



CREATING AND EXTRACTING ARCHIVES

A PROJECT REPORT

ON

LINUX ADMINISTRATION LAB

24CAP-6067

Submitted by

VIVEK SINGH (24MCC20083)

in partial fulfillment for the award of the degree of

**MASTER OF COMPUTER APPLICATION
(CLOUD COMPUTING AND DEVOPS)**

Chandigarh University

November 2024



CERTIFICATE

Certified that this project report “**CREATING AND EXTRACTING ARCHIVES**” is the bonafide work of “**VIVEK SINGH**” who carried out the project work under my/our supervision.

SIGNATURE

SIGNATURE

HEAD OF THE DEPARTMENT
UIC

SUPERVISOR
Assistant Professor

Submitted for the project viva voce examination held on

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGMENT

I would like to express my sincere gratitude to everyone who contributed to the successful completion of this project. First and foremost, I would like to thank Mr. Rishabh Tomar Sir, my project mentor, for their invaluable guidance, support, and encouragement throughout the entire process. Their insights and expertise have been instrumental in shaping this project, and I am deeply grateful for their time and effort.

I would also like to thank University Institute of Computing for providing the resources and platform necessary for conducting this project. The tools and facilities made available were crucial in bringing this project to fruition. A special mention to my family and friends for their unwavering support, patience, and encouragement, without which this project would not have been possible.

Lastly, I would like to express my heartfelt thanks to my peers and colleagues for their collaborative spirit and helpful feedback during various stages of the project. Your input helped improve the quality and scope of the work.

Thank you all for your contributions to making this project a success.

Vivek Singh

TABLE OF CONTENTS

TITLE	PAGE NO.
Certificate	2
Acknowledgement	3
Project Description	5
Introduction	6
Overview	7
Implementation	8-10
Results	11
Key Learnings and Observations	12
Application In Real-World	12
Conclusion	13
Plagiarism Report	14

Project Description

This project aims to learn how to create and extract compressed archives using standard tools in Linux such as tar (Tape Archive) and gzip (GNU zip). It is aimed at hands-on experience in creating tarballs that Compressed archive of directories and files as well as decompression and extract these archives to a specific location. to achieve This project focuses on understanding the core concepts of file compression, storage, and extraction. While exploring practical command line operations for efficient file management...

INTRODUCTION

Growing demand for efficient file management This is especially true when dealing with large numbers of files and directories. This makes mastery of data storage and compression an important skill. This is especially relevant in administrative development environments. Managing large data sets requires not only organizations but also organizations. But it also increases the efficiency of storing and transferring yen. These skills are invaluable.

This project focuses on increasing expertise in creating and extracting compressed archives. The tar utility is used to combine multiple files and directories into a single manageable collection, while gzip is used to compress collections. these functions.

This further reduces the size for more efficient storage and transfer. In addition to improvements, it's also easier to transfer compressed files between them. The network improves the overall information sharing process.

The main objectives of the project are:

1. Understand basic syntax and usage of tar and gzip - Develop familiarity with available commands, flags, and options to efficiently manage logs.
2. Practice creating a Tarball archive – to learn how to store directories and files in a single .tar file.
3. Compressing these records using gzip – applies compression to the collection. Reduced size to optimize storage and transfer.
4. Deleting compressed files to different locations – Practice how to decompress and delete files in a specific directory. to ensure recovery

TOOLS AND TECHNOLOGY USED

1. **Operating System:** Linux (any distribution with tar and gzip installed)
2. **Archive Tool:** tar (Tape Archive)
3. **Compression Tool:** gzip (GNU zip)
4. **File System:** Any local or networked file system in Linux
5. **Terminal:** A command-line interface to execute the required commands

OVERVIEW

Tar

tar is a widely used command in Linux for archiving files. It combines multiple files and directories into a single file known as a tarball (with a .tar extension). Tarballs do not provide compression on their own, but they serve as a container for organizing files.

Common tar commands:

1)Creating archive

```
tar -cvf archive_name.tar directory_or_file
```

-c : Create a new archive

-v : Verbose, lists files being added to the archive

-f : Specifies the file name of the archive

2)Extracting archive

```
tar -xvf archive_name.tar
```

-x : Extract the archive

-v : Verbose, lists files being extracted

-f : Specifies the file name of the archive

Gzip

gzip is a compression tool that reduces the size of files, making them easier to store or transfer. It compresses the contents of a file, typically a tarball, reducing its size while preserving the integrity of the data.

Common gzip commands:

1)Compressing an archive:

gzip archive_name.tar

This command compresses the tarball, producing a file with the .tar.gz extension.

2)Decompressing an archive:

gunzip archive_name.tar.gz

This command restores the archive to its original .tar format.

IMPLEMENTATION

The project consists of three main tasks:

- Creating a tarball archive using tar.
- Compressing the archive using gzip.
- Extracting the compressed archive to a different location.

Each task is performed using specific commands in the Linux terminal, and the project follows a sequential approach to ensure the correct steps are followed for effective file management.

1)Creating a Tarball Archive

The first step involves creating a tarball from a directory. The tar command is used to bundle multiple files and directories into a single archive. This is helpful for organizing data into a single file for easier storage or transfer.

Command used:

```
tar -cvf archive_name.tar /path/to/directory
```

Explanation:

-c is for creating the archive.

-v is for verbose output, meaning the process lists the files being added to the archive.

-f specifies the archive name.

Example:

```
tar -cvf archive.tar /home/user_name/Documents/project_files
```

```
[viveksingh@vbox Archives]$ tar -cvf archive.tar /home/viveksingh/Documents
tar: Removing leading '/' from member names
/home/viveksingh/Documents/
/home/viveksingh/Documents/viveksingh.txt
/home/viveksingh/Documents/vivek.txt
/home/viveksingh/Documents/file1.txt
/home/viveksingh/Documents/logfile.txt
/home/viveksingh/Documents/sampleddata.txt
/home/viveksingh/Documents/archive.tar
[viveksingh@vbox Archives]$ ls
archive.tar
```

This command creates a tarball named project_files.tar containing all the files and subdirectories from /home/user/documents/project_files.

2) Compressing the Archive with gzip

Once the tarball is created, the next step is to compress the archive using gzip. Compressing reduces the size of the archive, which is particularly useful when dealing with large files or when transferring files over a network.

Command used:

```
gzip archive_name.tar
```

This command compresses the archive, appending a .gz extension to the file.

Example:

gzip project_files.tar

```
[viveksingh@vbox Archives]$ gzip archive.tar
[viveksingh@vbox Archives]$ ls
archive.tar.gz  home
[viveksingh@vbox Archives]$
[viveksingh@vbox Archives]$
```

After compression, the file project_files.tar becomes project_files.tar.gz. The size of the file is significantly reduced, making it easier to store or transfer.

3) Extracting the Compressed Archive

To restore the files from the compressed archive, we first need to decompress it using the gunzip command, then extract the tarball using tar.

Commands used:

gunzip archive_name.tar.gz

```
[viveksingh@vbox Archives]$ gunzip archive.tar
[viveksingh@vbox Archives]$ ls
archive.tar  home
[viveksingh@vbox Archives]$
```

tar -xvf archive_name.tar

```
[viveksingh@vbox Archives]$ tar -xvf archive.tar
home/viveksingh/Documents/
home/viveksingh/Documents/viveksingh.txt
home/viveksingh/Documents/vivek.txt
home/viveksingh/Documents/file1.txt
home/viveksingh/Documents/logfile.txt
home/viveksingh/Documents/sampleddata.txt
home/viveksingh/Documents/archive.tar
[viveksingh@vbox Archives]$ ls
archive.tar  home
[viveksingh@vbox Archives]$ cd home
[viveksingh@vbox home]$ ls
viveksingh
[viveksingh@vbox home]$ cd viveksingh
[viveksingh@vbox viveksingh]$ ls
Documents
[viveksingh@vbox viveksingh]$ cd Documents
[viveksingh@vbox Documents]$ ls
archive.tar  file1.txt  logfile.txt  sampleddata.txt  viveksingh.txt  vivek.txt
[viveksingh@vbox Documents]$
```

Alternatively, the following single command can be used to extract and decompress in one step:

tar -xvzf archive_name.tar.gz

Explanation:

-x is for extracting the archive.

-v for verbose output.

-z tells tar to first uncompress the .gz archive.

-f specifies the file name.

Example:

tar -xvzf project_files.tar.gz

```
[viveksingh@vbox Documents]$ tar -xvzf archive_compression.tar.gz
home/viveksingh/Documents/
home/viveksingh/Documents/viveksingh.txt
home/viveksingh/Documents/vivek.txt
home/viveksingh/Documents/file1.txt
home/viveksingh/Documents/logfile.txt
home/viveksingh/Documents/sampleddata.txt
home/viveksingh/Documents/archive.tar
[viveksingh@vbox Documents]$
```

This command decompresses and extracts all files from the project_files.tar.gz archive, restoring them to their original directory structure.

4)Extracting to a Different Location

To extract the files to a different location, the -C flag is used to specify the target directory:

tar -xvzf archive_name.tar.gz -C /path/to/target_directory

Example:

tar -xvzf project_files.tar.gz -C /home/user/extracted_files

```
[viveksingh@vbox Documents]$ tar -xvzf archive_compression.tar.gz
home/viveksingh/Documents/
home/viveksingh/Documents/viveksingh.txt
home/viveksingh/Documents/vivek.txt
home/viveksingh/Documents/file1.txt
home/viveksingh/Documents/logfile.txt
home/viveksingh/Documents/sampleddata.txt
home/viveksingh/Documents/archive.tar
[viveksingh@vbox Documents]$
```

This command decompresses and extracts the archive to /home/user/extracted_files, keeping the original directory structure intact.

RESULTS

according to the steps specified in the protocol The project produced the following results.

Successful tarball creation:

The tar utility efficiently groups multiple files into a single collection. As a result, Tarble preserves the directory and file structure. which is a simple method to manage large amounts of data in a single file

Effective compression:

Using gzip reduces the size of the tarball significantly. For example, a 200 MB tarball is reduced to about 100 MB, demonstrating the effectiveness of compression in saving disk space and increasing file transfer efficiency....

Efficient extraction:

The compressed file was successfully extracted back to its original format. It preserves the directory structure. The ability to extract archives to another location provides flexibility in file management. This is especially true when dealing with backups or moving files.

KEY LEARNINGS AND OBSERVATIONS

The following key learnings and observations occurred throughout the project.

Reducing file size:

Compressing the tarball with gzip reduces the file size significantly. This is ideal if storage space is limited. Or in cases where files need to be transferred quickly over the network.

Organizational archiving:

Using tar helps group files together into a single file. Makes it easy to manage and transport file collections Especially when dealing with multiple subdirectories...

Versatility of tar and gzip:

These utilities provide a variety of options and flags that can be customized for specific use cases, such as incremental backups. Archiving with rights and extracting files to another directory.

APPLICATION IN REAL-WORLD

This project has direct applications in various real-world situations, e.g.

System Backup: Archiving the project directory helps create regular backups of important data.

Data Transfer: Compressing collections before transferring them over the network reduces transfer time and saves bandwidth.

File storage optimization: Storing and compressing files frees up disk space. Helps to use storage space more efficiently

CONCLUSION

The project successfully demonstrates the process of creating and extracting logs using the tar and gzip utilities in a Linux environment, providing a practical understanding of how these tools can be used to efficiently manipulate files and directories. This process helps simplify management by combining multiple files. While adding gzip to compress the storage ensures that the storage is optimized. Reducing the file size also increases data transfer speed. This makes it easier to transfer large amounts of data across systems or networks.

Additionally, the ability to extract and decompress these archives to different locations. It also provides a high level of flexibility. Users can not only archive files for storage. But it can also browse and restore those files to any directory or location. easily This makes these tools essential for backup, deployment, and migration tasks. This adaptability is especially useful in real-world file management situations where server-resident data needs to move around. Back up data to another location or restore after archive

throughout the project Gain valuable insights into the benefits of consolidation and compression. These benefits include more efficient storage. This is because compressed files take up less disk space. This makes it easier to maintain a clean and organized file system. Additionally, simplifying file management makes it easier for administrators and developers to manage complex file structures. In particular, faster data transfer when managing backups or version control is another important benefit. This is because compressed files are smaller.

PLAGIARISM REPORT

Checked in Duplichecker.com

duplichecker.com

Results

Go Pro

✓ Deep search

✓ Support

✓ Upto 25,000 words

✓ Accurate Reports

✓ No Ads

Try Now

Scan Properties

Number of Words : 1000

Results Found : 3

To or From

Binary Translator

To or From

PDF Converter

7% Plagiarism

93% Unique

Make it Unique

Start New Search

Check Grammar

AI Content Detector

To check plagiarism in photos click here

Reverse Image Search

Don't Know Where to Find Content?

Find Content

logs using the tar and gzip utilities in a Linux environment, providing a practical understanding of how these tools can be used to efficiently manipulate files and directories. This process helps simplify management by combining multiple files. While adding gzip to compress the storage ensures that the storage is optimized. Reducing the file size also increases data transfer speed. This makes it easier to transfer large amounts of data across systems or networks.

Additionally, the ability to extract and decompress these archives to different locations. It also provides a high level of flexibility. Users can not only archive files for storage. But it can also browse and restore those files to any directory or location. easily This makes these tools essential for backup, deployment, and migration tasks. This adaptability is especially useful in real-world file management situations where server-resident data needs to move around. Back up data to another location or restore after archive

throughout the project Gain valuable insights into the benefits of consolidation and compression. These benefits include more efficient storage. This is because compressed files take up less disk space. This makes it easier to maintain a clean and organized file system. Additionally,

Similarity 17%

Looking To Improve Your Website's Load Speed? Meet PageSpeed ...

<https://seo360.com.mx/en/technical-seo/looking-to-improve-your-websites-load-speed-meet-pagespeed-insights>

Page | 16