

## ファイルの圧縮と転送について

外戸口, 武

<https://doi.org/10.15017/6770294>

---

出版情報：情報処理教育広報. 18 (2), pp.2-10, 1995-12. 九州大学情報処理教育センター  
バージョン：  
権利関係：



# ファイルの圧縮と転送について

外戸口 武

E-mail:kedo@kyudai.kugi.kyoto-u.ac.jp

## 1 はじめに

みなさんの中に、「手持ちのファイルが大き過ぎてハードディスクの容量が足りない。」とか、「重要なファイルだから、バックアップをとっておきたい。」などと考えたことがある方がいらっしゃると思います。

ハードディスクの空き容量を増やすには、ファイルを圧縮（もちろん、不要なファイルは削除するべきでしょうが）し、バックアップをとるには、フロッピーディスクなどにコピーをとることになるでしょう。もし、みなさんの使われているコンピューターでフロッピーディスク等が使用できない場合、例えば情報処理教育センターのワークステーションでは、講義室のパソコンにファイルを転送してフロッピーディスクに書き込むことになります。

今回はファイルの圧縮、転送について、簡単に説明したいと思います。具体的には、UNIX 上でファイルを圧縮、解凍するためのツールとして、gzip, gunzip, compress, uncompress を紹介します。また、パソコン上でファイルを圧縮、解凍するためのツールとして lha を紹介します。ファイルの転送に関しては ftp 等を使用してファイルを UNIX からパソコンに転送するという手段が一般的かとは思いますが、uuencode, uudecode, shar も紹介します。また、ファイルの圧縮とは直接に関係があるわけではないですが、ツリー構造を持ったディレクトリーを一つのファイルにまとめるための tar, gtar を紹介します。

今回みなさんに紹介する、これらのツールはたいへん多機能なものです。しかし、今回はその機能の一部、基本的な使い方を紹介させていただきたいと思います。

## 2 ファイルの圧縮、解凍について (UNIX で)

この節では UNIX でファイルの圧縮、解凍をおこなうためのツール、gzip, gunzip, compress, uncompress について、その基本的な使い方を解説します。

### 2.1 コマンドの表記について

当たり前と思われる方もいらっしゃると思いますが、以下の文章では、例をあげるなら、

```
gzip file
```

というように、ツールの使い方を説明します。これはコマンドラインで `gzip file` とタイプして、最後にリターンキーを打つ、という意味です。以下の文章中では、「最後にリターンキーを打ってください。」というような説明はしませんが、コマンドを打ち込んだら最後にリターンキーを打ってください。

### 2.2 gzip

gzip はファイルの圧縮、解凍をするためのツールです。もし、“file” というファイルを圧縮するには、

```
gzip file
```

と、実行してください。すると、“file” は圧縮されて、“file.gz” という名前のファイルが新しくできます。ただし、“file” は消えてしまいます。gzip を使ってファイルを圧縮すると“gz” という拡張子がファイルの末尾についたファイルができます。

### 2.3 gunzip,(gzip -d)

gzip により、圧縮されたファイルを解凍する（圧縮されたファイルを元にもどすとき）には、

```
gunzip file.gz または gunzip file
```

または、

```
gzip -d file.gz または gzip -d file
```

と、実行してください。

ここで一つ注意しなければならないことがあります。もし、圧縮の結果、新しくできたファイルの名前が気に入らなくて、ファイル名を変えたとしましょう。例えば、“file.gz”を“file.compressed”に変えたとします。そして gunzip（または gzip -d）でファイルを解凍しようとする、

```
gunzip: file.compressed: unknown suffix -- ignored
```

と、画面に表示されてしまい、“file.compressed”を解凍することができません。なぜなら gzip で圧縮されたファイルを解凍するときには、gunzip を使うわけですが、（もちろん gzip -d を使うこともできます）この場合ファイルの最後は“.gz”で終わっていなければならないからです。（実は後で紹介する、compress で圧縮した、最後が“.Z”で終わっているファイルも解凍できます。）ファイルの最後が“.gz”で終わっていれば gzip で圧縮したファイルだと一目でわかりますから、拡張子は変えないほうがいいでしょう。

## 2.4 compress,uncompress

次は同じくファイルの圧縮、解凍のためのツールである、compress , uncompress についてです。compress , uncompress は、gzip , gunzip と同じようにつかえて、圧縮なら

```
compress file
```

とタイプすると gzip のときと同様に“file.Z”というファイルができます。これを解凍するには、

```
uncompress file.Z または uncompress file
```

としてください。ここでも拡張子を変えるのはトラブルの元なので、なるべく変えない方がよいと思います。また、先ほど紹介した gunzip , gzip -d でも解凍することができます。しかし、gzip を使って圧縮したファイルを uncompress を使って解凍することはできませんから注意してください。

## 3 ファイルの転送に関するもの（UNIX で）

ファイル転送には ftp を使うのがポピュラーだと思いますが、この小節では、電子メールを使ってファイルを特にバイナリーファイルを送るのに便利なコマンドと

して uuencode、uudecode を紹介します。

### 3.1 uuencode

電子メールではバイナリーファイルを送ることができません。そこで、電子メールを用いてバイナリーファイルを送るには、一度 ASCII ファイルに変換してから送ることになります。

では、具体的にどうすればよいのでしょうか。先ほど gzip を用いて圧縮した “file.gz” を電子メールで送るとします。バイナリーファイルである “file.gz” を ASCII ファイルに変換するには、

```
uuencode file.gz file.gz
```

と、実行してください。uuencode の後に “file.gz” が二つ並んでいますが、左の方はバイナリーを ASCII に変換して送りたいファイルです。右の方は、後で説明する、uudecode を使って ASCII からバイナリーに戻した時に作られるファイルの名前です。実際にやってみると、暗号のようなものが画面に出力されます。これを電子メールにして送り、受け取った側は今度はバイナリーファイルに戻せばいいのです。電子メールにして “itsuki” に送るには、

```
uuencode file.gz file.gz > mail.txt
```

としてから mail itsuki< mail.txt

としても構いませんが、

```
uuencode file.gz file.gz | mail itsuki
```

と、いうようにすれば少し手間がはぶけます。uuencode を使ってバイナリーファイルを ASCII ファイルに変換すると、ファイルのサイズが 30 ～ 40 % 増えてしまいます。サイズの大きなバイナリーファイルを uuencode を使って送る時は気をつけてください。

### 3.2 uudecode

次に電子メールを受け取った側が元のバイナリーファイルに戻す方法を解説します。メールを受け取って、その内容を “itsuki.txt” というファイルに書き込んだとします。

```
uudecode ituki.txt
```

と、命令してください。すると、“file.gz” というファイルが作られます。これは先ほど、送り手が指定したファイル名です。

## 4 複数のファイルを一つにまとめる (UNIX で)

この節では、複数のファイルを一つにまとめるための `tar` , `gtar` を紹介します。これは、複数のファイルやディレクトリー（特にツリー構造を持ったもの）のバックアップをとろうとするときなどに便利なツールです。

例えば、情報処理教育センターで小講義室の UNIX を使っているとします。ファイルのバックアップをするためには、大講義室やオープン端末室のパソコンにファイルを転送し、フロッピーディスクに書き込むことになります。もし、ファイルのサイズが大きいのと思えば、先ほど紹介した `gzip` , `compress` を使って圧縮してから転送すればいいことです。

しかし、バックアップをとっておきたいファイルが複数ある場合があります。ただ単に複数のファイルを転送するだけなら、`ftp` で例えば、

```
mget *.txt
```

と、すれば簡単です。でも、ツリー構造をもったディレクトリー全てをバックアップとりたい場合は厄介なことになります。先ずフロッピーディスクに同じツリー構造のディレクトリーを作らなければなりません。さらにファイルをそのディレクトリーごとに転送しなければなりません。ファイルを圧縮するのも結構な手間がかかることでしょう。

### 4.1 tar

`tar` を使えば、ツリー構造を持ったディレクトリーごと一つのファイルにまとめることができます。例えば、“`kedoguchi`” というディレクトリーの中にあるファイルやディレクトリー、更にその下のファイルやディレクトリーを一つにまとめたいとしましょう。

```
tar cvf kedoguchi.tar kedoguchi
```

と、命令してみてください。“`cvf`” のところは “`-cvf`” でも構いません。すると、

```
tar cvf kedoguchi.tar kedoguchi
a kedoguchi/data/950101.dat 2 blocks
a kedoguchi/data/950101.dat 2 blocks
```

```
a kedoguchi/tex/report.tex 4 blocks
```

```
a kedoguchi/tex/report.dvi 4 blocks
```

```
...
```

```
..
```

```
.
```

と、画面に出力されます。そして、“kedoguchi.tar”というファイルができています。ただし、gzip や compress を使ったときとは違って“kedoguchi”というディレクトリーは消えません。もし、画面に出力されるメッセージが必要のない場合は、

```
tar cf kedoguchi.tar kedoguchi
```

と、してください。“v”をつければメッセージが画面にでできます。tar を使って複数のファイルやディレクトリーを一つのファイルにまとめた、そのファイルのことを普通“tarfile”といいます。ちなみに“c”は新しい tarfile を作る、“f”はその次に書かれたものを tarfile の名前とする。という意味です。また、

```
tar cvf file.tar file1 file2
```

のように、tarfile になるものは一つのディレクトリーでなく、複数のファイルであっても構いません。また、ファイルとディレクトリーを同時に指定することもできますし、複数のディレクトリーを同時に指定することもできます。このようにして作ったファイルを ftp 等を使って転送してください。

さて、先ほど作った“kedoguchi.tar”から元のツリー構造を持ったディレクトリーを再び作るにはどうしたらいいのでしょうか。今度は先ほどの“cvf”のかわりに“xvf”とします。“x”は“tarfile”からファイルやディレクトリーを取り出すという意味です。ですから、

```
tar xvf kedoguchi.tar
```

とすればいいわけです。

余談ですが、“f”の次に書かれるファイル名を“-”としたときは標準出力への書き込みか標準入力からの読み込みのどちらか適当な操作をしてくれます。例えば、“kedoguchi.tar”が“kedoguchi.tar.gz”のように圧縮されていたとします。ファイルを解凍して画面に表示（標準出力）させるための gzcac を使って、

```
gzcat kedoguchi.tar.gz | tar xvf -
```

と、すれば一気に終わらせることができます。gzcat は圧縮されたファイルにも使える cat だと思えばいいと思います。

ここで一つ注意しなければならないことがあります。

```
tar cvf kedoguchi.tar kedoguchi
```

の “kedoguchi” のかわりにパス名をつけて例えば、“/usr/home/kedoguchi” のように絶対パス名で指定したとします。こうしてできた tarfile を他のコンピュータに転送して、そこでファイルを取り出そうとしたとします。すると “/usr/home” に “kedoguchi” という（この場合は）ディレクトリーをつくろうとします。書き込みの許可がなければディレクトリーを作れませんし、もし作れても、予期しない別のところにディレクトリーができてしまいます。ですから絶対パス名は使わずに相対パス名で指定したほうがよいでしょう。

#### 4.1.1 gtar

先ほど gzcat , tar を使って、圧縮された tarfile から “kedoguchi” というディレクトリーを作りました。gtar は tar をもっと高機能にしたもので、tarfile を作ると同時に圧縮も行なうことができます。使い方は tar とほとんど同じで、

```
gtar cvzf kedoguchi.tar.gz kedoguchi
```

と、試しに命令してみてください。tar と違うのは “cvf” が “cvzf” に（もちろん “tar” が “gtar” に変わっています。）変わっているところです。この “z” はできた tarfile を圧縮して（gzip をつかっています。）から指定したファイルに書き込むという意味です。ですから、“cvzf” のつぎに書かれるファイルの名前は、tarfile であること、圧縮されていることが分かるように “kedoguchi.tar.gz” としました。“kedoguchi.tar.gz” からファイルを取り出すのもほとんど tar と同じで、

```
gtar xvzf kedoguchi.tar.gz
```

と、命令してください。“z” を使って作られた tarfile は元に戻すときには、やはり “z” をつけなければなりません。

## 5 複数のファイル、ディレクトリーをまとめてメールで送る

複数のファイルやディレクトリーごとまとめて電子メールで送るときはどうしたらいいのでしょうか。tar で一つのファイルにまとめて、uuencode で送ればいいですが、すこし面倒です。受け取った側もかなり面倒です。

この小節では、ファイルを電子メールで送るとき、特にディレクトリーごとまとめて送ったり、複数のファイルを送るのに便利な shar や unshar を紹介します。

### 5.1 shar

先ほど tar の説明のところで使った “kedoguchi” というディレクトリーを今度は電子メールで “itsuki” に送ってみましょう。この場合も tar の時と同様ですが “kedoguchi” というディレクトリーを絶対パス名で指定するのはやめたほうがいいでしょう。さて、具体的な shar の使い方ですが、

```
shar kedoguchi | mail itsuki
```

と命令してください。もちろん、uuencode のところで説明したように、一度ファイルに書き込んでから mail コマンドに入力リダイレクションしても構いません。(“shah kedoguchi > mail.txt” としてから mail itsuki < mail.txt” といった具合です。)

これを受け取った “itsuki” 君が “itsuki.txt” というファイルにその内容を書き込んだとしましょう。この “itsuki.txt” を元に戻して、“kedoguchi” というディレクトリーを作るには unshar を使います。使い方は uudecode に似ていて、

```
unshar itsuki.txt
```

と命令するだけです。

## 6 パソコンでファイルを圧縮、解凍

今までは UNIX で使う、ファイルの圧縮や解凍、転送に関するツールを紹介してきましたが、今度はパソコンで使えるツール lha を紹介します。(小講義室の UNIX でも使えます。) この節の標題には「パソコンでファイルを圧縮、解凍」としてありますが、lha は複数のファイルや、ディレクトリーを一つにまとめて、同時に圧縮することができます。

## 6.1 lha

さて、具体的な使い方ですが、“file1”と“file2”を一つのファイルにまとめて圧縮したいとしましょう。そうして新しくできるファイルの名前を“file.lzh”にしましょう。“lzh”というのは、lha で作ったファイルであることが分かりやすいようにつけました。（一般にそうしていると思います。）では、

```
lha c file.lzh file1 file2
```

と、命令してみてください。“file1”、“file2”の変わりにディレクトリーを指定すればその中のファイルが指定した一つのファイルに書き込まれます。もちろんファイルとディレクトリーを同時に指定しても構いません。

次に“file.lzh”から“file1”、“file2”を取り出す方法を説明します。

```
lha x file.lzh
```

と、命令してください。

## 7 最後に

ファイルの圧縮や、転送に関するツールを紹介してきましたが、ここで紹介した使い方は、それぞれの使い方のほんの一部です。もっと詳しい使い方については、オンラインマニュアル等を参考にしてください。今までの話をまとめますと、UNIXでファイルの圧縮を行なう場合には、gzip, gunzip, compress, uncompress 等が使え、ファイルのバックアップに関しては、tar, gtar, lha を使い、一つのファイルにまとめて、転送するのが便利であるといえます。また、電子メールを使って、ファイルを送りたい場合は uuencode, uudecode, shar, unshar を使えばバイナリーファイルも送ることができます。