



Universidad Luterana Salvadoreña

Cátedra: Redes I

Docente: Ing. Manuel de Jesús Flores

Evaluación: Perfil de proyecto

Fecha de entrega: 28/03/2015

“Linux Terminal Server Project (LTSP)”

Estudiante:

Carnet	Apellidos	Nombres	Participación
AE1106004	Domínguez Parada	Luis Antonio	100%
GM02110987	Granados Maradiaga	Elmer Alexander	100%

INDICE

Introducción	4
Objetivos	5
Objetivo general	5
Objetivo específico	5
Diagrama de red	6
Marco teórico	7
LTSP	7
PXE.....	8
Funcionamiento	8
Protocolo	8
Servidor de arranque	8
TFTP	9
DHCP.....	9
Puertos	10
Funciones de DHCP	10
NFS	10
OPENSsh	10
SSH.....	11
Sistema de ventanas X	11
Chroot.....	11
XDMCP.....	12
Apoptes.....	12
Descripción del proyecto.....	13
Como funciona LTSP	14
Requisitos	15
Cliente Ligero.....	15
Servidor LTS.....	15
Construcción del proyecto	16
Instalación y configuración:	16
Instalación de epoptes	19
Hardware de red	20

Escenario de pruebas	20
Buenas prácticas.....	22
Factibilidad del proyecto.....	24
Presupuesto	24
Factibilidad Operativa	24
Factibilidad Técnica.....	24
Factibilidad Legal.....	25
Conclusión	25
Bibliografía	26

Introducción

El Proyecto Linux Terminal Server añade soporte de cliente ligero para servidores Linux. LTSP es una solución flexible y rentable que se faculta escuelas, empresas y organizaciones de todo el mundo para instalar y desplegar fácilmente los clientes ligeros.

Nuevos clientes ligeros y PC heredados por igual se pueden utilizar para navegar por Internet, enviar correo electrónico, crear documentos, y ejecutar otras aplicaciones de escritorio. Puesto que los clientes ligeros tienen un número limitado de tareas a realizar, el hardware utilizado por ellos puede ser pequeño y barato. El costo de mantenimiento de los clientes ligeros es básicamente nulo. Ellos duran más tiempo porque no tienen almacenamiento con partes que se mueven como pueden ser los discos duros. Si se avería un cliente ligero no se pierde la información, porque ésta está almacenada en el servidor. Simplemente se cambia el cliente por otro y se continúa trabajando. Si el cliente ligero es robado o se tira a la basura la información no terminara en manos de un tercero.

En el servidor terminal se ejecutan todas las aplicaciones y contiene todos los datos. Todo el mantenimiento habitual (actualizaciones de software, administración) se lleva a cabo en el servidor terminal. El número de clientes ligeros que un servidor terminal puede soportar es proporcional a la potencia del servidor. Dado que GNU/Linux hace uso eficiente de los recursos, puede soportar un número sorprendente de clientes ligeros con una máquina que sería considerada como potente para un sistema mono-usuario ejecutando otros sistemas operativos.

LTSP no sólo mejora el coste total de propiedad (TCO), pero lo más importante, proporciona mayor valor en soluciones informáticas tradicionales.

Estaciones de trabajo LTSP pueden ejecutar aplicaciones de servidores Linux y Windows. Clientes ligeros basados en Linux han demostrado ser extremadamente fiable, porque la manipulación y los virus son prácticamente inexistentes.

Objetivos

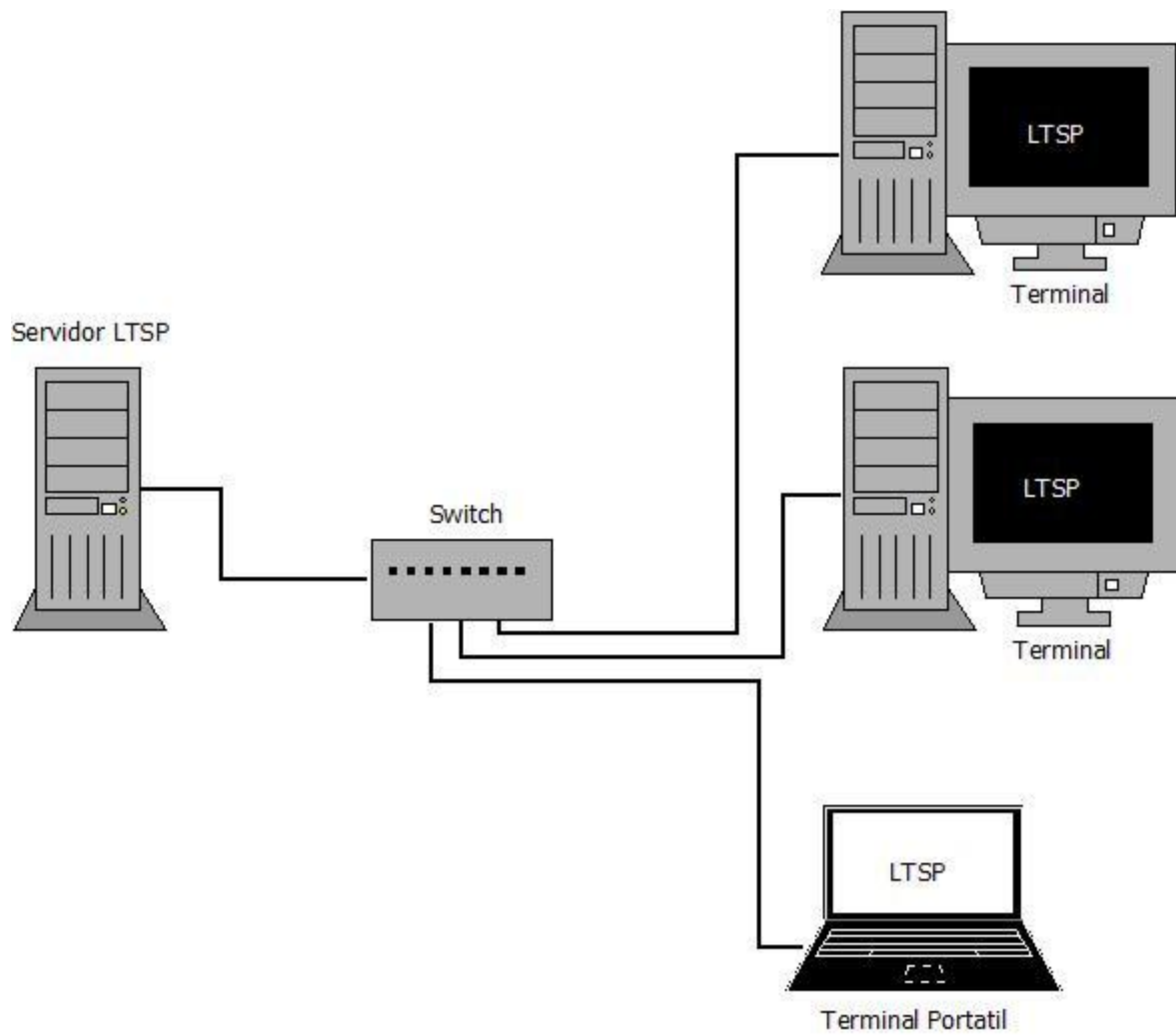
Objetivo general

- Investigar sobre el proyecto de **Linux Terminal Server Project**, configuración, demostración e implementación de clientes.

Objetivo específico

- Comprender como funciona las tecnologías implicadas en el proyecto
- Dar a conocer las ventajas de LTSP y en qué caso se podría utilizar
- Explicar en clase los pasos a seguir para implementar el proyecto

Diagrama de red



Marco teórico

LTSP

Linux Terminal Server Project o LTSP son un conjunto de aplicaciones servidores que proporcionan la capacidad de ejecutar Linux en computadores de pocas prestaciones de velocidad o de bajo costo, permitiendo reutilizar equipos que actualmente resultan obsoletos debido a los altos requisitos que piden los sistemas operativos. LTSP se distribuye bajo licencia GNU GPL de software libre. La última versión estable es la 5.0 (10 de marzo del 2007).

El sistema de funcionamiento del LTSP consiste en repartir por medio de la red el núcleo Linux que es ejecutado por los clientes y que posteriormente ejecutarán secuencias de scripts típicos de una mini distribución. Los clientes podrán acceder a las aplicaciones por medio de una consola textual o por un servidor gráfico que se comparte utilizando el protocolo XDMCP.

Actualmente uno de los campos donde se utiliza bastante LTSP es en la educación, debido a su bajo costo de implantación que suele tener.

LTSP ha servido de base para varias Distribuciones Linux, la más destacada es K12LTSP (basada en Fedora Core), también se encuentran Deworks, Edubuntu, Skolelinux y Trisquel GNU/Linux en su versión para educación.

Actualmente, la compatibilidad de este servidor de terminales se ha extendido a todas las plataformas Linux de uso común, y su rendimiento y capacidad ha mejorado con la última versión.

Otro uso, aunque con mayor complejidad de implantación, es para el manejo y gestión de estaciones de trabajo de ofimática para empresas u otras aplicaciones que no se basen en artes gráficas o cualquier aplicación que requiera alto rendimiento gráfico.

Otros sistemas de clientes delgados (como Microsoft Windows[®] Terminal Services) exige que cada cliente tenga el software que arranca el sistema para poder conectarse al servidor terminal. Esto podría ser un sistema completo o un sistema operativo mínimo que sólo ofrece una interfaz para conectarse con el servidor. Sistemas como éste generalmente requieren más mantenimiento y administración, ya que el software que arranca el cliente ligero puede llegar a estar corrupto o contengan errores que requieren atención.

PXE

"Preboot eXecution Environment (PXE) (Entorno de ejecución de prearranque), es un entorno para arrancar e instalar el sistema operativo en ordenadores a través de una red, de manera independiente de los dispositivos de almacenamiento de datos disponibles (como discos duros) o de los sistemas operativos instalados.

En nuestro caso PXE se utiliza para cargar una minimagen de Linux que no se instalara en el equipo destino. Etherboot o Yaboot se utilizan para lo mismo en este sistema para obtener del servidor la minimagen de Linux.

Funcionamiento

El firmware del cliente trata de encontrar un servicio de redirección PXE en la red para recabar información sobre los servidores de arranque PXE disponibles. Tras analizar la respuesta, el firmware solicitará al servidor de arranque apropiado el *file path* de un *network bootstrap program* (NBP), lo descargará en la memoria RAM del ordenador mediante TFTP, probablemente lo verificará, y finalmente lo ejecutará. Si se utiliza un único NBP para todos los clientes PXE se puede especificar mediante BOOTP sin necesidad de un proxy DHCP, pero aún será necesario un servidor TFTP.

Protocolo

El protocolo PXE consiste en una combinación de los protocolos DHCP y TFTP con pequeñas modificaciones en ambos. DHCP es utilizado para localizar el servidor de arranque apropiado, con TFTP se descarga el programa inicial de bootstrap y archivos adicionales.

Para iniciar una sesión de arranque con PXE el firmware envía un paquete de tipo DHCPDISCOVER extendido con algunas opciones específicas de PXE al puerto 67/UDP (puerto estándar del servicio DHCP). Estas opciones indican que el firmware es capaz de manejar PXE, pero serán ignoradas por los servidores DHCP estándar.

Servidor de arranque

Para contactar con cualquier servidor de arranque PXE el firmware debe obtener una dirección IP y el resto de información de un único paquete DHCP OFFER extendido. Tras elegir el servidor de arranque PXE apropiado el firmware envía un paquete DHCPREQUEST extendido mediante multicast o unicast al puerto 4011/UDP o broadcast al puerto 67/UDP. Este paquete contiene el servidor de arranque PXE y la capa de arranque PXE, permitiendo ejecutar múltiples tipos de servidores de arranque mediante un único *daemon* (o programa) de arranque. El paquete DHCPREQUEST extendido también puede ser un paquete DHCPINFORM.

Si un servidor de arranque PXE recibe un paquete DHCPREQUEST extendido como el descrito anteriormente y si está configurado para el tipo de servidor de arranque PXE y la arquitectura de cliente solicitado, debe responder devolviendo un paquete DHCPACK extendido con opciones específicas de PXE.

El contenido más importante de un paquete DHCPACK extendido es:

- el *file path* completo para descargar el NBP vía TFTP
- el tipo de servidor de arranque PXE y la capa de arranque PXE
- la configuración multicast TFTP, si debe utilizarse multicast TFTP

Un servidor de arranque PXE debe soportar *Boot Integrity Services* (BIS). BIS permite al cliente PXE verificar los NBPs descargados mediante un archivo de checksum que es descargado desde el mismo servidor de arranque que el NBP.

TFTP

"TFTP son las siglas de Trivial file transfer Protocol (Protocolo de transferencia de archivos trivial). Es un protocolo de transferencia muy simple semejante a una versión básica de FTP. TFTP a menudo se utiliza para transferir pequeños archivos entre ordenadores en una red, como cuando un terminal X Window o cualquier otro cliente ligero arranca desde un servidor de red." (Wikipedia)

- Utiliza UDP (en el puerto 69) como protocolo de transporte (a diferencia de FTP que utiliza los puertos 20 y 21 TCP).
- No puede listar el contenido de los directorios.
- No existen mecanismos de autenticación o cifrado.
- Se utiliza para leer o escribir archivos de un servidor remoto.
- Soporta tres modos diferentes de transferencia, "netascii", "octet" y "mail", de los que los dos primeros corresponden a los modos "ascii" e "imagen" (binario) del protocolo FTP.

DHCP

"DHCP (siglas en inglés de Dynamic Host Configuration Protocol - Protocolo de configuración dinámica de host) es un protocolo de red que permite a los clientes de una red IP obtener sus parámetros de configuración automáticamente. Se trata de un protocolo de tipo cliente/servidor en el que generalmente un servidor posee una lista de direcciones IP dinámicas y las va asignando a los clientes conforme éstas van estando libres, sabiendo en todo momento quién ha estado en posesión de esa IP, cuánto tiempo la ha tenido y a quién se la ha asignado después."(Wikipedia)"

Puertos

67/UDP (servidor)

68/UDP (cliente)

Funciones de DHCP

Un servidor DHCP puede proveer de una configuración opcional al dispositivo cliente.

- Dirección del servidor DNS
- Nombre DNS
- Puerta de enlace de la dirección IP
- Dirección de Publicación Masiva (*broadcast address*)
- Máscara de subred
- Tiempo máximo de espera del ARP (*Protocolo de Resolución de Direcciones* según siglas en inglés)
- MTU (*Unidad de Transferencia Máxima* según siglas en inglés) para la interfaz
- Servidores NIS (*Servicio de Información de Red* según siglas en inglés)
- Dominios NIS
- Servidores NTP (*Protocolo de Tiempo de Red* según siglas en inglés)
- Servidor SMTP
- Servidor TFTP
- Nombre del servidor WINS

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. 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).¹ El protocolo NFS está incluido por defecto en los Sistemas Operativos UNIX y la mayoría de distribuciones Linux.

Puertos que utiliza:

nfs 2049/tcp nfsd
nfs 2049/udp nfsd

OPENSSSH

OpenSSH (Open Secure Shell) es un conjunto de aplicaciones que permiten realizar comunicaciones cifradas a través de una red, usando el protocolo SSH. Fue creado como

una alternativa libre y abierta al programa Secure Shell, que es software propietario. El proyecto está liderado por Theo de Raadt, residente en Calgary.

SSH

SSH (Secure SHell, en español: intérprete de órdenes segura) es el nombre de un protocolo y del programa que lo implementa, y sirve para acceder a máquinas remotas a través de una red. Permite manejar por completo la computadora mediante un intérprete de comandos, y también puede redirigir el tráfico de X para poder ejecutar programas gráficos si tenemos ejecutando un Servidor X (en sistemas Unix y Windows).

Además de la conexión a otros dispositivos, SSH nos permite copiar datos de forma segura (tanto archivos sueltos como simular sesiones FTP cifradas), gestionar claves RSA para no escribir claves al conectar a los dispositivos y pasar los datos de cualquier otra aplicación por un canal seguro tunelizado mediante SSH.

Sistema de ventanas X

El Sistema de Ventanas X (en inglés, *X Window System*) es un software que fue desarrollado a mediados de los años 1980 en el MIT para dotar de una interfaz gráfica a los sistemas Unix. Este protocolo permite la interacción gráfica en red entre un usuario y una o más computadoras haciendo transparente la red para éste. Generalmente se refiere a la versión 11 de este protocolo, X11, el que está en uso actualmente. X es el encargado de mostrar la información gráfica de forma totalmente independiente del sistema operativo.

X fue diseñado primariamente para implementar clientes ligeros, donde mucha gente usaba simultáneamente la capacidad de procesamiento de un mismo computador trabajando en tiempo compartido. Cada persona usaba un terminal en red que tenía capacidades limitadas para dibujar la pantalla y aceptar la entrada del usuario. Debido a la ubicuidad del soporte para el software X en Unix, es usado en los computadores personales incluso cuando no hay necesidad del tiempo compartido.

Chroot

Un **chroot** en Unix sistemas operativos es una operación que cambia la aparente directorio raíz para el proceso en ejecución actual y sus hijos . Un programa que se ejecuta en un entorno tan modificado no puede nombrar (y por lo tanto normalmente no acceder) archivos fuera del árbol de directorio designado. El término "chroot" puede

referirse a la chroot (2) llamada al sistema o el chroot (8) Programa de envoltura. El entorno modificado se llama un "chroot la cárcel".

XDMCP

XDMCP (siglas de "X Display Manager Control Protocol", "Protocolo de Control de Administrador de la Pantalla X" en castellano) es un protocolo utilizado en redes para comunicar un ordenador servidor que ejecuta un sistema operativo con un gestor de ventanas basado en X, con el resto de clientes que se conectarán a éste con propósitos interactivos.

Un X Display Manager (administrador de pantalla X) mantiene los procesos del servidor X activos en él, conectándolo a una pantalla física y proporcionando un cuadro de diálogo que permita iniciar sesión en el mismo a todos los ordenadores que lo requieran. Escucha el User Datagram Protocol (UDP) en el puerto 177 y responde a peticiones de tipo *QUERY* y *BROADCAST_QUERY* enviando un paquete tipo *WILLING* al equipo que le contactó.

Apopes

Epopetes (Επόπτης Una palabra griega que significa supervisor) es un código abierto de gestión de laboratorio de computación y una herramienta de monitoreo. Permite la transmisión de la pantalla y la supervisión, ejecución remota de comandos, envío de mensajes, la imposición de restricciones, como el bloqueo de pantalla o el sonido silenciar los clientes y mucho más!

Se puede instalar en Ubuntu , Debian y openSUSE laboratorios basados que pueden contener *cualquier* combinación de las siguientes: LTSP servidores, delgadas y gordas clientes, servidores, estaciones de trabajo no LTSP independientes, NX o XDMCP clientes etc. Parches para otras distros son bienvenidos.

Es una reescritura parcial de una aplicación más antiguo llamado SCH-scripts , que se ha utilizado con éxito el año pasado en cerca de 250 escuelas griegas. Epoptes ya tiene un montón de características, pero se han previsto mucho más, y está siendo desarrollado en un muy rápido ritmo . Ha mantenido por profesores y estudiantes de TI, por lo que debe estar alrededor durante mucho tiempo!

Epoptes usa el puerto 789 y algunos puertos aleatorios (entre 1024 y 65535) por lo que tenemos que permitir el tráfico a esos puertos en el servidor, podemos hacerlo en forma gráfica con gufw o desde la consola (considerar que si gufw está habilitado se ignoran las reglas guardadas desde la consola en /etc/iptables/rules.v4).

Descripción del proyecto

Este proyecto está basado en la implementación de un sistema LTSP, usando el sistema operativo Debian Jessie, el proyecto está enfocado en que más de 5 terminales puedan trabajar bajo el mismo entorno por medio de un servidor que le proveerá todas las aplicaciones y herramientas necesarias para utilizar una terminal o cliente ligero.

LTSP es una colección de software que convierte una instalación Gnu/Linux normal en un servidor de terminal. Esto permite usar clientes ligeros de baja potencia y bajo costo (o hardware viejo que ya se tenga) para ser utilizados como terminales para armar una red servidor-clientes ligeros. LTSP es único entre otros sistemas de clientes ligeros porque es considerado por muchos como el más fácil de mantener. LTSP, no requiere software del lado del cliente. Se requiere sólo una interfaz de red PXE , que muchos clientes ligeros y PCs ya tienen incorporado. Esto significa que usted no necesita absolutamente de ningún medio físico de almacenamiento (disco duro, compacto-flash, etc) para que su cliente ligero arranque un LTSP. Esto reduce significativamente la cantidad de administración necesaria para mantener su red en funcionamiento.

En el servidor LTSP, un ambiente chroot es iniciado con un sistema operativo Linux y un ambiente X mínimos.

Cuando un cliente es cargado desde un dispositivo de arranque local (como un disco duro, CD-ROM o disco USB), se carga un pequeño núcleo Linux desde ese dispositivo que inicializa el sistema y todos los periféricos que reconozca. Cuando está configurado para network booting (carga por red), con los mecanismos Etherboot, Preboot Execution Environment (PXE) o NetBoot, el cliente primero solicita su propia dirección IP y la dirección IP para el servidor LTSP por medio de DHCP y carga el núcleo Linux de una imagen Linux pre-configurada en el servidor LTSP vía el servicio Trivial File Transfer Protocol (TFTP) que funciona en el servidor LTSP.

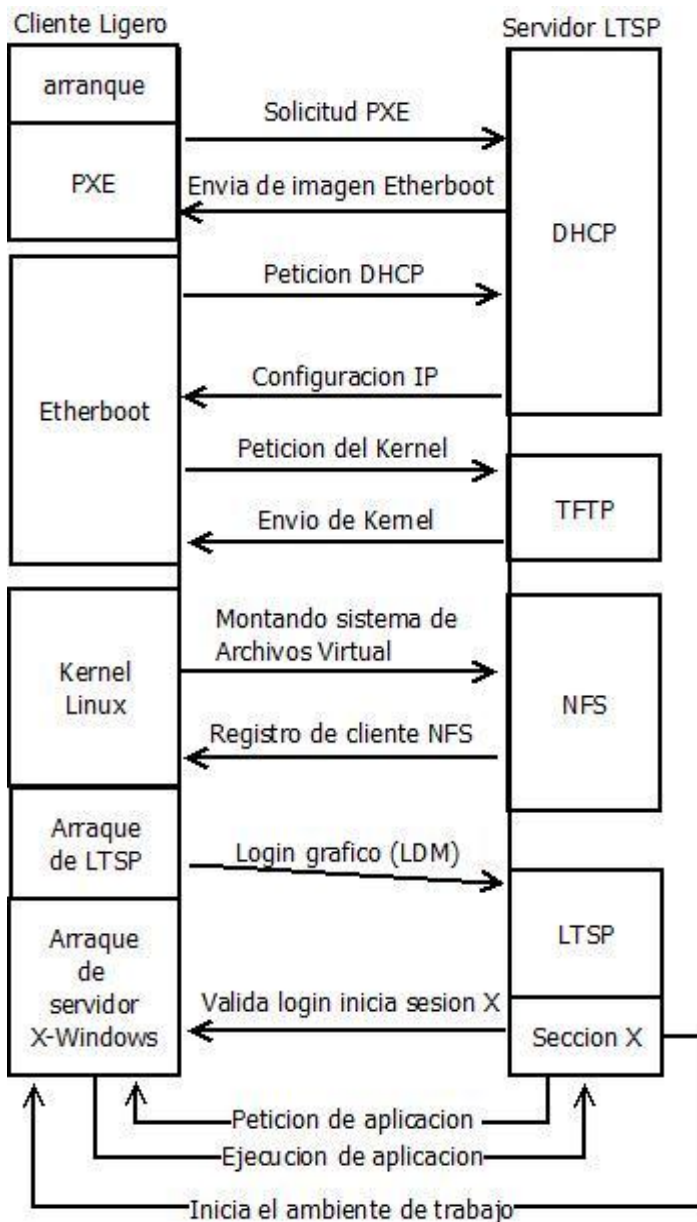
Durante este proceso el cliente hace una (nueva) solicitud DHCP para la dirección IP del servidor LTSP y la ruta a su ambiente chroot. Cuando esta información es recuperada, el cliente monta la ruta en su sistema de archivos raíz vía el servicio Network File System (NFS) que corre en el servidor LTSP.

El cliente carga Linux del sistema de ficheros raíz montado NFS y finalmente inicia el sistema de ventanas X. El cliente se conecta con el manejador de login XDMCP en el servidor LTSP.

En caso del nuevo setup del MueKow (LTSP 5), el cliente primero construye un túnel SSH hacia el ambiente X del servidor LTSP, a través del cual iniciará al manejador de login del LDM (en el servidor LTSP).

Desde este punto adelante, todos los programas son iniciados en el servidor LTSP, pero mostrados en la pantalla del cliente y operados desde el cliente.

Como funciona LTSP



Requisitos

Cliente Ligero

Los requisitos a nivel de hardware son los mismos que pusimos en la tabla anterior:

- Procesador: Funciona con procesadores desde 300 MHz para un mejor rendimiento se recomienda usar procesadores superiores a 600 MHz
- Red: Posibilidad de arrancar mediante PXE, Etherboot o Yaboot.
- Tarjeta Gráfica: Al menos 2 MB de memoria gráfica.
- 128 MB de memoria RAM

Servidor LTS

Como Debian es un sistema escalable y ligero un servidor de rango medio podrá servir muchos terminales y en caso de ser necesario podrá ampliarse.

- Memoria RAM: Los requisitos de memoria RAM son como mínimo 256 MB para el sistema más 128 MB por cada usuario, aunque se recomienda 512MB para el sistema y 256MB para cada usuario. Dependiendo de la carga de trabajo que genera cada cliente se necesitara más o menos RAM por lo que sera recomendable tener una gran cantidad de memoria RAM en nuestro servidor.
- Procesador: El procesador que requiere nuestro servidor LTS deberá ajustarse a la carga de trabajo a la que lo vayamos a someter ya que el sistema se puede implementar en prácticamente cualquier procesador actual. Con un procesador I5 2500k o I7 950 podríamos dar servicio a muchos clientes.
- Disco Duro: Los discos duros deben ser lo más rápidos posible ya que todos los usuarios estarán generando operaciones de lectura/escritura sobre ellos y son la parte más lenta de cualquier ordenador actual. Podríamos instalar discos duros SSD o discos raptor de la marca Western Digital en Raid 5.

Construcción del proyecto

Instalación y configuración:

Arrancamos nuestro Linux, abrimos una consola(terminal) y como root escribimos las siguientes líneas:

1º)Actualizamos los repositorios:

```
aptitude update
```

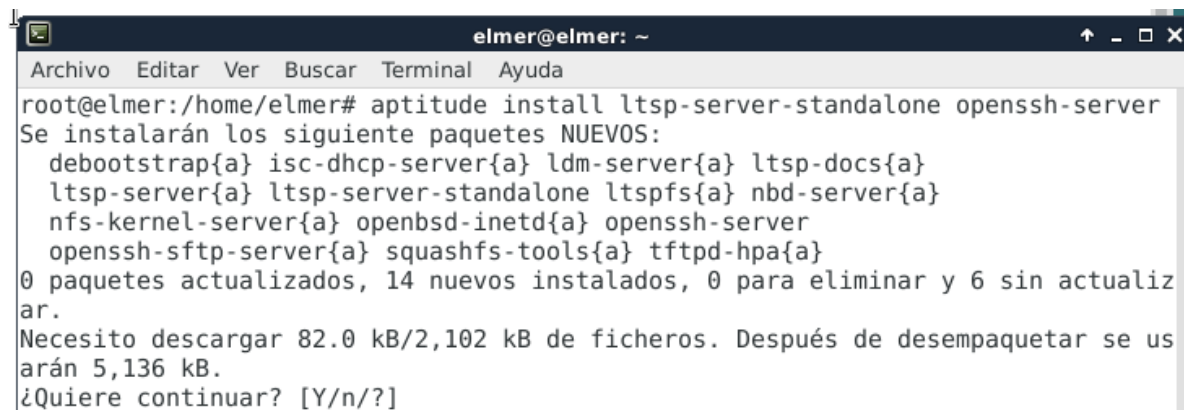
```
root@elmer:/home/elmer# aptitude update
Hit http://security.debian.org jessie/updates InRelease
Ign http://ppa.launchpad.net precise InRelease
Ign http://ftp.de.debian.org wheezy InRelease
Hit http://security.debian.org jessie/updates/main Sources
Hit http://ftp.de.debian.org wheezy Release.gpg
Des: 1 http://ppa.launchpad.net precise Release.gpg [316 B]
Hit http://security.debian.org jessie/updates/contrib Sources
Hit http://security.debian.org jessie/updates/non-free Sources
Hit http://ppa.launchpad.net precise Release
Ign http://ppa.launchpad.net precise Release
Hit http://ftp.de.debian.org wheezy Release
Hit http://http.debian.net jessie InRelease
Hit http://security.debian.org jessie/updates/main amd64 Packages
Ign http://ppa.launchpad.net precise/main amd64 Packages/DiffIndex
Hit http://ftp.de.debian.org wheezy/main amd64 Packages
Hit http://security.debian.org jessie/updates/contrib amd64 Packages
Des: 2 http://http.debian.net jessie-updates InRelease [126 kB]
88% [Packages 28.5 MB] [2 InRelease 14.3 kB/126 kB 11%] [Esperando las cabecera
```

2º) Actualizamos el sistema (opcional):

```
aptitude upgrade
```

3º)Esto instalará todo el paquete ltsp de internet:

```
aptitude install ltsp-server-standalone openssh-server
```



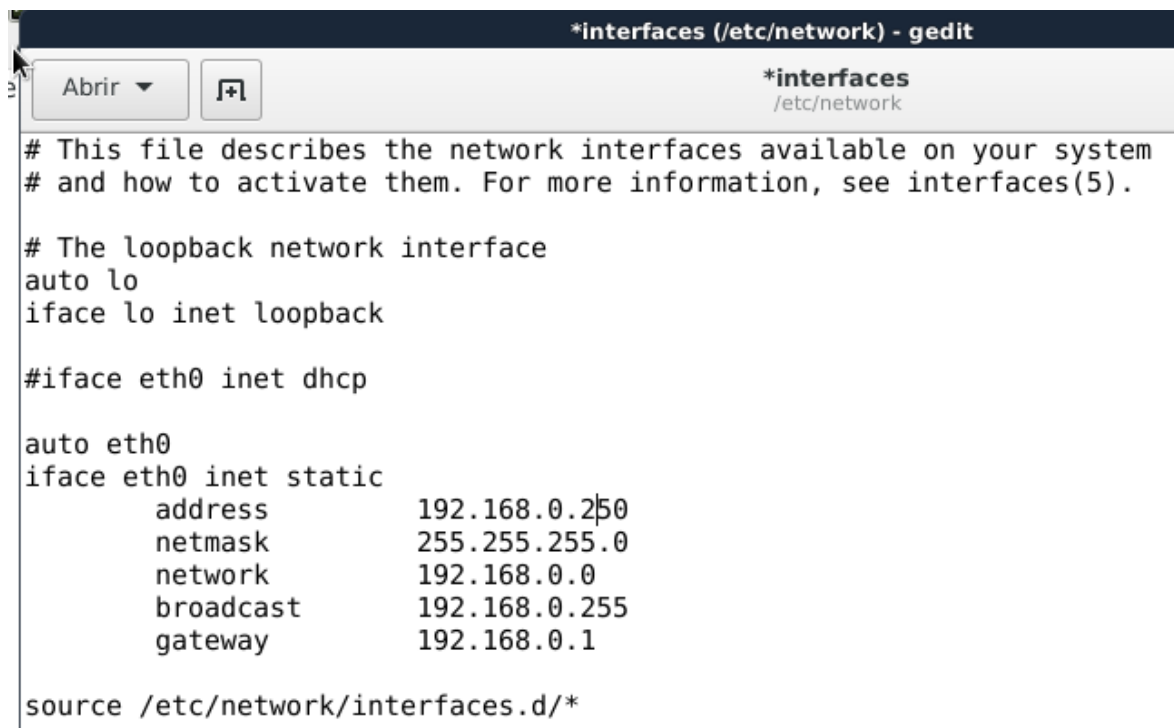
```
elmer@elmer: ~
Archivo Editar Ver Buscar Terminal Ayuda
root@elmer:/home/elmer# aptitude install ltsp-server-standalone openssh-server
Se instalarán los siguiente paquetes NUEVOS:
  debootstrap{a} isc-dhcp-server{a} ldm-server{a} ltsp-docs{a}
  ltsp-server{a} ltsp-server-standalone ltspfs{a} nbd-server{a}
  nfs-kernel-server{a} openbsd-inetd{a} openssh-server
  openssh-sftp-server{a} squashfs-tools{a} tftpd-hpa{a}
0 paquetes actualizados, 14 nuevos instalados, 0 para eliminar y 6 sin actualiz
ar.
Necesito descargar 82.0 kB/2,102 kB de ficheros. Después de desempaquetar se us
arán 5,136 kB.
¿Quiere continuar? [Y/n/?]
```


Presionamos la tecla “Y” para que continúe la instalación de todos los paquetes necesarios para que nuestro servidor LTSP funciones

4º) cuando finalice la instalación escribimos en la consola:

```
gedit /etc/network/interfaces
```

5º) y deberemos modificar este archivo para asignar una ip estática a nuestro servidor:



```
*interfaces (/etc/network) - gedit
*interfaces
/etc/network

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

#iface eth0 inet dhcp

auto eth0
iface eth0 inet static
    address          192.168.0.250
    netmask          255.255.255.0
    network          192.168.0.0
    broadcast        192.168.0.255
    gateway          192.168.0.1

source /etc/network/interfaces.d/*
```

Es nuestro proyecto el servidor tendrá la IP 192.168.0.250.

6º) Después de modificar el archivo interfaces, escribimos en la consola:

```
gedit /etc/ltsp/dhcpd.conf
```

y nos aparece un fichero de texto con la configuración por defecto de nuestro servidor, el cual tendremos que modificar de acuerdo a nuestras necesidades :

```
dhcpcd.conf (/etc/ltsp) - gedit
Abrir  Guardar
dhcpcd.conf
/etc/ltsp

#
# Default LTSP dhcpcd.conf config file.
#

authoritative;

# Rango de los clientes
subnet 192.168.0.0 netmask 255.255.255.0 {
    range 192.168.0.200 192.168.0.249;          # rango de 200 a 249
    option domain-name "ltsp.elmer";          # nombre de de dominio para los clientes
    option domain-name-servers 192.168.0.250; # DNS
    option broadcast-address 192.168.0.255;    # Direccion de broadcast
    option routers 192.168.0.1;               # ip de router
    next-server 192.168.0.250;                # ip del servidor LTSP
    get-lease-hostnames true;
    option subnet-mask 255.255.255.0;
    option root-path "/opt/ltsp/i386";
    if substring( option vendor-class-identifier, 0, 9 ) = "PXEClient" {
        filename "/ltsp/i386/pxelinux.0";
    } else {
        filename "/ltsp/i386/nbi.img";
    }
}
```

7º) Una vez tenemos todo esto hecho ejecutamos en la consola:

```
ltsp-build-client -arch i385
```

8º) Si se ha creado anteriormente la imagen se debe borrar:

```
rm -rf /opt/ltsp/i386
```

9º) la única configuración que haremos al a nuestro servidor DHCP es incluir la configuración previa hecha al archivo /etc/ltsp/dhcpcd.conf.

Para ello escribimos al final del archivo /etc/dhcp/dhcpcd.conf la siguiente línea:

```
include "/etc/ltsp/dhcpcd.conf";
```

10º) Y nos creará el cliente LTSP. Cuando haya terminado con este proceso (tarda unos minutos, dependerá de nuestro ancho de banda),

ejecutamos en la consola:

```
/etc/init.d/networking restart
```

```
/etc/init.d/isc-dhcp-server restart
```

11º) Actualizaciones de sistema:

- Cuando hacemos una actualización del sistema tenemos que ejecutar los siguientes comandos:

```
sudo ltsp-update-kernels  
sudo ltsp-update-sshkeys  
sudo ltsp-update-image
```

Instalación de epoptes

Para instalar la aplicación epoptes ejecutamos los siguientes comandos como usuario root:

1º) aptitude install epoptes

2º) agregamos a los usuarios al grupo epoptes con el comando *gpasswd -a "usuario" epoptes*

3º) instalamos epoptes-client en la imagen que bootearan los clientes

Para ello debemos entrar al chroot o imagen para los clientes.

```
#chroot /etc/ltsp/i386
```

```
/# apt-get install epoptes-client
```

4º) obtenemos los certificados de Openssh con el comando

```
/# epoptes-client -c
```

Hardware de red

1º) switch de 8 puertos 10/100 mbps

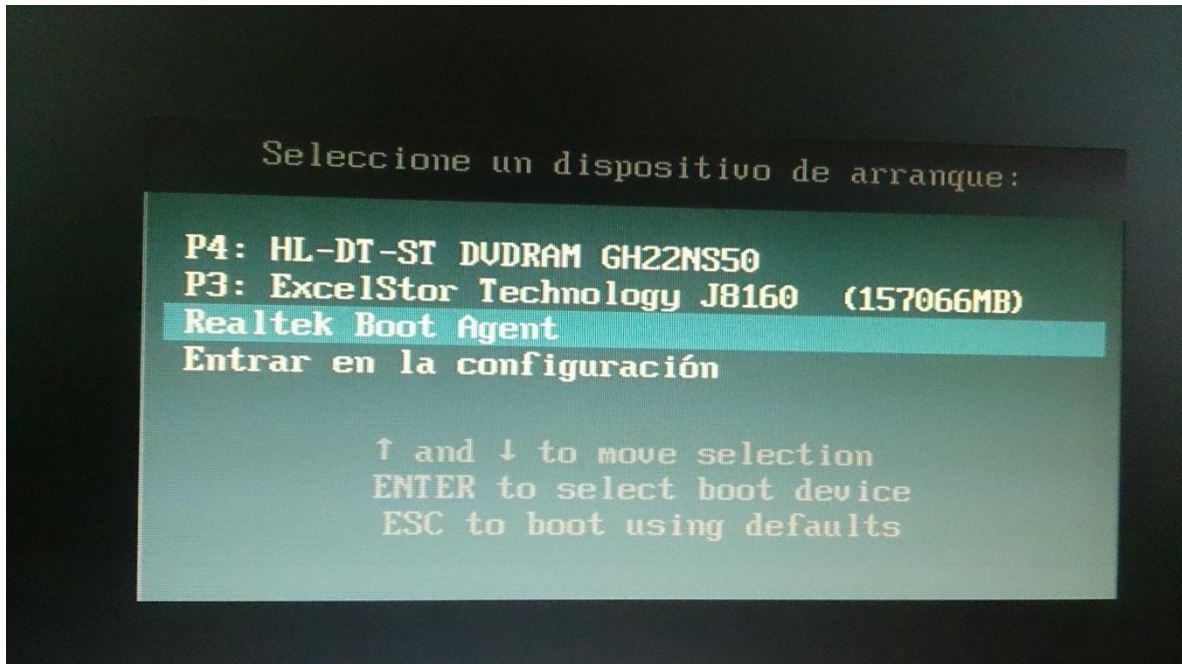


2º) cable utp con conectores rj45 tia/eia-568-b

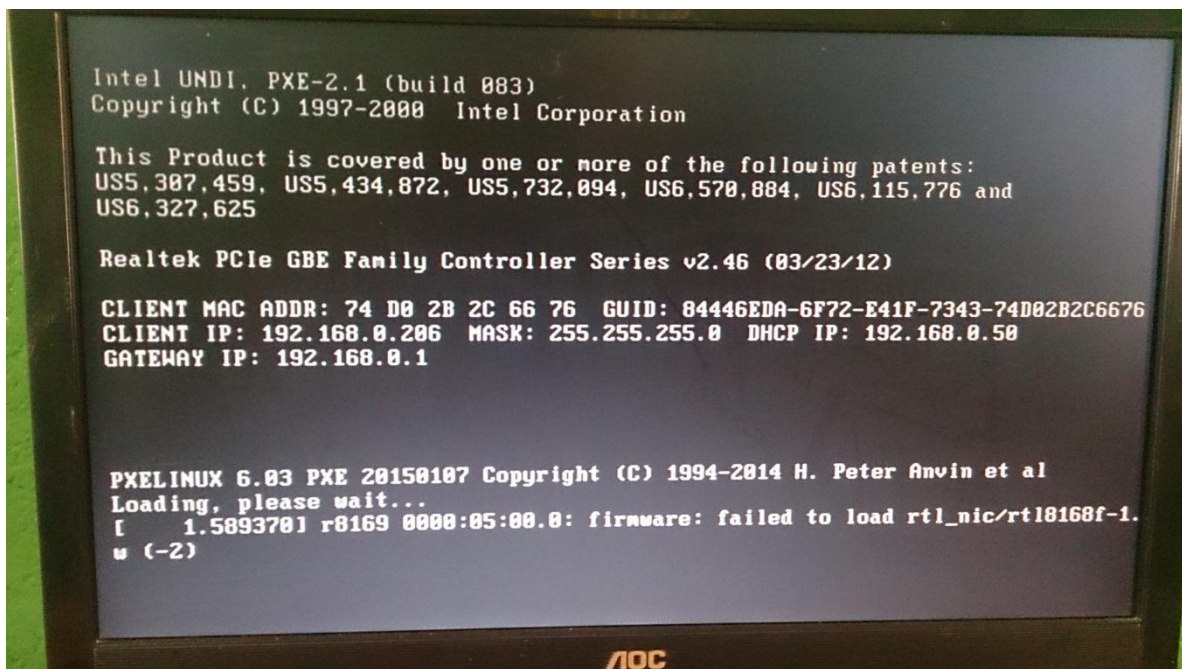


Escenario de pruebas

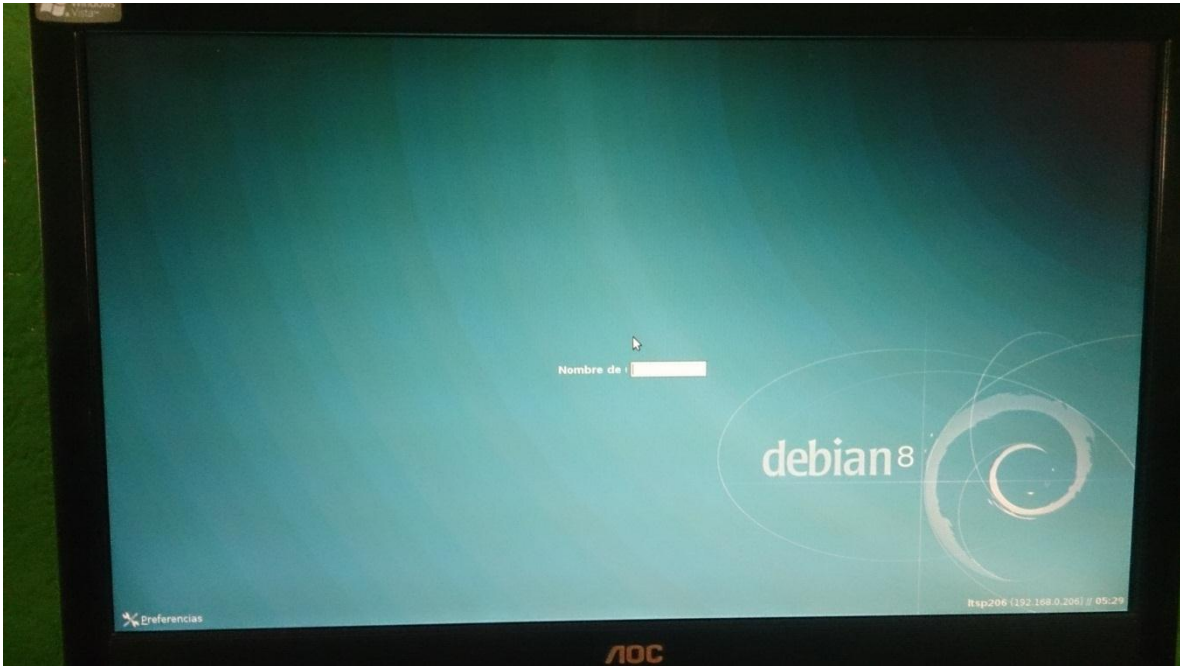
1º) configurar o cambiar el orden de arranque para que bootee desde red.



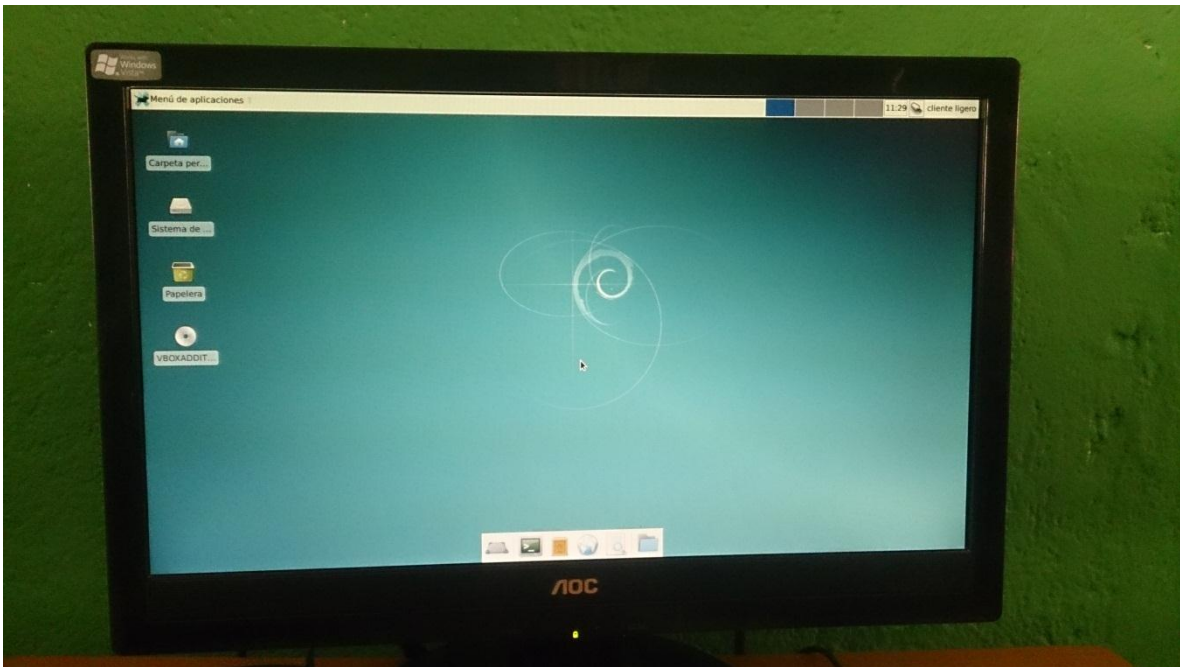
2º) Inicializado configuración ip mediante peticiones al servidor DHCP



3º) inicio de sección LDM



4º) Sección iniciada con entorno XFCE



Buenas prácticas

- 1- Instalar solo los paquetes necesarios para un buen y óptimo funcionamiento de la red y hardware de nuestro servidor
- 2- Instalar el entorno grafico más liviano para los clientes ligeros.
- 3- Crear un usuario para cada terminal, para mejor manejo de archivos y para evitar conflicto entre aplicaciones
- 4- Instalar y configurar un solo servidor DHCP para evitar conflictos en la red
- 5- Usar una red mínimo a 100 Mbps, para que la trasferencia sea optima

Factibilidad del proyecto

La factibilidad de implementar el proyecto de LTSP es muy fiable, ya que se pueden usar computadoras viejas o con pocos recursos e implementar un centro de cómputo con solo la inversión en un computador que cuente con propiedades de hardware potentes.

Presupuesto

cantidad	Descripción	costo
1	Computadora servidor (microprocesador Dual core 2.0 GHZ o más, RAM de 4 GB o más)	\$300.00
1	Placa Raspberry pi accesorios	\$55.00 \$25.00
1	Computadora vieja con pocos recursos	\$100.00
1	Sistema operativo para maquina servidor(debían)	\$0.00
1	Switch de 8 puertos	\$10.00
10	Metros de cable UTP	\$2.50
10	Conectores RJ45	\$2.00
50	Impresiones de (documento)	\$10.00
2	Honorarios de un mes de \$150	\$300.00
TOTAL		\$804.50

Observando los datos anteriores y valorando la importancia que aportara este proyecto en cualquier tipo de entidad que lo sugiera, es relativamente accesible y cómodo.

Ya que se puede implementar en un laboratorio de computo u oficina que sentralice toda su información en una sola computadora, además de que puede reutilizar equipos antiguos y seguir sacando provecho de ellos.

Factibilidad Operativa

El equipo operacional del proyecto es la ejecución y control del proyecto,

Factibilidad Técnica.

LTSP es un proyecto libre que es una opción acertada en un ambiente en que se requiera compartir el mismo sistema a varias personas con el fin de usarcé como una computadora normal, pero que todos los procesos, información y acciones realizadas se ejecuten en una sola computadora, asiendo así mucho más fácil el mantenimiento del centro de cómputo o laboratorio informático

Factibilidad Legal

El software que se utilizara será en su totalidad libre lo que quiere decir que no se violara ningún contrato de licencia. No habrá ningún contrato de alguna licencia.

Conclusión

LTSP posee licencia GPL y utiliza estándares abiertos. El hecho de que sea un proyecto de comunidad asegura un amplio soporte así como su rápido avance, debido al gran número de personas y empresas involucradas.

Además de ser GPL, LTSP es gratuito, lo que supone un ahorro importante de costos en Licencias. Incorporando a LTSP alguno de los escritorios libres, es posible añadir multitud de aplicaciones para entornos académicos, profesionales, domésticos sin costo económico.

LTSP está especialmente ideado para el aprovechamiento de PCs que han quedado obsoletos. Permite, además, la incorporación de terminales especialmente diseñados para este tipo de arquitectura. Estos equipos suponen un ahorro de hasta el 60%

A esto hay que sumar el hecho de que los requerimientos técnicos del servidor son superiores a los de otros servicios, salvo en el apartado de memoria RAM. No obstante, el costo global del servidor es mucho menor que en sistemas tradicionales de servicios centralizados usando software propietario, ya que este tipo de software, tradicionalmente, requiere más recursos.

Puesto que los procesos no se ejecutan en los terminales sino en el servidor, que además alberga el software y los datos, la administración de un sistema LTSP se reduce a la administración de un sólo equipo: "el Servidor". Desciende, por tanto, drásticamente, el número de horas que debe dedicarse a mantenimiento y configuraciones, simplifica la administración del sistema y de los datos y aumenta el control en todos los sentidos. Además de esto, se ahorra en mano de obra cualificada.

En conclusión el proyecto de LTSP, es una buena opción para ser implementado en laboratorios de cómputo con carácter educativo o profesional, ya que se reduce el costo hasta en un 50% y se tiene menores costos de administración.

Bibliografía

- «Epoptes». [En línea]. Disponible en: <http://www.epoptes.org/>. [Accedido: 30-may-2015]. [1]
- «Linux Terminal Server Project - Welcome to LTSP.org». [En línea]. Disponible en: <http://www.ltsp.org/>. [Accedido: 27-mar-2015]. [2]
- «Lts linux terminal server project-». [En línea]. Disponible en: <http://es.slideshare.net/jeysonh/ltsp-linux-terminal-server-project>. [Accedido: 27-mar-2015]. [3]
- «LTSP - Wikipedia, la enciclopedia libre». [En línea]. Disponible en: <http://es.wikipedia.org/wiki/LTSP>. [Accedido: 27-mar-2015]. [4]
- «PROYECTOS CICLOS FORMATIVOS DE INFORMATICA - Proyecto Servidor LTSP Clientes Ligeros». [En línea]. Disponible en: <https://proyectosinformatica.wikispaces.com/Proyecto+Servidor+LTSP+Clientes+Ligeros#indice>. [Accedido: 27-mar-2015]. [5]
- «Servidor Terminal Linux/7.3.3. Debian LTSP - Wikilibros». [En línea]. Disponible en: http://es.wikibooks.org/wiki/Servidor_Terminal_Linux/7.3.3._Debian_LTSP. [Accedido: 27-mar-2015]. [6]