

Cientes delgados usando Software Libre

Ing. Fernando Cuenca, Maximiliano Morales, Ignacio Sanchez, Gustavo Yapura

El proyecto tiene como objetivo montar un laboratorio de computación utilizando clientes delgados y software libre de forma de utilizar hardware que de otra forma sería obsoleto para las necesidades mínimas actuales.

Existen dos formas preponderantes de utilizar aplicaciones en una computadora. Los “Clientes Gordos” en donde las aplicaciones se ejecuten en la máquina que está el usuario y todos los datos se procesan ahí y los “Clientes Delgados” en donde todo el procesamiento se realiza en un servidor y solo se muestran los resultados y las pantallas en la máquina que está el usuario.

Como en todo, existen muchas categorías en el medio, por ejemplo en aplicaciones cliente/servidor en donde el programa pide los datos a una base de datos, el procesamiento de los datos se hace en el servidor y la representación y el manejo de los mismos se hace en la máquina del cliente. Otro caso pasado son las aplicaciones web donde gran parte del procesamiento se realiza en el servidor.

El proyecto nació como una necesidad puntual que tenía el laboratorio de sistemas cuando su infraestructura estaba basada en Novell/DOS/Windows 3.11 bootando vía red ya que las máquinas no tenían disco rígido.

El problema residía que las máquinas eran demasiado lentas para correr el software nuevo que había en la época y que no había drivers para Windows 3.11 para las máquinas que se habían comprado nuevas por lo que la infraestructura de ese momento no podía crecer en ningún aspecto.

Se plantearon tres posibles cursos de acción, el primero consistía en cargar en forma remota Windows 95 y el segundo consistía en poner discos rígidos en todas las máquinas y reemplazar las que fueran lentas por nuevas y por último el de armar clientes delgados usando Linux usando las máquinas que había en el momento.

Las pruebas utilizando Windows 95 no fueron satisfactorias, principalmente por la lentitud y los problemas de corrupción del registro. Las máquinas necesitaban más memoria y una buena parte del equipamiento no se podría utilizar. La versión siguiente de Windows que ya estaba en el mercado no permitía el boot remoto por lo que esa línea de trabajo fue descartada.

La alternativa de instalar discos rígidos en todas las máquinas y reemplazar las que no tenían la capacidad necesaria para ser utilizadas era demasiado costosa.

Durante la evaluación de las otras alternativas se decidió probar con clientes delgados utilizando GNU/Linux como base para los mismos ya que es software libre, por lo que teníamos el acceso al código fuente y la posibilidad de modificar lo que necesitábamos para adaptarlo a nuestras necesidades.

Etapa de experimentación

Antes de poder evaluar si era viable o no la utilización de GNU/Linux como cliente delgado y el modelo en si nos hacia falta una implementación básica que funcionara. Para lo cual se utilizo 2 máquinas, una que actuaría como servidor y otra como cliente delgado sin disco.

Como primer paso se busco una distribución de GNU/Linux que estuviera disponible, en ese momento se eligió RedHat 5.2 y se instalo pensando que fuera tanto una estación de trabajo como un servidor.

El kernel de linux es muy configurable y entre las opciones que nos brindaba estaba la posibilidad de usar el sistema de archivos raiz (root filesystem) via un protocolo de red llamado NFS (network filesystem). Otra de las opciones es que el mismo kernel obtubiera la configuración de la red via los protocolos rarp, bootp o dhcp. La parte final que tambien esta incluida es la posibilidad de tener incorporado dentro del kernel el driver de la placa de red.

Se procedió a compilar el kernel para que tuviera lo mínimo e indispensable pero que incluyera las tres partes que hacían que el mismo pudiera trabajar como cliente delgado. Luego de varios intentos se consiguió un kernel que funcionaba.

Por cada intento se grababa el kernel de linux en un disquete, se modificaba un registro en el mismo para que pudiera ser leído cargado desde el mismo disquete y luego se lo probaba en el cliente delgado.

Una vez que el kernel consiguió bootear correctamente, detectar la placa de red y pedir la configuración de red se procedió a instalar en el servidor un servicio de bootp para proveer de IPs de las máquinas. Siguiendo la documentación y luego de varios intentos se pudo configurar el programa correctamente y la estación remota ya podía tomar su configuración de red.

El primer escollo apareció debido a que no conocíamos como armar una estructura de archivos para que fuera el root filesystem de la máquina remota. Como primera prueba se exporto el sistema de archivo completo de la máquina que estaba trabajando como servidor. Lo primero que observamos es que los scripts de inicio no estaban preparados para levantar una estación teniendo el sistema de archivos vía NFS y también vimos que al arrancar necesita poder escribir ciertos directorios y que estos tienen que ser únicos por cada maquina.

Debido al poco espacio de disco que se disponía lo que se realizo fue copiar todo el sistema de archivos del servidor excepto /usr a un directorio y exportar ese directorio y /usr.

Como los scripts de inicio no estaban preparados para cargar el sistema desde NFS hubo que revisar que es lo que fallaba cuando se iniciaba la estación, corregir en el servidor apropiadamente e intentar de nuevo hasta que los scrips funcionaran correctamente.

El tener el /usr en otro recurso compartido implicaba un trabajo extra a la hora de modificar los scripts ya que durante el proceso de arranque hacen falta programas que están ahí y el recurso debe estar montado antes.

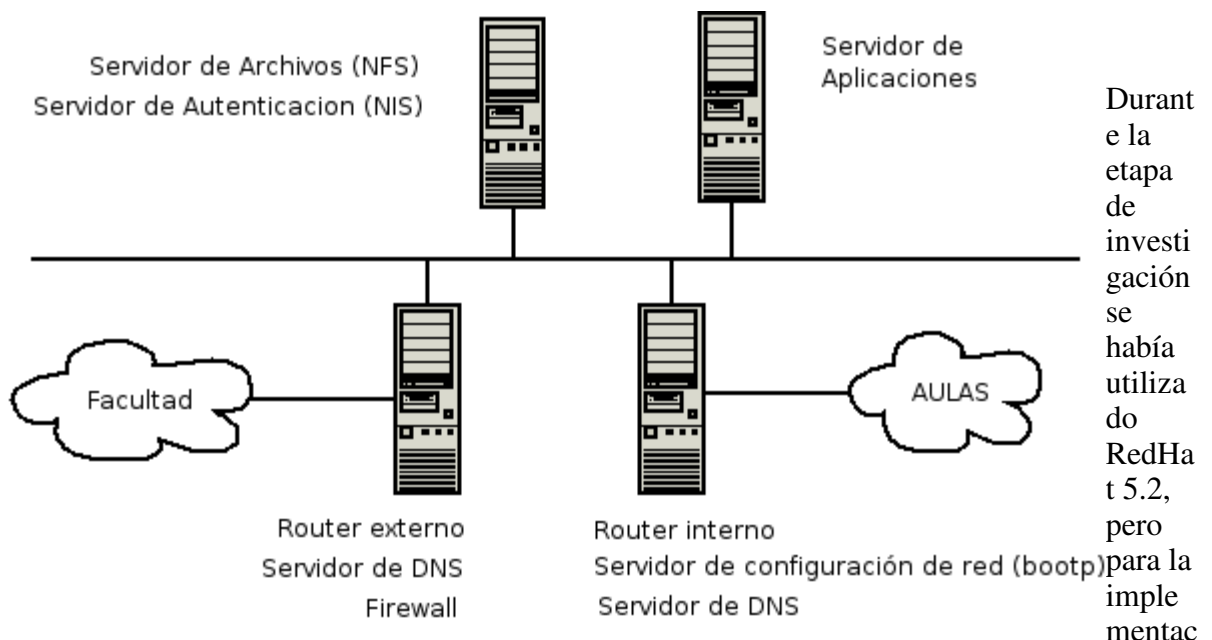
Diseño de la red y servidores

Usando los equipos disponible se diseño la estructura de la red. Se disponía de 2 servidores principales y se deseaba tener una red individual por cada una de las 3 aulas.

Se designó a un de los servidores para que preste el servicio de archivo, incluyendo esto ultimo los root filesystem de las estaciones remotas y Autenticación de usuarios de la red. Para el sistema de archivos vía de red se utilizó NFS. Se exportaron los recursos donde se almacenan los directorios de los usuarios y los directorios que forman la estación remota. Para la autenticación se utilizó (NIS). El servidor fue instalado con RedHat 6.0

El segundo servidor se destino para servidor de aplicaciones. Al igual que el servidor de archivos se instalo RedHat 6.0. Para proveer el login gráfico se habilito **XDMCP** (X Display Manager Control Protocol). Como entorno de trabajo se eligió KDE porque era el que mejor se adaptaba a lo que los alumnos estaban acostumbrados.

Se armaron 2 equipos más para que actúen como routers, servidores de DN, servidor de bootp y firewall. El router interno conectaba cada una de las aulas con los servidores y el route externo conectaba los servidores con el resto de la facultad.



ión se opto por RedHat 6.0 ya que tenia mejoras importante y mejor soporte de hardware. Este cambio implicó rehacer las modificaciones necesarias para que la nueva imagen de booteo funcionara correctamente.

Como el espacio en el servidor de archivos era limitado y no podimos tener la cantidad de copias necesarias para bootear todas las maquina, se comenzó a optimizar el espacio que necesitaba la estaciones. Para lo cual se busco lo mínimo que hacia falta para que la estación arrancara y el resto eran enlaces simbólicos hacia recursos compartidos. En los prieros intentos se logro bajar el consumo de espacio por estación a menos de 25MB y luego eso se lo llevo a apenas 6MB por estación.

Como todos los usuarios tenían que trabajar sobre la misma máquina en realidad (el servidor de aplicaciones) no se podía usar un usuario único, por lo que a cada máquina se le asocio un usuario genérico propio. Así e usuario para la máquina 1 era alumno1.

Para facilitar las cosas se decidió que los usuarios no tuvieran que ingresar una clava para ingresar. Para evitar que los usuarios usaran otra cuenta que no fuera la que le correspondía a su

máquina se implementó la política de límites de concurrencia simultánea en el servidor. Eso quiere decir que no puede haber dos personas usando una sola cuenta desde 2 máquinas distintas.

Las políticas de límites también se utilizaron para acotar el consumo de recursos que podía hacer un usuario del servidor. Los recursos que se limitaron fueron la cantidad de procesos simultáneos, la cantidad de memoria utilizada y el tiempo máximo de ejecución de cada proceso.

Con esas restricciones no se alteraba en uso normal del sistema ya que los valores estaban contemplado para eso. Ante un problema con un programa o una intención del usuario de producir un daño a través de una denegación de servicio el servidor seguiría andando sin perjudicar al resto de los usuarios.

Otra parte importante de la puesta en marcha era asegurar una forma de que siempre un usuario correspondiera a una máquina. La solución de este problema se realizó mediante un script que se ponía en el proceso de validación del usuario en el programa gdm que es el que nos proveía el login en el entorno gráfico.

Durante el periodo de puesta en marcha surgieron varios problemas más, entre ellos el acceso a la disquera que estaba en la estación desde el servidor. La primera solución consistía en un programa propio que mandaba comando a la disquera y hacía la transferencia de archivos vía NFS. Ese método resultó ser poco confiable.

El software libre de nuevo proveyó la solución con un programa llamado floppyd que es parte de la mtools. Con eso se pudo poner en funcionamiento las disqueteras.

Las eproms de booteo que tenían las máquinas no servía para cargar el kernel de linux vía red por lo que al principio se booteaba con disquette. Poco tiempo después utilizando el software creado por el proyecto ethboot se pudieron armar un cargador que entrara en las eproms viejas y que permitiera cargar así automáticamente el kernel sin depender de algún otro medio.

Análisis de costos y beneficios

La discusión sobre este tema parece infinita. Hay dos grandes bandos formados, los que están a favor y los que están en contra de los clientes delgados. Casi todos cometen el error de tomar los parámetros de una tecnología y extrapolarlos a la otra tecnología.

En una red nueva donde hay que adquirir todo el hardware y donde no se sabe que tipo de infraestructura se va a necesitar, los estudios de costos indican que en un gran número de casos es más conveniente el modelo de clientes gordos. Aunque hay casos particulares donde no es cierto, como es en lugares donde el uso de la computadora está restringido a un grupo reducido de programas, generalmente propios de la empresa.

En redes que ya existen y que han quedado desactualizadas por el paso del tiempo, el modelo de cliente delgado muestra que es posible realizar una reducción muy importante en el costo de mantenimiento. El poder reciclar los equipos hacen que estos tengan una vida útil que puede duplicar o triplicar los tiempos normales y por ende que no se tengan que comprar equipamiento nuevo.

La complejidad del modelo de cliente delgado está en el servidor. Los clientes son fácilmente

intercambiables y a veces ni requieren intervención. La administración está concentrada en los servidores. En el modelo de cliente gordo la complejidad esta distribuida entre los servidores y las estaciones. Como en muchos lugares la administración de la estación la debe hacer el propio usuario, este es un costo oculto poco valorados.

Otro de los errores que se cometen es suponer que se va a utilizar el mismo software e incluir costos por licenciamiento que no necesariamente tiene que ir, sobre todo cuando lo que estamos usando es software libre.