Procedimiento para instalar la sala de sistemas del SSF+UN

William Fdo. Oquendo Patiño*

October 9, 2012

Contents

*woquendo@gmail.com

1 Generalidades

- LA ÚNICA MANERA DE ENTENDER LO QUE SE MUESTRA ACÁ Y DE MEJORARLO ES LEYENDO LA DOCUMENTACIÓN. POR FAVOR LEA LOS MANUALES, LOS HOWTO Y APÓYESE EN GOOGLE.
- **Since Ubuntu does not have ssh server by default, which is a huge problem, I am creating a live cd with ssh, nfs, and nis by default to install it. The live dvd is created with thebackup option of remastersys.**
- La distribución a instalar es Ubuntu 12.04.1 LTS.
- Sistemas de 32 bits. Sistemas de 64 bits quedarán con OS de 32. Es posible mezclar 64 y 32, pero eso no se tratará en este documento. Para resolver el problema de la memoria RAM, se instalara un kernel con high memory support en los equipos grandes.
- Los servicios a configurar son: red interna con DNS MASQ (DHCP + DNS + NTP + MASQUERADING) , NFS, NIS, FIREWALL,, BACKUP.
- Se supone que el servidor tiene dos tarjetas de red: una para la red interna y otra para la externa. La tarjeta de red que conecta con la red exterior se denotará por ETH0, la tarjeta de red de la red interna se denotará por ETH1. Tambien se supone la existencia de unn hub al que se conecta la red interna. La externa se conecta al punto de red.
- El segmento de red utilizado en este ejemplo será 192.168.123.xxx , y se supone que el dominio escogido será ssf.net
- Al final del documento se muestran algunos ejemplos de los archivos de configuración.

2 Configuración de la red por cada computador

See `/usr/share/doc/ifupdown/examples/`

2.1 SERVER

Se debe configurar a ETH1 con ip fijo igual a 192.168.123.1 , y se debe configurar a ETH0 para que haga DHCP, ojalá con ip fijo solicitado anteriormente, en este caso 168.176.14.174.

NOTE: What is the problem of using networking or interfaces file? DEACTIVATE THE NETWORK MANAGER BY EDITING `/etc/NetworkManager/NetworkManager.conf` and comment the dnsmasq stuff and change managed to true.

For eth0 with dhcp, add the following to `/etc/network/interfaces` :

```
auto eth0
iface eth0 inet dhcp
dns-nameservers 192.168.123.1 8.8.8.8 8.8.4.4
```

For eth1 with fixed ip address, add the following to `/etc/network/interfaces` :

```
auto eth1
iface eth1 inet static
address 192.168.123.1
netmask 255.255.255.0
gateway 192.168.123.1
```

Restart network : `sudo service networking restart`

Now you can use the commands `ifup eth0` or `ifdown eht0` (or `eth1`) to set up or down each interface.

What about hooks after updating interface?

2.2 CLIENT

(For each client)
For eth0 with dhcp, add the following to `/etc/network/interfaces` :

```
auto eth0
iface eth0 inet dhcp
```

(Check if timeout should be configured, in case of slow dhcp response)

2.3 Test

Aún no se puede hacer testing dado que no se ha configurado completamente el servidor DHCP.

3 Mirror configuration for apt-get

3.1 Server

Since the unal proxy requires user and password info, and I do not want to store it on each computer, I will use the local mirror at unal. (If it is not reliable I will configure my own mirror in the future).

- Backup the old config file
  `sudo cp /etc/apt/sources.list /etc/apt/sources.orig`.
- Create a new config file with the following content:
  `# unal`
  `deb http://168.176.34.158/ubuntu/ precise main multiverse restricted universe`
- Now run the update of the database:
  `sudo apt-get update`

3.2 Clients

The config file will be copied-synced later and all the procedure will be repeated.

4 SSH

4.1 SERVER

- `apt-get install openssh-client openssh-server`
- Maybe, add the line `sshd : ALL` to `/etc/hosts.allow`
- Since we are installing from a live dvd, there could be problems like `no hostkey alg`. The fix is (`http://www.cyberciti.biz/faq/howto-regenerate-openssh-host-keys/`) :
  - Dleete or rename the old system wide keys:
    `mv /etc/ssh/ssh_host_* ./`
  - Reconfigure the server ssh:
    `dpkg-reconfigure openssh-server`
  - Maybe, update the known hosts
    `~/.ssh/known_hosts`
- `service ssh restart`

4.2 CLIENT

- `apt-get install openssh-client openssh-server`
- `service ssh restart`

5 Firewall (NAT)

Se usará al firewall de arno (Arno iptables, http://rocky.eld.leidenuniv.nl/). El firewall se instala en el server conectado a la red externa, se supone que la red interna es confiable.

Furthermore, the software fail2ban is adviced on all pcs to control attacks.

5.1 SERVER

Just install it by using the command
`sudo apt-get install arno-iptables-firewall` You can either config the firewall by using the initial config option or the command `dpkg-reconfigure arno-iptables-firewall`, or from scratch or use an old config file.

La configuración queda en el archivo `/etc/arno-iptables-firewall/firewall.conf` . Revisarla, editar las rutas de los comandos `iptables` y `ip6tables`.

If you are using the old config file from slackware, check the environment and plugins settings (remove local).

5.2 CLIENT

No se instala el firewall en los clientes, sólo en el servidor de red externo. Se supone que la red interna es confiable.

5.3 TEST

Start the firewall with `sudo service arnot-iptables-firewall start`

Clients cannot browse yet because dns is not configured

6 Dnsmasq (DHCP, MASQUERADING, NTP, DNS, etc)

La configuración de todos estos servicios se realiza facilmente con el paquete dnsmasq. Una vez instalado, el archivo /etc/dnsmasq.conf debe ser configurado para el caso particular. Se remite al lector a toda la documentación que acompaña a el paquete. El ejemplo del archivo dnsmasq.conf se encuentra al final de este documento. El servicio dnsmasq debe ser activado desde el inicio

Las opciones importantes a configurar son:

- Interface en la que se escuchan los request de dhcp (ETH1 y ETH0):

```
# If you want dnsmasq to listen for DHCP and DNS requests only on
# specified interfaces (and the loopback) give the name of the
# interface (eg eth0) here.
# Repeat the line for more than one interface.
interface=eth1
interface=eth0

listen-address=127.0.0.1
```

- Dominio:

```
domain=ssf.net
```

- Rango de dhcp (hay muchas formas de hacerlo, ver la documentación):

```
dhcp-range=192.168.123.2,192.168.123.250,255.255.255.0,12h
```

- Ignorar request de otras máquinas: Esta medida requiere conocer las macs de los clientes y SE ACONSEJA.

```
dhcp-host=*:*:*:*:*:*,ignore
```

- Asignar ips de acuerdo a la mac, tener en cuenta que los nosmbres de DNS serán los asignados por el DHCP:

```
dhcp-host=00:11:11:82:EB:26,ssf32-02,192.168.123.2
dhcp-host=00:12:3F:A7:28:C3,ssf32-03,192.168.123.3
dhcp-host=00:07:E9:F0:C4:C9,ssf32-04,192.168.123.4
dhcp-host=00:14:22:3B:24:F3,ssf64-01,192.168.123.5
```

  Y así sucesivamente.

- Opciones de acuerdo al RFC 2132 (google ?) en este caso la mascara de red, el gateway, winsservers y dns server.

```
dhcp-option=1,255.255.255.0
dhcp-option=6,192.168.123.1
dhcp-option=44,168.176.160.22,168.176.160.23
dhcp-option=41,192.168.123.1
```

- Ntp server:

```
# Set the NTP time server addresses to 192.168.0.4 and 10.10.0.5
#dhcp-option=option:ntp-server,192.168.0.4,10.10.0.5
dhcp-option=option:ntp-server,192.168.123.1
# Set the NTP time server address to be the same machine as
# is running dnsmasq
dhcp-option=42,0.0.0.0
```

- NIS domain (for user authentification)

  ```
  dhcp-option=40,ssfservernis
  ```

  NOTE: IS THIS NEEDED??? Por último, es importante comentar la línea de 127.0.0.1 en el `/etc/hosts` y reemplazarla por el ip del servidor, 192.168.123.1 . El archivo debe quedar así:

  ```
  #
  # hosts         This file describes a number of hostname-to-address
  #               mappings for the TCP/IP subsystem.  It is mostly
  #               used at boot time, when no name servers are running.
  #               On small systems, this file can be used instead of a
  #               "named" name server.  Just add the names, addresses
  #               and any aliases to this file...
  #
  # By the way, Arnt Gulbrandsen <agulbra@nvg.unit.no> says that 127.0.0.1
  # should NEVER be named with the name of the machine.  It causes problems
  # for some (stupid) programs, irc and reputedly talk. :^)
  #

  # For loopbacking.
  127.0.0.1               localhost
  #127.0.0.1              ssf1.ssf.net ssf1
  192.168.123.1           ssf1.ssf.net ssf1

  # End of hosts.
  ```

6.1 TEST

Reiniciar el (servicio en el) servidor, luego reiniciar los (servicios en los) clientes. Desde un cliente hacer ping al server

```
ping 192.168.123.1
```

y debe ser exitoso, mostrando algo como

```
PING 192.168.123.1 (192.168.123.1) 56(84) bytes of data.
64 bytes from 192.168.123.1: icmp_seq=1 ttl=64 time=0.158 ms
64 bytes from 192.168.123.1: icmp_seq=2 ttl=64 time=0.147 ms
64 bytes from 192.168.123.1: icmp_seq=3 ttl=64 time=0.142 ms
64 bytes from 192.168.123.1: icmp_seq=4 ttl=64 time=0.145 ms

--- 192.168.123.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2999ms
rtt min/avg/max/mdev = 0.142/0.148/0.158/0.006 ms
```

De lo contrario hay que encontrar el problema. Un error común es que dnsmasq no esté corriendo.

7 VPN : Computers outside the internal LAN but inside University LAN

In this section we are going to configure show how to setup a VPN, using openVPN, to connect computers outside the internal LAN to the main server in order to share NFS files and to allow NIS authentification. The network segment will be 10.88.55.0/24 .

Heavily based on the openVPN howto: `http://openvpn.net/howto.html`

7.1 SERVER

https://help.ubuntu.com/12.04/serverguide/openvpn.html

- sudo apt-get install openvpn
- mkdir /etc/openvpn/easy-rsa/
- cp -r /usr/share/doc/openvpn/examples/easy-rsa/2.0/* /etc/openvpn/easy-rsa/
- OPTIONAL : Edit /etc/openvpn/easy-rsa/vars :

  ```
  export KEY_COUNTRY="US"
  export KEY_PROVINCE="NC"
  export KEY_CITY="Winston-Salem"
  export KEY_ORG="Example Company"
  export KEY_EMAIL="steve@example.com"
  ```

- cp openssl-1.0.0.cnf openssl.cnf
- Generate the master Certificate Authority (CA) certificate and key:

  ```
  cd /etc/openvpn/easy-rsa/
  source vars
  ./clean-all
  ```

- Build the certificate: ./build-ca , you can use default answers for everything but Common Name , where you should put something to describe your vpn network, like OpenVPN-SSF.
- Generate certificate and key for server: ./build-key-server ssfvpnserver , you can use default parameters, BUT Common Name should be set to ssfvpnserver. Answer y for Sign the certificate and y to commit.
- Generate Diffie Hellman parameters: ./build-dh
- cd keys/
- cp ssfvpnserver.crt ssfvpnserver.key ca.crt dh1024.pem /etc/openvpn/
- Generate certificates and keys for the clients:
  cd /etc/openvpn/easy-rsa/
  source vars
  ./build-key client1
  ./build-key client2
  ./build-key client3
  MAKE SURE TO PUT THE NAME OF THE CLIENT IN Common Name : First one is client1, second one is client2, etc.
- copy to each client:

  ```
  /etc/openvpn/ca.crt
  /etc/openvpn/easy-rsa/keys/client1.crt
  /etc/openvpn/easy-rsa/keys/client1.key
  ```

- sudo cp /usr/share/doc/openvpn/examples/sample-config-files/server.conf.gz /etc/openvpn/
- sudo gzip -d /etc/openvpn/server.conf.gz
- Edit /etc/openvpn/server.conf

```
ca ca.crt
cert ssfvpnserver.crt
key ssfvpnserver.key
dh dh1024.pem
```

You can also set the following options to `/etc/openvpn/server.conf`:

```
client-to-client  # allows communication between clients
push "dhcp-option DNS 168.176.14.174"
```

- Change the config for the network if you want to change the default. (For ssf 10.88.55.0/24)

You must open the port 1194 in the firewall `OPEN_UDP="53,67,1194"`, and add the interface tun0 to the fully trsuted interfaces `TRUSTED_IF="tun0"`

7.2 CLIENT(S)

- `sudo apt-get install openvpn`
- `sudo cp /usr/share/doc/openvpn/examples/sample-config-files/client.conf /etc/openvpn/`
- Copy clients keys and server certificate to `/etc/openvpn/`
- Make sure the `client.conf` has

```
ca ca.crt
cert client1.crt
key client1.key
remote vpnserver.example.com 1194
```

- `service openvpn start`
- Check the tun interface: `ifconfig tun0`

Change ssfvpnclient1 for the actual name of the client in the vpn.

7.3 TEST

Ping server and clients from server and clients.

NOTE: You will have to add the network 10.88.55.0/24 to the trusted networks for nfs and nis.

On the server, the file /etc/openvpn/ipp.txt can be used to configure the ips for each client.

8 Comandos de bash múltiples máquinas (paralelos)

The original I was using was the c3 tools. Now, in Ubuntu, I will switch to kanif, which is a wrapper over TakTuk, a tool for large scale cluster managment.

DNS or hostname resolution must be supported and working.

8.1 SERVER

kanif install

- `sudo apt-get install kanif`
- The configuration is the same for c3, and is stored at `/etc/kanif.conf`

Passwordless ssh

- `ssh-keygen`
- `ssh-copy-id -i ~/.ssh/id_rsa.pub remote-host`

On c3 use ip numbers or hostnames only if name resolution is working.

8.2 TEST

`kash ls -l` should work, showing results on each computer.

9 Configuración del NFS

NFS allows to share directories across the net in transparent form. We are going to share the home directory.

9.1 SERVER

- Install the NFS tools :

  ```
  sudo apt-get install nfs-kernel-server
  ```

- Configure the export by adding the following to `/etc/exports`:

  ```
  /home 192.168.123.0/255.255.255.0(fsid=0,rw,sync,no_root_squash,no_subtree_check) \
  10.88.55.0/255.255.255.0(fsid=0,rw,sync,no_root_squash,no_subtree_check)+
  ```

- `service nfs-kernel-server restart`

Edit `/etc/hosts.allow` and `/etc/hosts.deny`, related with the security of RPC daemon:

- `/etc/host.deny`

  ```
  rpcbind nfsd portmap lockd mountd rquotad statd : ALL
  ```

- `/etc/host.allow`

  ```
  rpcbind nfsd portmap lockd mountd rquotad statd : 127.0.0.1, 192.168.123.0/24 , 10.88.55.0/24
  ```

9.2 CLIENT

- `sudo apt-get install nfs-common`
- `mkdir /home`
- (Internal clients) Add the following to `/etc/fstab`

  ```
  192.168.123.1:/home     /home   nfs     rw,hard,intr    0       0
  ```

- (VPN clients) Add the following to `/etc/fstab`

  ```
  10.88.55.1:/home     /home   nfs     rw,hard,intr    0       0
  ```

You could add `nfs` to `/etc/modules` to load it at startup.

9.3 TEST

On the clients, run `mount -a` and check all the exported resources are mounted.

10 NIS

This service allows to authenticate users on several machines from a central database. `https://help.ubuntu.com/community/SettingUpNISHowTo`

10.1 SERVER

- Add to /etc/host.allow :

  ```
  portmap ypserv ypbind : 127.0.0.1, 192.168.123.0/24 , 10.88.55.0/24
  ```

- sudo apt-get install portmap nis . If you are asked for the nis domain, use the current in use ssfservernis .
- Nis domain (if not configured previously):

  ```
  echo domain_name > /etc/defaultdomain
  ```

  where domain_name is , in this case, ssfservernis .
- Edit /etc/default/nis and set the NISSERVER line to NISSERVER=master, and NISCLIENT=true to NISCLIENT=.
- Edit /etc/yp.conf and add a server line of the form:
  ypserver 192.168.123.1
- NIS Makefile edit:

  ```
  # cd /var/yp
  (edit Makefile: MINUID=1000)
  (edit Makefile: MINGID=2)
  (edit Makefile: MERGE_PASSWD = true -> false)
  (edit Makefile: MERGE_GROUP = true -> false)
  ```

- If desired, edit /etc/ypserv.securenets and add lines to restrict access to domain members. I use lines for specific hosts, like:

  ```
  host 192.168.1.1
  host 192.168.1.2
  ```

  And comment out the 0.0.0.0 stuff.
- make
- Build the database for the first time
  sudo /usr/lib/yp/ypinit -m
- make
- service portmap restart
- Check nice output from rpcinfo -p localhost

10.2 CLIENT

- Add server to /etc/hosts. This means that you can still find the server if there is a DNS failure.
- sudo apt-get install portmap nis
- Domain name (if not asked or wrongly setted up previously):

  ```
  echo domain_name > /etc/defaultdomain
  ```

- Internal clients: add to /etc/yp.conf : ypserver 192.168.123.1
- VPN clients: add to /etc/yp.conf : ypserver 10.88.55.1

- Setup login and password files to use NIS

```
echo +:::::: >> /etc/passwd
echo +::::::::: >> /etc/shadow
echo +::: >> /etc/group
```

- Reboot the system (sometimes is the only solution ot make NIS work)

10.3 TEST

Restart both server and clients and check login.

11 BACKUP

Use the backup system of Ubuntu

11.1 SERVER

11.2 CLIENT

12 Recrear la base de datos a partir de una ya existente

12.1 SERVER

```
#!/bin/bash

for usernamedir in /home/*; do
    if [ -d $usernamedir ]; then
username=$(basename $usernamedir)
if [ "ftp" != "$username" ] && [ "localuser" != "$username" ] ; then
    #echo "Deleting account $username"
    #userdel $username
    echo Creating account $username
    useradd -d /home/$username -G audio,cdrom,floppy,plugdev,video -m -s /bin/bash $username
    echo "Changing password for $username to ${username}123"
    echo ${username}:${username}123 | chpasswd
    echo "Recursive chown ... &"
    chown -R $username.$username /home/$username &
fi
    fi
done
echo "Updating nis database"
make -C /var/yp/
service portmap restart
service ypserv  restart

echo "DONE."
```

Restart nis :

```
service portmap restart
service ypserv restart
/etc/init.d/nis restart
```

12.2 CLIENT

```
service portmap restart
/etc/init.d/nis restart
```

12.3 TEST

Log with different users on different computers

13 QUOTA

See quota-howto. Set the quota por a propotype user, configure fstab to mount with quota support, and assign the quota for all users using the prototype. Journaled quota (to speed up quota check) will be configured.

13.1 SERVER

```
sudo apt-get install quota quotatool
```

Modified fstab (change /dev/sda5 in favor of your local config):

```
/dev/sda5        /home        reiserfs    defaults,usrjquota=aquota.user,jqfmt=vfsv0         1   2
```

Now execute:

```
touch /home/aquota.user
chmod 600 /aquota.*
mount -o remount /home
```

Add to crontab:

```
0 3 * * 0 /sbin/quotacheck -avug
```

RESTART
Edit the prototype user:

```
export EDITOR=emacs
edquota -u tmpuser
```

For 3.2GB, set soft to 3000000, and hard to 3200000. Typical grace time is 7 days.
Now edit the cuota for the remaining users.

```
for a in $(cat /etc/passwd | awk '{ FS=":";if ($3 > 1000) print $1}'); \
do edquota -p testuser $a ;done
```

Por último, se puede editar la cuota de usuarios particulares para que tengan menos o más quota.

14 Impresora (to be updated)

El SSF cuenta con una impresora que tiene su propia tarjeta de red. El ip asignado a la impresora es 192.168.123.9 . En este caso se configurará CUPS para que "hable" con esa impresora, una HP Laserjet 2420dn.

14.1 SERVER

- Add the network printer via the cups interface on localhost:631
- Backup the cups directory on each slave:
  `cexec :1-2 "cp -a /etc/cups ~/cupsOLD"`

- Copy (overwrite) the server cups directory to the slaves and restart:
  ```
  tar czf cups.tar.gz /etc/cups
  cpush :1-2 cups.tar.gz cupsHEAD.tar.gz
  cexec :1-2 "tar xzf cupsHEAD.tar.gz"
  cexec :1-2 "rsync -av etc/cups/ /etc/cups/"
  cexec :1-2 "/etc/rc.d/rc.cups restart"
  ```

14.2 TEST

Print test pages on each computer

15 Scanner (to be updated)

The scanner at the office is a Benq 4300 old scanner. To make it work, a firmware is needed, look at the information related to sane and snapscan backend, `http://snapscan.sourceforge.net/` . The scanner is a Acer / Benq 4300 FlatbedScanner23, something that you can know by running the commands `sane-find-scanner` and, if successfull, `scanimage -L` .

Fortunately, SLackware 13.1 detects successfully the Scanner!

They say it is a `FlatbedScanner21`, but the working firmware is for the `FlatbedScanner23` From this information, we know we need the firmware `u176v046.bin`. It is quite tricky to find this file, you can google and/or try to download it from the vendor site. One link is `http://outlands.ca/linux/snapscan-firmware.html` `http://ubuntu-col.blogspot.com/2009/03/ubuntu-8_18.html` . After downloading, copy the firmware `u176v046.bin` to the folder `/usr/share/sane/snapscan/`, open the file `/etc/sane.d/snapscan.conf` and put the full path to the firmware. Unplug and unplug the scanner. Done. You can scan by using xsane or gimp or xscanimage.

The users will to be in the group scanner. Restart the messagebus `/etc/rc.d/rc.messagebus restart`

To share the scanner on the network: `http://tldp.org/HOWTO/Scanner-HOWTO/sane.html#CONFIG-SANE`

15.1 SERVER (PC with the scanner)

```
if ! id saned; then
 groupadd saned;
 useradd -g saned -G scanner -s /bin/false -d /dev/null saned;
fi
```

Add the network range in `/etc/sane.d/saned.conf`

Add the saned service, `sane 6566/tcp` , to `/etc/services`

Add

```
sane-port stream tcp nowait saned.saned /usr/sbin/tcpd/usr/sbin/saned
```

to `/etc/inetd.conf`

Add ALL:ALL to /etc/hosts.allow

Restart inetd

15.2 CLIENT

- Add ip address of scanner server to the file `/etc/sane.d/net.conf`