

User installed software

Lab groups are welcome to install software in their `hpc/group/<groupname>` space if `sudo` access is not required. This can be helpful for lab specific software or for when a group wants to strongly control software versions and packages.

Miniconda installation sample

Login to the DCC using [SSH](#):

```
mkdir -p /hpc/group/<groupname>/<netid>
wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
sh Miniconda3-latest-Linux-x86_64.sh
```

and then follow the instructions. The place to give as Miniconda install location should be `/hpc/group/<groupname>/<netid>/miniconda3`. It will offer to update your `~/.bashrc`, (init) which is what you want. Log out log back in and then you can run `conda install`, `pip install`, create environments, etc.

Python

Users who will be using Python through SSH and OnDemand should setup virtual environments using miniconda with the desired Python version and packages. After installing Minconda, create a virtual environment. In the example below, `test` is the name of the environment and `python=3.9` is the python version number.

```
conda create -n test python=3.9
conda activate test
conda install ipykernel
python -m ipykernel install --user --name test
```

In addition to using at the command line (using `conda activate test`), you will also see this new environment in OnDemand as an option on the Launcher when creating a new notebook. To switch to this new environment on a running notebook: - Go to Kernel menu on the left to 'change kernel' and select the environment. - click on the present kernel name on the top right corner and select the environment.

To deactivate the environment, you can use `conda deactivate` command.

Note: You can use `conda` or `pip` to install packages in this virtual environment.

For more information on using conda virtual environments, see [Managing Environments](#) in the conda user guide.