README

This constitutes the Functional Specification for Red Hat Linux Enterprise 4-U1.

The content is organized into the following tables.

System Calls — The list of all the system calls for the Red Hat Enterprise Linux 4-U1 kernel. It gives information about their security relevance with respect to the Security Target, their numbers relating to eServer series product lines, the High Level Design module to which they belong, the prototype of the calls, test case coverage, and documentation.

Programs — This lists the trusted programs in the Target Of Evaluation (TOE) with pointers to the location where they are specified.

Database Files — This lists the files that make up the Target of Evaluation Security Function (TSF) Database with pointers to the location where they are specified.

Note: The documentation referenced in tables "System Calls", "Programs", and "Database Files" is the security relevant subset of the functional specification of the TOE.

Note: The Target of Evaluation Security Functions Interfaces (TSFI) consist of the security relevant system calls, the TOE's trusted programs, and the TSF Database. Hence, the Tables "System Calls" (marked as security relevant), "Programs" and "Database Files" contain the complete list of TSFI.

Note: The functional specification of the TSF identified in the Security Target (ST) is provided by the description of the external interfaces (TSFI) associated with these functions. The exceptions to this rule are the following functions that cannot be described completely via their external interface, either because there are no external interfaces, or the security functionality is not directly visible at the interface.

AU.3 "Audit Record Format"
TP.1 "TSF Invocation Guarantees"
TP.6 "Internal TOE Protection Mechanisms"
OR.1 "Object Reuse: File System Objects"
OR.2 "Object Reuse: IPC Objects"
OR.3 "Object Reuse: Memory Objects"

For these TSF, their functional specification is already provided as their respective functional description in the TOE summary specification of the Security Target. Their implementation in the High Level Design directly stems from the functional specification.

TSF Legend		HLD Legend		Privilege	
TP: TSF Protection		PC	Process Control		
TP.1	TSF Invocation Guarantees	FS	File System and IO	no	no privilege required
TP.2	Kernel	MM	Memory Management	addtl	mode of operation different if privileged
TP.3	Kernel Modules	NI	Networking	admin only	can only be run by authorized administrator
TP.4	Trusted Processes	IPC	Interprocess Communications		
TP.5	TSF Databases				
		SM	System Management		
SC: Secure Communications		IA	Identification and Authentication		
SC.1	Secure Protocols	NA	Network Applications		
		UA	User Level Audit		
SM: Security Management		BP	Batch Processing		
SM.1	Roles (user, admin)	SI	System Initialization		
SM.2	Access Control Configuration & Management	KM	Kernel Modules		
SM.3	Management of User, Group & Authentication Data				
SM.4	Management of Audit Configuration				
SM.5	Reliable Time Stamps				
OR: Object Reuse					
OR.1	File System Objects				
OR.2	IPC Objects				
OR.3	Memory Objects				
DA: Discretionary Access Control					
DA.1	General DAC Policy				
DA.2	Permission Bits				
DA.3	Access Control Lists				
DA.4	DAC: IPC objects				
AU: Audit					
AU.1	Audit Configuration				
AU.2	Audit Processing				
AU.3	Audit Record Format				
AU.4	Audit Post-processing				
IA: Identification and Authentication					
IA.1	User Identification and Authentication Data Mgmt				
IA.2	Common Authentication Mechanism				
IA.3	Interactive Login and Related Mechanisms				
IA.4	User Identity Changing				
IA.5	Login Processing				

Source of this system call list:

1. We employed a self-written Perl script that sources arch/<PLATFORM>/[entry.S|misc.S] and/or include/asm-<PLATFORM>/unistd.h for retrieving all system calls with their system call numbers for each platform.

2. The prefixes of [sys_|ppc_|ppc64] are stripped to retrieve the valid names of the system calls.

3. The table collapses variants of a system call into a single row, such as obsolescent APIs retained for backwards compatibility or minor variations in argument sizes (for example setuid and setuid16, or truncate and truncate64). The corresponding system call numbers for each variant are preserved in the columns listing the architecture-specific syscall numbers.

4. This script sources the man page for each system call found in the source code to extract the description and the function definition.

5. This script uses a preliminary version of the output table to check for given information about security relevance. This information about security relevance is retrieved by checking the functionality documented in the man page of each system call against the TSF statements in the Security Target.

6. In case a system call does not find the man page, or the security relevance information, it would visibly mark this system call being incompletely documented.

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
accept	accept a connection on a socket	no	ltp-full/testcases/kernel/syscalls/accept/accept01.c		no	IPC	int accept(int s, struct sockaddr *addr, socklen_t *addrlen);	accept.2.gz
access	check user's permissions for a file	yes	ltp-full/testcases/kernel/syscalls/access/access01.c ltp-full/testcases/kernel/syscalls/access/access02.c ltp-full/testcases/kernel/syscalls/access/access03.c ltp-full/testcases/kernel/syscalls/access/access04.c ltp-full/testcases/kernel/syscalls/access/access05.c ltp-full/testcases/audit/syscalls/access_test.c	DA.1, DA.2, DA.3	no	FS	int access(const char *pathname, int mode);	access.2.gz
acct	switch process accounting on or off	no			admin only	PC	int acct(const char *filename);	acct.2.gz
adjtimex	tune kernel clock	yes	ltp-full/testcases/kernel/syscalls/adjtimex/adjtimex01c ltp-full/testcases/kernel/syscalls/adjtimex/adjtimex02.c ltp-full/testcases/audit/syscalls/adjtimex_test.c	SM.5	addtl	PC	int adjtimex(struct timex *buf);	adjtimex.2.gz
alarm	set an alarm clock for delivery of a signal	no	ltp-full/testcases/kernel/syscalls/alarm/alarm01.c ltp-full/testcases/kernel/syscalls/alarm/alarm02.c ltp-full/testcases/kernel/syscalls/alarm/alarm03.c ltp-full/testcases/kernel/syscalls/alarm/alarm04.c ltp-full/testcases/kernel/syscalls/alarm/alarm05.c ltp-full/testcases/kernel/syscalls/alarm/alarm06.c ltp-full/testcases/kernel/syscalls/alarm/alarm07.c		no	PC	unsigned int alarm(unsigned int seconds);	alarm.2.gz
arch_prctl	Set architecture specific thread state.	no			no	PC	int arch_prctl(int code, unsigned long addr)	arch_prctl.2.gz
bdflush	start, flush, or tune buffer-dirty-flush daemon	no			admin only	MM	int bdflush(int func, long *address);	bdflush.2.gz
bind	bind a name to a socket	yes	ltp-full/testcases/kernel/syscalls/bind/bind01.c ltp-full/testcases/kernel/syscalls/bind/bind02.c ltp-full/testcases/audit/syscalls/bind_test.c	TP.4	addtl	NI	int bind(int sockfd, struct sockaddr *my_addr, socklen_t addrlen);	bind.2.gz
brk	change data segment size	yes	ltp-full/testcases/kernel/syscalls/brk/brk01.c	OR.3	no	MM	int brk(void *end_data_segment);	brk.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
capget	set/get process capabilities	no	ltp-full/testcases/kernel/syscalls/capset/capset01.c ltp-full/testcases/kernel/syscalls/capset/capset02.c		no	PC	int capget(cap_user_header_t header, cap_user_data_t data);	capget.2.gz
capset	set/get process capabilities	yes	ltp-full/testcases/kernel/syscalls/capset/capset01.c ltp-full/testcases/kernel/syscalls/capset/capset02.c ltp-full/testcases/audit/syscalls/capset_test.c	SM.1	no	PC	int capset(cap_user_header_t header, const cap_user_data_t data);	capset.2.gz
chdir	change working directory	yes	ltp-full/testcases/kernel/syscalls/chdir/chdir01.c ltp-full/testcases/kernel/syscalls/chdir/chdir02.c ltp-full/testcases/kernel/syscalls/chdir/chdir03.c ltp-full/testcases/kernel/syscalls/chdir/chdir04.c ltp-full/testcases/audit/syscalls/chdir_test.c	DA.1, DA.2, DA.3	addtl	FS	int chdir(const char *path);	chdir.2.gz
chmod	change permissions of a file	yes	ltp-full/testcases/kernel/syscalls/chmod/chmod01.c ltp-full/testcases/kernel/syscalls/chmod/chmod02.c ltp-full/testcases/kernel/syscalls/chmod/chmod03.c ltp-full/testcases/kernel/syscalls/chmod/chmod04.c ltp-full/testcases/kernel/syscalls/chmod/chmod05.c ltp-full/testcases/kernel/syscalls/chmod/chmod06.c ltp-full/testcases/kernel/syscalls/chmod/chmod07.c ltp-full/testcases/audit/syscalls/chmod_test.c	DA.1, DA.2, DA.3, SM.2	addtl	FS	int chmod(const char *path, mode_t mode);	chmod.2.gz
chown	change ownership of a file	yes	ltp-fulltestcases/kernel/syscalls/chown/chown01.c ltp-full/testcases/kernel/syscalls/chown/chown02.c ltp-full/testcases/kernel/syscalls/chown/chown03.c ltp-full/testcases/kernel/syscalls/chown/chown04.c ltp-full/testcases/kernel/syscalls/chown/chown05.c ltp-full/testcases/audit/syscalls/chown_test.c	DA.1, DA.2, DA.3, SM.2	addtl	FS	int chown(const char *path, uid_t owner, gid_t group);	chown.2.gz
chroot	change root directory	no	ltp-full/testcases/audit/syscalls/chroot_test.c		admin only	FS	int chroot(const char *path);	chroot.2.gz
clock_getres	clock and time functions	no			no	PC	int clock_getres(clockid_t clk_id, struct timespec *res);	clock_getres.3.gz
clock_gettime	clock and time functions	no			no	PC	int clock_gettime(clockid_t clk_id, struct timespec *tp);	clock_gettime.3.gz
clock_nanosleep	high resolution sleep with specifiable clock	no			no	PC	int clock_nanosleep(clockid_t clock_id, int flags, const struct timespec *rqtp, struct timespec *rmtp);	clock_nanosleep.3p.gz
clock_settime	clock and time functions	no			no	PC	int clock_settime(clockid_t clk_id, const struct timespec *tp);	clock_settime.3.gz
clone	Create a child process	yes	ltp-full/testcases/kernel/syscalls/clone/clone01.c ltp-full/testcases/kernel/syscalls/clone/clone02.c ltp-full/testcases/kernel/syscalls/clone/clone03.c ltp-full/testcases/kernel/syscalls/clone/clone04.c ltp-full/testcases/kernel/syscalls/clonet/clone05.c ltp-full/testcases/kernel/syscalls/clone/clone06.c ltp-full/testcases/kernel/syscalls/clone/clone07.c ltp-full/testcases/audit/syscalls/clone_test.c	OR.3	no	PC	i386 : int sys_clone (struct pt_regs regs);	clone.2.gz
close	close a file descriptor	no	ltp-full/testcases/kernel/syscalls/close/close01.c ltp-full/testcases/kernel/syscalls/close/close02.c ltp-full/testcases/kernel/syscalls/close/close08.c		no	FS	int close(int fd);	close.2.gz
connect	initiate a connection on a socket	no	ltp-full/testcases/kernel/syscalls/connect/connect01.c		no	IPC	int connect(int sockfd, const struct sockaddr *serv_addr, socklen_t	connect.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
creat	open and possibly create a file or device	yes	ltp-full/testcases/kernel/syscalls/creat/creat01.c ltp-full/testcases/kernel/syscalls/creat/creat03.c ltp-full/testcases/kernel/syscalls/creat/creat04.c ltp-full/testcases/kernel/syscalls/creat/creat05.c ltp-full/testcases/kernel/syscalls/creat/creat06.c ltp-full/testcases/kernel/syscalls/creat/creat07.c ltp-full/testcases/kernel/syscalls/creat/creat08.c ltp-full/testcases/kernel/syscalls/creat/creat09.c ltp-full/testcases/audit/syscalls/creat_test.c	DA.1, DA.2, DA.3, OR.1	addtl	FS	int creat(const char *pathname, mode_t mode);	creat.2.gz
delete_module	Delete a loadable module	yes	ltp-full/testcases/audit/syscalls/delete_module_test.c	TP.3	admin only	KM	long sys_delete_module (const char *name_user, unsigned int flags,	delete_module.2.gz
dup	duplicate a file descriptor	no	ltp-full/testcases/kernel/syscalls/dup/dup01.c ltp-full/testcases/kernel/syscalls/dup/dup02.c ltp-full/testcases/kernel/syscalls/dup/dup03.c ltp-full/testcases/kernel/syscalls/dup/dup04.c ltp-full/testcases/kernel/syscalls/dup/dup05.c ltp-full/testcases/kernel/syscalls/dup/dup06.c ltp-full/testcases/kernel/syscalls/dup/dup07.c		no	FS	int dup(int oldfd);	dup.2.gz
dup2	duplicate a file descriptor	no	ltp-full/testcases/kernel/syscalls/dup2/dup201.c ltp-full/testcases/kernel/syscalls/dup2/dup202.c ltp-full/testcases/kernel/syscalls/dup2/dup203.c ltp-full/testcases/kernel/syscalls/dup2/dup204.c ltp-full/testcases/kernel/syscalls/dup2/dup205.c		no	FS	int dup2(int oldfd, int newfd);	dup2.2.gz
epoll_create	open an epoll file descriptor	no			no	FS	int epoll_create(int size)	epoll_create.2.gz
epoll_ctl	control interface for an epoll descriptor	no			no	FS	int epoll_ctl(int epfd, int op, int fd, struct epoll_event *event)	epoll_ctl.2.gz
epoll_wait	wait for an I/O event on an epoll file descriptor	no			no	FS	int epoll_wait(int epfd, struct epoll_event * events, int maxevents, int timeout)	epoll_wait.2.gz
execve	execute program	yes	ltp-full/testcases/kernel/syscalls/execve/execve01.c ltp-full/testcases/kernel/syscalls/execve/execve02.c ltp-full/testcases/kernel/syscalls/execve/execve03.c ltp-full/testcases/kernel/syscalls/execve/execve04.c ltp-full/testcases/kernel/syscalls/execve/execve05.c ltp-full/testcases/kernel/syscalls/execve/execve06.c ltp-full/testcases/audit/syscalls/execve_test.c ltp-full/testcases/audit/inheritance/child_loginuid_test.c	DA.1, DA.2, DA.3, TP.4	addtl	PC	int execve(const char *filename, char *const argv [], char *const	execve.2.gz
exit	terminate the current process	no	ltp-full/testcases/kernel/syscalls/exit/exit01.c ltp-full/testcases/kernel/syscalls/exit/exit02.c		no	PC	void _exit(int status);	exit.2.gz
exit_group	Same as _exit(2), but kills all threads in the current thread group, not just the current thread.	no			no	PC	void sys_exit_group (int error_code);	sys_exit_group.2.gz
fadvise	Advise the system about the expected behaviour of the application with respect to the file associated with FD	no			no	MM	int sys_fadvise64_64(int fd, loff_t offset, loff_t len, int advice)	fadvise.2.gz
fchdir	change working directory	no	ltp-full/testcases/kernel/syscalls/fchdir/fchdir01.c ltp-full/testcases/kernel/syscalls/fchdir/fchdir02.c ltp-full/testcases/kernel/syscalls/fchir/fchdir03.c ltp-full/testcases/audit/syscalls/fchdir_test.c		no	FS	int fchdir(int fd);	fchdir.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
fchmod	change permissions of a file	yes	ltp-full/testcases/kernel/syscalls/fchmod/fchmod01.c ltp-full/testcases/kernel/syscalls/fchmod/fchmod02.c ltp-full/testcases/kernel/syscalls/fchmod/fchmod03.c ltp-full/testcases/kernel/syscalls/fchmod/fchmod04.c ltp-full/testcases/kernel/syscalls/fchmod/fchmod05.c ltp-full/testcases/kernel/syscalls/fchmod/fchmod06.c ltp-full/testcases/kernel/syscalls/fchmod/fchmod07.c ltp-full/testcases/audit/syscalls/fchmod_test.c	DA.1, DA.3, SM.2	addtl	FS	int fchmod(int fildes, mode_t mode);	fchmod.2.gz
fchown	change ownership of a file	yes	ltp-full/testcases/kernel/syscalls/fchown/fchown01.c ltp-full/testcases/kernel/syscalls/fchown/fchown02.c ltp-full/testcases/kernel/syscalls/fchown/fchown03.c ltp-full/testcases/kernel/syscalls/fchown/fchown04.c ltp-full/testcases/kernel/syscalls/fchown/fchown05.c ltp-full/testcases/audit/syscalls/fchown_test.c	DA.1, DA.3, SM.2	addtl	FS	int fchown(int fd, uid_t owner, gid_t group);	fchown.2.gz
fcntl	manipulate file descriptor	no			no	FS	int fcntl(int fd, int cmd);	fcntl.2.gz
fdatasync	synchronize a file's in-core data with that on disk	no	ltp-full/testcases/kernel/syscalls/fdatasync/fdatasync01.c ltp-full/testcases/kernel/syscalls/fdatasync/fdatasync02.c		no	FS	int fdatasync(int fd);	fdatasync.2.gz
fgetxattr	retrieve an extended attribute value	no			no	FS	ssize_t fgetxattr (int filedes, const char *name,	fgetxattr.2.gz
flistxattr	list extended attribute names	no			no	FS	ssize_t flistxattr (int filedes,	flistxattr.2.gz
flock	apply or remove an advisory lock on an open file	no	ltp-full/testcases/kernel/syscalls/flock/flock01.c ltp-full/testcases/kernel/syscalls/flock/flock02.c ltp-full/testcases/kernel/syscalls/flock/flock03.c ltp-full/testcases/kernel/syscalls/flock/flock04.c ltp-full/testcases/kernel/syscalls/flock/flock05.c ltp-full/testcases/kernel/syscalls/flock/flock06.c		no	FS	int flock(int fd, int operation);	flock.2.gz
fork	create a child process	yes	ltp-full/testcases/kernel/syscalls/fork/fork01.c ltp-full/testcases/kernel/syscalls/fork/fork02.c ltp-full/testcases/kernel/syscalls/fork/fork03.c ltp-full/testcases/kernel/syscalls/fork/fork04.c ltp-full/testcases/kernel/syscalls/fork/fork05.c ltp-full/testcases/kernel/syscalls/fork/fork06.c ltp-full/testcases/kernel/syscalls/fork/fork07.c ltp-full/testcases/kernel/syscalls/fork/fork08.c ltp-full/testcases/kernel/syscalls/fork/fork09.c ltp-full/testcases/audit/syscalls/fork_test.c	OR.3	no	PC	pid_t fork(void);	fork.2.gz
fremovexattr	remove an extended attribute	yes	ltp-full/testcases/kernel/fs/acls/acl_test01 ltp-full/testcases/audit/syscalls/fremovexattr_test.c	DA.1, DA.3, SM.2	addtl	FS	int fremovexattr (int filedes, const char *name);	fremovexattr.2.gz
fsetxattr	set an extended attribute value	yes	ltp-full/testcases/kernel/fs/acls/acl_test01 ltp-full/testcases/audit/syscalls/fsetxattr_test.c	DA.1, DA.3, SM.2	addtl	FS	int fsetxattr (int filedes, const char *name,	fsetxattr.2.gz
fstat	get file status	no	ltp-full/testcases/kernel/syscalls/fstat/fstat01.c ltp-full/testcases/kernel/syscalls/fstat/fstat02.c ltp-full/testcases/kernel/syscalls/fstat/fstat03.c ltp-full/testcases/kernel/syscalls/fstat/fstat04.c ltp-full/testcases/kernel/syscalls/fstat/fstat05.c		addtl	FS	int fstat(int filedes, struct stat *buf);	fstat.2.gz
fstatfs	get file system statistics	no	ltp-full/testcases/kernel/syscalls/fstatfs/fstatfs01.c ltp-full/testcases/kernel/syscalls/fstatfs/fstatfs02.c		no	FS	int fstatfs(int fd, struct statfs *buf);	fstatfs.2.gz
fsync	synchronize a file's complete in-core state with that on disk	no	ltp-full/testcases/kernel/syscalls/fsync/fsync01.c ltp-full/testcases/kernel/syscalls/fsync/fsync02.c ltp-full/testcases/kernel/syscalls/fsync/fsync03.c		no	FS	int fsync(int fd);	fsync.2.gz
ftruncate	truncate a file to a specified length	no	ltp-full/testcases/kernel/syscalls/ftruncate/ftruncate01.c ltp-full/testcases/kernel/syscalls/ftruncate/ftruncate02.c ltp-full/testcases/kernel/syscalls/ftruncate/ftruncate03.c ltp-full/testcases/audit/syscalls/ftruncate_test.c		no	FS	int ftruncate(int fd, off_t length);	ftruncate.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
futex	Fast Userspace Locking system call	no			no	PC	int sys_futex (void *futex, int op, int val, const struct timespec	futex.2.gz
get_thread_area	get a Thread Local Storage (TLS) area	no			no	PC	int get_thread_area (struct user_desc *u_info);	get_thread_area.2.gz
getcwd	get current working directory	no	ltp-full/testcases/kernel/syscalls/getcwd/getcwd01.c ltp-full/testcases/kernel/syscalls/getcwd/getcwd02.c ltp-full/testcases/kernel/syscalls/getcwd/getcwd03.c		no	FS	char *getcwd(char *buf, size_t size)	sys_getcwd.2.gz
getdents	get directory entries	no	ltp-full/testcases/kernel/syscalls/getdents/getdents02.c ltp-full/testcases/kernel/syscalls/getdents/getdents03.c ltp-full/testcases/kernel/syscalls/getdents/getdents04.c		no	FS	int getdents(unsigned int fd, struct dirent *dirp, unsigned int count);	getdents.2.gz
getegid	get group identity	no	ltp-full/testcases/kernel/syscalls/getegid/getegid01.c		no	PC	gid_t getegid(void);	getegid.2.gz
geteuid	get user identity	no	ltp-full/testcases/kernel/syscalls/geteuid/geteuid01.c		no	PC	uid_t geteuid(void);	geteuid.2.gz
getgid	get group identity	no	ltp-full/testcases/kernel/syscalls/getgid/getgid01.c ltp-full/testcases/kernel/syscalls/getgid/getgid02.c ltp-full/testcases/kernel/syscalls/getgid/getgid03.c		no	PC	gid_t getgid(void);	getgid.2.gz
getgroups	get/set list of supplementary group IDs	no	ltp-full/testcases/kernel/syscalls/getgroups/getgroups01.c ltp-full/testcases/kernel/syscalls/getgroups/getgroups02.c ltp-full/testcases/kernel/syscalls/getgroups/getgroups04.c		no	PC	int getgroups(int size, gid_t list[]);	getgroups.2.gz
getitimer	get or set value of an interval timer	no	ltp-full/testcases/kernel/syscalls/getitimer/getitimer01.c ltp-full/testcases/kernel/syscalls/getitimer/getitimer02.c ltp-full/testcases/kernel/syscalls/getitimer/getitimer03.c		no	PC	int getitimer(int which, struct itimerval *value);	getitimer.2.gz
get_mempolicy	retrieves the NUMA policy of the current process or an memory address	no			no	MM	int get_mempolicy(int *policy, unsigned long *nodemask, unsigned long maxnode, unsigned long addr, unsigned long flags);	get_mempolicy.2.gz
getpeername	get name of connected peer socket	no	ltp-full/testcases/kernel/syscalls/getpeername/getpeername01.c		no	PC	int getpeername(int s, struct sockaddr *name, socklen_t *namelen);	getpeername.2.gz
getpgid	set/get process group	no	ltp-full/testcases/kernel/syscalls/getpgid/getpgid01.c ltp-full/testcases/kernel/syscalls/getpgid/getpgid02.c		no	PC	pid_t getpgid(pid_t pid);	getpgid.2.gz
getpgrp	set/get process group	no	ltp-full/testcases/kernel/syscalls/getpgrp/getpgrp01.c		no	PC	pid_t getpgrp(void);	getpgrp.2.gz
getpid	get process identification	no	ltp-full/testcases/kernel/syscalls/getpid/getpid01.c ltp-full/testcases/kernel/syscalls/getpid/getpid02.c		no	PC	pid_t getpid(void);	getpid.2.gz
getppid	get process identification	no	ltp-full/testcases/kernel/syscalls/getppid/getppid01.c ltp-full/testcases/kernel/syscalls/getppid/getppid02.c		no	PC	pid_t getppid(void);	getppid.2.gz
getpriority	get/set program scheduling priority	no			no	PC	int getpriority(int which, int who);	getpriority.2.gz
getresgid	get real, effective and saved user or group ID	no	ltp-full/testcases/kernel/syscalls/getresgid/getresgid01.c ltp-full/testcases/kernel/syscalls/getresgid/getresgid02.c ltp-full/testcases/kernel/syscalls/getresgid/getresgid03.c		no	PC	int getresgid(gid_t *rgid, gid_t *egid, gid_t *sgid);	getresgid.2.gz
getresuid	get real, effective and saved user or group ID	no	ltp-full/testcases/kernel/syscalls/getresuid/getresuid01.c ltp-full/testcases/kernel/syscalls/getresuid/getresuid02.c ltp-full/testcases/kernel/syscalls/getresuid/getresuid03.c		no	PC	int getresuid(uid_t *ruid, uid_t *euid, uid_t *suid);	getresuid.2.gz
getrlimit	get/set resource limits and usage	no	ltp-full/testcases/kernel/syscalls/getrlimit/getrlimit01.c ltp-full/testcases/kernel/syscalls/getrlimit/getrlimit02.c		no	PC	int getrlimit(int resource, struct rlimit *rlim);	getrlimit.2.gz
getrusage	get/set resource limits and usage	no	ltp-full/testcases/kernel/syscalls/getrusage/getrusage01.c ltp-full/testcases/kernel/syscalls/getrusage/getrusage02.c		no	PC	int getrusage(int who, struct rusage *usage);	getrusage.2.gz
getsid	get session ID	no	ltp-full/testcases/kernel/syscalls/getsid/getsid01.c ltp-full/testcases/kernel/syscalls/getsid/getsid02.c		no	PC	pid_t getsid(pid_t pid);	getsid.2.gz
getsockname	get socket name	no	ltp-full/testcases/kernel/syscalls/getsockname/getsockname01.c		no	IPC	int getsockname(int s, struct sockaddr *name, socklen_t *namelen);	getsockname.2.gz
getsockopt	get and set options on sockets	no	ltp-full/testcases/kernel/syscalls/getsockopt/getsockopt01.c		no	IPC	int getsockopt(int s, int level, int optname, void *optval, socklen_t	getsockopt.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
gettid	get thread identification	no			no	PC	pid_t gettid(void);	gettid.2.gz
gettimeofday	get / set time	no	ltp-full/testcases/kernel/syscalls/gettimeofday/gettimeofday01.c ltp-full/testcases/kernel/syscalls/gettimeoday/gettimeofday02.c		no	PC	int gettimeofday(struct timeval *tv, struct timezone *tz);	gettimeofday.2.gz
getuid	get user identity	no	ltp-full/testcases/kernel/syscalls/getuid/getuid01.c ltp-full/testcases/kernel/syscalls/getuid/getuid02.c ltp-full/testcases/kernel/syscalls/getuid/getuid03.c		no	PC	uid_t getuid(void);	getuid.2.gz
getxattr	retrieve an extended attribute value	no			no	FS	ssize_t getxattr (const char *path, const char *name,	getxattr.2.gz
init_module	Initialize a loadable module entry	yes	ltp-full/testcases/audit/syscalls/init_module_test.c	TP.3	admin only	KM	long sys_init_module (void *umod, unsigned long len,	init_module.2.gz
io_cancel	Cancel an outstanding asynchronous I/O operation	no			no	FS	long io_cancel (aio_context_t ctx_id, struct iocb *iocb,	io_cancel.2.gz
io_destroy	Destroy an asynchronous I/O context	no			no	FS	long io_destroy (aio_context_t ctx);	io_destroy.2.gz
io_getevents	Read asynchronous I/O events from the completion queue	no			no	FS	long io_getevents (aio_context_t ctx_id, long min_nr, long nr,	io_getevents.2.gz
io_setup	Create an asynchronous I/O context	no			no	FS	long io_setup (unsigned nr_events, aio_context_t *ctxp);	io_setup.2.gz
io_submit	Submit asynchronous I/O blocks for processing	no			no	FS	long io_submit (aio_context_t ctx_id, long nr, struct iocb **iocbpp);	io_submit.2.gz
ioctl	control device	yes	ltp-full/testcases/kernel/syscalls/sockioctl/sockioctl101.c ltp-full/testcases/audit/syscalls/ioctl_test.c	AU.1	no	FS	int ioctl(int d, int request, ...);	ioctl.2.gz
ioperm	set port input/output permissions	yes	ltp-full/testcases/kernel/syscalls/ioperm/ioperm01.c ltp-full/testcases/kernel/syscalls/ioperm/ioperm02.c ltp-full/testcases/audit/syscalls/ioperm_test.c	TP.2	admin only	PC	int ioperm(unsigned long from, unsigned long num, int turn_on);	ioperm.2.gz
iopl	change I/O privilege level	yes	ltp-full/testcases/kernel/syscalls/iopl/iopl01.c ltp-full/testcases/kernel/syscalls/iopl/iopl02.c ltp-full/testcases/audit/syscalls/iopl_test.c	TP.2	admin only	PC	int iopl(int level);	iopl.2.gz
ipc	System V IPC system calls	yes	ltp-full/testcases/audit/syscalls/msgctl_test.c ltp-full/testcases/audit/syscalls/msgget_test.c ltp-full/testcases/audit/syscalls/msgrecv_test.c ltp-full/testcases/audit/syscalls/msgsend_test.c ltp-full/testcases/audit/syscalls/semctl_test.c ltp-full/testcases/audit/syscalls/semget_test.c ltp-full/testcases/audit/syscalls/semop_test.c ltp-full/testcases/audit/syscalls/shmat_test.c ltp-full/testcases/audit/syscalls/shmctl_test.c ltp-full/testcases/audit/syscalls/shmdt_test.c ltp-full/testcases/audit/syscalls/shmget_test.c	see comment1	see comment1	IPC	int ipc(unsigned int call, int first, int second, int third, void *ptr,	ipc.2.gz
kexec_load	Execute new kernel	no			admin only	SI	long kexec_load(unsigned long entry, unsigned long nr_segments, struct kexec_segment *segments, unsigned long flags);	sys_kexec_load.2.gz
kill	send signal to a process	no	ltp-full/testcases/kernel/syscalls/kill/kill01.c ltp-full/testcases/kernel/syscalls/kill/kill02.c ltp-full/testcases/kernel/syscalls/kill/kill03.c ltp-full/testcases/kernel/syscalls/kill/kill04.c ltp-full/testcases/kernel/syscalls/kill/kill05.c ltp-full/testcases/kernel/syscalls/kill/kill06.c ltp-full/testcases/kernel/syscalls/kill/kill07.c ltp-full/testcases/kernel/syscalls/kill/kill08.c ltp-full/testcases/kernel/syscalls/kill/kill09.c ltp-full/testcases/audit/syscalls/kill_test.c		addtl	PC	int kill(pid_t pid, int sig);	kill.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
lchown	change ownership of a file	yes	ltp-full/testcases/kernel/syscalls/lchown/lchown01.c ltp-full/testcases/kernel/syscalls/lchown/lchown02.c ltp-full/testcases/audit/syscalls/lchown_test.c	DA.1, DA.3, SM.2	addtl	FS	int lchown(const char *path, uid_t owner, gid_t group);	lchown.2.gz
lgetxattr	retrieve an extended attribute value	no			no	FS	ssize_t lgetxattr (const char *path, const char *name,	lgetxattr.2.gz
link	make a new name for a file	yes	ltp-full/testcases/kernel/syscalls/link/link01.c ltp-full/testcases/kernel/syscalls/link/link02.c ltp-full/testcases/kernel/syscalls/link/link03.c ltp-full/testcases/kernel/syscalls/link/link04.c ltp-full/testcases/kernel/syscalls/link/link05.c ltp-full/testcases/kernel/syscalls/link/link06.c ltp-full/testcases/kernel/syscalls/link/link07.c ltp-full/testcases/audit/syscalls/link_test.c	DA.1, DA.3	addtl	FS	int link(const char *oldpath, const char *newpath);	link.2.gz
listen	listen for connections on a socket	no	ltp-full/testcases/kernel/syscalls/listen/listen01.c		no	IPC	int listen(int s, int backlog);	listen.2.gz
listxattr	list extended attribute names	no			no	FS	ssize_t listxattr (const char *path,	listxattr.2.gz
llistxattr	list extended attribute names	no			no	FS	ssize_t llistxattr (const char *path,	llistxattr.2.gz
llseek	reposition read/write file offset	no			no	FS	int _llseek(unsigned int fd, unsigned long offset_high, unsigned long	llseek.2.gz
lookup_dcookie	return a directory entry's path	no			admin only	FS	int lookup_dcookie(u64 cookie, char * buffer, size_t len);	lookup_dcookie.2.gz
lremovexattr	remove an extended attribute	yes	ltp-full/testcases/kernel/fs/acls/acl_test01 ltp-full/testcases/audit/syscalls/lremovexattr_test.c	DA.1, DA.3, SM.2	addtl	FS	int lremovexattr (const char *path, const char *name);	lremovexattr.2.gz
lseek	reposition read/write file offset	no	ltp-full/testcases/kernel/syscalls/lseek/lseek01.c ltp-full/testcases/kernel/syscalls/lseek/lseek02.c		no	FS	off_t lseek(int fildes, off_t offset, int whence);	lseek.2.gz
lsetxattr	set an extended attribute value	yes	ltp-full/testcases/kernel/fs/acls/acl_test01 ltp-full/testcases/audit/syscalls/lsetxattr_test.c	DA.1, DA.3, SM.2	addtl	FS	int lsetxattr (const char *path, const char *name,	lsetxattr.2.gz
lstat	get file status	no	ltp-full/testcases/kernel/syscalls/lstat/lstat01.c ltp-full/testcases/kernel/syscalls/lstat/lstat02.c ltp-full/testcases/kernel/syscalls/lstat/lstat03.c		addtl	FS	int lstat(const char *file_name, struct stat *buf);	lstat.2.gz
madvise	give advice about use of memory	no			no	MM	int madvise(void *start, size_t length, int advice);	madvise.2.gz
mbind	set memory policy for a memory range	no			no	MM	int mbind(void *start, unsigned long len, int policy, unsigned long *nodemask, unsigned long maxnode,unsigned flags);	mbind.2.gz
mincore	get information on whether pages are in core	no			no	MM	int mincore(void *start, size_t length, unsigned char *vec);	mincore.2.gz
mkdir	create a directory	yes	ltp-full/testcases/kernel/syscalls/mkdir/mkdir01.c ltp-full/testcases/kernel/syscalls/mkdir/mkdir02.c ltp-full/testcases/kernel/syscalls/mkdir/mkdir03.c ltp-full/testcases/kernel/syscalls/mkdir/mkdir04.c ltp-full/testcases/kernel/syscalls/mkdir/mkdir05.c ltp-full/testcases/kernel/syscalls/mkdir/mkdir08.c ltp-full/testcases/kernel/syscalls/mkdir/mkdir09.c ltp-full/testcases/audit/syscalls/mkdir_test.c	DA.1, DA.3, OR.1	addtl	FS	int mkdir(const char *pathname, mode_t mode);	mkdir.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
mknod	create a special or ordinary file	yes	ltp-full/testcases/kernel/syscalls/mknod/mknod01.c ltp-full/testcases/kernel/syscalls/mknod/mknod02.c ltp-full/testcases/kernel/syscalls/mknod/mknod03.c ltp-full/testcases/kernel/syscalls/mknod/mknod04.c ltp-full/testcases/kernel/syscalls/mknod/mknod05.c ltp-full/testcases/kernel/syscalls/mknod/mknod06.c ltp-full/testcases/kernel/syscalls/mknod/mknod07.c ltp-full/testcases/kernel/syscalls/mknod/mknod08.c ltp-full/testcases/kernel/syscalls/mknod/mknod09.c ltp-full/testcases/audit/syscalls/mknod_test.c	DA.1, DA.3	addtl	FS	int mknod(const char *pathname, mode_t mode, dev_t dev);	mknod.2.gz
mlock	disable paging for some parts of memory	no	ltp-full/testcases/kernel/syscalls/mlock/mlock01.c ltp-full/testcases/kernel/syscalls/mlock/mlock02.c		admin only	MM	int mlock(const void *addr, size_t len);	mlock.2.gz
mlockall	disable paging for calling process	no	ltp-full/testcases/kernel/syscalls/mlockall/mlockall01.c ltp-full/testcases/kernel/syscalls/mlockall/mlockall02.c		admin only	MM	int mlockall(int flags);	mlockall.2.gz
mmap	map or unmap files or devices into memory	no	ltp-full/testcases/kernel/syscalls/mmap/mmap01.c ltp-full/testcases/kernel/syscalls/mmap/mmap02.c ltp-full/testcases/kernel/syscalls/mmap/mmap03.c ltp-full/testcases/kernel/syscalls/mmap/mmap04.c ltp-full/testcases/kernel/syscalls/mmap/mmap05.c ltp-full/testcases/kernel/syscalls/mmap/mmap06.c ltp-full/testcases/kernel/syscalls/mmap/mmap07.c ltp-full/testcases/kernel/syscalls/mmap/mmap08.c ltp-full/testcases/kernel/syscalls/mmap/mmap09.c		no	FS	void * mmap(void *start, size_t length, int prot , int flags, int fd,	mmap.2.gz
mmap2	map files or devices into memory	no			no	FS	void * mmap2(void *start, size_t length, int prot, int flags, int fd, off_t offset);	mmap2.2.gz
modify_ldt	get or set ldt	no	ltp-full/testcases/kernel/syscalls/modify_ldt/modify_ldt01.c ltp-full/testcases/kernel/syscalls/modify_ldt/modify_ldt02.c		no	PC	int modify_ldt(int func, void *ptr, unsigned long bytecount);	modify_ldt.2.gz
mount	mount and unmount filesystems	yes	ltp-full/testcases/kernel/syscalls/mount/mount01.c ltp-full/testcases/kernel/syscalls/mount/mount02.c ltp-full/testcases/kernel/syscalls/mount/mount03.c ltp-full/testcases/kernel/syscalls/mount/mount04.c ltp-full/testcases/audit/syscalls/mount_test.c mount01,02,03,04 run in manual test, not by LTP.	DA.1, DA.3	admin only	FS	int mount(const char *source, const char *target, const char *filesystemtype, unsigned long mountflags, const void *data);	mount.2.gz
mprotect	control allowable accesses to a region of memory	no	ltp-full/testcases/kernel/syscalls/mprotect/mprotect01.c ltp-full/testcases/kernel/syscalls/mprotect/mprotect02.c ltp-full/testcases/kernel/syscalls/mprotect/mprotect03.c		no	MM	int mprotect(const void *addr, size_t len, int prot);	mprotect.2.gz
mq_getsetattr	get/set message queue attributes	no			no	IPC	int mq_getsetattr(mqd_t mqdes, const struct mq_attr __user *mqstat, struct mq_attr __user *omqstat);	mq_getattr.3p.gz/mq_setattr.3p.gz
mq_notify	notify process that a message is available	no			no	IPC	int mq_notify(mqd_t mqdes, const struct sigevent *notification);	mq_notify.3p.gz
mq_open	open a message queue	no			no	IPC	mqd_t mq_open(const char *name, int oflag, ...);	mq_open.3p.gz
mq_timedreceive	receive a message from a message queue	no			no	IPC	ssize_t mq_timedreceive(mqd_t mqdes, char *restrict msg_ptr,size_t msg_len, unsigned *restrict msg_prio, const struct timespec *restrict abs_timeout);	mq_timedreceive.3p.gz
mq_timedsend	send a message to a message queue	no			no	IPC	int mq_timedsend(mqd_t mqdes, const char *msg_ptr, size_t msg_len, unsigned msg_prio, const struct timespec *abs_timeout);	mq_timedsend.3p.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
mq_unlink	remove a message queue	no			no	IPC	int mq_unlink(const char *name);	mq_unlink.3p.gz
mremap	Increase or decrease an existing memory mapping	no	ltp-full/testcases/kernel/syscalls/mremap/mremap01.c ltp-full/testcases/kernel/syscalls/mremap/mremap02.c ltp-full/testcases/kernel/syscalls/mremap/mremap03.c ltp-full/testcases/kernel/syscalls/mremap/mremap04.c		no	MM	unsigned long sys_mremap (unsigned long addr, unsigned long old_len,	mremap.2.gz
msgctl	message control operations	yes	ltp-full/testcases/kernel/syscalls/ipc/msgctl/msgctl01.c ltp-full/testcases/kernel/syscalls/ipc/msgctl/msgctl02.c ltp-full/testcases/kernel/syscalls/ipc/msgctl/msgctl03.c ltp-full/testcases/kernel/syscalls/ipc/msgctl/msgctl04.c ltp-full/testcases/kernel/syscalls/ipc/msgctl/msgctl05.c ltp-full/testcases/kernel/syscalls/ipc/msgctl/msgctl06.c ltp-full/testcases/kernel/syscalls/ipc/msgctl/msgctl07.c ltp-full/testcases/kernel/syscalls/ipc/msgctl/msgctl08.c ltp-full/testcases/kernel/syscalls/ipc/msgctl/msgctl09.c ltp-full/testcases/audit/syscalls/msgctl_test.c	DA.1, DA.4, SM.2	addtl	IPC	int msgctl(int msqid, int cmd, struct msqid_ds *buf);	msgctl.2.gz
msgget	get a message queue identifier	yes	ltp-full/testcases/kernel/syscalls/ipc/msgget/msgget01.c ltp-full/testcases/kernel/syscalls/ipc/msgget/msgget02.c ltp-full/testcases/kernel/syscalls/ipc/msgget/msgget03.c ltp-full/testcases/kernel/syscalls/ipc/msgget/msgget04.c ltp-full/testcases/audit/syscalls/msgget_test.c	DA.1, DA.4, SM.2, OR.2	addtl	IPC	int msgget(key_t key, int msgflg);	msgget.2.gz
msgrcv	message operations	yes	ltp-full/testcases/kernel/syscalls/ipc/msgrcv/msgrcv01.c ltp-full/testcases/kernel/syscalls/ipc/msgrcv/msgrcv02.c ltp-full/testcases/kernel/syscalls/ipc/msgrcv/msgrcv03.c ltp-full/testcases/kernel/syscalls/ipc/msgrcv/msgrcv04.c ltp-full/testcases/kernel/syscalls/ipc/msgrcv/msgrcv05.c ltp-full/testcases/kernel/syscalls/ipc/msgrcv/msgrcv06.c ltp-full/testcases/audit/syscalls/msgrecv_test.c	DA.1, DA.4	no	IPC	ssize_t msgrcv(int msqid, struct msgbuf *msgp, size_t msgsz, long msg-	msgrcv.2.gz
msgsnd	message operations	yes	ltp-full/testcases/kernel/syscalls/ipc/msgsmd/msgsnd01.c ltp-full/testcases/kernel/syscalls/ipc/msgsmd/msgsnd02.c ltp-full/testcases/kernel/syscalls/ipc/msgsmd/msgsnd03.c ltp-full/testcases/kernel/syscalls/ipc/msgsmd/msgsnd04.c ltp-full/testcases/kernel/syscalls/ipc/msgsmd/msgsnd05.c ltp-full/testcases/kernel/syscalls/ipc/msgsmd/msgsnd06.c ltp-full/testcases/audit/syscalls/msgsend_test.c	DA.1, DA.4	addtl	IPC	int msgsnd(int msqid, struct msgbuf *msgp, size_t msgsz, int msgflg);	msgsnd.2.gz
msync	synchronize a file with a memory map	no	ltp-full/testcases/kernel/syscalls/msync/msync01.c ltp-full/testcases/kernel/syscalls/msync/msync02.c ltp-full/testcases/kernel/syscalls/msync/msync03.c ltp-full/testcases/kernel/syscalls/msyncmsync04.c ltp-full/testcases/kernel/syscalls/msync/msync05.c		no	FS	int msync(void *start, size_t length, int flags);	msync.2.gz
munlock	reenable paging for some parts of memory	no	ltp-full/testcases/kernel/syscalls/munlock/munlock01.c ltp-full/testcases/kernel/syscalls/munlock/munlock02.c		admin only	MM	int munlock(const void *addr, size_t len);	munlock.2.gz
munlockall	reenable paging for calling process	no	ltp-full/testcases/kernel/syscalls/munlockall/munlockall01.c ltp-full/testcases/kernel/syscalls/munlockall/munlockall02.c		admin only	MM	int munlockall(void);	munlockall.2.gz
munmap	map or unmap files or devices into memory	no			no	FS	int munmap(void *start, size_t length);	munmap.2.gz
nanosleep	pause execution for a specified time	no	ltp-full/testcases/kernel/syscalls/nanosleep/nanosleep01.c ltp-full/testcases/kernel/syscalls/nanosleep/nanosleep03.c ltp-full/testcases/kernel/syscalls/nanosleep/nanosleep04.c		no	PC	int nanosleep(const struct timespec *req, struct timespec *rem);	nanosleep.2.gz
nfsservctl	system call interface to the kernel NFS daemon	no			no	KM	long sys_nfsservctl (int cmd, struct nfsctl_arg, void *res);	nfsservctl.2.gz
nice	change process priority	no	ltp-full/testcases/kernel/syscalls/nice/nice01.c ltp-full/testcases/kernel/syscalls/nice/nice02.c ltp-full/testcases/kernel/syscalls/nice/nice03.c ltp-full/testcases/kernel/syscalls/nice/nice04.c ltp-full/testcases/kernel/syscalls/nice/nice05.c		addtl	PC	int nice(int inc);	nice.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
open	open and possibly create a file or device	yes	ltp-full/testcases/kernel/syscalls/open/open01.c ltp-full/testcases/kernel/syscalls/open/open02.c ltp-full/testcases/kernel/syscalls/open/open03.c ltp-full/testcases/kernel/syscalls/open/open04.c ltp-full/testcases/kernel/syscalls/open/open05.c ltp-full/testcases/kernel/syscalls/open/open06.c ltp-full/testcases/kernel/syscalls/open/open07.c ltp-full/testcases/kernel/syscalls/open/open08.c ltp-full/testcases/kernel/syscalls/open/open09.c ltp-full/testcases/kernel/syscalls/open/open10.c ltp-full/testcases/audit/syscalls/open_test.c	DA.1, DA.2, DA.3, OR.1	addtl	FS	int open(const char *pathname, int flags);	open.2.gz
pause	wait for signal	no	ltp-full/testcases/kernel/syscalls/pause/pause01.c ltp-full/testcases/kernel/syscalls/pause/pause02.c ltp-full/testcases/kernel/syscalls/pause/pause03.c		no	PC	int pause(void);	pause.2.gz
pciconfig_iobase	pci device information handling.	no			no	FS	int pciconfig_iobase(long which, unsigned long bus,	pciconfig_iobase.2.gz
pciconfig_read	pci device information handling.	no			admin only	FS	int pciconfig_read(unsigned long bus, unsigned long dfn,	pciconfig_read.2.gz
pciconfig_write	pci device information handling.	no			admin only	FS	int pciconfig_write(unsigned long bus, unsigned long dfn,	pciconfig_write.2.gz
personality	set the process execution domain	no			no	PC	long personality(unsigned long persona);	personality.2.gz
pipe	create pipe	no	ltp-full/testcases/kernel/syscalls/pipe/pipe01.c ltp-full/testcases/kernel/syscalls/pipe/pipe02.c ltp-full/testcases/kernel/syscalls/pipe/pipe03.c ltp-full/testcases/kernel/syscalls/pipe/pipe04.c ltp-full/testcases/kernel/syscalls/pipe/pipe05.c ltp-full/testcases/kernel/syscalls/pipe/pipe06.c ltp-full/testcases/kernel/syscalls/pipe/pipe07.c ltp-full/testcases/kernel/syscalls/pipe/pipe08.c ltp-full/testcases/kernel/syscalls/pipe/pipe09.c		no	FS/IPC	int pipe(int filedes[2]);	pipe.2.gz
pivot_root	change the root file system	no			admin only	FS	int pivot_root(const char *new_root, const char *put_old);	pivot_root.2.gz
poll	wait for some event on a file descriptor	no	ltp-full/testcases/kernel/syscalls/poll/poll01.c		no	FS	int poll(struct pollfd *ufds, unsigned int nfds, int timeout);	poll.2.gz
prctl	operations on a process	no	ltp-full/testcases/kernel/syscalls/prctl/prctl01.c ltp-full/testcases/kernel/syscalls/prctl/prctl02.c		no	PC	int prctl(int option, unsigned long arg2, unsigned long arg3 , unsigned	prctl.2.gz
pread	read from or write to a file descriptor at a given offset	no	ltp-full/testcases/kernel/syscalls/pread/pread01.c ltp-full/testcases/kernel/syscalls/pread/pread02.c ltp-full/testcases/kernel/syscalls/pread/pread03.c		no	FS	ssize_t pread(int fd, void *buf, size_t count, off_t offset);	pread.2.gz
ptrace	process trace	yes	ltp-full/testcases/kernel/syscalls/ptrace/ptrace01.c ltp-full/testcases/kernel/syscalls/ptrace/ptrace02.c ltp-full/testcases/kernel/syscalls/ptrace/ptrace03.c ltp-full/testcases/audit/syscalls/ptrace_test.c	TP.4	addtl	PC	long ptrace(enum __ptrace_request request, pid_t pid, void *addr, void	ptrace.2.gz
pwrite	read from or write to a file descriptor at a given offset	no	ltp-full/testcases/kernel/syscalls/pwrite/pwrite01.c ltp-full/testcases/kernel/syscalls/pwrite/pwrite02.c ltp-full/testcases/kernel/syscalls/pwrite/pwrite03.c ltp-full/testcases/kernel/syscalls/pwrite/pwrite04.c		no	FS	ssize_t pwrite(int fd, const void *buf, size_t count, off_t offset);	pwrite.2.gz
quotactl	Manipulate disk quotas	no			addtl	FS	int sys_quotactl (unsigned int cmd, const char *special, qid_t id,	quotactl.2.gz
read	read from a file descriptor	no	ltp-full/testcases/kernel/syscalls/read/read01.c ltp-full/testcases/kernel/syscalls/read/read02.c ltp-full/testcases/kernel/syscalls/read/read03.c ltp-full/testcases/kernel/syscalls/read/read04.c		no	FS	ssize_t read(int fd, void *buf, size_t count);	read.2.gz
readahead	Read in advance one or more pages of a file within a page	no			no	FS	ssize_t sys_readahead (int fd, loff_t offset, size_t count);	readahead.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
readdir	read directory entry	no	ltp-full/testcases/kernel/syscalls/readdir/readdir01.c ltp-full/testcases/kernel/syscalls/readdir/readdir02.c		no	FS	int readdir(unsigned int fd, struct dirent *dirp, unsigned int count);	readdir.2.gz
readlink	read value of a symbolic link	no	ltp-full/testcases/kernel/syscalls/readlink/readlink01.c ltp-full/testcases/kernel/syscalls/readlink/readlink02.c ltp-full/testcases/kernel/syscalls/readlink/readlink03.c ltp-full/testcases/kernel/syscalls/readlink/readlink04.c		no	FS	int readlink(const char *path, char *buf, size_t bufsiz);	readlink.2.gz
readv	read or write data into multiple buffers	no	ltp-full/testcases/kernel/syscalls/readv/readv01.c ltp-full/testcases/kernel/syscalls/readv/readv02.c ltp-full/testcases/kernel/syscalls/readv/readv03.c		no	FS	ssize_t readv(int fd, const struct iovec *vector, int count);	readv.2.gz
reboot	reboot or enable/disable Ctrl-Alt-Del	no	ltp-full/testcases/kernel/syscalls/reboot/reboot01.c ltp-full/testcases/kernel/syscalls/reboot/reboot02.c ltp-full/testcases/audit/syscalls/reboot_test.c		admin only	PC	int reboot(int magic, int magic2, int flag, void *arg);	reboot.2.gz
recv	receive a message from a socket	no	ltp-full/testcases/kernel/syscalls/recv/recv01.c		no	IPC	ssize_t recv(int s, void *buf, size_t len, int flags);	recv.2.gz
recvfrom	receive a message from a socket	no	ltp-full/testcases/kernel/syscalls/recvfrom/recvfrom01.c		no	IPC	ssize_t recvfrom(int s, void *buf, size_t len, int flags, struct sock-	recvfrom.2.gz
recvmsg	receive a message from a socket	no	ltp-full/testcases/kernel/syscalls/recvmsg/recvmsg01.c		no	IPC	ssize_t recvmsg(int s, struct msghdr *msg, int flags);	recvmsg.2.gz
remap_file_pages	create a non-linear file mapping	no			no	FS	void *remap_file_pages(void *start, size_t size, int prot, ssize_t	remap_file_pages.2.gz
removexattr	remove an extended attribute	yes	misc_test/ext3_ACLs/acl-tests/misc.test ltp-full/testcases/audit/syscalls/removexattr_test.c	DA.1, DA.3, SM.2	addtl	FS	int removexattr (const char *path, const char *name);	removexattr.2.gz
rename	change the name or location of a file	yes	ltp-full/testcases/kernel/syscalls/rename/rename01.c ltp-full/testcases/kernel/syscalls/rename/rename02.c ltp-full/testcases/kernel/syscalls/rename/rename03.c ltp-full/testcases/kernel/syscalls/rename/rename04.c ltp-full/testcases/kernel/syscalls/rename/rename05.c ltp-full/testcases/kernel/syscalls/rename/rename06.c ltp-full/testcases/kernel/syscalls/rename/rename07.c ltp-full/testcases/kernel/syscalls/rename/rename08.c ltp-full/testcases/kernel/syscalls/rename/rename09.c ltp-full/testcases/kernel/syscalls/rename/rename10.c ltp-full/testcases/kernel/syscalls/rename/rename12.c ltp-full/testcases/kernel/syscalls/rename/rename13.c ltp-full/testcases/kernel/syscalls/rename/rename14.c ltp-full/testcases/audit/syscalls/rename_test.c	DA.1, DA.3	addtl	FS	int rename(const char *oldpath, const char *newpath);	rename.2.gz
restart_syscall	Restart a system call	no			no	PC	long sys_restart_syscall(void);	sys_restart_syscall.2.gz
rmdir	delete a directory	yes	ltp-full/testcases/kernel/syscalls/rmdir/rmdir01.c ltp-full/testcases/kernel/syscalls/rmdir/rmdir02.c ltp-full/testcases/kernel/syscalls/rmdir/rmdir03.c ltp-full/testcases/kernel/syscalls/rmdir/rmdir04.c ltp-full/testcases/kernel/syscalls/rmdir/rmdir05.c ltp-full/testcases/audit/syscalls/rmdir_test.c	DA.1, DA.3	addtl	FS	int rmdir(const char *pathname);	rmdir.2.gz
rt_sigaction	Alter an action taken by a process	no			no	IPC	long sys_rt_sigaction (int sig, const struct sigaction *act,	sys_rt_sigaction.2.gz
rt_sigpending	Examine a pending signal that has been raised while blocked	no			no	IPC	long sys_rt_sigpending (sigset_t *set, size_t sigsetsize);	sys_rt_sigpending.2.gz
rt_sigprocmask	Change the list of currently blocked signals	no			no	IPC	long sys_rt_sigprocmask (sigset_t *set, sigset_t *oset,	sys_rt_sigprocmask.2.gz
rt_sigqueueinfo	Send signal information to a signal	no			no	IPC	long sys_rt_sigqueueinfo (int pid, int sig, siginfo_t *uinfo);	sys_rt_sigqueueinfo.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
rt_sigreturn	Return from a signal handler and clean up the stack	no			no	IPC	int sys_rt_sigreturn (unsigned long __unused); xSeries	sys_rt_sigreturn.2.gz
rt_sigsuspend	Replace the signal mask for a value with the unewset	no			no	IPC	long sys_rt_sigsuspend (sigset_t *unewset, size_t sigsetsize,	sys_rt_sigsuspend.2.gz
rt_sigtimedwait	Synchronously wait for queued signals specified in	no			no	IPC	long sys_rt_sigtimedwait (const sigset_t *uthese, siginfo_t *uinfo,	sys_rt_sigtimedwait.2.gz
rtas	Allows userspace to call RTAS (Run Time Abstraction Services)	no			admin only	FS	int ppc_rtas (struct rtas_args *uargs);	sys_rtas.2.gz
sched_get_priority_max	get static priority	no	ltp-full/testcases/kernel/syscalls/sched_get_priority_max/sched_get_priority_max01.c ltp-full/testcases/kernel/syscalls/sched_get_priority_max/sched_get_priority_max02.c		no	PC	int sched_get_priority_max(int policy);	sched_get_priority_max.2.gz
sched_get_priority_min	get static priority	no	ltp-full/testcases/kernel/syscalls/sched_get_priority_min/sched_get_priority_min01.c ltp-full/testcases/kernel/syscalls/sched_get_priority_min/sched_get_priority_min02.c		no	PC	int sched_get_priority_min(int policy);	sched_get_priority_min.2.gz
sched_getaffinity	set and get a process's CPU	no			no	PC	int sched_getaffinity(pid_t pid, unsigned int len, unsigned long	sched_getaffinity.2.gz
sched_getparam	set and get scheduling parameters	no	ltp-full/testcases/kernel/syscalls/sched_getparam/sched_getparam01.c ltp-full/testcases/kernel/syscalls/sched_getparam/sched_getparam02.c ltp-full/testcases/kernel/syscalls/sched_getparam/sched_getparam03.c		no	PC	int sched_getparam(pid_t pid, struct sched_param *p);	sched_getparam.2.gz
sched_getscheduler	set and get scheduling algorithm/parameters	no	ltp-full/testcases/kernel/syscalls/sched_getscheduler/sched_getscheduler01.c ltp-full/testcases/kernel/syscalls/sched_getscheduler/sched_getscheduler02.c		no	PC	int sched_getscheduler(pid_t pid);	sched_getscheduler.2.gz
sched_rr_get_interval	get the SCHED_RR interval for the named process	no	ltp-full/testcases/kernel/syscalls/sched_rr_get_interval/sched_rr_get_interval01.c ltp-full/testcases/kernel/syscalls/sched_rr_get_interval/sched_rr_get_interval02.c ltp-full/testcases/kernel/syscalls/sched_rr_get_interval/sched_rr_get_interval03.c		no	PC	int sched_rr_get_interval(pid_t pid, struct timespec *tp);	sched_rr_get_interval.2.gz
sched_setaffinity	set and get a process's CPU	no			addtl	PC	int sched_setaffinity(pid_t pid, unsigned int len, unsigned long	sched_setaffinity.2.gz
sched_setparam	set and get scheduling parameters	no	ltp-full/testcases/kernel/syscalls/sched_setparam/sched_setparam01.c ltp-full/testcases/kernel/syscalls/sched_setparam/sched_setparam02.c ltp-full/testcases/kernel/syscalls/sched_setparam/sched_setparam03.c ltp-full/testcases/kernel/syscalls/sched_setparam/sched_setparam04.c ltp-full/testcases/kernel/syscalls/sched_setparam/sched_setparam05.c		addtl	PC	int sched_setparam(pid_t pid, const struct sched_param *p);	sched_setparam.2.gz
sched_setscheduler	set and get scheduling algorithm/parameters	no	ltp-full/testcases/kernel/syscalls/sched_setscheduler/sched_setscheduler01.c ltp-full/testcases/kernel/syscalls/sched_setscheduler/sched_setscheduler02.c		addtl	PC	int sched_setscheduler(pid_t pid, int policy, const struct sched_param	sched_setscheduler.2.gz
sched_yield	yield the processor	no	ltp-full/testcases/kernel/syscalls/sched_yield/sched_yield01.c		no	PC	int sched_yield(void);	sched_yield.2.gz
select	synchronous I/O	no	ltp-full/testcases/kernel/syscalls/select/select01.c ltp-full/testcases/kernel/syscalls/select/select02.c ltp-full/testcases/kernel/syscalls/select/select03.c		no	FS	int select(int n, fd_set *readfds, fd_set *writefds, fd_set *exceptfds,	select.2.gz
semctl	semaphore control operations	yes	ltp-full/testcases/kernel/syscalls/ipc/semctl/semctl02.c ltp-full/testcases/kernel/syscalls/ipc/semctl/semctl03.c ltp-full/testcases/kernel/syscalls/ipc/semctl/semctl04.c ltp-full/testcases/kernel/syscalls/ipc/semctl/semctl05.c ltp-full/testcases/kernel/syscalls/ipc/semctl/semctl06.c ltp-full/testcases/kernel/syscalls/ipc/semctl/semctl07.c ltp-full/testcases/audit/syscalls/semctl_test.c	DA.1, DA.4, SM.2	addtl	IPC	int semctl(int semid, int semnum, int cmd, ...);	semctl.2.gz
semget	get a semaphore set identifier	yes	ltp-full/testcases/kernel/syscalls/ipc/semget/semget01.c ltp-full/testcases/kernel/syscalls/ipc/semget/semget02.c ltp-full/testcases/kernel/syscalls/ipc/semget/semget03.c ltp-full/testcases/kernel/syscalls/ipc/semget/semget05.c ltp-full/testcases/kernel/syscalls/ipc/semget/semget06.c ltp-full/testcases/audit/syscalls/semget_test.c	DA.1, DA.4, SM.2, OR.2	addtl	IPC	int semget(key_t key, int nsems, int semflg);	semget.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
semop	semaphore operations	yes	ltp-full/testcases/kernel/syscalls/ipc/semop/semop01.c ltp-full/testcases/kernel/syscalls/ipc/semop/semop02.c ltp-full/testcases/kernel/syscalls/ipc/semop/semop03.c ltp-full/testcases/kernel/syscalls/ipc/semop/semop04.c ltp-full/testcases/kernel/syscalls/ipc/semop/semop05.c ltp-full/testcases/audit/syscalls/semop_test.c	DA.1, DA.4	no	IPC	int semop(int semid, struct sembuf *sops, unsigned nsops);	semop.2.gz
semtimedop	semaphore operations	yes	ltp-full/testcases/audit/semtimedopt_test.c	DA.1, DA.4	no	IPC	int semtimedop(int semid, struct sembuf *sops, unsigned nsops, struct	semtimedop.2.gz
send	send a message from a socket	no	ltp-full/testcases/kernel/syscalls/send/send01.c		no	IPC	ssize_t send(int s, const void *msg, size_t len, int flags);	send.2.gz
sendfile	transfer data between file descriptors	no	ltp-full/testcases/kernel/syscalls/sendfile/sendfile02.c ltp-full/testcases/kernel/syscalls/sendfile/sendfile03.c		no	IPC	ssize_t sendfile(int out_fd, int in_fd, off_t *offset, size_t count);	sendfile.2.gz
sendmsg	send a message from a socket	no	ltp-full/testcases/kernel/syscalls/sendmsg/sendmsg01.c		no	IPC	ssize_t sendmsg(int s, const struct msghdr *msg, int flags);	sendmsg.2.gz
sendto	send a message from a socket	no	ltp-full/testcases/kernel/syscalls/sendto/sendto01.c		no	IPC	ssize_t sendto(int s, const void *msg, size_t len, int flags, const	sendto.2.gz
set_mempolicy	set the NUMA memory policy of the current process to policy	no			no	MM	int set_mempolicy(int policy, unsigned long *nodemask, unsigned long maxnode);	set_mempolicy.2.gz
set_thread_area	set a Thread Local Storage (TLS) area	no			no	PC	int set_thread_area (struct user_desc *u_info);	set_thread_area.2.gz
set_tid_address	sets the current clear_child_tid to tidptr	no			no	PC	long sys_set_tid_address (int *tidptr);	sys_set_tid_address.2.gz
setdomainname	get/set domain name	no	ltp-full/testcases/kernel/syscalls/setdomainname/setdomainname01.c ltp-full/testcases/kernel/syscalls/setdomainname/setdomainname02.c ltp-full/testcases/kernel/syscalls/setdomainname/setdomainname03.c ltp-full/testcases/audit/syscalls/setdomainname_test.c		admin only	IPC	int setdomainname(const char *name, size_t len);	setdomainname.2.gz
setfsgid	set group identity used for file system checks	yes	ltp-full/testcases/kernel/syscalls/setfsgid/setfsgid01.c ltp-full/testcases/audit/syscalls/setfsgid_test.c	IA.4	addtl	PC	int setfsgid(uid_t fsgid);	setfsgid.2.gz
setfsuid	set user identity used for file system checks	yes	ltp-full/testcases/kernel/syscalls/setfsuid/setfsuid01.c ltp-full/testcases/audit/syscalls/setfsuid_test.c	IA.4	addtl	PC	int setfsuid(uid_t fsuid);	setfsuid.2.gz
setgid	set group identity	yes	ltp-full/testcases/kernel/syscalls/setgid/setgid01.c ltp-full/testcases/kernel/syscalls/setgid/setgid02.c ltp-full/testcases/kernel/syscalls/setgid/setgid03.c ltl-full/testcases/audit/syscalls/setgid_test.c	IA.4	addtl	PC	int setgid(gid_t gid);	setgid.2.gz
setgroups	get/set list of supplementary group IDs	yes	ltp-full/testcases/kernel/syscalls/setgroups/setgroups01.c ltp-full/testcases/kernel/syscalls/setgroups/setgroups02.c ltp-full/testcases/kernel/syscalls/setgroups/setgroups03.c ltp-full/testcases/kernel/syscalls/setgroups/setgroups04.c ltp-full/testcases/audit/syscalls/setgroups_test.c	IA.4	addtl	PC	int setgroups(size_t size, const gid_t *list);	setgroups.2.gz
sethostname	get/set host name	no	ltp-full/testcases/kernel/syscalls/sethostname/sethostname01.c ltp-full/testcases/kernel/syscalls/sethostname/sethostname02.c ltp-full/testcases/kernel/syscalls/sethostname/sethostname03.c ltp-full/testcases/audit/syscalls/sethostname_test.c		admin only	NI	int sethostname(const char *name, size_t len);	sethostname.2.gz
setitimer	get or set value of an interval timer	no	ltp-full/testcases/kernel/syscalls/setitimer/setitimer01.c ltp-full/testcases/kernel/syscalls/setitimer/setitimer02.c ltp-full/testcases/kernel/syscalls/setitimer/setitimer03.c		no	PC	int setitimer(int which, const struct itimerval *value, struct itimer-	setitimer.2.gz
setpgid	set/get process group	no	ltp-full/testcases/kernel/syscalls/setpgid/setpgid01.c ltp-full/testcases/kernel/syscalls/setpgid/setpgid02.c ltp-full/testcases/kernel/syscalls/setpgid/setpgid03.c ltp-full/testcases/audit/syscalls/setpgid_test.c		no	PC	int setpgid(pid_t pid, pid_t pgid);	setpgid.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
setpriority	get/set program scheduling priority	no	ltp-full/testcases/kernel/syscalls/setpriority/setpriority01.c ltp-full/testcases/kernel/syscalls/setpriority/setpriority02.c ltp-full/testcases/kernel/syscalls/setpriority/setpriority03.c ltp-full/testcases/kernel/syscalls/setpriority/setpriority04.c ltp-full/testcases/kernel/syscalls/setpriority/setpriority05.c ltp-full/testcases/audit/syscalls/setpriority_test.c		addtl	PC	int setpriority(int which, int who, int prio);	setpriority.2.gz
setregid	set real and/or effective user or group ID	yes	ltp-full/testcases/kernel/syscalls/setregid/setregid01.c ltp-full/testcases/kernel/syscalls/setregid/setregid02.c ltp-full/testcases/kernel/syscalls/setregid/setregid03.c ltp-full/testcases/kernel/syscalls/setregid/setregid04.c ltp-full/testcases/audit/syscalls/setregid_test.c	IA.4	addtl	PC	int setregid(gid_t rgid, gid_t egid);	setregid.2.gz
setresgid	set real, effective and saved user or group ID	yes	ltp-full/testcases/kernel/syscalls/setresgid/setresgid01.c ltp-full/testcases/kernel/syscalls/setresgid/setresgid02.c ltp-full/testcases/kernel/syscalls/setresgid/setresgid03.c ltp-full/testcases/audit/syscalls/setresgid_test.c	IA.4	addtl	PC	int setresgid(gid_t rgid, gid_t egid, gid_t sgid);	setresgid.2.gz
setresuid	set real, effective and saved user or group ID	yes	ltp-full/testcases/kernel/syscalls/setresuid/setresuid01.c ltp-full/testcases/kernel/syscalls/setresuid/setresuid02.c ltp-full/testcases/kernel/syscalls/setresuid/setresuid03.c ltp-full/testcases/audit/syscalls/setresuid_test.c	IA.4	addtl	PC	int setresuid(uid_t ruid, uid_t euid, uid_t suid);	setresuid.2.gz
setreuid	set real and/or effective user or group ID	yes	ltp-full/testcases/kernel/syscalls/setreuid/setreuid01.c ltp-full/testcases/kernel/syscalls/setreuid/setreuid02.c ltp-full/testcases/kernel/syscalls/setreuid/setreuid03.c ltp-full/testcases/kernel/syscalls/setreuid/setreuid04.c ltp-full/testcases/kernel/syscalls/setreuid/setreuid05.c ltp-full/testcases/kernel/syscalls/setreuid/setreuid06.c ltp-full/testcases/audit/syscalls/setreuid_test.c	IA.4	addtl	PC	int setreuid(uid_t ruid, uid_t euid);	setreuid.2.gz
setrlimit	get/set resource limits and usage	no	ltp-full/testcases/kernel/syscalls/setrlimit/setrlimit01.c ltp-full/testcases/kernel/syscalls/setrlimit/setrlimit02.c ltp-full/testcases/kernel/syscalls/setrlimit/setrlimit03.c ltp-full/testcases/audit/syscalls/setrlimit_test.c		addtl	PC	int setrlimit(int resource, const struct rlimit *rlim);	setrlimit.2.gz
setsid	creates a session and sets the process group ID	no	ltp-full/testcases/kernel/syscalls/setsid/setsid01.c ltp-full/testcases/audit/syscalls/setsid_test.c		no	PC	pid_t setsid(void);	setsid.2.gz
setsockopt	get and set options on sockets	no	ltp-full/testcases/kernel/syscalls/setsockopt/setsockopt01.c		no	IPC	int setsockopt(int s, int level, int optname, const void *optval,	setsockopt.2.gz
settimeofday	get / set time	yes	ltp-full/testcases/kernel/syscalls/settimeofday/settimeofday01.c ltp-full/testcases/kernel/syscalls/settimeofday/settimeofday02.c ltp-full/testcases/audit/syscalls/settimeofday_test.c	SM.5	admin only	PC	int settimeofday(const struct timeval *tv , const struct timezone *tz);	settimeofday.2.gz
setuid	set user identity	yes	ltp-full/testcases/kernel/syscalls/setuid/setuid01.c ltp-full/testcases/kernel/syscalls/setuid/setuid02.c ltp-full/testcases/kernel/syscalls/setuid/setuid03.c ltp-full/testcases/audit/syscalls/setuid_test.c	IA.4	addtl	PC	int setuid(uid_t uid);	setuid.2.gz
setxattr	set an extended attribute value	yes	misc_test/ext3_ACLs/acl-tests/permissions.test misc_test/ext3_ACLs/acl-tests/setfacl.test misc_test/ext3_ACLs/acl-tests/getfacl-noacl.test misc_test/ext3_ACLs/acl-tests/misc.test ltp-full/testcases/audit/syscalls/setxattr_test.c	DA.1, DA.3, SM.2	addtl	FS	int setxattr (const char *path, const char *name,	setxattr.2.gz
sgetmask	returns or sets the signal mask	no			no	IPC	long sys_sgetmask (void);	sgetmask.2.gz
shmat	Attach the shared memory segment identified by shmid to the address space of the calling process	yes	ltp-full/testcases/kernel/syscalls/ipc/shmat/shmat01.c ltp-full/testcases/kernel/syscalls/ipc/shmat/shmat02.c ltp-full/testcases/kernel/syscalls/ipc/shmat/shmat03.c ltp-full/testcases/audit/syscalls/shmat_test.c	DA.1, DA.4	addtl	IPC	long sys_shmat (int shmid, char *shmaddr, int shmflg, ulong *raddr);	shmat.2.gz
shmctl	shared memory control	yes	ltp-full/testcases/kernel/syscalls/ipc/shmctl/shmctl01.c ltp-full/testcases/kernel/syscalls/ipc/shmctl/shmctl02.c ltp-full/testcases/kernel/syscalls/ipc/shmctl/shmctl03.c ltp-full/testcases/kernel/syscalls/ipc/shmctl/shmctl04.c ltp-full/testcases/audit/syscalls/shmctl_test.c	DA.1, DA.4, SM.2	addtl	IPC	int shmctl(int shmid, int cmd, struct shmid_ds *buf);	shmctl.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
shmdt	shared memory operations	no	ltp-full/testcases/audit/syscalls/shmdt_test.c		no	IPC	int shmdt(const void *shmaddr);	shmdt.2.gz
shmget	allocates a shared memory segment	yes	ltp-full/testcases/kernel/syscalls/ipc/shmget/shmget01.c ltp-full/testcases/kernel/syscalls/ipc/shmget/shmget02.c ltp-full/testcases/kernel/syscalls/ipc/shmget/shmget03.c ltp-full/testcases/kernel/syscalls/ipc/shmget/shmget04.c ltp-full/testcases/kernel/syscalls/ipc/shmget/shmget05.c ltp-full/testcases/audit/syscalls/shmget_test.c	DA.1, DA.4, SM.2, OR.2, OR.3	addtl	IPC	int shmget(key_t key, size_t size, int shmflg);	shmget.2.gz
shutdown	shut down part of a full-duplex connection	no			no	IPC	int shutdown(int s, int how);	shutdown.2.gz
sigaction	POSIX signal handling	no	ltp-full/testcases/kernel/syscalls/sigaction/sigaction01.c ltp-full/testcases/kernel/syscalls/sigaction/sigaction02.c		no	IPC	int sigaction(int signum, const struct sigaction *act, struct sigaction	sigaction.2.gz
sigaltstack	Allow a process to define an alternate signal stack or	no			no	PC	long sys_sigaltstack (const stack_t *uss, stack_t *uoss,	sigaltstack.2.gz
signal	ANSI C signal handling	no	ltp-full/testcases/kernel/syscalls/signals/signal01.c ltp-full/testcases/kernel/syscalls/signals/signal02.c ltp-full/testcases/kernel/syscalls/signals/signal03.c ltp-full/testcases/kernel/syscalls/signals/signal04.c ltp-full/testcases/kernel/syscalls/signals/signal05.c		no	IPC	sighandler_t signal(int signum, sighandler_t handler);	signal.2.gz
sigpending	POSIX signal handling	no	ltp-full/testcases/kernal/syscalls/sigpending/sigpending02.c		no	IPC	int sigpending(sigset_t *set);	sigpending.2.gz
sigprocmask	POSIX signal handling	no	ltp-full/testcases/kernal/syscalls/sigprocmask/sigprocmask01.c		no	IPC	int sigprocmask(int how, const sigset_t *set, sigset_t *oldset);	sigprocmask.2.gz
sigreturn	return from signal handler and cleanup stack frame	no			no	IPC	int sigreturn(unsigned long __unused);	sigreturn.2.gz
sigsuspend	POSIX signal handling	no	ltp-full/testcases/kernel/syscalls/sigsuspend/sigsuspend01.c		no	IPC	int sigsuspend(const sigset_t *mask);	sigsuspend.2.gz
socket	create an endpoint for communication	no	ltp-full/testcases/kernel/syscalls/socket/socket01.c		no	IPC	int socket(int domain, int type, int protocol);	socket.2.gz
socketcall	socket system calls	no	ltp-full/testcases/kernel/syscalls/socketcall/socketcall01.c ltp-full/testcases/kernel/syscalls/socketcall/socketcall02.c ltp-full/testcases/kernel/syscalls/socketcall/socketcall03.c ltp-full/testcases/kernel/syscalls/socketcall/socketcall04.c	See comment2	See comment2	IPC	int socketcall(int call, unsigned long *args);	socketcall.2.gz
socketpair	create a pair of connected sockets	no	ltp-full/testcases/kernel/syscalls/socketpair/socketpair01.c		no	IPC	int socketpair(int d, int type, int protocol, int sv[2]);	socketpair.2.gz
ssetmask	returns or sets the signal mask	no			no	IPC	long sys_ssetmask (int newmask);	ssetmask.2.gz
stat	get file status	no	ltp-full/testcases/kernel/syscalls/stat/stat01.c ltp-full/testcases/kernel/syscalls/stat/stat02.c ltp-full/testcases/kernel/syscalls/stat/stat03.c ltp-full/testcases/kernel/syscalls/stat/stat05.c ltp-full/testcases/kernel/syscalls/stat/stat06.c		addtl	FS	int stat(const char *file_name, struct stat *buf);	stat.2.gz
statfs	get file system statistics	no	ltp-full/testcases/kernel/syscalls/statfs/statfs01.c ltp-full/testcases/kernel/syscalls/statfs/statfs02.c		no	FS	int statfs(const char *path, struct statfs *buf);	statfs.2.gz
stime	set time	yes	ltp-full/testcases/audit/syscalls/stime_test.c	SM.5	admin only	PC	int stime(time_t *t);	stime.2.gz
swapcontext	manipulate user context	no			no	IPC	int swapcontext (ucontext_t *oucp, ucontext_t *ucp);	swapcontext.3p.gz
swapoff	start/stop swapping to file/device	no	ltp-full/testcases/kernel/syscalls/swapoff/swapoff01.c ltp-full/testcases/kernel/syscalls/swapoff/swapoff02.c ltp-full/testcases/audit/syscalls/swapoff_test.c		admin only	FS	int swapoff(const char *path);	swapoff.2.gz
swapon	start/stop swapping to file/device	yes	ltp-full/testcases/kernel/syscalls/swapon/swapon01.c ltp-full/testcases/audit/syscalls/swapon_test.c	DA.1, DA.3	admin only	PC	int swapon(const char *path, int swapflags);	swapon.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
symlink	make a new name for a file	yes	ltp-full/testcases/kernel/syscalls/symlink/symlink01.c ltp-full/testcases/kernel/syscalls/symlink/symlink02.c ltp-full/testcases/kernel/syscalls/symlink/symlink03.c ltp-full/testcases/kernel/syscalls/symlink/symlink04.c ltp-full/testcases/kernel/syscalls/symlink/symlink05.c ltp-full/testcases/audit/syscalls/symlink_test.c	DA.1, DA.3	addtl	FS	int symlink(const char *oldpath, const char *newpath);	symlink.2.gz
sync	commit buffer cache to disk	no	ltp-full/testcases/kernel/syscalls/sync/sync01.c		no	PC	void sync(void);	sync.2.gz
sysctl	read/write system parameters	no	ltp-full/testcases/kernel/syscalls/sysctl/sysctl01.c ltp-full/testcases/kernel/syscalls/sysctl/sysctl03.c ltp-full/testcases/kernel/syscalls/sysctl/sysctl04.c ltp-full/testcases/kernel/syscalls/sysctl/sysctl05.c		admin only	FS / PC	int _sysctl(struct __sysctl_args *args);	sysctl.2.gz
sysfs	get file system type information	no	ltp-full/testcases/kernel/syscalls/sysfs/sysfs01.c ltp-full/testcases/kernel/syscalls/sysfs/sysfs02.c ltp-full/testcases/kernel/syscalls/sysfs/sysfs03.c ltp-full/testcases/kernel/syscalls/sysfs/sysfs04.c ltp-full/testcases/kernel/syscalls/sysfs/sysfs05.c ltp-full/testcases/kernel/syscalls/sysfs/sysfs06.c		no	FS	int sysfs(int option, const char *fsname);	sysfs.2.gz
sysinfo	returns information on overall system statistics	no	ltp-full/testcases/kernel/syscalls/sysinfo/sysinfo01.c		no	PC	int sysinfo(struct sysinfo *info);	sysinfo.2.gz
syslog	read and/or clear kernel message ring buffer; set	no	ltp-full/testcases/audit/syscalls/syslog_test.c		addtl	KM	int syslog(int type, char *bufp, int len);	syslog.2.gz
tgkill	send signal sig to one specific thread, tgid	no			addtl	PC	long sys_tgkill (int tgid, int pid, int sig);	sys_tgkill.2.gz
time	get time in seconds	no	ltp-full/testcases/kernel/syscalls/time/time01.c		no	PC	time_t time(time_t *t);	time.2.gz
timer_create	create a per-process timer (REALTIME)	no			no	PC	int timer_create(clockid_t clockid, struct sigevent *restrict evp, timer_t *restrict timerid);	timer_create.3p.gz
timer_delete	delete a per-process timer (REALTIME)	no			no	PC	int timer_delete(timer_t timerid);	timer_delete.3p.gz
timer_gettime	per-process timer (REALTIME)	no			no	PC	int timer_gettime(timer_t timerid, struct itimerspec *value);	timer_gettime.3p.gz
timer_getoverrun	per-process timer (REALTIME)	no			no	PC	int timer_getoverrun(timer_t timerid);	timer_getoverrun.3p.gz
timer_settime	per-process timer (REALTIME)	no			no	PC	int timer_settime(timer_t timerid, int flags, const struct itimerspec *restrict value, struct itimerspec *restrict ovalue);	timer_settime.3p.gz
times	get process times	no	ltp-full/testcases/kernel/syscalls/times/times01.c ltp-full/testcases/kernel/syscalls/times/times02.c ltp-full/testcases/kernel/syscalls/times/times03.c		no	PC	timer_t *restrict timerid);	times.2.gz
tkill	send a signal to a single process	no	ltp-full/testcases/audit/syscalls/tkill_test.c		addtl	PC	int tkill(pid_t tid, int sig);	tkill.2.gz
truncate	truncate a file to a specified length	yes	ltp-full/testcases/kernel/syscalls/truncate/truncate01.c ltp-full/testcases/kernel/syscalls/truncate/truncate02.c ltp-full/testcases/kernel/syscalls/truncate/truncate03.c ltp-full/testcases/kernel/syscalls/truncate/truncate04.c ltp-full/testcases/audit/syscalls/truncate_test.c	DA.1, DA.3, OR.1	addtl	FS	int truncate(const char *path, off_t length);	truncate.2.gz
tux	interact with the TUX kernel subsystem	no			no	NI	int tux (unsigned int action, user_req_t * req);	tux.2.gz
umask	set file creation mask	yes	ltp-full/testcases/kernel/syscalls/umask/umask01.c ltp-full/testcases/kernel/syscalls/umask/umask02.c ltp-full/testcases/kernel/syscalls/umask/umask03.c ltp-full/testcases/audit/syscalls/umask_test.c	SM.2	no	FS	mode_t umask(mode_t mask);	umask.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
umount	remove the file system mounted at name clock	no	ltp-full/testcases/audit/syscalls/umount_test.c		admin only	FS	long sys_umount (char *name, int flags);	umount.2.gz
uname	get name and information about current kernel	no	ltp-full/testcases/kernel/syscalls/uname/uname01.c ltp-full/testcases/kernel/syscalls/uname/uname02.c ltp-full/testcases/kernel/syscalls/uname/uname03.c		no	PC	int uname(struct utsname *buf);	uname.2.gz
unlink	delete a name and possibly the file it refers to	yes	ltp-full/testcases/kernel/syscalls/unlink/unlink05.c ltp-full/testcases/kernel/syscalls/unlink/unlink06.c ltp-full/testcases/kernel/syscalls/unlink/unlink07.c ltp-full/testcases/kernel/syscalls/unlink/unlink08.c ltp-full/testcases/audit/syscalls/unlink_test.c	DA.1, DA.3	addtl	FS	int unlink(const char *pathname);	unlink.2.gz
uselib	select shared library	no			no	PC	int uselib(const char *library);	uselib.2.gz
ustat	get file system statistics	no			no	FS	int ustat(dev_t dev, struct ustat *ubuf);	ustat.2.gz
utime	change access and/or modification times of an inode	yes	ltp-full/testcases/kernel/syscalls/utime/utime01.c ltp-full/testcases/kernel/syscalls/utime/utime02.c ltp-full/testcases/kernel/syscalls/utime/utime03.c ltp-full/testcases/kernel/syscalls/utime/utime04.c ltp-full/testcases/kernel/syscalls/utime/utime05.c ltp-full/testcases/kernel/syscalls/utime/utime06.c ltp-full/testcases/audit/syscalls/utime_test.c	DA.1, DA.3	addtl	FS	int utime(const char *filename, struct utimbuf *buf);	utime.2.gz
utimes	change access and/or modification times of an inode	yes	ltp-full/testcases/kernel/syscalls/utime/utime01.c ltp-full/testcases/kernel/syscalls/utime/utime02.c ltp-full/testcases/kernel/syscalls/utime/utime03.c ltp-full/testcases/kernel/syscalls/utime/utime04.c ltp-full/testcases/kernel/syscalls/utime/utime05.c ltp-full/testcases/kernel/syscalls/utime/utime06.c ltp-full/testcases/audit/syscalls/utimes_test.c	DA.1, DA.3	addtl	FS	int utimes(char* filename, struct timeval *tvp);	utimes.2.gz
vfork	create a child process and block parent	yes	ltp-full/testcases/kernel/syscalls/vfork/vfork01.c ltp-full/testcases/kernel/syscalls/vfork/vfork02.c ltp-full/testcases/audit/syscalls/vfork_test.c	OR.3	no	PC	pid_t vfork(void);	vfork.2.gz
vhangup	virtually hangup the current tty	no	ltp-full/testcases/kernel/syscalls/vhangup/vhangup01.c ltp-full/testcases/kernel/syscalls/vhangup/vhangup02.c		admin only	PC	int vhangup(void);	vhangup.2.gz
vm86	enter virtual 8086 mode	no			no	PC	int vm86(unsigned long fn, struct vm86plus_struct *v86);	vm86.2.gz
wait4	wait for process termination, BSD style	no	ltp-full/testcases/kernel/syscalls/wait4/wait401.c ltp-full/testcases/kernel/syscalls/wait4/wait402.c		no	PC	pid_t wait4(pid_t pid, int *status, int options,	wait4.2.gz
waitid	wait for process to change state	no			no	PC	int waitid(idtype_t idtype, id_t id, siginfo_t *infop, int options);	waitid.2.gz
waitpid	wait for process termination	no	ltp-full/testcases/kernel/syscalls/waitpid/waitpid01.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid02.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid03.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid04.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid05.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid06.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid07.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid08.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid09.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid10.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid11.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid12.c ltp-full/testcases/kernel/syscalls/waitpid/waitpid13.c		no	PC	pid_t waitpid(pid_t pid, int *status, int options);	waitpid.2.gz
write	write to a file descriptor	no	ltp-full/testcases/kernel/syscalls/write/write01.c ltp-full/testcases/kernel/syscalls/write/write02.c ltp-full/testcases/kernel/syscalls/write/write03.c ltp-full/testcases/kernel/syscalls/write/write04.c ltp-full/testcases/kernel/syscalls/write/write05.c		no	FS	ssize_t write(int fd, const void *buf, size_t count);	write.2.gz

Syscall	Description	Security Relevant	Test Case(s)	TSF	Privilege	HLD Module	libc prototype	Manpage
writev	read or write data into multiple buffers	no	ltp-full/testcases/kernel/syscalls/writev/writev01.c ltp-full/testcases/kernel/syscalls/writev/writev02.c ltp-full/testcases/kernel/syscalls/writev/writev03.c ltp-full/testcases/kernel/syscalls/writev/writev04.c ltp-full/testcases/kernel/syscalls/writev/writev05.c		no	FS	ssize_t writev(int fd, const struct iovec *vector, int count);	writev.2.gz

Comment1
Except for the x86_64 architecture, **ipc** related system calls are mapped to one kernel entry point (syscall number) which has a number of subfunctions for the individual ipc related system calls. This entry point is named "ipc". The subfunctions are documented individually, and indicated by an "i" suffix in the system call number column.

Comment2
Except for the x86_64 architecture, socket related system calls are mapped to one kernel entry point (syscall number) which has a number of subfunctions for the individual socket related system calls. This entry point is named "socketcall". The subfunctions are documented individually, and indicated by an "s" suffix in the system call number column.

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
accept		102s	43	102s	102s	102s	102s	102s	102s	102s
access		33	21	33	33	33	33	33	33	33
acct		51	163	51	51	51	51	51	51	51
adjtimex		124	159	124	124	124	124	124	124	124
alarm		27	37	27	27	27	27	27	27	27
arch_prctl		none	158	none	none	none	none	none	none	none
bdflush		134	none	none	134	134	134	134	134	134
bind		102s	49	102s	102s	102s	102s	102s	102s	102s
brk		45	12	45	45	45	45	45	45	45

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
capget		184	125	184	183	183	183	184	184	184
capset		185	126	185	184	184	184	185	185	185
chdir		12	80	12	12	12	12	12	12	12
chmod		15	90	15	15	15	15	15	15	15
chown		182 212	92	182 212	181	181	181	182 212	212	182 212
chroot		61	161	61	61	61	61	61	61	61
clock_getres		266	229	266	247	247	247	261	261	261
clock_gettime		265	228	265	246	246	246	260	260	260
clock_nanosleep		267	230	267	248	248	248	262	262	262
clock_settime		264	227	264	245	245	245	259	259	259
clone		120	56	120	120	120	120	120	120	120
close		6	3	6	6	6	6	6	6	6
connect		102s	42	102s	102s	102s	102s	102s	102s	102s

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
creat		8	85	8	8	8	8	8	8	8
delete_module		129	176	129	129	129	129	129	129	129
dup		41	32	41	41	41	41	41	41	41
dup2		63	33	63	63	63	63	63	63	63
epoll_create		254	213	254	236	236	120 236	249	249	249
epoll_ctl		255	214	255	237	237	237	250	250	250
epoll_wait		256	215	256	238	238	238	251	251	251
execve		11	59	11	11	11	11	11	11	11
exit		1	60	1	1	1	1	1	1	1
exit_group		252	231	252	234	234	234	248	248	248
fadvise		250 272	221	250 272	233 254	233	233 254	253 264	253	253 264
fchdir		133	81	133	133	133	133	133	133	133

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
fchmod		94	91	94	94	94	94	94	94	94
fchown		95 207	93	95 207	95	95	95	95 207	207	95 207
fcntl		55 221	72	55 221	55 204	55	55 204	55 221	55	55 221
fdatasync		148	75	148	148	148	148	148	148	148
fgetxattr		231	193	231	214	214	214	229	229	229
flistxattr		234	196	234	217	217	217	232	232	232
flock		143	73	143	143	143	143	143	143	143
fork		2	57	2	2	2	2	2	2	2
fremovexattr		237	199	237	220	220	220	235	235	235
fsetxattr		228	190	228	211	211	211	226	226	226
fstat		28 108 197	5	28 108 197	28 108 197	108	108 197	108 197	108	108 197
fstatfs		100	138	100 269	100 253	100 253	100 253	100 266	100 266	100 266
fsync		118	74	118	118	118	118	118	118	118
ftruncate		93 194	77	93 194	93 194	93	93 194	93 194	93	93 194

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
futex		240	202	240	221	221	221	238	238	238
get_thread_area		244	none	244	none	none	none	none	none	none
getcwd		183	79	183	182	182	182	183	183	183
getdents		141 220	78 217	141 220	141 202	141 202	141 202	141 220	141 220	141 220
getegid		50 202	108	50 202	50	50	50	50 202	202	50 202
geteuid		49 201	107	49 201	49	49	49	49 201	201	49 201
getgid		47 200	104	47 200	47	47	47	47 200	200	47 200
getgroups		80 205	115	80 205	80	80	80	80 205	205	80 205
getitimer		105	36	105	105	105	105	105	105	105
get_mempolicy		275	239	275	none	none	none	none	none	none
getpeername		102s	52	102s	102s	102s	102s	102s	102s	102s
getpgid		132	121	132	132	132	132	132	132	132
getpgrp		65	111	65	65	65	65	65	65	65
getpid		20	39	20	20	20	20	20	20	20
getppid		64	110	64	64	64	64	64	64	64
getpriority		96	140	96	96	96	96	96	96	96
getresgid		171 211	120	171 211	170	170	170	171 211	211	171 211
getresuid		165 209	118	165 209	165	165	165	165 209	209	165 209
getrlimit		76 191	97	76 191	76 190	190	76 190	76 191	76 191	76 191
getrusage		77	98	77	77	77	77	77	77	77
getsid		147	124	147	147	147	147	147	147	147
getsockname		102s	51	102s	102s	102s	102s	102s	102s	102s
getsockopt		102s	55	102s	102s	102s	102s	102s	102s	102s

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
gettid		224	186	224	207	207	207	236	236	236
gettimeofday		78	96	78	78	78	78	78	78	78
getuid		24 199	102	78 199	24	24	24	24 199	199	24 199
getxattr		229	191	229	212	212	212	227	227	227
init_module		128	175	128	128	128	128	128	128	128
io_cancel		249	210	249	231	231	231	247	247	247
io_destroy		246	207	246	228	228	228	244	244	244
io_getevents		247	208	247	229	229	229	245	245	245
io_setup		245	206	245	227	227	227	243	243	243
io_submit		248	209	248	230	230	230	246	246	246
ioctl		54	16	54	54	54	54	54	54	54
ioperm		101	173	101	none	none	none	none	none	none
iopl		110	172	110	none	none	none	none	none	none
ipc		117	none	117	117	117	117	117	117	117
kexec_load		283	247	none	268	none	none	none	none	none
kill		37	62	37	37	37	37	37	37	37

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
lchown		16 198	94	16 198	16	16	16	16 198	198	16 198
lgetxattr		230	192	230	213	213	213	228	228	228
link		9	86	9	9	9	9	9	9	9
listen		102s	50	102s	102s	102s	102s	102s	102s	102s
listxattr		232	194	232	215	215	215	230	230	230
llistxattr		233	195	233	216	216	216	231	231	231
llseek		140	none	140	140	140	140	140	140	140
lookup_dcookie		253	212	253	235	235	235	110	110	110
lremovexattr		236	198	236	219	219	219	234	234	234
lseek		19	8	19	19	19	19	19	19	19
lsetxattr		227	189	227	210	210	210	225	225	225
lstat		84 107 196	6	84 107 196	84 107 196	107	107 196	107 196	107	107 196
madvise		219	28	219	205	205	205	219	219	219
mbind		274	237	274	none	259	259	none	none	none
mincore		218	27	218	206	206	206	218	218	218
mkdir		39	83	39	39	39	39	39	39	39

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
mknod		14	133	14	14	14	14	14	14	14
mlock		150	149	150	150	150	150	150	150	150
mlockall		152	151	152	152	152	152	152	152	152
mmap		90	9	90	90	90	90	90	90	90
mmap2		192	none	192	192	none	192	192	192	192
modify_ldt		123	154	123	none	none	none	none	none	none
mount		21	165	21	21	21	21	21	21	21
mprotect		125	10	125	125	125	125	125	125	125
mq_getsetattr		282	245	282	267	267	267	276	276	276
mq_notify		281	244	281	266	266	266	275	275	275
mq_open		277	240	277	262	262	262	271	271	271
mq_timedreceive		280	243	280	265	265	265	274	274	274
mq_timedsend		279	242	279	264	264	264	273	273	273

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
mq_unlink		278	241	278	263	263	263	272	272	272
mremap		163	25	163	163	163	163	163	163	163
msgctl		117i	71	117i	117i	117i	117i	117i	117i	117i
msgget		117i	68	117i	117i	117i	117i	117i	117i	117i
msgrcv		117i	70	117i	117i	117i	117i	117i	117i	117i
msgsnd		117i	69	117i	117i	117i	117i	117i	117i	117i
msync		144	26	144	144	144	144	144	144	144
munlock		151	150	151	151	151	151	151	151	151
munlockall		153	152	153	153	153	153	153	153	153
munmap		91	11	91	91	91	91	91	91	91
nanosleep		162	35	162	162	162	162	162	162	162
nfsservctl		169	180	169	168	168	168	169	169	169
nice		34	none	34	34	34	34	34	34	34

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
open		5	2	5	5	5	5	5	5	5
pause		29	34	29	29	29	29	29	29	29
pciconfig_iobase		none	none	none	200	none	200	none	none	none
pciconfig_read		none	none	none	198	none	198	none	none	none
pciconfig_write		none	none	none	199	none	199	none	none	none
personality		136	135	136	136	136	136	136	136	136
pipe		42	22	42	42	42	42	42	42	42
pivot_root		217	155	217	203	203	203	217	217	217
poll		168	7	168	167	167	167	168	168	168
prctl		172	157	172	171	171	171	172	172	172
pread		180	17	180	179	179	179	180	180	180
ptrace		26	101	26	26	26	26	26	26	26
pwrite		181	18	181	180	180	180	181	181	181
quotactl		131	179	131	131	131	131	131	131	131
read		3	0	3	3	3	3	3	3	3
readahead		225	187	225	191	191	191	222	222	222

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
readdir		89	none	89	89	none	89	none	none	89
readlink		85	89	85	85	85	85	85	85	85
readv		145	19	145	145	145	145	145	145	145
reboot		88	169	88	88	88	88	88	88	88
recv		102s	none	102s	102s	102s	102s	102s	102s	102s
recvfrom		102s	45	102s	102s	102s	102s	102s	102s	102s
recvmsg		102s	47	102s	102s	102s	102s	102s	102s	102s
remap_file_pages		257	216	257	239	239	239	none	none	none
removexattr		235	197	235	218	218	218	233	233	233
rename		38	82	38	38	38	38	38	38	38
restart_syscall		0	219	0	0	0	0	7	7	7
rmdir		40	84	40	40	40	40	40	40	40
rt_sigaction		174	13	174	173	173	173	174	174	174
rt_sigpending		176	127	176	175	175	175	176	176	176
rt_sigprocmask		175	14	175	174	174	174	175	175	175
rt_sigqueueinfo		178	129	178	177	177	177	178	178	178

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
rt_sigreturn		173	15	173	172	172	172	173	173	173
rt_sigsuspend		179	130	179	178	178	178	179	179	179
rt_sigtimedwait		177	128	177	176	176	176	177	177	177
rtas		none	none	none	none	255	255	none	none	none
sched_get_priority_max		159	146	159	159	159	159	159	159	159
sched_get_priority_min		160	147	160	160	160	160	160	160	160
sched_getaffinity		242	204	242	223	223	223	240	240	240
sched_getparam		155	143	155	155	155	155	155	155	155
sched_getscheduler		157	145	157	157	157	157	157	157	157
sched_rr_get_interval		161	148	161	161	161	161	161	161	161
sched_setaffinity		241	203	241	222	222	222	239	239	239
sched_setparam		154	142	154	154	154	154	154	154	154
sched_setscheduler		156	144	156	156	156	156	156	156	156
sched_yield		158	24	158	158	158	158	158	158	158
select		82 142	23	82 142	82 142	142	142	142	142	142
semctl		117i	66	117i	117i	117i	117i	117i	117i	117i
semget		117i	64	117i	117i	117i	117i	117i	117i	117i

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
semop		117i	65	117i	117i	117i	117i	117i	117i	117i
semtimedop		117i	220	117i	117i	117i	117i	117i	117i	117i
send		102s	none	102s	102s	102s	102s	102s	102s	102s
sendfile		187 239	40	187 239	186 226	186	186 226	187 223	187	187 223
sendmsg		102s	46	102s	102s	102s	102s	102s	102s	102s
sendto		102s	44	102s	102s	102s	102s	102s	102s	102s
set_mempolicy		276	238	276						
set_thread_area		243	none	243	none	none	none	none	none	none
set_tid_address		258	218	258	232	232	232	252	252	252
setdomainname		121	171	121	121	121	121	121	121	121
setfsgid		139 216	123	139 216	139	139	139	139 216	216	139 216
setfsuid		138 215	122	138 215	138	138	138	138 215	215	138 215
setgid		46 214	106	46 214	46	46	46	46 214	214	46 214
setgroups		81 206	116	81 206	81	81	81	81 206	206	81 206
sethostname		74	170	74	74	74	74	74	74	74
setitimer		104	38	104	104	104	104	104	104	104
setpgid		57	109	57	57	57	57	57	57	57

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
setpriority		97	141	97	97	97	97	97	97	97
setregid		71 204	114	71 204	71	71	71	71 204	204	71 204
setresgid		170 210	119	170 210	169	169	169	170 210	210	170 210
setresuid		164 208	117	164 208	164	164	164	164 208	208	164 208
setreuid		70 203	113	70 203	70	70	70	70 203	203	70 203
setrlimit		75	160	75	75	75	75	75	75	75
setsid		66	112	66	66	66	66	66	66	66
setsockopt		102s	54	102s	102s	102s	102s	102s	102s	102s
settimeofday		79	164	79	79	79	79	79	79	79
setuid		23 213	105	23 213	23	23	23	23 213	213	23 213
setxattr		226	188	226	209	209	209	224	224	224
sgetmask		68	none	68	68	68	68	none	none	none
shmat		117i	30	117i	117i	117i	117i	117i	117i	117i
shmctl		117i	31	117i	117i	117i	117i	117i	117i	117i

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
shmdt		117i	67	117i	117i	117i	117i	117i	117i	117i
shmget		117i	29	117i	117i	117i	117i	117i	117i	117i
shutdown		102s	48	102s	102s	102s	102s	102s	102s	102s
sigaction		67	none	67	67	none	67	67	67	67
sigaltstack		186	131	186	185	185	185	186	186	186
signal		48	none	48	48	48	48	48	48	48
sigpending		73	none	73	73	none	73	73	73	73
sigprocmask		126	none	126	126	none	126	126	126	126
sigreturn		119	none	119	119	none	119	119	119	119
sigsuspend		72	none	72	72	none	72	72	72	72
socket		102s	41	102s	102s	102s	102s	102s	102s	102s
socketcall		102	none	102	102	102	102	102	102	102
socketpair		102s	53	102s	102s	102s	102s	102s	102s	102s
ssetmask		69	none	69	69	69	69	none	none	none
stat		18 106	4	18 106 195	18 106 195	106	106 195	106 195	106	106 195
statfs		99	137	99 268	99 252	99 252	99 252	99 265	99 265	99 265
stime		25	none		25	25	25	25	none	25
swapcontext		none	none	none	249	249	249	none	none	none
swapoff		115	168	115	115	115	115	115	115	115
swapon		87	167	87	87	87	87	87	87	87

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
symlink		83	88	83	83	83	83	83	83	83
sync		36	162	36	36	36	36	36	36	36
sysctl		149	156	149	149	149	149	149	149	149
sysfs		135	139	135	135	135	135	135	135	135
sysinfo		116	99	116	116	116	116	116	116	116
syslog		103	103	103	103	103	103	103	103	103
tgkill		270	234	270	250	250	250	241	241	241
time		13	201	13	13	13	13	13	none	13
timer_create		259	222	259	240	240	240	254	254	254
timer_delete		263	226	263	244	244	244	258	258	258
timer_gettime		261	224	261	242	242	242	256	256	256
timer_getoverrun		262	225	262	243	243	243	257	257	257
timer_settime		260	223	260	241	241	241	255	255	255
times		43	100	43	43	43	43	43	43	43
tkill		238	200	238	208	208	208	237	237	237
truncate		92 193	76	92 193	92 193	92	92 193	92 193	92	92 193
tux		222	184	none	none	225	225	242	242	none
umask		60	95	60	60	60	60	60	60	60

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
umount		22 52	166	22 52	22 52	52	22 52	22 52	22 52	22 52
uname		59 109 122	63	59 109 122	59 109 122	122	59 109 122	122	122	122
unlink		10	87	10	10	10	10	10	10	10
uselib		86	134	86	86	86	86	86	86	86
ustat		62	136	62	62	62	62	62	62	62
utime		30	132	30	30	30	30	30	30	30
utimes		271	235	271	251	251	251	none	none	none
vfork		190	58	190	189	189	189	190	190	190
vhangup		111	153	111	111	111	111	111	111	111
vm86		113 166	none	none	none	none	none	none	none	none
wait4		114	61	114	114	114	114	114	114	114
waitid		284	248	284	none	none	none	none	none	none
waitpid		7	none	7	7	7	7	none	none	none
write		4	1	4	4	4	4	4	4	4

Syscall	Subsystem Interfaces	i386	x86_64	x86_64e32	ppc	ppc64	ppc64e32	s390/31	s390/64	s390/64e31
writev		146	20	146	146	146	146	146	146	146

Trusted Programs in RHEL-EAL4

The list of trusted applications is created by considering all applications that fall in at least one of the following categories:
- SUID root bit for the corresponding application is enabled
- applications that access any of the trusted databases according to the administrator guidance document, especially the Security Guide
- applications that either implement or provide access to any TOE security function

File Name	Test Case(s)	TSF	Privilege	HLD Module	Documentation Path	Package
/bin/date	ltp-full/testcases/kernel/syscalls/settimeofday/settimeofday01.c ltp-full/testcases/kernel/syscalls/settimeofday/settimeofday02.c ltp-full/testcases/audit/syscalls/settimeofday_test.c	SM.5,TP.4	addtl	SM	/usr/share/man/man1/date.1.gz	coreutils-5.2.1-31.1
/bin/login	/ltp-full/testcases/audit/pam_laf/login_test.c /rhcc/eal4/tests/manual/login.bash	IA.1, IA.2, IA.3, IA.5, TP.4	addtl	IA	/usr/share/man/man1/login.1.gz	util-linux-2.12a-16
/bin/ping	ltp-full/testcases/network/tcp_cmds/ping/ping01	TP.4	no	NA	/usr/share/man/man8/ping.8.gz	iputils-20020927-16
/bin/su	ltp-full/testcases/commands/su/su01 ltp-full/testcases/audit/pam_laf/su_test.c	IA.1, IA.2, IA.3, IA.4, IA.5, SM.1, TP.4	addtl	IA	/usr/share/man/man1/su.1.gz	coreutils-5.2.1-31.1
/sbin/agetty	/rhcc/eal4/tests/manual/serialterm.txt	IA.3, TP.4	no	IA	/usr/share/man/man8/agetty.8.gz	util-linux-2.12a-16
/sbin/auditd	ltp-full/testcases/audit/auditd/auditd_test.sh Implicitly all audit tests exercise auditd.	AU.1, AU.2, SM.4, TP.4, TP.5	admin only	UA	/usr/share/man/man8/auditd.8.gz	audit-0.8
/sbin/init	/rhcc/eal4/tests/manual/inittab.bash	TP.4	admin only	SI	/usr/share/man/man8/init.8.gz	SysVinit-2.85-34
/sbin/mingetty	/rhcc/eal4/tests/manual/mingetty.bash	IA.3, TP.4	no	IA	/usr/share/man/man8/mingetty.8.gz	mingetty-1.07-3
/usr/bin/amtu	/rhcc/eal4/tests/amtu/src/amtu*.c	TP.4, TP.7	admin only	SM	/usr/share/man/man8/amtu.8.gz	amtu-1.0.2-2.EL4
/usr/bin/at	/rhcc/eal4/tests/misc_test/at_crontab/runme.sh ltp-full/testcases/audit/trustedpgms/at_test.c	TP.4	no	BP	/usr/share/man/man1/at.1.gz	at-3.1.8-77
/usr/bin/chage	/rhcc/eal4/tests/misc_test/databases/shadow01	IA.1, TP.4	addtl	SM	/usr/share/man/man1/chage.1.gz	shadow-utils-4.0.3-41
/usr/bin/chfn	/rhcc/eal4/tests/misc_test/databases/passwd02	IA.1, TP.4	addtl	SM	/usr/share/man/man1/chfn.1.gz	util-linux-2.12a-16
/usr/bin/chsh	/rhcc/eal4/tests/misc_test/databases/passwd03	IA.1, TP.4	addtl	SM	/usr/share/man/man1/chsh.1.gz	util-linux-2.12a-16
/usr/bin/crontab	ltp-full/testcases/commands/cron/cron02 ltp-full/testcases/audit/trustedpgms/crontab_test.c	TP.4	no	BP	/usr/share/man/man1/crontab.1.gz	vixie-cron-4.1-20
/usr/bin/gpasswd	ltp-full/testcases/audit/trustedpgms/gpasswd_test.c	IA.1, TP.4	addtl	SM	/usr/share/man/man1/gpasswd.1.gz	shadow-utils-4.0.3-41
/usr/bin/passwd	/rhcc/eal4/tests/misc_test/databases/pam01 /rhcc/eal4/tests/misc_test/ddatabases/passwd01 /rhcc/eal4/tests/misc_test/databases/passwd02 /rhcc/eal4/tests/misc_test/databases/passwd03 ltp-full/testcases/audit/trustedpgms/passwd_test.c	IA.1, TP.4	addtl	IA	/usr/share/man/man1/passwd.1.gz	passwd-0.68-10
/usr/bin/openssl	/rhcc/eal/4/tests/OpenSSL/testcases/openssl/openssl01	SC.1, TP.4	no	NA	/usr/share/man/man1/openssl.1ssl.gz	openssl-0.9.7a-43.1
/usr/bin/ssh	ltp-full/testcases/network/tcp_cmds/ssh/ssh01 ltp-full/testcases/network/tcp_cmds/ssh/ssh02 ltp-full/testcases/network/tcp_cmds/ssh/ssh03 /rhcc/eal4/tests/misc_test/eal/ssh04 ltp-full/testcases/audit/pam_laf/sshd_test.c	SC.1, TP.4	addtl	NA	/usr/share/man/man1/ssh.1.gz	openssh-3.9p1-8
/usr/sbin/stunnel	/rhcc/eal4/tests/OpenSSL/testcases/openssl/openssl01	SC.1, TP.4	no	NA	/usr/share/man/man8/stunnel.8.gz	stunnel-4.05-3
/usr/sbin/atd	/rhcc/eal4/tests/misc_test/at_crontab/runme.sh ltp-full/testcases/audit/trustedpgms/at_test.c	TP.4	admin only	BP	/usr/share/man/man8/atd.8.gz	at-3.1.8-77
/usr/sbin/auditctl	ltp-full/testcases/audit/filters/filter1_test.c ltp-full/testcases/audit/filters/filter2_test.c ltp-full/testcases/audit/filters/filter3_test.c ltp-full/testcases/audit/filters/filter4_test.c ltp-full/testcases/audit/filters/filter5_test.c ltp-full/testcases/audit/filters/filter6_test.c ltp-full/testcases/audit/filters/filter_arch1_test.c ltp-full/testcases/audit/filters/filter_arch2_test.c auditctl implicitly tested in all ltp-full/testcases/audit/sycalls tests.	AU.2, TP.4, TP.5	admin only	UA	/usr/share/man/man8/auditctl.8.gz	audit-0.8
/usr/sbin/ausearch	ltp-full/testcases/audit/audit_tools/ausearch_test	AU.4, TP.4	admin only	UA	/usr/share/man/man8/ausearch.8.gz	audit-0.8
/usr/sbin/crond	ltp-full/testcases/commands/cron/cron02 ltp-full/testcases/audit/trustedpgms/crontab_test.c	TP.4	admin only	BP	/usr/share/man/man8/cron.8.gz	vixie-cron-4.1-20
/usr/sbin/groupadd	/rhcc/eal4/tests/misc_test/databases/group01 ltp-full/testcases/audit/trustedpgms/groupadd_test.c	SM.3, TP.4	admin only	SM	/usr/share/man/man8/groupadd.8.gz	shadow-utils-4.0.3-41
/usr/sbin/groupdel	/rhcc/eal4/tests/misc_test/databases/group01 ltp-full/testcases/audit/trustedpgms/groupadd_test.c	SM.3, TP.4	admin only	SM	/usr/share/man/man8/groupdel.8.gz	shadow-utils-4.0.3-41
/usr/sbin/groupmod	/rhcc/eal4/tests/misc_test/databases/group01 ltp-full/testcases/audit/trustedpgms/groupmod_test.c	SM.3, TP.4	admin only	SM	/usr/share/man/man8/groupmod.8.gz	shadow-utils-4.0.3-41
/usr/sbin/sshd	ltp-full/testcases/network/tcp_cmds/ssh/ssh01 ltp-full/testcases/network/tcp_cmds/ssh/ssh02 ltp-full/testcases/network/tcp_cmds/ssh/ssh03 ltp-full/testcases/audit/pam_laf/sshd_test.c	IA.1, IA.2, IA.3, IA.5, SC.1, TP.4	admin only	NA	/usr/share/man/man8/sshd.8.gz	openssh-3.9p1-8
/usr/sbin/useradd	"misc_test/databases/passwd01 misc_test/databases/passwd02 /rhcc/eal4/tests/misc_test/databases/passwd03 /rhcc/eal4/tests/misc_test/databases/pam01 /rhcc/eal4/tests/misc_test/databases/shadow01 /rhcc/eal4/tests/misc_test/databases/group01 /rhcc/eal4/tests/misc_test/databases/ftpusers01 ltp-full/testcases/audit/trustedpgms/useradd_test.c	SM.3, TP.4	admin only	SM	/usr/share/man/man8/useradd.8.gz	shadow-utils-4.0.3-41
/usr/sbin/userdel	/rhcc/eal4/tests/misc_test/databases/passwd01 /rhcc/eal4/tests/misc_test/databases/passwd02 /rhcc/eal4/tests/misc_test/databases/passwd03 /rhcc/eal4/tests/misc_test/databases/pam01 /rhcc/eal4/tests/misc_test/databases/shadow01 /rhcc/eal4/tests/misc_test/databases/group01 /rhcc/eal4/tests/misc_test/databases/ftpusers01 ltp-full/testcases/audit/trustedpgms/userdel_test.c	SM.3, TP.4	admin only	SM	/usr/share/man/man8/userdel.8.gz	shadow-utils-4.0.3-41
/usr/sbin/usermod	/rhcc/eal4/tests/misc_test/databases/group01 ltp-full/testcases/audit/trustedpgms/usermod_test.c	SM.3, TP.4	admin only	SM	/usr/share/man/man8/usermod.8.gz	shadow-utils-4.0.3-41
/usr/sbin/vsftpd	/rhcc/eal4/tests/misc_test/databases/ftpusers01 ltp-full/testcases/network/tcp_cmds/ftp/ftp02 ltp-full/testcases/network/tcp_cmds/ftp/ftp03 ltp-full/testcases/network/tcp_cmds/ftp/ftp04 ltp-full/testcases/network/tcp_cmds/ftp/ftp05 ltp-full/testcases/audit/pam_laf/vsftpd_test.c	IA.1, IA.2, IA.3, IA.5, TP.4	admin only	NA	/usr/share/man/man8/vsftpd.8.gz	vsftpd-2.0.1-5
/sbin/hwclock	ltp-full/testcases/audit/trustedpgms/hwclock.c	SM.5,TP.4	addtl	SM	/usr/share/man/man8/hwclock.8.gz	util-linux-2.12a-16.

The trusted database list is taken from the Security Target's TOE summary specification, in particular the table found in definition of TP.5.This table is extended with functional specification information in the current mapping table.

File Name	Test case(s)	TSF	HLD Subsystem	Documentation	Package
/etc/at.allow	ltp-full/testcases/commands/at/at_allow01	TP.5	BP	/usr/share/man/man5/at.allow.5.gz	at-3.1.8-77
/etc/at.deny	ltp-full/testcases/commands/at/at_deny01	TP.5	BP	/usr/share/man/man5/at.deny.5.gz	at-3.1.8-77
/etc/auditd.conf	ltp-full/testcases/audit/config/config2_test.c ltp-full/testcases/audit/config/config3_test.c ltp-full/testcases/audit/config/config4_test.c	AU.1, SM.4, TP.5	UA	/usr/share/man/man/auditd.conf.5.gz	audit-0.8
/etc/audit.rules	ltp-full/testcases/audit/config/config1_test.c	AU.1, SM.4, TP.5	UA	/usr/share/man/man8/auditctl.8.gz	audit-0.8
/etc/cron.d/*	ltp-full/testcases/commands/cron/cron_dirs_checks01	TP.5	BP	/usr/share/man/man8/cron.8.gz /usr/share/man/man5/crontab.5.gz	vixie-cron-4.1-20
/etc/cron.{ weekly hourly daily monthly}	ltp-full/testcases/commands/cron/cron_dirs_checks01	TP.5	BP	/usr/share/man/man8/cron.8.gz /usr/share/man/man5/crontab.5.gz	crontabs-1.10-7
/etc/crontab	ltp-full/testcases/commands/cron/cron02	TP.5	BP	/usr/share/man/man8/cron.8.gz /usr/share/man/man5/crontab.5.gz	crontabs-1.10-7
/etc/vsftpd.ftpusers	/rhcc/eal4/tests/misc_test/databases/ftpusers01	TP.5	NA	/usr/share/man/man5/ftpusers.5.gz	vsftpd-2.0.1-5
/etc/group	/rhcc/eal4/tests/misc_test/databases/group01	IA.1, IA.3, IA.4, SM.3, TP.5	SM	/usr/share/man/man5/group.5.gz	setup-2.5.37-1
/etc/gshadow		IA.1, IA.3, IA.4, SM.3, TP.5	SM	/usr/share/doc/shadow/HOWTO	setup-2.5.37-1
/etc/hosts	/rhcc/eal4/tests/misc_test/eal/checkaccess.c /etc/hosts	TP.5	NA	/usr/share/man/man5/hosts.5.gz	man-pages-1.67-3
/etc/rc.d/init.d/*	/rhcc/eal4/tests/misc_test/eal/checkaccess.c /etc/init.d	TP.4, TP.5	SI	/usr/share/man/man8/chkconfig.8.gz /usr/share/man/man8/init.8.gz rhel-rg-en.pdf - reference guide	chkconfig-1.3.13.2-1 SysVinit-2.85-34
/etc/rc.d/init.d/auditd	Used in some ltp-full/testcases/audit/syscalls tests	SM.4, TP.4, TP.5	SI / UA	/usr/share/man/man8/auditd.8.gz	audit-0.8
/etc/inittab	/rhcc/eal4/tests/manual/inittab.bash	TP.4, TP.5	SI	/usr/share/man/man5/inittab.5.gz	initscripts-7.93.11.EL-1
/etc/ld.so.conf		TP.4, TP.5	PC / MM	/usr/share/man/man8/ld.so.8.gz	glibc-2.3.4-2
/etc/login.defs	/rhcc/eal4/tests/misc_test/databases/passwd01 /rhcc/eal4/tests/misc_test/databases/passwd02 /rhcc/eal4tests/misc_test/databases/passwd03	IA.1, TP.5	IA	/usr/share/doc/shadow-utils-4.0.3-41/HOWTO	shadow-utils-4.0.3-41
/etc/modprobe.conf		TP.3, TP.5	KM	/usr/share/man/man5/modprobe.5.gz	module-init-tools-3.1-0.pre5.3
/etc/pam.d/*	/rhcc/eal4/misc_test/databases/pam01 ltp-full/testcases/audit/pam_laf/sshd_test.c ltp-full/testcases/audit/pam_laf/vsftpd_test.c"	IA.1, IA.2, IA.3, IA.4, IA.5, TP.5	IA	/usr/share/man/man8/pam.8.gz /usr/share/doc/pam-0.75/txts/pam_wheel. /usr/share/man/pam-0.75/txtspam_nologin. /usr/share/doc/pam-0.75/txts/README /usr/share/doc/pam_passwdqc-0.7.5/README /usr/share/doc/pam-.75/txts/README.pam_rootok /usr/share/doc/pam-0.75/txts/README.pam_securetty /usr/share/doc/pam-0.75/txts/README.pam_stack /usr/share/doc/pam-0.75/txts/README.pam_tally /usr/share/doc/pam-0.75/txts/README.pam_unix	pam-0.78/pam_passwdqc-0.7.5
/etc/passwd	/rhcc/eal4/tests/misc_test/databases/passwd01 /rhcc/eal4/tests/misc_test/databases/passwd02 /rhcc/eal4/tests/misc_test/databases/passwd03	IA.1, IA.2, IA.3, IA.4, IA.5, SM.3, TP.5	IA	/usr/share/man/man5/passwd.5.gz	setup-2.5.37-1
/etc/securetty	/rhcc/eal4/tests/manual/serialterm.txt	IA.3, TP.5	IA	/usr/share/man/man5/securetty.5.gz	setup-2.5.37-1
/etc/shadow	/rhcc/eal4/tests/misc_test/databases/shadow01	IA.1, IA.2, IA.3, IA.4, IA.5, SM.3, TP.5	IA	/usr/share/man/man5/shadow.5.gz	setup-2.5.37-1
/etc/ssh/sshd_config	ltp-full/testcases/network/tcp_cmds/ssh/ssh01 ltp-full/testcases/network/tcp_cmds/ssh/ssh02 ltp-full/testcases/network/tcp_cmds/ssh/ssh03 ltp-full/testcases/audit/pam_laf/sshd_test.c	TP.5, SC.1	NA	/usr/share/man/man5/sshd_config.5.gz	openssh-server-3.8.1p1-4
/etc/sysconfig/*	/rhcc/eal4/tests/misc_test/eal/checkaccess.c /etc/sysconfig	TP.5	SI	rhel-rg-en.pdf - reference guide	
/etc/vsftpd/vsftpd.conf	/rhcc/eal4/tests/misc_test/databases/ftpusers01	TP.4, TP.5	NA	/usr/share/man/man5/vsftpd.conf.5.gz	vsftpd-2.0.1-5
/etc/stunnel/stunnel.conf	/rhcc/eal4/tests/misc_test/Openssl/openssl01	TP.5,SC.1	NA	/usr/share/man/man8/stunnel.8.gz	stunnel-4.05-3
/etc/stunnel/stunnel.pem	/r/hcc/eal4/tests/misc_test/Openssl/openssl01	TP.5,SC.1	NA	/usr/share/man/man8/stunnel.8.gz	stunnel-4.05-3
/var/log/lastlog	/rhcc/eal4/tests/misc_test/databases/lastlog01	IA.1, IA.2, IA.3, IA.4, TP.5	IA	/usr/share/man/man5/lastlog.5.gz	setup-2.5.37-1
/var/log/faillog	/rhcc/eal4/tests/misc_test/databases/faillog01	IA.1, IA.2, IA.3, IA.4, TP.5	IA	/usr/share/man/man8/faillog.5.gz	shadow-utils-4.0.3-41
/var/spool/at	/rhcc/eal4/tests/misc_test/at_crontab/runme.sh	TP.5	BP	/usr/share/man/man8/atd.8.gz	at-3.1.8-77
/var/spool/cron/tabs/root	ltp-full/testcases/commands/cron/cron_dirs_checks01	TP.5	BP	/usr/share/man/man8/cron.8.gz,	vixie-cron-4.1-20
/etc/cron.allow	ltp-full/testcases/commands/cron/cron_allow01	TP.5	BP	/usr/share/man/man1/crontab.1.gz	vixie-cron-4.1-20
/etc/cron.deny	ltp-full/testcases/commands/cron/cron_deny01	TP.5	BP	/usr/share/man/man1/crontab.1.gz	vixie-cron-4.1-20
/etc/security/opasswd	/rhcc/eal4/tests/misc_test/databases/pam01 /rhcc/eal4/tests/misc_testdatabases/shadow01	IA.1, IA.2, TP.5	IA	/usr/share/doc/pam-0.75/txts/README.pam_unix	
/etc/localtime	/rhcc/eal4/tests/misc_test/eal/checkaccess.c /etc/localtime	TP.5	SM	/usr/share/doc/glibc-2.3.4-2 /usr/share/doc/initscripts-7.93.11.EI/sysconfig.txt	glibc-2.3.4-2
/etc/sysctl.conf	/rhcc/eal4/tests/misc_test/eal/checkaccess.c /etc/sysctl.conf	TP.5	SI	/usr/share/man/man5/sysctl.conf.5.gz	initscripts-7.93.11.EL-1

Function Name	Test Case(s)	TSF	HLD Subsys
Discretionary Access Control	/rhcc/eal4/tests/misc_test/permission/fileperm.c /rhcc/eal4/tests/misc_test/permission/dirperm.c /rhcc/eal4/tests/misc_test/permission/shmperm.c /rhcc/eal4/tests/misc_test/permission/msqperm.c /rhcc/eal4/tests/misc_test/permission/semperm.c /rhcc/eal4/tests/misc_test/permission/unixdomainsocketperm.c /rhcc/eal4/tests/misc_test/permission/devfileperm.c /rhcc/eal4/tests/misc_test/permission/namedpipes_fifoperm.c /rhcc/eal4/tests/misc_test/permission/procperm.c /rhcc/eal4/tests/misc_test/permission/suid_sgid.c	DA.1, DA.2, DA.4	FS IPC
Object Reuse	/rhcc/eal4/tests/misc_test/object_reuse/objreuse-brk.c /rhcc/eal4/tests/misc_test/object_reuse/objreuse-ftruncate.c /rhcc/eal4/tests/misc_test/object_reuse/objreuse-lseek.c /rhcc/eal4/tests/misc_test/object_reuse/objreuse-shm.c /rhcc/eal4/tests/misc_test/object_reuse/objreuse-msg.c /rhcc/eal4/tests/misc_test/object_reuse/objreuse-sem.c /rhcc/eal4/tests/misc_test/object_reuse/objreuse-mmap.c	OR.1, OR.2, OR.3	MM FS IPC
OpenSSL Interoperability Tests	/rhcc/eal4/tests/manual/openssl.bash RC4, TDES, AES128, AES256 algorithms only	SC.1	NA

Implicit Tests

The audit record format testing (AU.3) is addressed implicitly by all tests located in ltp-full/testcases/audit

Devices (block and character) are generally tested implicitly by all test suites through disk and tty access for test data and result logging.