List of Linux/i386 system calls

This list is NOT READY and is under heavy construction, a lot of entries are missing, and some may be incorrect. This is more a template than a real document. Meanwhile, I suggest you to examine this list by H-Peter Recktenwald.

Table of Contents (template)

---

1. Introduction

First of all note that these are not libc "system calls", but real system calls provided by Linux Kernel.

List is intended to cover Linux 2.4 / 2.2 / 2.0.

---

2. System call in depth

.. not ready yet ..

---

3. Linux/i386 system calls

All system calls introduced/removed in specific Linux version are marked with (VER+/-) label (f.e. 2.2+ means that this call was introduced in Linux 2.2, and is missing in Linux 2.0). Square brackets hold real kernel name of system call from `arch/i386/kernel/entry.S` (as appeared in *Syntax*), if it differs from "official" in `include/asm-i386/unistd.h`.

Complete list of system calls with description

0. sys_setup

Syntax: `int sys_setup(void)`

Source: fs/filesystems.c

Action: return -ENOSYS on Linux 2.2

Details: old sys_setup call

1. sys_exit

Syntax: `int sys_exit(int status)`

Source: kernel/exit.c

Action: terminate the current process

Details: `status` is return code

2. sys_fork

Syntax: `int sys_fork()`

Source: arch/i386/kernel/process.c

Action: create a child process

Details:

3. sys_read

Syntax: `ssize_t sys_read(unsigned int fd, char * buf, size_t count)`

Source: fs/read_write.c

Action: read from a file descriptor

Details:

4. sys_write

Syntax: `ssize_t sys_write(unsigned int fd, const char * buf, size_t count)`

Source: fs/read_write.c

Action: write to a file descriptor

Details:

5. sys_open

Syntax: `int sys_open(const char * filename, int flags, int mode)`

Source: fs/open.c

Action: open and possibly create a file or device

Details:

6. sys_close

Syntax: `sys_close(unsigned int fd)`

Source: fs/open.c

Action: close a file descriptor

Details:

7. sys_waitpid

Syntax: `int sys_waitpid(pid_t pid,unsigned int * stat_addr, int options)`

Source: kernel/exit.c

Action: wait for process termination

Details:

8. sys_creat

Syntax: `int sys_creat(const char * pathname, int mode)`

Source: fs/open.c

Action: create a file or device

Details:

9. sys_link

Syntax: `int sys_link(const char * oldname, const char * newname)`

Source: fs/namei.c

Action: make a new name for a file

Details:

10. sys_unlink

Syntax: `int sys_unlink(const char * pathname)`

Source: fs/namei.c

Action: delete a name and possibly the file it refers to

Details:

11. sys_execve

Syntax: `int sys_execve(struct pt_regs regs)`

Source: arch/i386/kernel/process.c

Action: execute program

Details:

12. sys_chdir

Syntax: `int sys_chdir(const char * filename)`

Source: fs/open.c

Action: change working directory

Details:

13. sys_time

Syntax: `int sys_time(int * tloc)`

Source: kernel/time.c

Action: get time in seconds

Details:

14. sys_mknod

Syntax: `int sys_mknod(const char * filename, int mode, dev_t dev)`

Source: fs/namei.c

Action: create a directory or special or ordinary file

Details:

15. sys_chmod

Syntax: `int sys_chmod(const char * filename, mode_t mode)`

Source: fs/open.c

Action: change permissions of a file

Details:

16. sys_lchown

Syntax: `int sys_lchown(const char * filename, uid_t user, gid_t group)`

Source: fs/open.c

Action: change ownership of a file

Details:

17. sys_break

Syntax: `int sys_break()`

Source: kernel/sys.c

Action: return -ENOSYS

Details: call exists only for compatibility

18. sys_oldstat

Syntax: `int sys_stat(char * filename, struct __old_kernel_stat * statbuf)`

Source: fs/stat.c

Action:

Details: obsolote

19. sys_lseek

Syntax: `off_t sys_lseek(unsigned int fd, off_t offset, unsigned int origin)`

Source: fs/read_write.c

Action: reposition read/write file offset

Details:

20. sys_getpid

Syntax: `int sys_getpid(void)`

Source: kernel/sched.c

Action: get process identification

Details:

21. sys_mount

Syntax: `int sys_mount(char * dev_name, char * dir_name, char * type, unsigned long new_flags, void * data)`

Source: fs/super.c

Action: mount filesystems

Details:

22. sys_umount

Syntax: `int sys_oldumount(char * name)`

Source: fs/super.c

Action: unmount filesystem

Details:

23. sys_setuid

Syntax: `int sys_setuid(uid_t uid)`

Source: kernel/sys.c

Action: set user identity

Details:

24. sys_getuid

Syntax: `int sys_getuid(void)`

Source: kernel/sys.c

Action: get user identity

Details:

25. sys_stime

Syntax: `int sys_stime(int * tptr)`

Source: kernel/time.c

Action: set time

Details:

26. sys_ptrace

Syntax: `int sys_ptrace(long request, long pid, long addr, long data)`

Source: arch/i386/kernel/ptrace.c

Action: process trace

Details:

27. sys_alarm

Syntax: `unsigned int sys_alarm(unsigned int seconds)`

Source: kernel/sched.c

Action: set an alarm clock for delivery of a signal

Details:

28. sys_oldfstat

Syntax: `int sys_fstat(unsigned int fd, struct __old_kernel_stat * statbuf)`

Source: fs/stat.c

Action:

Details: obsolete

29. sys_pause

Syntax: `int sys_pause(void)`

Source: arch/i386/kernel/sys_i386.c

Action: wait for signal

Details:

30. sys_utime

Syntax: `int sys_utime(char * filename, struct utimbuf * times)`

Source: fs/open.c

Action: change access and/or modification times of an inode

Details:

...

List by system call number

00 sys_setup [sys_ni_syscall]
01 sys_exit
02 sys_fork
03 sys_read
04 sys_write
05 sys_open
06 sys_close
07 sys_waitpid
08 sys_creat
09 sys_link
10 sys_unlink
11 sys_execve
12 sys_chdir
13 sys_time
14 sys_mknod
15 sys_chmod
16 sys_lchown
17 sys_break [sys_ni_syscall]
18 sys_oldstat [sys_stat]
19 sys_lseek
20 sys_getpid
21 sys_mount
22 sys_umount [sys_oldumount]
23 sys_setuid
24 sys_getuid
25 sys_stime
26 sys_ptrace
27 sys_alarm
28 sys_oldfstat [sys_fstat]
29 sys_pause
30 sys_utime
31 sys_stty [sys_ni_syscall]
32 sys_gtty [sys_ni_syscall]
33 sys_access
34 sys_nice
35 sys_ftime [sys_ni_syscall]
36 sys_sync
37 sys_kill
38 sys_rename
39 sys_mkdir
40 sys_rmdir
41 sys_dup
42 sys_pipe
43 sys_times
44 sys_prof [sys_ni_syscall]
45 sys_brk
46 sys_setgid
47 sys_getgid
48 sys_signal
49 sys_geteuid
50 sys_getegid
51 sys_acct
52 sys_umount2 [sys_umount] (2.2+)
53 sys_lock [sys_ni_syscall]
54 sys_ioctl
70 sys_setreuid
71 sys_setregid
72 sys_sigsuspend
73 sys_sigpending
74 sys_sethostname
75 sys_setrlimit
76 sys_getrlimit
77 sys_getrusage
78 sys_gettimeofday
79 sys_settimeofday
80 sys_getgroups
81 sys_setgroups
82 sys_select [old_select]
83 sys_symlink
84 sys_oldlstat [sys_lstat]
85 sys_readlink
86 sys_uselib
87 sys_swapon
88 sys_reboot
89 sys_readdir [old_readdir]
90 sys_mmap [old_mmap]
91 sys_munmap
92 sys_truncate
93 sys_ftruncate
94 sys_fchmod
95 sys_fchown
96 sys_getpriority
97 sys_setpriority
98 sys_profil [sys_ni_syscall]
99 sys_statfs
100 sys_fstatfs
101 sys_ioperm
102 sys_socketcall
103 sys_syslog
104 sys_setitimer
105 sys_getitimer
106 sys_stat [sys_newstat]
107 sys_lstat [sys_newlstat]
108 sys_fstat [sys_newfstat]
109 sys_olduname [sys_uname]
110 sys_iopl
111 sys_vhangup
112 sys_idle
113 sys_vm86old
114 sys_wait4
115 sys_swapoff
116 sys_sysinfo
117 sys_ipc
118 sys_fsync
119 sys_sigreturn
120 sys_clone
121 sys_setdomainname
122 sys_uname [sys_newuname]
123 sys_modify_ldt
124 sys_adjtimex
140 sys__llseek [sys_lseek]
141 sys_getdents
142 sys__newselect [sys_select]
143 sys_flock
144 sys_msync
145 sys_readv
146 sys_writev
147 sys_getsid
148 sys_fdatasync
149 sys__sysctl [sys_sysctl]
150 sys_mlock
151 sys_munlock
152 sys_mlockall
153 sys_munlockall
154 sys_sched_setparam
155 sys_sched_getparam
156 sys_sched_setscheduler
157 sys_sched_getscheduler
158 sys_sched_yield
159 sys_sched_get_priority_max
160 sys_sched_get_priority_min
161 sys_sched_rr_get_interval
162 sys_nanosleep
163 sys_mremap
164 sys_setresuid (2.2+)
165 sys_getresuid (2.2+)
166 sys_vm86
167 sys_query_module (2.2+)
168 sys_poll (2.2+)
169 sys_nfsservctl (2.2+)
170 sys_setresgid (2.2+)
171 sys_getresgid (2.2+)
172 sys_prctl (2.2+)
173 sys_rt_sigreturn (2.2+)
174 sys_rt_sigaction (2.2+)
175 sys_rt_sigprocmask (2.2+)
176 sys_rt_sigpending (2.2+)
177 sys_rt_sigtimedwait (2.2+)
178 sys_rt_sigqueueinfo (2.2+)
179 sys_rt_sigsuspend (2.2+)
180 sys_pread (2.2+)
181 sys_pwrite (2.2+)
182 sys_chown (2.2+)
183 sys_getcwd (2.2+)
184 sys_capget (2.2+)
185 sys_capset (2.2+)
186 sys_sigaltstack (2.2+)
187 sys_sendfile (2.2+)
188 sys_getpmsg [sys_ni_syscall]
189 sys_putpmsg [sys_ni_syscall]
190 sys_vfork (2.2+)

55 sys_fcntl
56 sys_mpx [sys_ni_syscall]
57 sys_setpgid
58 sys_ulimit [sys_ni_syscall]
59 sys_oldolduname
60 sys_umask
61 sys_chroot
62 sys_ustat
63 sys_dup2
64 sys_getppid
65 sys_getpgrp
66 sys_setsid
67 sys_sigaction
68 sys_sgetmask
69 sys_ssetmask
125 sys_mprotect
126 sys_sigprocmask
127 sys_create_module
128 sys_init_module
129 sys_delete_module
130 sys_get_kernel_syms
131 sys_quotactl
132 sys_getpgid
133 sys_fchdir
134 sys_bdflush
135 sys_sysfs
136 sys_personality
137 sys_afs_syscall [sys_ni_syscall]
138 sys_setfsuid
139 sys_setfsgid

List by system call name

.. not ready yet ..

List by kernel source

arch/i386/ (23) fs/ (62) ipc/ (11) kernel/ (81) mm/ (12) net/ (1)

arch/i386/

arch/i386/kernel/sys_i386.c

```
int sys_pipe(unsigned long * fildes)
int sys_ipc (uint call, int first, int second, int third, void *ptr, long fifth)
int sys_uname(struct old_utsname * name)
int sys_olduname(struct oldold_utsname * name)
int sys_pause(void)
int old_mmap(struct mmap_arg_struct *arg)
```

arch/i386/kernel/ioport.c

```
int sys_ioperm(unsigned long from, unsigned long num, int turn_on)
int sys_iopl(unsigned long unused)
```

arch/i386/kernel/process.c

```
int sys_idle(void)
int sys_fork(struct pt_regs regs)
int sys_clone(struct pt_regs regs)
int sys_vfork(struct pt_regs regs)
int sys_execve(struct pt_regs regs)
```

arch/i386/kernel/vm86.c

```
int sys_vm86old(struct vm86_struct * v86)
int sys_vm86(unsigned long subfunction, struct vm86plus_struct * v86)
```

arch/i386/kernel/ptrace.c

```
int sys_ptrace(long request, long pid, long addr, long data)
```

arch/i386/kernel/signal.c

```
int sys_sigsuspend(int history0, int history1, old_sigset_t mask)
int sys_rt_sigsuspend(sigset_t *unewset, size_t sigsetsize)
int sys_sigaction(int sig, const struct old_sigaction *act, struct old_sigaction *oact)
int sys_sigaltstack(const stack_t *uss, stack_t *uoss)
int sys_sigreturn(unsigned long __unused)
int sys_rt_sigreturn(unsigned long __unused)
```

arch/i386/kernel/ldt.c

```
int sys_modify_ldt(int func, void *ptr, unsigned long bytecount)
```

fs/

fs/stat.c

```
int sys_stat(char * filename, struct __old_kernel_stat * statbuf)
int sys_newstat(char * filename, struct stat * statbuf)
int sys_lstat(char * filename, struct __old_kernel_stat * statbuf)
int sys_newlstat(char * filename, struct stat * statbuf)
int sys_fstat(unsigned int fd, struct __old_kernel_stat * statbuf)
int sys_newfstat(unsigned int fd, struct stat * statbuf)
int sys_readlink(const char * path, char * buf, int bufsiz)
```

fs/read_write.c

```
off_t sys_lseek(unsigned int fd, off_t offset, unsigned int origin)
int sys_llseek(unsigned int fd, unsigned long offset_high, unsigned long offset_low, loff_t * result, unsigned int origin)
ssize_t sys_read(unsigned int fd, char * buf, size_t count)
ssize_t sys_write(unsigned int fd, const char * buf, size_t count)
ssize_t sys_readv(unsigned long fd, const struct iovec * vector, unsigned long count)
ssize_t sys_writev(unsigned long fd, const struct iovec * vector, unsigned long count)
ssize_t sys_pread(unsigned int fd, char * buf, size_t count, loff_t pos)
ssize_t sys_pwrite(unsigned int fd, const char * buf, size_t count, loff_t pos)
```

fs/buffer.c

```
int sys_sync(void)
int sys_fsync(unsigned int fd)
int sys_fdatasync(unsigned int fd)
int sys_bdflush(int func, long data)
```

fs/open.c

```
int sys_statfs(const char * path, struct statfs * buf)
int sys_fstatfs(unsigned int fd, struct statfs * buf)
int sys_truncate(const char * path, unsigned long length)
int sys_ftruncate(unsigned int fd, unsigned long length)
int sys_utime(char * filename, struct utimbuf * times)
int sys_utimes(char * filename, struct timeval * utimes)
int sys_access(const char * filename, int mode)
int sys_chdir(const char * filename)
int sys_fchdir(unsigned int fd)
int sys_chroot(const char * filename)
int sys_fchmod(unsigned int fd, mode_t mode)
int sys_chmod(const char * filename, mode_t mode)
int sys_chown(const char * filename, uid_t user, gid_t group)
int sys_lchown(const char * filename, uid_t user, gid_t group)
int sys_fchown(unsigned int fd, uid_t user, gid_t group)
int sys_open(const char * filename, int flags, int mode)
int sys_creat(const char * pathname, int mode)
int sys_close(unsigned int fd)
int sys_vhangup(void)
```

fs/exec.c

```
int sys_uselib(const char * library)
```

fs/super.c

```
int sys_sysfs(int option, unsigned long arg1, unsigned long arg2)
int sys_ustat(dev_t dev, struct ustat * ubuf)
int sys_umount(char * name, int flags)
int sys_oldumount(char * name)
int sys_mount(char * dev_name, char * dir_name, char * type, unsigned long new_flags, void * data)
```

fs/fcntl.c

```
int sys_dup2(unsigned int oldfd, unsigned int newfd)
int sys_dup(unsigned int fildes)
long sys_fcntl(unsigned int fd, unsigned int cmd, unsigned long arg)
```

fs/namei.c

```
int sys_mknod(const char * filename, int mode, dev_t dev)
int sys_mkdir(const char * pathname, int mode)
int sys_rmdir(const char * pathname)
int sys_unlink(const char * pathname)
int sys_symlink(const char * oldname, const char * newname)
int sys_link(const char * oldname, const char * newname)
int sys_rename(const char * oldname, const char * newname)
```

fs/ioctl.c

```
int sys_ioctl(unsigned int fd, unsigned int cmd, unsigned long arg)
```

fs/select.c

```
int sys_select(int n, fd_set *inp, fd_set *outp, fd_set *exp, struct timeval *tvp)
int sys_poll(struct pollfd * ufds, unsigned int nfds, long timeout)
```

fs/locks.c

```
int sys_flock(unsigned int fd, unsigned int cmd)
```

fs/filesystems.c

```
int sys_nfsservctl(int cmd, void *argp, void *resp) [fs/nfsd/nfsctl.c]
```

fs/dquot.c

```
int sys_quotactl(int cmd, const char *special, int id, caddr_t addr)
```

fs/dcache.c

```
int sys_getcwd(char *buf, unsigned long size)
```

fs/readdir.c

```
int sys_getdents(unsigned int fd, void * dirent, unsigned int count)
```

ipc/

ipc/msg.c

```
int sys_msgsnd (int msqid, struct msgbuf *msgp, size_t msgsz, int msgflg)
int sys_msgrcv (int msqid, struct msgbuf *msgp, size_t msgsz, long msgtyp, int msgflg)
int sys_msgget (key_t key, int msgflg)
int sys_msgctl (int msqid, int cmd, struct msqid_ds *buf)
```

ipc/sem.c

```
int sys_semget (key_t key, int nsems, int semflg)
int sys_semctl (int semid, int semnum, int cmd, union semun arg)
int sys_semop (int semid, struct sembuf *tsops, unsigned nsops)
```

ipc/shm.c

```
int sys_shmget (key_t key, int size, int shmflg)
int sys_shmctl (int shmid, int cmd, struct shmid_ds *buf)
int sys_shmat (int shmid, char *shmaddr, int shmflg, ulong *raddr)
int sys_shmdt (char *shmaddr)
```

kernel/

kernel/sched.c

```
unsigned int sys_alarm(unsigned int seconds)
int sys_getpid(void)
int sys_getppid(void)
int sys_getuid(void)
int sys_geteuid(void)
int sys_getgid(void)
int sys_getegid(void)
int sys_nice(int increment)
int sys_sched_setscheduler(pid_t pid, int policy, struct sched_param *param)
int sys_sched_setparam(pid_t pid, struct sched_param *param)
int sys_sched_getscheduler(pid_t pid)
int sys_sched_getparam(pid_t pid, struct sched_param *param)
int sys_sched_yield(void)
int sys_sched_get_priority_max(int policy)
int sys_sched_get_priority_min(int policy)
int sys_sched_rr_get_interval(pid_t pid, struct timespec *interval)
int sys_nanosleep(struct timespec *rqtp, struct timespec *rmtp)
```

kernel/exit.c

```
int sys_exit(int error_code)
int sys_wait4(pid_t pid,unsigned int * stat_addr, int options, struct rusage * ru)
int sys_waitpid(pid_t pid,unsigned int * stat_addr, int options)
```

kernel/signal.c

```
int sys_rt_sigprocmask(int how, sigset_t *set, sigset_t *oset, size_t sigsetsize)
int sys_rt_sigpending(sigset_t *set, size_t sigsetsize)
int sys_rt_sigtimedwait(const sigset_t *uthese, siginfo_t *uinfo, const struct timespec *uts, size_t sigsetsize)
int sys_kill(int pid, int sig)
int sys_rt_sigqueueinfo(int pid, int sig, siginfo_t *uinfo)
int sys_sigprocmask(int how, old_sigset_t *set, old_sigset_t *oset)
int sys_sigpending(old_sigset_t *set)
int sys_rt_sigaction(int sig, const struct sigaction *act, struct sigaction *oact, size_t sigsetsize)
int sys_sgetmask(void)
int sys_ssetmask(int newmask)
unsigned long sys_signal(int sig, __sighandler_t handler)
```

kernel/printk.c

```
int sys_syslog(int type, char * buf, int len)
```

kernel/sys.c

```
int sys_ni_syscall(void)
int sys_setpriority(int which, int who, int niceval)
int sys_getpriority(int which, int who)
int sys_reboot(int magic1, int magic2, int cmd, void * arg)
int sys_setregid(gid_t rgid, gid_t egid)
```

```
int sys_setgid(gid_t gid)
int sys_setreuid(uid_t ruid, uid_t euid)
int sys_setuid(uid_t uid)
int sys_setresuid(uid_t ruid, uid_t euid, uid_t suid)
int sys_getresuid(uid_t *ruid, uid_t *euid, uid_t *suid)
int sys_setresgid(gid_t rgid, gid_t egid, gid_t sgid)
int sys_getresgid(gid_t *rgid, gid_t *egid, gid_t *sgid)
int sys_setfsuid(uid_t uid)
long sys_times(struct tms * tbuf)
int sys_setpgid(pid_t pid, pid_t pgid)
int sys_getpgid(pid_t pid)
int sys_getpgrp(void)
int sys_getsid(pid_t pid)
int sys_setsid(void)
int sys_getgroups(int gidsetsize, gid_t *grouplist)
int sys_setgroups(int gidsetsize, gid_t *grouplist)
int sys_newuname(struct new_utsname * name)
int sys_sethostname(char *name, int len)
int sys_gethostname(char *name, int len)
int sys_setdomainname(char *name, int len)
int sys_getrlimit(unsigned int resource, struct rlimit *rlim)
int sys_setrlimit(unsigned int resource, struct rlimit *rlim)
int sys_getrusage(int who, struct rusage *ru)
int sys_umask(int mask)
int sys_prctl(int option, unsigned long arg2, unsigned long arg3, unsigned long arg4, unsigned long arg5)
```

kernel/module.c

```
unsigned long sys_create_module(const char *name_user, size_t size)
int sys_init_module(const char *name_user, struct module *mod_user)
int sys_delete_module(const char *name_user)
int sys_query_module(const char *name_user, int which, char *buf, size_t bufsize, size_t *ret)
int sys_get_kernel_syms(struct kernel_sym *table)
unsigned long sys_create_module(const char *name_user, size_t size)
```

kernel/itimer.c

```
int sys_getitimer(int which, struct itimerval *value)
int sys_setitimer(int which, struct itimerval *value, struct itimerval *ovalue)
```

kernel/info.c

```
int sys_sysinfo(struct sysinfo *info)
```

kernel/time.c

```
int sys_time(int * tloc)
int sys_stime(int * tptr)
int sys_gettimeofday(struct timeval *tv, struct timezone *tz)
int sys_settimeofday(struct timeval *tv, struct timezone *tz)
int sys_adjtimex(struct timex *txc_p)
```

kernel/exec_domain.c

```
int sys_adjtimex(struct timex *txc_p)
```

kernel/sysctl.c

```
int sys_sysctl(struct __sysctl_args *args)
```

kernel/acct.c

```
int sys_acct(const char *name)
```

kernel/capability.c

```
int sys_capget(cap_user_header_t header, cap_user_data_t dataptr)
int sys_capset(cap_user_header_t header, const cap_user_data_t data)
```

mm/

mm/mmap.c

```
unsigned long sys_brk(unsigned long brk)
int sys_munmap(unsigned long addr, size_t len)
```

mm/mprotect.c

```
int sys_mprotect(unsigned long start, size_t len, unsigned long prot)
```

mm/filemap.c

```
ssize_t sys_sendfile(int out_fd, int in_fd, off_t *offset, size_t count)
int sys_msync(unsigned long start, size_t len, int flags)
```

mm/mlock.c

```
int sys_mlock(unsigned long start, size_t len)
int sys_munlock(unsigned long start, size_t len)
int sys_mlockall(int flags)
int sys_munlockall(void)
```

mm/swapfile.c

```
int sys_swapoff(const char * specialfile)
int sys_swapon(const char * specialfile, int swap_flags)
```

mm/mremap.c

```
unsigned long sys_mremap(unsigned long addr, unsigned long old_len, unsigned long new_len, unsigned long flags)
```

net/

net/socket.c

```
int sys_socketcall(int call, unsigned long *args)

 int sys_socket(int family, int type, int protocol)
 int sys_socketpair(int family, int type, int protocol, int usockvec[2])
 int sys_bind(int fd, struct sockaddr *umyaddr, int addrlen)
 int sys_listen(int fd, int backlog)
 int sys_accept(int fd, struct sockaddr *upeer_sockaddr, int *upeer_addrlen)
 int sys_connect(int fd, struct sockaddr *uservaddr, int addrlen)
 int sys_getsockname(int fd, struct sockaddr *usockaddr, int *usockaddr_len)
 int sys_getpeername(int fd, struct sockaddr *usockaddr, int *usockaddr_len)
 int sys_sendto(int fd, void * buff, size_t len, unsigned flags, struct sockaddr *addr, int addr_len)
 int sys_send(int fd, void * buff, size_t len, unsigned flags)
 int sys_recvfrom(int fd, void * ubuf, size_t size, unsigned flags, struct sockaddr *addr, int *addr_len)
 int sys_recv(int fd, void * ubuf, size_t size, unsigned flags)
 int sys_setsockopt(int fd, int level, int optname, char *optval, int optlen)
 int sys_getsockopt(int fd, int level, int optname, char *optval, int *optlen)
 int sys_shutdown(int fd, int how)
 int sys_sendmsg(int fd, struct msghdr *msg, unsigned flags)
 int sys_recvmsg(int fd, struct msghdr *msg, unsigned int flags)
```

References

Sources of information (except other directly pointed):

include/asm-i386/unistd.h
arch/i386/kernel/entry.S
include/linux/sys.h

$Id: syscall.html,v 1.7 2000/05/18 12:18:52 konst Exp $