ARC·TS ADVANCED RESEARCH COMPUTING TECHNOLOGY SERVICES
UNIVERSITY OF MICHIGAN

Great Lakes Cheat Sheet

Guide to general Linux (Bash) and Slurm commands

Accessing Great Lakes

Logging in from a terminal (Duo required)

`ssh uniqname@greatlakes.arc-ts.umich.edu`

Transferring files between Great Lakes and your system

`scp input uniqname@greatlakes-xfer.arc-ts.umich.edu:output`
`scp -r input uniqname@greatlakes-xfer.arc-ts.umich.edu:output`
`scp uniqname@greatlakes-xfer.arc-ts.umich.edu:input output`

GUI Clients

PuTTY	SSH client for Windows
WinSCP	SCP client for Windows
FileZilla	FTP client for Windows, Mac, and Linux

Basic Linux file management

`man command`	Display the manual page for *command*
`pwd`	Print out the present working directory
`ls`	List the files in the current directory
`ls -lh`	Show long, human-readable listing
`ls dir`	List files inside directory *dir*
`rm file`	Delete *file*
`mkdir dir`	Create empty directory called *dir*
`rmdir dir`	Remove empty directory *dir*
`rm -r dir`	Remove directory *dir* and **all contents**
`cd dir`	Change working directory to *dir*
`cd ..`	Change working directory to parent
`cd`	Change working directory to home
`ls`	List the files in the current directory
`cp file1 file2`	Copy *file1* as *file2*
`cp file1 dir`	Copy *file1* into directory *dir*
`mv file1 file2`	Rename *file1* as *file2*
`mv file1 dir`	Move *file1* into directory *dir*
`~ (tilde)`	Home directory
`. (period)`	Current (working) directory
`.. (2 periods)`	Parent directory
`wget URL`	Download a file from Internet *URL*
`unzip file.zip`	Extract a ZIP file
`tar xzf file`	Extract a gzip compressed tarball (common extensions: `.tar.gz` and `.tgz`)

Viewing and editing text files

`cat file`	Print entire content of *file*
`less file`	Prints content of file page by page
`head file`	Print first 10 lines of *file*
`tail file`	Print last 10 lines of *file*
`nano`	Simple, easy to use text editor
`vim`	Minimalist yet powerful text editor
`emacs`	Extensible and customizable text editor

Advanced file management

`chmod`	Change read/write/execute permissions
`which cmd`	List the full file path of a command
`whereis cmd`	List all related file paths (binary, source, manual, etc.) of a command
`du dir`	List size of directory and its subdirectories
`find`	Find file in a directory

Aliases and system variables

`alias`	Create shortcut to command
`env`	Lists all environment variables
`export var=val`	Create environment variable $*var* with value *val*
`echo $var`	Print the value of variable $*var*
`.bashrc`	File that defines user aliases and variables

Input and output redirection

`$(command)`	Runs *command* first, then inserts output to the rest of the overall command
`<`	Standard input redirection
`>`	Standard output redirection
`2>`	Standard error redirection
`2>&1`	Standard error to standard output redirection
`cmd1 | cmd2`	Pipe the output of *cmd1* to *cmd2*

Filters

`wc`	Word, line, and character count
`grep`	Find and print text matching a regular expression
`sort`	Sort input
`uniq`	Filter duplicate lines
`cut`	Cut specific fields or columns
`sed`	Stream editor for search and replace
`awk`	Extensive tool for complex filtering tasks

Great Lakes directories

`/home/uniqname`	For use with running jobs, 80 GB quota
`/tmp`	Small file reads/writes, deleted after 10 days
`/scratch`	Large file reads/writes, purged periodically
`/afs`	Only on login node, 10 GB backed up

Lmod

`module keyword string`	Search for module names or descriptions matching *string*
`module spider string`	Search for modules matching *string*
`module avail`	Show modules that can be loaded now
`module load module`	Load *module* in the environment
`module show module`	Show the help and variables set by *module*
`module list`	List currently loaded modules
`module unload module`	Remove *module* from environment
`module purge`	Remove all modules from environment
`module save collection`	Save all currently loaded modules to *collection*
`module savelist`	Return all saved module collections
`module describe collection`	Return all modules in *collection*
`module restore collection`	Restore all modules from *collection*

Slurm

`sbatch filename`	Submit a job script *filename*
`squeue -u uniqname`	Show job queue for user *uniqname*
`scancel jobid`	Delete job *jobid*
`scontrol hold jobid`	Hold job *jobid*
`scontrol release jobid`	Release job *jobid*
`sinfo`	Cluster status
`srun`	Launch parallel job step
`sacct`	Display job accounting information

Slurm Environment Variables

`SLURM_JOBID`	Job ID
`SLURM_SUBMIT_DIR`	Job submission directory
`SLURM_SUBMIT_HOST`	Host from which job was submitted
`SLURM_JOB_NODELIST`	Node names allocated to job
`SLURM_ARRAY_TASK_ID`	Task ID within job array
`SLURM_JOB_PARTITION`	Job partition

#SBATCH directives and #PBS counterparts

#SBATCH	#PBS	Description
`--job-name=name`	`-N name`	Job name
`--account=name`	`-A name`	Account to charge
`--partition=name`	`-q name`	Submit to partition: standard, gpu. viz, largemem, oncampus, debug
`--time=dd-hh:mm:ss`	`-l walltime=time`	Time limit (walltime)
`--nodes=count`	`-l nodes=count`	Number of nodes
`--tasks-per-node=count`	`-l ppn=count`	Processes per node
`--cpus-per-task=count`	*n/a*	CPU cores per process
`--mem=count`	`-l mem=count`	RAM per node (e.g. 1000M, 1G)
`--mem-per-cpu=count`	`-l pmem=count`	RAM per CPU core
`--gres=gpu:count`	`-l gpus=count`	GPUs per node
`--nodelist=nodes`	`-l nodes=nodes`	Request nodes
`--array=arrayspec`	`-t arrayspec`	Define job array
`--output=%x-%j.log`	`-o filepath`	Standard output in run directory, formatted: jobName-jobID.log
`--error=%x-%j-E.log`	`-e filepath`	Standard error log
`--export=ALL`	`-V`	Copy environment
`--export=var=val`	`-v var=val`	Copy env variable
`--depend=var:jobid`	`-W depend=var:jobid`	Job dependency states (*var*): after, afterok, afterany, afternotok
`--mail-user=email`	`-M email`	Email for job alerts
`--mail-type=type`	`-m type`	Email alert types: BEGIN, END, NONE, FAIL, REQUEUE
`--exclude=nodes`	*n/a*	Nodes to avoid

ARC-TS custom commands

`my_usage`	Usage in CPU minutes
`my_accounts`	Show account membership and resource limits
`home-quota`	Show home quota and usage per user
`scratch-quota account_root`	Show scratch quota and usage per account
`maxwalltime`	Show walltime available for jobs (including upcoming maintenance)

ARC-TS Documentation & Support

GL User Guide: *https://arc-ts.umich.edu/greatlakes/user-guide*

OnDemand/remote desktop: *https://greatlakes.arc-ts.umich.edu*

Email *hpc-support@umich.edu* for further Great Lakes support

Sensitive data should **not** be stored or processed on Great Lakes