Fastscp project

Jacques Gélinas
jacquesgelinas2407 at gmail.com

January 15, 2018

Abstract

The **fastscp** project is a solution to get the fastest file copy on a local network. It uses a simpler crypt to achieve a very high performance.

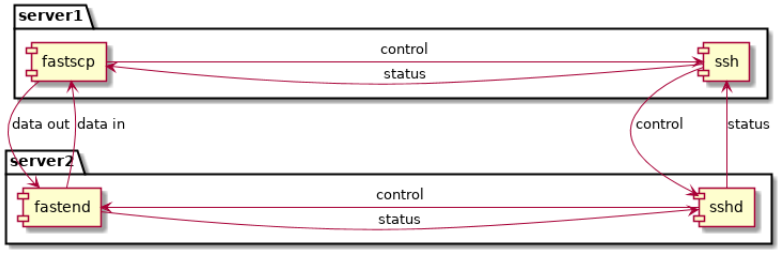
Introduction

The ssh application suite is the preferred way to establish a connection to a remote server. It is also the best way to copy files back and forth to this server. Unfortunately, it is kind of slow. You get something like 40 megabytes per second. Not bad, but if you have 100 gigabytes to transfer, it adds up.

A gigabit network (pretty much the norm today) can sustain roughly 110 megabytes per second. Is it possible to get that kind of speed or even more (10 gigabits network are getting cheaper by the day) ?

A quick view

Here is a schema explaining the concept. The arrows are showing the direction of the TCP connections.



Here are the steps of a connection:

- User execute **fastscp file1 user@server:/dir/file1**.
- fastscp executes: **ssh user@server fastend**.
- fastend setup a listening TCP socket on a OS allocated port.
- fastend sends the port number to fastscp through the ssh connection.
- fastscp connects to this port. This will become the data connection.
- fastscp sends some crypto initialization stuff to fastend using the ssh connection.
- fastscp sends a command telling it will send a file to be stored in /dir/file1 to fastend, again through the ssh connection.

- fastscp sends the file using the data connection.
- fastscp sends the amount of bytes sent on the data connection to fastend, using the ssh connection.
- fastend confirms reception of the bytes using the ssh connection.

Getting started

Just install the fastscp package on two servers. Then from one, just attempt a copy to the other. No configuration is needed. You only needed ssh connectivity. No requirements on ssh is needed. Connection may work with or without a password.

Fastscp works like scp syntax wise. You can do the following:

- fastscp file_path user@server:/dir/filename
- fastscp file1 file2 user@server:/dir
- fastscp user@server:/dir/file file_path
- tar zcf - | fastscp – - user@server:/dir/file.tar.gz

Fast crypto

The data is sent over a TCP session. It has its own encryption scheme.

The crypto is rather simple and should prove hard to crack. It adds an overhead to get started, but on high speed local network, it has little impact.

- 160kbytes of random data (taken from /dev/urandom) is transferred from the client to the server using the SSH control session. So this data is protected by SSH crypto.
- The first bytes of random data are used to decide how much of it will be used, from half of it to all. Someone spying on the connection will always see 160k of data transferred, but can't tell how much is used.
- This data is used in a loop on the client and server. Each byte sent is xored with one random byte. Each byte received is xored with the corresponding byte.

Implementation

The source code is available using **subversion** at

```
http://svn.solucorp.qc.ca/repos/solucorp/fastscp/trunk
```

You can build it simply by issuing

```
make
make install
```

or on fedora

```
make buildrpm
```

To compile it, you need the linuxconf-devel and linuxconf-lib package. You can grab the latest source here

```
http://svn.solucorp.qc.ca/repos/solucorp/linuxconf/trunk
```

then you can build an rpm for it

```
make buildrpm
```

Linuxconf is not maintained anymore, but the library is. At some point it will be renamed...