

# Hippolyta Windows Off-Campus Usage Guide

## 1 cmdr

If you are a Windows user, you have probably been using PuTTY to access Hippolyta. Although PuTTY works well for basic use-cases, it is not the ideal platform for extensive usage of SSH. In order to provide a better interface for extensive usage of LAMMPS on the Hippolyta supercomputer, we will focus on using [cmdr](#), a *console emulator* with a flexible user interface and useful macros. A primary and useful difference between PuTTY and cmdr is that cmdr provides a linux-like interface for navigating your local file directory (so we can do things like ‘scp’ and ‘sftp’ - transfer files quickly and easily from your local machine to a remote server). cmdr relies on a plethora of other software (primarily [Git](#), [Conemu](#), and [Clink](#)) to provide an easy-to-use console on Windows.

## 2 Installation

Follow [this ZIP download link](#) or the link for the ‘Full’ download link on the [cmdr](#) page. You will need the Full download - we will be using Git, included in the Full version, to create and handle RSA keys. Unzip the folder in any location of your choice (I would recommend the root directory C:/ or the user directory C:/Users/User). Navigate into the folder in which you unzipped the file, and now navigate into the cmdr (e.g. C:/Users/User/cmdr) folder. I would recommend right-clicking the Cmdr.exe file and clicking Pin to Taskbar. To create a shortcut on your desktop, copy the current directory (.../cmdr/), navigate to your desktop, right-click and then left-click on Create Shortcut. Paste the directory and then type in Cmdr.exe (i.e. .../cmdr/Cmdr.exe).

## 3 Unix Directory Traversal

Open cmdr using your shortcut. Congratulations, you’re now in a Linux console emulator! Now, rather than using your mouse and Windows Explorer to navigate your file directory, you can now use Unix commands. If you’re already familiar with Unix, feel free to skip over to Section 4.

The two first commands to learn are `ls` and `cd`. `ls` shows us what is within a directory, while `cd` navigates into a directory. In Unix, both the empty string,

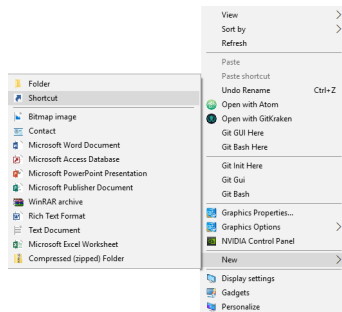


Figure 1: Create a shortcut by right-clicking on the desktop

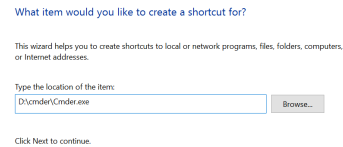


Figure 2: Provide the full path to `Cmder.exe`

space, and `..` indicate that you are querying the current directory (which you should see right above the blinking line. Type `ls` into your command line, and read through the output (cmdr either navigates to your user directory, which will take a while, or a cmdr user directory, which will be empty, by default). In Unix `..` moves *up* the ‘directory tree’ (if we imagine the filesystem as a tree, we are moving to the parent node of the current location). Try typing in `ls ..`.

Pick a directory within the folder you are currently in, which you saw when you typed `ls`. You can look inside the directory using `ls directory`. Moreover, you can look at sub-directories *ad infinitum* with `ls dir1/dir2`. A great time-saver when navigating file-systems with a Linux command interface is the `tab` key. Type out `ls` followed by a space and the first few letters of the directory name you just viewed, and then press `tab`. This makes your computer search for directories or files that start with the letters you’ve typed. If there was only one possible response, then your computer should have auto-completed your query, after which you can press **Enter** to view the contents of the directory. If nothing showed up, that means there were either no files/directories that matched your query, or there was more than one. Press `tab` again. If there were multiple results, then they should now be listed; otherwise, there will be no output.

Now let’s use the `cd` command. If you simply type `cd`, you won’t move anywhere - you’re telling the machine to move to the directory that it currently is. You can move up and down the directory tree by using the same format as `ls`. Try typing `cd ..` and `cd ../..`, and using the `tab` key as before. To switch between drives (for example, if you have a solid state `C:` drive and a hard drive `D:`), simply type the name of the drive followed by a colon (e.g. `D:`). This will place you in the default folder for that drive.

Experiment with these commands, and try to get comfortable moving around your filesystem - don’t worry, this comes with time and practice.

```
Host hippolyta
  HostName hippolyta.materials.cmu.edu
  User Username
  Port HippolytaPort
  IdentityFile ~/.ssh/id_rsa.pub
```

Figure 3: Run ssh-keygen from within the cmdr directory

Figure 4: Edit the parameters of the `config` file

As you probably already known from your experiments with PuTTY, SSH is tricky on Windows. Thankfully, this has been boiled down to just a few steps, with the help of the `cmdr`.

First, we'll need to generate your RSA keys. Open up `cmdr` and navigate to where you installed `cmdr` using the command line interface. Navigate into the `cmdr/vendor/git-for-windows/usr/bin/` directory. Now, enter the command `ssh-keygen`. It should prompt you for a file to place the generated key into, with the default set to `/c/Users/User/.ssh/id_rsa`, where `User` is your Windows username. Type **Enter** to accept this location - we'll be using the `.ssh` folder a bit here on out. Type **Enter** twice again (you generally don't put a passphrase on RSA keys). You now have RSA public and private keys for secure and quick off-campus logins - but we still have some work to do.

To make logging into Hippolyta easier, download the `config` file available on Canvas (or [here](#)). Navigate to where you downloaded the file and type `cp config ~/.ssh/`. Open the file using `vim` by typing `vim ~/.ssh/config` (note: Windows won't allow you to `cd ~/.ssh`, but you can use `ls ~/.ssh` to see what's in the folder). The file is configured to work well with how we've set things up - all you have to do is replace the word `Username` in the line `User Username` with your Hippolyta username, and the word `HippolytaPort` in the line `Port HippolytaPort` (omitted for security reasons). To do this with `vim`, press the `'i'` key (this puts you in `Insert` mode, so you can type and edit the file), scroll down using the down arrow key to the line including `User Username`, scroll to the right using the right arrow key, press backspace to delete `Username`, and replace it with your username. Do the same with `HippolytaPort`. After you've done this, press the `escape` key then type `:'wq'` (this command saves and closes the file - note that `vim` commands are case-sensitive).

Next, we need to make certain that the ssh-agent (the program that handles

[illegible]

Figure 5: Example `authorized_keys` file

the RSA keys) is setup properly. Download the `ssh-setup.bat` script available on canvas (or [here](#)). Copy the file into the `config/profile.d` subdirectory of the folder where `cmdr` is installed. Close and then re-open `cmdr`: it should take a second, and then print out “Identity added: C:/Users/User/.ssh/id\_rsa (C:/Users/User/.ssh/id\_rsa)”.

For the final part, you’ll need to be on campus. Open up your public key by typing `vim ~/.ssh/id_rsa.pub`. Using your mouse, select all of the text in the file. Copy the text using `Control+C` (note: right-clicking to copy the text will *not* work). Close the file using the command `:q!` (just in case you accidentally typed something, the `!` will preserve the original file contents). Now type `ssh hippolyta`. This will try to use your RSA key, but fail, since the server is not currently recognizing it (it may also ask for a passphrase for the key, just press `Enter`). Type in your Hippolyta password to log in (this is why you need to be on-campus - off-campus, Hippolyta won’t let you use a password). Once you are logged in, type the command `cd ~/.ssh`. Type out `vim authorized_keys` (this will create a new file). Type in “i” and then right-click - hopefully, this should paste your public key into the `authorized_keys` file. Press the `escape` key, then type `:wq`. Now type in `exit` to log-out of the server.

If all went well, you should be able to type `ssh hippolyta` and *not* be prompted for a password - this means that your RSA key is working. Usefully, you should also be able to use the command `scp` to copy files from and to Hippolyta, using `scp folder/to/file hippolyta:/directory/for/file/` to place files on the server and `scp hippolyta:/directory/containing/file folder/for/file/` to grab a file from the server ([here](#) is a useful guide on using `scp`). Use `scp -r` to copy folders (recursively down the directory tree). You can also use wildcards (\*) to grab multiple files with a similar naming pattern (e.g. ‘`scp hippolyta:/folder/file_*.extension .`’).