

# C++ (Cpp) utimensat Examples



WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=11079&source=23669231040574275666f21078051R1521071661f7-4141ps76372F2Fwww.directrelief.org%2Femergency%2Fukra  
of examples.

**Programming Language:** C++ (Cpp)

**Method/Function:** utimensat

**Examples at hotexamples.com:** 30

[Facebook](#)[Twitter](#)[Print](#)[Email](#)[Pinterest](#)[More](#) **Direct Relief****WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.**

ebfd9c1af4ee&id=110798&img\_src=23769231040574275666f21078051R1521071661f7-4141ps76372F2Fwww.directrelief.org%2Femergency%2Fukra

**EXAMPLE #1**

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x2930e>

[Show file](#)

X

**File:** compress.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fbeastix%2Fcompress.c>)

**Project:** beastix/bsdtools (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fbeastix%2Fbsdtools>)



 **Direct Relief**

**WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.**

<https://www.directrelief.org/emergency/fundraising>

X

```

360 static void
361 setfile(const char *name, struct stat *fs)
362 {
363     static struct timespec tspec[2];
364
365     fs->st_mode &= S_ISUID|S_ISGID|S_IRWXU|S_IRWXG|S_IRWXO;
366
367     tspec[0] = fs->st_atim;
368     tspec[1] = fs->st_mtim;
369     if (utimensat(AT_FDCWD, name, tspec, 0))
370         cwarn("utimensat: %s", name);
371
372     /*
373      * Changing the ownership probably won't succeed, unless we're root
374      * or POSIX_CHOWN_RESTRICTED is not set. Set uid/gid before setting
375      * the mode; current BSD behavior is to remove all setuid bits on
376      * chown. If chown fails, lose setuid/setgid bits.
377      */
378     if (chown(name, fs->st_uid, fs->st_gid)) {
379         if (errno != EPERM)
380             cwarn("chown: %s", name);
381         fs->st_mode &= ~(S_ISUID|S_ISGID);

```



 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=11079&url=2366923104057427566612107051&id=21071661714141p76372F2Fwww.directrelief.org%2Femergency%2Fukra

```

386     if (chflags(name, fs->st_flags) && errno != EOPNOTSUPP)
387         cwarn("chflags: %s", name);
388 }

```

## EXAMPLE #2

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xfbbff>)

Show file

**File:** cmds-receive.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FKurc>

**Project:** KuronekoDungeon/android\_external\_btrfsprogs (<https://cpp.hotexamples.com/site/redirect?url>



 Direct Relief

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://www.directrelief.org/emergency/fundraising>

```

965 static int process_utimes(const char *path, struct timespec *at,
966                          struct timespec *mt, struct timespec *ct,
967                          void *user)
968 {
969     int ret = 0;
970     struct btrfs_receive *r = user;
971     char full_path[PATH_MAX];
972     struct timespec tv[2];
973
974     ret = path_cat_out(full_path, r->full_subvol_path, path);
975     if (ret < 0) {
976         error("utimes: path invalid: %s", path);
977         goto out;
978     }
979
980     if (g_verbose >= 2)
981         fprintf(stderr, "utimes %s\n", path);
982
983     tv[0] = *at;
984     tv[1] = *mt;
985     ret = utimensat(AT_FDCWD, full_path, tv, AT_SYMLINK_NOFOLLOW);
986     if (ret < 0) {

```



 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=11079&utm\_source=social&utm\_medium=twitter&utm\_campaign=20210616f7-4141ps76372F2Fwww.directrelief.org%2Femergency%2Fukra

```

991 }
992
993 out:
994     return ret;
995 }

```

## EXAMPLE #3

0

<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x154dc>[Show file](#)**File:** serv.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fjonasj76%2Ff>**Project:** jonasj76/finit (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fj>

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x154dc>

```

178 int serv_touch(char *arg)
179 {
180     char corr[40];
181
182     if (!arg || !arg[0])
183         return serv_list(NULL);
184
185     if (!strstr(arg, ".conf")) {
186         snprintf(corr, sizeof(corr), "%s.conf", arg);
187         arg = corr;
188     }
189
190     pushd(FINIT_RCSD);
191     if (!fexist(arg)) {
192         popd();
193         if (!strstr(arg, "finit.conf"))
194             errx(1, "Service %s is not enabled", arg);
195         arg = FINIT_CONF;
196     }
197
198     /* libite:touch() follows symlinks */
199     if (utimensat(AT_FDCWD, arg, NULL, AT_SYMLINK_NOFOLLOW))

```



Direct Relief

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee6d\_id=11079gdnqzqdn-zcm65a3710f05147256656f2107080aBub2L00U1G54f7L-41t4t5p73d3-2F2Fwww.directrelief.org%2Femergency%2Fukra



## EXAMPLE #4

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x8f82t>)

Show file

**File:** sys\_stat.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F007gzs%2Fandroid-platform-ndk>)

**Project:** 007gzs/android-platform-ndk (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F007gzs%2Fandroid-platform-ndk>)



**Direct Relief**

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://www.directrelief.org/emergency/fundraising>

```

122 void sys_stat_check_functions()
123 {
124     (void)chmod((const char *)1234, (mode_t)0);
125     (void)fchmod(0, (mode_t)0);
126     #if HAVE_XXXAT
127     (void)fchmodat(0, (const char *)1234, (mode_t)0, 0);
128     #endif
129     (void)fstat(0, (struct stat *)1234);
130     #if HAVE_XXXAT
131     (void)fstatat(0, (const char *)1234, (struct stat *)1234, 0);
132     #endif
133     (void)futimens(0, (const struct timespec *)1234);
134     (void)lstat((const char *)1234, (struct stat *)1234);
135     (void)mkdir((const char *)1234, (mode_t)0);
136     #if HAVE_XXXAT
137     (void)mkdirat(0, (const char *)1234, (mode_t)0);
138     #endif
139     (void)mknod((const char *)1234, (mode_t)0);
140     (void)mknodat(0, (const char *)1234, (mode_t)0, (dev_t)0);
141     (void)mknod((const char *)1234, (mode_t)0, (dev_t)0);
142     (void)mknodat(0, (const char *)1234, (mode_t)0, (dev_t)0);
143     (void)stat((const char *)1234, (struct stat *)1234);

```



**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&utm\_source=237659231040574275666f21070051-BUS-210716&utm\_medium=210716&utm\_campaign=210716&utm\_content=210716&utm\_term=210716&utm\_referrer=https://www.directrelief.org/2Emergency/2Fukra

## EXAMPLE #5

0



(https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xbc329

Show file

**File:** common.c (https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FKnightKu%2F

**Project:** KnightKu/fileutils (https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com



WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&img\_id=23669231040574275666f210780518f52107166f714141ps/303/2F%2Fwww.directrelief.org%2Femergency%2Fukra



```

299 void DCOPY_copy_timestamps(
300     bayer_flist flist,
301     uint64_t idx,
302     const char* dest_path)
303 {
304     /* get atime seconds and nsecs */
305     uint64_t atime      = bayer_flist_file_get_atime(flist, idx);
306     uint64_t atime_nsec = bayer_flist_file_get_atime_nsec(flist, idx);
307
308     /* get mtime seconds and nsecs */
309     uint64_t mtime      = bayer_flist_file_get_mtime(flist, idx);
310     uint64_t mtime_nsec = bayer_flist_file_get_mtime_nsec(flist, idx);
311
312     /* fill in time structures */
313     struct timespec times[2];
314     times[0].tv_sec  = (time_t) atime;
315     times[0].tv_nsec = (long)  atime_nsec;
316     times[1].tv_sec  = (time_t) mtime;
317     times[1].tv_nsec = (long)  mtime_nsec;
318
319     /* set times with nanosecond precision using utimensat,
320      * assume path is relative to current working directory.

```



WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&url=https://www.directrelief.org/2Emergency%2Fukra

```

325     dest_path, errno, strerror(errno)
326 );
327 }
328
329 #if 0
330     /* TODO: see stat-time.h and get_stat_atime/mtime/ctime to read sub-second times,
331      * and use utimensat to set sub-second times */

```



```

332     /* as last step, change timestamps */
333     if(! S_ISLNK(statbuf->st_mode)) {
334         struct utimbuf times;
335         times.actime = statbuf->st_atime;
336         times.modtime = statbuf->st_mtime;
337         if(utime(dest_path, &times) != 0) {
338             BAYER_LOG(BAYER_LOG_ERR, "Failed to change timestamps on %s utime() errno=%d %s",
339                     dest_path, errno, strerror(errno)
340                     );
341         }
342     }
343     else {
344         struct timeval tv[2];
345         tv[0].tv_sec = statbuf->st_atime;
346         tv[0].tv_usec = 0;
347         tv[1].tv_sec = statbuf->st_mtime;
348         tv[1].tv_usec = 0;
349         if(lutimes(dest_path, tv) != 0) {
350             BAYER_LOG(BAYER_LOG_ERR, "Failed to change timestamps on %s utime() errno=%d %s",
351                     dest_path, errno, strerror(errno)
352                     );
353         }

```



 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&utm\_source=social&utm\_medium=social&utm\_campaign=20210516&utm\_term=110798&utm\_content=110798&utm\_referrer=https://www.directrelief.org/2Emergency/2Fukra

## EXAMPLE #6

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xd80a3>)

Show file

**File:** update-done.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FAlexE>

**Project:** AlexBaranosky/systemd (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FAlexBaranosky%2Fsystemd>)





```

30 static int apply_timestamp(const char *path, struct timespec *ts) {
31     struct timespec twice[2] = {
32         *ts,
33         *ts
34     };
35     struct stat st;
36
37     assert(path);
38     assert(ts);
39
40     if (stat(path, &st) >= 0) {
41         /* Is the timestamp file already newer than the OS? If
42          * so, there's nothing to do. We ignore the nanosecond
43          * component of the timestamp, since some file systems
44          * do not support any better accuracy than 1s and we
45          * have no way to identify the accuracy
46          * available. Most notably ext4 on small disks (where
47          * 128 byte inodes are used) does not support better
48          * accuracy than 1s. */
49         if (st.st_mtim.tv_sec > ts->tv_sec)
50             return 0;
51

```



**Direct Relief**

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&utm\_source=23662231460574275666f2107051&utm\_medium=210716&utm\_campaign=4141ps76372Fwww.directrelief.org%2Femergency%2Fukra

```

56     return log_debug("Can't update timestamp file %s, file system i
57
58     return log_error_errno(errno, "Failed to update timestamp on %s: %m", p
59 }
60
61 } else if (errno == ENOENT) {
62     _cleanup_close_ int fd = -1;

```





```

63         int r;
64
65         /* The timestamp file doesn't exist yet? Then let's create it. */
66
67         r = mac_selinux_create_file_prepare(path, S_IFREG);
68         if (r < 0)
69             return log_error_errno(r, "Failed to set SELinux context for %s: %m", p
70
71         fd = open(path, O_CREAT|O_EXCL|O_WRONLY|O_TRUNC|O_CLOEXEC|O_NOCTTY|O_NOFOLLOW,
72         mac_selinux_create_file_clear());
73
74         if (fd < 0) {
75             if (errno == EROFS)
76                 return log_debug("Can't create timestamp file %s, file system i
77
78             return log_error_errno(errno, "Failed to create timestamp file %s: %m",
79         }
80
81         (void) loop_write(fd, MESSAGE, strlen(MESSAGE), false);
82
83         if (futimens(fd, twice) < 0)
84             return log_error_errno(errno, "Failed to update timestamp on %s: %m", p
85

```



**Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&utm\_source=230602231040574275666f2107051&utm\_medium=210716&utm\_campaign=4141p76372F%2Fwww.directrelief.org%2Femergency%2Fukra



## EXAMPLE #7

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xd5067>)

Show file

**File:** utils.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fyaccz%2Futi>)

**Project:** yaccz/util-linux (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2>)



WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xd5067>

```

157  /*
158   * Check if @path is on read-only filesystem independently on file permissions.
159   */
160  int mnt_is_readonly(const char *path)
161  {
162      if (access(path, W_OK) == 0)
163          return 0;
164      if (errno == EROFS)
165          return 1;
166      if (errno != EACCES)
167          return 0;
168
169  #ifdef HAVE_FUTIMENS
170      /*
171       * access(2) returns EACCES on read-only FS:
172       *
173       * - for set-uid application if one component of the path is not
174       *   accessible for the current rUID. (Note that euidaccess(2) does not
175       *   check for EROFS at all).
176       *
177       * - for read-write filesystem with read-only VFS node (aka -o remount,ro,bind)
178       */

```



 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=11079&utm\_source=23662231040574275666f210785181521071664f7-4141p-76372F%2Fwww.directrelief.org%2Femergency%2Fukra

```

183      times[1].tv_nsec = UTIME_OMIT; /* mtime */
184
185      if (utimensat(AT_FDCWD, path, times, 0) == -1)
186          return errno == EROFS;
187  }
188  #endif
189

```

```
190 |     return 0;  
    | }
```

(X)



 **Direct Relief**

**WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.**

ebfd9c1af4ee&id=110798&img\_id=235692371040574275666f21078051R1521071664f714141ps7C3Z2F2Fwww.directrelief.org%2Femergency%2Fukra

(X)

## EXAMPLE #8

0

<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xe9cee>[Show file](#)<https://cpp.hotexamples.com/site/redirect>

X

 Direct ReliefWITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.<https://www.directrelief.org/emergency/fundraising>

X



```

114 void
115 psync_utimes(const char *fn, const struct pfl_timespec *pts, int flags)
116 {
117     #ifdef HAVE_FUTIMENS
118         struct timespec ts[2];
119
120         ts[0].tv_sec = pts[0].tv_sec;
121         ts[0].tv_nsec = pts[0].tv_nsec;
122
123         ts[1].tv_sec = pts[1].tv_sec;
124         ts[1].tv_nsec = pts[1].tv_nsec;
125
126         if (utimensat(AT_FDCWD, fn, ts, flags) == -1)
127             psynclog_warn("utimes %s", fn);
128     #else
129         struct timeval tv[2];
130
131         (void)flags;
132
133         tv[0].tv_sec = pts[0].tv_sec;
134         tv[0].tv_usec = pts[0].tv_nsec / 1000;
135

```



WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&url=https://www.directrelief.org/2Emergency%2Fukra

```

140         psynclog_warn("utimes %s", fn);
141     #endif
142 }

```



## EXAMPLE #9

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xe8a0c>)

Show file

**File:** rpmostree-postprocess.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fmijos%2Frpm-ostree>)

**Project:** mijos/rpm-ostree (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fmijos%2Frpm-ostree>)



 **Direct Relief**

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xe8a0c>



```

643  /* SELinux uses PCRE pre-compiled regexps for binary caches, which can
644   * fail if the version of PCRE on the host differs from the version
645   * which generated the cache (in the target root).
646   *
647   * Note also this function is probably already broken in Fedora
648   * 23+ from https://bugzilla.redhat.com/show_bug.cgi?id=1265406 (https://bugzilla.redhat.com/sh
649   */
650  static gboolean
651  workaround_selinux_cross_labeling_recurse (int          dfd,
652      const char    *path,
653      GCancellable  *cancellable,
654      GError        **error)
655  {
656      gboolean ret = FALSE;
657      g_auto(GLnxDirFdIterator) dfd_iter = { 0, };
658
659      if (!glx_dirfd_iterator_init_at (dfd, path, TRUE, &dfd_iter, error))
660          goto out;
661
662      while (TRUE)
663      {
664          struct dirent *dent = NULL;

```



WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110778&img\_id=23662231040574275666f210780518152107166f714141ps/2F?Fwww.directrelief.org%2Femergency%2Fukra

```

669
670      if (!dent)
671          break;
672
673      name = dent->d_name;
674
675      if (dent->d_type == DT_DIR)

```



```

676     {
677         if (!workaround_selinux_cross_labeling_recurse (dfd_iter.fd, name, cancellable, err
678             goto out;
679     }
680     else if (g_str_has_suffix (name, ".bin"))
681     {
682         struct stat stbuf;
683         const char *lastdot;
684         gs_free char *nonbin_name = NULL;
685
686         if (TEMP_FAILURE_RETRY (fstatat (dfd_iter.fd, name, &stbuf, AT_SYMLINK_NOFOLLOW)) !
687         {
688             glx_set_error_from_errno (error);
689             goto out;
690         }
691
692         lastdot = strrchr (name, '.');
693         g_assert (lastdot);
694
695         nonbin_name = g_strdup (name, lastdot - name);
696
697         g_print ("Setting mtime of '%s' to newer than '%s'\n", nonbin_name, name);
698         if (TEMP_FAILURE_RETRY (utimensat (dfd_iter.fd, nonbin_name, NULL, 0)) != 0)
699         {
700             glx_set_error_from_errno (error);
701             goto out;
702         }
703     }
704 }
705
706 ret = TRUE;
707 out:
708 return ret;
709 }

```



 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110778&utm\_source=23669231040574275666f21078051RUS210716&utm\_medium=210716&utm\_campaign=210716&utm\_content=210716&utm\_term=210716&utm\_referrer=https://www.directrelief.org%2Femergency%2Fukra



## EXAMPLE #10

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x53804>)

Show file

**File:** touch.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Frovaughn%2F>

**Project:** rovaughn/distro (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F>



 Direct Relief

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x53804> <https://www.directrelief.org/emergency/fundraising>



```
19 static void
20 touch(const char *file)
21 {
22     int fd;
23     struct stat st;
24     int r;
25
26     if ((r = stat(file, &st)) < 0) {
27         if (errno != ENOENT)
28             eprintf("stat %s:", file);
29         if (cflag)
30             return;
31     } else if (!r) {
32         if (!aflag)
33             times[0] = st.st_atim;
34         if (!mflag)
35             times[1] = st.st_mtim;
36         if (utimensat(AT_FDCWD, file, times, 0) < 0)
37             eprintf("utimensat %s:", file);
38         return;
39     }
40 }
```



WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4eed\_id=1107780mq\_id=23669231040574275666f21078051\_RU521071661f7-414hp-76372F%2Fwww.directrelief.org%2Femergency%2Fukra

```
45 touch(file);
46 }
```





EXAMPLE #11

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xead0a>

[Show file](#)

X

**File:** cmds-receive.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fchrestomanci%2Fbtrfs-progs>)**Project:** chrestomanci/btrfs-progs (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fchrestomanci%2Fbtrfs-progs>)

```

764 static int process_utimes(const char *path, struct timespec *at,
765                          struct timespec *mt, struct timespec *ct,
766                          void *user)
767 {
768     int ret = 0;
769     struct btrfs_receive *r = user;
770     char *full_path = path_cat(r->full_subvol_path, path);
771     struct timespec tv[2];
772
773     if (g_verbose >= 2)
774         fprintf(stderr, "utimes %s\n", path);
775
776     tv[0] = *at;
777     tv[1] = *mt;
778     ret = utimensat(AT_FDCWD, full_path, tv, AT_SYMLINK_NOFOLLOW);
779     if (ret < 0) {

```


 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://www.directrelief.org/emergency/fundraising>

```

785
786 out:
787     free(full_path);
788     return ret;
789 }

```

X



ebfd9c1af4ee&id=110798&img\_id=23569231040574275666f21078051B1521071664f7-4141ps7C3Z2F2Fwww.directrelief.org%2Femergency%2Fukra

EXAMPLE #12

0

(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x969ff>)

Show file

**File:** file\_subs.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fmarctmi>)

**Project:** marctmiller/bitrig (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com>)



 Direct Relief

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://www.directrelief.org/emergency/fundraising>

```

652 void
653 set_ftime(char *fnm, time_t mtime, time_t atime, int frc)
654 {
655     struct timespec tv[2];
656
657     tv[0].tv_sec = atime;
658     tv[0].tv_nsec = 0L;
659     tv[1].tv_sec = mtime;
660     tv[1].tv_nsec = 0L;
661     if (!frc) {
662         /*
663          * if we are not forcing, only set those times the user wants
664          * set.
665          */
666         if (!patime)
667             tv[0].tv_nsec = UTIME_OMIT;
668         if (!pmtime)
669             tv[1].tv_nsec = UTIME_OMIT;
670     }
671
672     /*
673     * set the times

```



 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&img\_src=235652231040574275666f21078051R1521071661f7-4141ps76372F2Fwww.directrelief.org%2Femergency%2Fukra

EXAMPLE #13

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x54c2e>

[Show file](#)

X

**File:** mbox.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fkdave%2Fneon>)

**Project:** kdave/neomutt (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fkdave%2Fneomutt>)



X



```

1595  /**
1596   * mbox_path_probe - Is this an mbox mailbox? - Implements MxOps::path_probe()
1597   */
1598   enum MailboxType mbox_path_probe(const char *path, const struct stat *st)
1599   {
1600       if (!path || !st)
1601           return MUTT_UNKNOWN;
1602
1603       if (S_ISDIR(st->st_mode))
1604           return MUTT_UNKNOWN;
1605
1606       if (st->st_size == 0)
1607           return MUTT_MBOX;
1608
1609       FILE *fp = fopen(path, "r");
1610       if (!fp)
1611           return MUTT_UNKNOWN;
1612
1613       int ch;
1614       while ((ch = fgetc(fp)) != EOF)
1615       {
1616           /* Some mailbox creation tools erroneously append a blank line to

```



 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110779&utm\_source=social&utm\_medium=social&utm\_campaign=2021-07-16&utm\_term=directrelief.org%2Femergency%2Fukra

```

1621         ungetc(ch, fp);
1622         break;
1623     }
1624 }
1625
1626 enum MailboxType magic = MUTT_UNKNOWN;
1627 char tmp[256];

```

```

1628     if (fgets(tmp, sizeof(tmp), fp))
1629     {
1630         if (mutt_str_startswith(tmp, "From ", CASE_MATCH))
1631             magic = MUTT_MBOX;
1632         else if (mutt_str_strcmp(tmp, MMDF_SEP) == 0)
1633             magic = MUTT_MMDF;
1634     }
1635     mutt_file_fclose(&fp);
1636
1637     if (!C_CheckMboxSize)
1638     {
1639         /* need to restore the times here, the file was not really accessed,
1640          * only the type was accessed. This is important, because detection
1641          * of "new mail" depends on those times set correctly. */
1642         #ifdef HAVE_UTIMENSAT
1643             struct timespec ts[2];
1644             mutt_file_get_stat_timespec(&ts[0], &st, MUTT_STAT_ETIME);
1645             mutt_file_get_stat_timespec(&ts[1], &st, MUTT_STAT_MTIME);
1646             utimensat(0, path, ts, 0);
1647         #else
1648             struct utimbuf times;
1649             times.actime = st->st_atime;
1650             times.modtime = st->st_mtime;

```



 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=11079&utm\_source=23669231040574275666f2107851RUS210716&utm\_medium=210716&utm\_campaign=210716&utm\_term=210716&utm\_content=210716&utm\_referrer=https://www.directrelief.org%2Femergency%2Fukra

```

1655     return magic;
1656 }

```

## EXAMPLE #14

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x2351c>)

Show file

**File:** streamfs.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FGunioRobot>)

**Project:** GunioRobot/skywriting (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FGunioRobot/skywriting>)



**Direct Relief**

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x2351c>

```

283     drop_privilege();
284     rc = (utimensat(-1, fixpath(path), tv, 0)) ? -errno : 0;
285     regain_privilege();
286     return rc;
287 }
```

## EXAMPLE #15

0



X

(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x9589f>)

Show file

(<https://cpp.hotexamples.com/site/redirect>)



 Direct Relief

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://www.directrelief.org/emergency/fundraising>

X

```

50  int
51  main(void)
52  {
53      static const char fname[] = "utimensat\\filename";
54
55      assert(utimensat(AT_FDCWD, fname, NULL, 0) == -1);
56      if (ENOENT != errno)
57          error_msg_and_skip("utimensat");
58
59      #define PREFIX "utimensat(AT_FDCWD, \"utimensat\\filename\", \"
60      printf(PREFIX "NULL, 0) = -1 ENOENT (%m)\\n");
61
62      struct timeval tv;
63      struct timespec ts[2];
64
65      if (gettimeofday(&tv, NULL))
66          perror_msg_and_skip("gettimeofday");
67
68      ts[0].tv_sec = tv.tv_sec;
69      ts[0].tv_nsec = tv.tv_usec;
70      ts[1].tv_sec = tv.tv_sec - 1;
71      ts[1].tv_nsec = tv.tv_usec + 1;

```



 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110778&img\_src=23662231040574275666f2107805aRb521071666f714141ps76332F2Fwww.directrelief.org%2Femergency%2Fukra

```

76      print_ts(&ts[1]);
77      printf("], AT_SYMLINK_NOFOLLOW) = -1 ENOENT ");
78
79      assert(utimensat(AT_FDCWD, fname, ts, AT_SYMLINK_NOFOLLOW) == -1);
80      if (ENOENT != errno)
81          error_msg_and_skip("utimensat");
82      printf("(%m)\\n");

```



```

83
84     ts[0].tv_nsec = UTIME_NOW;
85     ts[1].tv_nsec = UTIME_OMIT;
86     assert(utimensat(AT_FDCWD, fname, ts, AT_SYMLINK_NOFOLLOW) == -1);
87     if (ENOENT != errno)
88         error_msg_and_skip("utimensat");
89     printf(PREFIX "[UTIME_NOW, UTIME_OMIT], AT_SYMLINK_NOFOLLOW)"
90           " = -1 ENOENT (%m)\n");
91
92     puts("+++ exited with 0 +++");
93     return 0;
94 }

```



 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&utm\_source=23669231040574275666f21078051R1521071661f7-4141ps76372F%2Fwww.directrelief.org%2Femergency%2Fukra



#### EXAMPLE #16

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xecdbc>

Show file



**File:** fusexmp.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fstarius%2Fmylinuxprog>)

**Project:** starius/mylinuxprog (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fstarius%2Fmylinuxprog>)

```

164 static int xmp_utimens(const char *path, const struct timespec ts[2])
165 {
166     int res;
167     /* don't use utime/utimes since they follow symlinks */
168     res = utimensat(0, path, ts, AT_SYMLINK_NOFOLLOW);
169     if (res == -1)
170         return -errno;
171     return 0;
172 }
```

#### EXAMPLE #17

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x2c021>)

Show file

**File:** lutimes.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F4ian%2Fen>)



<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x2c021>



```

6  int lutimes(const char *filename, const struct timeval tv[2])
7  {
8      struct timespec times[2];
9      times[0].tv_sec = tv[0].tv_sec;
10     times[0].tv_nsec = tv[0].tv_usec * 1000;
11     times[1].tv_sec = tv[1].tv_sec;
12     times[1].tv_nsec = tv[1].tv_usec * 1000;
13     return utimensat(AT_FDCWD, filename, times, AT_SYMLINK_NOFOLLOW);
14 }

```



 Direct Relief

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4eef914c794640574256661a07051b15210016846741416376132F2Fwww.directrelief.org%2Femergency%2Fukra

Show file

**File:** t\_utimensat.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fasof>

**Project:** 2asoft/freebsd (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F>



```

128 ATF_TC_BODY(utimensat_fderr1, tc)
129 {
130     int dfd;
131
132     ATF_REQUIRE(mkdir(DIR, 0755) == 0);
133     ATF_REQUIRE((dfd = open(DIR, O_RDONLY, 0)) != -1);
134     ATF_REQUIRE(utimensat(dfd, FILEERR, tptr, 0) == -1);
135     ATF_REQUIRE(close(dfd) == 0);
136 }

```

## EXAMPLE #19

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x362dc>)

Show file

**File:** update-done.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F275288698%2Fsystemd-ubuntu-with-dbus>)

**Project:** 275288698/systemd-ubuntu-with-dbus (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F275288698%2Fsystemd-ubuntu-with-dbus>)



 Direct Relief

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x362dc> <https://www.directrelief.org/emergency/fundraising>



```

31 static int apply_timestamp(const char *path, struct timespec *ts) {
32     struct timespec twice[2];
33     struct stat st;
34
35     assert(path);
36     assert(ts);
37
38     if (stat(path, &st) >= 0) {
39         /* Is the timestamp file already newer than the OS? If so, there's nothing to d
40         if (st.st_mtim.tv_sec > ts->tv_sec ||
41             (st.st_mtim.tv_sec == ts->tv_sec && st.st_mtim.tv_nsec >= ts->tv_nsec))
42             return 0;
43
44         /* It is older? Then let's update it */
45         twice[0] = *ts;
46         twice[1] = *ts;
47
48         if (utimensat(AT_FDCWD, path, twice, AT_SYMLINK_NOFOLLOW) < 0) {
49
50             if (errno == EROFS)
51                 return log_debug("Can't update timestamp file %s, file system i
52

```



**Direct Relief**

**WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.**

ebfd9c1af4ee&id=11079&utm\_source=23669231040574275666f2107851&utm\_medium=2107166167-4141ps76372F2Fwww.directrelief.org%2Femergency%2Fukra

```

57     _cleanup_close_ int fd = -1;
58     int r;
59
60     /* The timestamp file doesn't exist yet? Then let's create it. */
61
62     r = mac_selinux_create_file_prepare(path, S_IFREG);
63     if (r < 0)

```

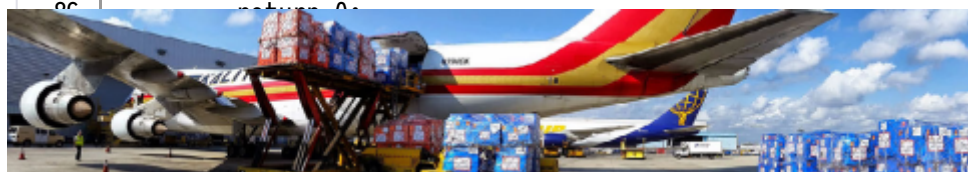


```

64         return log_error_errno(r, "Failed to set SELinux context for %s: %m", p
65
66         fd = open(path, O_CREAT|O_EXCL|O_WRONLY|O_TRUNC|O_CLOEXEC|O_NOCTTY|O_NOFOLLOW,
67         mac_selinux_create_file_clear());
68
69         if (fd < 0) {
70             if (errno == EROFS)
71                 return log_debug("Can't create timestamp file %s, file system i
72
73             return log_error_errno(errno, "Failed to create timestamp file %s: %m",
74         }
75
76         (void) loop_write(fd, MESSAGE, strlen(MESSAGE), false);
77
78         twice[0] = *ts;
79         twice[1] = *ts;
80
81         if (futimens(fd, twice) < 0)
82             return log_error_errno(errno, "Failed to update timestamp on %s: %m", p
83     } else
84         log_error_errno(errno, "Failed to stat() timestamp file %s: %m", path);
85
86     return 0;

```

X



**Direct Relief**

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

EXAMPLE #20

X

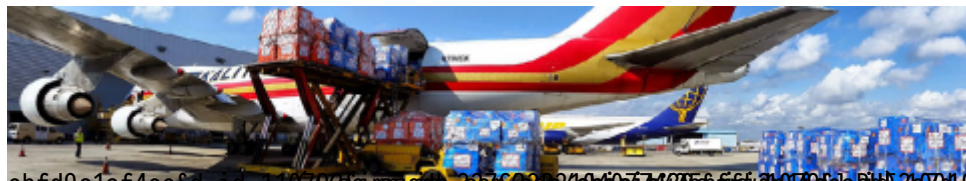
(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x9083e>

Show file

**File:** sdcard.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F1Anastaska>

**Project:** 1Anastaska/platform\_system\_core ([https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FAnastaska%2Fplatform\\_system\\_core](https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FAnastaska%2Fplatform_system_core))

X



 **Direct Relief**

**WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.**

ebfd9c1af4ee&id=110793&img\_id=23569231040574275666f2107805a8b521071664f714141ps%3A%2F%2Fwww.directrelief.org%2Femergency%2Fukra

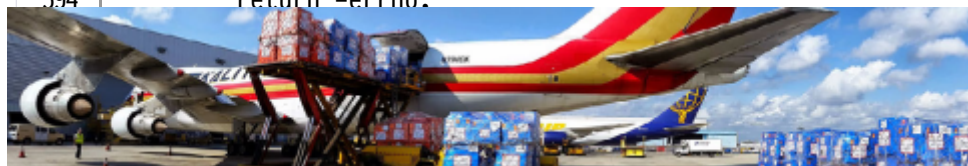
X



```

573 static int handle_setattr(struct fuse* fuse, struct fuse_handler* handler,
574     const struct fuse_in_header *hdr, const struct fuse_setattr_in *req)
575 {
576     struct node* node;
577     char path[PATH_MAX];
578     struct timespec times[2];
579
580     pthread_mutex_lock(&fuse->lock);
581     node = lookup_node_and_path_by_id_locked(fuse, hdr->nodeid, path, sizeof(path));
582     TRACE("[%d] SETATTR fh=%llx valid=%x @ %llx (%s)\n", handler->token,
583         req->fh, req->valid, hdr->nodeid, node ? node->name : "?");
584     pthread_mutex_unlock(&fuse->lock);
585
586     if (!node) {
587         return -ENOENT;
588     }
589
590     /* XXX: incomplete implementation on purpose.
591      * chmod/chown should NEVER be implemented.*/
592
593     if ((req->valid & FATTR_SIZE) && truncate(path, req->size) < 0) {
594         return -errno;

```



WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&utm\_source=23669231040574275666f2107851&utm\_medium=210716&utm\_campaign=76372F2Fwww.directrelief.org%2Femergency%2Fukra

```

599     * time specified in the request. Same goes for mtime. Use utimensat(2)
600     * as it allows ATIME and MTIME to be changed independently, and has
601     * nanosecond resolution which fuse also has.
602     */
603     if (req->valid & (FATTR_ATIME | FATTR_MTIME)) {
604         times[0].tv_nsec = UTIME_OMIT;
605         times[1].tv_nsec = UTIME_OMIT;

```



```

606     if (req->valid & FATTR_ETIME) {
607         if (req->valid & FATTR_ETIME_NOW) {
608             times[0].tv_nsec = UTIME_NOW;
609         } else {
610             times[0].tv_sec = req->etime;
611             times[0].tv_nsec = req->etimensec;
612         }
613     }
614     if (req->valid & FATTR_MTIME) {
615         if (req->valid & FATTR_MTIME_NOW) {
616             times[1].tv_nsec = UTIME_NOW;
617         } else {
618             times[1].tv_sec = req->mtime;
619             times[1].tv_nsec = req->mtimensec;
620         }
621     }
622     TRACE("[%d] Calling utimensat on %s with atime %ld, mtime=%ld\n",
623           handler->token, path, times[0].tv_sec, times[1].tv_sec);
624     if (utimensat(-1, path, times, 0) < 0) {
625         return -errno;
626     }
627 }

```



 Direct Relief

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebf9c1af4ee&id=110798&url=https%3A%2F%2Fgithub.com%2F2asoft%2Ffreebsd&url=https%3A%2F%2Fwww.directrelief.org%2Femergency%2Fukra

(https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x8bb33

Show file

**File:** t\_utimensat.c (https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F2asof

**Project:** 2asoft/freebsd (https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F2

```

163 ATF_TC_BODY(utimensat_fderr3, tc)
164 {
165     int fd;
166
167     ATF_REQUIRE(mkdir(DIR, 0755) == 0);
168     ATF_REQUIRE((fd = open(FILE, O_CREAT|O_RDWR, 0644)) != -1);
169     ATF_REQUIRE(close(fd) == 0);
170
171     ATF_REQUIRE(utimensat(-1, FILE, tptr, 0) == -1);
172 }

```

## EXAMPLE #22

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xcd9ac>)

Show file



WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://www.directrelief.org/emergency/fundraising>





```

343 static int pifs_utimens(const char *path, const struct timespec times[2])
344 {
345     DIR *dir = opendir(options.mdd);
346     if (!dir) {
347         return -errno;
348     }
349     int ret = utimensat(dirfd(dir), basename((char *) path), times, 0);
350     closedir(dir);
351     return ret == -1 ? -errno : ret;
352 }

```

## EXAMPLE #23

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x90aa3>)

Show file

**File:** bru.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FHardolaf%2Fbedrocklinux-userland>)

**Project:** Hardolaf/bedrocklinux-userland (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FHardolaf%2Fbedrocklinux-userland>)



 Direct Relief

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x90aa3>





```

814  /*
815   * TODO: To simplify things, we're mandating absolute paths. We should
816   * probably properly handle relative paths for this later and remove this
817   * restriction. Given this, the first argument to utimensat() is ignored.
818   */
819  static int bru_utimens(const char *path, const struct timespec *times)
820  {
821      SET_CALLER_UID();
822      REDIR_PATH(path, new_path);
823
824      int ret = utimensat(0, new_path, times, AT_SYMLINK_NOFOLLOW);
825
826      SET_RET_ERRNO();
827      return ret;
828  }

```

## EXAMPLE #24

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xaf105>)

Show file



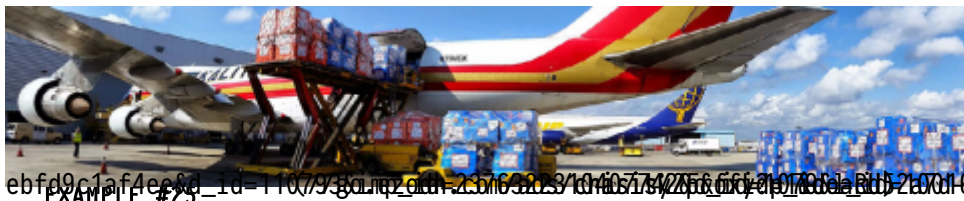
WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://www.directrelief.org/emergency/fundraising>

```

248 static int xmp_utimens(const char *path, const struct timespec ts[2])
249 {
250     fprintf(f, "utimes path: %s\n", path);
251     int res = utimensat(0, path, ts, AT_SYMLINK_NOFOLLOW);
252     if (res == -1){
253         fprintf(f, "operation failed\n");
254         return -errno;
255     }
256     fprintf(f, "operation succeeded\n");
257     return 0;
258 }

```



 Direct Relief

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

EXAMPLE #23 - [https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x8cbd5ebfd9c1af4e5d\\_id=110779&url=https%3A%2F%2Fgithub.com%2FSkyLandTW%2Ffiles2.c](https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x8cbd5ebfd9c1af4e5d_id=110779&url=https%3A%2F%2Fgithub.com%2FSkyLandTW%2Ffiles2.c)

([https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x8cbd5ebfd9c1af4e5d\\_id=110779&url=https%3A%2F%2Fgithub.com%2FSkyLandTW%2Ffiles2.c](https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x8cbd5ebfd9c1af4e5d_id=110779&url=https%3A%2F%2Fgithub.com%2FSkyLandTW%2Ffiles2.c))

Show file

**File:** files2.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FSkyLandTW%2Ffiles2.c>)

**Project:** SkyLandTW/JXTN (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F>

X



 **Direct Relief**

**WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.**

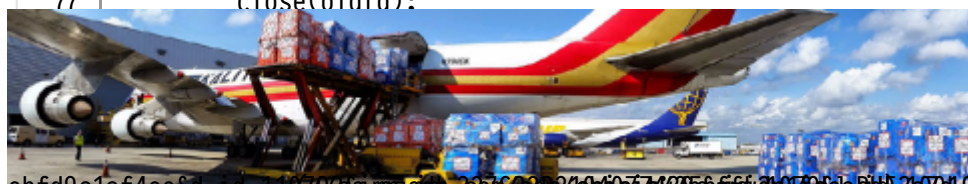
ebfd9c1af4ee&id=110798&img\_id=23569231040574275666f2107805aRt521071664f7-4141ps76372F2Fwww.directrelief.org%2Femergency%2Fukra

X

```

56 static int copyat(int olddirfd, const char* oldpath, int newdirfd, const char* newpath) {
57     int err;
58     int oldfd = openat(olddirfd, oldpath, O_RDONLY | O_CLOEXEC | O_NOFOLLOW | O_NOATIME);
59     if (oldfd == -1 && errno == EPERM) {
60         oldfd = openat(olddirfd, oldpath, O_RDONLY | O_CLOEXEC | O_NOFOLLOW);
61     }
62     if (oldfd == -1 && errno == ELOOP) {
63         char oldtarget[PATH_MAX];
64         ssize_t oldlen = readlinkat(olddirfd, oldpath, oldtarget, sizeof(oldtarget));
65         if (oldlen == -1) {
66             return -1;
67         }
68         oldtarget[oldlen] = '\0';
69         return symlinkat(oldtarget, newdirfd, newpath);
70     }
71     if (oldfd == -1) {
72         return -1;
73     }
74     struct stat oldstat;
75     if (fstat(oldfd, &oldstat) == -1) {
76         err = errno;
77         close(oldfd);

```



 **Direct Relief**

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110778&utm\_source=25662231040574275666f2107805aRt5210716&utm\_medium=26016&utm\_campaign=4141ps76372F%2Fwww.directrelief.org%2Femergency%2Fukra

```

82     if (newfd == -1 && errno == EPERM) {
83         newfd = openat(newdirfd, newpath, O_WRONLY | O_CLOEXEC | O_CREAT | O_TRUNC, oldstat.st_
84     }
85     if (newfd == -1) {
86         err = errno;
87         close(oldfd);
88         errno = err;

```

```

89         return -1;
90     }
91     if (fchown(newfd, oldstat.st_uid, oldstat.st_gid) == -1) {
92         // ignore error
93     }
94     if (copyfile_sparse(oldfd, newfd) == -1) {
95         err = errno;
96         close(newfd);
97         close(oldfd);
98         errno = err;
99         return -1;
100    }
101    close(newfd);
102    close(oldfd);
103    struct timespec times[2];
104    times[0] = oldstat.st_atim;
105    times[1] = oldstat.st_mtim;
106    utimensat(newdirfd, newpath, times, 0); // ignore error
107    return 0;
108 }

```



Direct Relief

**WITHOUT REGARD TO**  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd991af4ee6a1d-11079-b01420a1-25b6-20231010171422566f1d10703a1b1b2100164ur7l-1n4t4ps73a0-2F2Fwww.directrelief.org%2Femergency%2Fukra

**File:** utimes.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FVargMon%2F>

**Project:** VargMon/bionic (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2FVargMon%2Fbionic>)



```

35 int utimes(const char* path, const struct timeval tv[2]) {
36     struct timespec ts[2];
37     struct timespec* ts_ptr = NULL;
38     if (tv != NULL) {
39         if (!timespec_from_timeval(&ts[0], &tv[0]) || !timespec_from_timeval(&ts[1], &tv[1])) {
40             errno = EINVAL;
41             return -1;
42         }
43         ts_ptr = ts;
44     }
45     return utimensat(AT_FDCWD, path, ts_ptr, 0);
46 }

```

**EXAMPLE #27**

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xb586c>)

Show file

**File:** utils.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fa565109863%2Fsrc>)

**Project:** a565109863/src (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fa565109863%2Fsrc>)



**Direct Relief**

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0xb586c>





```

252 int
253 setfile(struct stat *fs, int fd)
254 {
255     struct timespec ts[2];
256     int rval;
257
258     rval = 0;
259     fs->st_mode &= S_ISTXT | S_ISUID | S_ISGID | S_IRWXU | S_IRWXG | S_IRWXO;
260
261     ts[0] = fs->st_atim;
262     ts[1] = fs->st_mtim;
263     if (fd >= 0 ? futimens(fd, ts) :
264         utimensat(AT_FDCWD, to.p_path, ts, AT_SYMLINK_NOFOLLOW)) {
265         warn("update times: %s", to.p_path);
266         rval = 1;
267     }
268     /*
269     * Changing the ownership probably won't succeed, unless we're root
270     * or POSIX_CHOWN_RESTRICTED is not set. Set uid/gid before setting
271     * the mode; current BSD behavior is to remove all setuid bits on
272     * chown. If chown fails, lose setuid/setgid bits.
273     */

```



WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&url=https%3A%2F%2Fwww.directrelief.org%2Femergency%2Fukra



```

278         rval = 1;
279     }
280     fs->st_mode &= ~(S_ISTXT | S_ISUID | S_ISGID);
281 }
282 if (fd >= 0 ? fchmod(fd, fs->st_mode) :
283     fchmodat(AT_FDCWD, to.p_path, fs->st_mode, AT_SYMLINK_NOFOLLOW)) {
284     warn("chmod: %s", to.p_path);

```



```
285         rval = 1;
286     }
287
288     /*
289     * XXX
290     * NFS doesn't support chflags; ignore errors unless there's reason
291     * to believe we're losing bits. (Note, this still won't be right
292     * if the server supports flags and we were trying to *remove* flags
293     * on a file that we copied, i.e., that we didn't create.)
294     */
295     errno = 0;
296     if (fd >= 0 ? fchflags(fd, fs->st_flags) :
297         chflagsat(AT_FDCWD, to.p_path, fs->st_flags, AT_SYMLINK_NOFOLLOW))
298         if (errno != EOPNOTSUPP || fs->st_flags != 0) {
299             warn("chflags: %s", to.p_path);
300             rval = 1;
301         }
302     return (rval);
303 }
```

X

 Direct ReliefWITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee&id=110798&img\_src=23669231040574275666f21078051aR1521071661f714141p76332F2Fwww.directrelief.org%2Femergency%2Fukra

X



## EXAMPLE #28

0

(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x3435c>)

[Show file](#)

**File:** syscalls-2007.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F520SRig%2Fvalgrind>)

**Project:** 520SRig/valgrind (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F520SRig%2Fvalgrind>)



 Direct Relief

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

<https://www.directrelief.org/emergency/fundraising>



```

32 int main (void)
33 {
34     #if defined(HAVE_SIGNALFD) && defined(HAVE_EVENTFD) \
35         && defined(HAVE_EVENTFD_READ) && defined(HAVE_PPOLL)
36     {
37         sigset_t mask;
38         int fd, fd2;
39         eventfd_t ev;
40         struct timespec ts = { .tv_sec = 1, .tv_nsec = 0 };
41         struct pollfd pfd[2];
42
43         sigemptyset (&mask);
44         sigaddset (&mask, SIGUSR1);
45         fd = signalfd (-1, &mask, 0);
46         sigaddset (&mask, SIGUSR2);
47         fd = signalfd (fd, &mask, 0);
48         fd2 = eventfd (5, 0);
49         eventfd_read (fd2, &ev);
50         pfd[0].fd = fd;
51         pfd[0].events = POLLIN|POLLOUT;
52         pfd[1].fd = fd2;
53         pfd[1].events = POLLIN|POLLOUT;

```



**Direct Relief**

WITHOUT REGARD TO  
POLITICS, RELIGION, OR ABILITY TO PAY.

ebfd9c1af4ee8d-11079801231040574275666f2107051B1521071661f7-4141ps7C3Z2F2Fwww.directrelief.org%2Femergency%2Fukra



```

58     #if defined(HAVE_UTIMENSAT)
59         unlink("/tmp/valgrind-utimensat-test");
60         close (creat ("/tmp/valgrind-utimensat-test", S_IRUSR | S_IWUSR));
61         {
62             struct timespec ts2[2] = { [0].tv_sec = 10000000, [1].tv_sec = 20000000 };
63             utimensat (AT_FDCWD, "/tmp/valgrind-utimensat-test", ts2, 0);
64         }

```



```
65     unlink("/tmp/valgrind-utimensat-test");
66 #endif
67
68 #if defined(HAVE_EPOLL_CREATE) && defined(HAVE_EPOLL_PWAIT)
69 {
70     int fd3;
71     struct epoll_event evs[10];
72     sigset_t mask;
73
74     sigemptyset (&mask);
75     sigaddset (&mask, SIGUSR1);
76     sigaddset (&mask, SIGUSR2);
77     fd3 = epoll_create (10);
78     epoll_pwait (fd3, evs, 10, 0, &mask);
79 }
80 #endif
81
82     return 0;
83 }
```

**EXAMPLE #29**

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x8bb3c>)

Show file

**File:** t\_utimensat.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F2asof>

**Project:** 2asoft/freebsd (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2F>

```
179 ATF_TC_BODY(utimensat_fdlink, tc)
180 {
181     int dfd;
182     struct stat st;
183
184     ATF_REQUIRE(mkdir(DIR, 0755) == 0);
185     ATF_REQUIRE(symlink(FILE, LINK) == 0); /* NB: FILE does not exists */
186
187     ATF_REQUIRE((dfd = open(DIR, O_RDONLY, 0)) != -1);
188
189     ATF_REQUIRE(utimensat(dfd, BASELINK, tptr, 0) == -1);
190     ATF_REQUIRE(errno == ENOENT);
191
192     ATF_REQUIRE(utimensat(dfd, BASELINK, tptr, AT_SYMLINK_NOFOLLOW) == 0);
193
194     ATF_REQUIRE(close(dfd) == 0);
195
196     ATF_REQUIRE(lstat(LINK, &st) == 0);
197     ATF_REQUIRE(st.st_atimespec.tv_sec == tptr[0].tv_sec);
198     ATF_REQUIRE(st.st_atimespec.tv_nsec == tptr[0].tv_nsec);
199     ATF_REQUIRE(st.st_mtimespec.tv_sec == tptr[1].tv_sec);
200     ATF_REQUIRE(st.st_mtimespec.tv_nsec == tptr[1].tv_nsec);
201 }
```

## EXAMPLE #30

0



(<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html#0x9d978>)

Show file

**File:** utime.c (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.com%2Fapprisi%2Fi>)

**Project:** apprisi/illumos-gate (<https://cpp.hotexamples.com/site/redirect?url=https%3A%2F%2Fgithub.c>)

```
198 int
199 utimesys(int code,
200          uintptr_t arg1, uintptr_t arg2, uintptr_t arg3, uintptr_t arg4)
201 {
202     switch (code) {
203     case 0:
204         return (futimens((int)arg1, (timespec_t *)arg2));
205     case 1:
206         return (utimensat((int)arg1, (char *)arg2,
207                          (timespec_t *)arg3, (int)arg4));
208     default:
209         return (set_errno(EINVAL));
210     }
211 }
```

Top Functions (<https://cpp.hotexamples.com/site/trends?type=cpp%7Cf>) | Top Classes  
(<https://cpp.hotexamples.com/site/trends?type=cpp%7Cc>) | Privacy Policy (</site/privacy>)  
PHP (<https://hotexamples.com/>) | C# (CSharp) (<https://csharp.hotexamples.com/>) | Java  
(<https://java.hotexamples.com/>) | GoLang (<https://golang.hotexamples.com/>) | C++ (Cpp)  
(<https://cpp.hotexamples.com/>) | Python (<https://python.hotexamples.com/>) | JavaScript  
(<https://javascript.hotexamples.com/>) | TypeScript (<https://typescript.hotexamples.com/>)  
EN (<https://cpp.hotexamples.com/examples/-/-/utimensat/cpp-utimensat-function-examples.html>) | RU  
(<https://cpp.hotexamples.com/ru/examples/-/-/utimensat/cpp-utimensat-function-examples.html>) | DE  
(<https://cpp.hotexamples.com/de/examples/-/-/utimensat/cpp-utimensat-function-examples.html>) | FR  
(<https://cpp.hotexamples.com/fr/examples/-/-/utimensat/cpp-utimensat-function-examples.html>) | ES  
(<https://cpp.hotexamples.com/es/examples/-/-/utimensat/cpp-utimensat-function-examples.html>) | PT  
(<https://cpp.hotexamples.com/pt/examples/-/-/utimensat/cpp-utimensat-function-examples.html>) | IT  
(<https://cpp.hotexamples.com/it/examples/-/-/utimensat/cpp-utimensat-function-examples.html>) | JP  
(<https://cpp.hotexamples.com/jp/examples/-/-/utimensat/cpp-utimensat-function-examples.html>) | ZH  
(<https://cpp.hotexamples.com/zh/examples/-/-/utimensat/cpp-utimensat-function-examples.html>)



([www.liveinternet.ru/click](http://www.liveinternet.ru/click))

