr -oraw -Php2210 print.prn
```

En caso de devolvernos un error el comando lpr significa que el formato en el que se encuentra codificado el fichero print.prn no es reconocido por el servidor de impresión de Linux. Es decir que el formato en que los drivers propietario de HP nos ha codificado el documento a imprimir no es reconocido por el servidor de impresión.

En este caso una posible solución es estar pendiente de nuevas actualizaciones del driver foozjs y actualizarse frecuentemente.

Puesto que la opción de esperar a una nueva versión de los drivers foozjs no es una opción muy agradable existe otra alternativa. Esta alternativa consiste en lo siguiente:

Ya los drivers proporcionados por HP para el modelo psc 2210 nos han dado problemas probamos a elegir los drivers de otro tipo de impresora. Como por ejemplo los drivers de la 2100.

Comprobamos que con estos drivers se produce una salida del documento por la impresora, aunque con caracteres extraños y errores varios obteniendo un documento impreso que no era el deseado.

Si indicamos que la impresora remota es una impresora genérica de sólo texto comprobamos que la impresión se produce correctamente. Esto corrobora la causa del problema anterior es el formato en el que envía Windows no es compatible con el servidor de impresión.

Pero esta solución no es aceptable ya que únicamente se nos permitiría imprimir texto en esa impresora.

Para evitar esto proponemos la siguiente solución: El uso de los drivers PostScript de Adobe.

Estos drivers son descargables desde:

<http://www.adobe.com/support/downloads/product.jsp?product=44&platform=Windows>

Cuando conectemos esta impresora debemos indicarle que los drivers que vamos a usar no van a ser los de HP, sino los drivers de Adobe correspondientes a impresoras PostScript

Puesto que ahora los documentos de impresión el cliente Windows los enviará en formato PostScript en la configuración Samba de Linux debemos quitar la opción de impresión de "printer command" -oraw.

Por lo tanto la opción *print command* debe quedar como sigue:

```
lpr -r -P%p %sç
```

De esta forma el servidor de impresión será el encargado de formatear el documento de impresión de formato PostScript a uno entendible por la impresora que finalmente saldrá por la impresora de una forma transparente para el usuario.

Si desea contratar formación, consultoría o desarrollo de piezas a medida puede contactar con



[Autentia S.L.](#) Somos expertos en:
J2EE, C++, OOP, UML, Vignette, Creatividad ..
 y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	
	Enviar

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto

[Automatización de procesos con Scripts en Windows](#)

[Acceso seguro a CVS a través de SSH](#)

[Instalar JBoss](#)

[Activar SSL en IIS](#)

[PHP Nuke en Windows 2000](#)

[Gestión de contenidos y errores comunes](#)

[Repositorio CVS en Windows](#)

[Patrones de GRASP](#)

[Apache, MySQL y PHP](#)

Descripción

En este tutorial aprenderéis a simplificar la actualización automática de ficheros de vuestro Web, utilizando comandos ftp.

Os mostramos como securizar los accesos a CVS a través de SSH, utilizando herramientas gratuitas

Os mostramos como instalar en servidor gratuito de aplicaciones JBOSS así como a automatizar su arranque y parada.

Os mostramos como activar el soporte de https en IIS, creando vuestros propios certificados autofirmados, usando OpenSSL

Os mostramos como instalar paso a paso Php-Nuke en vuestro entorno Windows con MySQL

Os explicamos en que consiste la gestión de contenidos y cuales son los errores cometidos por multitud de empresas a la hora de abordar su implantación

Os mostramos como montar un servidor para el control de versiones CVS en Windows así como acceder a él a través de WinCVS

Os presentamos una introducción a los patrones de asignación de responsabilides y su relación con el proceso unificado.

Os mostramos como configurar Apache, MySQL y PHP en vuestra máquina

Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)

