

Yafc ftp client

DATA SHOVEL



The console-based yafc client gives users a convenient, simple, and interactive approach to exchanging data with ftp and SSH servers. **BY HAGEN HÖPFNER**

Although most Linux distributions typically include an ftp program, it is likely to be less than intuitive. And the same thing can be said about the web-based ftp solutions offered by various providers.

Modern desktops like Gnome or KDE include ftp extensions that support drag and drop, but they are of little use if you need to set the permissions for the files on the server, and they do not lend themselves to scripting. The yafc ftp client [1] (Yet Another Ftp Client) might be slightly older – and console based, like *ftp* – but it also offers more options and is more convenient to use (Figure 1).

On Ubuntu, you can *sudo apt-get install yafc* to install the tool on your hard disk. SUSE has version 1.1.1 as a binary package, and you can run the package manager to install it. If the distribution of your choice does not include yafc, you will find the source code for a build-it-yourself version on the project website.

Before you can use ftp and yafc to transfer data, you need access to the ftp server at the other end. On top of this,

you need credentials – a username and a password. To open a connection to the server in a terminal window, type a command with the following syntax:

```
$ yafc Options Protocol://User@
:Password@Server:Port/Directory
```

The most useful options here include *-a* for anonymous access (which the server

needs to support) or *-q* to suppress the welcome text. Typing *yafc -h* displays a full list of options.

If you do not specify the protocol, yafc will default to *ftp://*. Alternatively, the client also supports encrypted *ssh://* communications (see Figure 2).

The server at the other end will need to be an SSH server rather than an ftp server in this case. In contrast to ftp, the

```
yafc hopfnehn@172.17.0.221:~
File Edit View Terminal Tabs Help
hopfnehn@wrkst221:~$ yafc 172.17.0.221
yafc 1.1.1 Copyright (C) 1998-2001 Martin Hedenfalk <mhe@home.se>.
This program comes with ABSOLUTELY NO WARRANTY; for details type 'warranty'.
This is free software; type 'copyright' for details.

Connecting to wrkst221.i-u.de (172.17.0.221) at port 21...
ProFTPD 1.3.1 Server (Debian) [172.17.0.221]
Password required for hopfnehn
password: *****
User hopfnehn logged in
remote system: UNIX Type: L8
yafc hopfnehn@172.17.0.221:~> ls
Bilder/      Dokumente/  SVNROOT/    Vorlagen/   öffentlich/
Desktop/    Musik/      Videos/     texmf/
yafc hopfnehn@172.17.0.221:~>
```

Figure 1: The console-based yafc ftp client supports text commands rather than mouse signals.

```

yafc hoepfner@gi-mms.de:~
File Edit View Terminal Tabs Help
hopfnehn@wrkst221:~$ yafc ssh://gi-mms.de
yafc 1.1.1 Copyright (C) 1998-2001 Martin Hedenfalk <mhe@home.se>.
This program comes with ABSOLUTELY NO WARRANTY; for details type 'warranty'.
This is free software; type 'copyright' for details.

login (hopfnehn): hoepfner
Password:
remote system: SSH (version 3)
yafc hoepfner@gi-mms.de:~>

```

Figure 2: Yafc supports ftp-based data transmission and can also handle encrypted SSH connections.

advantage is that cleartext passwords are not sent.

Besides the protocol, you can also omit the username. Yafc will then supply your local username. The `-u` option suppresses this default behavior, and then yafc will prompt you to enter a username. In addition to the username, you can type a password for your account on the server with a colon as the separating character. For security reasons, this is only recommended for automated scripts – or situations in which you are sure that nobody is shoulder surfing.

Although you must type the server name, the port is not required unless the server at the other end uses a non-standard port. If needed, you can specify the directory on the server in which you want yafc to start working.

The command

```
$ yafc ssh://hopfnehn@172.17.0.221:/documents
```

opens an SSH-encrypted connection to the server at `172.17.0.221` for the user `hopfnehn`. Then the server prompts the user for the password and grants the ftp client to the `/documents` directory.

```

yafc hopfnehn@172.17.0.221:~/Dokumente
File Edit View Terminal Tabs Help
hopfnehn@wrkst221:~$ yafc hopfnehn@172.17.0.221
yafc 1.1.1 Copyright (C) 1998-2001 Martin Hedenfalk <mhe@home.se>.
This program comes with ABSOLUTELY NO WARRANTY; for details type 'warranty'.
This is free software; type 'copyright' for details.

hopfnehn@172.17.0.221's password:
remote system: SSH (version 3)
yafc hopfnehn@172.17.0.221:~/Dokumente> ls -l --color
total 95525
drwxr-xr-x  2 hopfnehn hopfnehn  4096 Apr 11 16:10 Eigene Bilder/
drwxr-xr-x  2 hopfnehn hopfnehn  4096 Apr 11 16:10 Meine Musik/
drwxr-xr-x  2 hopfnehn hopfnehn  4096 Apr 11 16:10 Meine Videos/
drwxr-xr-x  3 hopfnehn hopfnehn  4096 Apr  4 13:28 icsoft08/
-rw-r--r--  1 hopfnehn hopfnehn 75045 Apr  2 17:26 icsoft08.zip
drwxr-xr-x  2 hopfnehn hopfnehn  4096 Apr  2 17:26 icsoft08_submission/
yafc hopfnehn@172.17.0.221:~/Dokumente>

```

Figure 3: In interactive mode, yafc behaves almost like a shell and even supports color highlighting (`--color`) and detailed information (`-l`).

Yafc can handle multiple, parallel connections to servers. The command

```
$ yafc ssh://hopfnehn@172.17.0.221:/documents &
ssh://hopfnehn@172.17.0.222
```

tells yafc to open another connection parallel to the existing one. To toggle between the two connections, you can type the `switch` command in interactive mode.

Instead of opening a connection to a server when you launch yafc, you can launch the client in interactive mode. The command

```
yafc> open ssh://hopfnehn@172.17.0.221:/documents ssh://hopfnehn@172.17.0.222
```

at the yafc command-line prompt opens connections to the required servers.

Interactive Mode

After opening a connection, yafc switches to interactive mode, from which you can type commands to transfer files. Several commands are available

for this, and they also support navigating the server and client directories.

Just like the familiar shell command, `ls` lists the current server directory, `cd` changes to another directory, and `pwd` tells you your current working directory (Figure 3). To run these commands locally instead of on the server, just type an exclamation mark before the command. To get a list of commands, enter `help`.

The `put` command transfers a file from the client to the server, and `get` retrieves a file from the server and stores it on the client. Yafc might prompt you before overwriting existing files or directories. One big advantage over other console-based ftp programs is yafc's ability to recursively transfer directories. The `put -r test/` command not only copies the `test` directory to the server, but also all the files and subdirectories below it. If needed, recursion in combination with `get` helps users create complete backup copies of their server data.

After exchanging data, you can close your connection(s) with the `close` command and quit yafc by typing `exit`.

Conclusions

Console-based ftp clients are the tool of choice for manual or scripted file transfers for large numbers of files. Yafc, as one of the most convenient representatives of this species, has far more in the line of functionality to offer than the standard ftp client. Besides the options I looked at here, it also supports proxies or NOHUP data transmissions, during which transmission continues despite the user logging off – as long as the computer is running. ■

INFO

[1] yafc: <http://yafc.sourceforge.net/>

THE AUTHOR

Hagen Höpfner is an Assistant Professor for databases and information systems at the International University in Germany (<http://www.i-u.de>) at



Bruchsal. He has written, co-written, and published a number of text books and research papers. In his free time, Hagen writes lyrics, composes songs, sings, and plays guitar.