

1. Taking a Backup of Ubuntu Server with rsync

[+] This takes a backup of the specified dirs (`/home`, `/etc`, and `/var/www`) and saves to dest dir (`/backups/files`).

Command:

```
rsync -av --delete /home /etc /var/www /backups/files
```

- **-a**: Archive mode, which preserves permissions, timestamps, symbolic links, and other attributes.
- **-v**: Verbose mode, which provides detailed information about the transfer process.
- **--delete**: Deletes files in the destination directory that are not present in the source directory.

2. Compressing the Backup Directory with gzip

[+] This creates an archive of the directory and compresses it using gzip.

Command:

```
tar -czvf /backups/backup_files.tar.gz -C /backups files
```

- **-c**: Create a new archive.
- **-z**: Compress the archive with gzip.
- **-v**: Verbose mode, which provides details of the files being processed.
- **-f**: Specifies the name of the archive file.
- **-C /backups**: Change to the `/backups` directory before processing the `files` directory to avoid including the full path in the archive.

3. Transferring the Zipped Backup File to a Local Machine with scp

[+] run this command on your Local NOT server

Command:

```
scp -r root@49.13.23.4:/backups/backup_files.tar.gz /backup-folder-on-
```

```
local
```

- **-r**: Recursively copy entire directories (useful if copying a directory instead of a single file).

4. Transferring the Zipped Backup File from Local to Server Using scp

Command:

```
scp /backup-folder-on-local/backup_files.tar.gz  
user@49.13.23.4:/path/on/server
```

5. Restoring the Compressed (tar.gz) Backup File on the Server

[+] You need to extract the contents using the **tar** command (step 1) and then copy the necessary files to their appropriate locations (step two).

Command to Extract the Backup File:

```
tar -xzvf /path/on/your/server/backup_files.tar.gz -C /path/to/extract
```

- **-x**: Extract the archive.
- **-z**: Decompress the archive with gzip.
- **-v**: Verbose mode.
- **-f**: Specifies the name of the archive file.
- **-C /path/to/extract**: Change to the specified directory before extracting the files.

Command to Copy Specific Files (e.g., nginx.conf):

```
cp /path/to/extract/files/etc/nginx/nginx.conf /etc/nginx/nginx.conf
```

This command copies the extracted **nginx.conf** file to its appropriate location.

Example Workflow:

1. Backup:

```
rsync -av --delete /home /etc /var/www /backups/files  
tar -czvf /backups/backup_files.tar.gz -C /backups files
```

2. Transfer to Local:

```
scp -r root@49.13.23.4:/backups/backup_files.tar.gz  
/Users/username/Desktop/your-backup-folder-on-local
```

3. Transfer Back to Server:

```
scp /Users/username/Desktop/your-backup-folder-on-  
local/backup_files.tar.gz username@49.13.23.4:/path/on/your/server
```

4. Restore:

```
tar -xzvf /path/on/your/server/backup_files.tar.gz -C /path/to/extract  
cp /path/to/extract/files/etc/nginx/nginx.conf /etc/nginx/nginx.conf
```

These steps should provide a robust process for backing up, compressing, transferring, and restoring data on our Ubuntu server.

Please let me know if anything needs be improved. xoxo DFK