Notes for accessing Perseus and/or Tigressdata*

* For Ubuntu/Linux users

First time access to Perseus and/or Tigressdata

The first time you try to access Perseus and/or Tigressdata from you local machine you must use a wired connection or a VPN connection.

Information about the VPN installation for **Ubuntu/Linux** can be found at:
https://princeton.service-now.com/kb_view.do?sysparm_article=1157

Note: If you do not follow the next steps, then you will ALWAYS have to use VPN to connect to the cluster.

Setting up ssh keys

To avoid using VPN or wired connection anytime you want to access the cluster, you will need to set up a pair of ssh keys that will be "shared" by your local machine and the remote server. The pair consists of a **private key** (default name: id_rsa) and a **public key** (default name: id_rsa.pub). **You must never share your private key.**
For general info about ssh keys go to: https://wiki.archlinux.org/index.php/SSH_keys

1. Check first if you have already an ssh key in your local folder **~/.ssh**

2. If you do not have any ssh key in your local folder, you must generate a pair. To generate RSA keys, on the command line, enter: ssh-keygen -t rsa
   You will be prompted to supply a filename (for saving the key pair) and a password (for protecting your private key):

- **Filename:** To accept the default filename (and location) for your key pair, press **Enter or Return** without entering a filename. Alternatively, you can enter a filename (e.g., my_ssh_key) at the prompt, and then press Enter or Return. However, many remote hosts are configured to accept private keys with the default filename and path (**~/.ssh/id_rsa for RSA keys**; **~/.ssh/id_dsa for DSA keys**) by default. Consequently, to authenticate with a private key that has a different filename, or one that is not stored in the default location, you must explicitly invoke it either on the SSH command line or in an SSH client configuration file (**~/.ssh/config**); (see https://kb.iu.edu/d/aews)

- **Password:** Enter a password that contains at least five characters, and then press Enter or Return. If you press Enter or Return without entering a password, your private key will be generated without password-protection.

3. If you do have an ssh key in your local folder you can either generate a different pair of keys for the connection to the remote server (e.g. id_rsa_perseus.pub) or you can use the already existing key (e.g. id_rsa.pub). In the latter case, you can skip step 2.

4. After generating the pair of keys, you must add the contents of your **public key** to the **authorized_keys** file in the remote server [if no such file exists in the ~/.ssh folder of the remote server, you must create one (see https://kb.iu.edu/d/aews)] There are several ways of adding the contents of your **public key** to the **authorized_keys**[1], as explained in https://kb.iu.edu/d/aews

5. If you have only one pair of keys on your local machine then you can access the remote server by typing: ssh username@perseus.princeton.edu or ssh username@tigressdata.princeton.edu

   If you **have more than one pair of keys** on your local machine, then you need to specify which key to be used for accessing the cluster. For example, let's assume you have two keys in your local machine (**id_rsa** and **id_rsa_new**) and you have copied the contents of **id_rsa_new.pub** into the ~/.ssh/authorized_keys file on Perseus. Then, you can access the cluster by typing: ssh -i ~/.ssh/id_rsa_new username@perseus.princeton.edu [2,3]

Sometimes one wants to connect via ssh from remote server 1 to remote server 2. To do so, one has to follow the steps 1-5 described above. Remote server 1 plays the role of the local machine.

Notes

1. If you add manually the content of the public key to the file authorized_keys, be careful not to copy paste the content of the key as displayed on the terminal to the file. This may create broken lines and the key will not be recognized! For more details about the authorized_keys file see: https://wiki.mcs.anl.gov/IT/index.php/SSH_Keys:authorized_keys

2. If you want to avoid typing long commands on the terminal you can do one of the following:

- create an **alias** to your .bashrc file, e.g., alias sshpers='ssh -i $HOME/.ssh/id_rsa_new username@perseus.princeton.edu'
- modify your **ssh_config** file. This can be found in ~/.ssh/config or /etc/ssh/ssh_config folders. This is recommended for your local machine, but usually one cannot modify the ssh_config file of a remote server. For more info check here : https://kb.iu.edu/d/aews

3. If you get the following error:
   ***Permission denied* (*publickey*,*gssapi- keyex*,*gssapi*-with-*mic*,password)**
   you should read the log by typing**:**
   ssh -vvv -i ~/.ssh/id_rsa_new username@perseus.princeton.edu

Useful information can be found at:
https://askubuntu.com/questions/692480/ssh-authentication-with-id-rsa-key-not-working

https://www.experts-exchange.com/questions/27464003/SSH-Pub-Key-Exchange-Failure.html