SYSTEM CALL IN LINUX

System Call	Category	Syntax	Description	Source
fork	Process Control	pid_t fork(void)	Create a child process	arch/i386/kernel/process.c
kill	Process Control	int kill(pid_t pid, int sig)	Send a signal to a process	kernel/signal.c
getpid	Process Control	pid_t getpid (void)	Get process ID of the calling process	kernel/sched.c
getppid	Process Control	pid_t getppid (void)	Get process ID of the parent of the calling process	kernel/sched.c
setpgid	Process Control	int setpgrp (void); int setpgrp(pid_t pid, pid_t pgid)	Set process group ID	kernel/sys.c
getpgrp	Process Control	pid_t getpgrp (void); pid_t getpgrp (pid_t pid)	Get the process Group ID	kernel/sys.c
open	File Management	int open (const char * filename, int flags, int mode)	Open a file or device	fs/open.c
close	File Management	int close (unsigned int fd)	Close a file descriptor	fs/open.c
chmod	File Management	int chmod (const char *path, mode_t mode)	Changes the permissions of a file specified by path	fs/open.c
fchmod	File Management	int fhmod (int fd, mode_t mode)	Changes the permissions of the file referred to by the open file descriptor fd	fs/open.c
chown	File Management	int chown (const char *path, uid_t owner, gid_t group)	Changes the ownership of a file specified by path	fs/open.c
fchown	File Management	int fchown (int fd, uid_t owner, gid_t group)	Changes the ownership of the file referred to by the open file descriptor fd	fs/open.c
chdir	File Management	int chdir (const char *path)	Change working directory	fs/open.c

mount	File Management	int mount(const char *source, const char *target, const char *filesystemtype, unsigned long mountflags, const void *data)	Attaches the filesystem specified by source (which is often a device name) to the directory specified by *target*.	fs/super.c
umount	File Management	int umount(const char *target)	Remove the attachment of the filesystem mounted on *target*	fs/super.c
ioctl	Device Management	int ioctl(int d, int request, …);	Control Device	fs/ioctl.c
fsync	Device Management	int fsync(int fd);	Synchronize a file's complete in-core state with that on disk	fs/buffer.c
alarm	Information Maintenance	unsigned int alarm(unsigned int seconds);	Arranges for a SIGALRM signal to be delivered to the calling process in seconds.	kernel/sched.c
getpriority	Information Maintenance	int getpriority(int which, int who)	Get program scheduling priority	kernel/sys.c
setpriority	Information Maintenance	int setpriority(int which, int who, int prio)	Set Program Scheduling Priority	kernel/sys.c
statfs	Information Maintenance	int statfs (const char *path, struct statfs *buf)	Get file system Statistics	fs/open.c
getdents	Information Maintenance	int getdents(int fd, struct linux_dirent *dirp, int count)	Read directory entries	fs/readdir.c
socket	Communication	int socket(int domain, int type, int protocol);	Create an endpoint for communication	net/socket.c
socketcall	Communication	int socketcall(int call, unsigned long *args);	Socket System Calls	net/socket.c
connect	Communication	int connect(int sockfd, const struct sockaddr *addr, socklen_t addrlen);	Connects the socket referred to by the file descriptor sockfd to the address specified by addr and addrlen argument specifies the size of addr	net/socket.c
send	Communication	ssize_t send(int sockfd, const void *buf, size_t len, int flags);	Used to transmit a message to another socket	mm/filemap.c