CÉCI HPC Training

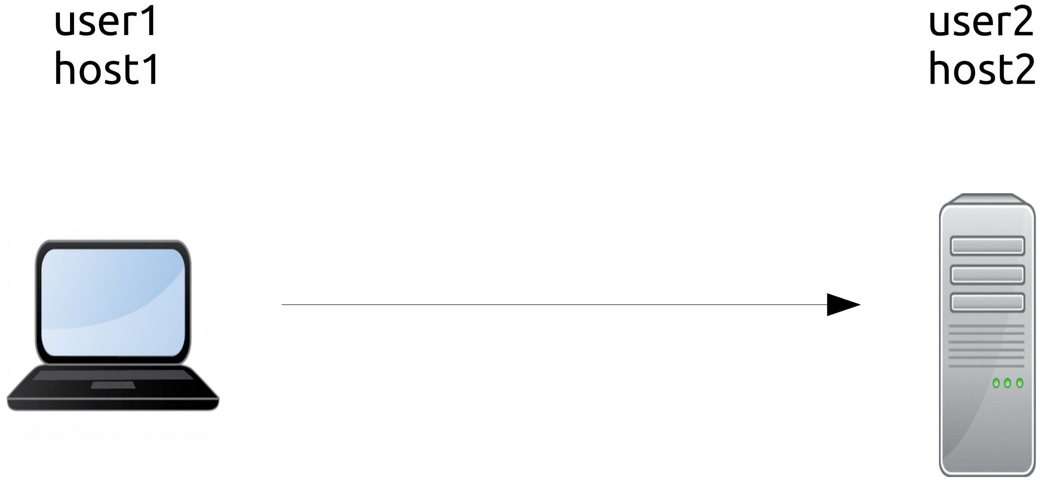
Connecting with SSH from Linux or Mac: Introduction and advanced topics

Juan.Cabrera@unamur.be

INTRODUCTION

SSH Secure Shell

SSH Secure Shell



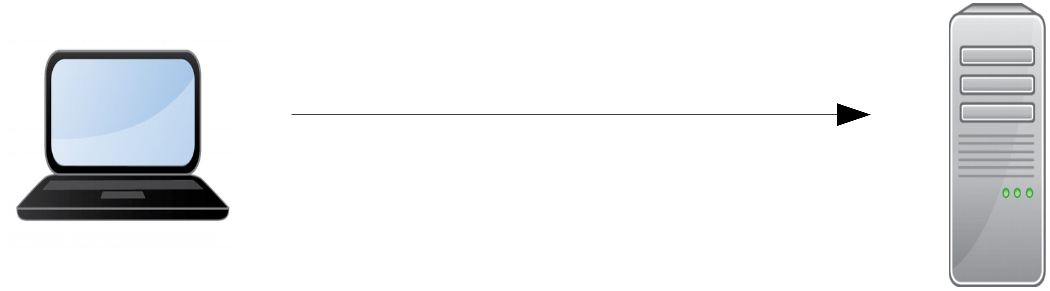
user1
host1

user2
host2

SSH Secure Shell

user1
host1

user2
host2



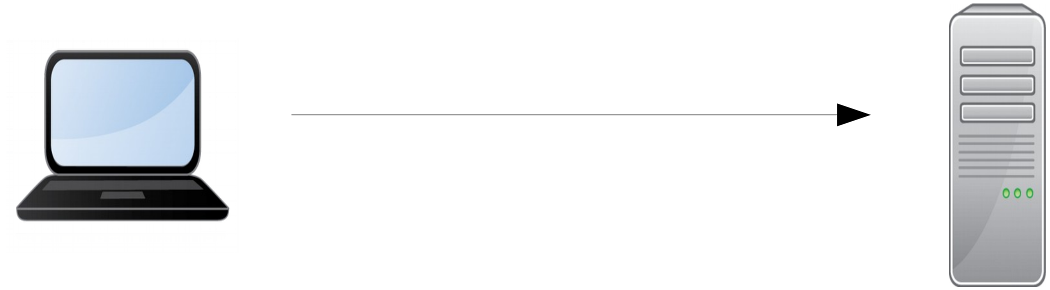
client

openssh-client

SSH Secure Shell

user1
host1

user2
host2



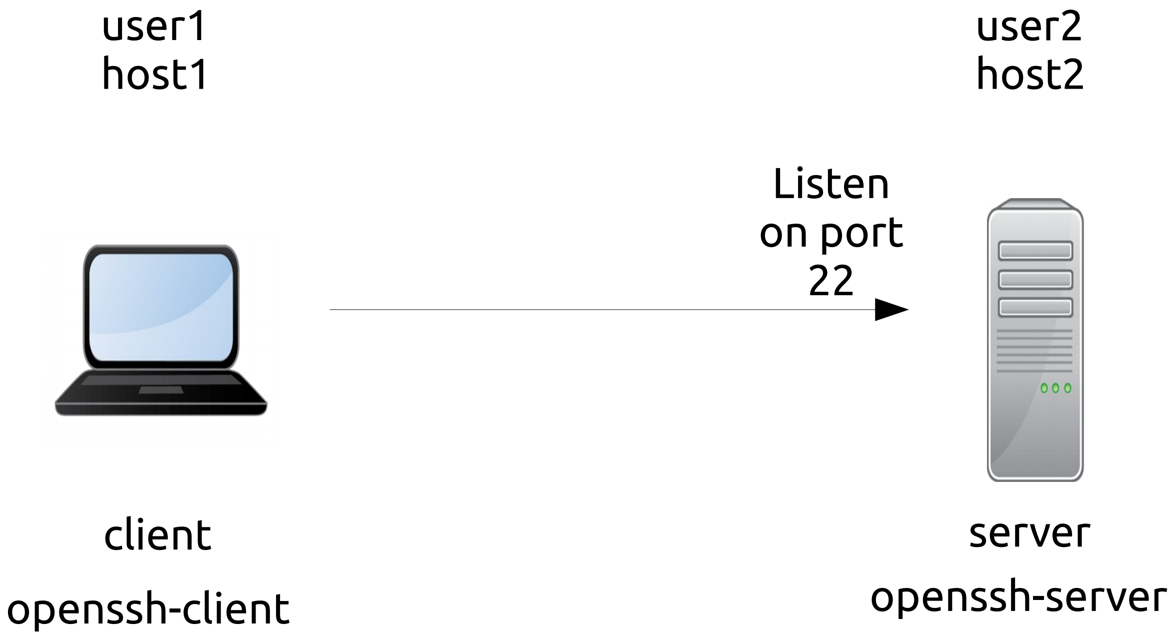
client

openssh-client

server

openssh-server

SSH Secure Shell



user1
host1

user2
host2

Listen
on port
22

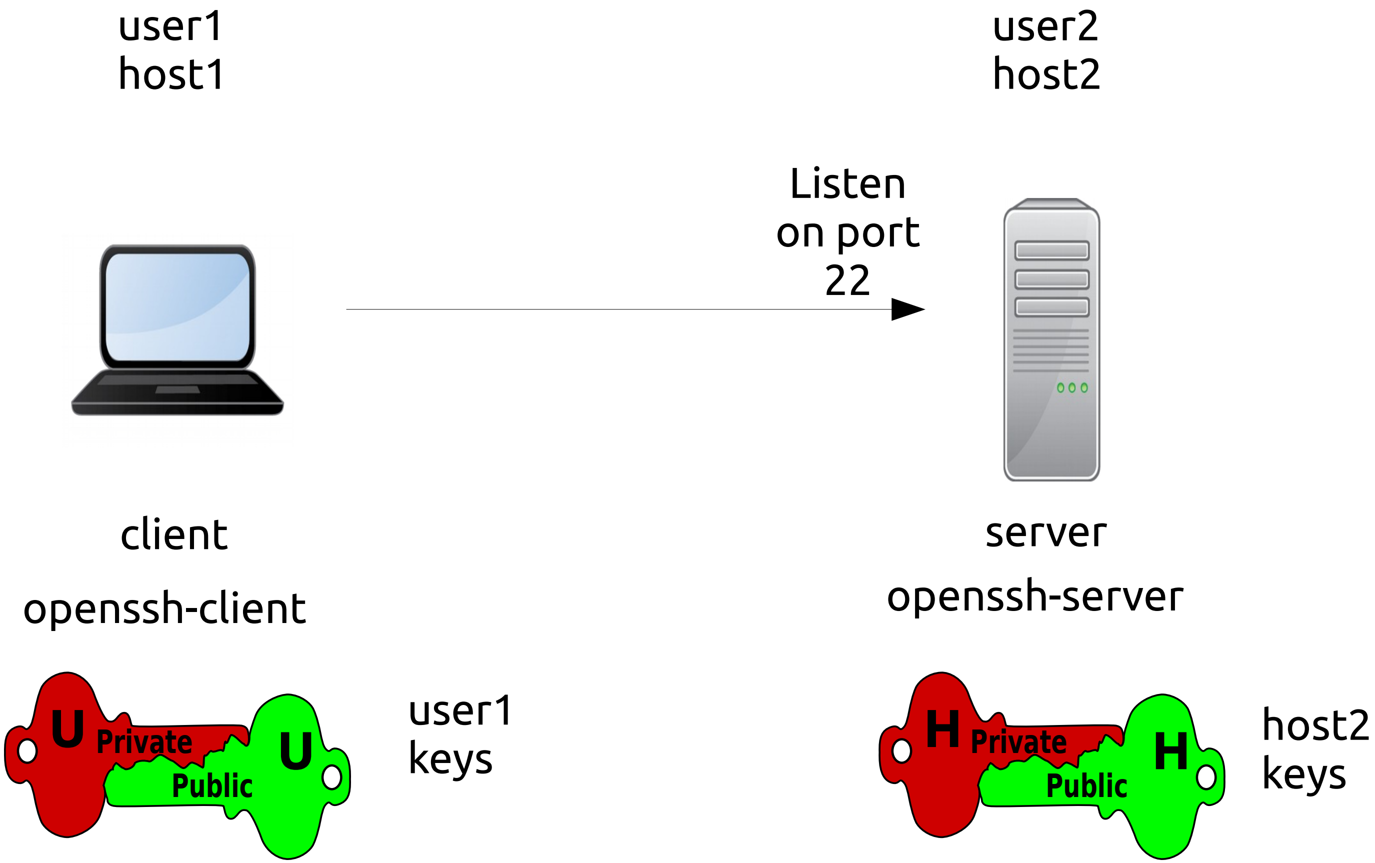
client

openssh-client

server

openssh-server

SSH Secure Shell



user1
host1

user2
host2

Listen
on port
22

client

openssh-client

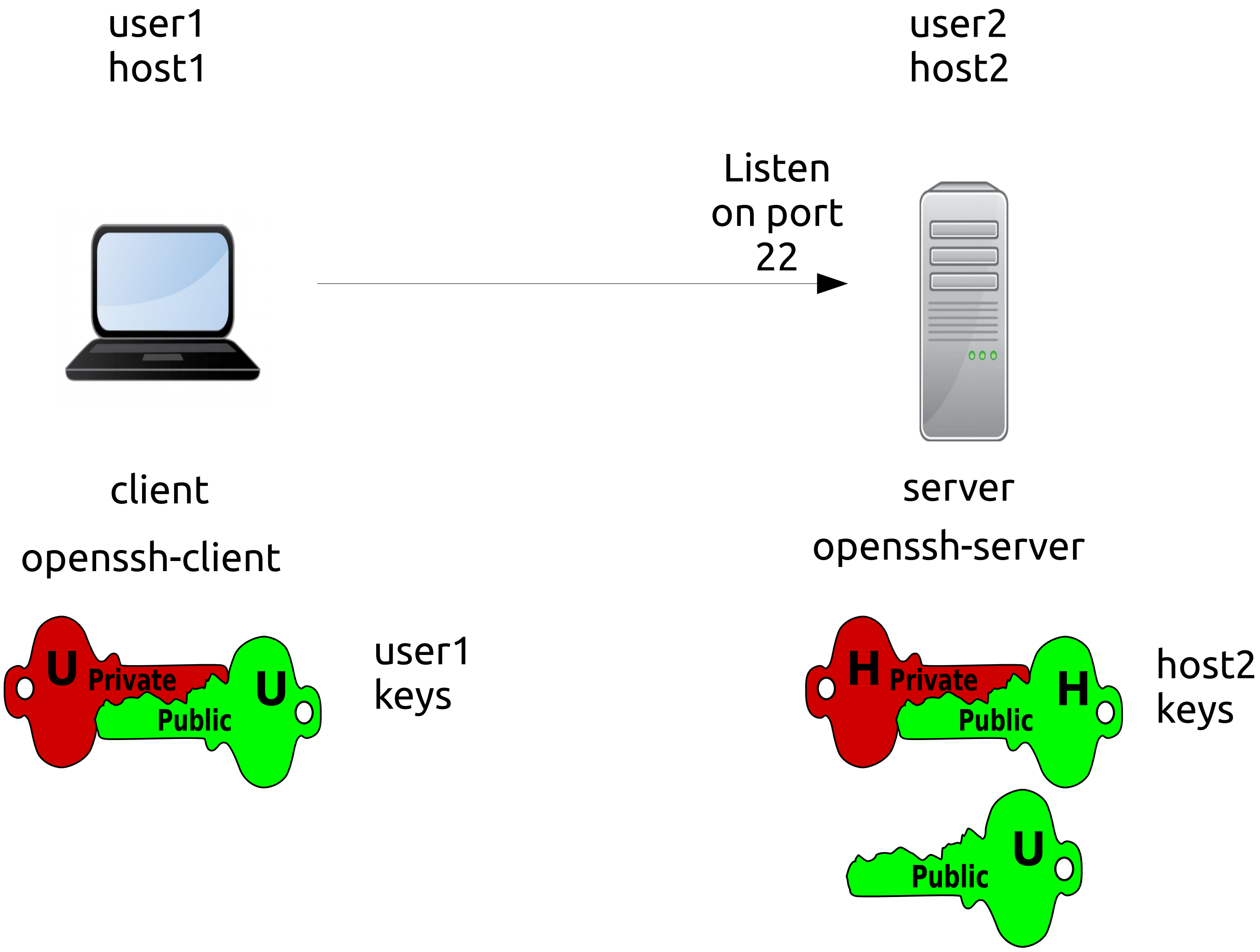
server

openssh-server

user1
keys

host2
keys

SSH Secure Shell



user1
host1

user2
host2

Listen
on port
22

client

openssh-client

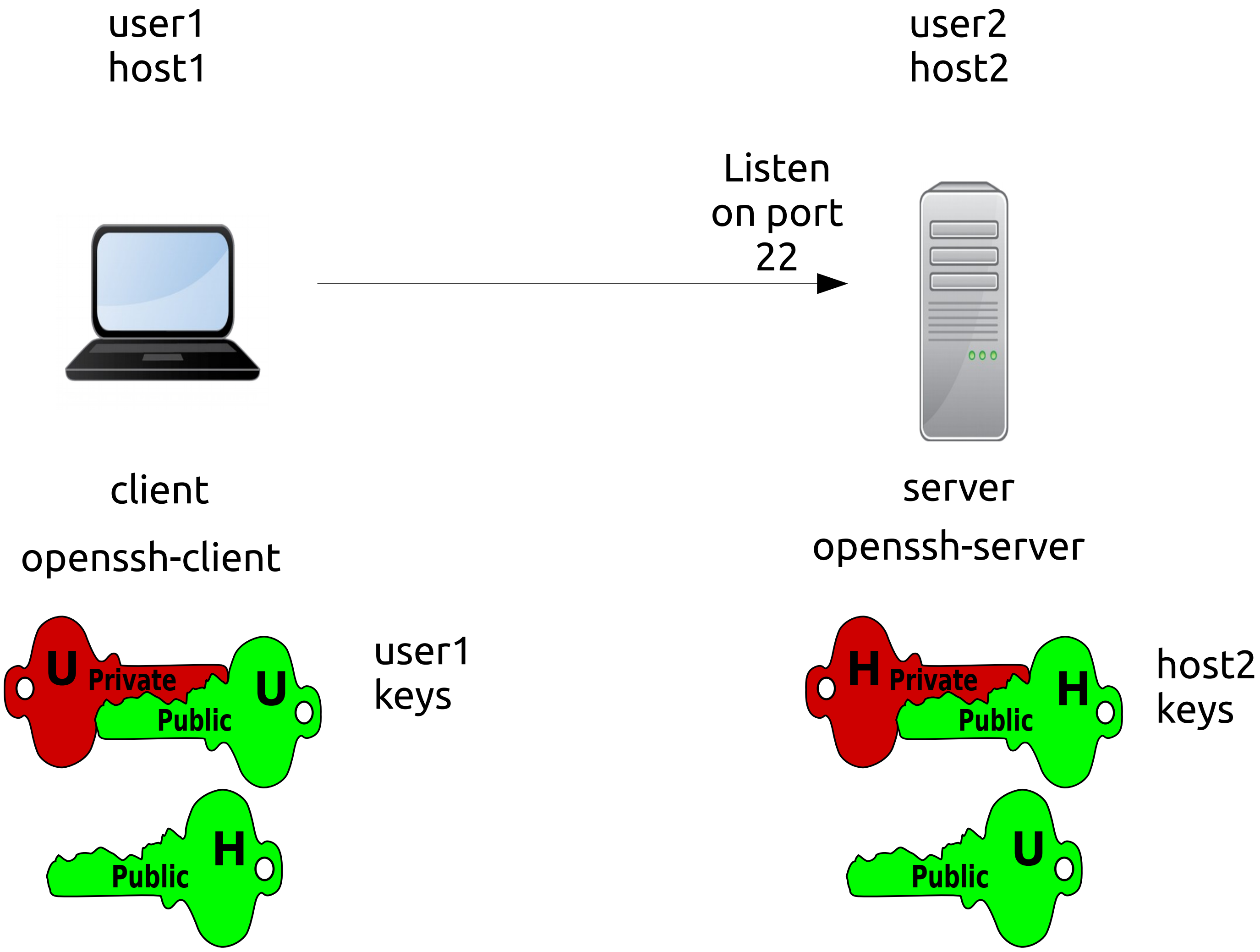
server

openssh-server

user1
keys

host2
keys

SSH Secure Shell



user1
host1

Listen
on port
22

user2
host2

client

openssh-client

server

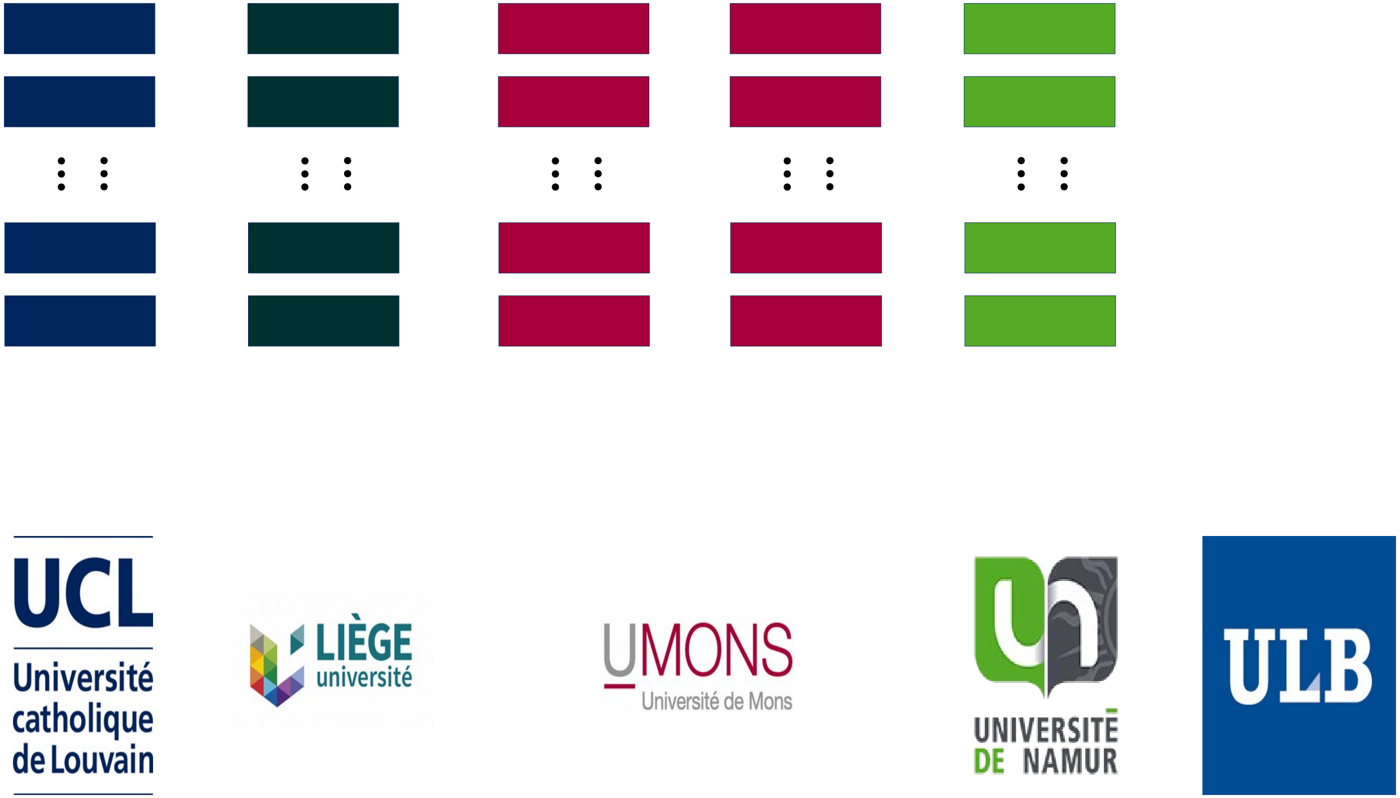
openssh-server

user1
keys

host2
keys

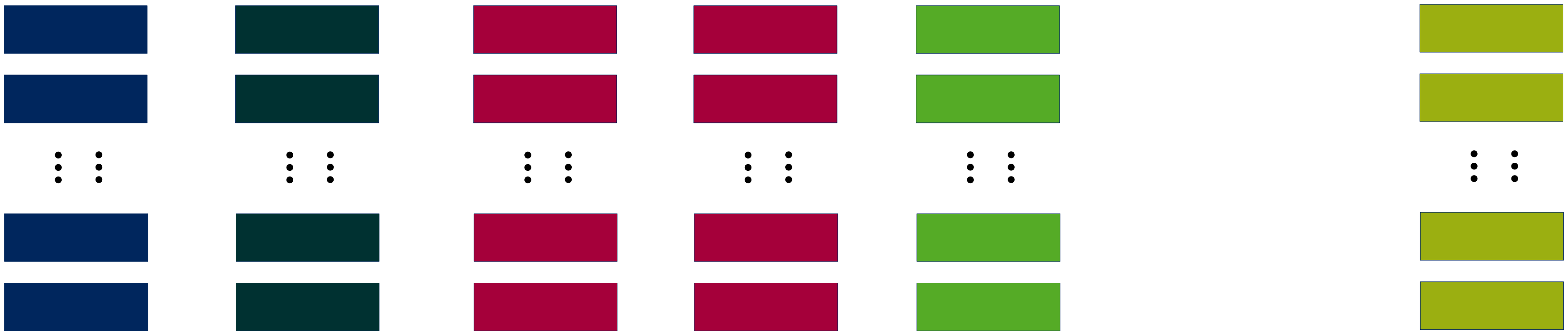
CONTEXT

Storage and compute nodes



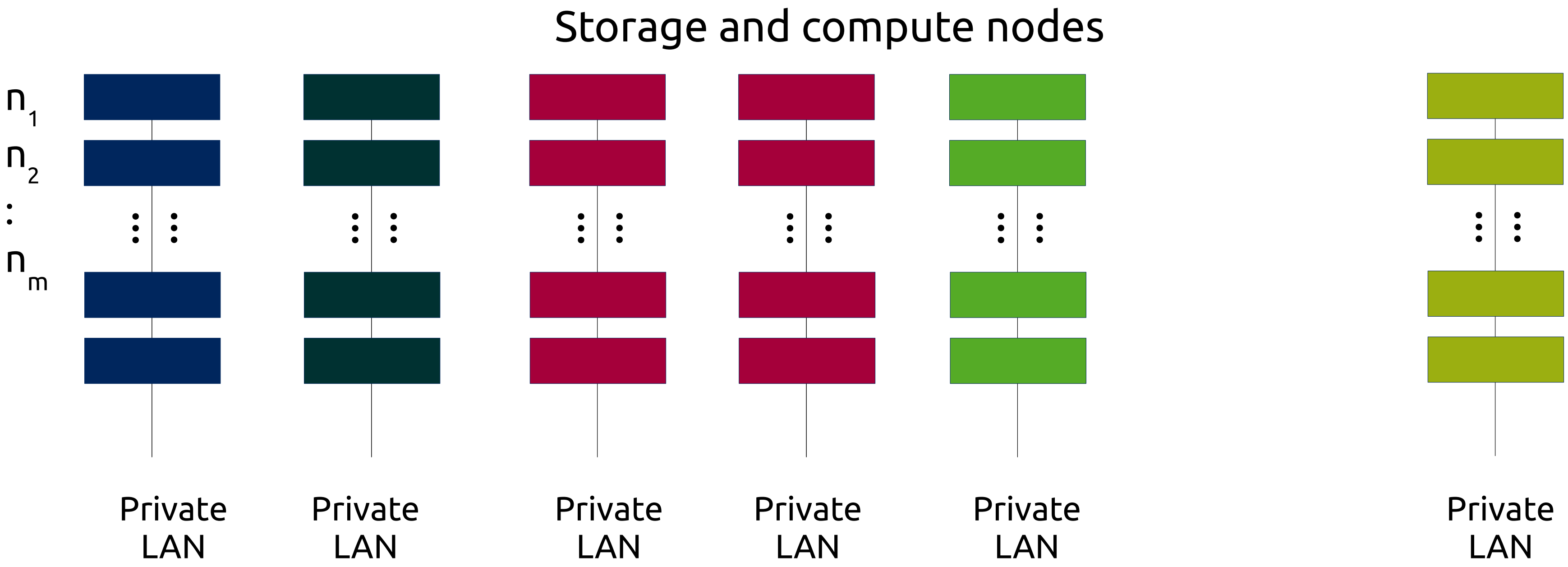
CÉCI is:
5 clusters from 5 french-speaking universities

Storage and compute nodes



Cenaero

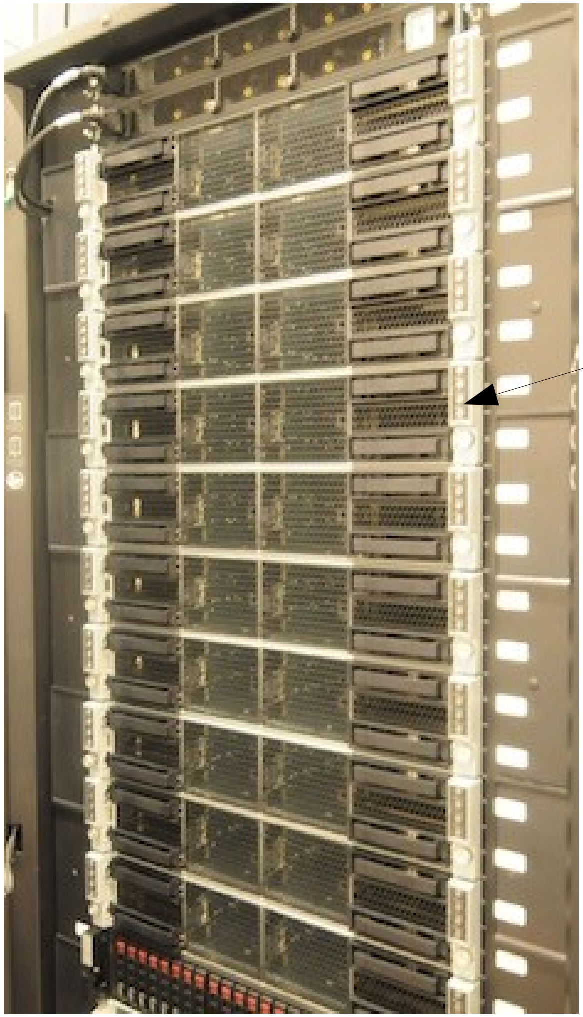
Tier-1 facility access for CÉCI users under special conditions



On each cluster
storage, compute nodes and frontend are interconnected
in a private network

Example

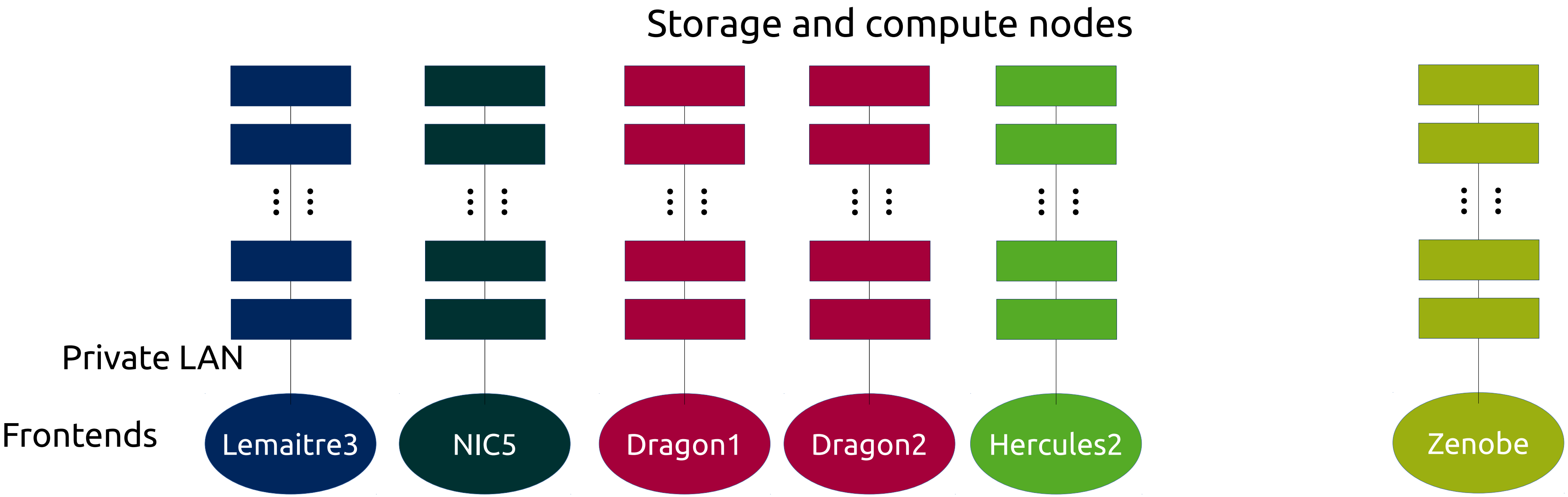
Lemaitre3 (UCLouvain)



Dragon2 (UMons)

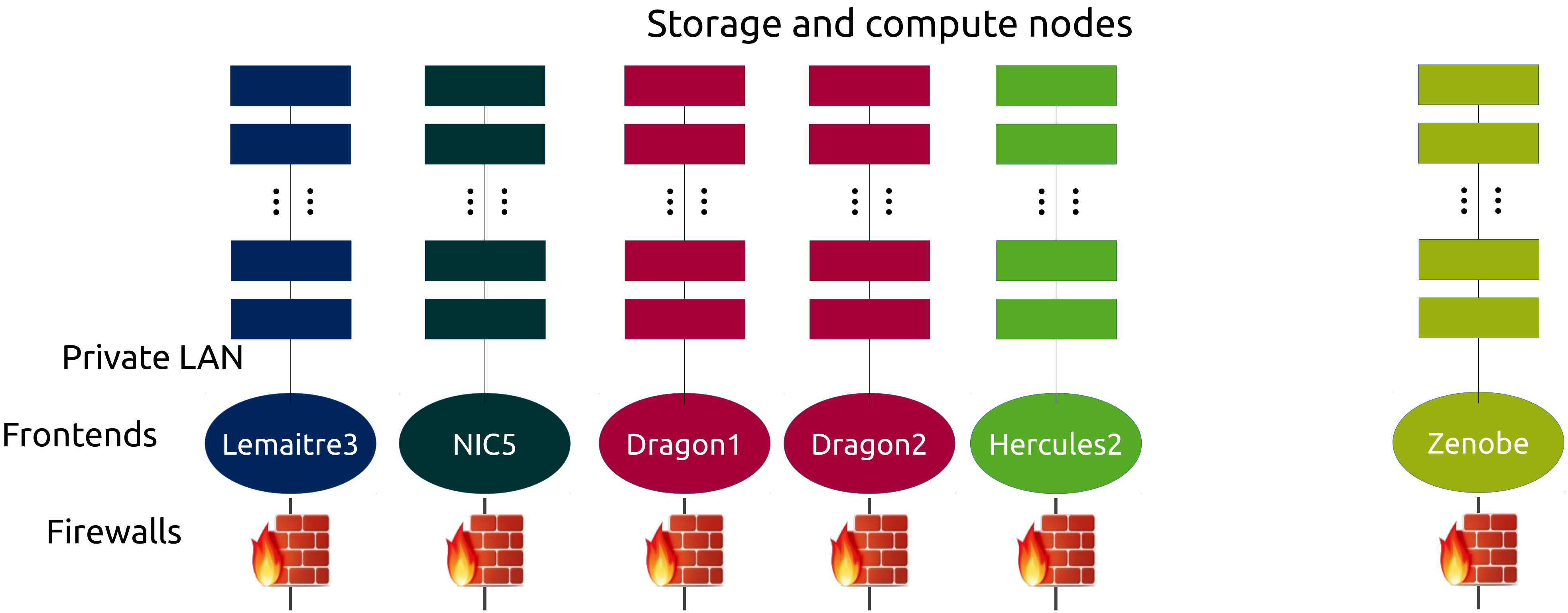
Compute nodes

Interconnections

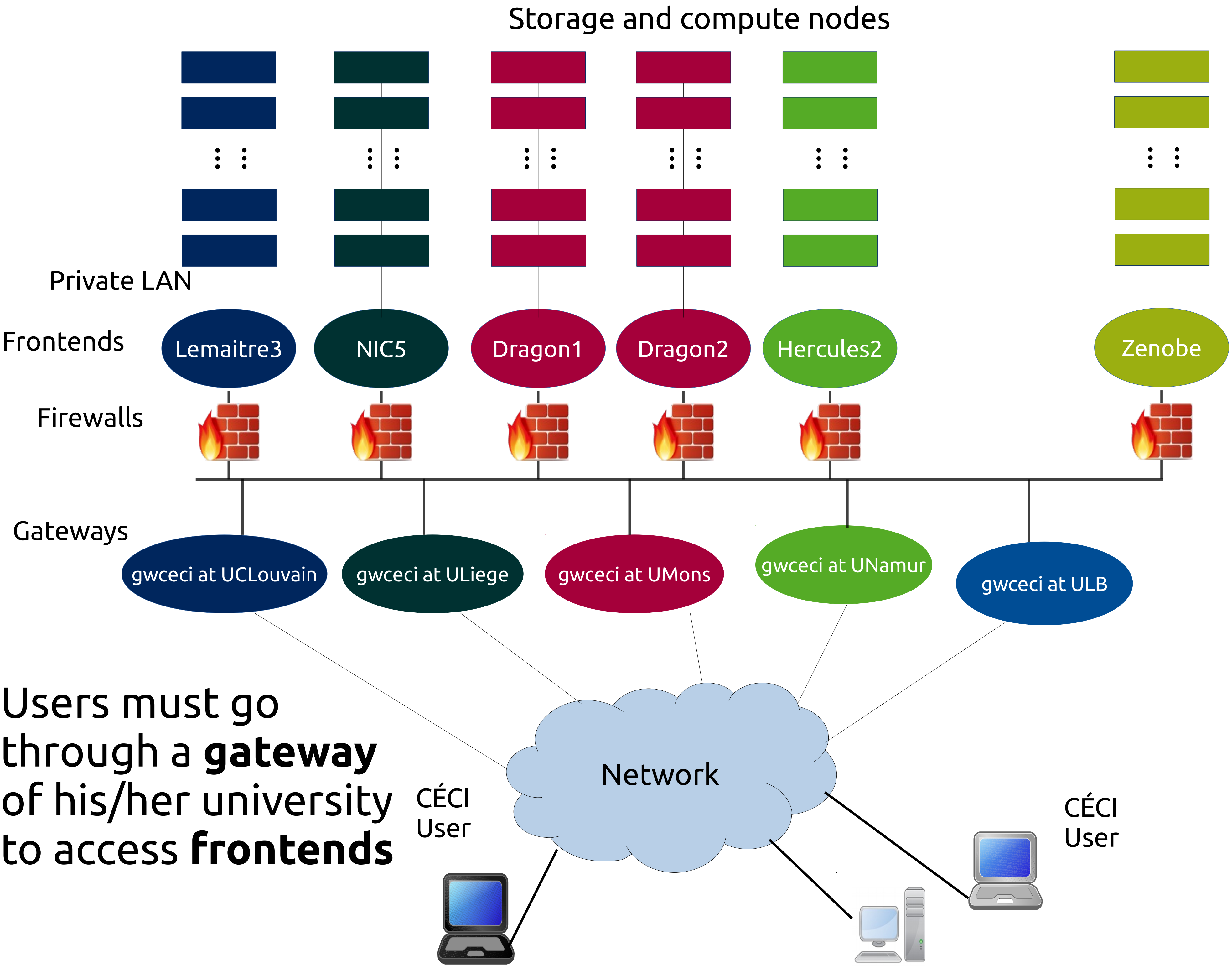


You need to connect to the **frontend** to
- **submit jobs** to the compute nodes
- **access** your results
- **edit** your files
- **compile** and **debug**
- **transfer** your data

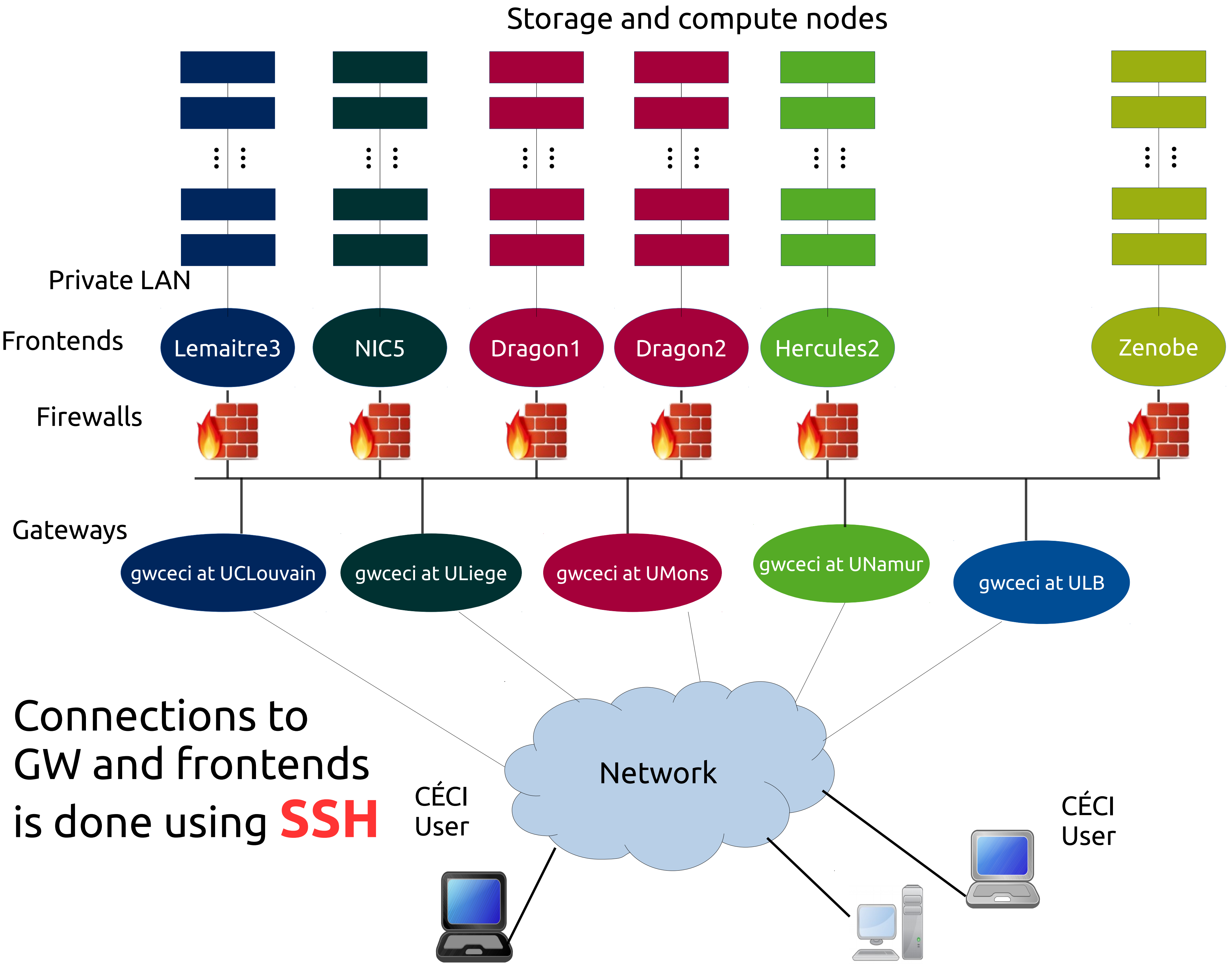
Do not run heavy jobs on the frontend



Frontends access is protected by a firewall that allows **only** connections **from a gateway**



Users must go through a **gateway** of his/her university to access **frontends**



Connections to GW and frontends is done using **SSH**

Fronted hostnames:

- Lemaitre3 (UCL): lemaitre3.cism.ucl.ac.be
- NIC4 (ULiège) : login-nic4.segi.ulg.ac.be
- Hercules2 (UNamur): hercules.ptci.unamur.be
- Dragon1 (UMons): dragon1.umons.ac.be
- Dragon2 (UMons): dragon2.umons.ac.be
- Vega (ULB): vega.ulb.ac.be

Gateway hostnames:

- UCL: gwceci.cism.ucl.ac.be
- ULB: gwceci.ulb.ac.be (use ULB VPN outside Belgium network)
- UMons: dragon2.umons.ac.be (use UMons VPN outside Univerity network)
- UNamur: gwceci.unamur.be (aka hal.unamur.be)
- ULiège: gwceci.uliege.be (use ULiège VPN outside Univerity network)

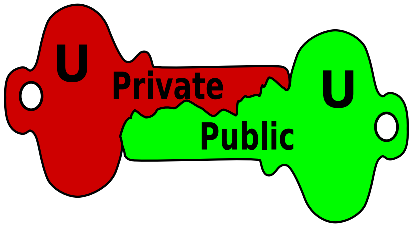
CONNECTING TO THE FRONTEND

SSH authentication uses **asymmetric cryptography** with **a pair of keys**, one private and one public

When you ask for a new CÉCI account or renew your account at https://login.ceci-hpc.be, two keys are generated

id_rsa.ceci

Private

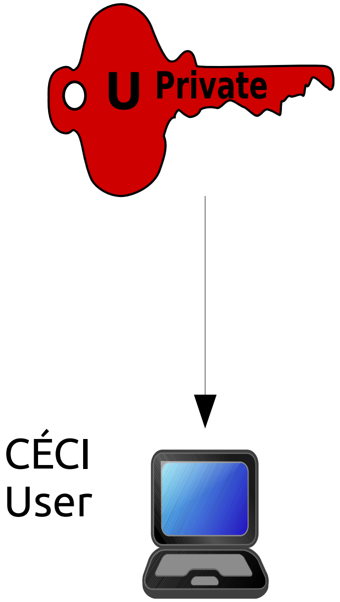


Public

id_rsa.ceci.pub

The private key is **encrypted using the passphrase** and **sent to you by email**

Your key must be stored in a safe place in your computer.



```
$ cat id_rsa.ceci

-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,798194AFB2800B27

KnvjN+KM4NogUADgdVI7GawGEmxJtXl2NKbezDyI8aeUAYxHemgThcRMswe2DAPs
fCeAJkTZ/B23uAWRppVvuPwJtp/AD3cvYxY5jBvSwVlAUdrfOJauegGc99CqvDEV
...
...
wT/yGuuRi9xfn6/yY7wTDxeaJg5WRd54oq0jbpTPUQmZWjJ1cuzBNiioNBXAFTGD
OJkZChE7fLD+C7kvYH0J6u4NiXUWqVheNerl0OnCZuM770gY5P0Q7w==
-----END RSA PRIVATE KEY-----
```

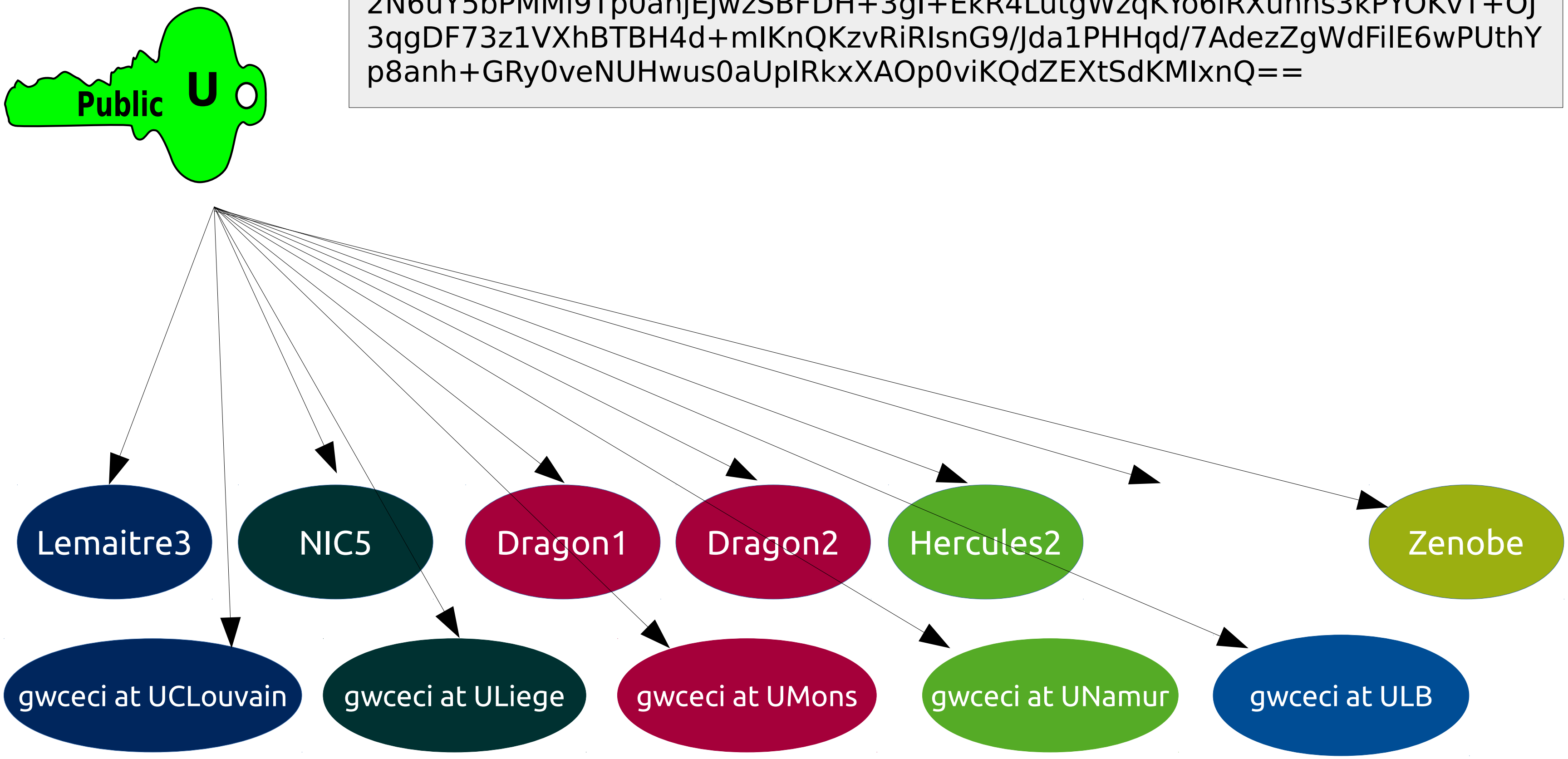
For security reasons

CÉCI does not keep a copy of your private key.

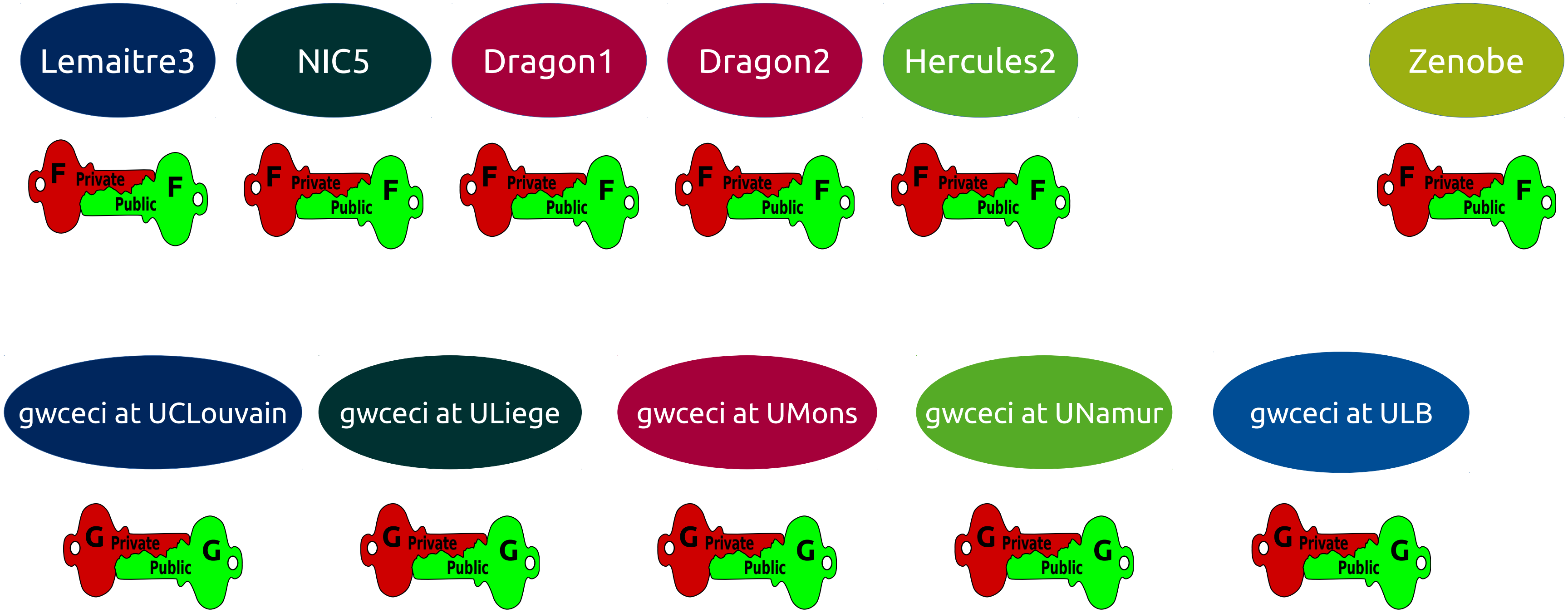
If you lose your key or passphrase or think it is compromised you must **renew your CÉCI account** at https://login.ceci-hpc.be

Your public key is copied to each CÉCI frontend and gateway for authentication

```
$ cat id_rsa.ceci.pub

ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEA2U59janaM1uhC4R1yL4Iozlx4FvQ6a
Q0tqIv9c6EHGj2wafVG8bxR1StYYecQ1oaY2C3AUeu9bTjtH9Rj5IPlvFf4OPAFMgU5
9SFabgeCZcNJbvZdpyI3mrEhTZLRTNhlohRoMACRot7rAxiKg62j2myfwWPXygwC4j
2N6uY5bPMMi9Tp0anjEJwzSBFDH+3gI+EkR4LutgWzqKYo6lRXuhhs3kPYOKvT+OJ
3qgDF73z1VXhBTBH4d+mIKnQKzvRiRIsnG9/Jda1PHHqd/7AdezZgWdFilE6wPUthY
p8anh+GRy0veNUHwus0aUpIRkxXAOp0viKQdZEXtSdKMIxnQ==
```



Each frontend as it's own private and public key



Lemaitre3 NIC5 Dragon1 Dragon2 Hercules2 Zenobe

gwceci at UCLouvain gwceci at ULiege gwceci at UMons gwceci at UNamur gwceci at ULB

Getting your private key

Users with CÉCI university email can ask for an account at:

https://login.ceci-hpc.be/init/

- Click 'Create Account'
- Type in your email address
- Click on the link sent to you by email.
- Fill-in the form and hit the “Submit” button.
- Wait … (A sysadmin is reviewing your information).
- Receive your private key by email.
- Save your key id_rsa.ceci file from your e-mail to your Downloads directory

Getting your private key

1) Open a terminal
2) Create the .ssh directory if it does not exist and set permissions

```
$ mkdir ~/.ssh
$ chmod 700 ~/.ssh
```

3) Move your key to this directory

```
$ mv id_rsa.ceci  ~/.ssh/.
```

4) Change the permissions of the file so that only you can read it

```
$ chmod 600 ~/.ssh/id_rsa.ceci
```

5) Check the permissions. Use the follow commands :

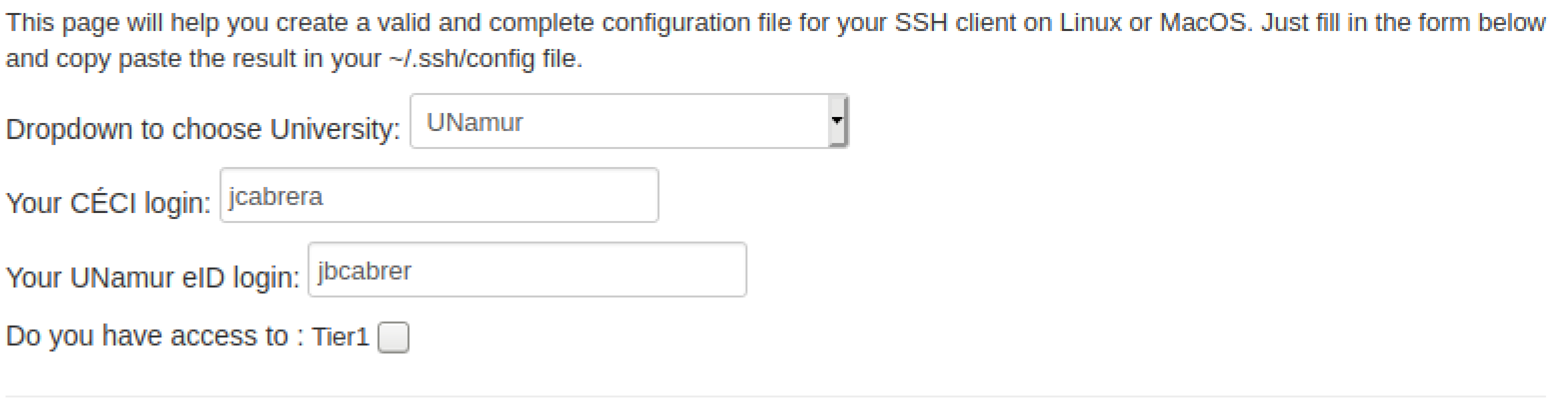
```
$ ls -l ~/.ssh/id_rsa.ceci
-rw------- 1 user user 1743 oct 18 06:48 .ssh/id_rsa.ceci
$ ls -ld .ssh
drwx------ 2 user user 4096 oct 18 06:45 .ssh
```

Must output -rw------ and drwx------ permissions
6) Create the public key

```
$ ssh-keygen -y -f ~/.ssh/id_rsa.ceci > ~/.ssh/id_rsa.ceci.pub
```

Creating your configuration file

- Go to the CÉCI wizard http://www.ceci-hpc.be/sshconfig.html
- Chose your university.
- Set your CÉCI and gateway login name.
- Depending on your university, the number of inputs fields will change.
- Tick the field "tier 1" if you have access to zenobe.
  If you are not sure, leave it unchecked.



This page will help you create a valid and complete configuration file for your SSH client on Linux or MacOS. Just fill in the form below and copy paste the result in your ~/.ssh/config file.

Dropdown to choose University: UNamur

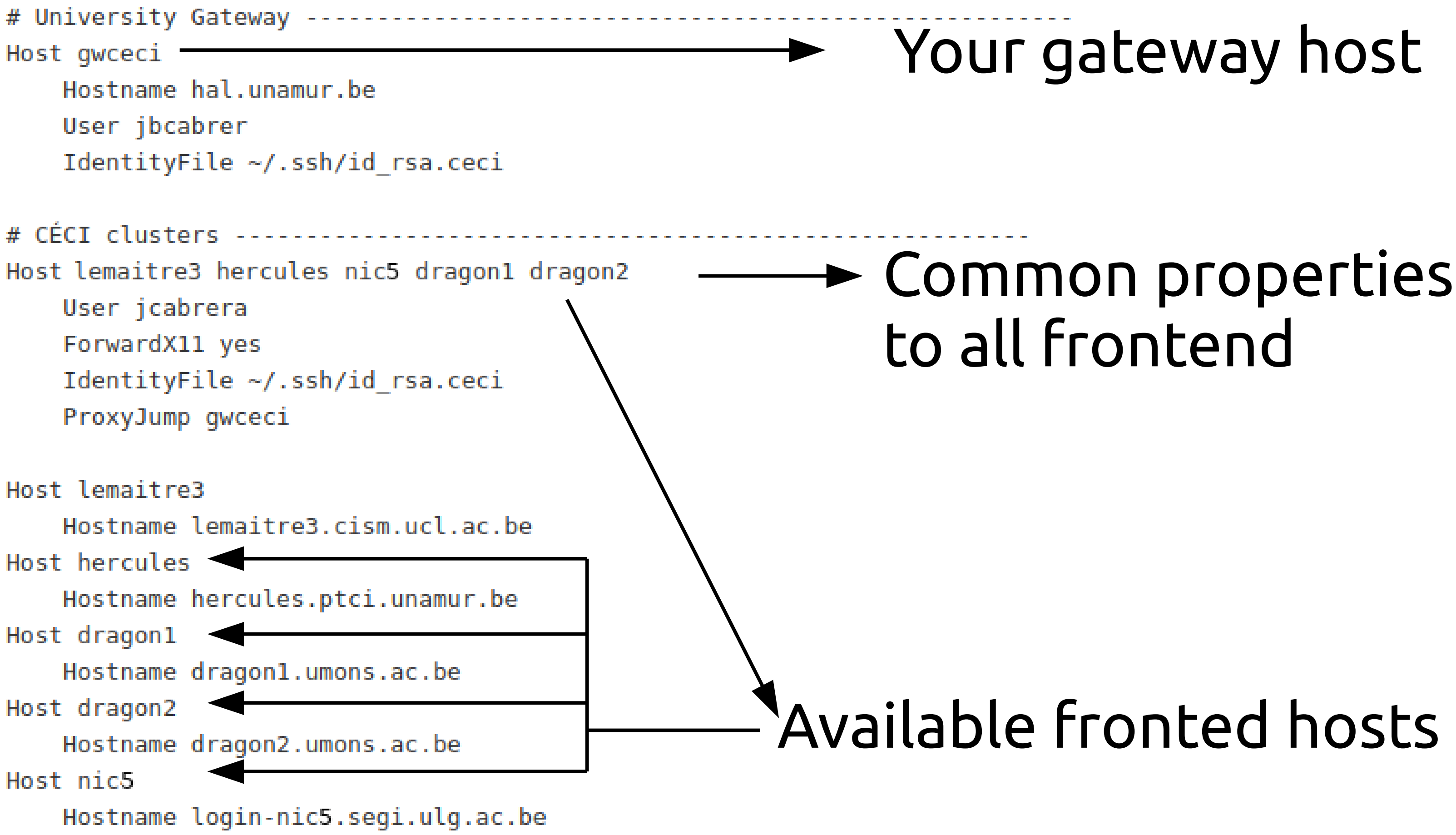
Your CÉCI login: jcabrera

Your UNamur eID login: jbcabrer

Do you have access to : Tier1 □

Creating your configuration file

Copy and paste the result in the .ssh/config file



```
# University Gateway -----------------------------------------------------
Host gwceci
    Hostname hal.unamur.be
    User jbcabrer
    IdentityFile ~/.ssh/id_rsa.ceci

# CÉCI clusters -------------------------------------------------------
Host lemaitre3 hercules nic5 dragon1 dragon2
    User jcabrera
    ForwardX11 yes
    IdentityFile ~/.ssh/id_rsa.ceci
    ProxyJump gwceci

Host lemaitre3
    Hostname lemaitre3.cism.ucl.ac.be
Host hercules
    Hostname hercules.ptci.unamur.be
Host dragon1
    Hostname dragon1.umons.ac.be
Host dragon2
    Hostname dragon2.umons.ac.be
Host nic5
    Hostname login-nic5.segi.ulg.ac.be
```

Your gateway host

Common properties to all frontend

Available fronted hosts

First connexion

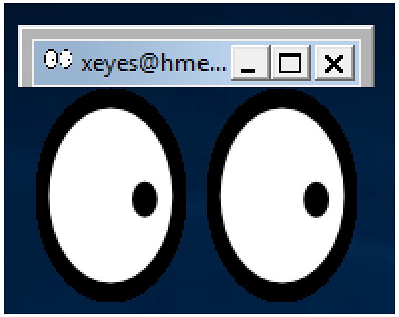
Connect to a cluster with the command

```
$ ssh host
```

where **host** is one of the frontend names defined in the configuration file.

The option **ForwarX11** in your configuration file allows you to open a remote window. For this, on **MacOs > 10.7** users need to install xquartz (needs reboot)

Try in **lemaitre3** the command xeyes



Agent and Passphrase managers

Use an SSH agent which will remember the passphrase so you do not have to type it in each time you issue the SSH command.

Most of the time an ssh-agent starts automatically at login if a password managing software is installed :

Mac OS Keychain, KDE KWallet, Gnome Keyring (Seahorse), etc.

Gnome Keyring loads all private keys in ~/.ssh **which have the corresponding public key**.

In MacOS add in ~/.ssh/config

```
Host *
   UseKeychain yes
   AddKeysToAgent yes
```

Agent and Passphrase managers

Make sure you have an agent running

```
$ ssh-add -l
Could not open a connection to your authentication agent.
```

```
$ ssh-add -l
The agent has no identities.
```

If you get "Could not open a connection to your authentication agent." start an agent with

```
$ eval $(ssh-agent)
```

If you get "The agent has no identities." The agent is already running. Add your key. Your key is decrypted and stored in memory

```
$ ssh-add ~/.ssh/id_rsa.ceci
Enter passphrase for /home/user/.ssh/id_rsa.ceci:
Identity added: /home/user/.ssh/id_rsa.ceci (/home/user/.ssh/id_rsa.ceci)
```

check the loaded key

```
$ ssh-add -l
2048 20:6c:8c:cd:e8:e6:9b:4f:8c:9c:d6:8a:eb:37:6d:17 /home/user/.ssh/id_rsa.ceci (RSA)
```

Frequent mistakes

The permissions on your key file are not correct

- **Error**: bad permissions

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@         WARNING: UNPROTECTED PRIVATE KEY FILE!          @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
Permissions 0644 for '/home/user/.ssh/id_rsa.ceci' are too open.
It is recommended that your private key files are NOT accessible by others.
This private key will be ignored.
bad permissions: ignore key: /home/user/.ssh/id_rsa.ceci
user@host's password:
it means that Permissions 0644 for '/home/user/.ssh/id_rsa.ceci' are too open.
Change them to 600 as explained in the first section of this document.
```

- **Problem:** Permissions 0644 for '/home/user/.ssh/id_rsa.ceci' are too open.
- **Solution**: Change them to 600 as explained previously

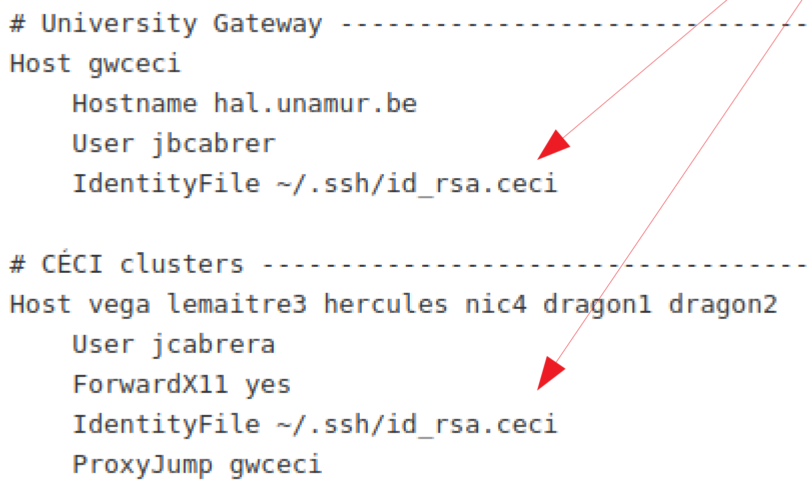
```
$ chmod 600 ~/.ssh/id_rsa.ceci
```

You did not specify the correct path to your SSH key

- **Error**: you are being asked for a password directly

```
$ ssh frontend
user@frontend's password:
```

- **Problem**: your SSH client did not use the SSH key.
- **Solution**: Make sure that your .ssh/config is properly configured and the key is present.



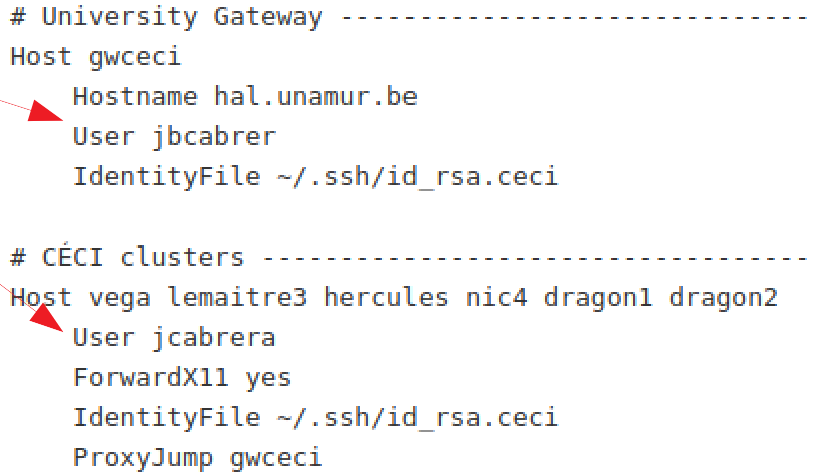
```
# University Gateway ------------------------------
Host gwceci
    Hostname hal.unamur.be
    User jbcabrer
    IdentityFile ~/.ssh/id_rsa.ceci

# CÉCI clusters -----------------------------------
Host vega lemaitre3 hercules nic4 dragon1 dragon2
    User jcabrera
    ForwardX11 yes
    IdentityFile ~/.ssh/id_rsa.ceci
    ProxyJump gwceci
```

You used a wrong username or tried to connect before your keys are synchronized

- **Error**: you are being asked for a passphrase, then a password

```
$ ssh frontend
Enter passphrase for key '/home/user/.ssh/id_rsa.ceci':
user@frontend's password:
```

- **Problem**: the user name you are using is not the correct one or you are trying to connect with the new private key while it has not been synchronized to the cluster yet.
- **Solution**: Verify your user name or wait ~30 min



```
# University Gateway -------------------------------
Host gwceci
    Hostname hal.unamur.be
    User jbcabrer
    IdentityFile ~/.ssh/id_rsa.ceci

# CÉCI clusters -------------------------------------
Host vega lemaitre3 hercules nic4 dragon1 dragon2
    User jcabrera
    ForwardX11 yes
    IdentityFile ~/.ssh/id_rsa.ceci
    ProxyJump gwceci
```

Troubleshooting

You can use -v, -vv or -vvv to troubleshooting a session

```
$ ssh frontend -v
OpenSSH_7.6p1 Ubuntu-4ubuntu0.5, OpenSSL 1.0.2n  7 Dec 2017
debug1: Reading configuration data /home/user/.ssh/config
debug1: /home/user/.ssh/config line 4: Applying options for *
debug1: /home/user/.ssh/config line 126: Applying options for hercules
...
debug1: SSH2_MSG_KEXINIT sent
debug1: SSH2_MSG_KEXINIT received
...
debug1: Server host key: ssh-rsa SHA256:GfUSNZEFZg28WRCaxJvDNSCCIhrX1IujNIky29ui7IY
debug1: Host 'gwceci' is known and matches the RSA host key.
debug1: Found key in /home/user/.ssh/known_hosts:33
...
debug1: Offering public key: RSA SHA256:IMDnFOL/9DI4otUnSUJBMxLc0v3jXSHkGUsM4ogi5Us
/home/user/.ssh/id_rsa.ceci
debug1: Server accepts key: pkalg rsa-sha2-512 blen 277
debug1: Authentication succeeded (publickey).
Authenticated to gwceci ([YYY.YYY.YYY.YYY]:22).
...
debug1: Server host key: ecdsa-sha2-nistp256 SHA256:SyLaaBe7CuO7Dpa6vJa0vbAUxnYSpl30xaJo5yBF//c
debug1: Host 'frontend' is known and matches the ECDSA host key.
debug1: Found key in /home/user/.ssh/known_hosts:217
...
debug1: Offering public key: RSA SHA256:IMDnFOL/9DI4otUnSUJBMxLc0v3jXSHkGUsM4ogi5Us
/home/user/.ssh/id_rsa.ceci
debug1: Server accepts key: pkalg rsa-sha2-512 blen 277
debug1: Authentication succeeded (publickey).
Authenticated to frontend (via proxy).
...
```

SSH-based file transfer (SCP, rsync, SSHFS)

SCP

You can copy files/directories back and forth between computers

- Verify your agent is running and you have the ssh config file
- Create a temporary directory with dummy files on your computer

```
$ mkdir -p cours_ssh/scp_test; touch cours_ssh/scp_test/file{1..4}.txt
$ ssh frontend 'mkdir cours_ssh'
```

- Copy the directory to your home directory in one of the frontends and check

```
$ scp -r cours_ssh/scp_test host:cours_ssh/.
$ ssh frontend 'ls cours_ssh/scp_test/'
```

- Copy it back

```
$ scp -r frontend:cours_ssh/scp_test cours_ssh/scp_test2
```

- Copy between frontends is not permitted. Use $CECITRSF partition

- For a copy throw your computer use -3 option

```
$ scp -r -3 frontend1:cours_ssh/scp_test frontend2:cours_ssh/.
```

rsync

rsync is widely used for backups and mirroring and as an improved copy command for everyday use

Most common usage is to synchronize files with archive option ‘a’, and compress option ‘z’. If you want to get a copy of your hard work you did in the frontend to your laptop:

```
$ ssh frontend 'mkdir cours_ssh/rsync_test; touch cours_ssh/rsync_test/file{1..4}.txt'
$ rsync -avz --progress frontend:cours_ssh/rsync_test cours_ssh/.
```

Modify a file at the frontend and synchronize

```
$ ssh frontend 'echo "Adding hello1 word in $(hostname)" >> coursssh/rsynctest/file4.txt'
$ rsync -avz --progress frontend:coursssh/rsynctest coursssh/.
```

Modify a file in your computer and prevent Overwrite when synchronize -u

```
$ echo 'Adding hello in client' > cours_ssh/rsync_test/file3.txt
$ rsync -avzu --progress frontend:cours_ssh/rsync_test cours_ssh/.
```

Delete a file at the frontend and force delete it in your computer.

```
$ ssh host rm cours_ssh/rsync_test/file1.txt
$ rsync -avz --del --progress frontend:cours_ssh/rsync_test cours_ssh/.
```

SSHFS

Use SSHFS to mount a remote file system - accessible via SSH

Linux install:

Debian, Ubuntu

```
$ sudo apt-get install sshfs
```

Fedora/CentOs

```
$ yum install sshfs
```

MacOS Install:

Install FUSE and SSHFS from https://osxfuse.github.io/

SSHFS

Example: Mount your CECIHOME

Create on your computer a repository to mount the CÉCI home

```
$ mkdir frontend_home
```

Mount the remote CÉCI Home on your computer

```
$ cluster=frontend;
$ sshfs -o uid=`id -u` -o gid=`id -g` $cluster:$(ssh $cluster 'echo $CECIHOME')/ host_home
```

Create a file in the mounted directory

```
$ echo 'file content' > frontend_home/file_fuse.txt
```

Check the file content in the frontend

```
$ ssh frontend 'cat $CECIHOME/file_fuse.txt'
```

disconnect

```
$ fusermount -u frontend_home
```

ANNEXES

SSH Details

- OpenSSH Manual Pages
- RSA Cryptography Specifications Version 2.2
- The Secure Shell (SSH) Transport Layer Protocol