

NAME

utmp, wtmp — utmp and wtmp entry formats

DESCRIPTION

These files, which hold user and accounting information for such commands as *who(1)*, *wall(1)*, *write(1)*, *getty(1M)*, and *login(1)*, have the following structure as defined by `<utmp.h>`:

```

/*      @(#)utmp.h      3.2      */
/*      <sys/types.h> must be included.      */

#define      UTMP_FILE      "/etc/utmp"
#define      WTMP_FILE      "/etc/wtmp"

struct utmp
{
    char ut_user[8];          /* User login name */
    char ut_id[2];          /* /etc/lines id(usually line #) */
    char ut_line[12];       /* device name (console, lnxx) */
    short ut_pid;           /* process id */
    struct exit_status
    {
        char e_termination; /* Process termination status */
        char e_exit;        /* Process exit status */
    }
    ut_exit;                /* The exit status of a process
    * marked as DEAD_PROCESS.
    */
    short ut_type;          /* type of entry */
    time_t ut_time;        /* time entry was made */
};

/*      Definitions for ut_type      */

#define      EMPTY          0
#define      RUN_LVL        1
#define      BOOT_TIME      2
#define      OLD_TIME       3
#define      NEW_TIME       4
#define      INIT_PROCESS   5          /* Process spawned by "init" */
#define      LOGIN_PROCESS  6          /* A "getty" process waiting for login */
#define      USER_PROCESS   7          /* A user process */
#define      DEAD_PROCESS   8

#define      UTMAXTYPE      DEAD_PROCESS /* Largest legal value of ut_type */

/*      Special strings or formats used in the "ut_line" field when
/*      accounting for something other than a process.
/*      ** Note ** each message is such that it takes exactly 11
/*      spaces + a null, so that it fills the "ut_line" array.

#define      RUNLVL_MSG      "run_level_%c"
#define      BOOT_MSG        "system_boot"
#define      OTIME_MSG       "old_time "
#define      NTIME_MSG       "new_time "

```

FILES

```

/usr/include/utmp.h
/etc/utmp
/etc/wtmp

```

SEE ALSO

login(1), who(1), write(1), getut(3C)

