



Taller del Administrador: El Sistema X Window

EXPEDIENTE X

Gracias a la detección automática del hardware, los administradores de hoy en día raras veces tienen que configurar un sistema X Window manualmente. Pero si se quieren aprovechar al máximo las excelentes características de red de X11 y afinar su configuración, será conveniente que se tenga algún conocimiento de cómo funciona. **POR MARC ANDRÉ SELIG**

Linux continúa avanzando en el terreno de los escritorios X-Windows. Uno de los factores principales en el éxito de Linux es la facilidad de uso tanto del escritorio como de las aplicaciones. Esta abundancia de software, sin embargo, significa que los administradores deben pasar más tiempo instalando las herramientas subyacentes que se encargan de los controles del ratón y del teclado, de los gráficos y de la salida por pantalla de estas aplicaciones.

El servidor X es el componente que se encarga de casi todas las representaciones gráficas en Linux. Las principales excepciones a este punto son la consola y la biblioteca SVGA-Lib [2], una biblioteca que soporta la representación directa de imágenes en los sistemas Linux. El servidor X, por sí mismo, proporciona rutinas primitivas para mostrar jerarquías de ventanas. El servidor X también proporciona aceleración hardware, además de ofrecer componentes para el control del teclado y el ratón.

Si no conoce las arquitecturas cliente/servidor, el protocolo X puede que le resulte algo confuso al principio, ya que el servidor X11 a menudo se encuentra situado en una pequeña estación de trabajo bajo el escritorio del usuario y los

clientes residen en grandes máquinas en el centro de datos.

Intercambio de Roles del Cliente y el Servidor

En el oscuro mundo de las X, el servidor es un programa que interactúa directamente con el usuario. Representa los widgets (elementos gráficos) en la pantalla y se encarga de aceptar la entrada. Los clientes usan el servidor en vez de representar ellos mismos los gráficos en la pantalla. Es decir, un cliente normalmente no necesitará acceder directamente al hardware gráfico, sino que simplemente le pedirá al servidor que realice las funciones gráficas. Los clientes X típicos suelen ser los procesadores de texto, los navegadores de Internet o los emuladores de terminales.

Si el cliente y el servidor se ejecutan en máquinas diferentes, el servidor X está casi siempre ubicado en la máquina del escritorio. Este es también el caso de los *thin clients* basados en Linux, que incluyen un servidor X completo.

Realmente no importa mucho donde resida el cliente. Teniendo una conexión de red rápida, el cliente puede residir en el centro de datos de la habitación de al lado o en la otra punta del mundo. A

menudo los clientes residen en la misma máquina que el servidor.

Los denominados sistemas kioscos que ejecutan una única aplicación – un Cybercafé, por ejemplo, o un visor de imágenes de rayos X – son un buen ejemplo de aplicaciones útiles de un servidor X con un único cliente.

Dos comandos son lo único que se necesita para instalar un sistema kiosco. El administrador primero tiene que ejecutar el servidor, y luego el cliente o los clientes requeridos. En el siguiente comando, la opción `:3` estipula que el servidor se ejecutará en el cuarto (se empieza a contar desde cero) display local (y no en el cuarto monitor):

```
X :3 < /dev/null > ⌘
/dev/null 2>&1 &
exec firefox --display :3 &
```

El programa *xinit* facilita la sintaxis encargándose de ejecutar el servidor X.

Las siguientes líneas ejecutan el servidor X y luego el Xterm, a menos que *.xinitrc* en el directorio home del usuario tenga una configuración diferente:

```
xinit -fn 9x13 -- :3 > ⌘
/dev/null 2>&1 &
```

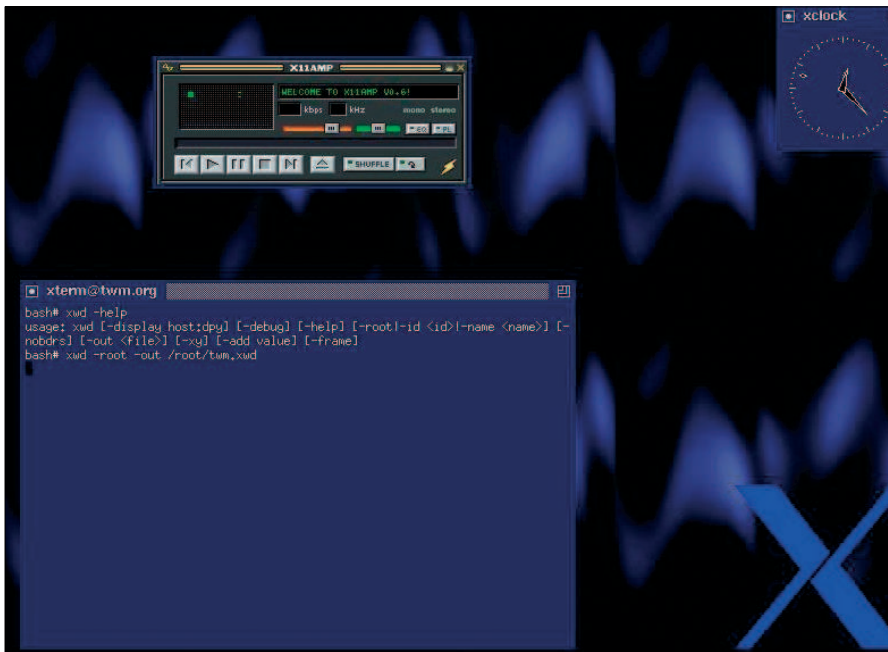


Figura 1: Aunque el administrador de ventanas TWM proporciona una funcionalidad mínima, es fácil de usar y conveniente.

El fichero `/etc/X11/XF86Config-4` es el núcleo de cualquier entorno X. Dependiendo de la variante del servidor X que se tenga y de la distribución, este fichero podría llamarse `xorg.conf` o algo similar. El fichero contiene la configuración de todos los dispositivos que cooperan con el servidor X.

Como Linux ha conseguido progresos sorprendentes con la detección del hardware, los administradores normalmente no necesitan modificar la configuración del servidor y si lo hacen, los cambios son mínimos. Los programas que automáticamente crean este fichero son

Listado 1: Importar Cookies

```
01 #mas:~$ xauth nextract myxkey
02 #mas:~$ chmod 640 myxkey
03 #mas:~$ su unsafe
04 #Password:
05 #unsafe:/home/mas$ firefox
06 #Xlib: connection to ":0.0"
    refused by server
07 #Xlib: No protocol specified
08 #
09 #(firefox-bin:3086):Gtk-WAR-
    NING **: cannot open display:
    :0
10 #unsafe:/home/mas$ xauth nmer-
    ge myxkey
11 #unsafe:/home/mas$ firefox
```

`xf86cfg`, `xf86config`, o para Suse, Sax o Sax 2.

El fichero de configuración no es sólo específico para el ordenador y el adaptador gráfico, sino también para el monitor. Antes de cambiar el monitor o el adaptador gráfico, hay que asegurarse de que se tiene un método alternativo para acceder al ordenador, (a través de la red o de la consola), en el caso de que las opciones no sean correctas y el servidor X no se ejecute.

Gestores de Ventanas

Por sí solo, el servidor X no es muy amigable, especialmente si se desea trabajar con varios programas al mismo tiempo. Se necesita un componente que estructure de forma conveniente el uso del ordenador. El gestor de ventanas añade la barra de títulos a las ventanas individualmente, con el objeto de poderlas identificar fácilmente, y un marco para permitir que los usuarios puedan escalar las ventanas. La Figura 1 muestra TWM, uno de los gestores de ventanas más antiguos.

TWM permite a los usuarios mover y solapar las ventanas e incluso ordenarlas automáticamente. También implementa un sistema sencillo de menús que posibilita a los usuarios a ejecutar los programas y minimizarlos o escalar las ventanas.

Los gestores de ventanas reconocidos como TWM o FVWM son sencillos, eficientes y muy útiles. Hoy en día los “thin

clients” o las máquinas antiguas los utilizan para ahorrar espacio y recursos.

Tan sólo se necesitan tres líneas para ejecutar una sesión X con TWM y Firefox:

```
X :3 < /dev/null > 2
/dev/null 2>&1 &
twm -display :3 &
exec firefox --display :3 &
```

En la actualidad, la mayoría de los usuarios prefieren los entornos de escritorio completos como KDE o Gnome. Estos sistemas aportan herramientas, utilidades y extensas bibliotecas, además del gestor de ventanas. Las bibliotecas ayudan a simplificar la programación de la GUI, proporcionándole a los programas un aspecto uniforme y optimizando la comunicación entre aplicaciones.

Los entornos de escritorio modernos añaden un gestor de sesiones al gestor de ventanas. El gestor de sesiones restaura la sesión previa, ejecutando los programas que se usaron la última vez y colocando las ventanas donde estaban. Tanto Gnome como KDE tienen potentes gestores de sesiones.

Login Gráfico

El servidor X maneja la entrada y la salida por pantalla, y el gestor de ventanas facilita el uso de múltiples ventanas. Esto tan sólo deja un deseo en la lista de los usuarios, la posibilidad de poder conectarse al sistema en modo gráfico. No hay muchos usuarios hoy en día que deseen ejecutar una sesión basada en GUI usando comandos de la consola.

Un gestor de visualización es una característica bastante deseada. Se encarga de llamar al servidor X y abre un cuadro de diálogo como el mostrado en la Figura 2. Después de autentificar al usuario, el gestor de visualización ejecuta automáticamente el entorno configurado. Si se necesita, puede hacer uso de las capacidades de red del sistema X11, ya que el gestor de visualización no tiene por qué estar en la misma máquina que el servidor X. El gestor de visualización se ejecuta como un simple cliente. El servidor X puede entonces usar el protocolo XDMCP para comunicarse con el gestor de visualización en la otra máquina.

El administrador bien puede asignar un equipo como la máquina del gestor de visualización, o bien el servidor X puede

retransmitir paquetes a través de la LAN para localizar automáticamente un gestor de visualización adecuado. El comando es como el siguiente: `X -broadcast`. Para ejecutar el gestor de visualización en un ordenador dedicado hay que teclear `X -query host` para ejecutar el servidor X.

Todas las distribuciones modernas de Linux instalan ahora un gestor de visualización y una configuración junto con X11. La forma más fácil de averiguar dónde están estos componentes es ejecutando `locate` [3], especificando el nombre del componente del gestor que se está buscando. Los componentes de XDM, del alternativo de KDE (KDM) (Figura 2) y del Gnome (GDM) están ampliamente difundidos.

Controlando la Visualización

Los ordenadores no están restringidos a ejecutar un único servidor X. Con sus múltiples terminales virtuales, Linux puede simultáneamente soportar múltiples servidores. Aplicaciones de software X especiales pueden incluso ejecutarse sin monitor y teclado, que es importante para los servidores terminales. Como el cliente y el servidor se ejecutan independientemente, e incluso no tienen porqué residir en la misma máquina, es importante para los clientes conocer a qué display enviar su salida. La variable de entorno `DISPLAY` se encarga de especificar un destino para los datos que se vayan a visualizar.

La variable contiene un hostname, un número de display y opcionalmente un número de pantalla: `(Host):Display(.Screen)`. El número de la pantalla normalmente no hace falta, excepto en los entornos Xinerama donde un único servidor X controla múltiples displays. Si el hostname no se especifica en `DISPLAY`, el cliente automáticamente usa la máquina local, usando sockets Unix en vez de puertos TCP para comu-

Listado 2: Tunel X11 sobre SSH

```
01 #mas@ishi:~$ ssh -X
    kanat.pair.com
02 #[...]
03 #mas@kanat:$ echo $DISPLAY
04 #localhost:10.0
05 #mas@kanat:$ firefox &
```

nicarse (como en una conexión de red de X11).

Cuando un usuario se conecta por medio del gestor de visualización, las variables de entorno se establecen automáticamente. Para ejecutar automáticamente un programa X, primero hay que definir una variable. La siguiente sintaxis ejecuta el navegador Firefox en la máquina local; el servidor X envía la salida de Firefox a un equipo denominado `sun14`:

```
export DISPLAY=sun14:0
firefox &
```

Si se está usando X11 en una red, hay dos consideraciones importantes que se deben tener en cuenta. Primero, los servidores no aceptan conexiones de clientes arbitrarios ya que el acceso por parte de clientes arbitrarios provocaría riesgos de seguridad. Segundo, el flujo de datos de la sesión X, por defecto, no está encriptada. Los atacantes podrían fácilmente fisgar los protocolos y hacerse con la sesión.

Casi todos los servidores X modernos aplican controles de acceso por defecto. Hay dos sistemas alternativos para el control de acceso: Xhost permite acceso a máquinas dedicadas, lo que no lo hace muy seguro; Xauth usa cookies criptográficas y soporta controles muy exhaustivos.

Estos esquemas de protección son una parte importante del entorno X, ya que los clientes X pueden inducir al servidor y a otros clientes a usar una gran variedad de protocolos. Por ejemplo, la entrada por teclado es muy sencilla. Hay que asegurarse de permitir solamente a los programas en los que se confía el acceso al servidor X.

Control de Acceso Basado en Cookies

La herramienta Xauth almacena sus cookies en un fichero denominado `.Xauthority`, en el directorio home del usuario. Cuando un usuario ejecuta un sistema cliente, el cliente analiza el fichero `.Xauthority` y utiliza la cookie apropiada para autenticarse con el servidor X. Si se está ejecutando una sesión en otra máquina o si se va a utilizar una credencial diferente, habrá que importar la cookie a `.Xauthority` primero a (véase el Listado 1). `xauth nextract` almacena la



Figura 2: KDM de KDE tiene un menú que permite a los usuarios seleccionar un escritorio o apagar el sistema.

cookie en un fichero y `xauth nmerge` importa el fichero a `.Xauthority`.

La variante Xhost se utiliza en entornos donde se trabaja con una sola máquina para deshabilitar totalmente el control de acceso. El comando `xhost +` permite a cualquier cliente en cualquier equipo acceder al display. El comando `xhost` devuelve el control de acceso al cliente.

Xauth sólo controla a quién se le permite acceder al servidor X. No protege la transmisión de datos en absoluto. Si se usa el mecanismo del Listado 1 para ejecutar clientes en máquinas remotas, debería estar al tanto de que un atacante podría fisgar sus datos.

El popular protocolo SSH ayuda a establecer un túnel para el protocolo X a través de una conexión encriptada. El programa proporciona encriptación, establece las variables de entorno con los valores apropiados y utiliza Xauth para crear un fichero `.Xauthority` en la máquina remota.

Estableciendo un Túnel con SSH

Para montar el túnel, hay que establecer a `yes` la variable `X11Forwarding` en `/etc/ssh/ssh_config` y una vez hecho esto, los usuarios pueden poner el flag `-X` en la línea de comandos para crear un túnel con X11 (Listado 2). SSH proporciona un servidor X virtual en la máquina remota (normalmente `:10`); y sus clientes usan el display del servidor X local. ■

RECURSOS

- [1] X.org: <http://www.x.org/>
- [2] SVGALib: <http://www.svgalib.org>
- [3] Marc André Selig, "Admin Workshop: Finding Files": Linux Magazine 02/05, p. 62.