A Next Generation CIFS Client

Steven French
Linux Technology Center
IBM Austin

Connectathon 2002
March 5, 2002

email:sfrench@us.ibm.com

Outline

- Value of the client perspective
- Some current clients
- CIFS Client (VFS) for Linux Status
- Name Resolution
- Reliability
- Compatibility
- Performance
- Security
- Areas to extend the dialect

Client Perspective

Value of the client

- Client drives protocol flow so can determine ...
- What is the smallest set of SMB PDUs that could be used by a client without loss of function or performance?
  - How different is the answer for Windows 2000 vs. Linux?
- And a well written, easily available, modifiable reference client would promote interoperability.
  - smbfs? JCIFS?

Value of the client

- A consensus is emerging that small enhancements are needed to keep CIFS ahead of NFS and WebDAV in intranet file sharing
- Many of these changes could be invaluable for a smart CIFS client
- ... but there is constant tension (especially for clients) between perfect reliability/interoperability and enhanced features/performance

Client

- Most attention has been on server
  - Example: Samba vs. smbfs
    - » Just smbd (main server daemon) alone is more than 20KLOC vs
    - » VFS filesystem Smbfs (less than 4KLOC) and all related utilities for mount and session establishment (client and library functions less than 14 KLOC)
  - Server performance optimization is big foucs
    - » benchmarks often show performance of server with client constant (rather than reverse)

Current Clients

Current Clients

- Filesystems (IFS or VFS like)
  - Windows 2000 and XP
    - » Viewed by many as the reference client
  - Windows 9x, Windows ME
  - Thursby Macintosh Clients
  - BSD smbfs
  - Apple
  - Sharity
  - Linux Smbfs
  - OS/400

Current Clients - other types

- Client SMB Utilities such as
  - Smbclient - ftp like
  - Various utilities built on smblib
- Java Clients such as
  - JCIFS

Some URLs for more info

- Smbclient http://www.samba.org
- JCIFS http://jcifs.samba.org
- Another JCIFS: www.hranitzky.purespace.de/jcifs/jcifs.htm
- jSMB http://members.nbci.com/harikris_v
- Sharity light: http://www.obdev.at/products/sharity-light
- Linux Smbfs:http://www.kernel.org/pub/linux/kernel/2.4
- http://cvsweb.netbsd.org/bsdweb.cgi/syssrc/sys/smbfs/

Status

CIFS Status

- Working:
  - Mount in kernel at NTLM level
  - Unicode or ASCII
  - Most Current level of SMBs following CIFS T/R
  - Open, Read, Write, Release (close)
  - Get and Set Attributes
  - Readdir (for small to medium sized directories)
  - Passes 5 of first 7 tests attempted
- Not implemented:
  - Chmod (777 returned), Native CIFS ACL/SecurityDescriptors
  - Locking
  - Distrubuted caching/oplock, MMAP
  - Kerberos/SPNEGO Authentication

Problem areas

- Mount does not pass UNC name:
  - Mount //server/share /mnt –o user=name,pass=
- Hard to detect mandatory vs. advisory locks
- Invalidating mmap pages
- AccessFlags
- Setting Security Descriptors
- Group field of 777 permissions
- Add security code and DNS resolution helper code in kernel
- Efficiency of large scale directory notification
- O_ACCESS flag maps poorly
- Many missing or hard to set flags (such as sparse file, access pattern hints)
- AD Integration, Winbind/NSS integration

Name Resolution

Key name resolutions questions

- Skipping NBT (RFC 1001 name service) - "pure TCP" on port 139. Is always ok?
- In pure TCP - what should go in UNC path in tConX? IP address works but what about multiple servers on one port? *SMBSERVER does not work for Win2K
- Does ipv6 address work in tConx path?
- Are there cases in which Unicode translation should be disabled even though both server and client support it? (same codepage on each - how can the client detect the server's code page)

More name resolution ...

- Is there a recommend order for trying address resolution? Big performance hit
  - Probably not - many Unix/Linux clients may be satisfied with pure DNS resolution
- Need for DFS support in more clients is a given but ...
  - How do you locate the global root?
    - » DNS lookup of reserved domain specific name
    - » LDAP query (Win2K seems to store it in both)
    - » UDDI query
    - » Local config file (worst case for most)

Reliability

Reliability points to consider

- Three big issues:
  - Logon reliability, redundancy
  - Redundancy of DFS root
  - Redundancy of (especially r/o) resources using DFS replicas
  - Reconnection after transient TCP session failure without user intervention - transparent to an application?
  - What if TGT has expired? Can KDE or GNOME prompt the user to reenter the password or is a local service necessary?

Reliability points to consider

- It is a given that clients should leverage DFS when available for reconnection -
  - But how long can we keep info on replicas safely without asking for a referral again?
  - What order should clients try to connect to replicas? Round robin in same subnet 1st?
  - And the mgmt API for DFS (AddRoot, etc.) is only partially implemented ...

Reliability points to consider

- And how do we handle out of disk space on write? If storage is replicated can client initiate failover to replica? Should server move directory, then drop session then generate referral on reconnection?
- Should we add a new return code on write "ERR_FILE_MOVED" so servers can auto-failover to another server that is less full?

Compatibility

VFS Operations

- RAMFS in Linux 2.4 is a good example of a minimal, cleanly written filesystem
- Network filesystems unfortunately are much more complicated for Linux see include/linux/fs.h to see a complete list

VFS Operations (file ops)

- Llseek
- Read
- Write
- Readdir
- Poll
- Ioctl
- Mmap
- Open
- Flush
- Release
- Fsync
- Fasync

VFS Operations (file ops part 2)

- Lock
- Readv
- Writev
- Sendpage
- Get_unmapped_area

VFS Operations (part 3) inode ops

- Create
- Lookup
- Link
- Unlink
- Symlink
- Mkdir
- Rmdir
- Mknod
- Rename
- readlink

VFS Operations (part 4) inode ops

- Follow_link
- Truncate
- Permission
- Revalidate
- Setattr
- Getattr

- And superblock operations
  - Statfs
  - Unlockfs
  - Write_super
  - Inode_from_fh

Key Data Structures

- Superblock matches ok (connected to CIFS servers)
- Inode is easy match except for 777 permissions (vs. SecurityDescriptors) and to lesser extend FileAttributes
- File structure is ok match but missing AccessFlags
- Dentry structure is mostly transparent
- Lock structure is reasonable match (behavior differences though) – mandatory vs. advisory

Error codes

- Lists of error codes by SMB PDU are nowhere near complete
- Error mapping, especially from (large list of) NT STATUS codes to (smaller) errno.h Posix errors is adhoc
- Out of band alerts, management API, CIFS client MIB etc. is possibility for future documentation and/or standardization

Subtle OS specific features

- Underdocumented and less understood features that affect the client are:
  - Reparse points (e.g. copy on write like links used by SIS on Win2K among others)
  - Symbolic links – Unix extensions can be used
  - Sparse file attribute bit
  - Compressed and encrypted file attribute bits
  - Extended attributes and streams (tend to be application/desktop specific). Stream mechanism itself is understood though.
  - Everchanging new ioctl and fsctl call and new Transact2 Info levels …

Performance

Performance points to consider for client

- For common cases in (at least) a typical intranet - let's show (and fix if need)
  - Maximize parallelism from each client
  - Maximize client caching opportunities
  - Minimize roundtrips from clt to srv
    - » Command chaining - does it need improvements ala NFS v4?
  - Minimize protocol overhead (frame hdrs)
  - Latency when lightly loaded (timers ...)
  - Session establishment and auth overhead

Performance points - what is missing?

- Change open type - rerequest oplock (on long opened file) without close

(Breaking news … ioctl discovered that seems to be for this purpose)

- Do we need a new token/oplock manager to better handle replicas?
- Secondary session (maxvs > 1) - rawVC or new RDMA secondary connection
- Client timing/observations of server - autotuning read-ahead and oplock/lazy close behavior

Performance points - what is missing?

- Lots of distinct infrequently used long paths are big problem for Linux
- ... but could be tough to skip due to referrals
- Vnode invalidation tricky on client due to kernel caching
- Are access hints on file sufficient? Is something similar for directories needed?

Data integrity in presence of data cached by client kernel

- Invalidating Pinned mmaped pages are tricky invalidate_inode_pages2() may help but not demonstrated since not directly invoked by any filesystems in 2.4
- What having a distinct lock protocol help?
- How to handle directory entry caching?
  - One option is the notify SMBs
    - Performance issues unproven so ignored by most

Security

Security features

- Minimum
  - NTLM authentication
- But we should be offering
  - Kerberos authentication
  - NTLMv2 & packet signing
  - Per file (server decides) signing and encryption (ala NFSv4 proposal) – requires addition of new RC
- The CIFS and Kerberos protocol should add support for
  - Optional AES encryption (mostly a Kerberos issue)
  - Optional per-packet privacy or mode widespread support for SSL or dual transport SSL and TCP
  - Are changes needed for Hardware assist for session establishment

Security features

- For Linux filesysystems in particular a choice must be made early on –
  - Do all users get the same SMB UID when going to the server (relying on local ACLs to prevent users from modifying server data)?
  - Do you do a new SMB SessSetupX for each new user of the mount point?
- Complications – where do you get the password (especially on implicit dfs connections)? Desktop to FS interactions not strong suite of Unixes today
- And … what about password during reconnect? Store in cifs fs?

Some interesting security references

For more information - Security

- Kerberos and interoperable authentication
  - Project Pismere/MIT http://web.mit.edu/pismere/M
- MIT kerberos -
  - http://web.mit.edu/kerberos/www/
- Heimdal Kerberos implementation
  - http://www.pdc.kth.se/heimdal/L
- Luke Leighton's Book on DCE/RPC & SMB
  - **DCE/RPC over SMB: Samba and Windows NT Domain Internals**

For more information - Security

- Westerlund, A and Danielsson, J, "Heimdal and Windows 2000 Kerberos -- How to get them to play together", USENIX 2001
- Swift, M and Brezak, J, "The Windows 2000 RC4-HMAC Kerberos Encryption Type" Work in progress, draft-brezak-win2k-krb-rc4-hmac-02.txt

Options for the Future

Assistance Welcome