



## Tar Overview

### Abstract:

Tar files are the preferred method of bundling files, such as html pages, applications programs, or configuration files for clean and simple installation in an IPm controller. This technical note discusses this capability and provides overview information on using and creating tar files.

### Introduction

Tar files are often used to backup and install files because they retain path information and compress the constituent files in much the way a zip file does. In IPm systems, SIXNET recommends the use of Tar files as installation files to cleanly install packages on an IPm for several key reasons:

- Tar files are standard and mature technology accepted by the Linux world
- Public documentation is readily available (see links below)
- Tar files may include scripts giving them sophisticated installation capabilities
- **The I/O Tool Kit software will automatically install Tar files into an IPm**
- During extraction, time is saved and mistakes are reduced because path information is retained

File sets generated from the Tar utility are easily recognizable because they end with the extension: '.tar'. These types of files are commonly known as tar files or "tarballs". When the tarball is extracted, all files will be extracted to their corresponding paths.

Tar, by default, does not use compression. The files contained in the tarball are more or less concatenated. The size of the tarball will be slightly larger than the combined sizes of the files contained within the tarball. A '.tar.gz' or '.tgz' implies compression (via gzip). Although the two extensions are equivalent, '.tar.gz' is considered to be more standard than '.tgz'.

Below is a list of commonly used tar commands and examples on how to use them. Tar has many more options, not discussed in this tech note.

For more examples and elaborate uses of tar made by GNU, consult the web page:

[www.gnu.org/manual/tar/html\\_node/tar\\_toc.html#TOC41](http://www.gnu.org/manual/tar/html_node/tar_toc.html#TOC41)

The GNU web page regularly maintains and holds the manuals to GNU programs, and is the recommended site to visit when more information is needed for tar.

Note that to find more documentation on the 'tar' available in the IPm, consult the BusyBox site. Here is the direct link to the tar documentation available in BusyBox: [www.busybox.net/downloads/BusyBox.html](http://www.busybox.net/downloads/BusyBox.html)

The IPm runs BusyBox because it includes a common set of utilities, including 'tar', at the benefit of having a small footprint. Therefore, when having questions regarding 'tar' in the IPm, the BusyBox site is the recommended site to visit.



## Creating a tarball

Working with an example, we will consider a collection of html pages (that perhaps you wish to install in an IPm controller). These html pages contain path sensitive references in them. In addition, all the html page information is stored in directories labeled: '/directory1', '/directory2', '/directory3'. Here, using tar to backup these directories is beneficial because path information is retained and thus the html pages would work properly when transferred into an IPm. To back up these directories and store it in a tar file named 'web\_set.tar'. The way you type it into the command prompt would be:

```
'tar cvpf web_set.tar /directory1 /directory2 /directory3'
```

Please note that:

- The 'c' option means to create a new archive
- The 'v' option means to show the current command (operation)
- The 'f' option means to store the files in the designated archive
- The 'p' option preserves the permissions of the files/directories

## Listing what is currently in the tar archive

Sometimes it is important to make sure that the tar file to be extracted into the IPm is the correct one. The 't' option is therefore used to list out what are the files and directory paths are currently stored in an archive. The way that the command would be typed would be:

```
'tar tvf web_set.tar'
```

## Extracting from an archive

One of the most common uses of the tar utility as mentioned before is for installation purposes. To extract a tar file once it is transferred, use the 'x' option. A comprehensive extraction involving all the files and directories in the tarball is performed by the following command:

```
'tar xvf web_set.tar'
```

Sometimes only a select amount of directories are to be extracted, to save space or retain existing files and directories. For example, say that only the directories 'directory1' and 'directory3' are to be extracted from the tarball. To do so, the command would be typed as:

```
'tar xvf web_set.tar /directory1 /directory3'
```

## Compressing an archive

To reduce the time transferring a tarball from the toolkit to the IPm, the tarball is usually compressed using gzip. Tarballs that are compressed with gzip end with the extension '.tar.gz'. To compress a tarball simply type:

```
'gzip web_set.tar'
```

The resulting compressed tarball should now be web\_set.tar.gz

Note: The SIXNET I/O Tool Kit software will automatically extract compressed files when tar files are installed as part of an automated project load or when using the "Load Tar File" action button in the File Operation window.



It is also possible to tar and compress a file file with one command:

```
'tar cvpzf web_set.tgz /directory1 /directory2 /directory3'
```

Note:

- The 'z' option compresses the tarball

### **Extracting all the contents of a tarball which is also gzipped**

Instead of using the gzip utility to decompress and then the tar utility to extract the tarball, there is a much faster and easier way. Using the given command below, a gzipped tarball can be decompressed and its contents extracted from the tarball at the same time:

```
'tar zvxf web_set.tar'
```