

SSH2 Subsystems

Finnbarr P. Murphy

(fpm@fpmurphy.com)

SSH2 subsystems are a useful convenience feature to predefine remote commands for SSH clients to invoke easily. Subsystems provide a layer of abstraction for defining and invoking the remote commands. A subsystem need not be a separate program; it can invoke a function built into the SSH server itself.

SFTP is the most common SSH subsystem that you are going to encounter. For example on Linux distributions, the default `/etc/ssh/sshd_config` file defines one subsystem, This is the configuration line on Fedora 20:

```
# override default of no subsystems
Subsystem      sftp      /usr/libexec/openssh/sftp-server
```

Do not remove or comment out the above line. It is required for `sftp` to work.

Note that the subsystem syntax is slightly different between OpenSSH and SSH2 implementations.

```
# SSH version 2
Subsystem-sftp      /usr/libexec/openssh/sftp-server

# OpenSSH version 2
Subsystem sftp      /usr/libexec/openssh/sftp-server
```

Subsystems can be defined in the SSH v2 server configuration file using the following syntax.

```
Subsystem-      argument
```

The argument is the command which will be executed when the subsystem is requested.

```
$ ssh user@remote -s
```

The argument can be a list of commands separated with a semicolon, or it could be the path to a shell script.

Alternately, you can use the syntax `internal-` to invoke an in-process server.

```
internal-      argument
```

This may simplify configurations using `ChrootDirectory` to force a different filesystem root on clients. This should be used for example when the user is chrooted and does not have access to the server binary.

For personal use only