

Using the Samba Testsuite

Andrew Tridgell
Samba Team

tridge@samba.org

Basic Components

- The Samba4 testsuite consists of the following basic components:
 - smbtorure ad-hoc tests
 - smbtorure individual (RAW) tests
 - level scanners
 - smbclient
 - gentest
 - locktest
 - locktest2
 - masktest

How big is CIFS?

- The core file sharing part of CIFS currently consists of approximately 140 commands
 - nb: each trans2 information level is considered a separate command
- The Samba4 test suite currently tests approximately 116 of those commands.
- The gentest suite covers approximately 60 commands

Running smbtorure

- smbtorure is typically run like this:
 - `smbtorure //server/share -User%pass TESTNAME`
- You can list the available tests using `-h` for help
- Some tests require additional arguments, see `-h`
- Wildcards can be used in the TESTNAME
- For randomised tests, it is useful to specify the seed so the test can be repeated exactly

RAW tests

- The RAW tests in smbtoriture are the largest component of the new Samba4 testsuite
- run smbtoriture with a test argument of RAW-*
- takes advantage of internal protocol redundancy
- For each call every field is tested, where possible
 - RAW-QFSINFO RAW-QFILEINFO RAW-SFILEINFO
RAW-SFILEINFO-BUG RAW-SEARCH RAW-CLOSE
RAW-OPEN RAW-MKDIR RAW-OPLOCK RAW-
NOTIFY RAW-MUX RAW-IOCTL RAW-CHKPATH
RAW-UNLINK RAW-READ RAW-WRITE RAW-LOCK
RAW-CONTEXT RAW-RENAME RAW-SEEK

Running RAW-QFSINFO
Running level DSKATTR
Running level ALLOCATION
Running level VOLUME
Running level VOLUME_INFO
Running level SIZE_INFO
Running level DEVICE_INFO
Running level ATTRIBUTE_INFO
Running level UNIX_INFO
Running level VOLUME_INFORMATION
Running level SIZE_INFORMATION
Running level DEVICE_INFORMATION
Running level ATTRIBUTE_INFORMATION
Running level QUOTA_INFORMATION
Running level FULL_SIZE_INFORMATION
Running level OBJECTID_INFORMATION
check for correct aliases
volume_info.volume_name = 'test'
attribute_info.fs_type = 'NTFS'
check for consistent disk sizes
total disk = 4186 MB
check consistent free disk space
free disk = 4110 MB
volume info consistency
volume/volume_name.s [tes] != volume_info/volume_name.s [test] at torture/raw/qfsinfo.c(241)
check for non-zero unknown fields
check for correct termination
Expected wire_length 6 but got 8 for 'tes'
(285) incorrect string termination in volume/volume_name
TEST RAW-QFSINFO FAILED!
RAW-QFSINFO took 0.296887 secs

Level Scanners

- A level scanner is a program that tries every subcall and information level of a CIFS transaction request such as TRANS2
- smbtoriture includes 3 level scanners:
 - SCAN-ALIASES looks for aliases in known TRANS2 levels
 - SCAN-TRANS2 finds trans2 levels
 - SCAN-NTTRANS finds NTTRANS levels

QFSINFO scan

Checking for QFSINFO aliases

Found level 1 (0x001) of size 18 (0x12)
Found level 2 (0x002) of size 12 (0x0c)
Found level 258 (0x102) of size 26 (0x1a)
Found level 259 (0x103) of size 24 (0x18)
Found level 260 (0x104) of size 8 (0x08)
Found level 261 (0x105) of size 20 (0x14)
Found level 1001 (0x3e9) of size 26 (0x1a)
Found level 1003 (0x3eb) of size 24 (0x18)
Found level 1004 (0x3ec) of size 8 (0x08)
Found level 1005 (0x3ed) of size 20 (0x14)
Found level 1006 (0x3ee) of size 48 (0x30)
Found level 1007 (0x3ef) of size 32 (0x20)
Found level 1008 (0x3f0) of size 64 (0x40)

Found 13 levels with success status

Level 261 (0x105) and level 1005 (0x3ed) are possible aliases
Level 260 (0x104) and level 1004 (0x3ec) are possible aliases
Level 259 (0x103) and level 1003 (0x3eb) are possible aliases
Level 258 (0x102) and level 1001 (0x3e9) are possible aliases

Found 4 aliased levels

Ad-Hoc Tests

- The smbtoriture ad-hoc tests are what I call the old tests from Samba3. These tests include:
 - FDPASS LOCK1 LOCK2 LOCK3 LOCK4 LOCK5
LOCK6 LOCK7 UNLINK ATTR TRANS2 MAXFID
TORTURE NEGNOWAIT NBENCH DIR DIR1 DENY1
DENY2 TCON TCONDEV RW1 RW2 OPEN DENY3
OPENATTR XCOPY RENAME DELETE PROPERTIES
MANGLE UTABLE CASETABLE PIPE_NUMBER
IOCTL CHKPATH

Running TCONDEV

Trying share IPC\$ with devtype A:

Trying share IPC\$ with devtype ?????

Trying share IPC\$ with devtype LPT:

Trying share IPC\$ with devtype IPC

Trying share IPC\$ with devtype FOOBA

Trying share test3 with devtype A:

Trying share test3 with devtype ?????

Trying share test3 with devtype LPT:

Trying share test3 with devtype IPC

Trying share test3 with devtype FOOBA

Passed tcondevtest

TCONDEV took 0.267684 secs

Masktest

- Masktest is a standalone test suite for wildcard handling
- Random wildcard patterns and filenames are generated
- The servers wildcard matching is compared to a known good implementation

seed=1060637141

```
++- +- 1 mask=[\masktest\>?] file=[\masktest\cceefj] rfile=[cceefj/cceefj]
--- --- 2 mask=[\masktest\**.<"c??".>?"??] file=[\masktest\ekicbmdhdajieeih] rfile=[ekicbmdhdajieeih/ekicbd~1]
--- --- 3 mask=[\masktest\a*b>a<"] file=[\masktest\hmjhe] rfile=[hmjhe/hmjhe]
--- --- 4 mask=[\masktest\*.??.>*a*<b"] file=[\masktest\df] rfile=[df/df]
--- --- 5 mask=[\masktest\<*ab"b"><"b"><"] file=[\masktest\jcmjeeahi.fdd] rfile=[jcmjeeahi.fdd/jcmjee~1.fdd]
--- --- 6 mask=[\masktest\b<?b"b????ac??<a] file=[\masktest\jda.hmecmjd.ggck] rfile=[jda.hmecmjd.ggck/jdahme~1.ggc]
--+ --+ 7 mask=[\masktest\><b*] file=[\masktest\mkde.b.hgiigfadiab] rfile=[mkde.b.hgiigfadiab/mkdeb~1.hgi]
--- --- 8 mask=[\masktest\ "><.*"c"b] file=[\masktest\b.icdieb.khhblhefh] rfile=[b.icdieb.khhblhefh/bicdie~1.khh]
```

gentest

- gentest is a dual-server randomised test suite. It provides the best test for compatibility between two implementations
- `gentest //server1/share1 //server2/share2 -User1%pass1 -User2%pass2`
- Major uses for gentest are:
 - comparison with a reference server
 - comparison between versions of your own server
 - consistency testing between 2 shares on your server

```
seed=3
Loaded 2 seeds from gentest_seeds.dat
Connecting to \\win2003\test2 as tridge - instance 0
Connecting to \\win2003\test2 as tridge - instance 1
Connecting to \\win2003\test3 as tridge - instance 0
Connecting to \\win2003\test3 as tridge - instance 1
Deleted 1 files on server 0
Deleted 1 files on server 1
OPEN num_open_handles=1 h=0 s1=0x8006 s2=0x800d (\gentest\a_very_long_name.bin)
0
Mismatch in all_info.out.fname - \test2\gentest\a_very_long_name.bin \test3\gentest\a_very_long_name.bin
Failed at operation 1 - QFILEINFO
```

locktest

- locktest is like gentest but only tests byte range locking requests
- A second test 'locktest2' is also available for testing the interaction of CIFS and NFS byte range locking
- `locktest //server1/share1 //server2/share2 -User1%pass1 -User2%pass2`

```
seed=1060637440 base=0 range=100 min_length=0
testing 1000 ...
reopen conn=0 f=0
reopen conn=1 f=0
unlock conn=0 f=1 range=98(0)      -> NT_STATUS_RANGE_NOT_LOCKED:NT_STATUS_RANGE_NOT_LOCKED
lock   conn=1 f=1 range=32(11) op=WRITE_LOCK -> NT_STATUS_OK:NT_STATUS_OK
reopen conn=0 f=1
lock   conn=1 f=1 range=80(17) op=READ_LOCK -> NT_STATUS_OK:NT_STATUS_OK
unlock conn=1 f=1 range=45(53)     -> NT_STATUS_RANGE_NOT_LOCKED:NT_STATUS_RANGE_NOT_LOCKED
```

A testing plan

- Start with nmblookup - this will test basic NBT name resolution
- Next move to smbclient to test basic SMB connectivity
- Then start on the smbtoriture commands, running them one at a time. Keep a sniffer running.
- Next, use masktest to test wildcard handling
- Then use locktest for byte range locking
- Finally use gentest