Samba as a file and print server

Michael MacIsaac
IBM
mikemac@us.ibm.com

Abstract

Samba as a file and print server

Although the material was created using z/VM and Linux on zSeries, this presentation will discuss Samba from a generic point of view. It will present the basics quickly in order to get to some relatively advanced topics such as winbind, ACLs and automatically downloading Windows printer drivers.

IBM @server. For the next generation of e-business.

Outline

1. Level set
2. Introduction to Samba, brief history
   - Samba services, binaries and documentation
3. Installation of Samba
   - Via Linux install/RPMs
   - Via source code
4. Configuration and customization of Samba
   - The Samba configuration file - smb.conf
   - Starting and stopping Samba
   - Setting up and using SWAT
   - Permissions and Access Control Lists (ACLs)
   - Authenticating users
   - Using winbind
   - Sharing files read/write in teams
   - Printing
   - Security
   - Performance and tuning
5. References

Level set - who am I?

Michael MacIsaac - Linux Center of Competency, Poughkeepsie, NY

- 15+ years with IBM
  - 10 years programmer
  - 5+ years with S/390
  - Led teams to produce redbooks in 2001:
    - Linux on zSeries and S/390: Distributions
    - Linux on zSeries and S/390: ISP/ASP solutions
- Linux (open source/freeware) advocate
- e-mail - mikemac@us.ibm.com

IBM @server. For the next generation of e-business.

Level set - who are you?

How is Linux in your enterprise?
- None yet
- Some in test only
- Some in production
- Majority of servers in production

On what platform will you work with Linux?
- S/390 + VM
- PC
- Other

Where have you used Samba?
- Not at all
- In test/personal
- Unofficial for small teams
- In production

What is your primary desktop OS?
- Windows
- Linux
- Other

Level set - Software licensing

Samba is shipped under GNU General Public License (GPL)
- You can run, copy and modify the software
- You can redistribute and charge $$ for the software
- You cannot add restrictions to the software
- You must make the source code available
- If you include software which is GPL'd, your software must also carry the GPL (viral nature)

BSD license does not have the viral nature

Level set - Enterprise perspective

Samba usually crosses enterprise-political boundaries

- Bridges the Windows and UNIX world
- Political and solution rules of thumb:
  - Windows clients should not have to be modified.
  - When a change is needed to the Windows clients, see rule 1.
  - Authentication decisions should follow the enterprise's security policy (or infrastructure)
  - The Window server administrators will probably not want to help you.
- Solutions such as AFS and Intermezzo are or will be technically superior:
  - Kerberos authentication
  - Caching servers
  - Local caching
  - "Replication" or synchronization and resolution of conflicts (Intermezzo)

Introduction - Samba history

- Coincidentally, started at the same time as Linux by Andrew Tridgell
  - At the Australian National University in Canberra, Australia
- NFS on Andrew's PC had to be removed
  - But he still wanted to share files from a UNIX server
- "Server 0.1" (later to be smbd) was released in January 1992
- Andrew announced a mailing list
  - Feb 1994 - 100+ people
  - May 1995 - 1400 people
  - March 1996 - 3000 people
- 1998
  - "Samba - Integrating UNIX and Windows" - first Samba book
  - Samba 2.0
  - SWAT
  - NT Domain logons
  - Performance improvements

See the file .../samba/docs/history

IBM @server. For the next generation of e-business.

Introduction - Services, binaries and documentation

Services

- File sharing - via "Map Network Drive" or DOS "net use"
- Browse lists - Network neighborhood/My network places or "net view"
- Print sharing - However, Samba is a small piece of a printing system
- Act as a Domain Controller

Binaries

- smbd - SMB daemon - file sharing
- nmbd - NetBIOS name daemon - browse lists
- winbind - Name service switch daemon - Domain authentication
- swat - Samba Web Administration Tool - mini Web browser
- smbpasswd - Password crypt to make MS compatible passwords
- smbclient - Client with FTP-like interface

Documentation - lots of it, but overall is poor

- Books
  - *Using Samba*, Robert Eckstein, et al is free - available through SWAT
  - *Samba Essentials for Windows Administrators*, Gary Wilson
- HOWTO collection - many formats
- A **lot** of documentation is out of date

Introduction - Samba versions and "head's up"

Samba versions

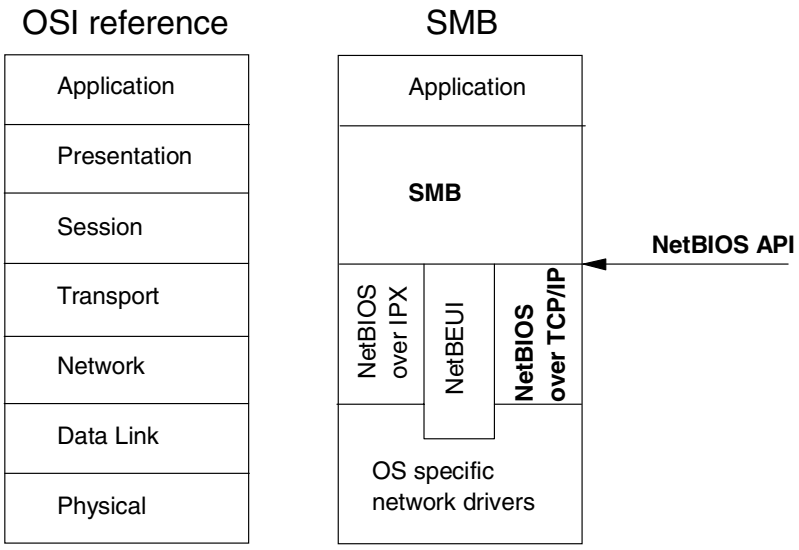
- 2.2.3a - stable and recommended?
- 2.2.4 - winbind will core dump on s390 (but not i686)
- 2.2.5 - June 18, 2002 - most recent (??most stable??)

Head's up

- Bug in Boeblingen developerWorks code will cause kernel to "oops"
  - Writing to a Samba share will cause it
  - SuSE SLES-7 (2.4.7) requires kernel patches 3 and 4 (1+2 are applied)
- Avoid Samba 2.2.4

IBM @server. For the next generation of e-business.

Introduction - Services, binaries and docs (cont'd)



Installation - Via Linux install/RPMs

Samba comes installed with most Linux distributions

- samba-2.2.0a-21 with SuSE SLES-7 (Oct 2001)

```
# rpm -qa | grep samba
samba-2.2.0a-21
```

- samba-2.2.0-20010417 with Turbolinux v6.5
- samba-2.2.1a-5 with Red Hat 7.2

Samba RPMs are available:

- From the distributor, on the CD
  - You can install manually:

```
# rpm --install /suse/cd1/suse/n2/samba.rpm
```

  - With SuSE, you can use yast:

```
yast ->
  Package Management (Update, Installation, Queries) ->
    Change or create configuration ->
      n     Network-Support (TCP/IP, UUCP, Mail, News) ->
        [i] samba             An SMB file server for UNIX
```

Installation - Via Linux install/RPMs

Updating Samba

- Possible approaches:
  1. Build Samba in the default directory /usr/local/samba
  2. Replace all Samba files in the correct location in the distribution
  3. Create your own RPM from a source RPM
  4. Get an updated RPM for your distribution and platform

A warning from the Samba developers:

```
     # tail -7 Manifest
     NOTE: OS Vendors who provide Samba binary packages will generally
integrate all Samba files into their preferred directory locations.
These may differ from the default location ALWAYS used by the Samba
sources. Please be careful when upgrading a vendor provided binary
distribution from files you have built yourself.
```


Installation - Via Linux install/RPMs (cont'd)

Get an updated RPM for your distribution and platform (4)

- Get s390 RPMs from the distributor (IF you have support)
- You can get i686 RPMs from the Internet

Create your own RPM from a source RPM (3)

```
# cd /usr/src/packages/SRPMS
ftp suse.distro.server
ftp> cd suse-sles7/cd2/suse/zq1
ftp> get samba.spm
ftp> quit
# rpm --rebuild samba.spm
...
# cd ../SPECS
# head -2 samba.spec
#
# spec file for package samba (Version 2.2.0a)
```

- Modify the spec file ??

```
# cd ../RPMS/s390
# ls
samba.rpm
# rpm -Uvh samba.rpm
...
```

Installation - Via source code

Build Samba in the default directory (1)

- Download Samba - via tar file or CVS
  - Via tar file
    - Go to `http://www.samba/org` -> choose a download mirror
    - For example:
      `ftp://us6.samba.org/pub/samba/`
    - Get a tar file - for example:
      `samba-2.2.5.tar.gz`
  - Via CVS - need some form of direct access to the Internet

```
$ export CVSROOT=:pserver:cvs@pserver.samba.org:/cvsroot
$ cvs login
$ cvs -z3 checkout -r SAMBA_2_2 samba
```

- Build from source

```
# cd /usr/src/samba
# tar xzf samba-2.2.5.tar.gz
# ln -s samba-2.2.5 samba
# ls -ld samba
lrwxrwxrwx  1 root   root   12 Mar 13 13:56 samba -> samba-2.2.5/
# cd samba
# cd source
# ./configure --with-winbind --with-acl-support --with-smbmount
# make
...
# make install
...
```


Installation - Via source code (cont'd)

- Set your environment
  - Set PATH and MANPATH env variables in $HOME/.bash_profile to pick up Samba executables and man pages first.

```
# cd /root
# vi .bash_profile
...
# grep PATH .bash_profile
export PATH=/usr/local/samba/bin:$PATH
export MANPATH=/usr/local/samba/man:$MANPATH
```

  - Run your profile in your current shell and verify settings

```
# . .bash_profile
# which smbd
/usr/local/samba/bin/smbd
# man -w smbd
/usr/local/samba/man/man8/smbd.8
```

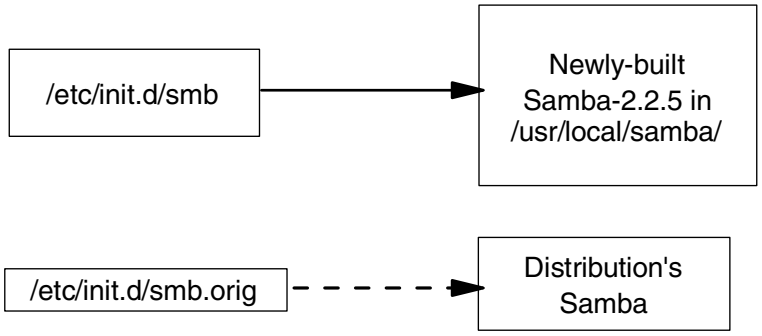
  - Sample /root/.bash_profile:

```
export PATH=/usr/local/samba/bin:$PATH:$HOME/bin
export MANPATH=/usr/local/samba/man:$MANPATH
export SMB=/usr/local/samba
export USERNAME="root"
set -o vi
alias ls='ls -F'
```

Installation - Starting and stopping Samba

- Integrate recompiled Samba (/usr/local/samba) into your system
  - Modify the smb script:

```
# cd /etc/init.d
# cp smb smb.orig
# vi smb
...
```



Installation - Starting and stopping Samba (cont'd)

- Modified smb script on SuSE:

```
# diff smb smb.orig
19,23c19,23
< SMB_BIN=/usr/local/samba/bin/smbd
< NMB_BIN=/usr/local/samba/bin/nmbd
< SMB_CONF=/usr/local/samba/lib/smb.conf
< SMB_PID=/usr/local/samba/var/locks/smbd.pid
< NMB_PID=/usr/local/samba/var/locks/nmbd.pid
---
> SMB_BIN=/usr/sbin/smbd
> NMB_BIN=/usr/sbin/nmbd
> SMB_CONF=/etc/smb.conf
> SMB_PID=/var/lock/samba/smbd.pid
> NMB_PID=/var/lock/samba/nmbd.pid
```

- Modified smb script on Red Hat:

```
# diff smb smb.orig
35c35
< [ -f /usr/local/samba/lib/smb.conf ] || exit 0
---
> [ -f /etc/samba/smb.conf ] || exit 0
43c43
<       daemon /usr/local/samba/bin/smbd $SMBDOPTIONS
---
>       daemon smbd $SMBDOPTIONS
48c48
<       daemon /usr/local/samba/bin/nmbd $NMBDOPTIONS
---
>       daemon nmbd $NMBDOPTIONS
```

Customization - The smb.conf file

- All Samba configuration is done in the smb.conf file
  - Modeled after Microsoft ".ini" files
  - Shipped in /etc/ (SuSE) or /etc/samba (Red Hat)
  - Most Samba executables read this file first
- Comprised of sections, parameters and values

```
[section]
        parameter = value
```

  - Sections named `[global]`, `[homes]` and `[printers]` are reserved
- Almost 300 parameters can be set !!! - some of the more common:
  - The security parameter determines how the user is authenticated:

```
security = share   password is on the share - deprecated by MS
security = user    UNIX authentication - the default value
security = server  To offload authentication to another server
security = domain  To join a NT/2000/XP domain
```

    - netbios name is the name of the Samba server in the browse list. For example:

```
netbios name = PBC99215
```

    - socket options allows you to set TCP/IP options when talking with the client. For example:

```
socket options = TCP_NODELAY IPTOS_LOWDELAY SO_SNDBUF=8192 SO_RCVBUF=8192
```

    - log level allows the debug level to be specified. This is useful when initially testing Samba:

```
log level = 4
```

    - interfaces normally does not need to be set, but s390 point to point networking has the wror subnet mask (has 255.255.255.255 but should be 255.255.255.0)

```
interfaces = 9.12.6.73/24
```


Customization - The smb.conf file (cont'd)

Adding a file "share" is very easy

- Name of a new section in the smb.conf file becomes the name of the share -
- Only parameters necessary are share name and path:

```
[smbdocs]
        path = /usr/local/src/samba/docs/htmldocs
        read only = no
```

Adding printers is very easy

- If you have a [printers] section, all printers in /etc/printcap will be available

```
[printers]
        path = /var/spool/lpd
        guest ok = Yes
        printable = Yes
        browseable = No
```

Customization - Setting up SWAT

- Verify swat is in /etc/services:

```
# grep swat /etc/services
swat            901/tcp         # XXX Samba Web Adminisration Tool
```

- Enable swat on a inetd-based system (SuSE)

```
# cp inetd.conf inetd.conf.orig
# vi inetd.conf ...
# diff inetd.conf inet.conf.orig
< swat stream tcp  nowait.400 root  /usr/local/samba/bin/swat swat
---
> # swat stream tcp  nowait.400 root  /usr/sbin/swat  swat
```

- Enable swat on an xinetd-based system (Red Hat)

```
# cat /etc/xinetd.d/swat
# description: swat is the Samba Web Administration Tool
service swat
{
        disable                 = no
        socket_type             = stream
        wait                    = no
        user                    = root
        server                  = /usr/local/samba/bin/swat
}
```

- Restart inetd or xinetd

```
# /etc/init.d/inetd restart
```


Customization - Using SWAT

- Restart inetd (or xinetd):

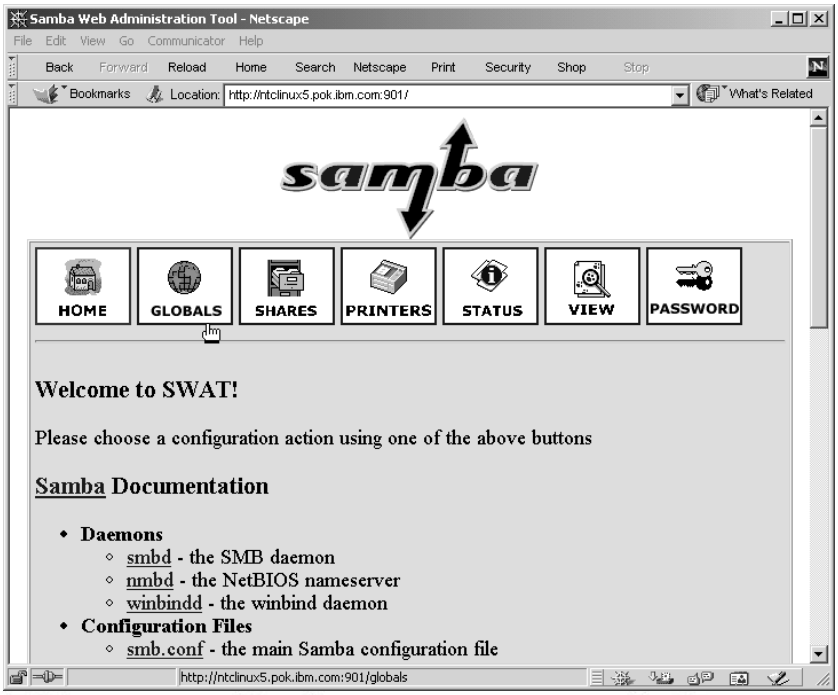
```
# /etc/init.d/inetd restart
Shutting down inetd                                        done
Starting inetd                                             done
```

- Go to the URL

```
http://your.server:901
```

- Pros:
  - GUI front end makes administration easier (arguable)
  - Comments and default values are removed from smb.conf so it is easier to read
  - Documentation is readily available, especially, fast access to a description of each parameter
- Cons:
  - Comments are removed from smb.conf which may be important information
  - Security - another port is open and the communications are not encrypted
  - Sometimes a clunky user interface

Customization - SWAT screenshot



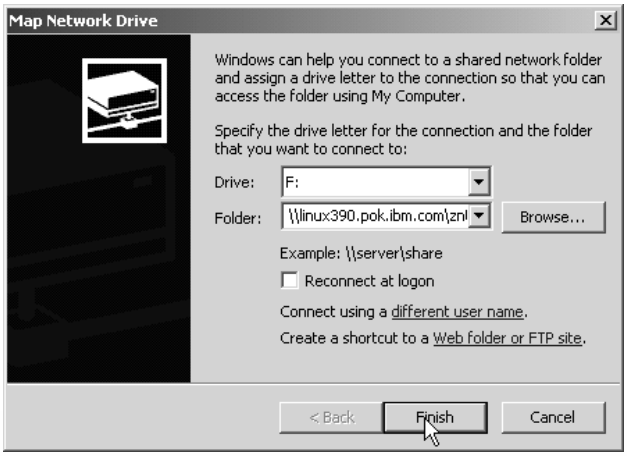
Customizing - Getting a file share from Windows

You can get a share via:

- Browse lists
  - Not recommended because master browser becomes a point of failure
- Mapping a drive
  - From DOS

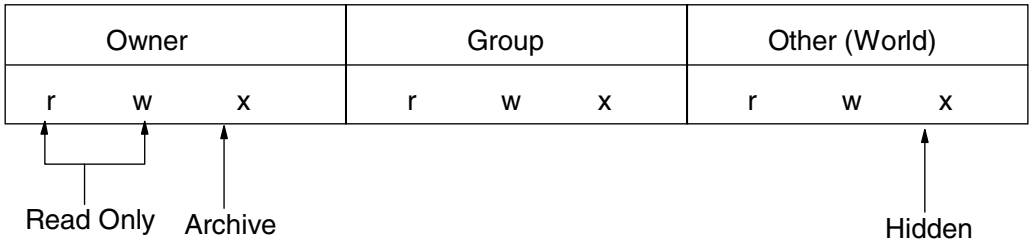
```
C:> net use * \\linux390.pok.ibm.com\zntc
```

  - From Explorer:



Customization - Permissions

UNIX permissions and DOS (FAT file system) attributes



- smb.conf parameters:
  - map archive - **map archive** attribute to owner execute bit (default = Yes)
  - map system - **map system** attribute to group execute bit (default = No)
  - map hidden - **map hidden** attribute to other execute bit (default = No)
- There is no parameter for the Windows **read only** attribute because it is always mapped
- For a complete DOS attribute mapping, add:

```
map system = Yes
map hidden = Yes
```


Customization - Access Control Lists (ACLs)

- Standard UNIX permissions do not allow for ACLs, but NTFS does
  - The "nt acl support" parameter tells smbd to attempt to map UNIX permissions into Windows NT access control lists.
- There is a project to bring ACLs to Linux - see:

```
http://acl.bestbits.at/
```

- To use ACLs you have to rebuild the 2.4.17 kernel:

```
# cd /usr/src/acl
# ls
acl-2.0.11-0.src.rpm           linux-2.4.17acl-0.8.26.diff.gz
attr-2.0.8-0.src.rpm           linux-2.4.17ea-0.8.26.diff.gz
e2fsprogs-1.27ea-26.4.src.rpm
```

- Then you have to build the ACL tools:

```
# rpm --rebuild attr-2.0.8-0.src.rpm
# rpm -Uvh /usr/src/packages/RPMS/s390/*.rpm  // On SuSE
# rpm -Uvh /usr/src/redhat/RPMS/s390/*.rpm    // on Red Hat
# rpm --rebuild acl-2.0.11-0.src.rpm
# rpm -Uvh /usr/src/packages/RPMS/s390/*acl*.rpm   // on SuSE
# rpm --rebuild e2fsprogs-1.27ea-26.4.src.rpm
# rpm -Uvh /usr/src/packages/RPMS/s390/e2fs*.rpm  // On SuSE
```

- When built you will have the **setfacl** and **getfacl** commands

Customization - Authenticating users

- Authentication can be done in one of the following ways:
  - (1) On Linux via non-encrypted passwords
    - /etc/passwd file must be maintained
    - Uses the parameter:

```
security = user
```

  - (2) On Linux via encrypted passwords
    - /etc/passwd and the smbpasswd file must be maintained
    - Uses the parameters:

```
security = user
encrypt passwords = yes
smb password file = <location of smbpasswd file>
```

  - (3) On a Windows Primary Domain Controller (PDC)
    - /etc/passwd must still be maintained
    - Uses the parameters:

```
security = domain
password server = <NetBIOS name of PDC>
```

  - (4) On a Windows PDC with users added automatically
    - /etc/passwd is "self-maintained"
    - Uses the parameters:

```
security = domain
password server = <NetBIOS name of PDC>
add user script = useradd -d /dev/null -s /bin/false %u
```


Customization - Authenticating users (cont'd)

- (5) Using Samba as a PDC
  - Uses the parameters:

```
os level = 64
preferred master = yes
domain master = yes
local master = yes
security = user
encrypt passwords = yes
domain logons = yes
logon path = \\%N\profiles\%u
logon drive = H:
logon home = \\homeserver\%u
logon script = logon.cmd
```

- (6) Allow Windows NT domain users to appear and operate as UNIX users
  - The new winbind daemon must be set up and running
  - Uses the parameters:

```
workgroup = <NT DOMAIN NAME>
security = DOMAIN
password server = <Windows PDC IP@ or DNS name>
winbind uid = 10000-20000
winbind gid = 10000-20000
winbind separator = +
```

Customization - Using winbind

- In a "Windows shop", additional users and passwords on Linux are a burden, or simply not permitted
- With winbind authenication is passed to a Domain Controller
- Overall steps for using winbind:
  1. Get the **`winbindd`** executable (described)
  2. Modify the smb.conf file for winbind (described)
  3. Set the lmhosts file
  4. Set the name service switch to use winbind
  5. Create a trust account for each machine to access the domain
  6. Join the Windows NT/2000 domain
  7. Get the **`winbindd`** executable

- Set the lmhosts file

```
# cd /usr/local/samba/lib
# vi lmhosts
...
# cat lmhosts
127.0.0.1   localhost
9.117.73.31 LCCWIN2K
9.117.73.31 POKLCC
```


Customization - Using winbind (cont'd)

- Set the name service switch to use winbind

```
# ls /lib/libnss_winbind*
ls: /lib/libnss_winbind*: No such file or directory
# cd /usr/src/samba/samba/source/nsswitch/
# cp libnss_winbind.so /lib/libnss_winbind.so.2
```

- Modify the file /etc/nsswitch.conf to access winbind:

```
# cd /etc
# cp nsswitch.conf nsswitch.conf.orig
# vi nsswitch.conf
...
```

- For SuSE:

```
# diff nsswitch.conf nsswitch.conf.orig
31,32c31,32
< passwd: files winbind
< group:  files winbind
---
> passwd: compat
> group:  compat
```

- For Red Hat:

```
# diff nsswitch.conf nsswitch.conf.orig    // for Red Hat
33,34c33,34
< passwd:     files winbind
< shadow:     files winbind
---
> passwd:     files nisplus
> shadow:     files nisplus
```

Customization - Using winbind (cont'd)

- Create a trust account for each machine to access the domain
  - Typically this is done on the Windows NT/2000/XP via the **Active Directory Users and Computers** interface.
  - **Action -> New -> Computer** menu choice should be invoked.
  - NetBIOS name of the Samba server is used

- Join the Windows NT/2000 domain

```
# smbpasswd -j poklcc -r lccwin2k
2002/03/21 11:42:57 : change_trust_account_password: Changed
 password for domain POKLCC.
```

- Start the winbindd executable

```
# winbindd
```

  - Now check that you have a good secret.

```
# wbinfo -t
Secret is good
```

- Now authentication is done by the Windows Domain Controller

Customization - Sharing files read/write in teams

You may want to share files R/W by teams with some users on multiple teams

- UNIX groups work fine with this

Customization is required on the Linux and Samba side

- Linux side
  - Groups have to be added

```
# groupadd team1
# groupadd team2
# grep team /etc/group
team1:x:501:
team2:x:502:
```

  - Users have to be added

```
# useradd -G team1 user1
# useradd -G team2 user2
# useradd -G team1,team2 user3
# grep team /etc/group
team1:x:501:user1,user3
team2:x:502:user2,user3
```

  - New files should have the group write bit set (considered a security exposure by Red Hat)

```
# grep umask /etc/profile
umask 002
```

Customization - Sharing files in teams (cont'd)

- Linux side (cont'd)
  - Create directories owned by group and with group write, "setgid" bits set

```
# mkdir /project1 /project2
# chgrp team1 /project1
# chmod g+ws /project1
# chgrp team1 /project2
# chmod g+ws /project2
```

- Samba side
  - Add shares which propogate group write bits

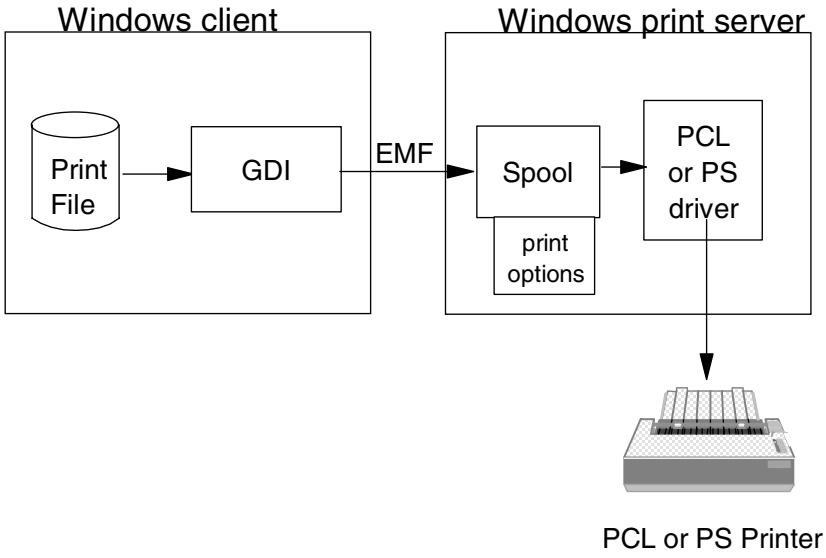
```
[project1]
        path = /project1
        force group = +team1
        read only = No
        create mask = 0664
        directory mask = 0775
        force directory mode = 02775

[project2]
        path = /project2
        force group = +team2
        read only = No
        create mask = 0664
        directory mask = 0775
        force directory mode = 02775
```


Customization - Printing

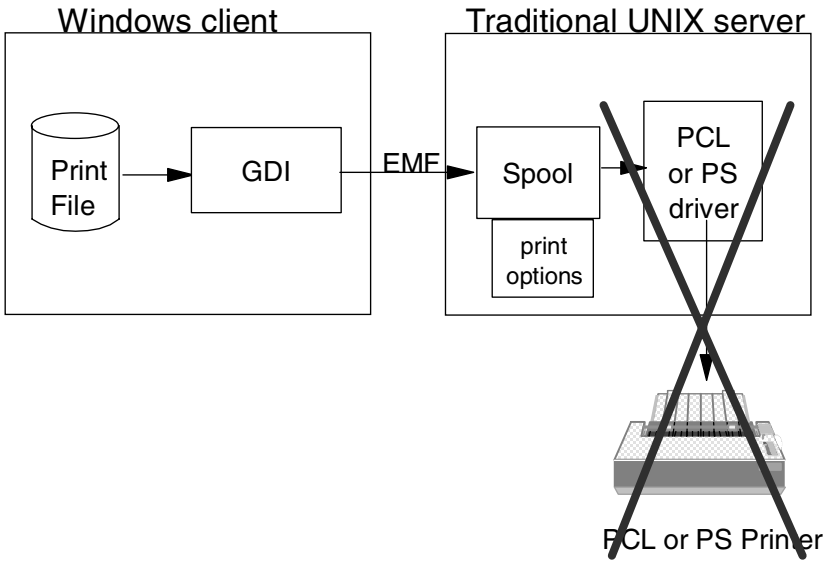
- Samba is not a true print server, just a "middle-man"
  - lpd is a print server
  - LPRng (lpr next generation) is now available
  - CUPS (Common UNIX Printing System) is an IPP print server
    - Internet Printing Protocol (IPP) is a new IETF standard
    - Printers are now being made IPP-ready
- There are still no drivers for S/390 channel-attached printers
- Samba offers the automatic downloading of printer drivers
  - From a paper on CUPS:

"It took until (Samba) 2.2.4 to get this feature fairly stable and make it work for a lot of different environments."

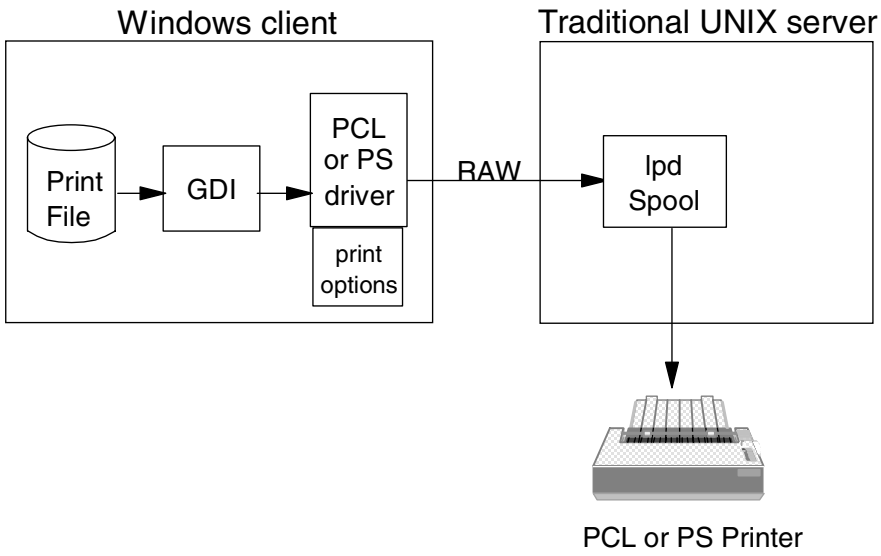
Customization - Printing (cont'd)



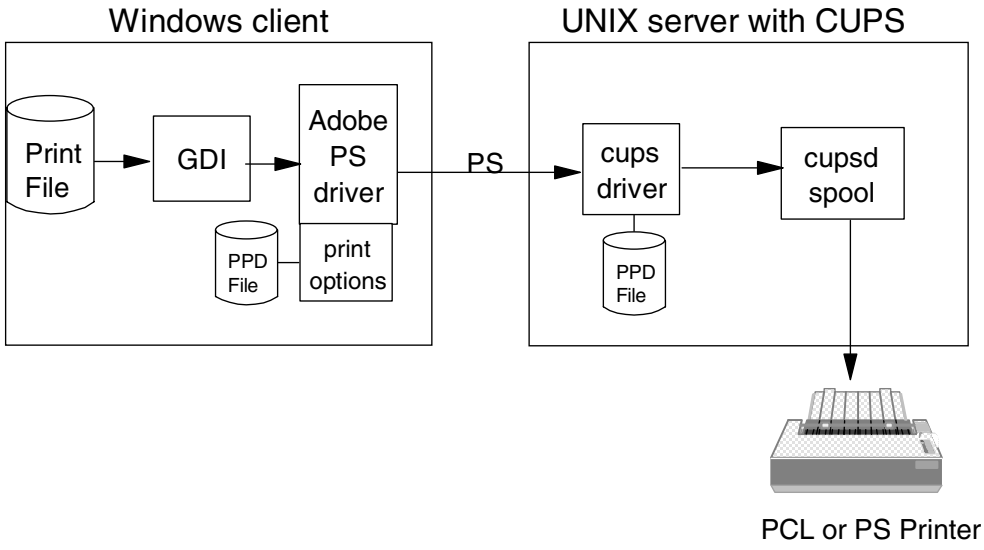
Customization - Printing (cont'd)



Customization - Printing (cont'd)



Customization - Printing (cont'd)



Customization - Printing (cont'd)

- smb.conf parameters when using CUPS

```
[global]
    load printers = yes
    printing = cups
    printcap name = cups
[printers]
    comment = All Printers
    path = /var/spool/samba
    browseable = no
    public = yes
    guest ok = yes
    writable = no
    printable = yes
    printer admin = root
[print$]
    comment = Printer Drivers
    path = /etc/samba/drivers
    browseable = yes
    guest ok = no
    read only = yes
    write list = root
```


Customization - Security

- Normal SMB communications encrypts passwords but not data
- Samba can be compiled with the **--with-ssl** configure option
- Data can be encrypted - however, Rule 1 has to be broken (Windows client has to be set up for encryption)
  - When security is involved, Rule 2 is invoked
- An SSL proxy machine can be used via the **stunnel** package

Customization - Performance and tuning

Disclaimer: I am not a performance expert, so this is "off the record"

- Tune the hardware
  - Ensure the network is not the bottleneck
  - Use volumes of 3390-3s rather than 3390-9s
  - Down the road ... Use SCSI disks instead of ECKD
- Tune Linux
  - If under VM, use VDISK for swap space
  - noatime arg in /etc/fstab (file access time is not updated on a read)
  - Try to utilize multiple channel paths (??)
- Tune Samba
  - **socket options = TCP_NODELAY IPTOS_LOWDELAY** - showed a small performance gain - however if you are accessing the share from a Wide Area Network (WAN), you should use IPTOS_THROUGHPUT instead of IPT_LOWDELAY
  - **socket options = TCP_NODELAY IPTOS_LOWDELAY SO_SNDBUF=14596 SO_RCVBUF=14596** - showed a slight performance gain
  - **socket options = TCP_NODELAY IPTOS_LOWDELAY IPTOS_THROUGHPUT SO_SNDBUF=14596 SO_RCVBUF=14596** - showed a performance degradation

Customization - Performance and tuning (cont'd)

Tune Samba (cont'd)

- **read raw = No** and **write raw = No** - showed a performance degradation
- **read size = 2048** - showed mixed results
- **max xmit = 8192** - showed a decent performance gain
- **oplocks = no** - showed an *extreme* performance degredation
- **write cache size = 262144** - showed mixed results

- Net recommendation:

```
max xmit = 8192
socket options = TCP_NODELAY IPTOS_LOWDELAY \
  SO_SNDBUF=14596 SO_RCVBUF=14596
```

- But - test for yourself

Customization - Samba as a time server

- Samba can act as a time server.
- First Linux must be a time client so it has an accurate time.
- Then it can serve this accurate time to Windows clients via Samba.
- Overall the steps are
  - (1) Set Linux hardware clock - however, **hwclock** is not enabled on s390
  - (2) Set Linux software clock
  - (3) Set Windows clients to use the Samba server
- Set Linux software clock
  - Make sure /etc/init.d/xntpd is enabled (SuSE)
  - Set xntp configuration file:

```
# cat /etc/ntp.conf
server    clock.llnl.gov
server    tock.usno.navy.mil
driftfile /etc/ntp.drift        # path for drift file
logfile   /var/log/ntp          # alternate log file
```

- Set Windows clients to use the Samba server
  - Create a set-time.bat file in the startup folder which has one line

```
net use \\<IP@ of Samba server> /set /yes
```

  - Folder is usually C:\Documents and Settings\<user ID>\Start Menu\Programs\Startup

References - The linux-390 list server

The linux-390 list server:

- Archives are on the Web at:
  http://www.marist.edu/htbin/wlvindex?linux-390
  - Subscribe and tailor by sending an e-mail to: **listserv@vm.marist.edu**
  - In the first line put: **sub linux-390 first_name last_name**
  - Follow and save the directions that follow
  - Some useful "commands":

```
set linux-390 mail    // 1 e-mail/message
set linux-390 digest  // 1 e-mail/day
set linux-390 nomail  // no e-mail
get linux-390 log0109 // get Sept 2001 archives
signoff linux-390     // to unsubscribe
```

  - To append to the list send an e-mail to linux-390@vm.marist.edu

References - Web sites

- Linuxvm.org - the Linux on zSeries portal:
  `http://linuxvm.org`
- DeveloperWorks - IBM Boeblingen
  `http://www10.software.ibm.com/developerworks/opensource/linux390/index.shtml`
- ISV applications for Linux on zSeries:
  `http://www-1.ibm.com/servers/eserver/zseries/solutions/s390da/linuxproduct.html`
- z/VM and Linux:
  `http://www.vm.ibm.com/linux`
- linux-390 archives:
  `http://www.marist.edu/htbin/wlvindex?linux-390`
- z/VM publications:
  `http://www.vm.ibm.com/pubs/`
- Redbooks
  - "Linux for S/390"
    `http://www.redbooks.ibm.com/abstracts/sg244987.html`
  - "Linux for zSeries and S/390: Distributions"
    `http://www.redbooks.ibm.com/abstracts/sg246264.html`
  - "Linux for zSeries and S/390: ISP/ASP Solutions"
    `http://www.redbooks.ibm.com/abstracts/sg246299.html`