

Re: Request for Comments: libarchive, bsdtar

Source: <http://unix.derkeiler.com/Mailing-Lists/FreeBSD/arch/2004-01/0058.html>

From: Tim Kientzle (kientzle_at_acm.org)

Date: 01/14/04

Date: Wed, 14 Jan 2004 12:42:17 -0800

To: "M. Warner Losh" <imp@bsdimp.com>

M. Warner Losh wrote:

> *In message: <Pine.NEB.3.96L.1040114142135.49872F-100000@fledge.watson.org>*

> *Robert Watson <rwatson@freebsd.org> writes:*

> *: I know NetBSD has a neat tool to create file systems from userspace*

> *: without privilege, but my understanding is that it has to pull in a lot of*

> *: code from the kernel in fairly messy ways. Since tar files are a well*

> *: supported portable format... :-)*

>

> *The problem then reduces to needing to be able to create the tar file*

> *with ownership/permissions different than is in the unpriv'd build*

> *tree. Generation of this list, as well as getting tar/whatever to*

> *honor it are interesting problems.*

Not all that interesting, really. :-)

For example, the following simple C program uses libarchive to read an archive from stdin, and write a new ustar/gzip archive to stdout with all entries owned by 'root'. You just create the archive as usual and then feed it through this filter to create a new archive with the proper ownership/permissions. The same technique can be used add entries to an archive, remove entries from an archive, rename archive entries, convert archive formats, etc.

```
struct archive *in;
struct archive *out;
struct archive_entry *entry;

/* Read an archive from stdin: format/compression
   will be auto-detected. */
in = archive_read_new();
archive_read_support_format_all(in);
archive_read_support_compression_all(in);
archive_read_open_file(in, NULL); /* NULL == stdin */

/* Write a ustar/gzip archive to stdout. */
out = archive_write_new();
```

freebsd-arch: Re: Request for Comments: libarchive, bsdtar

```
archive_write_set_compression_gzip(out);
archive_write_set_format_ustar(out);
archive_write_open_file(out, NULL); /* NULL == stdout */

/* Copy entries over, doing any necessary editing */
while (archive_read_next_header(in, &entry) == ARCHIVE_READ_OK) {
    /* Edit the entry as appropriate */
    archive_entry_set_uid(entry, 0);
    archive_entry_set_uname(entry, "root");
    /* Write a header to the output archive */
    archive_write_header(out, entry);
    /* TODO: XXX Copy data to output archive XXX */
}
/* All done */
archive_write_finish(out);
archive_read_finish(in);
```

freebsd-arch@freebsd.org mailing list

<http://lists.freebsd.org/mailman/listinfo/freebsd-arch>

To unsubscribe, send any mail to "freebsd-arch-unsubscribe@freebsd.org"