COMO CONFIGURAR UNA RED DE CLIENTES LIGEROS

Cliente Ligero: un cliente ligero no es más que una máquina que arranca por la red contra un servidor que le proporciona una sesión interactiva donde además se ejecutaran todas las tareas y programas. <u>http://es.wikipedia.org/wiki/Cliente_ligero</u>

1. Requisitos:

Para crear una red de clientes ligeros vamos a necesitar:

1.1. Servidor:

S.O Fedora Core 4. (http://fedora.redhat.com/Download/) Pxes (para generar la imagen de arranque).http://downloads.2x.com/pxes/

1.2. Cliente:

Solo necesitaremos que el equipo tenga soporte para arranque por red (PXE) o en su defecto una disquetera para etherboot.

2. Funcionamiento:

Cliente		Servidor
	Petición de IP	-
	Petición de IMAGEN	
	Envio Imagen - TFTP	
Fedora	Levanta Servidor X - XDMCP + XFS	
	Inicio de Sesión OK	

3. Instalación del servidor:

Para instalar los paquetes necesarios para configurar nuestro servidor vamos a utilizar yum (yellow dog updater modified).

En el servidor necesitamos los siguiente paquetes:

3.1. Servidor tftp (trivial ftp)

Este servidor nos ayudara a enviar las imágenes de arranque a los clientes que lo soliciten sin ningún tipo de autenticación por eso lo de trivial.

[root@adminhouse]# yum install tftp-server //servidor de tftp

Esto nos creará un directorio en el raíz llamado /tftpboot donde pxes (programa con el que crearemos la imagen) creara las imágenes de arranque, que más adelante veremos como se hace.

Tras instalar el servidor de tftp tenemos que arrancarlo en nuestra máquina servidor como servicio y para ello tendremos que editar el archivo:

[root@adminhouse]# vi /etc/xinetd.d/tftp

<pre># defaul # descri file tra # diskless</pre>	lt: off ption: The tftp server s ansfer \ protocol. The tftp prot s \	serves files using the trivial tocol is often used to boot
#	workstations, download o	configuration files to network-
aware pi	cinters, \	
#	and to start the install	lation process for some operating
systems.		
service {	tftp	
•	socket_type	= dgram
	protocol	= udp
	wait	= yes
	user	= root
	server	= /usr/sbin/in.tftpd
	server_args	= -s /tftpboot
#	disable	<mark>= true</mark>
	per_source	= 11
	cps	= 100 2
	flags	= IPv4
}	-	

Por defecto está parado por lo que tenemos que comentar la línea de disable=true y ya tendremos el servidor tftp listo.

Solo falta iniciarlo y asegurarnos que se ejecute en el init 5 y para ello iniciaremos xinetd del cual depende tftp:

[root@adminhouse]# service xinetd start //ejecutamos el servicio [root@adminhouse]# chkconfig --list | grep xinetd

Si está apagado en el nivel 5 debemos ejecutar: [root@adminhouse]# chkconfig --level 5 xinetd on

3.2. Servidor bootp ó dhcp

Ahora pasamos a instalar el servidor bootp(<u>http://es.wikipedia.org/wiki/BOOTP</u>) que al igual que tftp son servidores antiguos, por llamarlo de alguna manera, y que se encarga de asignar ip's a los clientes, en este caso vamos a instalar un servidor dhcp que soporta bootp.

[root@adminhouse]# yum install dhcp

Una vez que lo tenemos instalado tenemos que configurarlo y para ello editamos el archivo.

[root@adminhouse]# vi /etc/dhcpd.conf

```
ddns-update-style ad-hoc;
 # red y mascara
 subnet 192.168.0.0 netmask 255.255.255.0 {
range 192.168.0.100 192.168.0.254;
option routers 192.168.0.1;
                                                 // Gateway
option broadcast-address 192.168.0.255;
                                                 // DNS
option domain-name-servers 80.58.0.33;
default-lease-time 86400; # segundos (24 horas)
max-lease-time 86400;
option tftp-server-name "192.168.0.10";
next-server 192.168.0.10;
filename "/pxes/pxelinux.0";
                                  //Imagen para transferir
   # Relacion de hosts con IP fijas
 host nodo1 {
           hardware ethernet 00:0B:BB:01:37:CE;
           fixed-address 192.168.0.100;
   }
 host nodo2 {
       hardware ethernet 00:F0:19:FC:A2:3A;
       fixed-address 192.168.0.101;
 }
```

En este ejemplo hemos definido una red 192.168.0.0/24 con un rango de ip's asignables de la 100 a la 254 con una gateway y unos dns's para que nuestros equipos clientes puedan navegar, y más adelante hemos definido 2 equipos a los que le asignaremos una ip estática conociendo su MAC.

Hasta aquí podemos decir que solo hemos configurado un típico servidor dhcp para que asigne ip's a equipos conocidos o desconocidos pero lo importante son las 3 líneas en las que indicamos cual es el servidor de títp, el nombre de la imagen y el siguiente servidor al que tiene que hacerle peticiones que en este caso es la misma máquina ya que vamos a montar todos los servidores en un mismo equipo.

Decir que la ruta de la imagen es /tftpboot/pxes/pxelinux.0 aunque no es necesario indicar la carpeta del servidor tftp ya que el servidor tftp lo sabe.

Por último, al igual que hemos hecho con tftp tenemos que activar el demonio de dhcpd y asegurarnos que se activara en el próximo arranque en init 5.

```
[root@adminhouse]# service dhcpd start //activamos el demonio
[root@adminhouse]# chkconfig --list | grep dhcpd
```

3.3. Servidor de X (XDMCP)

El servidor de X (xdmcp) es el que indicaremos al crear la imagen, y claro está, el que nos proporcionara el entorno gráfico. Que tengamos una resolución mayor o menor en los clientes viene limitado por la gráfica que tenga nuestra máquina servidor, ya que la gráfica de los clientes no influye en nada.

}

Por defecto los servidores X solo escuchan peticiones locales por lo que tenemos que habilitar un servidor X para que escuche conexiones tcp, en este caso vamos a habilitar GNOME que es más ligero que KDE y para ello editamos el archivo:

[root@adminhouse]# vi /etc/X11/gdm/gdm.conf

XDMCP is the protocol that allows remote login. If you want to log into # gdm remotely (I'd never turn this on on open network, use ssh for such # remote usage that). You can then run X with -query <thishost> to log in, # or -indirect <thishost> to run a chooser. Look for the 'Terminal' server # type at the bottom of this config file. [xdmcp] # Distributions: Ship with this off. It is never a safe thing to leave # out on the net. Setting up /etc/hosts.allow and /etc/hosts.deny to only # allow local access is another alternative but not the safest. # Firewalling port 177 is the safest if you wish to have xdmcp on. # Read the manual for more notes on the security of XDMCP. Enable=true # Honour indirect queries, we run a chooser for these, and then redirect # the user to the chosen host. Otherwise we just log the user in locally. #HonorIndirect=true # Maximum pending requests #MaxPending=4 #MaxPendingIndirect=4 # Maximum open XDMCP sessions at any point in time *#MaxSessions=16* # Maximum wait times #MaxWait=15 #MaxWaitIndirect=15 # How many times can a person log in from a single host. Usually better to # keep low to fend off DoS attacks by running many logins from a single # host. This is now set at 2 since if the server crashes then qdm doesn't # know for some time and wouldn't allow another session. #DisplaysPerHost=2 # The number of seconds after which a non-responsive session is logged off. # Better keep this low. #PingIntervalSeconds=15 # The port. 177 is the standard port so better keep it that way Port=177 # Willing script, none is shipped and by default we'll send # hostname system id. But if you supply something here, the # output of this script will be sent as status of this host so that # the chooser can display it. You could for example send load, # or mail details for some user, or some such. #Willing=/etc/X11/gdm/Xwilling

Y nos vamos a la sección de [xdmcp] y allí cambiaremos enable=false por enable=true. Tendremos que tener en cuenta que xdmcp escucha en el puerto 177 para configurar iptables correctamente.

Para reiniciar las X tenemos que ejecutar: [root@adminhouse]# service portmap restart

3.4. Servidor de Fuentes

El servidor de fuente como su nombre indica es que nos va a servir las fuentes que nos hagan falta dando un aspecto mucho mas conseguido a nuestra sesión X.

Para activarlo editamos el archivo:

```
[root@adminhouse]# vi /etc/X11/fs/config
```

```
#
# Default font server configuration file for Red Hat Linux
#
# allow a max of 10 clients to connect to this font server
client-limit = 10
# when a font server reaches its limit, start up a new one
clone-self = on
# alternate font servers for clients to use
#alternate-servers = foo:7101,bar:7102
# where to look for fonts
catalogue = /usr/X11R6/lib/X11/fonts/misc:unscaled,
      /usr/X11R6/lib/X11/fonts/75dpi:unscaled,
      /usr/X11R6/lib/X11/fonts/100dpi:unscaled,
      /usr/X11R6/lib/X11/fonts/Type1,
      /usr/share/fonts/default/Type1,
# in 12 points, decipoints
default-point-size = 120
# 100 x 100 and 75 x 75
default-resolutions = 75,75,100,100
# use lazy loading on 16 bit (usually Asian) fonts
deferglyphs = 16
# how to log errors
use-syslog = on
# don't listen to TCP ports by default for security reasons
<mark>#no-listen = tcp</mark>
```

Solo tenemos que comentar la última línea donde dice que no escuche conexiones tcp.

[root@adminhouse]# service xfs start //Iniciamos el servicio

3.5. Pxes-base y Pxeconfig

Para instalar el programa con el que vamos a crear la imagen lo primero que vamos ha hacer es instalar el repositorio de pxes para poder instalarlo con yum.

[root@adminhouse root]# wget -c http://downloads.2x.com/pxes/yum/fedora/4/i386/RPMS.2X-extra/2X-release-4-1.noarch.rpm //lo descargamos

[root@adminhouse root]# rpm -ivh 2X-release-4-1.noarch.rpm //lo instalamos

A continuación pasamos a instalar el programa en sí:

[root@adminhouse]# yum install pxes-base [root@adminhouse]# yum install pxesconfig // front-end para pxes

Una vez que tenemos el programa lo ejecutamos:



Nos aparecerá una ventana de bienvenida y pulsaremos siguiente.

Kernel	and initial ram	disk selection	2)				
Select th If you wa If the ram check "F	e desired splash screer ant to initialize the ram d n disk image already exi Read saved configuratio	n image, kernel and initial ram disk. isk image check "Initialize ram disk", the image must exist. st and you want to re-read the configuration n".					
Splash s	creen:	/tftpboot/pxes/2X.lss	← Buscar				
Initial ra	um disk						
en.	Ram disk name:	/tftpboot/pxes/pxes-1.2.initrd	← Buscar				
6.3	🛃 Initialize ram disk co	ontents 🔲 Read saved configuration					
Kernel	configuration						
	Kernel architecture:	🔿 i386 🔿 i486 💿 i586 🔿 i686 🔿 via c3					
annan ann an	Kernel file name:	/opt/pxes-1.2/stock/arch/i586/kernel/vmlinuz-2.4.32-2pxes.i586	✓ Buscar				
Network	bootable image						
Sh	Enable network bo	otable image generation (needed by Etherboot)					
0		/tftpboot/pxes/pxes-1.2.nbl	→ Buscar				
ISO boo	otable image						
	Enable ISO 9660 b	ootable image generation					
9		/tmp/pxes-1.2.iso	← Buscar				
Squash	Squashed image						
	Enable Squashfs image generation						
	Squashed image name	: /tftpboot/pxes/pxes-1.2.squash	✓ Buscar				
			M Canada				
		i zi ontorior. ETE IN Siguionto II	i w i ancolar l				

En esta ventana ya empezamos a crear la imagen, de forma que marcamos la opción de inicializar la ram, por defecto trae una configuración de kernel para arquitecturas i586.

Required Local Device	89	2
Select the desired characteristics Keyboard configuration Layout: es	of the local devices	~
Mouse configuration Device: /dev/	ssaud Protocol: PS/2	~
Acceleration: Sensitivity:	slow (2)	fast high
Network configuration Network card: Kernel module options	Autodetect	>
		V Cancolar

En la siguiente pantalla seleccionaremos nuestro ratón, el idioma del teclado y aconsejo dejar la tarjeta de red en "autodetec".



En esta ventana vemos las múltiples opciones que nos da pxes de escritorios remotos, nosotros seleccionaremos la opción de Unix/Linux XDM que es lo que hemos estado preparando en nuestro servidor.

	3.3.6		XFree86 4.3.0	
Font Server				
A	🛃 Enable font s	erver		
60	Server: 192.168	.0.10	Port: 7100	-
Video Hardv	ware			
	Video driver:	Autodetect		
	L			
Video Mode	es			
1600×12	200 🗌 1280×10	024 🗌 1024×76	800×600	🗌 640×480
Custom	×	Modeline:		
Color Depth				○ 0 bits
Color Depth	32 bits	② 24 bits	16 bits	
Color Depth	O 32 bits	24 bits	O 16 bits	
Color Depth	O 32 bits quency	24 bits	O 16 bits	
Color Depth	O 32 bits quency ☑ Autodetect	 24 bits Horizontal: 31.5-57 	○ 16 bits KHz Vertical: 50-70	Hz Monitors DE
Color Depth Monitor Free Ontions	○ 32 bits quency ? Autodetect	24 bits Horizontal: 31.5-57	16 bits KHz Vertical: 50-70	Hz. Monitors DE

Ahora empezamos a configurar nuestro servidor X, seleccionaremos la versión más moderna, habilitaremos el servidor de fuentes e indicaremos la ip del nuestro. Y luego las características del escritorio.

XDM c	onfiguration			2)(
	XDM lets you acc such as GDM, KD	ess to Unix/Linux servers running M or XDM	ı display manager	
Connecti	on Method			
Select th	e connection metho	od used to locate your XDM serve	er.	
If you sp You can	ecify "Direct" or "In nostnone the value	direct" methods a server should a as until runtime	llso be provided.	
Aski	he values after boo	ting the thin client		
O Direction	t	Indirect	O Broadcast	
Server	192 168 0 10			
Dort	177			
Porc	177.			v

Ahora le indicaremos la ip del servidor de X.

Por último, seleccionaremos otras opciones de nuestro cliente y crearemos la imagen.

General configurat	ion			2
Enable debug				
Wait for key press before	starting X Window	'S		
Wait for key press before	connecting to logi	n server		
□ Wait for key press on erro	r connecting to lo	gin server		
-Console and Remote Access	s		tiantian)	
Enable login shell (acces	sible through CT	NEALTER with no authem	ucauony	
Enable login shell (acces	isible through CTF	RL-ALI-F3 authenticated)		
Enable teinet server		Enable SSH server		
Set root password				
Enable tiny web server (no authentication)	Enable session VNC	shadowing (no authentication)	
Boot Messages You may also need 'console Level: less	e=null' in kernel c ////	ommand line to be complet more	tely silent	
Remote Configuration The remote configuration se The server should be acces	rver lets you char ible via TFTP.	nge aspects of the thin clie	nt in a per client basis.	
Enable remote configura	tion			
Server:				
Directory:	/pxes/config			
1		∠ Anterior		Cancelar

3.6. Comprobación de servicios

Una vez que hemos configurado todos nuestros servicios tenemos que comprobar que efectivamente están a la escucha de conexiones, para ello haremos un:

[root@admi	nhous	e]# lsof -i					
COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE	NODE NAME
portmap	1616	rpc	3u	IPv4	4363	UDP	*:sunrpc
portmap	1616	rpc	4u	IPv4	4367	TCP	*:sunrpc (LISTEN)
rpc.statd	1634	rpcuser	4u	IPv4	4399	UDP	*:32768
rpc.statd	1634	rpcuser	5u	IPv4	4387	UDP	*:962
rpc.statd	1634	rpcuser	бu	IPv4	4413	TCP	*:37069 (LISTEN)
mDNSRespo	1827	nobody	8u	IPv4	4729	UDP	*:5353
sendmail	1861	root	4u	IPv4	4833	TCP	*:smtp (LISTEN)
<mark>xfs 🛛 👘 🔹 👘 🔹 👘 👘</mark>	1924	xfs	3u	IРvб	4982	TCP	*:xfs (LISTEN)
xfs	1924	xfs	4u	IPv4	4983	TCP	*:xfs (LISTEN)
<mark>gdm-binar</mark>	2252	root	3u	IPv4	6337	UDP	*:xdmcp
dhcpd 💦	2899	root	би	IPv4	17747	UDP	*:bootps
xinetd 💦	4602	root	<mark>5u</mark>	IPv4	35475	UDP	*:tftp

Y como podemos ver, entre otros servicios que podemos tener instalados, está el servidor de fuentes (xfs), el servidor de X (xdmcp), el servidor dhcp (bootps) y el servidor de tftp.

Si falta algún servicio por iniciar recomiendo reiniciar el equipo y si sigue fallando habría que volver a mirar los archivos de configuración.

El comando chkconfig --list es un comando muy útil para comprobar los niveles de ejecución de cada demonio o servicio y saber si deben estar iniciados o no. En caso de que no lo estén podemos indicar con este comando los niveles en los que queremos que estén activos o no: chkconfig --level 345 demonio on/off

4. Configuración clientes

Tenemos varias opciones a la hora de arrancar nuestros equipos clientes, según lo antiguos que sean:

4.1. Arranque por red soportado por la placa (Onboard Lan Boot)

Para poder arrancar de esta forma tenemos que tener un equipos con una placa relativamente moderna que nos lo soportes, por lo que para los equipos antiguos esta no es una opción válida.

Para configurar la bios tenemos que poner a la Lan como primer dispositivo en nuestro menú de arranque y además, algunas placas requieren que se le active una opción llamada Onboard Lan boot, con lo que en el arranque se activara un agente de arranque por red que trae la propia placa integrado.

4.2. Arranque por red a través de disquete.

Como su nombre indica solo tenemos que crear un disquete el cual viene preparado para enviar las peticiones al bootp y los demás pasos.

Podemos descargarnos un disquete de arranque personalizado según nuestra tarjeta de red en <u>http://rom-o-matic.net</u>.

Una vez que tengamos la imagen podemos copiarla al disquete con: [root@adminhouse root]# cat imagen.iso > /dev/fd0

Y configuraremos la bios de nuestro equipo para que arranque desde disquete.

4.3. Arranque por red con tarjeta de red con eprom

En la página anterior <u>http://rom-o-matic.net</u> tenemos la opción de crear una imagen para meterla en una memoria eprom de una tarjeta de red para que los equipos cojan la información de la propia tarjeta. Esta opción nunca la he probado por lo que no puedo comentar mucho más.

NOTA: para hacer todas estas pruebas de los clientes recomiendo que iptables esté bajado para evitar posibles fallos y una vez que todo funciona añadir las reglas pertinentes a iptables.

Entra en <u>http://www.adminhouse.com.es/</u> para descargar la última versión del manual y en <u>http://www.adminhouse.com.es/foro</u> para solucionar tus dudas o envía un correo a ayuda@adminhouse.com.es.