How to install and use 7zip file archiver on Ubuntu Linux

howtoforge.com/tutorial/how-to-install-and-use-7zip-file-archiver-on-ubuntu-linux

Scan your Web-Server for Malware with ISPProtect now. Get Free Trial.
If you are a Microsoft Windows user, and your work involves sending/receiving large files over a network, there's quite a possibility that you'd have at least heard of (if not used) the 7zip tool. For those who aren't aware, 7zip is an award-winning, open-source file archiver that is claimed to have "a high compression ratio."

Aside from 7z, the tool supports several other formats. For example, you can use it to package and extract the following type of files: 7z, XZ, BZIP2, GZIP, TAR, ZIP, and WIM. However, that's not it, as the 7zip tool can also be used to unpackage several other file types, including AR, ARJ, CAB, CHM, CPIO, CramFS, DMG, EXT, FAT, GPT, HFS, IHEX, ISO, LZH, LZMA, MBR, MSI, NSIS, NTFS, QCOW2, RAR, RPM, SquashFS, UDF, UEFI, VDI, VHD, VMDK, WIM, XAR and Z.

"Compression ratio in the new 7z format is 30-50% better than the ratio of the ZIP format," the tool's official documentation says. Following are some of the other features of 7zip that the official website lists:

- For ZIP and GZIP formats, **7-Zip** provides a compression ratio that is 2-10 % better than the ratio provided by PKZip and WinZip
- Strong AES-256 encryption in 7z and ZIP formats
- Self-extracting capability for 7z format
- Integration with Windows Shell
- Powerful File Manager
- A powerful command line version
- Plugin for FAR Manager
- Localizations for 87 languages

In case you're wondering which awards 7zip has won, from the tool's Wikipedia page: *In 2007, SourceForge.net granted it community choice awards for "Technical Design" and for "Best Project". In 2013, 7-Zip received Tom's Hardware Elite award due to superiority in speed and compression ratio.*

Enough reasons to learn how to use it, right? But, is it available on Linux? Well, yes - there is a port of the command line version to Linux/Unix, and it's dubbed **p7zip**.

In this article, we'll discuss how to install and use p7zip in Linux. Please note that all the instructions and commands mentioned in this tutorial have been tested on **Ubuntu 18.04 LTS**.

Download and Installation of 7zip

You can easily download and install the p7zip tool in Ubuntu using the following commands:
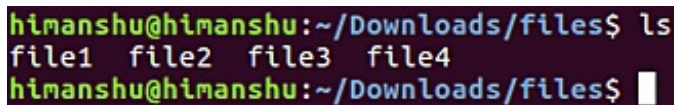
sudo apt-get install p7zip-full p7zip-rar

7zip Usage

The p7zip package installs a command line utility named 7z. Following is its syntax:

7z [adeltux] [-] [SWITCH] <ARCHIVE_NAME> <ARGUMENTS>...

If that seems a bit complex, don't worry, because we'll explain 7z's usage in baby steps.

To begin with, let's understand how we can create a .7z archive using this command. For this, suppose there's a directory named *files* which contains four files:
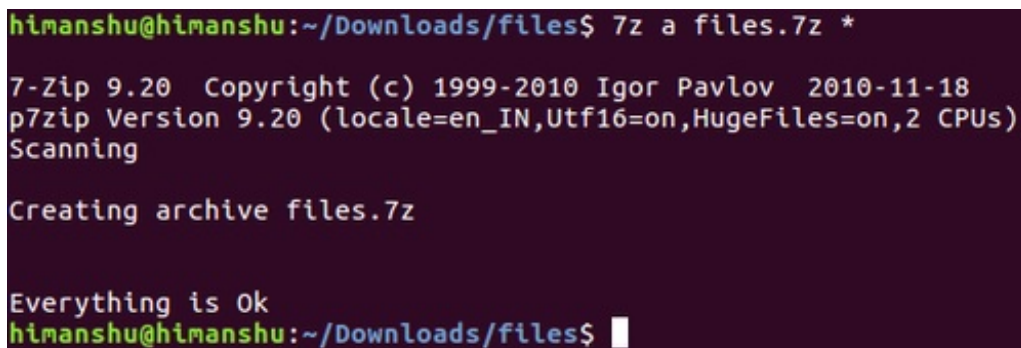

Now, the task is to package all these 4 files in a .7z file, say *files.7z*. So to do this, enter the *files* directory on the command line, and execute the following command:

$ 7z a files.7z *

The following screenshot shows the command in action:

After the above command is run successfully, a file named *files.7z* will be created in your current working directory.

```
himanshu@himanshu:~/Downloads/files$ ls
file1   file2   file3   file4   files.7z
himanshu@himanshu:~/Downloads/files$
```

Now, suppose you accidentally/intentionally delete the four files present in the *files* directory:

```
himanshu@himanshu:~/Downloads/files$ ls
files.7z
himanshu@himanshu:~/Downloads/files$
```

And want to retrieve them by extracting the .7z archive file we just created. This can be done using the 'e' command line option in the following way:

7z e files.7z

```
himanshu@himanshu:~/Downloads/files$ 7z e files.7z

7-Zip 9.20  Copyright (c) 1999-2010 Igor Pavlov  2010-11-18
p7zip Version 9.20 (locale=en_IN,Utf16=on,HugeFiles=on,2 CPUs)

Processing archive: files.7z

Extracting  file1
Extracting  file2
Extracting  file3
Extracting  file4

Everything is Ok

Files: 4
Size:       0
Compressed: 128
himanshu@himanshu:~/Downloads/files$ ls
file1   file2   file3   file4   files.7z
himanshu@himanshu:~/Downloads/files$
```

That was the basics of archiving and extracting using the 7z tool. Now suppose, the requirement is to create a .zip file. This can be easily done using the following command:

7z a [zipfile-name] [files-to-be-zipped]

For example:

```
himanshu@himanshu:~/Downloads/files$ 7z a files.zip *

7-Zip 9.20  Copyright (c) 1999-2010 Igor Pavlov  2010-11-18
p7zip Version 9.20 (locale=en_IN,Utf16=on,HugeFiles=on,2 CPUs)
Scanning

Creating archive files.zip

Compressing  file1
Compressing  file2
Compressing  file3
Compressing  file4
Compressing  files.7z

Everything is Ok
himanshu@himanshu:~/Downloads/files$ ls
file1  file2  file3  file4  files.7z  files.zip
himanshu@himanshu:~/Downloads/files$
```

The 7zip tool also allows you to easily update an archive file after its creation. For example, suppose you've created an archive and then realized that something more needs to be added to it, then instead of deleting the old archive and creating a new one, you can simply update it with the new stuff that you want to add.

For example, suppose the *files* directory now contains a total of 5 text files, and I want to add the *file5* to the files.7z archive as well. Then I can do that using the 'u' option in the 7z command. Here's the command in action:

```
himanshu@himanshu:~/Downloads/files$ 7z u files.7z *

7-Zip 9.20  Copyright (c) 1999-2010 Igor Pavlov  2010-11-18
p7zip Version 9.20 (locale=en_IN,Utf16=on,HugeFiles=on,2 CPUs)

Scanning

Updating archive files.7z

Compressing  files.7z

Everything is Ok
himanshu@himanshu:~/Downloads/files$
```

One way to confirm that the updated archive now contains a total of five files is to manually check the updated archive by double-clicking on it. But if you want to do this through the command line, then you can use the 'l' option provided by the 7z command.

```
himanshu@himanshu:~/Downloads/files$ 7z l files.7z

7-Zip 9.20  Copyright (c) 1999-2010 Igor Pavlov  2010-11-18
p7zip Version 9.20 (locale=en_IN,Utf16=on,HugeFiles=on,2 CPUs)

Listing archive: files.7z

--
Path = files.7z
Type = 7z
Method = LZMA
Solid = -
Blocks = 1
Physical Size = 310
Headers Size = 184

   Date      Time    Attr         Size   Compressed  Name
------------------- ----- ------------ ------------  ------------------------
2016-11-23 12:12:40 ....A          128          126  files.7z
2016-11-23 12:09:23 ....A            0            0  file1
2016-11-23 12:09:23 ....A            0            0  file2
2016-11-23 12:09:23 ....A            0            0  file3
2016-11-23 12:09:23 ....A            0            0  file4
2016-11-23 13:08:34 ....A            0            0  file5
------------------- ----- ------------ ------------  ------------------------
                               128          126  6 files, 0 folders
himanshu@himanshu:~/Downloads/files$
```

So as you can see in the screenshot above, the 7z command confirmed that the *files.7z* archive has been updated with the new text file.

Moving on, should you want to delete something from an archive created by 7z, you can do that as well. In this case, the command line option to be used is 'd'. The overall syntax of the command, however, remains the same.

7z d [archive-name] [file-to-be-deleted]

Keep in mind that you'll have to add a '-r' switch to the above command in case the file in question resides inside a sub-directory that has been archived, although for some reason, the official 7z documentation discourages users from using this switch (I tested it several times, and it works for me as expected).

Following is some important information that the creators of 7z utility want you to keep in mind:

DO NOT USE the 7-zip format for backup purpose on Linux/Unix because :
- 7-zip does not store the owner/group of the file.

On Linux/Unix, in order to backup directories you must use tar :
- to backup a directory : tar cf - directory | 7za a -si directory.tar.7z
- to restore your backup : 7za x -so directory.tar.7z | tar xf -

If you want to send files and directories (not the owner of file) to others Unix/MacOS/Windows users, you can use the 7-zip format.
example : 7za a directory.7z directory

Do not use "-r" because this flag does not do what you think.

Do not use directory/* because of ".*" files (example : "directory/*" does not match "directory/.profile")

For more information on 7z, head to the command's <u>man page</u>.

Conclusion

As you'd have observed over the course of this tutorial, 7zip is a really capable archiving tool. It's incarnation on Linux - p7zip - is not only easy to install but fairly easy to use as well. A lot of documentation is available in the form of a <u>man page</u>, <u>FAQ and support pages on the tool's official website</u>, as well as information on <u>its Sourceforge listing</u>. If you're impressed with 7zip, do give it a try.

About Himanshu Arora

Himanshu Arora has been working on Linux since 2007. He carries professional experience in system level programming, networking protocols, and command line. In addition to HowtoForge, Himanshu's work has also been featured in some of world's other leading publications including Computerworld, IBM DeveloperWorks, and Linux Journal.

Suggested articles

7 Comment(s)

Comments