NSM CFD Boot Camp
User Manual

Version 1.0

March 2022

National PARAM Supercomputing Facility

National PARAM Supercomputing Facility

Centre for Development of Advanced Computing, Pune

`

©Copyright Notice

®Trademark

Getting Help

For technical assistance, please send an email to npsfhelp@cdac.in

DISCLAIMER

The information contained in this document is subject to change without notice.

C-D AC shall not be liable for errors contained herein or for incidental or consequential damages in connection with the performance or use of this manual.

Contents

Contents

1. Prerequisites

The primary prerequisites of the PARAM Siddhi-AI system include basic knowledge of the Linux environment. The expected secondary prerequisites are listed below:

- Working knowledge of terminal/shell/command-line interface
- Basic knowledge of file and directory handling Linux commands
  Ex. ls, cd, mkdir, cp, mv, rm, chmod, tar, gzip etc.
- Use of the text editors
  Ex. GNU nano, vi, vim, emacs
- Program editor
  Jupyter notebook
- Knowledge of file transfer tools
  Ex. scp, sftp
- Knowledge of Enroot
- Knowledge of Slurm
- Knowledge about
  JupyterNotebook

Jupyter Notebook
https://jupyter.org/

2. Intended Audience

This document is meant for NSM India CFD GPU Boot camp participants. The document gives step-by-step instruction on

- How to login to Param Siddhi-AI.
- Copy & copy back files to/from remote system.
- Access Jupyter note book through browser of your local system.

Note: 1. Readers are assumed to have familiarity with basic Linux commands.

2. In the document "**your system", "local system"** means the computer system that participants are using to access NSM India CFD Boot Camp

3. Whereas **"remote system", "remote compute resource", "compute resource", "remote server", "GPU compute resource"** means the PARAM Siddhi-AI System

3. PARAM Siddhi-AI System Accessing Mode

3.1 Access Mode

ssh based access has been provided for accessing the PARAM Siddhi-AI system, which allows users to log in to the system through a two-stage process :

Step 1: Connect to C-DAC VPN

Step 2: Log in to PARAM Siddhi-AI System

3.1.1 Connect to C-DAC VPN

Users are required to install a VPN Client (FortiClient) on their local laptop/computer to connect to the PARAM Siddhi-AI system. Therefore, it is advisable to refer to the FortiClient installation cum user manuals for appropriate installation instructions. Upon successful installation of the VPN client (FortiClient), users may connect to the VPN as per the below steps:

A) Start the FortiClient application and then enter “User ID” and “password” provided at the time of account creation through an e-mail sent by npsfhelp@cdac.in

Figure : FortiClient Login Screen

B) Click on Connect button. After successfully establishing the VPN connection, you will see the below screen.

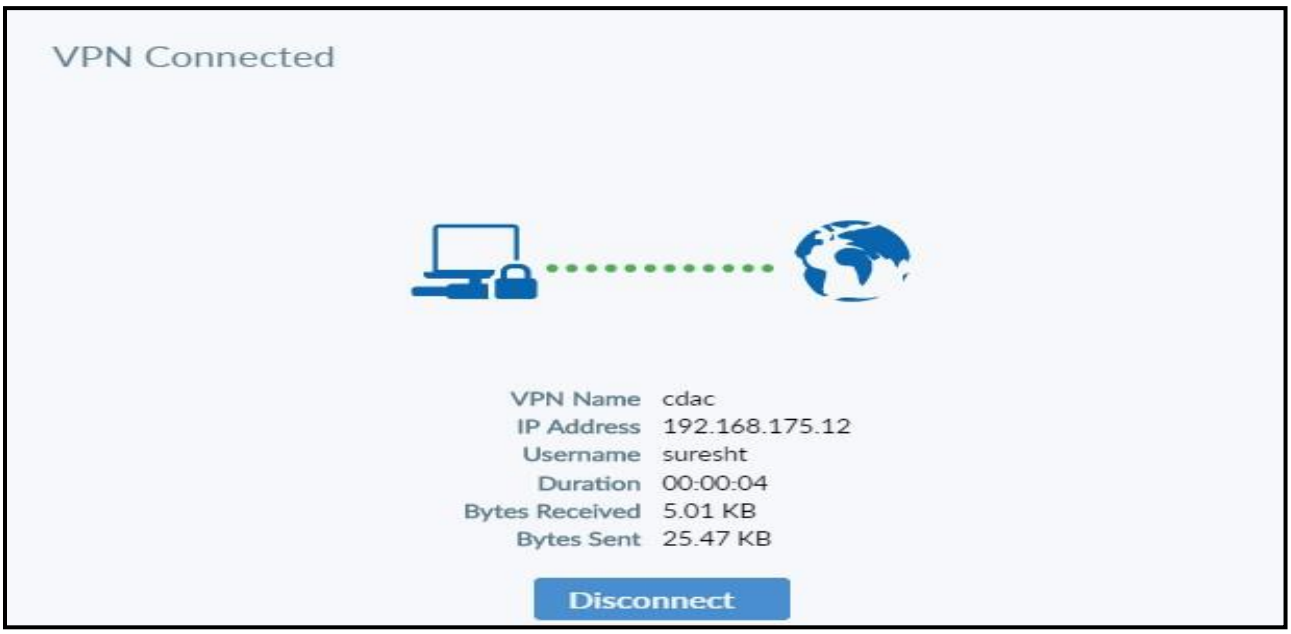


Figure : FortiClient Successful connection screen

Note: Step 2 must be followed only after successfully establishing the VPN connection.

3.1.2 Login to PARAM Siddhi-AI System

The login node is the primary gateway to the rest of the cluster, including a job scheduler (Slurm). You may submit jobs to the queue, which will run when the required resources are available. **It is advisable to do not run programs directly on the login node**. Instead, the login node is used to submit jobs, transfer data, and compile source code. (If your compilation takes more than a few minutes, you should submit the compilation job into the queue to be run on the cluster.) Secure Shell-based access to PARAM Siddhi- AI login/compile node can be accessed from Windows and Linux machines as per the steps given below :

A) Accessing from Windows machine

It is recommended to use MobaXterm or Putty software tool as an SSH client on a Windows machine to log in to PARAM Siddhi-AI System. Free Home Edition of the same can be downloaded from the following URLs:

https://mobaxterm.mobatek.net/download.html

or

https://the.earth.li/~sgtatham/putty/latest/w64/putty.exe

- **Access through MobaXterm**

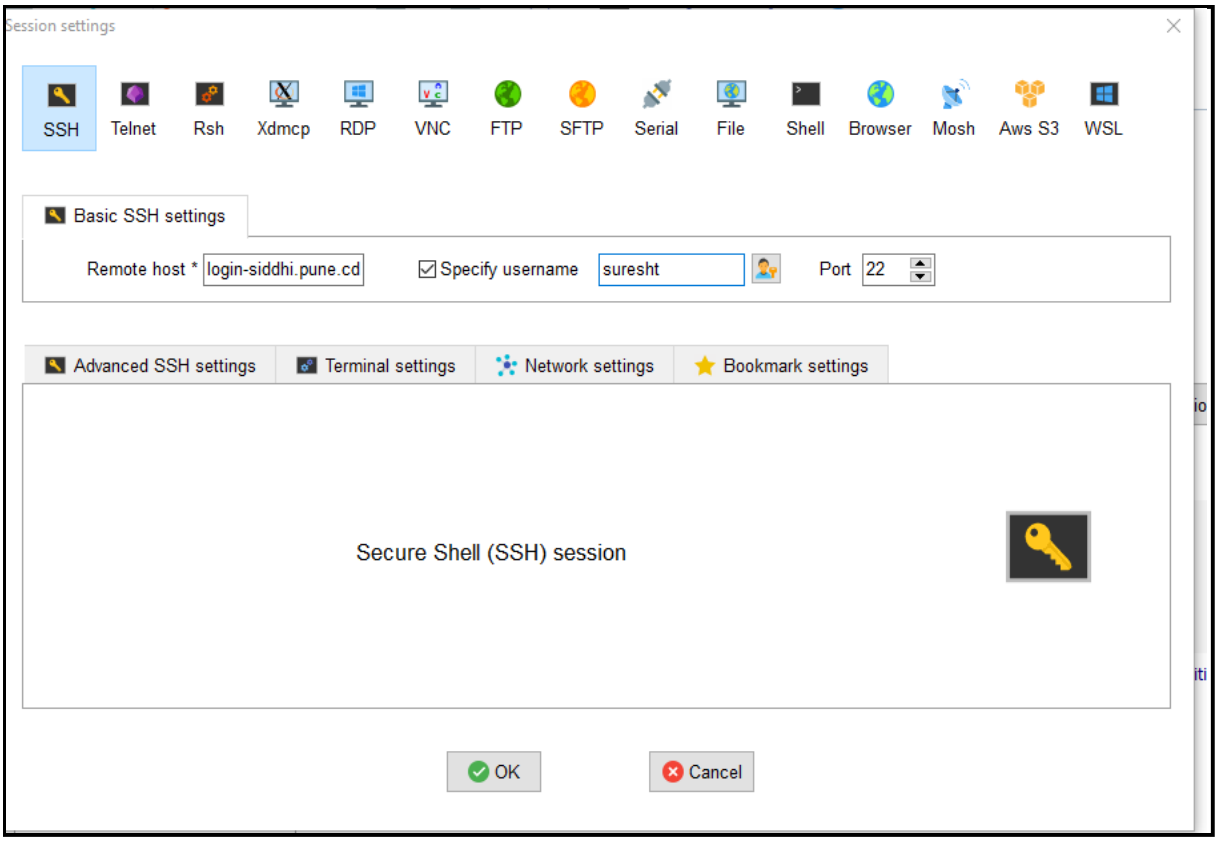


Figure 5.3 : Access through MobaXterm

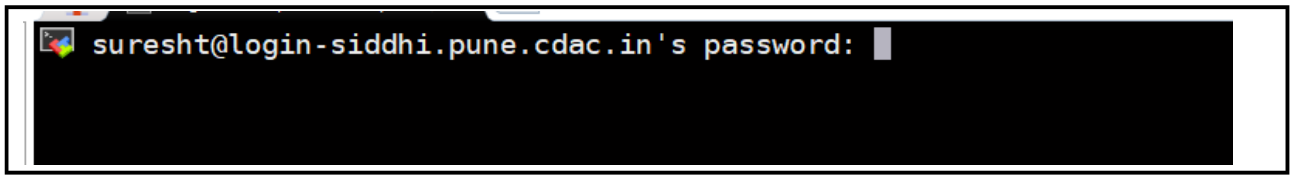


Figure: Login screen in MobaXterm

After entering the user's login password, you will get the shell of the login node

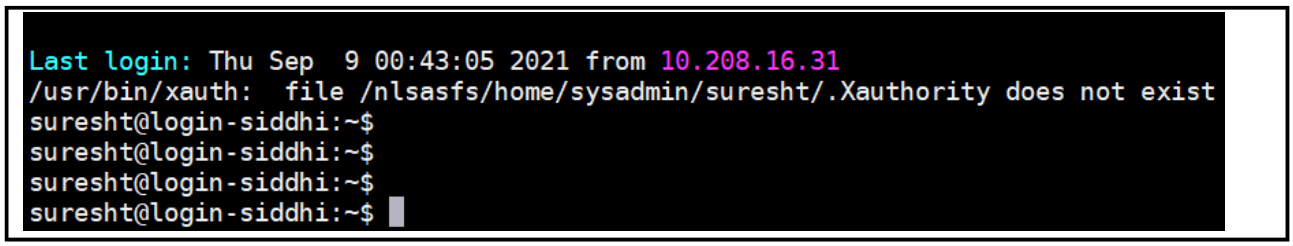


Figure: Terminal of login node in MobaXterm

- **Access through Putty**

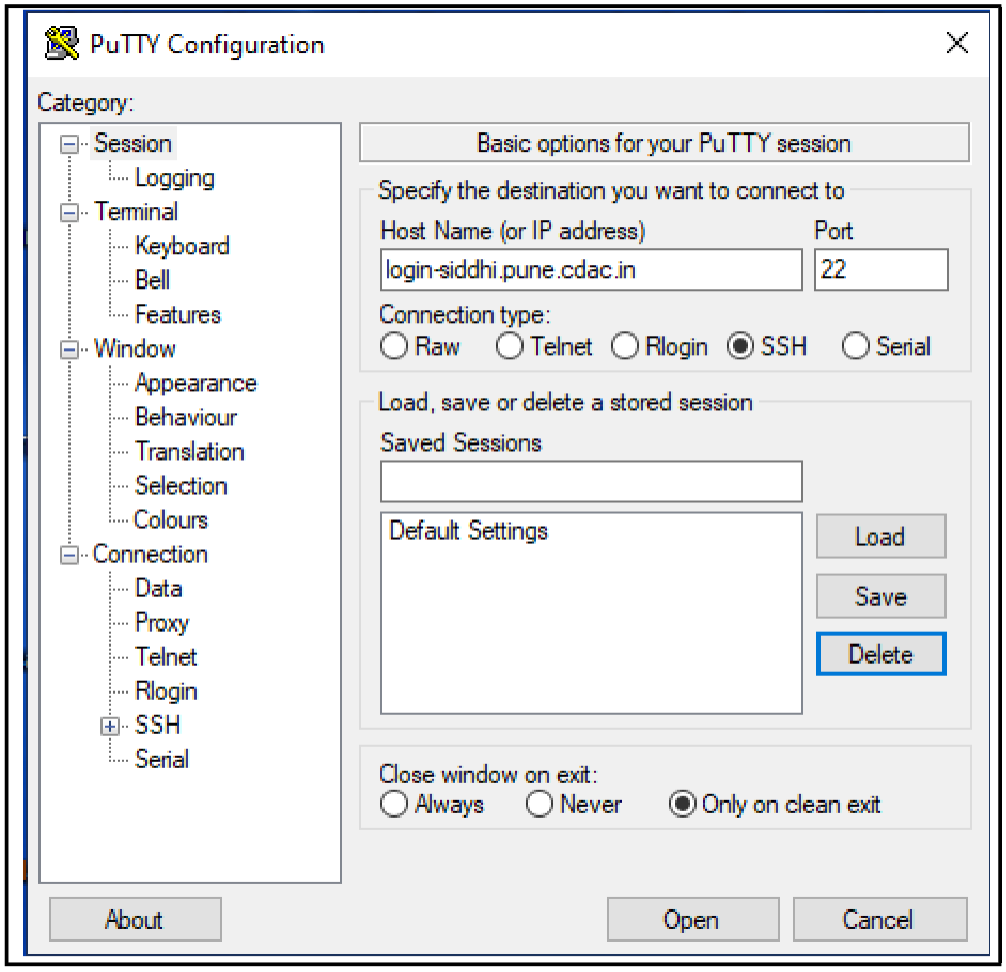


Figure : Access through putty

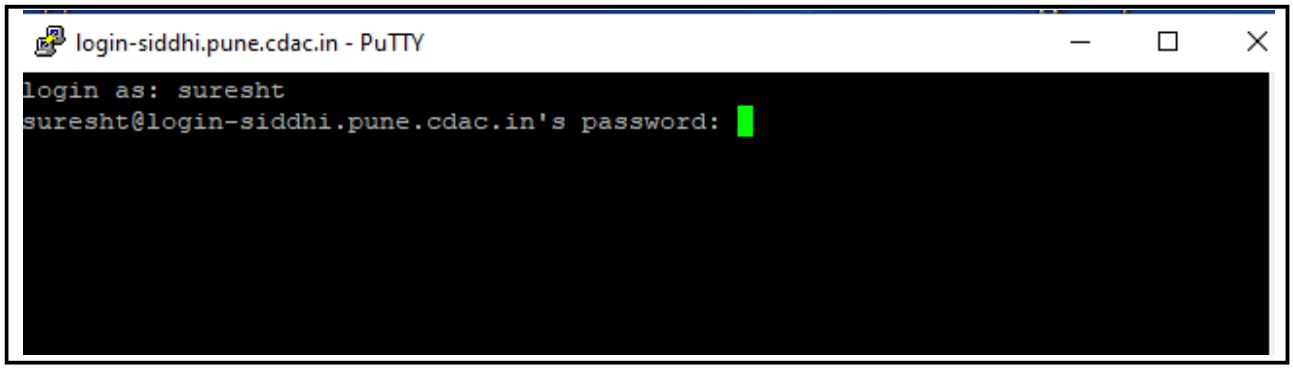


Figure: Login screen in Putty

After entering the user's login password, you will get the shell of the login node

```
Last login: Sat Oct  9 18:02:34 2021 from 10.208.16.31
suresht@login-siddhi:~$
suresht@login-siddhi:~$
```

B) Accessing from Linux machine:

Users can login to PARAM Siddhi-AI system using ssh command from their linux base local laptop/desktop by following the below steps :

```
$ ssh login-siddhi.pune.cdac.in -X -l <user name>
```

Example :

```
$ ssh login-siddhi.pune.cdac.in -X -l suresht
```

```
suresht@sysadm:~$ ssh login-siddhi.pune.cdac.in -X -l suresht
suresht@login-siddhi.pune.cdac.in's password:
```

Once the command is executed, you will be prompted for password. Enter the password. After entering the user password, you will get the shell of the login node as shown below.

```
Last login: Sun Sep 12 10:29:20 2021 from 192.168.175.5
suresht@login-siddhi:~$
suresht@login-siddhi:~$
suresht@login-siddhi:~$
suresht@login-siddhi:~$
suresht@login-siddhi:~$
```

3.2 Password Change

Use the command "passwd" to change your password and follow password criteria while changing the password

Syntax: passwd <user name>

Password setting criteria:

Password should be of a minimum of eight (8) characters in length. In addition, the password should contain at least two upper case alphabets, two lower case alphabets, two numerical, and two special characters. Password will be valid for six months, and one cannot re-use/repeat the last three(previous)passwords.

3.3 Data Transfer between local machine and PARAM Siddhi-AI System

Users need to have the data and application related to their project/research work on the PARAM Siddhi-AI system. To store the data, directories have been made available to the users with the name " home" the path to this directory is "/home". Whereas these directories are common to all the users, a user will get his directory with their username in /home/ directories where they can store the data.

However, there is a limit to the storage provided to the users, which is defined according to quota over these directories. All users will be allotted the same quota by default. When users wish to transfer data from their local system (laptop/desktop) to the PARAM Siddhi-AI system, they can use various methods and tools from Windows and Linux machines.

A) Windows Machine

- **Through MobaXterm**

Users can copy small files from/to their local machine and PARAM Siddhi-AI system using the MobaXterm tool by following the below steps :

i) Upload files from your local system (laptop/desktop) to the PARAM Siddhi-AI Cluster, click on the upload button as shown in the below image

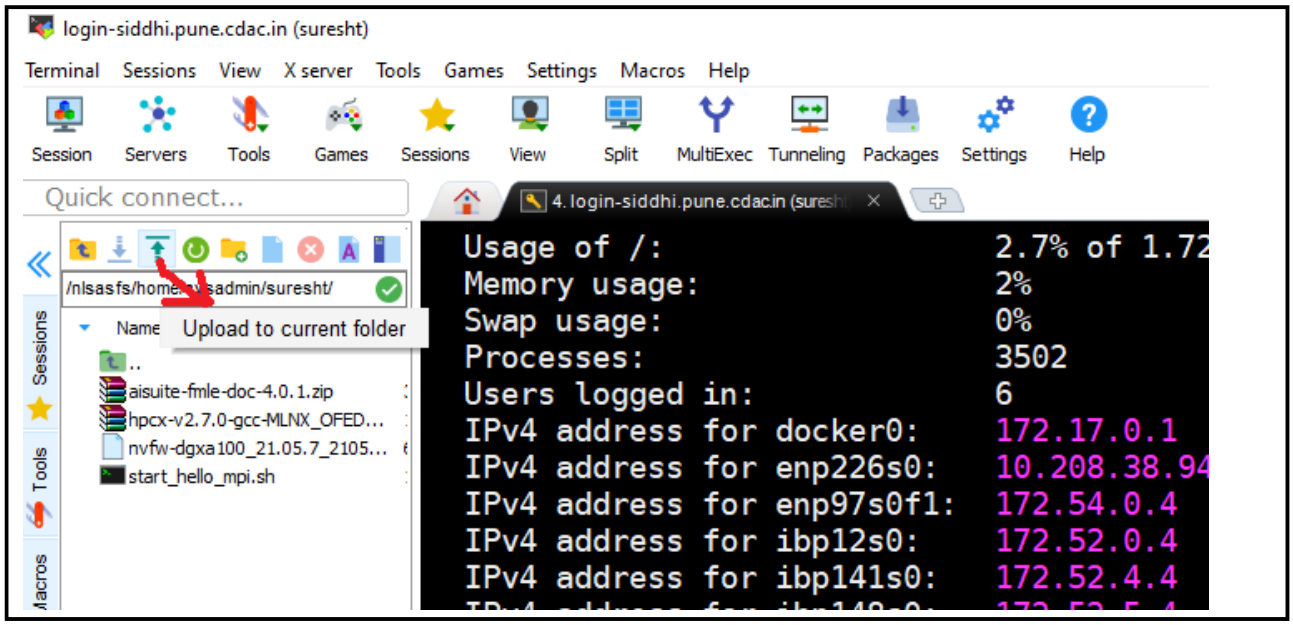


Figure: Data upload through MobaXterm

Example :

See the below image to upload DIRAC-19.0-Source.tar file from the local system to PARAM Siddhi-AI Cluster.

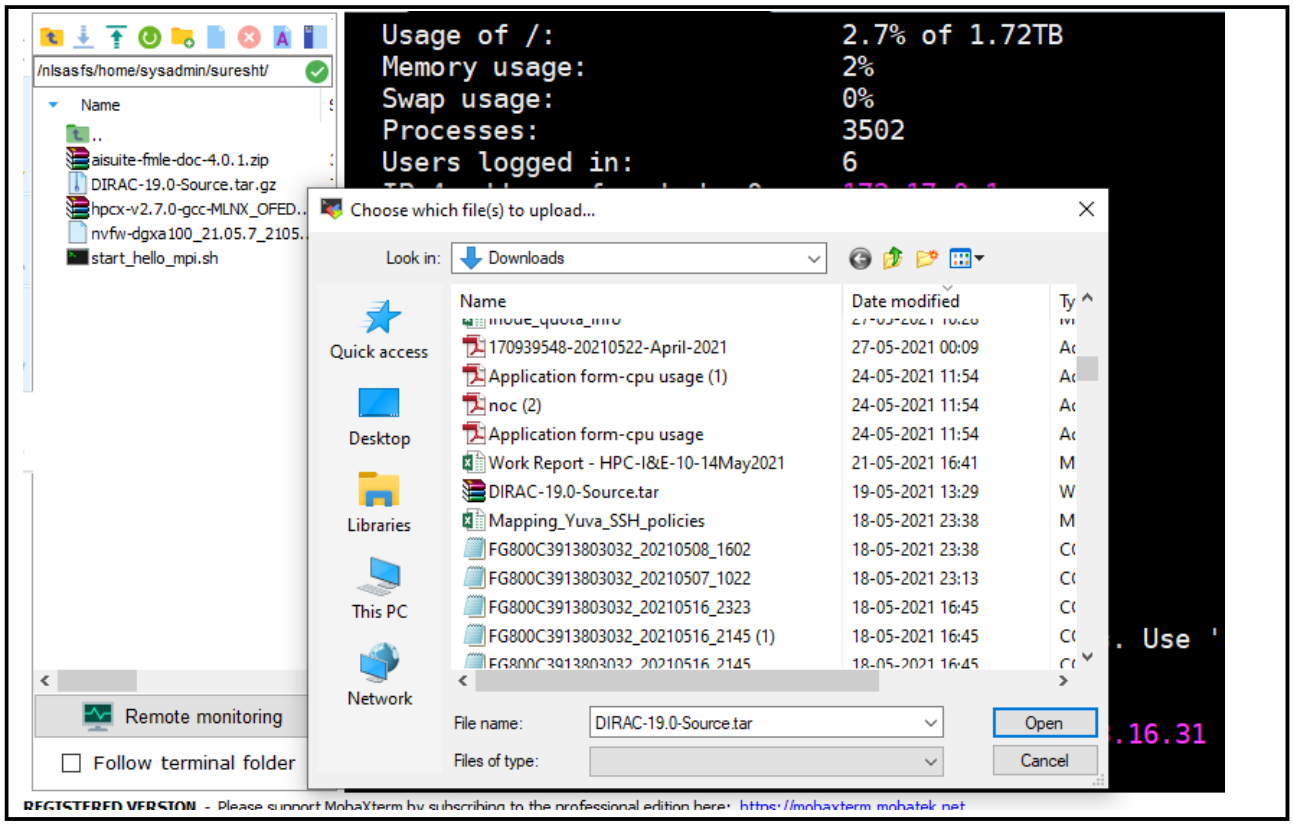
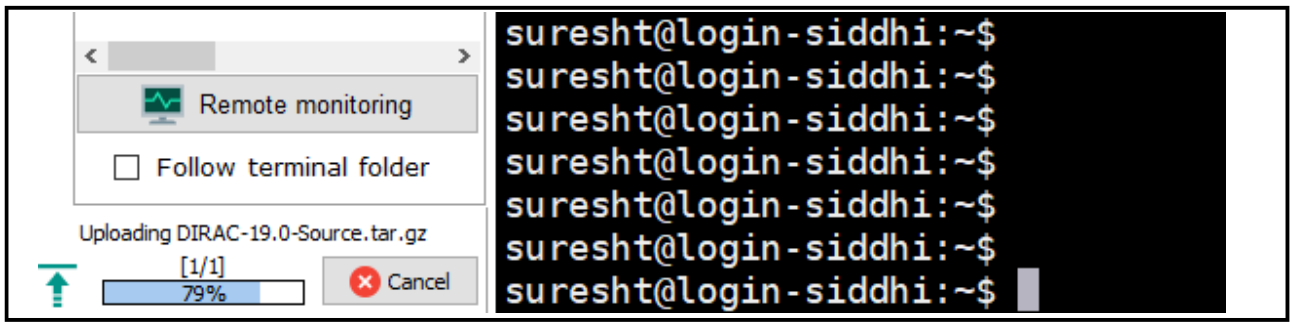


Figure : Data browse through MobaXTerm



ii) To download files from the PARAM Siddhi-AI system to your local system, follow the steps shown in below image:

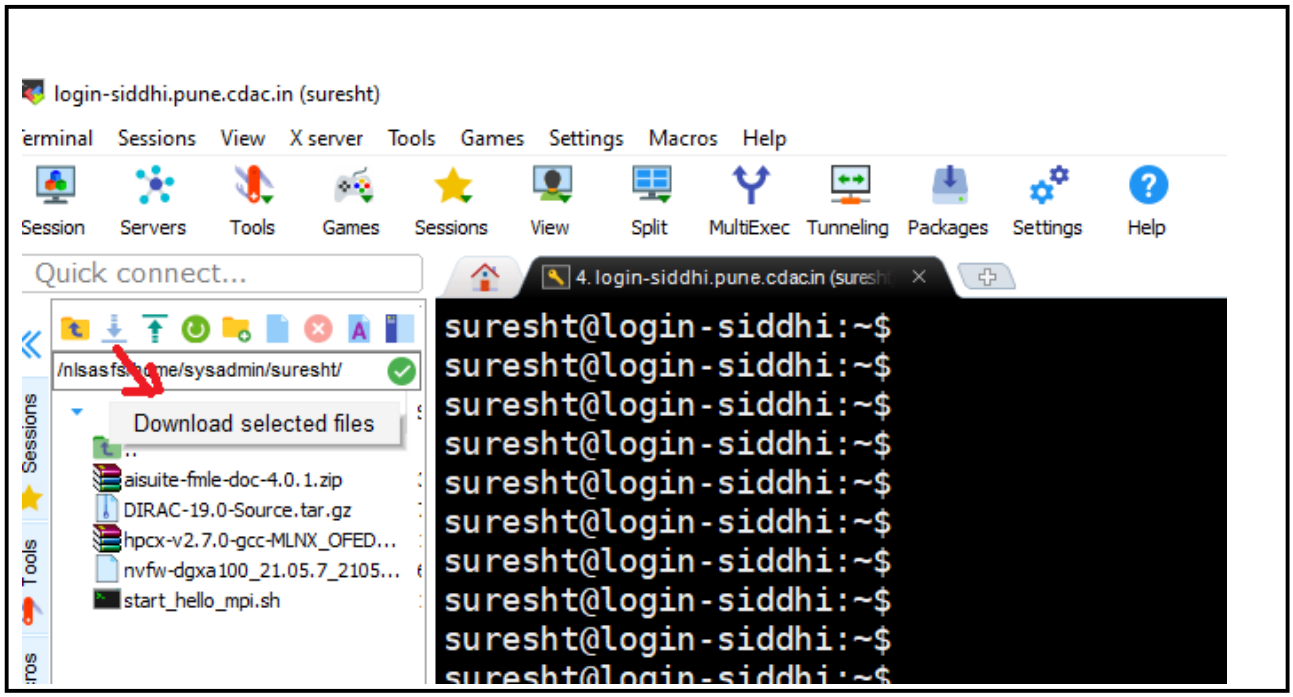


Figure: data download through MobaXTerm

Example :

Select the file “DIRAC-19.0-Source.tar” file from the left panel and click on the download selected files. It will get downloaded on your local system.

- **Through WinSCP**

Users can copy small files from/to their local machine and PARAM Siddhi-AI system using the WinSCP tool by following the below steps :

WinSCP can be downloaded from the URL: https://winscp.net/eng/download.php

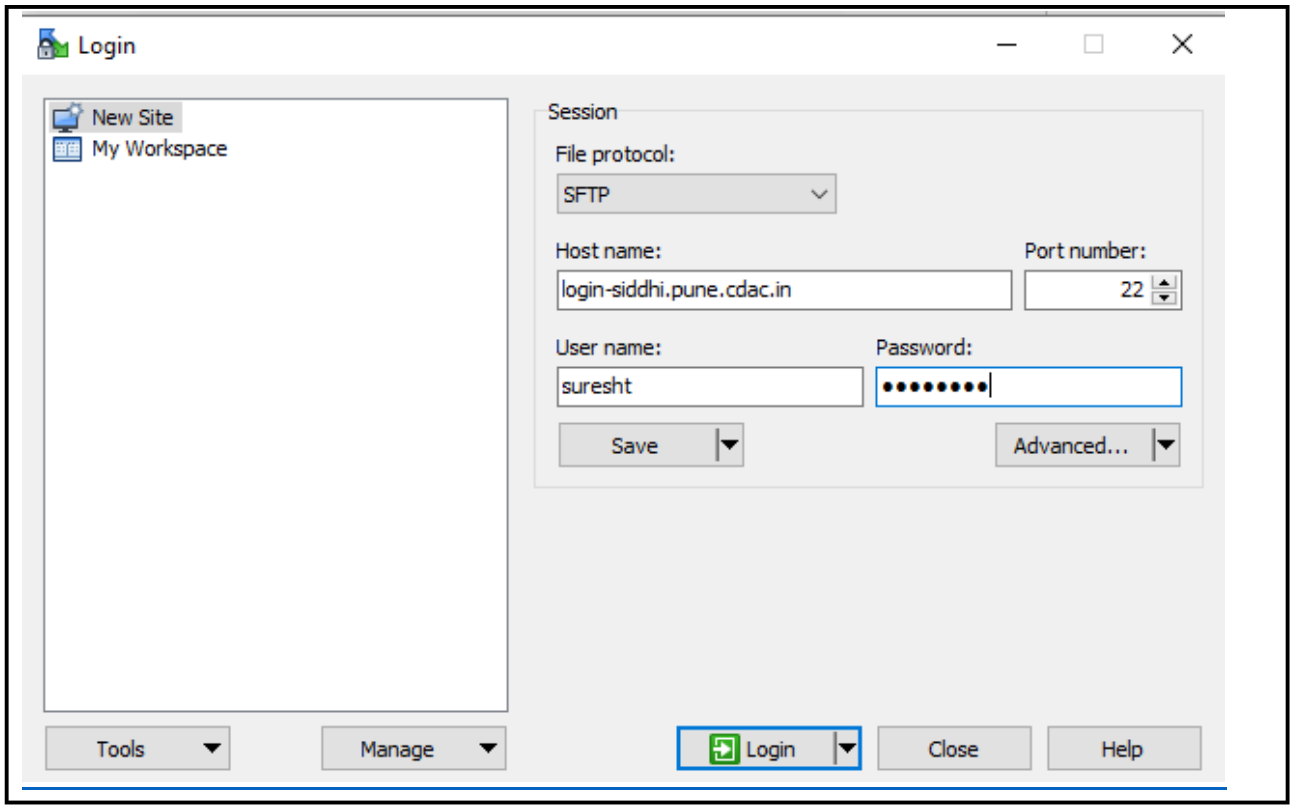


Figure : Data transfer through WinSCP

Note: replace with appropriate <user name> <password>that is provided to you.

ii) After establishing the connection, To upload files from your local system to the PARAM Siddhi-AI system and To download files from the remote system to your local system, drag and drop the files in the respective location.

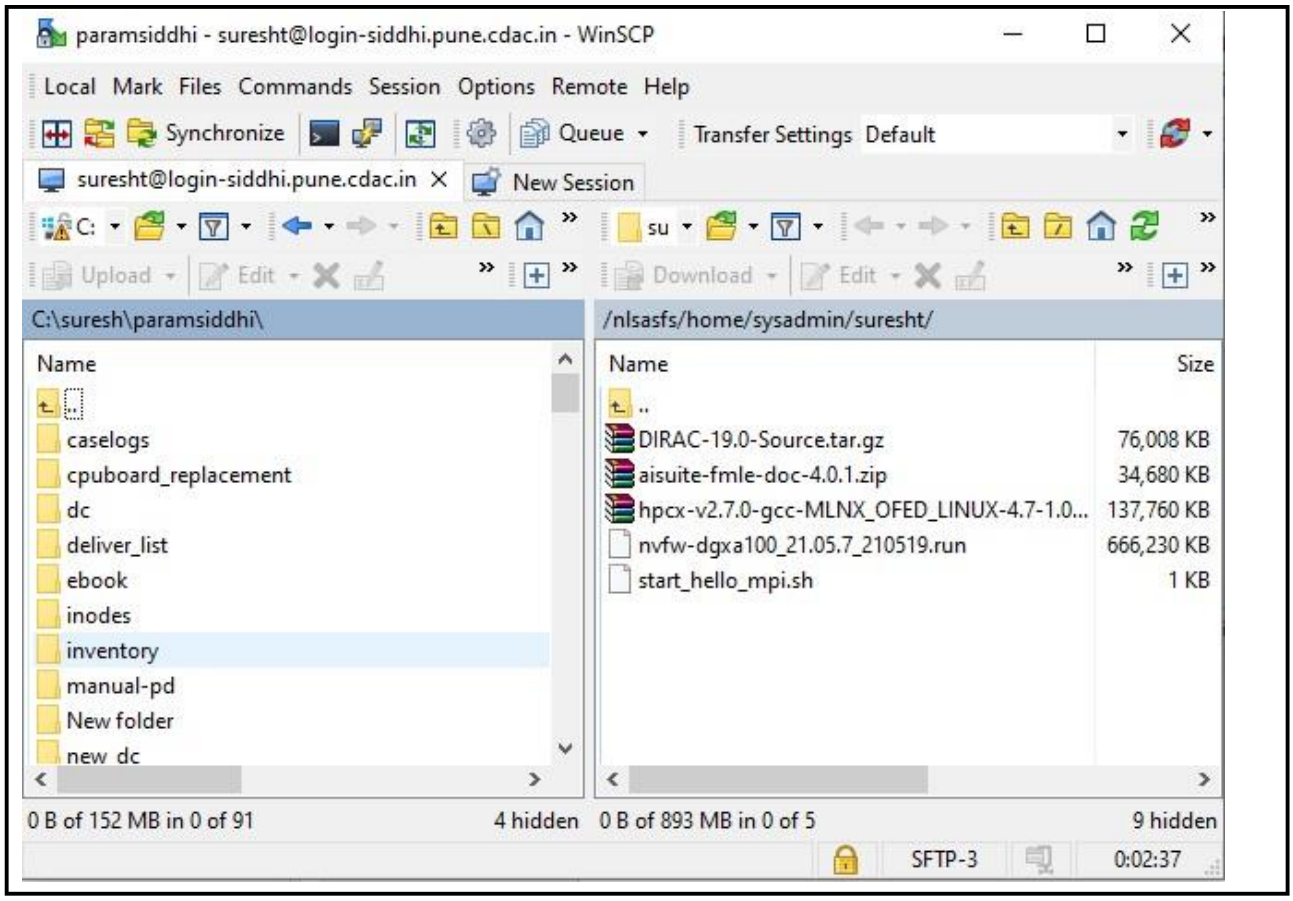


Figure : Upload/download from the remote system to your local system using WinSCP.

B) Linux Machine

Use the below commands to copy data/file(s) to/from your local Linux system and PARAM Siddhi-AI system.

i) Upload

```
$ scp filename.txt <user name>@<remote-system-ip>:<path-to-destination-directory>
```

Example :

```
$ scp myfile.txt suresht@login-siddhi.pune.cdac.in:~/
```

```
suresht@sysadm:~$ scp -r zxp.pdf suresht@login-siddhi.pune.cdac.in:~suresht/
suresht@login-siddhi.pune.cdac.in's password:
zxp.pdf                                       100%   18MB   9.0MB/s   00:02
suresht@sysadm:~$
```

Figure : Upload from local Linux system to PARAM Siddhi-AI system using command line

ii) Download

```
$ scp <user name>@<remote-system-ip>:<path-of-the-files-to-be-downloaded> .
```

Example :

```
$ scp suresht@login-siddhi.pune.cdac.in:~suresht/test.txt .
```

```
suresht@sysadm:~$ scp -r suresht@login-siddhi.pune.cdac.in:~suresht/zxp.pdf .
suresht@login-siddhi.pune.cdac.in's password:
zxp.pdf                                       100%   18MB   9.0MB/s   00:02
suresht@sysadm:~$
```

Figure: Download from PARAM Siddhi-AI to local Linux system using command line.

Note: replace with appropriate <user name> <password>that is provided to you.

However, the above methods should only be used if someone wants to transfer small files. If you wish to transfer a large file or data set, we recommend you upload the same on the web (maybe on GitHub) and download from there directly on the PARAM Siddhi-AI system by some method like wget, git clone, etc.

4. PARAM Siddhi-AI Computing Environment

This chapter introduces the PARAM Siddhi-AI system's computing environment. It introduces the user to work with the PARAM Siddhi-AI HPC cluster.

4.1 Hardware Configuration:

Server : NVIDIA DGX A100	
Server Specification	CPU : AMD EPYC 7742 64C 2.25GHz CPU Core: 128 cores (Dual Socket each with 64 cores)
	System Memory (RAM) : 1 TB
	GPU : NVIDIA A100 –SXM4 GPU Memory : 40GB Total No. of GPU's per system : 8
	Storage : local Storage 1.8TB /raid : 14 TB
Software Environment	OS : Ubuntu 20.04.2 (Kernel : 5.4.0-80-generic)
	Cuda : 11.0
	NVIDIA NGC Support

NVIDIA DGX-A100 system at NPSF:

4.2 Screen Utility

Screen or GNU Screen works as a terminal multiplexer. It allows users to start a screen session and open any virtual terminals inside that session. The main advantage of the Screen utility is that processes running in Screen will continue to run when their terminal is not visible, even if the user's connection drops and the SSH session is terminated while performing a long-running task on a remote machine. Thus, a process started with Screen, can be detached from the current session and then reattached later. In other words, we can say the session is detached, but the process that was originally started from the Screen is still running and managed by the Screen utility itself.

Follow the below steps to work with Screen utility :

A) Starting Linux Screen

To start a screen session, type screen in your console:

```
$ screen
```

This will open a screen session, create a new window, and start a shell in that window.

B) Starting Named Screen Session

Named screen sessions are useful when you run multiple screen sessions. To create a named session, run the screen command with the following arguments:

```
$ screen -S <session name>
```

Example: screen -S namd

C) Detach from Linux Screen Session

You can detach from the screen session at any time by typing: (press ctrl and a key together; then release both keys and press d)

```
$ ctrl+a d
```

The program running in the screen session will continue to run after you detach from the session

D) Re-join the screen

You can re-join the screen at any time by typing:

```
$ screen -x
```

Below are some most common commands for managing Linux Screen Windows:

Ctrl+a c create a new window (with shell)

Ctrl+a ”List all window

Ctrl+a 0 Switch to window 0 (by number)

Ctrl+a X Close the current region

E) Reattach to a Linux Screen

To resume your screen session, use the following command:

```
$ screen -r
```

In case you have multiple screen sessions running on your machine, you will need to append the screen session ID after the r switch.

F) Working with Linux Screen Windows

When you start a new screen session, it creates a single window with a shell in it. You can have multiple windows inside a screen session.

To create a new window with shell type Ctrl+a c, the first available number from the range 0...9 will be assigned to it.

G) Find the session ID

```
$ screen -ls
```

H) To resume your screen session with session use the following command

```
$ screen -r <session id>
```

‘B’ Support Details

For system support, please join the slack channel and post your queries @ **cluster support**