

Samba on AIX

Skill Level: Intermediate

[Radhika A. Parameswaran \(radhika.p@in.ibm.com\)](mailto:radhika.p@in.ibm.com)

Senior software development engineer

IBM

25 Nov 2008

Samba is an open source, free software suite that provides seamless file and print services between Windows® clients and UNIX®-like platforms. It can be run on a UNIX-like platform like AIX®, BSD® UNIX, Linux®, IBM® 390 and OpenVMS. Samba uses the TCP/IP protocol that is installed in the host server. When configured, this server software allows the host machine to share files and printers and make them accessible from Windows clients.

Introduction

Samba is an open source free, software suite that provides seamless file and print services between Windows clients and UNIX-like platforms.

It can be run on a UNIX-like platform like AIX, BSD UNIX, Linux, IBM 390 and OpenVMS. Samba uses the TCP/IP protocol that is installed in the host server. When configured, this server software allows the host machine to share files and printers and make them accessible from Windows clients.

Important features available in the AIX versions of Samba are:

- Installation of binaries through SMIT
- SMB-based file and print services
- Share and user creation
- Configuration and maintenance through SWAT, a Web-based System Manager
- Trace and log capabilities

- Send-File API support
- Support for long AIX usernames and filenames
- Resource Browsing Protocol (Network neighborhood)
- Pass-through authentication
- Guest logon support
- Share and domain-level security support
- Unicode support
- Multiplexed SMB support (Windows Terminal support)
- NETBIOS-less connections
- SMB signing
- Active Directory support
- Directory change notification
- MSDFS support
- Support for AIX Classic and NFSv4 ACL
- Network logon support, including roaming user profiles
- Server-level security - Domain Controller capabilities
- Large file support
- DOS file attribute mapping
- Auto-disconnect
- Browse master support

Some features that are not provided in the AIX version of Samba are:

- Samba as LDAP server or client
- Samba as Active Directory server
- Kerberos server support
- DCE/DFS support
- DNS Updates support
- Automount / smbmount support
- PAM support

- NISPLUS support
- Cluster support

Hardware and software requirements

Samba runs on any machine that supports AIX with the 6100-02 Technology Level or later. The server must have the following minimal memory requirements:

- 30MB of RAM
- TCP/IP-supported LAN adapters connected to the network

Each client PC must have an installed LAN adapter; should be physically connected to a network; and have one of the Windows flavors installed on the client such as Windows 98, Server 2003, NT®, Vista®, and XP®.

Note that Samba is incompatible with Fast Connect. Hence, any existing version of Fast Connect has to be uninstalled.

Packaging and installation requirements

Samba on AIX requires the following packages:

Samba.base	Samba.base	Samba server
Samba.license S	amba.license	Samba licenses
Samba.man	Samba.man	Man pages for Samba

Configuration and administration

Configuring SWAT

In order to be able to connect to the Samba server, you must create users and shares. The configuration file that stores information of shares is `/usr/lib/smb.conf`. This file can be edited using a text editor in AIX or using the SWAT interface.

In order to configure SWAT, do the following:

1. Add the following line to `/etc/inetd.conf`:

```
swat    stream  tcp    nowait  root    /usr/sbin/swat    swat
```

2. Add the following line to /etc/services:

```
swat          910/tcp
```

3. Refresh inetd as follows:

```
$ refresh -s inetd
```

Now that SWAT is configured, it can be connected using a browser and the following URL:

```
http://samba_server.my.domain.:910
```

The SWAT page gives help on all configuration parameters. These man pages are also accessible from the AIX command-line through the man command.

Starting the Samba server

Samba has two daemons, nmbd and smbd, that need to be running in order for Samba to work correctly.

nmbd is a server that understands and can reply to NetBIOS over IP name service requests, like those produced by SMB/CIFS clients such as Windows 95/98/ME, Windows NT, Windows 2000, Windows XP, and LanManager clients. It also participates in the browsing protocols that make up the Windows Network Neighborhood view.

smbd is the server daemon that provides the file-sharing and printing services to Windows clients. The server provides filespace and printer services to clients using the SMB (or CIFS) protocol.

These can be started from the command line as follows:

```
$ nmbd
$ smbd
```

The logs of these daemons can be found in the /var directory as log.nmbd and log.smbd, respectively.

These daemons can also be started automatically during system reboot by adding an entry into /etc/inittab, as follows:

```
mkittab nmb:2:once:/usr/sbin/nmbd
mkittab smb:2:once:/usr/sbin/smbd
```

Now the two daemons get started during every reboot. The daemons get listed in the process lists:

```
# ps -ef | grep mbd
root 667870 708792 0 12:49:24 - 0:00 smbd
root 675974 1 0 12:49:24 - 0:00 nmbd
root 708792 1 0 12:49:24 - 0:00 smbd
```

Creating users

New users can be created using the Password menu in SWAT or using the `pdbedit` utility:

```
# pdbedit -a guest
new password:
retype new password:
```

The new user to be added is essentially an AIX user. The password of existing users can be changed using the `smbpasswd` utility:

```
$ smbpasswd -U guest
```

Creating shares

New shares can be created by editing the `smb.conf` file or using the Shares menu in SWAT. The following is an example share definition from `smb.conf`:

```
[samba]
    path = /samba
```

Testing smb.conf

The contents of `smb.conf` can be tested using the `testparm` utility. If errors are listed, they can be solved by editing `smb.conf` and then be retested with the following utility:

```
# testparm /usr/lib/smb.conf
Load smb config files from /usr/lib/smb.conf
Processing section "[tmp]"
Processing section "[public]"
Loaded services file OK.
Server role: ROLE_STANDALONE
```

```
Press enter to see a dump of your service definitions
```

```
[global]
    workgroup = USERS

[samba]
    path = /samba
```

Testing connectivity

Connectivity to the share can be listed in the AIX machine using the smbclient utility:

```
# smbclient -L //beas.in.ibm.com/samba
Password:
Anonymous login successful
Domain=[USERS] OS=[UNIX] Server=[Samba 3.0.25b]

    Sharename      Type            Comment
    -----      -
    tmp            Disk
    public         Disk
    IPC$           IPC             IPC Service (Samba 3.0.25b)
Anonymous login successful
Domain=[USERS] OS=[UNIX] Server=[Samba 3.0.24]

    Server          Comment
    -----
    BEAS            Samba 3.0.24

    Workgroup       Master
    -----
    USERS
```

The share can also be connected using the smbclient utility:

```
# smbclient //beas.in.ibm.com/samba -U root
Password: Domain=[BEAS] OS=[UNIX] Server=[Samba 3.0.24]
smb: \>
```

The current directory can be checked as follows:

```
smb: \> pwd
Current directory is \\beas.in.ibm.com\samba
```

Server shutdown

The Samba server can be shut down as follows:

```
smbcontrol smbd shutdown
smbcontrol nmbd shutdown
```

Disconnect open shares

In order to disconnect open shares from the server machine, run the following command:

```
smbcontrol <process id of smbd> close-share <sharename>
```

For example:

```
# smbcontrol 352494 close-share homes
# smbcontrol 352494 close-share guest
```

Note that if any files are open in the client, these commands have no effect.

Timeout client

To timeout a client, run the following command from the server machine:

```
# smbcontrol --timeout=100 smbd close-share tmp
```

After the timeout, the client is disconnected.

Alternatively, a timer can be mentioned in the smb.conf, as follows:

```
[global]
dead time = 1
```

Connect a client and keep the drive idle for a minute. The drive is disconnected after a minute.

Pinging client

To ping a client from the server machine, run the following:

```
smbcontrol <pid of smbd> close-share <sharename>
```

For example:

```
# smbcontrol 352494 ping
```

```
PONG from pid 352494
```

The response from client is a PONG message.

Trace and log capabilities

Samba offers logs for both daemons, nmbd and smbd. The nmbd logs are appended to /var/log.nmbd and the smbd logs are appended to /var/log.smbd.

Salient features and configuration

This section discusses some salient features in Samba and how to configure them.

User-level security

The authentication described in the previous section is user-level security. Here users are defined in the Samba server. The client authenticates through one of the users defined in the server. The global option in smb.conf is:

```
[global]
    security = user
```

This is the default option for the Samba server.

Share-level security

A standard internal user list is compiled in Samba consisting of users such as nobody, Administrator, and more. Access to a file or printer resource is based on successfully authenticating as any one of these standard users.

If a resource should be accessible only by a specific user, it is configured in smb.conf as follows:

```
[global]
    security = share

[tmp]
    only user = yes
    user = ann
    path = /tmp

[samba]
    path = /samba
```

In this configuration example, /tmp can be accessed only by *ann*, but /samba can be accessed by a guest user.

The `only user` option indicates whether Samba allows connections to a share using share-level security based solely on the individuals specified in the `username` option, instead of those users compiled on Samba's internal list (`nobody`, `Administrator`, etc.). The default value for this option is `no`.

Server-level security

Server-level security is a kind of security in which authentication can be passed to another server. This is also referred as pass-through authentication. The following is the configuration in `smb.conf`:

```
[global]
workgroup = IN.IBM.COM
encrypt passwords = yes
security = server
password server = jhelum.in.ibm.com

[share2]
comment = mnt
path = /home/mary/mnt2
read only = No
valid users = mary

[share1]
comment = mary
path = /home/mary/mnt1
writeable = yes
valid users = nobody
```

In this example, the user "mary" is allowed to access share2 and "nobody" is allowed to access share1.

Domain controller capabilities

The domain controller is the authenticator for all machines in that domain, both Windows and AIX. A domain usually pertains to the same subnet. Samba can act as a domain controller. The `smb.conf` should contain code as shown below:

```
[global]
workgroup = SAMBA
security = user
domain master = yes
local master = yes
preferred master = yes
os level = 65
domain logons = yes
add machine script = /usr/sbin/useradd -d /var/lib/nobody -g 100 %u
```

In Windows client settings, join the Samba domain. Add permissions from the Samba domain to the remote user list. These users do not necessarily exist in

Windows, but do exist in the Samba domain.

Samba is a domain controller. Windows is a member of the domain. When a domain is selected and the user name and password are input from the Windows client, the request for authentication goes to the Samba server. You can implement a pass-through to another Samba server, which is a domain controller, as follows:

```
[global]
    security = server
    password server = jhelum.in.ibm.com
```

Here jhelum is running the Samba server and is the domain controller. The authentication is passed to jhelum.

In case the domain controller is not known for adding in smb.conf, the Samba server can search for the same in the domain. This should be indicated as follows in smb.conf:

```
password server = *
```

Active Directory support

If an Active Directory (AD) server is installed in Windows, Samba can be configured to re-direct all incoming connections to the Active Directory server for authentication.

This can be achieved as follows.

Create user *rocky* in both AD as well as in Samba, with different passwords. Assume AD is installed in the Windows client 9.124.101.235. Configure smb.conf as follows:

```
[global]
    WORKGROUP = mygroup
    security = server
    password server = 9.124.101.235

[tmp]
    path = /tmp
    msdfs root = yes
    public = yes

[rocky]
    path = /home/rocky
    public = yes
```

Map as user *rocky* using the two different passwords (for Samba and AD). Both authenticate.

Multi-user logon

Multi-user logon capability allows multiple clients to access the shared resources through one connection with the server.

To test this capability, install Windows Terminal Server in a Windows machine. Reboot the machine. Map a drive to the Samba server. Connect to this Windows machine from another client through the Remote Desktop service.

You should be able to see the mapped drive and access it as usual without a password prompt. Just logging into that machine is enough.

There should be interactions only between the mapped client and the Samba server. The second client's IP does not appear anywhere, while still interacting with the Samba server.

Guest logon

Guest logon can be enabled by setting the following parameter in `smb.conf`:

```
[global]
security = user
    guest ok = yes
```

The guest account can be implemented for wrong passwords or wrong usernames. If the option is bad password, then the server logs in for all/nil password. If it is bad user, the server logs only for wrong usernames, not in Samba. This is shown as follows:

```
[global]
    map to guest = Bad user
```

The user that is assigned for guest logon is indicated by a parameter *guest account*.

```
[global]
    guest account = mary
```

Username mapping

Samba allows for the enforcement of specific users or permissions on a shared file or printer.

The following options can be used in the definition of a share:

```
    ##force permissions on files created on a share
create mask = 0777

    ##force permissions on directories created on a share
```

```
directory mask = 0444

## force user and group for files and directories created on a share
force user = root (username)
force group = system (groupname)
```

Connect a Windows client drive to the share with any other user (not root) that has read/write permissions and create a new file. Check the permissions of the file that is created in that share. You can see that the new file has the permissions of root and system.

To allow only specific users to read/write to a share, the following configuration should be used in smb.conf:

```
[tmp]
path = /tmp
writable = no
write list = root
read list = root
```

Password encryption

When password encryption is disabled in Samba, authentication is done by AIX. Samba authenticates only when password encryption is enabled.

Add the following in smb.conf:

```
[global]
password encryption = no
```

Create a user in AIX that does not exist in Samba. If you connect a drive in the Windows client, using this user you find that AIX authenticates the user. Now change the following in smb.conf:

```
[global]
password encryption = yes
```

Try to connect the above user. It should fail, stating that the user does not exist.

This shows that AIX authenticates the user, when the user is present in AIX and not in Samba when encryption is turned off. Samba authenticates when the password is encrypted, as in the second case above.

MSDFS feature

MSDFS allows multiple file servers to be seamlessly integrated into one logical namespace, which results in the following:

- A single drive-mapping can be used to access multiple file servers, possibly dispersed across the entire network.
- Multiple file servers can be mapped to the same name, thus providing redundancy and locality of data access.
- This complexity of logical and physical topology appears as a single directory tree (drive mapping), with sub-directories that may actually be located on remote servers.

MSDFS is organized as a topology of MSDFS root file shares, which can contain MSDFS links to other local or remote file shares. These MSDFS links appear as sub-directories, and so that transparent re-direction to the remote file shares occurs, as long as the user is properly authenticated at the remote servers. (Windows client software manages the MSDFS re-direct and remote-server authentication.)

To use this feature in Samba, the smb.conf file should mention the MSDFS root.

```
[global]
    host msdfs = yes
    security = user

[tmp]
    path = /tmp
    msdfs root = yes
    public = yes
```

Link to the DFS root as follows:

```
cd /tmp
ln -s msdfs:beas.in.ibm.com\\radhika t1
```

where `beas.in.ibm.com\\radhika` is a share and `beas` is also running Samba, and is compiled with the `--with-msdfs` option. If you map the `tmp` share from a Windows client, the directories of `t1` can be accessed. Thus, this feature allows directories of one share to be universally accessed from a DFS root.

ACL support

Samba supports both AIXC and NFSv4 ACL. The following are the configuration details for using NFSv4 ACL with Samba.

Create a new file system with `v2=extended` attribute `/samba`. Check the ACL types supported by the file system:

```
# aclgettypes /samba
Supported ACL types are:
    AIXC
    NFS4
```

Create a new file called test1.txt and check the ACL of that file:

```
# aclget test1.txt
*
* ACL_type    AIXC
*
attributes:
base permissions
  owner(root):  rw-
  group(system): r--
  others:       r--
extended permissions
  enabled
```

If extended permissions is disabled, enable it. Convert the ACL type of test1.txt to NFS4:

```
# aclconvert -t NFS4 test1.txt
```

Get the ACL on test1.txt:

```
# aclget test1.txt
*
* ACL_type    NFS4
*
*
* Owner: root
* Group: system
*
s:(OWNER@):    a      rwpRwaAdcCs
s:(OWNER@):    d      xo
s:(GROUP@):    a      rRadcs
s:(GROUP@):    d      wpWxAco
s:(EVERYONE@): a      rRadcs
s:(EVERYONE@): d      wpWxAco

Include a new share for the created filesystem in smb.conf:
[samba]
  path = /samba
  public = yes
```

There is no predictable mapping between UNIX mapping and NFS4 ACL. Hence, it can be misleading if you compare NFSv4 ACL and permissions in the `ls -l` listing.

Map a drive from the Windows client using the user "john" (who belongs to the system group). Create a new file in /samba. The server does not allow deletion of

that file. Access is denied for delete, as D is denied for that group.

SMB signing

SMB signing provides mutual authentication and message authentication capabilities for the Samba server. By default, SMB signing is disabled. If enabled, each message is also validated with a digital signature.

The following should be added to smb.conf to enable SMB signing:

```
server signing = yes
```

Enable SMB signing in the Windows client. Thereafter, the SMB header contains the comment "Security signatures are enabled."

Large file support

Large files are those whose sizes exceed 4G. The Samba server supports large files by default. It does not require any specific parameter to be set. To test this feature, transfer a large file from AIX to the Windows client through a mapped drive. It will be successful.

Print services

Printer resources in AIX can be shared across and accessed from Windows clients.

Consider the following print queues configured in AIX.

```
# lpstat
Queue  Dev  Status  Job Files  User  PP %  Blks  Cp Rnk
-----
badq   lxx   READY
52ps   hp@9  READY
52c1   hp@9  READY
52vc1  lxx   READY
```

The printer share can be added in smb.conf as follows:

```
# cat /usr/lib/smb.conf
[global]
    print command = /usr/bin/lpr -r -P%p %s
    lpq command = cat %p >> /tmp/lpq.log ;/usr/bin/lpq -P%p
    lprm command = /usr/bin/lprm -P%p %j

[52vc1]
    printable = yes
    use client driver = yes
    browseable = yes
    print command = /usr/bin/lp -d 52vc1 %s
    path = /tmp
    printing = AIX
```

```

guest ok = yes
print ok = Yes
    
```

You can see the configured printer in the Windows network neighborhood in the same subnet. Alternatively, in the other subnets, add the networked printer as a new printer.

You can test print using the SMB client as follows:

```

# smbclient //aixfvt49.in.ibm.com/52vcl -U joe
Password:
Anonymous login successful
Domain=[SAMBA] OS=[UNIX] Server=[Samba 3.0.26a]
smb: \> put /etc/motd
putting file /etc/motd as \etc/motd (6.5 kb/s) (average 6.5 kb/s)
smb: \>
    
```

You can send requests from Windows to this networked printer.

DOS attribute mapping

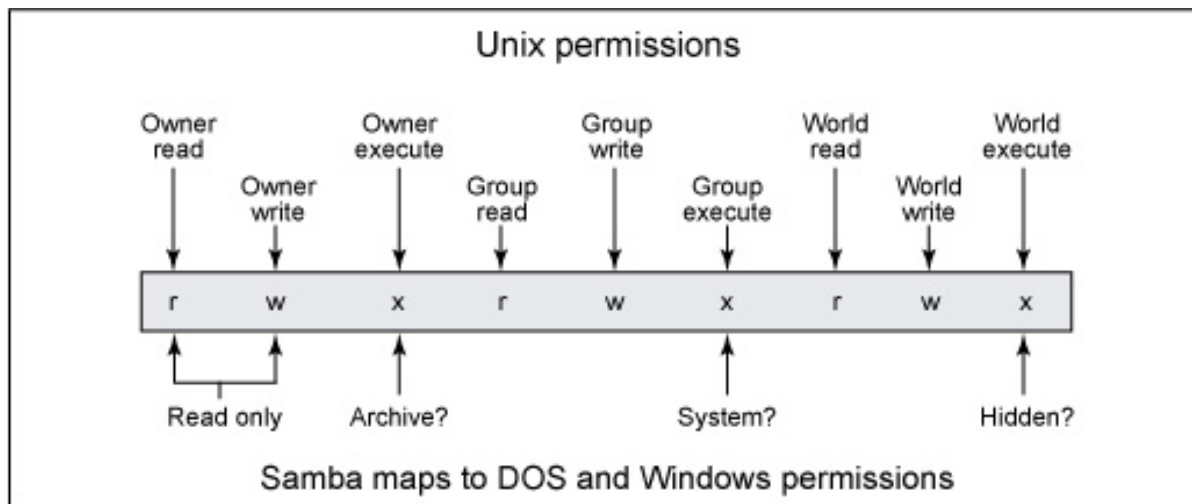
Samba supports DOS file attributes with the following options:

```

map archive = yes
map system = yes
map hidden = yes
    
```

The following figure shows mapping of permissions with DOS attributes.

Permissions with DOS attributes



You can create a file in the shared directory with some permission bits. You can

view the same using the SMB client tool as follows:

From AIX:

```
-rwxrwxrwx  1 john      staff      0 Jan 29 14:25 New Text Document (4).txt
-rwxrwxrwx  1 guest     usr        0 Feb 27 15:23 New Text Document (5).txt
drwxrwxrwx  2 ldapdb2  dbsysadm  256 Feb 08 10:46 SQLDIR.LK0
```

From smbclient:

```
New Text Document (4).txt      AHS      0 Tue Jan 29 14:25:56 2008
New Text Document (5).txt      AHS      0 Wed Feb 27 15:23:34 2008
SQLDIR.LK0                      D        0 Fri Feb 8 10:46:05 2008
```

Directory change notification

The directory change notification feature is implemented by default in Samba. Samba generates an "NT NOTIFY" message when a change to a directory is made from the client, such as creating a new file.

In case the change interval has to be changed, edit the smb.conf as follows:

```
change notify timeout = 10
```

The iptrace shows NT NOTIFY requests and responses.

NETBIOS-less connections

Disable NETBIOS over TCP/IP in a Windows client and reboot. Now connect a drive. The connection is successful, indicating that NETBIOS-less connections are possible in Samba.

Resource browsing protocol

Samba is visible in the network neighborhood of Windows clients in the same subnet.

This can be tested as follows. A Windows client machine in the same subnet as the Samba server is required. Assume that the Samba server runs in *jhelum* and 9.124.113.100 is a Windows client in the same subnet.

Edit the smb.conf as follows:

```
[global]
  workgroup = WORKGROUP
  security = user
[samba4]
```

```
path = /samba4
writeable = yes
valid users = root
```

Now browse the network neighborhood in the Windows client. You can see the Samba server in the network neighborhood of the client.

Browse master functionality

Network browsing is a concept that enables Windows and Samba servers to appear in the Windows network neighborhood. Inside the network neighborhood, icons are represented as servers and if opened, the server's shares and printers that are available are displayed.

A domain master browser collates the browse lists from the local master browsers on all subnets so that browsing can occur between workgroups and subnets. Also, the domain master browser should preferably be the local master browser for its own subnet.

Samba can act as browse master with the following setting:

```
[global]
local master = yes
preferred master = yes
WORKGROUP = SAMBA
```

The interval in seconds for which nmbd can wait before repeatedly broadcasting LAN Manager announcements can be set as follows in smb.conf:

```
lm interval = 120
```

Now test the connectivity from a Windows client.

Test the browse master functionality using the SMB client tool as follows:

```
# smbclient -L //aixfvt21.in.ibm.com/tmp
Password:
Anonymous login successful
Domain=[SAMBA] OS=[UNIX] Server=[Samba 3.0.26a]
  Sharename      Type            Comment
  -----
  tmp             Disk
  samba          Disk
  IPC$           IPC             IPC Service (Samba 3.0.26a)
Anonymous login successful
Domain=[SAMBA] OS=[UNIX] Server=[Samba 3.0.26a]
  Server          Comment
  -----
  AIXFVT21        Samba 3.0.26a
```

```
Workgroup      Master
-----
SAMBA AIXFVT21
```

The following is the relevant excerpt from log.nmbd:

```
[2008/03/27 13:22:10, 0] nmbd/nmbd_become_lmb.c:become_local_master_stage2(396)
*****
```

Samba name server AIXFVT21 is now a local master browser for the workgroup SAMBA on subnet 9.124.101.199

Note that `local master = yes` guarantees that Samba will participate in the election and that's all it does. `preferred master = yes` forces browse election when Samba first comes on-line.

Problem determination

Tracing

The message packets between client and server can be traced using the `tcpdump` command in AIX.

Start the `tcpdump` command in the server as follows:

```
tcpdump -s 0 -w <tracefile> host <hostname> and <hostname/ipaddress>
```

For example:

```
tcpdump -s 0 -w cap_1.cap host aixfvt21 and 9.126.241.144
```

The transaction between the client and server should be completed and then the `tcpdump` can be killed. The trace file created can be viewed using Ethereal or any tool.

Logs

`/var/log.smbd` records messages from the SMB daemon and `/var/log.nmbd` records messages from the nmb daemon.

The log levels can be increased by the following in `smb.conf`:

```
[global]
log_level = 5
```

The logs can be looked at for any error messages, whenever it is required.

Conclusion

This article showed how to set up and configure the Samba server. You saw how shares can be defined and accessed from Windows. Different authentication mechanisms are also discussed.

Resources

Learn

- [Make UNIX work with Windows XP and Mac OS X](#) (developerWorks, April 2006) discusses the free UNIX implementation of SMB or CIFS, called Samba.
- (developerWorks, February 2006) looks at the at the development of
- [Improve your memory programming](#) (developerWorks, May 2007) helps you conquer these pesky memory defects.
- [The importance of UNIX in SOA environments](#) (developerWorks, August 2008): Discover how and why existing systems and applications with which you are already familiar deployed on operating systems that you know well are so critical to the present and future of Web-based computing, particularly in the area of SOA.
- [POWER5 virtualization: How to set up the IBM Virtual I/O Server](#) (developerWorks, June 2005): Get more information on the virtualization capabilities of the IBM POWER5 servers.
- [System Administration Toolkit: Monitoring disk space and usage](#) (developerWorks, June 2006): Look at methods for determining disk usage across multiple UNIX systems and how to create a simple warning system to alert you of potential problems.
- [The AIX and UNIX developerWorks zone](#) provides a wealth of information relating to all aspects of IBM® AIX® systems administration and expanding your UNIX skills.
- [New to AIX and UNIX?](#) Visit the New to AIX and UNIX page to learn more.
- [developerWorks technical events and webcasts](#): Stay current with developerWorks technical events and webcasts.
- [Podcasts](#): Tune in and catch up with IBM technical experts.

Get products and technologies

- [IBM trial software](#): Build your next development project with software for download directly from developerWorks.

Discuss

- Participate in the AIX and UNIX forums:
 - [AIX Forum](#)
 - [AIX Forum for developers](#)
 - [Cluster Systems Management](#)

- [IBM Support Assistant Forum](#)
- [Performance Tools Forum](#)
- [Virtualization Forum](#)
- [More AIX and UNIX Forums](#)

About the author

Radhika A. Parameswaran

Radhika Parameswaran is a postgraduate from BITS Pilani, where she was also teaching the graduate and post graduate students. She joined IBM in 2004 and works on AIX operating system technologies. Her present engagement is on SMB file system-related products on AIX. You can reach her at radhika.p@in.ibm.com

Trademarks

IBM, AIX, and POWER6 are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.