

**NAME**

**archive\_entry\_atime**, **archive\_entry\_atime\_nsec**, **archive\_entry\_atime\_is\_set**, **archive\_entry\_set\_atime**, **archive\_entry\_unset\_atime**, **archive\_entry\_birthtime**, **archive\_entry\_birthtime\_nsec**, **archive\_entry\_birthtime\_is\_set**, **archive\_entry\_set\_birthtime**, **archive\_entry\_unset\_birthtime**, **archive\_entry\_ctime**, **archive\_entry\_ctime\_nsec**, **archive\_entry\_ctime\_is\_set**, **archive\_entry\_set\_ctime**, **archive\_entry\_unset\_ctime**, **archive\_entry\_mtime**, **archive\_entry\_mtime\_nsec**, **archive\_entry\_mtime\_is\_set**, **archive\_entry\_set\_mtime**, **archive\_entry\_unset\_mtime** - functions for manipulating times in archive entry descriptions

**LIBRARY**

Streaming Archive Library (libarchive, -larchive)

**SYNOPSIS**

```
#include <archive_entry.h>
```

*time\_t*

```
archive_entry_atime(struct archive_entry *a);
```

*long*

```
archive_entry_atime_nsec(struct archive_entry *a);
```

*int*

```
archive_entry_atime_is_set(struct archive_entry *a);
```

*void*

```
archive_entry_set_atime(struct archive_entry *a, time_t sec, long nanosec);
```

*void*

```
archive_entry_unset_atime(struct archive_entry *a);
```

*time\_t*

```
archive_entry_birthtime(struct archive_entry *a);
```

*long*

```
archive_entry_birthtime_nsec(struct archive_entry *a);
```

*int*

```
archive_entry_birthtime_is_set(struct archive_entry *a);
```

*void*

**archive\_entry\_set\_birthtime**(*struct archive\_entry \*a, time\_t sec, long nanosec*);

*void*

**archive\_entry\_unset\_birthtime**(*struct archive\_entry \*a*);

*time\_t*

**archive\_entry\_ctime**(*struct archive\_entry \*a*);

*long*

**archive\_entry\_ctime\_nsec**(*struct archive\_entry \*a*);

*int*

**archive\_entry\_ctime\_is\_set**(*struct archive\_entry \*a*);

*void*

**archive\_entry\_set\_ctime**(*struct archive\_entry \*a, time\_t sec, long nanosec*);

*void*

**archive\_entry\_unset\_ctime**(*struct archive\_entry \*a*);

*time\_t*

**archive\_entry\_mtime**(*struct archive\_entry \*a*);

*long*

**archive\_entry\_mtime\_nsec**(*struct archive\_entry \*a*);

*int*

**archive\_entry\_mtime\_is\_set**(*struct archive\_entry \*a*);

*void*

**archive\_entry\_set\_mtime**(*struct archive\_entry \*a, time\_t sec, long nanosec*);

*void*

**archive\_entry\_unset\_mtime**(*struct archive\_entry \*a*);

## DESCRIPTION

These functions create and manipulate the time fields in an *archive\_entry*. Supported time fields are atime (access time), birthtime (creation time), ctime (last time an inode property was changed) and mtime (modification time).

libarchive(3) provides a high-resolution interface. The timestamps are truncated automatically depending on the archive format (for archiving) or the filesystem capabilities (for restoring).

All timestamp fields are optional. The **XXX\_unset()** functions can be used to mark the corresponding field as missing. The current state can be queried using **XXX\_is\_set()**. Unset time fields have a second and nanosecond field of 0.

### SEE ALSO

archive\_entry(3), libarchive(3)

### HISTORY

The **libarchive** library first appeared in FreeBSD 5.3.

### AUTHORS

The **libarchive** library was written by Tim Kientzle <kientzle@acm.org>.