

F1/10 Autonomous Racing - Lab Session

How To Use SFTP to Transfer Files/Packages to the racecar from your local machine/VM

Instructor: Madhur Behl (madhur.behl@virginia.edu)

SFTP, which stands for Secure File Transfer Protocol, is a separate protocol packaged built into SSH that can be used as a method of transferring files between two remote systems.

By default, SFTP uses the SSH protocol to authenticate and establish a secure connection. Because of this, the same authentication methods are available that are present in SSH.

If you can connect to the machine using SSH, then you have completed all of the necessary requirements necessary to use SFTP to manage files.

Terminology:

- `local machine` : This is your Ubuntu machine/VM.
- `remote machine` : This is the Nvidia Jetson TX2 on the racecar.

[1] Establish SFTP link with the racecar

Ensure your local machine is connected to the racecar and you can successfully SSH. This is just a test - You can now close the SSH session.

Now establish an SFTP session with the racecar by issuing the following command from the terminal on the local machine.

```
sftp nvidia@192.168.1.1
```

The password is the same as what you used for the SSH connection.

You will connect to the remote machine and your prompt will change to an SFTP prompt.

```
racecar@racecar-vm:~$ sftp nvidia@192.168.1.1
nvidia@192.168.1.1's password:
Connected to 192.168.1.1.
sftp>
```

[2] Getting Help in SFTP

The most useful command to learn first is the help command. This gives you access to a summary of the other SFTP commands. You can call it by typing either of these in the prompt:

```
sftp> help
```

or

```
sftp> ?
```

This will display a list of the available commands:

```
Available commands:
bye                               Quit sftp
cd path                           Change remote directory to 'path'
chgrp grp path                    Change group of file 'path' to 'grp'
chmod mode path                  Change permissions of file 'path' to 'mode'
chown own path                   Change owner of file 'path' to 'own'
df [-hi] [path]                 Display statistics for current directory or
                                filesystem containing 'path'
exit                              Quit sftp
get [-Ppr] remote [local]       Download file
help                              Display this help text
lcd path                          Change local directory to 'path'
. . .
```

We will explore some of the commands you see in the following sections.

[3] Navigating with SFTP

We can navigate through the remote system's file hierarchy using a number of commands that function similarly to their shell counterparts.

First, let's orient ourselves by finding out which directory we are in currently on the remote system. Just like in a typical shell session, we can type the following to get the current directory:

```
sftp> pwd
```

We can view the contents of the current directory of the remote system with another familiar command:

```
sftp> ls
```

```
sftp> pwd
Remote working directory: /home/nvidia
sftp> ls
Desktop                Documents              Downloads
Music                  Pictures              Public
Templates              Videos               catkin_ws
depend_ws              examples.desktop     jetson_clocks.sh
report_ip_to_host.sh  ros_install.sh       tegrastats
test                   weston.ini
sftp>
```

Lets create a `test_ws` direcotry on the remote mahcine that we will use for testing some file transfer commands. This does not need to be a rela ROS workspace. Its only for geting fmailair with `sftp` .

```
sftp> mkdir test_ws
sftp> cd test_ws
sftp> mkdir src
sftp> cd src
```

So `test_ws` has a familiar folder structure as a ROS workspace. We will now learn how to send files from our local machine to the remote machine.

```
sftp> mkdir test_ws
sftp> cd tes
test/      test_ws/
sftp> cd test_ws
sftp> mkdir src
sftp> ls
src
sftp> cd src/
sftp>
```

We can now traverse the remote file system, but what if we need to access our local file system? We can direct commands towards the local file system by preceding them with an `l` for local.

All of the commands discussed so far have local equivalents. We can print the local working directory:

```
sftp>lpwd
```

We can list the contents of the current directory **on the local machine**:

```
sftp> lls
```

```
sftp> lpwd
Local working directory: /home/racecar
sftp> lls
bags          Desktop      Downloads    github       Pictures     snap         Videos
catkin_ws     Documents   examples.desktop  Music       Public      Templates
```

Lets create a dummy package on the local machine that we will transfer to the remote machine.

```
sftp> mkdir dummy_package
sftp> lcd dummy_package
sftp> mkdir scripts
sftp> lcd scripts
```

Lets add a blank text file into the `scripts` sub-directory of the `dummy_package` folder. You can do so from a seperate terminal outside of the sftp terminal session. IN this example we we assume that you have added a file called `example.txt` in the `scripts` folder on te local machine.

```
racecar@racecar-vm:~$ cd dummy_package/scripts/
racecar@racecar-vm:~/dummy_package/scripts$ touch example.txt
racecar@racecar-vm:~/dummy_package/scripts$
```

[4] Transferring Local Files to the remote machine

Transferring files to the remote machine works with a `put` command:

Let us double check all the paths once before issuing a file transfer.

1. On the local system, let us check that the `dummy_package` directory is present in our local path.
2. On the remote machine, let us verify that the working directory is the `test_ws/src` directory. This is the destination folder where the contents will be transferred in to.
3. If the paths are correct: we can now issue a `put` command that will transfer the `dummy_package` folder and all of its contents to the `test_ws/src` folder on the remote machine/racecar.

```
sftp> put -r dummy_package
```

```
sftp> lpwd
Local working directory: /home/racecar
sftp> ll
bags          Documents      examples.desktop  Pictures  Templates
catkin_ws     Downloads      github            Public    Videos
Desktop       dummy_package  Music             snap
sftp>
sftp>
sftp> pwd
Remote working directory: /home/nvidia/test_ws/src
sftp> put -r dummy_package/
Uploading dummy_package/ to /home/nvidia/test_ws/src/dummy_package
Entering dummy_package/
Entering dummy_package/scripts
dummy_package/scripts/example.txt          100%   0    0.0KB/s   00:00
sftp>
```

Now we can verify that the contents were indedd transferred to the remote mahcine.

```
sftp> cd dummy_package/scripts/  
sftp> ls  
example.txt  
sftp>
```

[5] Alternate to SFTP - SCP (Secure Copy)

Sometimes its faster to transfer the files directly using the `scp` command.

The `scp` command syntax take the following form:

```
scp [OPTION] [user@]SRC_HOST:]file1 [user@]DEST_HOST:]file2
```

Local files should be specified using an absolute or relative path, while remote file names should include a user and host specification.

So to acheive the same transfer as above using a single line we could just type the following into a terminal (not a sftp session) on the local machine.

```
scp -r /dummy_package nvidia@192.168.1.1:/test_ws/src
```