KLEE Implementation Research

Below is a list of all the POSIX functions along with System Calls as well as String functions. All the functions are color coded and each of the color code is explain below:

- Red: These functions are implemented in Klee and each of these have a separate file (can be found under KLEE/runtime/klee-lbic)
- Green: These functions/system calls are implemented in KLEE but not within their own file. These functions can be found in the files under KLEE/runtime/POSIX/
- Blue: These are the functions/system calls which have stub implementation in KLEE.
- Blue: Same as green. The only difference is that they are inside stubs.c

POSIX Functions

int open (const char *filename, int flags[, mode_t mode])
int open64 (const char *filename, int flags[, mode_t mode])
int creat (const char *filename, mode_t mode)
int creat64 (const char *filename, mode_t mode)
int close (int filedes)
ssize_t read (int filedes, void *buffer, size_t size)
ssize_t pread (int filedes, void *buffer, size_t size, off_t offset)
ssize_t pread64 (int filedes, void *buffer, size_t size, off64_t offset)
ssize_t pread64 (int filedes, void *buffer, size_t size, off64_t offset)
ssize_t pwrite (int filedes, const void *buffer, size_t size, off_t offset)
ssize_t pwrite64 (int filedes, const void *buffer, size_t size, off64_t offset)
FILE * fdopen (int filedes, const char *opentype)
int fileno (FILE *stream)
int fileno_unlocked (FILE *stream)
int ioctl (int filedes, int command, …)
ssize_t readv (int filedes, const struct iovec *vector, int count)
ssize_t writev (int filedes, const struct iovec *vector, int count) xxxxx
void * mmap (void *address, size_t length, int protect, int flags, int filedes, off_t offset)
void * mmap64 (void *address, size_t length, int protect, int flags, int filedes, off64_t offset)
int munmap (void *addr, size_t length)
int msync (void *address, size_t length, int flags)
void * mremap (void *address, size_t length, size_t new_length, int flag)
int madvise (void *addr, size_t length, int advice)
int symlink (const char *oldname, const char *newname)
ssize_t readlink (const char *filename, char *buffer, size_t size)
char * canonicalize_file_name (const char *name)
char * realpath (const char *restrict name, char *restrict resolved)
int shm_open (const char *name, int oflag, mode_t mode)
int shm_unlink (const char *name)
int select (int nfds, fd_set *read-fds, fd_set *write-fds, fd_set *except-fds, struct timeval *timeout)
FILE * fopen (const char *filename, const char *opentype)

FILE * fopen64 (const char *filename, const char *opentype)
FILE * freopen (const char *filename, const char *opentype, FILE *stream)
FILE * freopen64 (const char *filename, const char *opentype, FILE *stream)
int __freadable (FILE *stream)
int __fwritable (FILE *stream)
int __freading (FILE *stream)
int __fwriting (FILE *stream)
int fclose (FILE *stream)
int fcloseall (void)
void flockfile (FILE *stream)
int ftrylockfile (FILE *stream)
void funlockfile (FILE *stream)
int __fsetlocking (FILE *stream, int type)
int fputc (int c, FILE *stream) as int _IO_putc(int c, FILE *f)
wint_t fputwc (wchar_t wc, FILE *stream)
int fputc_unlocked (int c, FILE *stream)
wint_t fputwc_unlocked (wchar_t wc, FILE *stream)
int putc (int c, FILE *stream)
wint_t putwc (wchar_t wc, FILE *stream)
int putc_unlocked (int c, FILE *stream)
wint_t putwc_unlocked (wchar_t wc, FILE *stream)
int putchar (int c)
wint_t putwchar (wchar_t wc)
int putchar_unlocked (int c)
wint_t putwchar_unlocked (wchar_t wc)
int fputs (const char *s, FILE *stream)
int fputws (const wchar_t *ws, FILE *stream)
int fputs_unlocked (const char *s, FILE *stream)
int fputws_unlocked (const wchar_t *ws, FILE *stream)
int puts (const char *s)
int putw (int w, FILE *stream)
int fgetc (FILE *stream)
wint_t fgetwc (FILE *stream)
int __fgetc_unlocked (FILE *stream)
```
int __fputc_unlocked(int c, FILE *f);
```
wint_t fgetwc_unlocked (FILE *stream)
int getc (FILE *stream) as int _IO_getc(FILE *f)
wint_t getwc (FILE *stream)
int getc_unlocked (FILE *stream)
wint_t getwc_unlocked (FILE *stream)
int getchar (void)
wint_t getwchar (void)
int getchar_unlocked (void)
wint_t getwchar_unlocked (void)
int getw (FILE *stream)
ssize_t getline (char **lineptr, size_t *n, FILE *stream)
ssize_t getdelim (char **lineptr, size_t *n, int delimiter, FILE *stream)
char * fgets (char *s, int count, FILE *stream)
wchar_t * fgetws (wchar_t *ws, int count, FILE *stream)

char * fgets_unlocked (char *s, int count, FILE *stream)
wchar_t * fgetws_unlocked (wchar_t *ws, int count, FILE *stream)
char * gets (char *s)
int ungetc (int c, FILE *stream)
wint_t ungetwc (wint_t wc, FILE *stream)
size_t fread (void *data, size_t size, size_t count, FILE *stream)
size_t fread_unlocked (void *data, size_t size, size_t count, FILE *stream)
size_t fwrite (const void *data, size_t size, size_t count, FILE *stream)
size_t fwrite_unlocked (const void *data, size_t size, size_t count, FILE *stream)
int printf_size (FILE *fp, const struct printf_info *info, const void *const *args)
int printf_size_info (const struct printf_info *info, size_t n, int *argtypes)
int fflush (FILE *stream)
int fflush_unlocked (FILE *stream)
void _flushlbf (void)
void __fpurge (FILE *stream)
int setvbuf (FILE *stream, char *buf, int mode, size_t size)
void setbuf (FILE *stream, char *buf)
void setbuffer (FILE *stream, char *buf, size_t size)
void setlinebuf (FILE *stream)
int __flbf (FILE *stream)
size_t __fbufsize (FILE *stream)
size_t __fpending (FILE *stream)
int fgetpos (FILE *stream, fpos_t *position)
int fgetpos64 (FILE *stream, fpos64_t *position)
int fsetpos (FILE *stream, const fpos_t *position)
int fsetpos64 (FILE *stream, const fpos64_t *position)
long int ftell (FILE *stream)
off_t ftello (FILE *stream)
off64_t ftello64 (FILE *stream)
int fseek (FILE *stream, long int offset, int whence)
int fseeko (FILE *stream, off_t offset, int whence)
int fseeko64 (FILE *stream, off64_t offset, int whence)

void rewind (FILE *stream)
int feof (FILE *stream)
int feof_unlocked (FILE *stream)
int ferror (FILE *stream)
int ferror_unlocked (FILE *stream)
void sync (void)
int fsync (int fildes)
int fdatasync (int fildes)
int utimes (const char *filename, const struct timeval tvp[2])
char * getcwd (char *buffer, size_t size)
mode_t umask (mode_t mask)

int printf (const char *template, …)
int wprintf (const wchar_t *template, …)
int fprintf (FILE *stream, const char *template, …)

int fwprintf (FILE *stream, const wchar_t *template, …)
int sprintf (char *s, const char *template, …)
int swprintf (wchar_t *s, size_t size, const wchar_t *template, …)
int snprintf (char *s, size_t size, const char *template, …)
int asprintf (char **ptr, const char *template, …)
int obstack_printf (struct obstack *obstack, const char *template, …)
int vprintf (const char *template, va_list ap)
int vwprintf (const wchar_t *template, va_list ap)
int vfprintf (FILE *stream, const char *template, va_list ap)
int vfwprintf (FILE *stream, const wchar_t *template, va_list ap)
int vsprintf (char *s, const char *template, va_list ap)
int vswprintf (wchar_t *s, size_t size, const wchar_t *template, va_list ap)
int vsnprintf (char *s, size_t size, const char *template, va_list ap)
int vasprintf (char **ptr, const char *template, va_list ap)
int obstack_vprintf (struct obstack *obstack, const char *template, va_list ap)
size_t parse_printf_format (const char *template, size_t n, int *argtypes)

int scanf (const char *template, …)
int wscanf (const wchar_t *template, …)
int fscanf (FILE *stream, const char *template, …)
int fwscanf (FILE *stream, const wchar_t *template, …)
int sscanf (const char *s, const char *template, …)
int swscanf (const wchar_t *ws, const wchar_t *template, …)
int vscanf (const char *template, va_list ap)
int vwscanf (const wchar_t *template, va_list ap)
int vfscanf (FILE *stream, const char *template, va_list ap)
int vfwscanf (FILE *stream, const wchar_t *template, va_list ap)
int vsscanf (const char *s, const char *template, va_list ap)
int vswscanf (const wchar_t *s, const wchar_t *template, va_list ap)
FILE * fmemopen (void *buf, size_t size, const char *opentype)
FILE * open_memstream (char **ptr, size_t *sizeloc)
int fmtmsg (long int classification, const char *label, int severity, const char *text, const char *action, const char *tag)
int addseverity (int severity, const char *string)
void clearerr (FILE *stream)
void clearerr_unlocked (FILE *stream)

void * malloc (size_t size)
void free (void *ptr)
void cfree (void *ptr)
void * realloc (void *ptr, size_t newsize)
void * calloc (size_t count, size_t eltsize)
void * aligned_alloc (size_t alignment, size_t size)
void * memalign (size_t boundary, size_t size)
int posix_memalign (void **memptr, size_t alignment, size_t size)
void * valloc (size_t size)
int mallopt (int param, int value)
int mcheck (void (*abortfn) (enum mcheck_status status))

enum mcheck_status mprobe (void *pointer)

String

size_t strlen (const char *s)
size_t wcslen (const wchar_t *ws)
size_t strnlen (const char *s, size_t maxlen)
size_t wcsnlen (const wchar_t *ws, size_t maxlen)
void * memcpy (void *restrict to, const void *restrict from, size_t size) -> Also found under Intrinsic in Klee
wchar_t * wmemcpy (wchar_t *restrict wto, const wchar_t *restrict wfrom, size_t size)
void * memmove (void *to, const void *from, size_t size) -> Also found under Intrinsic in Klee

wchar_t * wmemmove (wchar_t *wto, const wchar_t *wfrom, size_t size)
void * memccpy (void *restrict to, const void *restrict from, int c, size_t size)
void * memset (void *block, int c, size_t size)
wchar_t * wmemset (wchar_t *block, wchar_t wc, size_t size)
wchar_t * wcscpy (wchar_t *restrict wto, const wchar_t *restrict wfrom)
char * strdup (const char *s)
wchar_t * wcsdup (const wchar_t *ws)
char * stpcpy (char *restrict to, const char *restrict from)
wchar_t * wcpcpy (wchar_t *restrict wto, const wchar_t *restrict wfrom)
char * strdupa (const char *s)
void bcopy (const void *from, void *to, size_t size)
void bzero (void *block, size_t size)
char * strcat (char *restrict to, const char *restrict from)
wchar_t * wcscat (wchar_t *restrict wto, const wchar_t *restrict wfrom)
char * strncpy (char *restrict to, const char *restrict from, size_t size)
wchar_t * wcsncpy (wchar_t *restrict wto, const wchar_t *restrict wfrom, size_t size)
char * strndup (const char *s, size_t size)
char * strndupa (const char *s, size_t size)
char * stpncpy (char *restrict to, const char *restrict from, size_t size)
wchar_t * wcpncpy (wchar_t *restrict wto, const wchar_t *restrict wfrom, size_t size)
char * strncat (char *restrict to, const char *restrict from, size_t size)
wchar_t * wcsncat (wchar_t *restrict wto, const wchar_t *restrict wfrom, size_t size)
int memcmp (const void *a1, const void *a2, size_t size)
int wmemcmp (const wchar_t *a1, const wchar_t *a2, size_t size)
int strcmp (const char *s1, const char *s2)
int wcscmp (const wchar_t *ws1, const wchar_t *ws2)
int strcasecmp (const char *s1, const char *s2)
int wcscasecmp (const wchar_t *ws1, const wchar_t *ws2)
int strncmp (const char *s1, const char *s2, size_t size)
int wcsncmp (const wchar_t *ws1, const wchar_t *ws2, size_t size)
int strncasecmp (const char *s1, const char *s2, size_t n)
int wcsncasecmp (const wchar_t *ws1, const wchar_t *s2, size_t n)
int strverscmp (const char *s1, const char *s2)
int bcmp (const void *a1, const void *a2, size_t size)
void * memchr (const void *block, int c, size_t size)
wchar_t * wmemchr (const wchar_t *block, wchar_t wc, size_t size)

void * rawmemchr (const void *block, int c)
void * memrchr (const void *block, int c, size_t size)
char * strchr (const char *string, int c)
wchar_t * wcschr (const wchar_t *wstring, int wc)
char * strchrnul (const char *string, int c)
wchar_t * wcschrnul (const wchar_t *wstring, wchar_t wc)
char * strrchr (const char *string, int c)
wchar_t * wcsrchr (const wchar_t *wstring, wchar_t c)
char * strstr (const char *haystack, const char *needle)
wchar_t * wcsstr (const wchar_t *haystack, const wchar_t *needle)
wchar_t * wcswcs (const wchar_t *haystack, const wchar_t *needle)
char * strcasestr (const char *haystack, const char *needle)
void * memmem (const void *haystack, size_t haystack-len,
const void *needle, size_t needle-len)
size_t strspn (const char *string, const char *skipset)
size_t wcsspn (const wchar_t *wstring, const wchar_t *skipset)
size_t strcspn (const char *string, const char *stopset)
size_t wcscspn (const wchar_t *wstring, const wchar_t *stopset)
char * strpbrk (const char *string, const char *stopset)
wchar_t * wcspbrk (const wchar_t *wstring, const wchar_t *stopset)
char * index (const char *string, int c)
char * rindex (const char *string, int c)
char * strtok (char *restrict newstring, const char *restrict delimiters)
wchar_t * wcstok (wchar_t *newstring, const wchar_t *delimiters, wchar_t **save_ptr)char *
strtok_r (char *newstring, const char *delimiters, char **save_ptr)
char * strsep (char **string_ptr, const char *delimiter)
char * basename (char *path)
char * dirname (char *path)
char * strfry (char *string)
void * memfrob (void *mem, size_t length)
char * l64a (long int n)
long int a64l (const char *string)
int strcoll (const char *s1, const char *s2)
int wcscoll (const wchar_t *ws1, const wchar_t *ws2)
size_t strxfrm (char *restrict to, const char *restrict from, size_t size)
size_t wcsxfrm (wchar_t *restrict wto, const wchar_t *wfrom, size_t size)

long int strtol (const char *restrict string, char **restrict tailptr, int base)
unsigned long int strtoul (const char *retrict string, char **restrict tailptr, int base)
int atoi (const char *string)
int tolower (int c)
int toupper (int c)

void abort (void)
int atexit (void (*function) (void))

unsigned long htonl(unsigned long val);

System Calls

```
int openat(int dirfd, const char *pathname, int flags, mode_t mode);
int futimesat(int dirfd, const char *pathname, const struct timeval times[2]);
off_t lseek(int fd, off_t offset, int whence);
int fstatat(int dirfd, const char *pathname, struct stat *buf,
            int flags);
int stat(const char *path, struct stat *buf);
int lstat(const char *path, struct stat *buf);
int fstat(int fd, struct stat *buf);
int chdir(const char *path);
int fchdir(int fd);
int chmod(const char *pathname, mode_t mode);
int fchmod(int fd, mode_t mode);
int chown(const char *pathname, uid_t owner, gid_t group);
int fchown(int fd, uid_t owner, gid_t group);
int lchown(const char *pathname, uid_t owner, gid_t group);
int ftruncate(int fd, off_t length);
int getdents(unsigned int fd, struct dirent *dirp, unsigned int count);
int fcntl (int filedes, int command, …)
int statfs(const char *path, struct statfs *buf);
int fstatfs(int fd, struct statfs *buf);
int rmdir(const char *pathname);
int unlink(const char *file);
int unlinkat(int dirfd, const char *pathname, int flags);
int dup (int old)
int dup2 (int old, int new)
int kill(pid_t pid, int sig)
int setjmp(jmp_buff env);
void longjmp(jmp_buf env, int val)
pid_t fork(void)
pid_t vfork(void)
char *getenv(const char *name)

int getfscreatecon(char **con);
int setfscreatecon(char * context);
int setfilecon(const char *path, char * con);
int lsetfilecon(const char *path, char * con);
int fsetfilecon(int fd, char * con);
void freecon(char * con);
void freeconary(char **con);
```

```
int __syscall_rt_sigaction(int signum, const struct sigaction *act, struct sigaction *oldact, size_t
_something);
int sigaction(int signum, const struct sigaction *act,
              struct sigaction *oldact);
int sigprocmask(int how, const sigset_t *set, sigset_t *oldset);
int __socketcall(int type, int *args);
int mkdir(const char *pathname, mode_t mode);
int mkfifo(const char *pathname, mode_t mode);
int mknod(const char *pathname, mode_t mode, dev_t dev);
int pipe(int filedes[2]);
int link(const char *oldpath, const char *newpath);
int rename(const char *oldpath, const char *newpath);
int nanosleep(const struct timespec *req, struct timespec *rem);
int clock_gettime(clockid_t clk_id, struct timespec *res);
int clock_settime(clockid_t clk_id, const struct timespec *res);
time_t time(time_t *t);
clock_t times(struct tms *buf);
struct utmpx *getutxent(void);
void setutxent(void);
void endutxent(void);
int utmpxname(const char *file);
int euidaccess(const char *pathname, int mode);
int eaccess(const char *pathname, int mode);
int group_member (gid_t __gid);
int utime(const char *filename, const struct utimbuf *buf);
int futimes(int fd, const struct timeval times[2]);
unsigned int gnu_dev_major(unsigned long long int __dev);
unsigned int gnu_dev_minor(unsigned long long int __dev);
unsigned long long int gnu_dev_makedev(unsigned int __major, unsigned int __minor);
int getloadavg(double loadavg[], int nelem) __attribute__((weak));
pid_t wait(int *status);
pid_t wait3(int *status, int options, struct rusage *rusage);
pid_t wait4(pid_t pid, int *status, int options, struct rusage *rusage);
pid_t waitpid(pid_t pid, int *status, int options);
pid_t waitid(idtype_t idtype, id_t id, siginfo_t *infop, int options);
int acl_delete_def_file(const char *path_p);
int acl_extended_file(const char path_p);
int acl_entries(acl_t acl);
acl_t acl_from_mode(mode_t mode);
acl_t acl_get_fd(int fd);
acl_t acl_get_file(const char *pathname, acl_type_t type);
int acl_set_fd(int fd, acl_t acl);
int acl_set_file(const char *path_p, acl_type_t type, acl_t acl);
int acl_free(void *obj_p);
int mount(const char *source, const char *target, const char *filesystemtype, unsigned long
mountflags, const void *data);
int umount(const char *target);
int umount2(const char *target, int flags);
int swapon(const char *path, int swapflags);
```

```
int swapoff(const char *path);
int setgid(gid_t gid);
int setgroups(size_t size, const gid_t *list);
int sethostname(const char *name, size_t len);
int setpgid(pid_t pid, pid_t pgid);
int setpgrp(void);
int setpriority(__priority_which_t which, id_t who, int prio);
int setresgid(gid_t rgid, gid_t egid, gid_t sgid);
int setresuid(uid_t ruid, uid_t euid, uid_t suid);
int setrlimit(__rlimit_resource_t resource, const struct rlimit *rlim);
int setrlimit64(__rlimit_resource_t resource, const struct rlimit64 *rlim);
pid_t setsid(void);
int setuid(uid_t uid);
int reboot(int flag);
int mlock(const void *addr, size_t len);
int munlock(const void *addr, size_t len);
int pause(void);
ssize_t readahead(int fd, off64_t *offset, size_t count);

/* This need to be weak because uclibc wants to define them as well,
   but we will want to make sure a definition is around in case we
   don't link with it. */
int execl(const char *path, const char *arg, ...);
int execlp(const char *file, const char *arg, ...);
int execle(const char *path, const char *arg, ...);
int execv(const char *path, char *const argv[]);
int execvp(const char *file, char *const argv[]);
int execve(const char *file, char *const argv[], char *const envp[]);
```