HP World/Interex 2002

Linux Network Configuration Basics

Chris Cooper
(734) 805-2172
chris_cooper@hp.com

George Vish II
(404) 648-6403
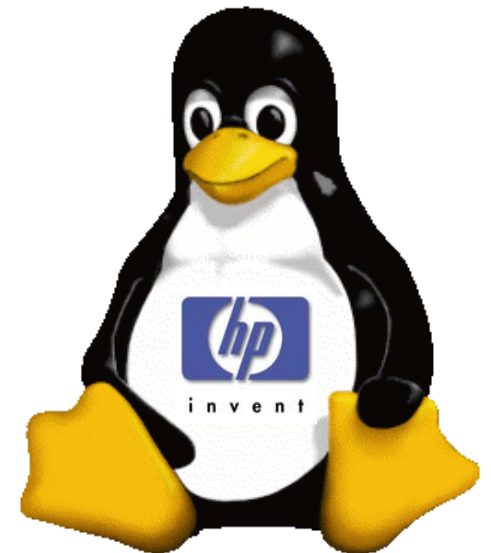george_vish@hp.com

Linux
Network Configuration Tools and Techniques
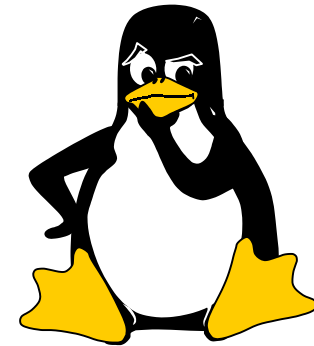
Version A.00

U2794S   Module 14 Slides

2

Internet Services

- What is a server?

- What is a client?

- What is a daemon?

- Internet x**inetd** (or **inetd**) daemon


```
/etc/services
/etc/xinetd.conf (/etc/inetd.conf)
```

Configuration Tools

- Universal commands to configure network interfaces:

  # `ifconfig`   (a command line utility)

  # `/etc/rc.d/init.d/network restart`   (an rc script)

- Kernel modules must be made available, configured through:

  `/etc/modules.conf`

- Kernel modules are loaded through:

  # `modprobe`

- Red Hat:

  # `netconf`   (a GUI)

  # `netconfig`   (a TUI)

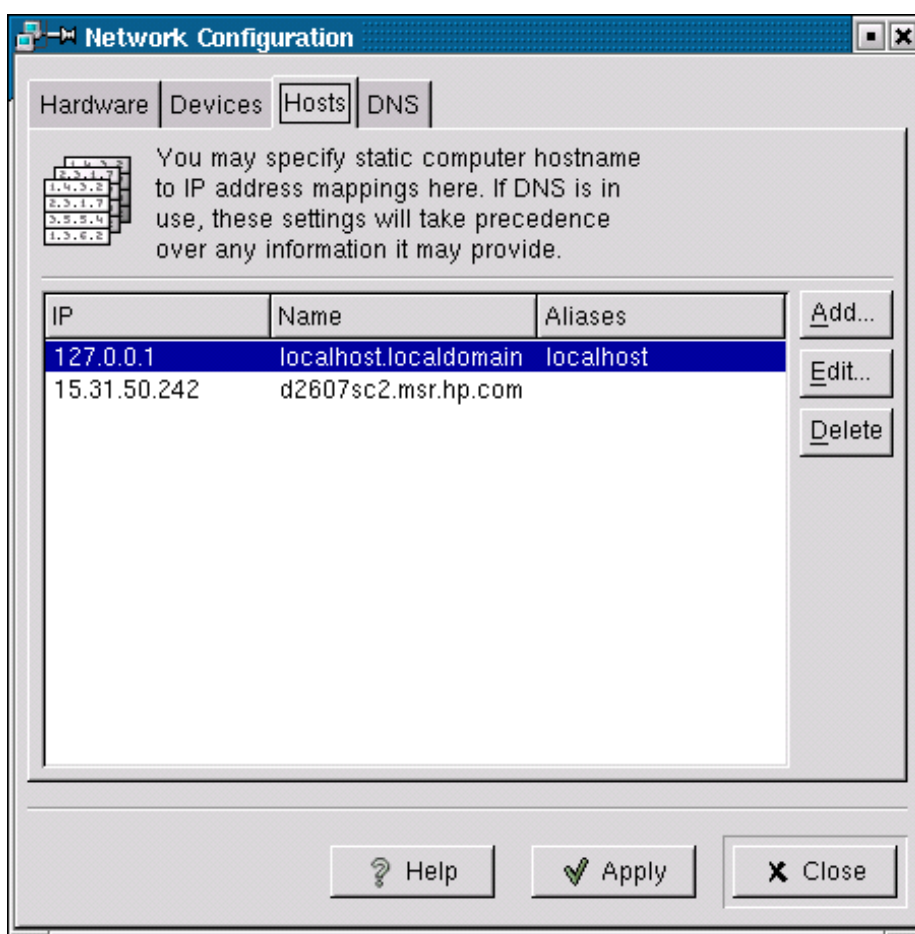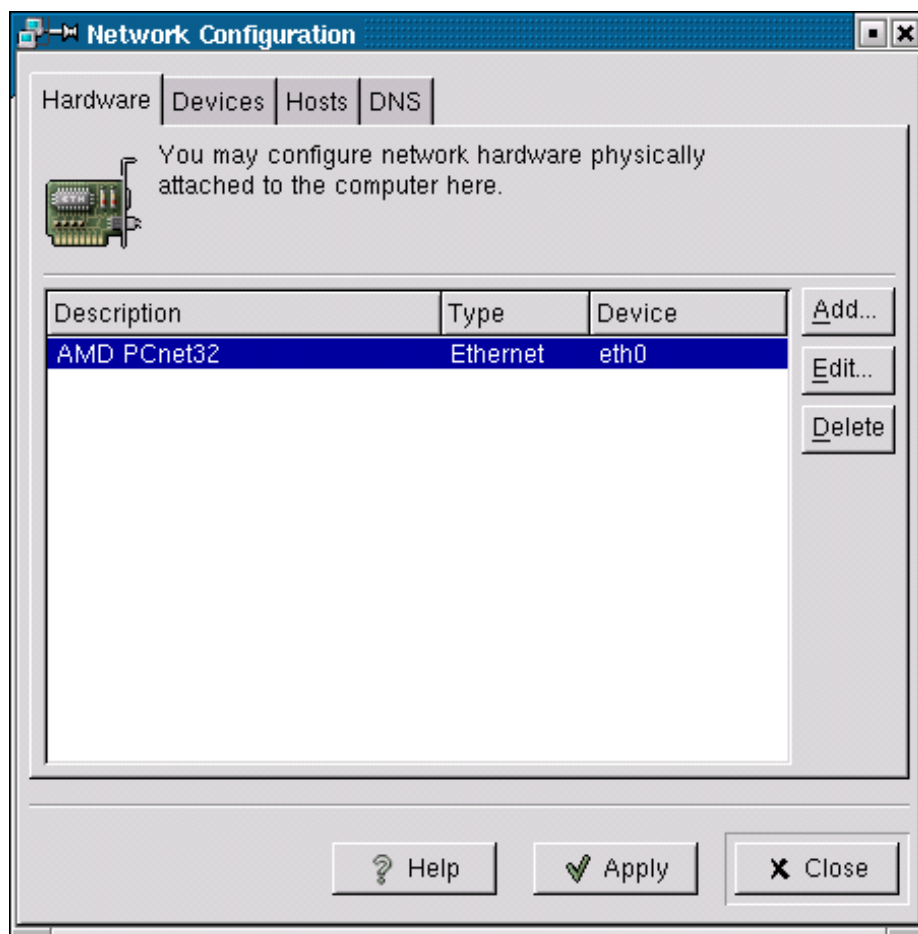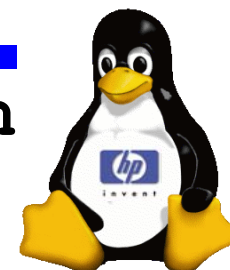 **# `linuxconf`**   (soon to be depricated)

- SuSE:

  Yast and Yast2

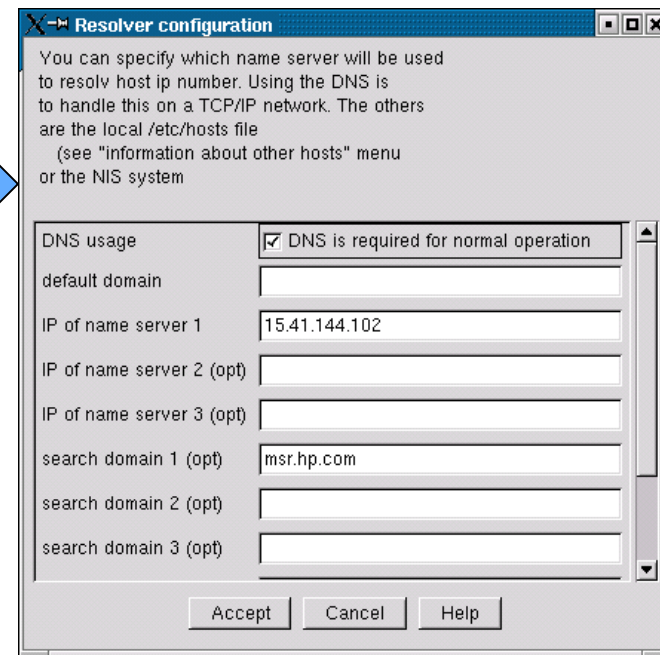**-> Start Here -> System Settings -> Network Configuration**

-> System -> Network Configuration

-> System -> Network Configuration

U2794S A.00

© 2002 Hewlett-Packard Company

netconf

Network configurator

This package allows you to configure a
TCP/IP network from scratch using ethernet
and modem (or other serial connection)

| Client tasks | Server tasks | Misc |

- Host name and IP network devices
- Name server specification (DNS)
- Routing and gateways
- IPX interface setup

Quit   Help

Resolver configuration

You can specify which name server will be used
to resolv host ip number. Using the DNS is
to handle this on a TCP/IP network. The others
are the local /etc/hosts file
   (see "information about other hosts" menu
or the NIS system

DNS usage	☑ DNS is required for normal operation
default domain	
IP of name server 1	15.41.144.102
IP of name server 2 (opt)	
IP of name server 3 (opt)	
search domain 1 (opt)	msr.hp.com
search domain 2 (opt)	
search domain 3 (opt)	

Accept   Cancel   Help

`-> Start Here -> System Settings -> Network Configuration`

Networking Configuration Files

```
/etc/hosts
/etc/host.conf
/etc/networks
/etc/resolv.conf
```
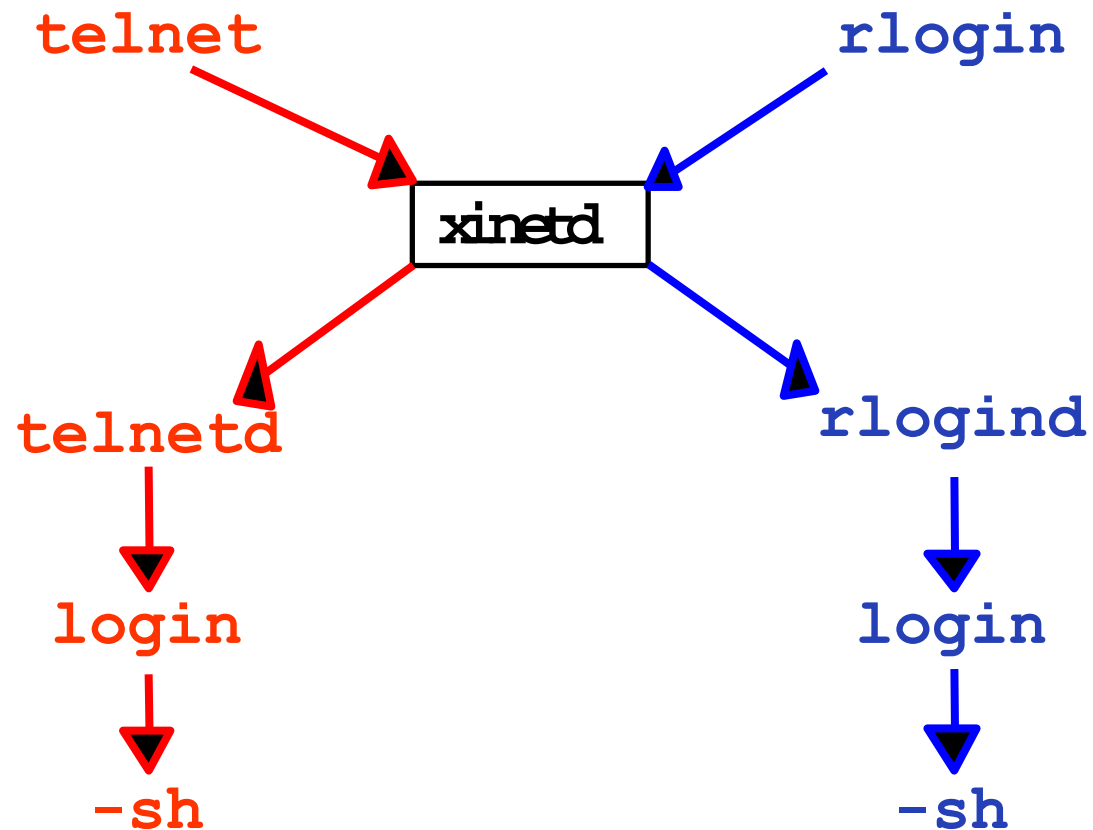
- Startup script configuration:
    - SuSE: `/etc/rc.config`
    - Red Hat: `/etc/sysconfig/network/`…
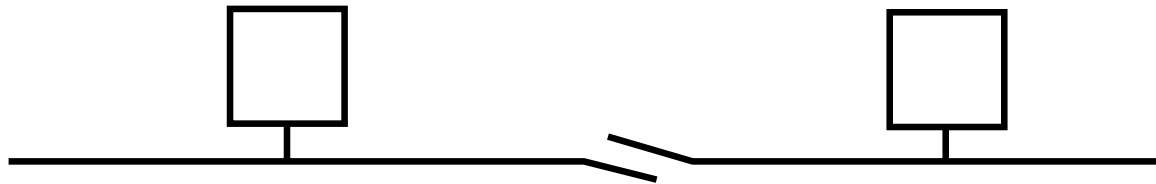
Name Service Switch

- **`/etc/nsswitch.conf`** allows you to specify what name services should be queried and in which order.

  - **`dns`**

  - **`nis`**

  - **`hosts`**

  - **`.db`**

- hosts: **`files [NOTFOUND=continue] dns`**

telnet & **rlogin**

telnet   rlogin

xinetd

telnetd   rlogind

login   login

-sh   -sh

Troubleshooting Network Connectivity

- Use a methodical approach.

- Work from the bottom upward.

- Divide the problem into pieces, and solve the pieces one at a time.

- Keep a record of your problem-solving tasks.

- Keep an open mind about what may be causing the problem.

- Be aware of security barriers.

- Pay attention to error messages.

- Try to duplicate the problem and solve it in a known environment.

- Stick to a few troubleshooting tools that you understand thoroughly.

- Don't neglect the obvious.

U2794S A.00   © 2002 Hewlett-Packard Company

Some Common Problems

- The LAN interface has failed due to a hardware fault.

- The network card driver is not active or is intended for another piece of hardware

- The network card driver may not have been compiled into the kernel or may not be available as a kernel module.

- The LAN interface has the wrong IP address.

- The subnet mask is wrong.

- The same IP address is used by two systems.

- The routing table is not configured correctly.

- The router is not available.

- The LAN cable is defective, or the LAN segment is much too long.

- The **/etc/hosts** file is incorrect or corrupt.

Information about your configuration

- Run **ifconfig** (with no options) to obtain the MAC address of your Ethernet card:

```
eth0   Link encap:Ethernet   HWaddr 00:10:4B:2E:C0:DB
inet addr:192.168.100.2  Bcast:192.168.100.255  Mask:255.255.255.0
UP BROADCAST RUNNING MULTICAST   MTU:1500   Metric:1
RX packets:0 errors:0 dropped:0 overruns:0 frame:0
TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:100
Interrupt:11 Base address:0xfc40
```

- The **HWaddr** field contains the MAC address.

- The **inet** field contains the Internet address.

- The **Bcast** field contains the broadcast address.

- The **Mask** field contains the subnet mask.

- Other fields contain performance statistics and hardware information.

Testing your resolve

 # nslookup   *Queries name servers*

– Interactive mode

– Line mode, pass it either the hostname or an IP-Addr. to resolve
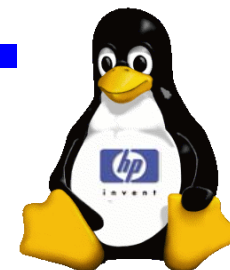
– Is on the list for deprecation !

 # dig   [@server] domain [query-type] [query-class]
[+query-option] [-dig-option] [%comment]

– **dig** *sends domain name query packets to name servers*

 # host   <hostname|IP-addr>

– host is a simplified utility for performing DNS lookups.

Networking sonar ... **ping** ...

ping tests the IP connectivity to a remote system.

```
Syntax: ping hostname [packet_size][-n [num_packets]]
Example:

# ping hpserv -c 4
PING hpserv: 64 byte packets
64 bytes from 15.3.44.103: icmp_seq=0. time=8. ms
64 bytes from 15.3.44.103: icmp_seq=1. time=5. ms
64 bytes from 15.3.44.103: icmp_seq=2. time=5. ms
64 bytes from 15.3.44.103: icmp_seq=3. time=6. ms
---- hpserv PING Statistics ------
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms) min/avg/max = 5/6/8
```
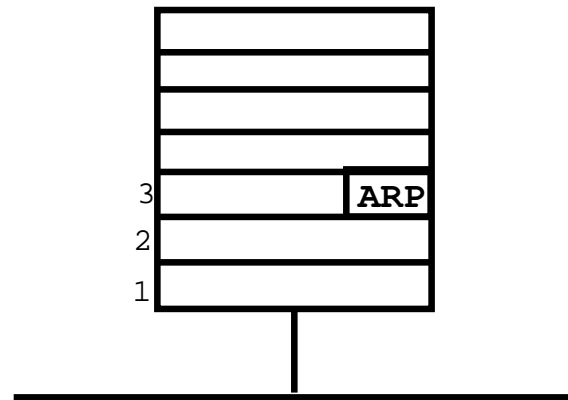
Using **arp**

- **arp** stands for **address resolution protocol**.
- **arp** is used to display or modify entries in the ARP table.
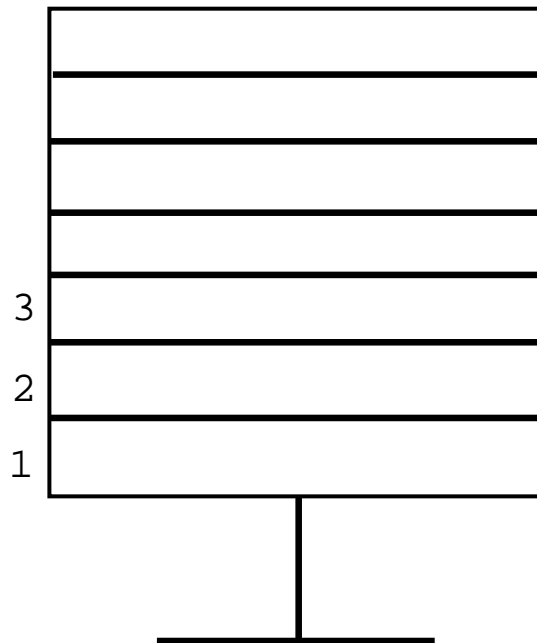- The modifying options require root privileges.

```
Example
# /usr/sbin/arp -a
 leo   (15.77.1.3) at 8:0:9:10:4:b1 ether
 hpserv (15.77.2.3) at 8:0:9:16:be:53 ether
```

netstat

netstat reports the status of the local LAN interfaces and reports network and protocol statistics.

```
3
2
1
```

netstat -r[n]
netstat -I[n]

Syntax: netstat -*[many_opt][some_args]*

Remote
File Systems
NFS & Samba

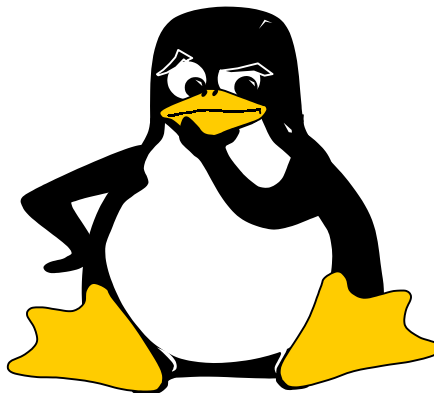Version A.00

U2794S   Module 16 Slides

NFS and Linux

**The Network File System (NFS)**

- It is a distributed file system.

- It allows transparent access to files and directories on remote systems.

- Remote files look just like local files.

Sharing Directories

- A server shares specified directories to the network.

- Clients mount the shared directory to a local mount-point.

- To the client, the files in the shared directory appear to be local files.
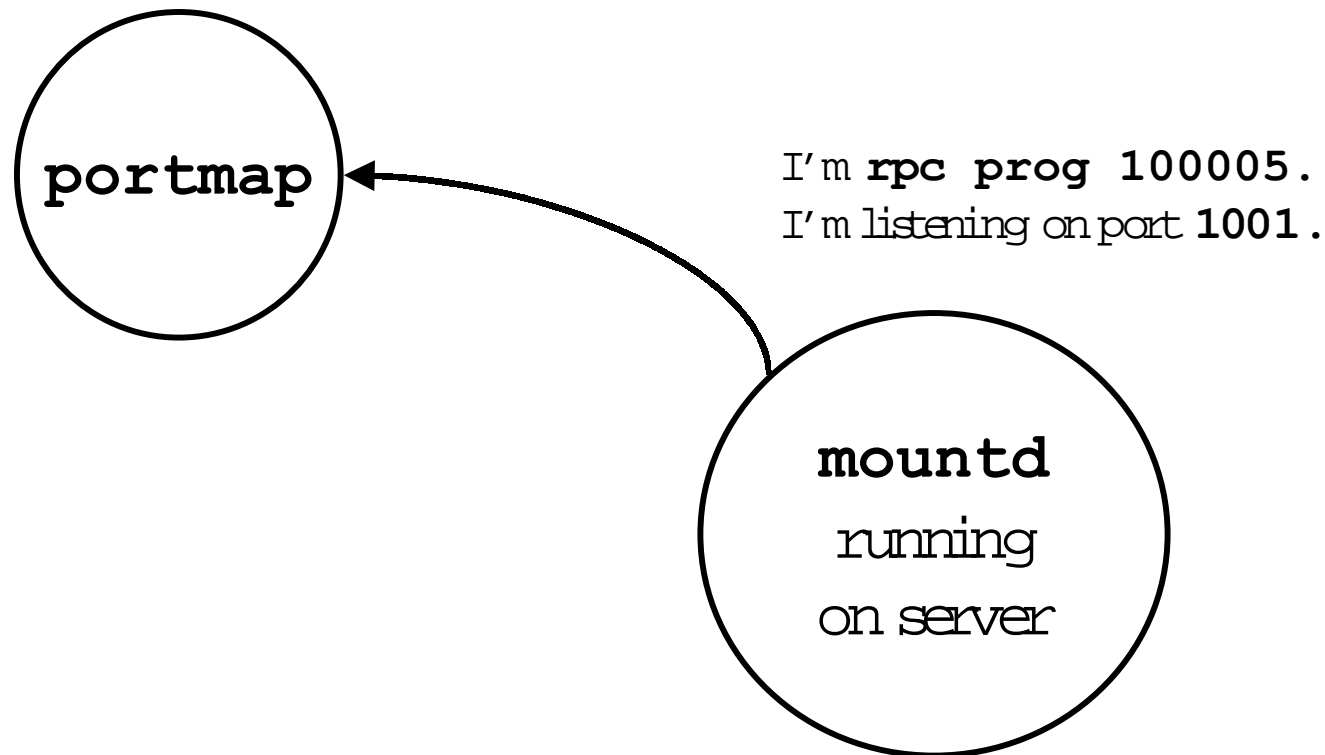
| server | | NFS mount | client |
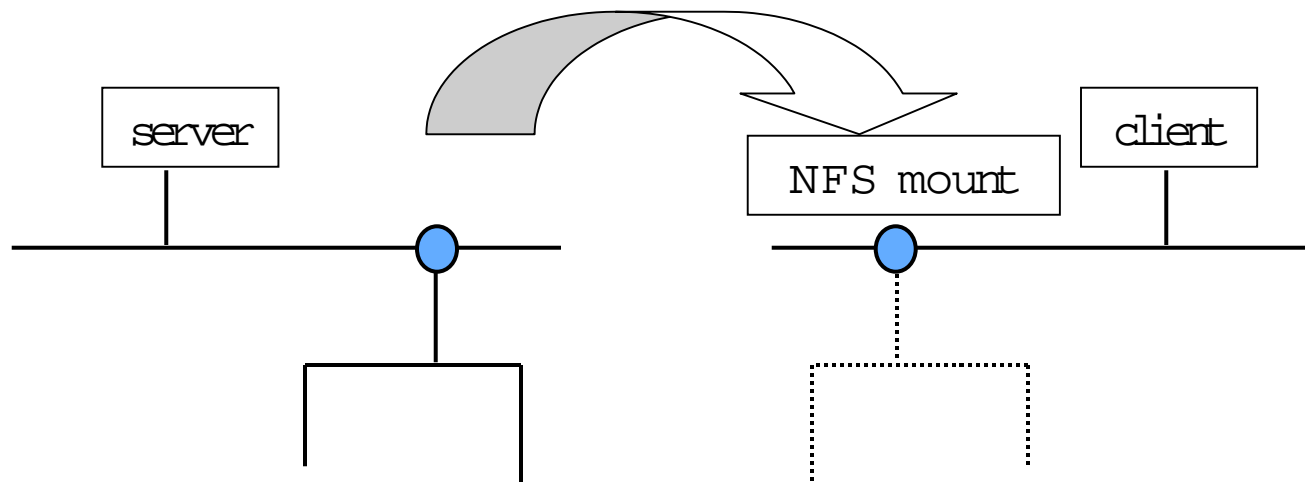
NFS Daemon Processes

- There are daemon processes for NFS server activity and client activity.

- Server daemons are:
    - **rpc.portmapper**
    - **rpc.nfsd**
    - **rpc.mountd**

- The client daemon is:
    - **rpc.portmapper**

RPC Client Processes

portmap

I'm **rpc prog 100005.**
I'm listening on port **1001.**

mountd
running
on server

Use **rpcinfo -p** to see view your **rpc** configuration.

The **/etc/exports** File

- The **/etc/exports** file contains the names of the directories to be shared to the network (one directory per line).

- Each entry can be followed by a set of hostnames with associated options.

```
# cat /etc/exports
/usr/share/doc  *.hpclassroom.com(ro,no_root_squash)
/home/jeff      *.hpclassroom.com(ro,no_root_squash)
```
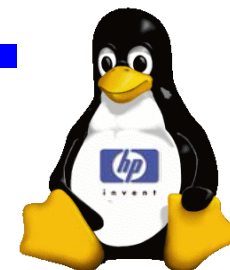
Export Options

- The options control how the directory will be shared to the network.

- A range of options is available. For example:

```
/usr/share/pfiles      europa(ro)
/usr/share/pfiles      europa.hpclassroom.com(ro)
/usr/share/pfiles      europa(ro) …  earth(root_squash)
/usr/share/pfiles      *.hpclassroom.com(ro)
```

The **exportfs** Command

- The **exportfs** command reads the **/etc/exports** file and exports (*shares*) all the appropriate directories.

- The **exportfs** command can be used, on the command line, to share a directory that does not have an entry in the **exports** file.

- Command options can be used to designate the share options.

- Examples:

```
exportfs -a[v]  [-o options] [client:/pathname]

exportfs -r[v]  [-o options] [client:/pathname]

exportfs -u[v]  [client:/pathname]

exportfs [-v]
```

Starting NFS Daemons

- View all currently running daemons:

  ```
  # ps aux | more
  ```

- Examine **/etc/services** for **portmapper** (port number **111**).

- Start the **nfsd** and **mountd** daemons manually:

  ```
  # /etc/rc.d/init.d/nfs start
  ```

- Stop and restart NFS whenever **/etc/exports** is modified.

Mounting NFS Directories (the Client Side)

- The client would mount the shared directory in a similar manner to mounting a local disk file system.

- Extra options are available (and applicable) when mounting an NFS file system.

```
# mount  -t  nfs  europa:/usr/doc   /localdir1


# mount -t nfs  /europa:/usr/share/pfiles /pfiles
```

NFS Mount Options

- Because files are being shared across network connections, NFS provides many more mount options.

- Some options are Boolean logic options, which are turned either on or off.

  suid   Turns ON the acknowledgement of files with
          suid permissions.
  nosuid   Turns this option OFF.

- Some options require argument values.

  wsize=8096   Sets the write block size to 8K bytes

Mounting NFS File Systems at Boot Time

- The `mount` information can be retained by adding an entry into the `/etc/fstab` file, allowing the NFS mount to occur at boot time.

- Instead of using a device pathname, the combination of server-name and directory-pathname, separated by a colon, would be used as the name of the device to mount from.
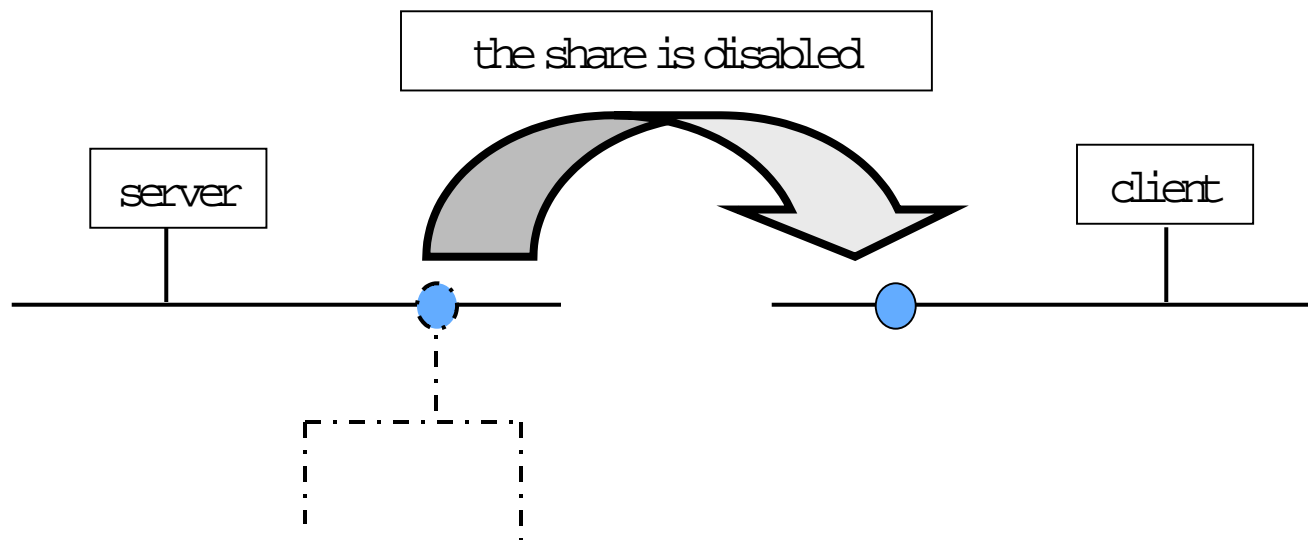
```
server:/pathname  /mount_point_dir  nfs option[s] 0 0

europa:/usr/share/pfiles /pfiles    nfs ro        0 0
```

Unexporting Directories

- Shared directories can be unexported.

- The **exportfs -u** option is used to unexport.

the share is disabled

server

client

29

Validating Current NFS Mounts

- All current mounts are stored in the **/var/lib/nfs/xtab** file.

- To see which directories are currently being mounted by clients, use the **showmount** command.

- The options used to create and sustain the mount are also displayed.

    #   showmount -a   <— to see who is currently sharing your files
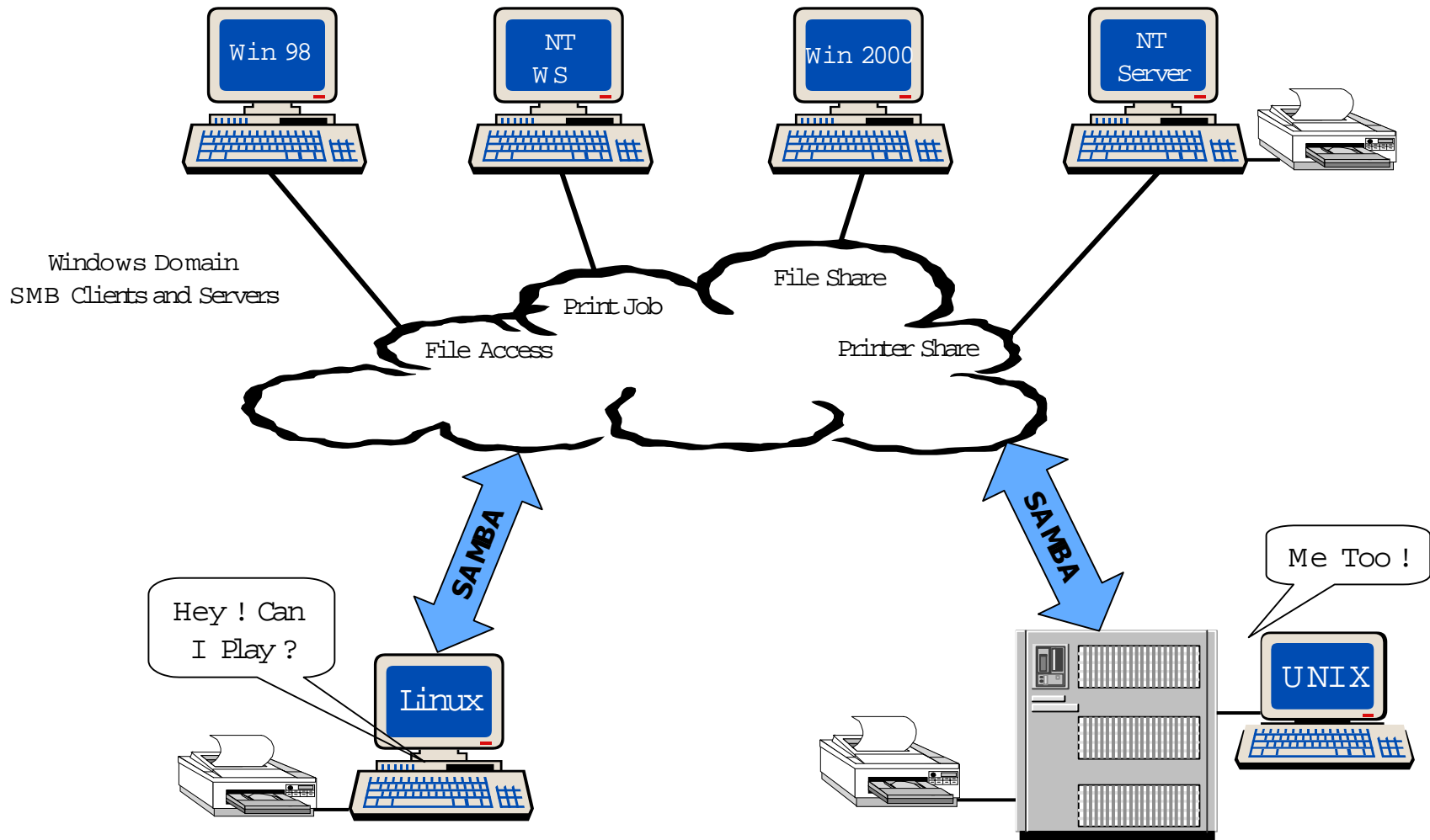
    # showmount -e <— to see what you have exported

Common NFS Server Problems

- Improper configuration of **/etc/exports**

- No **mountd** daemon running

- NFS server down or heavily loaded

- Failure to register the RPC properly
  - Usually caused by invoking **portmap** and **inetd** in the wrong order

The SAMBA Model

Windows Domain
SMB Clients and Servers

Win 98   NT WS   Win 2000   NT Server

Print Job

File Share

File Access

Printer Share

SAMBA

SAMBA

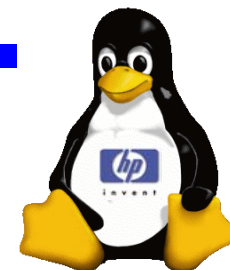Hey ! Can I Play ?

Linux

Me Too !
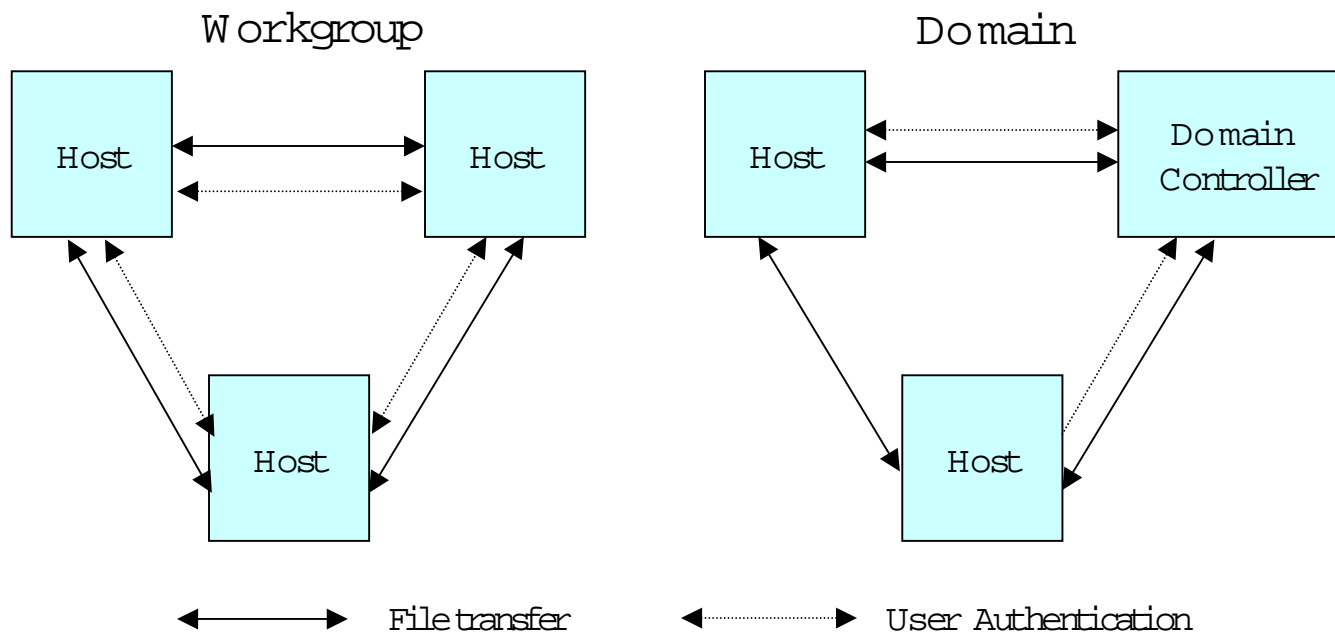
UNIX

The Server Message Block Protocol

- The Server Message Block protocol (SMB) is used by Microsoft Windows clients for local area networking.

- SAMBA is an SMB server for UNIX that allows a Linux host to participate in a Windows network as though it is running Windows.

Microsoft Networking Concepts

- Windows networks use peer-networking and are arranged in domains or workgroups.

The SAMBA suite for Linux

- The SAMBA package contains several programs:
  - **smbd**

    Daemon to provide SMB share access
  - **nmbd**

    Daemon to provide NETBIOS host naming
  - SWAT

    The SAMBA Web Administration Tool
  - **smbclient**

    A program to allow a UNIX host to act as an SMB client
  - A set of utilities to maintain SMB user information

- The entire SAMBA suite is contained in the SAMBA package.

The key players

On the Server

- smb.conf

- smbd
- nmbd
- smbclient
- smbtar
- nmblookup
- smbpasswd
- smbstatus
- testparm
- testprns

- swat
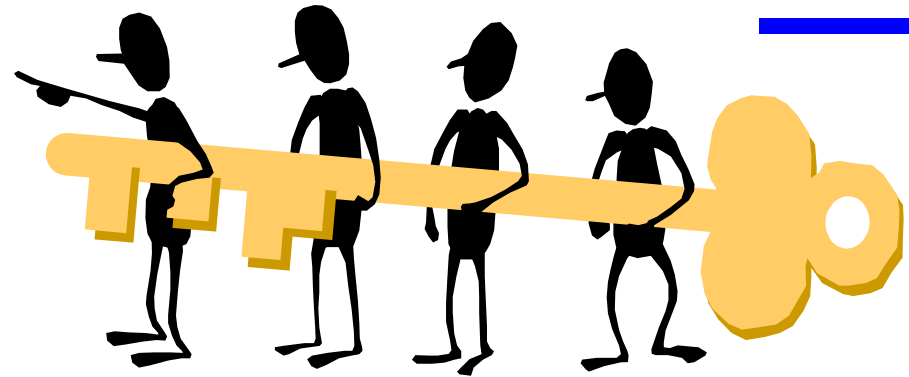
On the Client

- smbmount
- smbumount
    or
- smbwrapper

Activating the SAMBA Server

- The SAMBA script (**/etc/rc.d/init.d/smb**) starts and stops both the SMB and NMB daemons at the command line.

- SAMBA is not started by default at boot.

- Automatic startup can be enabled using **linuxconf**, **ntsysv**, or by manually adding a set of links in the **/etc/rc.d** directories.
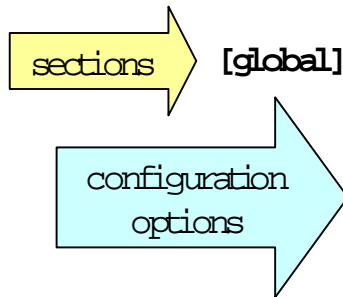
The `smb.conf` File

- SAMBA is configured with the **smb.conf** file.

- The configuration file is split into sections marked by a share name enclosed in square brackets [ ]. The first section within the **smb.conf** file is for global parameters. Defaults set here will be applied to other shares.

```
[global]
..
[share 1]
..
[share 2]
```

The master key:

smb.conf

sections → **[global]**

configuration options

```
log level = 1
max log size = 1000
socket options = TCP_NODELAY IPTOS_LOWDELAY
guest ok = no
read only = yes
```
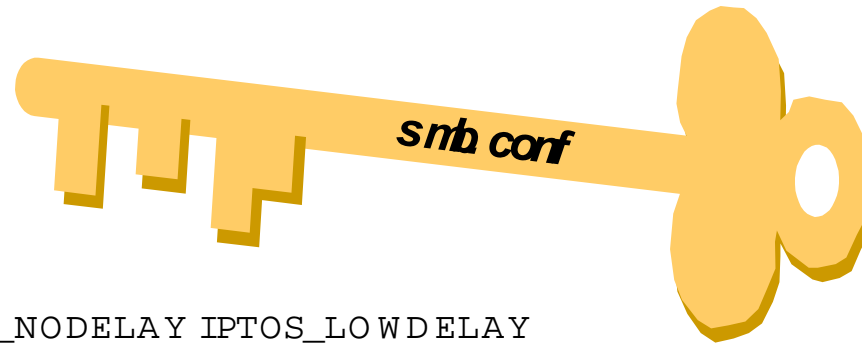
[homes]

```
browseable = yes
map archive = yes
```

[printers]

```
path = /usr/tmp
guest ok = yes
printable = yes
min print space = 2000
```

[sharename]

```
comment = "This is a shareable directory"
browseable = yes
writeable = yes
guest ok = yes
path = /local/sharepath
```

Use *testparm* to validate the *smb.conf* file

The *smb.conf* file is re-read by the **smbd** and **nmbd** daemons every 60 seconds (default setting)

Share Security

SAMBA supports four levels of security with the **security** command:

security = share   Every share is given a password.

security = user   Valid users may connect to the share.

security = server   User-level security, authenticated on another server.

security = domain   Server-level security, but the SAMBA server also joins a Windows NT domain.

Password Verification

- Non-NT Windows clients use unencrypted passwords to authenticate connections.

- Unencrypted passwords can be validated against the **/etc/passwd** file, but are insecure.

- NT clients use an encryption algorithm that is incompatible with UNIX password encryption, therefore, encrypted SAMBA passwords must be stored in a separate password file managed by **/etc/smbpasswd.**

The **smbclient** Program

- **smbclient** is an FTP-like SMB client that can be used to access windows networks and test SAMBA configurations.

- To list SMB services on a host:

  ```
  # smbclient -L <hostname> -U <username>
  ```

- To connect to an SMB service:

  ```
  # smbclient //<hostname>/<service> -U <username>
  ```

Linux as a Domain Controller

- In order for the SAMBA host to be considered the domain controller, it must be the *master* for the network browsing service, accept domain logons, and offer the **netlogon** service.

```
..
  domain logons = Yes
  os level = 64
  preferred master = Yes
  domain master = Yes
  local master = Yes
..
   [netlogon]
..
```

Adding NT Clients to a Domain

- SAMBA does not support the client-based creation of machine trust accounts, so adding new clients is slightly more complex than necessary.

- The machine trust account for a host is the host name followed by a dollar sign. Machine trust accounts are added using the **-m** and **-a** flags for **smbpasswd.**

```
smbpasswd -a -m neptune
```

Introducing the Amazing
S W A T !

○ Once you have installed **Samba** on your system it's behavior is controlled by the **smb.conf** configuration file.

This file may be edited manually or you may use the **swat** GUI (which is accessed through your local web browser)

Enabling SWAT

- SAMBA settings can be configured in a graphical environment using SWAT, which runs in a web browser.

- SWAT does not require installation or configuration of a web server on the SAMBA host. It is enabled through the **/etc/inetd.conf** and **/etc/services** files.

- SWAT is not compatible with hand-generated **/etc/smb.conf** files.