



Universidad Luterana Salvadoreña

Cátedra: Redes I

Docente: Ing. Manuel de Jesús Flores

Evaluación: Defensa Final

Fecha de entrega: 30/05/2015

“Linux Terminal Server Project (LTSP)”

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.

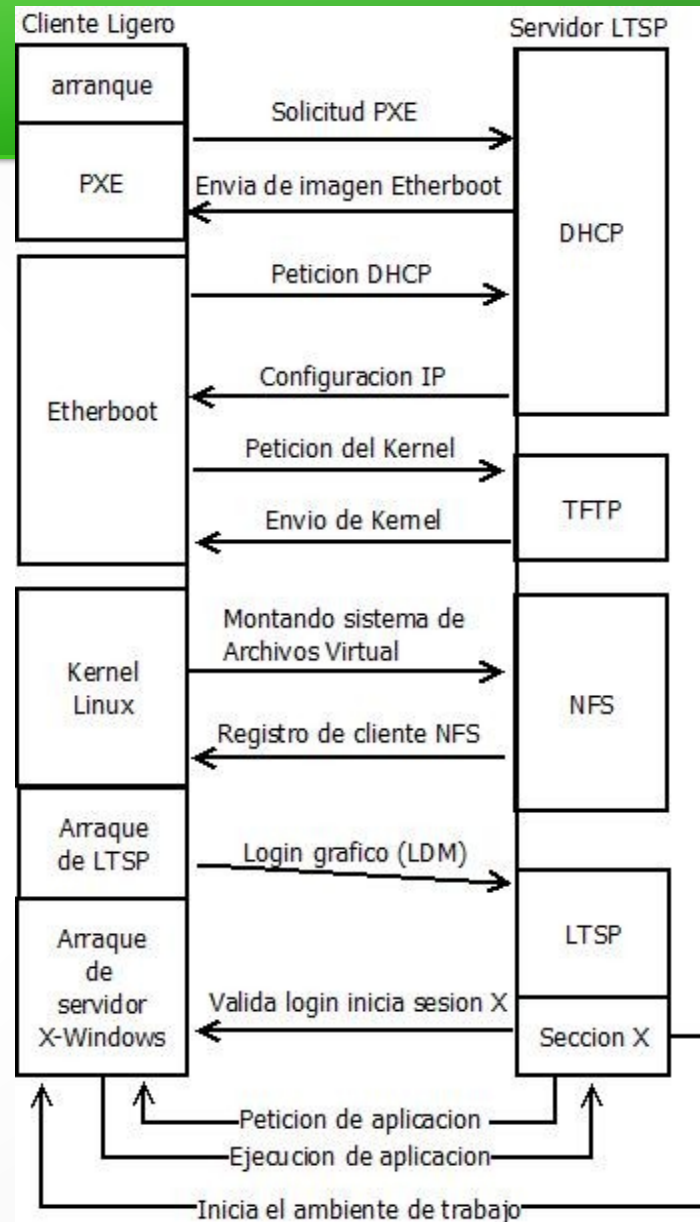
Como funciona LTSP

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).



Requisitos

Cliente Ligero

- 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

- 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

Construcción del proyecto

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 actualizar.
Necesito descargar 82.0 kB/2,102 kB de ficheros. Después de desempaquetar se usarán 5,136 kB.
¿Quiere continuar? [Y/n/?]
```

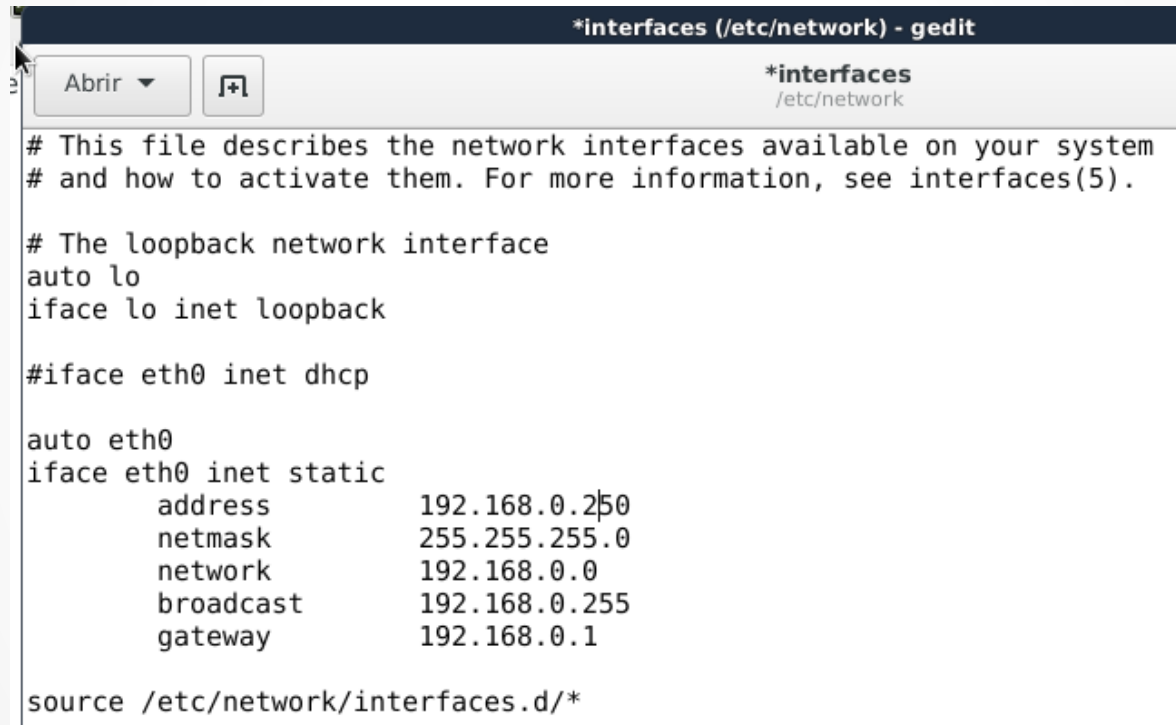
Presionamos la tecla “Y” para que continúe la instalación de todos los paquetes necesarios para que nuestro servidor LTSP funciones

Construcción del proyecto

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
# 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

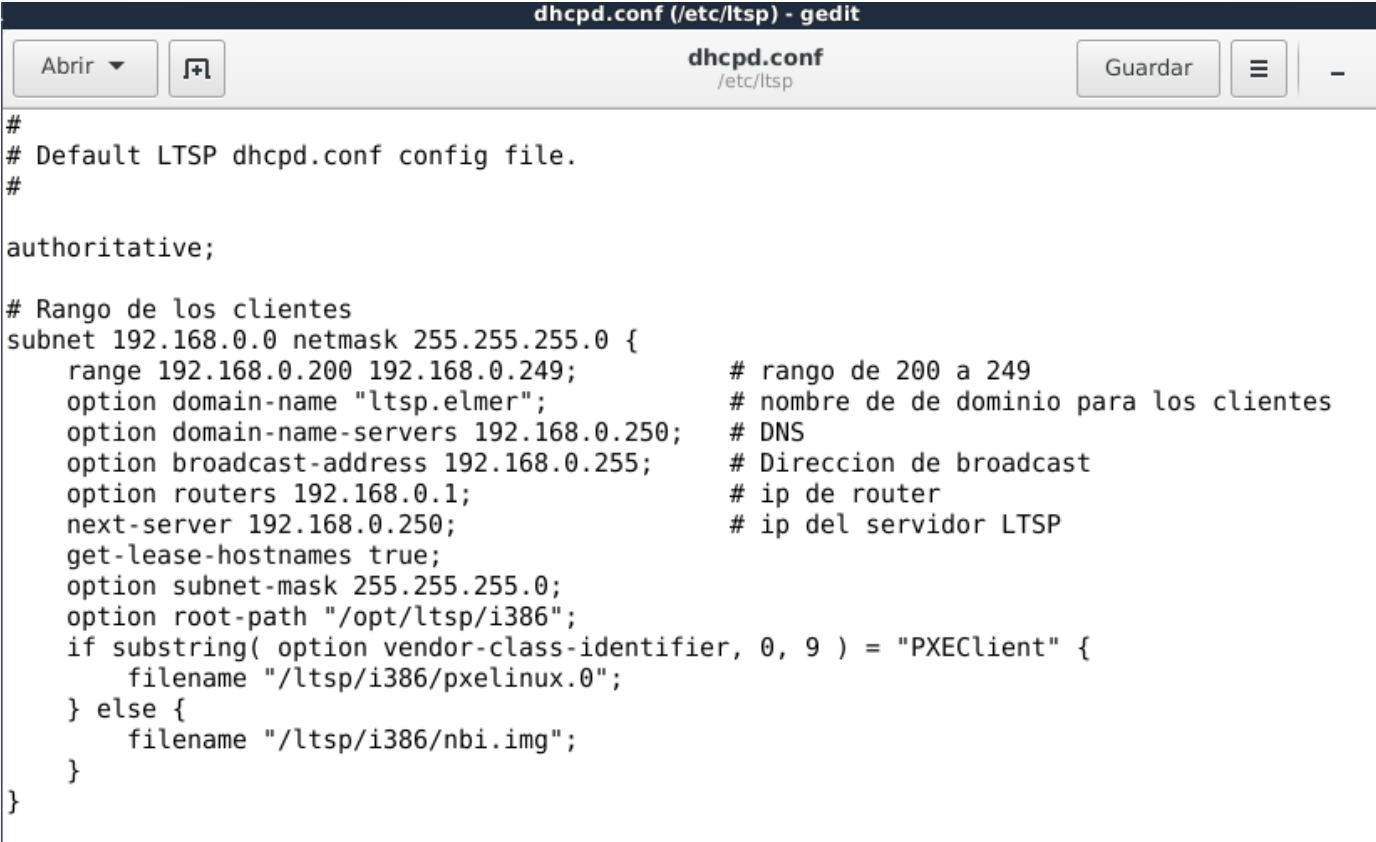
source /etc/network/interfaces.d/*
```

Es nuestro proyecto el servidor tendrá la IP 192.168.0.250.

Construcción del proyecto

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 :



```
#
# Default LTSP dhcpd.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";
    }
}
```


Construcción del proyecto

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

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

8º) 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
```

9º) Si se ha creado anteriormente la imagen se debe borrar:

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

10º) la única configuración que haremos al a nuestro servidor DHCP es incluir la configuración previa hecha al archivo `/etc/ltsp/dhcpd.conf`. Para ello escribimos al

final del archivo `/etc/dhcp/dhcpd.conf` la siguiente línea:

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

Construcción del proyecto

Instalación de eoptes

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

1°) aptitude install eoptes

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

3°) instalamos eoptes-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 eoptes-client
```

4°) obtenemos los certificados de Openssh con el comando

```
/# eoptes-client -c
```

Hardware de red

1º) switch de 8 puertos 10/100 mbps

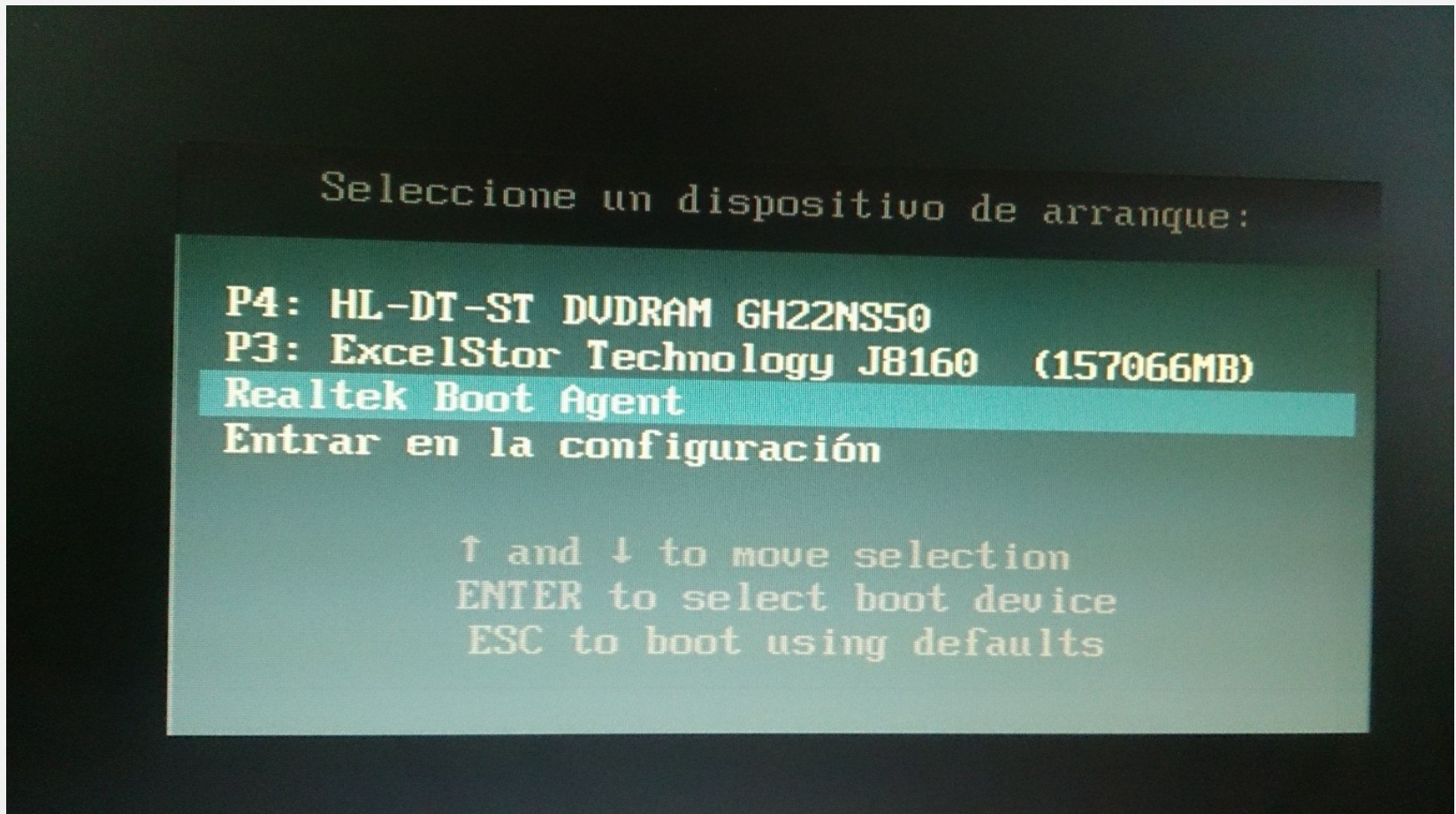


2º cable utp con conectores rj45 tia/eia-568-b



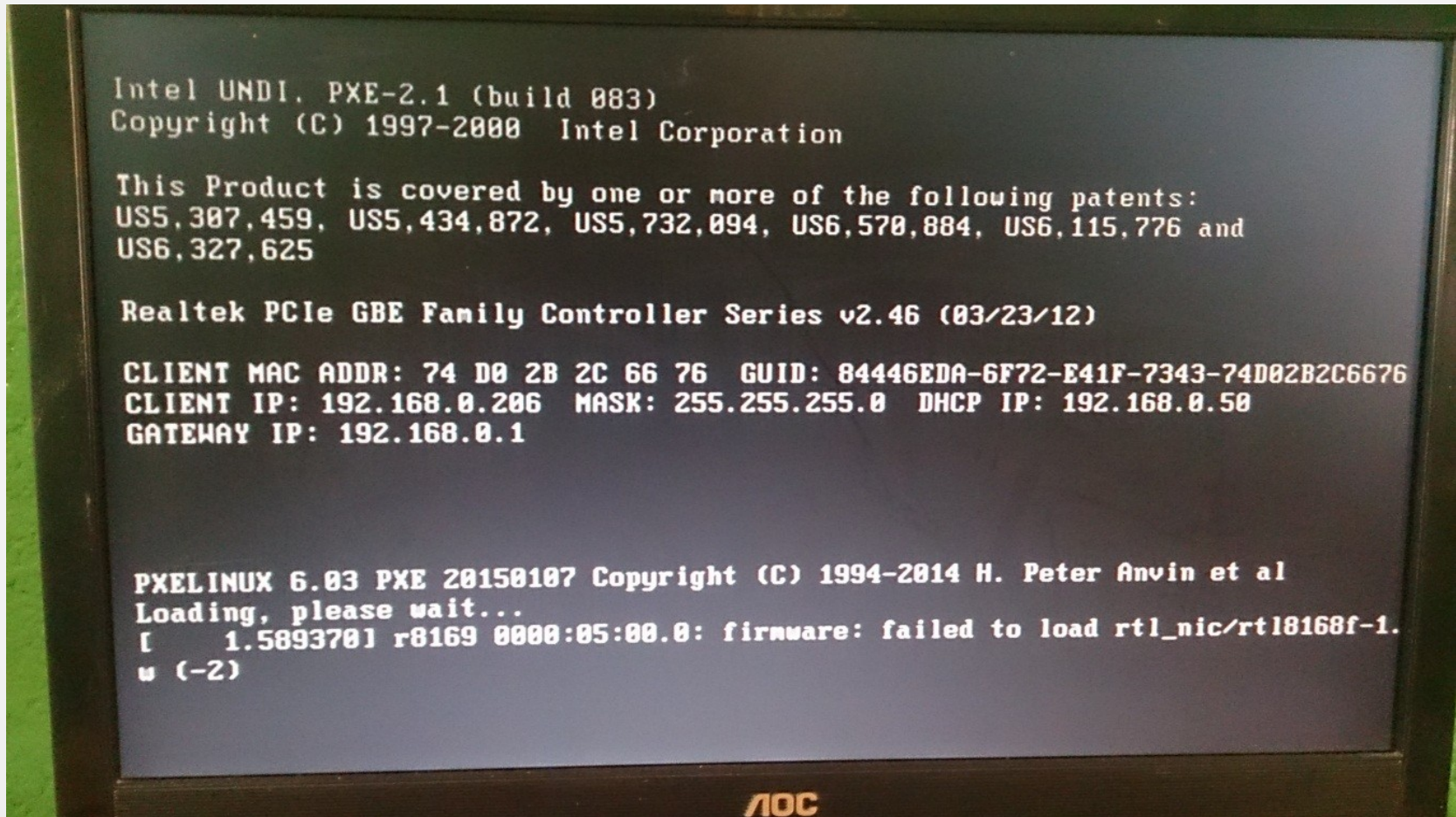
Escenario de pruebas

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



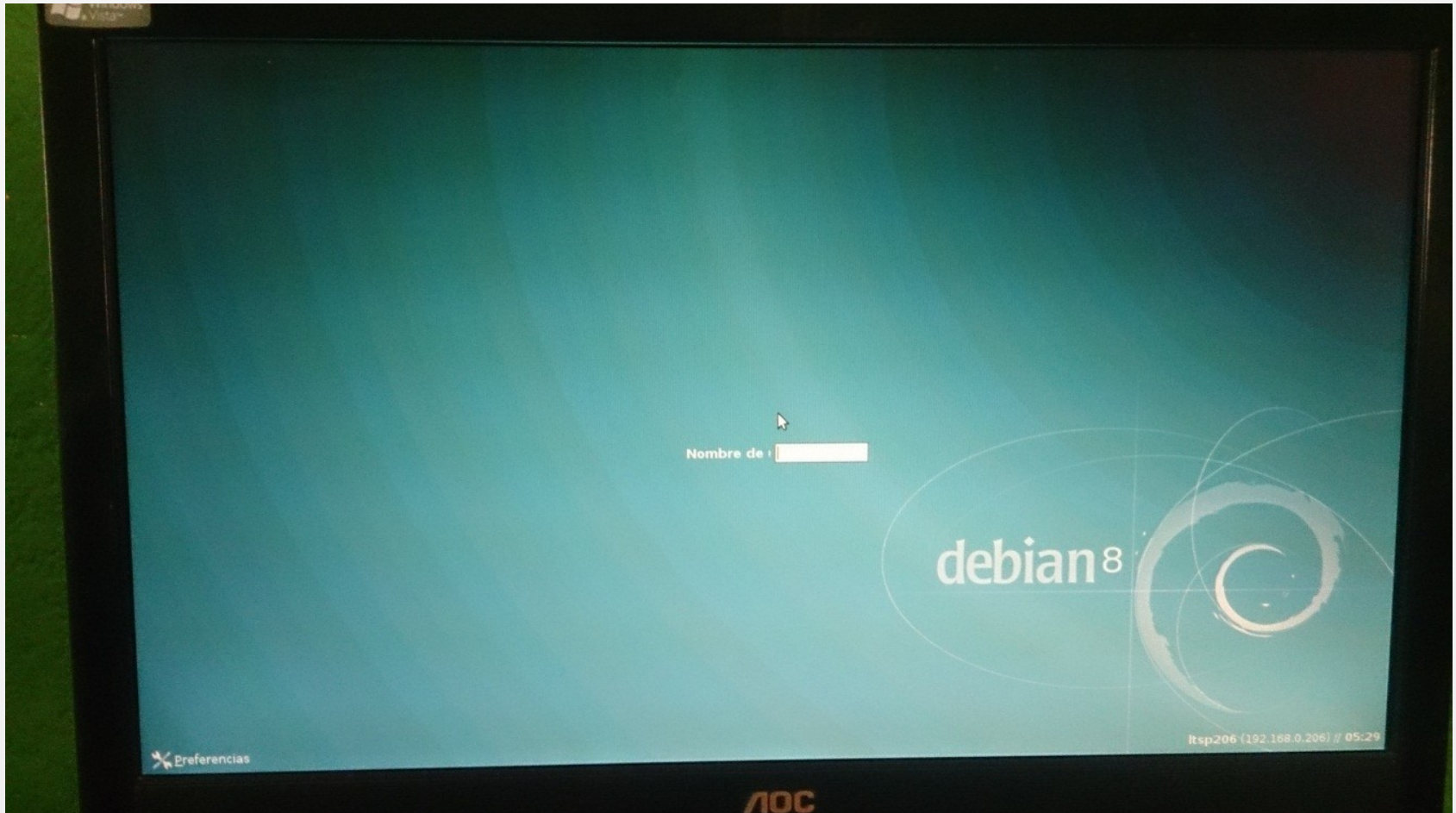
Escenario de pruebas

2º) Inicializado configuración ip mediante peticiones al servidor DHCP



Escenario de pruebas

3º) inicio de sección LDM



Escenario de pruebas

4º) Sección iniciada con entorno XFCE

