H16S35 - Managing a web server

4
- Backups
- Access

michael.ferrie@edinburghcollege.ac.uk

Edinburgh College

tar (tape archive)
Gzip and Bzip2

michael.ferrie@edinburghcollege.ac.uk

tar

- Creates an archive file – sometimes called a tarball
- Combines multiple files into one file
- Always use the –f option LAST
  tar --create --file=archive.tar file1 file2
- Is the same as
  `tar -c -f archive.tar file1 file2`
- Is the same as
  `tar -cf archive.tar file1 file2`
- Two more commands –v = verbose and you swap out c for x to extract

tar examples

- Create a backup of all of Bob's Documents
  `tar -cvf docs.tar /home/bob/Documents`
- Create a backup of all of Bob's Music
  `tar -cvf music.tar /home/bob/Music`
- Restore a backup of all of Bob's Documents to current directory
  `tar -xvf docs.tar`

Gzip (GNU Zip)

- Compresses a single file using gzip uses the Lempel–Ziv algorithm the amount of compression obtained depends on the size of the input
  Typically, text such as source code or English is reduced by 60–70%
- Compression is generally much better than that achieved by LZW, or adaptive Huffman coding.
- Example, gzip and gunzip:
  `gzip file.txt / gunzip file.gz`

Tar + Gzip = tar.gz

- Tar only combines files into one single file, Gzip only compresses single files joining them together = archiving + compression tar.gz

- Example – create a tar.gz file - with gzip
  `tar -cvzf docs.tar.gz /home/bob/Documents`

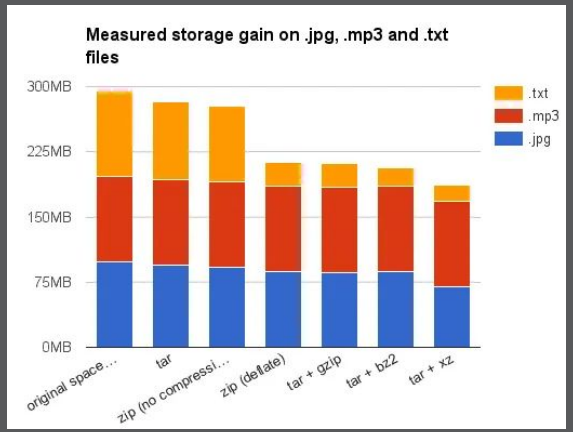
- Extract the file to the current directory:
  `tar -xvf docs.tar.gz`

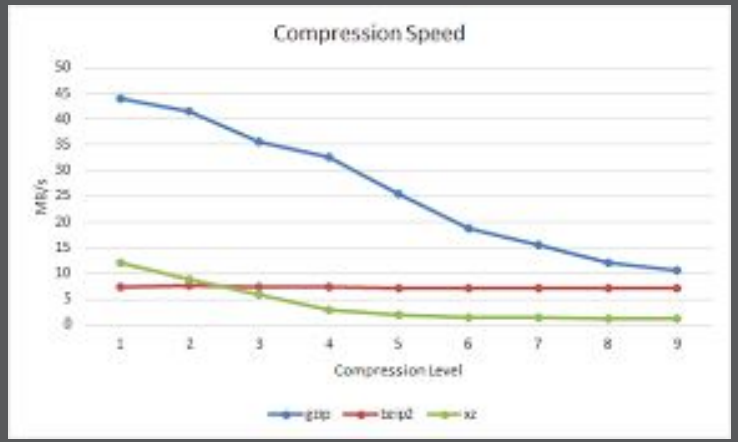
Tar + Bzip2 = tar.bz2

- Tar + bzip2 gives more compression than with gzip but is slower to run, tar + bizip2 gives a tar.bz file

- Example – create a tar.bz file - with bzip2
  `tar -cvjf docs.tar.bz2 /home/bob/Documents`
- Extract the file to the current directory:
  `tar -xvf docs.tar.bz2`

Gzip vs Bzip2





- Gzip is faster but bzip creates smaller archives

Edinburgh College

scp - secure copy

SCP

- SCP is a network protocol, based on the BSD RCP protocol, which supports file transfers between hosts on a network on TCP port 22 by default
- SCP uses Secure Shell (SSH) for data transfer and uses the same mechanisms for authentication, thereby ensuring the authenticity and confidentiality of the data in transit
- A client can send (upload) files to a server, optionally including their basic attributes (permissions, timestamps). Clients can also request files or directories from a server (download)
- Example scp a file from one machine to another:

```
scp SourceFile user@host:directory/TargetFile
scp myfile.txt michael@192.168.1.1:/home/michael/Documents
```

SCP Syntax

scp file.txt username@10.10.0.2:/remote/directory
-P Specifies the remote host ssh port
-p Preserves files modification and access times
-q Use this option if you want to suppress the progress meter and non-error messages
-C This option will force scp to compresses the data as it is sent to the destination machine
-r This option will tell scp to recursively copy directories

SCP Examples

- Copy a file from remote host to the current working directory

```
scp bob@192.168.100.5:/home/bob/Documents/file1
```

- Copy a file from the current working directory to a remote host

```
scp file1
bob@192.168.100.33:/home/bob/Documents/file1
```

Edinburgh College

ftp - file transfer protocol
sftp - ssh file transfer protocol

FTP

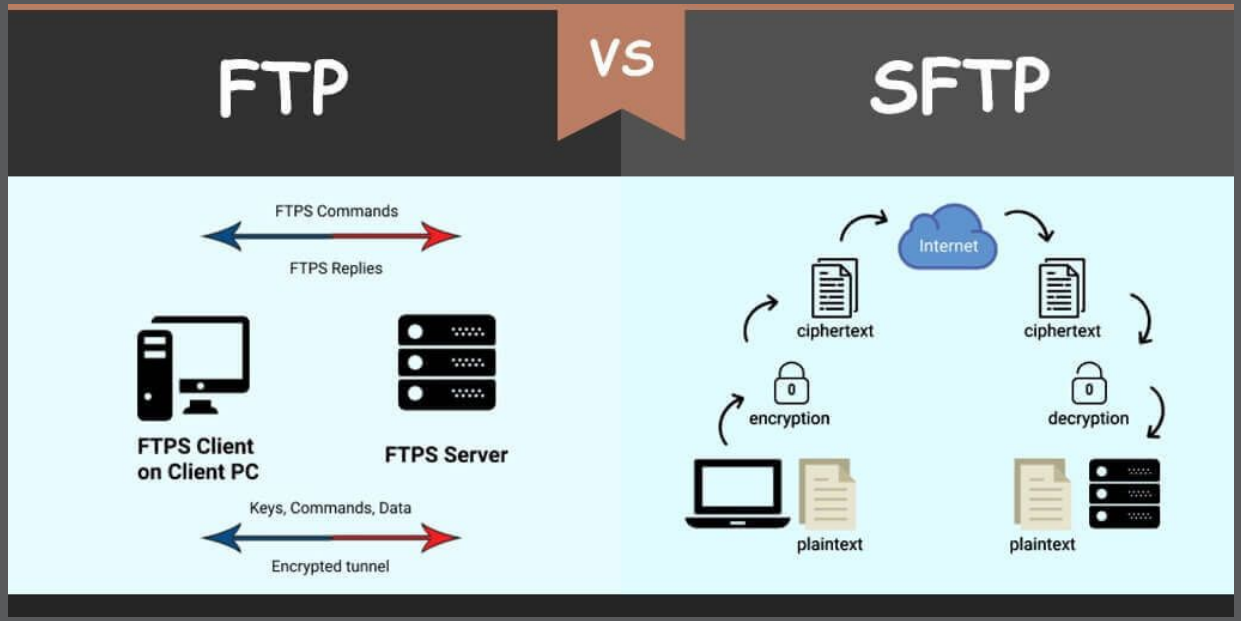
- The File Transfer Protocol is used for the transfer of files between a client and server on a computer network using port 21
- FTP is built on a client-server model architecture using separate control and data connections between the client and the server
- FTP users may authenticate themselves with a clear-text sign-in protocol, normally in the form of a username and password, but can connect anonymously if the server is configured to allow it
- For secure transmission that protects the username and password, and encrypts the content, FTP is often secured with SSL/TLS (FTPS) or replaced with SSH File Transfer Protocol (SFTP)

SFTP

- SSH File Transfer Protocol (also Secure File Transfer Protocol, or SFTP) is a network protocol that provides file access, file transfer, and file management over any reliable data stream
- Designed by IETF as an extension of the Secure Shell protocol (SSHv2) to provide secure file transfer capabilities
- Example connected to an sftp shell:

```
sftp sftpuser@216.200.116.229
sftpuser@216.200.116.229's password: *****
sftp> put Hello-World.txt
sftp> get file1.pdf
```

SFTP vs FTP


FTP
VS
SFTP
FTPS Commands
FTPS Replies
FTPS Client
on Client PC
FTPS Server
Keys, Commands, Data
Encrypted tunnel
Internet
ciphertext
ciphertext
encryption
decryption
plaintext
plaintext

SFTP vs FTP

FTP

FTP classic

- Plain FTP
- Clear-text password sent over the network
- Typically runs over TCP port 21
- Defined by RFC 959 and 1123
- Implemented in FTP/SSL component

FTP/SSL

FTP over TLS/SSL

- Often called 'FTPS'
- Often called 'Secure FTP'
- Plain FTP over TLS/SSL channel
- Password is encrypted
- Transfer is encrypted
- Typically runs over TCP port 21 or 990
- Defined by RFC 959, 1123, 4217 and 2228
- Implemented in FTP/SSL component

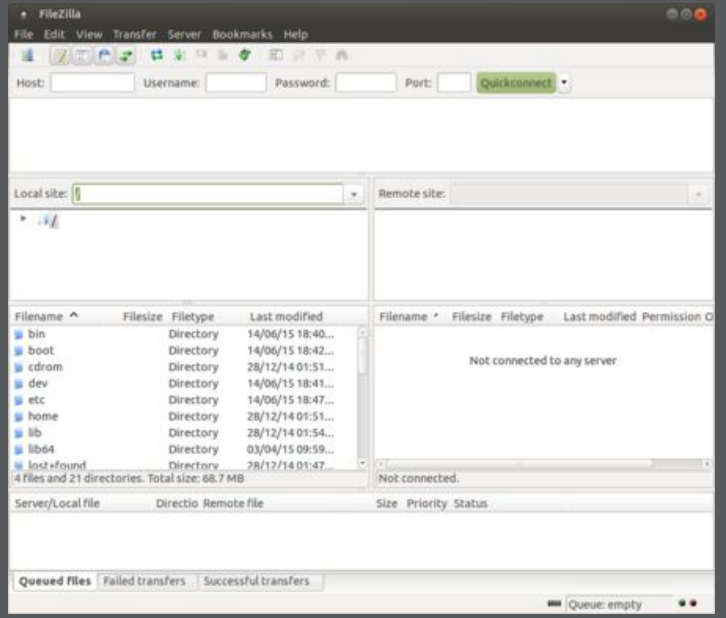
SFTP

SSH File Transfer Protocol

- SSH File Transfer Protocol
- Has nothing common with original FTP
- Often called 'Secure FTP'
- Password is encrypted
- Transfer is encrypted
- Typically runs over TCP port 22
- RFC not yet finished
- Implemented in SFTP client component
- Implemented in SFTP server component

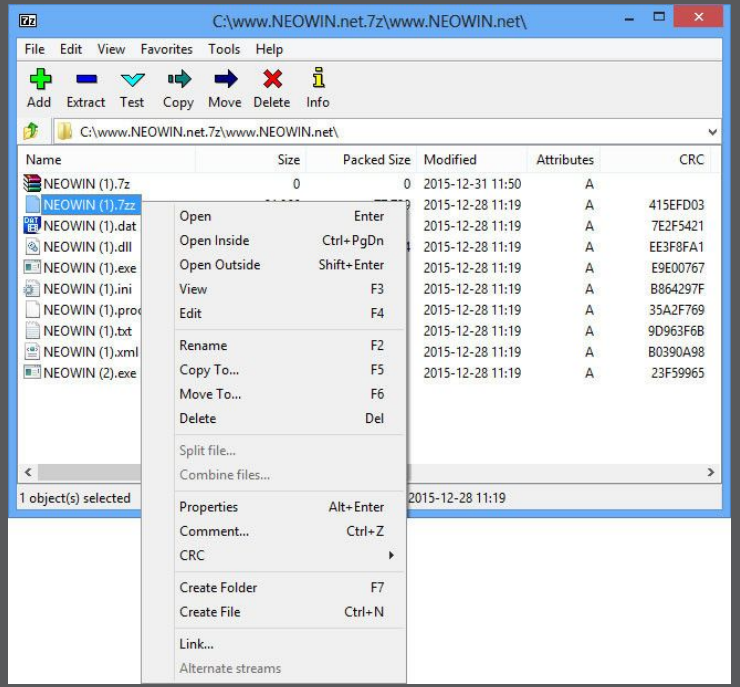
FileZilla

- FileZilla is a free software, cross-platform FTP application, consisting of FileZilla Client and FileZilla Server
- Available for Windows, Linux, and macOS, server binaries are available for Windows only - server and client support FTP and FTPS (FTP over SSL/TLS), while the client can in addition connect to SFTP servers



7-Zip

- A file archiver with a high compression ratio
- Free software with open source, most of the code is under the GNU LGPL license
- You can use 7-Zip on any computer, including a computer in a commercial organization



Edinburgh College

Thanks for listening
Any questions?

michael.ferrie@edinburghcollege.ac.uk