

Notes on the change from 16-bit UIDs to 32-bit UIDs

Author: Chris Wing <wingc@umich.edu>

Last updated: January 11, 2000

- kernel code MUST take into account `__kernel_uid_t` and `__kernel_uid32_t` when communicating between user and kernel space in an ioctl or data structure.
- kernel code should use `uid_t` and `gid_t` in kernel-private structures and code.

What's left to be done for 32-bit UIDs on all Linux architectures:

- Disk quotas have an interesting limitation that is not related to the maximum UID/GID. They are limited by the maximum file size on the underlying filesystem, because quota records are written at offsets corresponding to the UID in question. Further investigation is needed to see if the quota system can cope properly with huge UIDs. If it can deal with 64-bit file offsets on all architectures, this should not be a problem.
- Decide whether or not to keep backwards compatibility with the system accounting file, or if we should break it as the comments suggest (currently, the old 16-bit UID and GID are still written to disk, and part of the former pad space is used to store separate 32-bit UID and GID)
- Need to validate that OS emulation calls the 16-bit UID compatibility syscalls, if the OS being emulated used 16-bit UIDs, or uses the 32-bit UID system calls properly otherwise.

This affects at least:

- iBCS on Intel
 - sparc32 emulation on sparc64 (need to support whatever new 32-bit UID system calls are added to sparc32)
- Validate that all filesystems behave properly.

At present, 32-bit UIDs `_should_` work for:

- ext2
- ufs
- isofs
- nfs
- coda
- udf

Ioctl() fixups have been made for:

- ncpfs
- smbfs

Filesystems with simple fixups to prevent 16-bit UID wraparound:

- minix
- sysv
- qnx4

Other filesystems have not been checked yet.

- The ncpfs and smbfs filesystems cannot presently use 32-bit UIDs in all ioctl(s). Some new ioctl(s) have been added with 32-bit UIDs, but more are needed. (as well as new user<->kernel data structures)
- The ELF core dump format only supports 16-bit UIDs on arm, i386, m68k, sh, and sparc32. Fixing this is probably not that important, but would require adding a new ELF section.
- The ioctl(s) used to control the in-kernel NFS server only support 16-bit UIDs on arm, i386, m68k, sh, and sparc32.
- make sure that the UID mapping feature of AX25 networking works properly (it should be safe because it's always used a 32-bit integer to communicate between user and kernel)