Discurso de Temas Linux

Primer Taller CEDIA

1 de Marzo, 2004

Presentado por Hervey Allen
Network Startup Resource Center

Bienvenidos

- Introduccion
  - Instructores:
    Hervey Allen
    Brian Candler
    Carlos Vicente
  - Ayudantes:
    Albert Espinal
    Federico Domínguez
    Dolores Lizarzaburu
    Neil Nuñez
- ¿Que hemos hecho?
- Como corre el clase
- Que harémos hoy?

Horario

Mañana

- 08:45-10:45 Clase
- 10:45-11:00 Pausa
- 11:00-13:00 Clase

Almuerzo

- 13:00-14:00

Tarde

- 14:00-16:00 Clase
- 16:00-16:15 Pausa/Once
- 16:15-18:15 Clase

Sesion de Noche

Son opcionales. Tipo 19:30-21:00.

Laboratorio

Abierto despues que la cena (19:30) hasta tipo 21:00. Si hay clase puede usar las maquinas no en uso por ellos atendiendo la sesion.

Como se corre el taller

¡Con tu participacíon!

Por favor, haz preguntas. Si no entiende el Español de Hervey, dile.

Tendremos que compartir algunas PCs. Esto esta bien. Uno aprende mucho en grupos.

Al fin de la semana tendrémos una prueba escrita y practica sobre las temas que vamos a aprender.

Habrá un certificado de complecion para todos en Sabado cuando terminamos con el taller.

Como corre el taller – practíca

Tendremos un servidor por el taller. Su IP es 192.188.58.126.

Los PCs tienen acceso al Internet, pero durante el clase trabajamos mucho sin XWindows.

La contraseña de root es “espe2004”

Tenemos varios asistentes durante el Taller para hacer traducciones como necesario y para ayudar con los ejercicios. Gracas a Dolores, ESPE, ESPOL, FUNDACYT y CEDIA por todo esto.

Compendio*

- ¿Que distribucion de Linux deberiamos usar?
- Discurso de particiones en Linux y opciones - /etc/fstab, /dev.
- Permisos en Linux. Los comandos chmod y chown.
- Ifconfig, configurar dispositivos y interfazes de ethernet.
- Discurso de servicios de Linux y como saber que esta corriendo.
- El directorio especial de /proc
- Cambios de configuracion a /etc/sysconfig/network-scripts.
- Presentar /etc/crontab y practica del uso de comandos del servico cron.
- Linux kernel. Como recompilarlo.
- /etc/modules.conf
- Mencionar los 'firewalls'.
- Instalacion de software compilado.
- Gnome vs. KDE y XWindows. Que son. ¿Porque no son necesario por un servidor?
- Logs y donde se quedan. Inspecion de los Logs. Anota /etc/syslog.conf.

* Muestra de como *no hacer* un slide en una presentacion...

¿Que distribución?

Red Hat, Fedora, SuSE, Debian, Conectiva, Turbolinux, Mandrake, United Linux, Gentoo, Slackware, etc?

O, usamos FreeBSD, Solaris, OpenBSD, NetBSD, HP/UX, AIX, Mac OS X, SCO, etc?

¿Que corren Uds?

Nuestra respuesta:

- Red Hat 9 parece *casi* como estandar
- Si sabes Red Hat Linux (¡comandos en texto!), ya sabes bastante para usar lo demas.
- Ayuda local.
- Queremos enseñar como escalar servicios y las “buenas practicas” por redes en un sistema razonable.

Sistema de Archivos Linux

- ext2, ext3, reiserfs, jfs, hpfs, etc...
  - El “ganador” hoy en dia, ext3 (journaling)
- Estructura de particiones:
  - / (“root”)
  - /usr
  - /var
  - /tmp
  - /home
  - swap

/ (“root”)

El particion root es donde viven los archivos criticos para inicializar el sistema al modo de “single user” o nivel de initializacion 1 (initlevel 1).

La idea es que este sistema no cresca ni cambia y que se queda aislado al resto del sistema.

Red Hat instala, por defecto, asi:

- /boot (aprox. 100Mb)
- swap (1 a 2 x RAM instalado)
- / (todo el resto del disco duro)

/usr

Se lo usa para software del sistema como herramientas de usuarios, compiladores, X Windows, etc.

Si uno tiene que expander el espacio por este software, entonces tenerlo separado lo permite esta operacion.

*Hablaremos de esto. No siempre instalamos Linux con una particion de /usr separada.

/var

Aqui se guarda archivos que cambian muchos. Por ejemplo; los logs de servidor de Web, directorios de correo, imprenta, etc.

En un servidor es una idea buena tener /var en una particion aparte. Si alguien, por accidente o a proposito, crea un monton de correo o actividad de su servidor de Web, entonces eso se puede llenar la particion. Si /var no esta aparte otras particiones esto se puede parar su servidor.

/tmp

Donde los usuarios y aplicaciones pueden guardar archivos en forma temporaria. Normalmente Linux no se usa cuotas, asi es posible por un usuario llenar una particion con /tmp por accidente o a proposito.

Otro lugar por /tmp puede ser /var/usr/tmp, etc., pero muchas aplicaciones esperen ver /tmp.

Otro convenio en el mundo de Unix es usar “/scratch” en vez de /tmp.

/home

Aqui viven los directorios de los usuarios en su servidor (correo, tal vez, no esta guardado en esta particion).

Sin usar cuotas (comun en Linux) los usuarios pueden llenar una particion /home facilmente.

Ejemplo: Un usuario escribe un programa que produce texto en un archivo. Por accidente el usuario crea un loop y no se da cuenta. Se puede crear un archivo que llena la particion rapidamente.

Swap

Swap es donde existe su memoria virtual. Es un sistema de archivo proprio.

Sin swap su PC corre mas rapido, pero esto es peligroso.

Hay varias ideas sobre cuanto swap tener, y depende en que servicios esta corriendo. La regla es que swap deberia ser equivalente a su RAM hasta dos veces de su RAM (memoria).

Montando los sistemas de archivos

- Si quieres montar algo que no se describe en /etc/fstab tiene que usar el comando mount.
- Primero tiene que saber que entrada en /dev describe el dispositivo que quieres montar (un cd, floppy, otro disco duro, etc.)
- Tambien, tienes que saber que sistema de archivo.
- Por ejemplo un floppy:
  - mount -t msdos /dev/fd0 /mnt/floppy

Permisos de archivos

- Hay cinco categorias y tres tipos de permisos con que tienes que preocuparse.
- El permiso por defecto esta hecho con el comando `umask.`
- Dos categorias de permiso se trata con el usuario o el grupo que va a correr un archivo (setuid, setgid).
- Los permisos son “r” (leer), “w” (escribir), y “x” (ejecutar).
- Se puede asignar los permisos al mundo (a), grupo (g), y usuario (u).

Permisos de archivos continuado

- Un archivo se pertenece a un usuario. Se puede asignar el archivo a otro usuario y al otro grupo usando el comando `chown` (“CHange OWNer”).
- Se puede cambiar permisos y duenos de un grupo de archivos o a todo los archivos y los archivos en subdirectorios usando `chmod` y `chown`.
- Finalmente se puede cambiar los permisos a los directorios usando `chmod`.

Ifconfig

Durante la semana vas a usar este comando *mucho*.

Ifconfig se lo usa para ver el estado de los interfazes ethernet y para configurlos.

Por ejemplo:

- `/sbin/ifconfig`
- `/sbin/ifconfig eth0 192.188.58.66 netmask 255.255.255.224`

/etc/hosts

Este archivo requiere tener, por lo minimo, este linea:

```
127.0.0.1localhost.localdomain    localhost
```

En un red privado se puede usar este archivo en vez de tener un servidor de dns. El servidor busca en /etc/hosts antes de preguntar al dns para resolver un nombre a una direccion IP.

Se cambia este orden en /etc/nsswitch.conf

Agregamos una entrada en /etc/hosts por “noc”

Servicios corriendo y en que puertos

- Todo los servicios:
  - `ps -aux | more`
- Que puertos estan usando?
  - `sudo /usr/sbin/lsof -i`
  - `/bin/netstat -natu`
- Que corre al inicializacion? En /etc/rc.d/init.d
  - `/sbin/chkconfig –list | more`
  - `No olvida xinet.d y inet.d`

El directorio especial de /proc

- El directorio /proc se representa el estado de tu maquina. Es un abstracto de tu kernel y no existe como sistema de archivos en disco.
- Por ejemplo /proc/cpuinfo tiene informacion sobre el cpu (o cpu's) en tu maquina. Se puede leerlo asi:
  - `less /proc/cpuinfo`

Pero, tiene que ser root.

El directorio /proc continuado

Algo de los archivos en /proc:

/proc/meminfo	El estatus del uso de la memoria
/proc/version	Numero de version del kernel.
/proc/net/dev	Informacion sobre cada dispositivo de redes
/proc/interrupts	Uso de los IRQs en tu sistema
/proc/kcore	Los contenidos en RAM. (ojo)

Mas /proc

Entonces si piensas en /proc puedes realizar que hay varios comandos que solamente muestra que haya en /proc.

Por ejemplo, el comando `lsmod` es basicamente lo mismo que:

```
cat /proc/modules
```

Y, el comando ps recibe los datos que requiere mirando en /proc/processid

Cambios de configuracion

Si quieres que tu configuracion de interfaz de ethernet queda para reinicializacion tiene que cambiar un archivo /etc/sysconfig/network-scripts/ifcg-eth0 (si fuera por eth0).

Si el “default route” no se puede calcular usando la direccion de IP y el netmask, entonces tienes que usar el comando:

- route add default gw nnn.nnn.nnn.nnn

para salir afuera tu red.

Crontab

- El servicio “cron” te permite correr programas automaticamente cuando quieres.
- Esta configurado por /etc/cron.allow, /etc/cron.deny, si existen.
- Usa el comando `crontab` para cambiar los archivos que controlan como funciona el daemon crond.
- Tiene un formato muy especifico.

Crontab continuado

El archivo que muestra un trabajo de cron tiene un formato asi:

```
Minuto Hora Dia Mes DiaDeSemana Comando
```

Un ejemplo:

```
1 4 1 4 * /bin/mail usuario@punto.com < /home/usuario/joke
```

Mande un correo el primero de abril.

Existe “anacron” que usa /etc/anacrontab por maquinas que no siempre estan corriendo.

El Linux Kernel

El kernel, o corazon de Linux se puede ajustar como quieres. Tienes que instalar el fuente (“source”) por el kernel. Se reside en /usr/src

Si quieres recompilar el kernel se puede hacer esto usando “`make menuconfig`” o “`make xconfig`” en /usr/src/linux-2.4.

Ojo – si quieres hacer solo un cambio, entonces, usa el archivo de config y hecho que viene con la distribucion que corres.

El Linux kernel continuado

Por ejemplo, en Red Hat haz esto (solo ejemplo):

```
cd /usr/src/linux-2.4
cp configs/kernel-2.4.20-i386.config .config
```

Y, despues corre “make menuconfig” - Hay muchas elecciones. Para hacer el proceso bien vaya a:

http://www.kernel.org/

Y baja la documentacion sobre como compilar el kernel.

Modulos que el kernel se carga

- /etc/modules.conf describe los “drivers” que se carga para apoyar el hardware en tu PC.
- Se puede ver que modulos estan en uso, cargar un modulo manualmente, y remover un modulo:
  - `/sbin/lsmod`
  - `/sbin/insmod`
  - `/sbin/rmmod`

Firewalls

Aqui mencionamos esta tema:

- Con una distribucion “moderna” de Linux se puede esperar que el kernel ya tiene apoyo para “netfilter”
- Se puede controlar el ingreso y salida de paquetes (tcp/udp/icmp) usando los “iptables”
- En Red Hat se inicializa iptables como un servicio en /etc/rc.d/init.d/ que lea el script /etc/sysconfig/iptables
- Para hacer un grupo de reglas que funcionan bien como un firewall requiere que entiende los protocolos y configuracion de redes muy bien.

Instalacion de software compilado

Es posible que va a querer de instalar software que no esta disponible en un paquete de RPM, o que tienes que cambiar o reconfigurar antes de instalar.

En estes casos, tienes que compilarlo desde el fuente original.

Es muy tipico que el software viene en un archivo de “tar” que esta comprimido.

Un ejemplo de como funciona instalacion asi -->

Instalacion de software cont.

- Baja el archivo fn.tar.gz a /usr/local/src.
- tar xvzf /usr/local/src/fn.tar.gz
- cd /usr/local/src/fn-version
- ./configure
- make
- make install

Esto es todo si funciona, pero ahora no tienes ningun record de como desinstalar el software...

XWindows – Gnome – KDE

La primera cosa entender es que Gnome y KDE usan el sistema de XWindows como su base. Por esto, los programas de KDE corren en Gnome y vice-versa.

Por un servidor no es necesario correr ninguno de esto.

Si vas a correr uno se puede correr ambos, y mas sistemas graficas de interfaz.

XWindows – Gnome – KDE cont.

- Cual sistema es mejor? No hay respuesta.
- Para configurar el base de estes sistemas vea el archivo /etc/X11/XF86Config.
- Se puede configurar todo usando menus, pero si entiendes /etc/X11/XF86Config es mejor.
- Para salir el interfaz grafico tiene que cambiar al nivel de inicializacion 3 (o 2 o 1).
- Tambien, se puede ir a un shell usando alt-ctrl-f2 a f6.

Logs – como saber que pasa

- Para configurar que servicios va a reportar eventos vea el archivo /etc/syslog.conf.
- Ahora mira al archivo /var/log/messages. El comando "`tail`" es muy util por esto.
- Acustambrarte al usar /var/log/messages para resolver problemas. Por ejemplo corriendo un servicio
- Tipe el comando "`tail -f /var/log/messages`" mientras que inicializes y apagues un servicio.

Logs continuado

Hay mucho mas archivos de logs. Por ejemplo, si corres un servidor de web, como apache, todo los logs estan en /var/log/httpd

sendmail usa /var/log/maillog.

Hay software para leer los logs y reportar sobre eventos en una forma mas automatico. Vea al:

- http://nsrc.org/security/index.html#logging

Para ver algunos paquetes de software.

Resumen

El sistema Linux esta construido en una manera modular para permitir que sea estable, extensible, y seguro.

Tener acceso al codigo que hace el sistema permite desarollo bien rapido y reparacion de problemas en corto plazo.

Con el kernel 2.4 y luego 2.6 Linux esta acercando el nivel de utilidad para apoyar servicios en escala grande tanto como FreeBSD o Solaris.

Mas recursos

- http://www.google.com/linux
- http://www.linux.org/
- http://www.linuxdocs.org/
- Los libros de O'Reilly (http://www.oreilly.com/)
- http://www.sourceforge.net/
- http://www.redhat.com/

Hervey Allen - hervey@nsrc.org