Research Computing Tutorial

David, Mayara, Sebastian

MIT

September 26, 2021

Overview

Workshop Outline

Getting Help

- **Econ Help:** https://econ-help.mit.edu/
- **Carl Anderson (E52-311)** econ-support@mit.edu
- **Slack:** Join mit-econ.slack.com and message **#econ-support**
- **Quickstart:** https://econ-help.mit.edu/kb/quick-start
- **Overview of Econ clusters:** https://econ-help.mit.edu/kb/rc-systems
- **Unix Commands:**
  - https://econ-help.mit.edu/kb/useful-unix-commands
  - `man <command>`
- **Available Modules:** https://econ-help.mit.edu/kb/env-modules

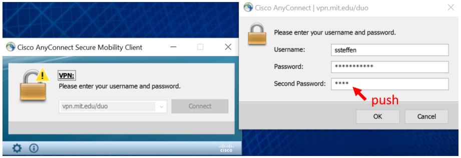
Workshop Outline

Servers and Clusters

- **Servers**: each has about has about 1.5TB of RAM and 72 CPUs.
    - supply.mit.edu
    - demand.mit.edu
    - blackmarket.mit.edu
    - econometrics.mit.edu
- **Clusters of servers**: High-performance computing.
    - reducedform.mit.edu
    - eosloan.mit.edu ("engaging" - Sloan only)
    - Host your own: AWS (Amazon), Microsoft Azure, IBM Cloud, Google, other IaaS/SaaS
- You connect to servers and clusters the same way, but running code on clusters is slightly different (won't cover today).

Workshop Outline

Before we begin: MIT Secure Wifi (on campus) or VPN (off campus)


Cisco AnyConnect Secure Mobility Client

VPN:
Please enter your username and password.
vpn.mit.edu/duo
Connect

Cisco AnyConnect | vpn.mit.edu/duo

Please enter your username and password.

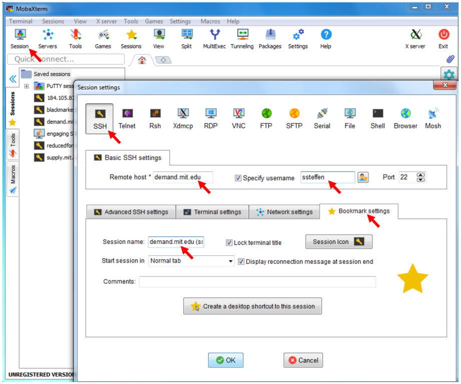
Username: ssteffen
Password: ***********
Second Password: ****

1. **Install Cisco AnyConnect:** http://kb.mit.edu/confluence/display/istcontrib/Cisco+AnyConnect+VPN+Landing+Page
2. **Connect to:** vpn.mit.edu/duo and follow Duo-Authentication steps

How to Connect to MIT Servers

- **Connect**: MobaXterm (Windows), Terminal (Mac).
  - Mac: `ssh -X <kerberos>@<hostname>.mit.edu`
  - Windows: Next slide

MobaXterm I


MobaXterm
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
Saved sessions
Session settings
SSH Telnet Rsh Xdmcp RDP VNC FTP SFTP Serial File Shell Browser Mosh
Basic SSH settings
Advanced SSH settings
Terminal settings
Network settings
Bookmark settings
Session Icon
Create a desktop shortcut to this session
OK
Cancel
UNREGISTERED VERSION

Remote host * demand.mit.edu

[✓] Specify username ssteffen

Port 22

Session name: demand.mit.edu (ss

[✓] Lock terminal title

Start session in Normal tab

[✓] Display reconnection message at session end

Comments:

1. Select **SSH session**
2. Enter **Host address** (i.e. supply.mit.edu) and **username** (your kerberos)
3. ...
4. Profit!

Brief Bash Tutorial

Navigation

- Get current dir: `pwd`
- Show current dir: `ls`, `ls -l`
- Move to dir: `cd <PATH>`, `cd ~`(Home), `cd ..` (Up), `cd -` (Toggle previous path)
- Get help: `man <COMMAND>`

Creation & Destruction

- Make a folder: `mkdir <NAME>`
- Delete a folder: `rm -r` $< PATH >$
- Make a file: `touch <NAME>`
- Open a file: `nano <PATH>` or `vim <PATH>`
- View file contents: `cat <NAME>`
- Delete a file: `rm <PATH>`
- Copy/Rename a file: `cp <PATH> <PATH>`
- Move a file: `mv <PATH> <PATH>`

Workshop Outline

How to Interact with MIT Servers: File transfers

- To run software on the server, the program you want to run cannot be saved (only) on your local machine - it needs to be accessible by server computers.
  - You can use your ECON account for this purpose. Each student with an ECON account has about 200GB of space that can be used for storing data and code (e.g. bbkinghome/mfelix).
- Files can be transferred from your local machine to server folders using Secure File Transfer Protocol (SFTP) software.
  - Windows: **MobaXterm**
  - Mac: **CyberDuck**. Alternatives: Fetch (download: https://ist.mit.edu/fetch) and FileZilla.
- You could also directly mount your ECON folder locally through Finder (Mac) or Windows File System (Windows).

How to Interact with MIT Servers: Run code

- **Interactive mode**
  - Windows: From MobaXterm, type xstata, rstudio, python, ...
  - Mac: Open **XQuartz** and type ssh -Y <kerberos>@<server>.mit.edu <program> (e.g. ssh -Y mfelix@demand.mit.edu stata-mp)
- **Non-interactive mode**
  - Same procedure from MobaXterm or from Mac Terminal.
  - E.g. stata-mp do myfile.do

Workshop Outline

Simple exercise: run a .do file

- Connect to any of the servers
- cd /bbkinghome/mfelix/workshop
- ls
- stata-mp -b do workshop_exercise.do

Screen

- The screen command creates persistent, virtual terminal shells that keep running even when you leave.
- **Attach a screen:** `screen -S <name>`
- **List your screens:** `screen -ls`
- **Detach a screen** `Ctrl+A, D`
- **Reattach a screen:** `screen -r`
- **Learn more:** `man screen`

Getting Help - Again!

- **Econ Help:** https://econ-help.mit.edu/
- **Carl Anderson (E52-311)** econ-support@mit.edu
- **Slack:** Join mit-econ.slack.com and message **#econ-support**
- **Overview of Econ clusters:** https://econ-help.mit.edu/kb/rc-systems
- **Unix Commands:**
  - https://econ-help.mit.edu/kb/useful-unix-commands
  - `man <command>`
- **Available Modules:** https://econ-help.mit.edu/kb/env-modules
- **Sloan Engaging:** https://wikis.mit.edu/confluence/display/sloanrc/Engaging+Platform

Thank You!

Please email feedback to ssteffen@mit.edu.

Workshop Outline

Cluster Etiquette

- Be a **Good Netizen**:
  - DO cancel stale jobs: scancel ¡JOB_ID¿
  - DO set reasonable cpu, time parameters.
  - DON'T run multiple interactive sessions at once.
  - DON'T submit too many batch jobs at once (try job arrays).
  - Up-Arrow to recall history of commands, Tab to Autocomplete.
  - Only install things for yourself (i.e. `pip install --user pandas`).
  - Don't clutter random (i.e. root) paths - double-check where you save your stuff.
  - Don't try to run stuff in the reducedform login node - it's only able to handle redirecting your programs to the (very powerful) cluster. There are special simplified commands.

Your Turn I: Bash Exercise

- Create a file inside a new folder and write in it.
- Move the file to your home directory.
- Create a renamed copy of the file.
- Show files sorted by modification time.
- Show file contents.

Solution I: Bash Exercise

- `mkdir SBTC_project`
- `cd SBTC_project (or cd !$)`
- `touch import_ad2013.py`
- `vi import_ad2013.py`
- `I (insert mode), "Nice.", <ESC> (normal mode), :x! (save & quit)`
- Better: `cat >import_ad2013.py "Nice." <Ctrl+D> (save & quit)`
- `mv import_ad2013.py ~/import_ad2013.py`
- `cp import_ad2013.py import_ss2019.py`
- `cd ..`
- `ls -t -l`
- `cat import_ad2013.py import_ss2019.py`

Useful Bash Stuff to know [Optional]

- Bash is one of many different shell programming languages. It was first released in 1989.
- **Set shortcuts:** `alias ..="cd .."`
- `!$` refers to the previous command's argument.
- Google more cool stuff: `screen`, `grep`, operators like |, >, ...

- Instead of directly writing bash commands in the terminal, you can also submit commands via a (bash) script (kind of like a stata .do file).
- Example:
  - Create a file and write: `#!/bin/bash [linebreak] echo $PWD`
  - Make file executable: `chmod u+x <file_name>`
  - Run it: `bash <file_name>`

Upload files to a Cluster: FTP [Optional]

- **FTP:** Protocol to transfer files.
- Again, possible via terminal (and command line), but visual clients are more convenient.
- Cyberduck (Mac)
- MobaXterm (Windows)

- A company sends you an ftp link with username and password to their data. Set up a screen connection (to prevent shutdown due to inactivity):

```
screen -S ftp
wget -m --user=<username> --password=<password> <ftp_addres
```

Interactive Sessions

- For easy/early data exploration that does not require large amount of resources.
- 'Normal' UI for your program of choice: xstata, rstudio, python, matlab, ...
- Windows: just type program in MobaXTerm terminal
- Mac: need to run XQuartz (for X-11, i.e. graphics, forwarding).
- **In X-Quartz:** `ssh -Y <kerberos>@<cluster>.mit.edu <program>`

Change Program Version [Optional]

- **See what other modules are available:** `module avail`
- **See currently loaded modules:** `module list`
- **Load module:** `module load <module>`
- Example I: Load Python 3 instead of default Python 2 (which retires in less than a year!)
  - View 'software collections': `scl -l`
  - `scl enable python33 bash`
- Example II: Load older Stata
  - `module load stata/13`
- **Caveat:** On reducedform nothing is preloaded. Need to specify everything in bash file.

ReducedForm - PBS Job Scheduler

- reducedform is a HPC cluster with a dedicated job scheduler - Portable Batch System (PBS).
- MIT Sloan's engaging uses SLURM, a slightly different job scheduler.
- Job Scheduler automatically schedules jobs and allocates resources efficiently.

- Need to submit bash file with info on what script to run, what modules (i.e. Python, R, CRAN, another file you wrote) are required to run it, as well as job parameters (partition, time, CPUs, messages, ...)

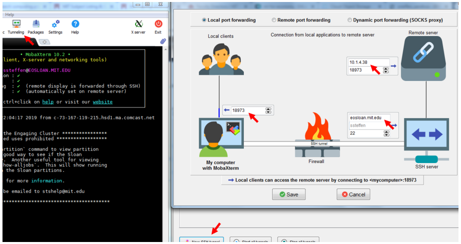
Submitting First ReducedForm 'Batch Job' via bash file [Optional]

- bash file, which specifies all parameters, files, and **modules**: ▸ Example
- **Submit a batch job:** `qsub job_file.sh (sbatch)`
- Check on job: `checkjob`
- Delete a job: `qdel job_id (scancel)`
- Show all jobs: `qstat (squeue)`
- List available modules: `module list`, `module avail`, `scl -l`
- `scl enable python33 bash`
- Break out of execution: `Command + D`
- easy (interactive) alternative for stata, matlab: `statajob`, `matlabjob`

Install Anaconda for Python Virtual Environments [Optional]

- `curl -O https://repo.continuum.io/archive/Anaconda3-5.0.1-Linux-x8`
- `bash Anaconda3-5.0.1-Linux-x86_64.sh`
- `source ~/.bashrc`
- Jupyter Notebooks can be run via tunneling on Engaging.

Tunneling [Optional]



```
        • MobaXterm 10.2 •
lient, X-server and networking tools)

ssteffen@EOSLOAN.MIT.EDU
on : ✓
   : ✓
g  : ✓  (remote display is forwarded through SSH)
   : ✓  (automatically set on remote server)

ctrl+click on help or visit our website

2:04:17 2019 from c-73-167-119-215.hsd1.ma.comcast.net

the Engaging Cluster ****************
ed uses prohibited *****************

rtition` command to view partition
good way to see if the Sloan
.  Another useful tool for viewing
how-alljobs`.  This will show running
 the Sloan partitions.

 for more information.

be emailed to stshelp@mit.edu

************************************
```

[✓ Local port forwarding] Remote port forwarding Dynamic port forwarding (SOCKS proxy)

Connection from local applications to remote server

10.1.4.38
18973

18973

eosloan.mit.edu
ssteffen
22

→ Local clients can access the remote server by connecting to <mycomputer>:18973

1. Select **SSH session**
2. Enter **Host address** (i.e. supply.mit.edu) and **username** (your kerberos)
3. ...
4. Profit!

Loose Ends I: Storage, Git, Access control [Optional]

- Dropbox was founded in 2007 by MIT students.
- $\Rightarrow$ Unlimited Dropbox storage for all MIT students! Link

- Quick and easy git guide: Link
- Can download 'Github Desktop' GUI to avoid terminal commands: Link

- To collaborate on cluster: make groups, assign them folders, and change access rights: Link
- Make a group: `groupadd <group>`
- Add collaborators to groups: `usermod -a -G <group> <user>`
- Change a folder's owner to group: `chgrp -hR <group> <folder>`
- Give read, write, execute rights to group members: `chmod 770`

Unsolicited, Biased Opinions - Languages

- Stata:
  - seductively simple and powerful.
  - not a real programming language.
  - has no/slow development for modern methods and awful graphics.
- R
  - excellent hybrid between real programming language, economics, data science.
  - okay learning curve
- Python
  - real programming language, can do anything.
  - okay learning curve
- Matlab
  - good at optimization.
  - only used a little bit in Econ and Engineering.

PBS Job [Optional]

```
#!/bin/bash
#PBS -N MyJob
#PBS -l mem=12G,walltime=2:00:00,nodes=1:ppn=1
#PBS -m be
#PBS -M <email>

python <script.py>
```

See more here: Link

back

SLURM Parallel Job [Optional]

```
#!/bin/bash
#SBATCH -a 2016-2018 #SBATCH --cpus-per-task=2 #SBATCH --mem-p
#SBATCH --time=2-00:00:00 #SBATCH --partition=sched_mit_sloan_l
#SBATCH --mail-type=BEGIN,END,FAIL #SBATCH --mail-user=ssteffe
python <script.py> $SLURM_ARRAY_TASK_ID
```

▸ back