RADICAL Best Practices to Access Compute Resource

This is a living document in which the radical team collects best practices to use the resources available to each member of the lab. When you discover something relevant, please add it to the document so that everyone in the lab can benefit from the shared knowledge.

SSH keys

Create key pairs. rsa is considered outdated in favour of ed25519 but rsa is guaranteed to work everywhere. So far, we did not find any machine/service that did not accept ed25519. Important: you must use a password to protect your private key. The tools you use to create your private/public key pair will let you enter an empty password/passphrase: resist the temptation!

Linux and MacOS

If you use a linux distribution or MacOS you can use `ssh-keygen` to generate your keys. Here the command to generate two keys, one with the ed25519 algorithm and another with the old rsa. Some of us use only ed25519 and, so far, had no issues.

```
ssh-keygen -t ed25519 -a 100
ssh-keygen -t rsa -b 4096 -o -a 100
```

Copy your public key to a remote machine ‘host’ on which you have already an account:

```
ssh-copy-id -i ~/.ssh/tatu-key-ecdsa user@host
```

On MacOS, you can reduce the amount of times you will have to enter the password for your private key (every time you ssh to a machine that requires that key to log in) by adding the following to your .ssh/config file (create one if you don’t have any):

```
Host *
  AddKeysToAgent yes
  IdentityFile ~/.ssh/id_ed25519
  IdentityFile ~/.ssh/id_rsa
```

On a *nix system (therefore also on MacOS) you can use `ssh-agent` to do the same:

```
ssh-agent -t 3600 ~/.ssh/id_ed25519
ssh-agent -t 3600 ~/.ssh/id_rsa
```

Note that, if you are running `ssh-agent` on your local machine (e.g., laptop), you can forward that agent when ssh-ing into a RADICAL-machine by using:

```
ssh -A <user_name>@<name_machine>
```

You can make this behavior automatic by adding the following to your .ssh/conf:

```
Host name_machine
  ForwardAgent yes
```

Microsoft Windows

If you are using MS Windows you may want to consider using a Linux virtual machine (VM). In general, it will make your life at the RADICAL lab easier. The Windows Subsystem for Linux (WSL) proved to be problematic with ssh and public key authentication. We suggest therefore to use a different hypervisor like, for example, Hyper-V or VirtualBox and create a linux VM to run on one of them. Once you have the VM running, you can follow the notes above.

If you decide to use MS Windows, you can use Putty and Puttygen to establish a ssh connection between your windows machine and a RADICAL machine. At the bottom of Puttygen's window, you can choose the algorithm to use to generate your key pair. rsa is considered outdated in favour of ed25519 but rsa is guaranteed to work everywhere. So far, we did not find any machine/service that did not accept ed25519. In case you choose to use rsa, you will have to enter the 4096 bits length in puttygen. Important: you **must** use a passphrase to protect your private key.

Puttygen keys need to be converted to the openssh format to be used with the RADICAL machines. Send your Puttygen key to one of the admins (Andre, Giannis, Matteo) and they will convert it for you. The command to convert the key is:

```
ssh-keygen -i -f puttygen_key.pub > openssh_key.pub
```

RADICAL Machines

These are server machines that RADICAL makes available to lab members and selected users to use for experiments and other research activities. As they are used as entry points to several high performance computing platforms around the world, we have to be careful about how securely we use them.

Obtaining an account

Contact one of RADICAL admins (Andre, Giannis, Matteo) to create an account. You will have to provide your public ssh key and your preferred login name. The best way to contact the admins is to attach the file with your public key to and email.

Connecting to the machine

The ssh command from a command line interface (i.e., terminal) is the only way to connect to these machines. Examples:

```
ssh <user_name>@<name_machine>
```

MongoDB and RabbitMQ

Currently, RADICAL-Cybertools (RCT) requires a MongoDB and, in case EnTK is used, a RabbitMQ server. We provide a publicly accessible instance of both on each RADICAL machine. You will have to request a MongoDB and RabbitMQ account to a RADICAL admin (Andre, Giannis, Matteo). Send an email with your preferred user name and, for MongoDB, database name.

Data Sanitization

This applies in case of a known or suspected security breach on one of the RADICAL machines on which you have an active account and some data you have to recover before the machine is destroyed. Before moving the data outside the machine, we:

- Delete all compiled binary files.
- Delete all virtual/conda environments.
- Unset the executable bit of all the files.

If your data are on a confirmed compromised machine, ask one of the admins (Andre, Giannis, Matteo) to sanitize your data, indicating where to move them. Otherwise, if your data are on a suspected machine, sanitize your data and transfer them to a new location before destroying the suspected machine.

Use sanitizer.sh to sanitize your data. Warning: this has been tested on test threes. It may still have undiscovered side effects. Place the script in a directory outside the directory three you need to sanitize and the run it with the dirthree to sanitize as an argument:

```
sanitize.sh directory_three_to_sanitize
```

Once the script exits, the directory_three_to_sanitize should be clean.