

Table of Contents

Access the clusters

Account

- Standard Account

- External Account

- Outsider Account

- Inactivity Notice and Account Deletion Policy

Cluster connection

- login nodes

- Connect using SSH

 - ssh PublicKey

 - multiple ssh key

 - From Linux and Mac OS

 - From Windows

- Access to the compute nodes

- GUI access / Desktop with X2Go

File transfer

- From Linux

- From Windows

- SSH tunnel and socks proxy

- Alternative to using ProxyJump

Access the clusters

Account

To access the HPC clusters (Baobab/Yggdrasil/bamboo) and submit jobs, you need a valid HPC account.

Access is reserved for:

1. Members of the University of Geneva (Unige)
2. Members of HES-SO GE
3. External collaborators
4. Guest



Each HPC account is linked to a Primary Investigator (PI) (=repondant = responsible (different terms but same purpose)). A PI is a member of the University of Geneva (Unige) who invites and takes responsibility for an user's access to the HPC service. The PI(repondant) ensures that the access is justified, appropriate, and compliant with university policies.

Standard Account

This is the default account type for Unige members. To request an account please follow fill the form on DW: <https://dw.unige.ch/openentry.html?tid=hpc>

To connect to the HPC cluster, you must authenticate using one of these methods:

1. Your [ISIS](#) username and password
2. An SSH key registered in your [ISIS account](#)

External Account

An **external account** is intended for collaborators from other institutions who are working closely with Unige researchers. (if you only need an HPC access [Outsiders account](#) is more appropriated)

How to request one:

- Ask your [department administrator](#) to register the external collaborator in the university database: <https://catalogue-si.unige.ch/en/isis>
- More info: <https://memento.unige.ch/doc/0214/>
- Once registered, follow the usual HPC account request procedure as described above.

Outsider Account

An **Outsider** is a person external to Unige who has been invited by a **PI** specifically to use the HPC service. This lightweight account is limited to HPC access and avoids standard administrative procedures. (24h waiting time to get an account)

Key details:

1. Authentication is done via the SSH Public Key (only) provided during the invitation process (Authentication with password is disabled)
2. SSH key can be updated at any time: [Update your SSH key](#)
3. SSH key updates are applied daily at 1:30 PM and 5:00 AM via the UNIGE Active Directory

Only PI can invite and manage Outsiders, visit:

<https://gestion-externe.unige.ch/main/outsider-requests>

(Access requires approval — please contact us with a short motivation if you'd like it enabled.)

Once access is granted, you will be able to create an invitation, which requires:

1. The email address of the future Outsider
2. Selection of the appropriate service (**High-Performance Computing**)
3. Setting an expiration date (maximum 1 year)
4. An optional note for the guest
5. Acceptance of the terms of use

Once invited, the future Outsider will receive an email with detailed instructions to finalize their registration. (**Tip: read it carefully!**)

To renew an expired Outsider account, a new invitation must be created.

← Formulaire de demande d'inscription

Indiquer l'adresse email fournie par l'institution à laquelle est rattachée l'invité-e

toto.tutu@gmail.ch Email(s) de(s) invité(s) séparés par ,

Détail de la demande

Service(s) demandé(s) * High performance computing Date d'expiration (j.mm.aaaa) *

Date limite d'utilisation du service par l'invité-e, maximum un an

Message complémentaire à l'attention de l'invité

☐ En validant la création ou la prolongation du compte pour toto.tutu@gmail.ch, je certifie que je le/la connais personnellement en tant que collègue ou étudiant/e, et que il/elle a le droit d'utiliser le service HPC selon les règles en vigueur ([Directives concernant le droit d'accès au service HPC](#)). J'accepte ainsi de devenir son/sa répondant/e envers l'université de Genève et les administrateurs du service HPC et les responsabilités qui en découlent.

Inactivity Notice and Account Deletion Policy

Accounts that have been inactive for a period of **one year** will be flagged for deletion. This is to ensure that we are not storing unused data and to avoid data being left without an owner in the event that the account is deleted from the UNIGE central user directory.

Users will receive an email notification prior to the scheduled deletion of their account, giving them the opportunity to log in and keep their account active. If no response is received within one month, the account will be deleted, along with all associated data.

Cluster connection

Once you have received an email confirming the creation of your account, you have access to our clusters: Baobab and Yggdrasil.

You can connect to the HPC clusters only through the login nodes. The clusters are reachable from outside UNIGE as well, without the need to use a VPN.

login nodes

- **For Baobab** : login1.baobab.hpc.unige.ch
- **For Yggdrasil** : login1.yggdrasil.hpc.unige.ch
- **For Bamboo** : login1.bamboo.hpc.unige.ch

Connect using SSH

You can access the clusters from anywhere through ssh with your [ISIs](#) account and password.

ssh PublicKey

If you want to authenticate to the cluster using your ssh key:



1. For security measure it isn't possible to add your ssh key to AuthorizedKeyFile. This has been disabled to prevent any non registered at UNIGE to log in.
2. Your ssh public key must be added to your UNIGE account profile. (it's like if the AuthorizedKeyFile is bonded to your UNIGE identity)
3. The UNIGE central directory synchronizes the ssh key every 10/15 min.
4. More information on <https://hpc-community.unige.ch/t/authentication-modification-sshpublickey-management/3011>

1. Generate your ssh-key (We strongly recommend to create it with password/passphrase for more security) by following [this documentation \(multi-platform\)](#)

2. Update your public key to your Isis profile by updating “My SSH public key” on:

- [my-account](#) for UNIGE/external users.
- [applicant](#) for Outsider users.

Note: Make sure you copy the public ssh key linked to the private key you're going to use. If you have regenerated your ssh key, you'll need to put your public key in [my-account](#) or [applicant](#)

multiple ssh key

It is possible to register multiple SSH public keys on the authentication server. However, [my-account.unige.ch](#) does not allow this at the moment (work in progress). In the meantime, please send your request to the LDAP team directly at dl-distric-windows-team@unige.ch.



After requesting an additional sshPublicKey, if you update it via my-account, all previous references will be overwritten.

The first time you connect, your SSH client will ask you to confirm the **fingerprint** of the server. These are the current SSH fingerprint for both Baobab and Yggdrasil :

```
[root@login1 ~]# ssh-keygen -l -E md5 -f /etc/ssh/ssh_host_rsa_key.pub
2048 MD5:8f:75:c4:18:8a:75:f1:f1:19:4d:85:92:3b:b6:2a:e1 no comment (RSA)
```

Or this one

```
(yggdrasil)-[root@login1 ~]$ ssh-keygen -l -f /etc/ssh/ssh_host_rsa_key
2048 SHA256:tKqp4nLjL+EGVKl8T0VF2nS36DkHVFMPxQOPg/gKvg
```

```
/etc/ssh/ssh_host_rsa_key.pub (RSA)
```

If you connect using x2go (at least on windows), you'll see another **fingerprint**:

```
[root@login1 ~]# for x in /etc/ssh/*.pub ; do echo $x && cut -d' ' -f2 < $x  
| base64 -d | openssl sha1 -c ; done ;  
/etc/ssh/ssh_host_rsa_key.pub  
(stdin)= 67:03:fc:6f:32:7c:19:9b:97:b9:e8:7b:12:1d:ad:a6:7b:c9:4c:9c
```

If you type your username/password **wrong 3 times** in a row, you will be **banned for 15 minutes** before you can try again. Please read more in the [Troubleshooting](#) section.

From Linux and Mac OS

Connect to the login node from the terminal:

```
ssh youruser@clusterhostname
```



add -Y if you need X forwarding

```
ssh -Y youruser@clusterhostname
```

N.B. : replace clusterhostname with the [login node](#) of the cluster you are willing to use.

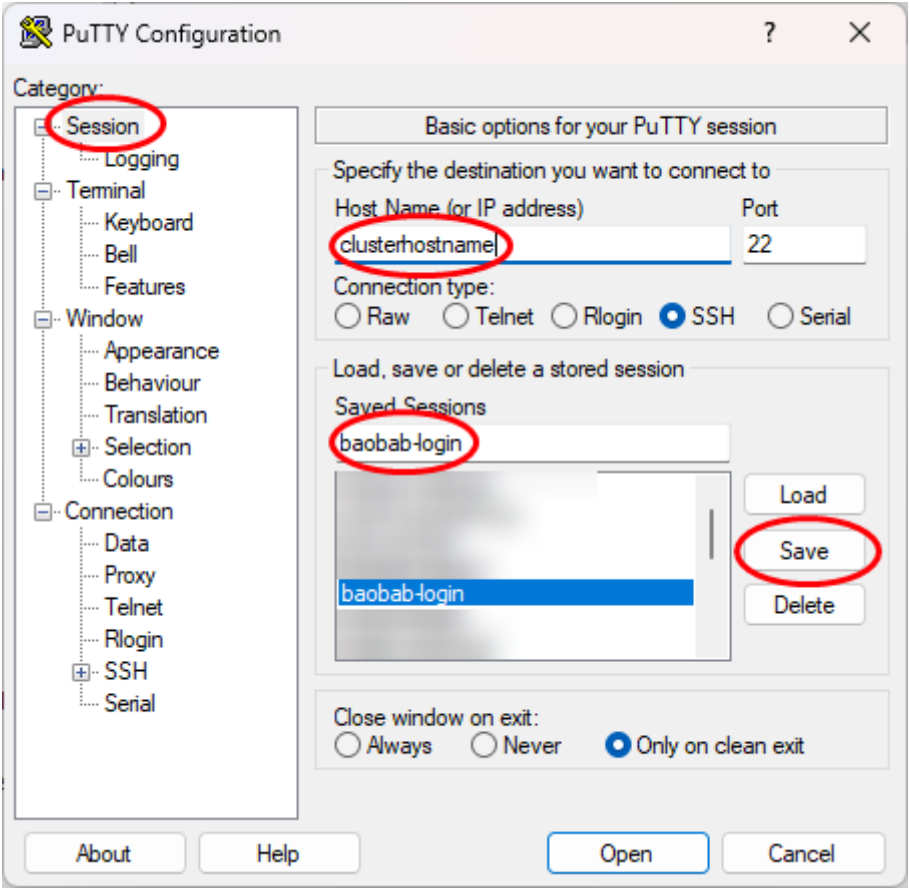
From Windows

To access Baobab or Yggdrasil in SSH from Windows, you can use :

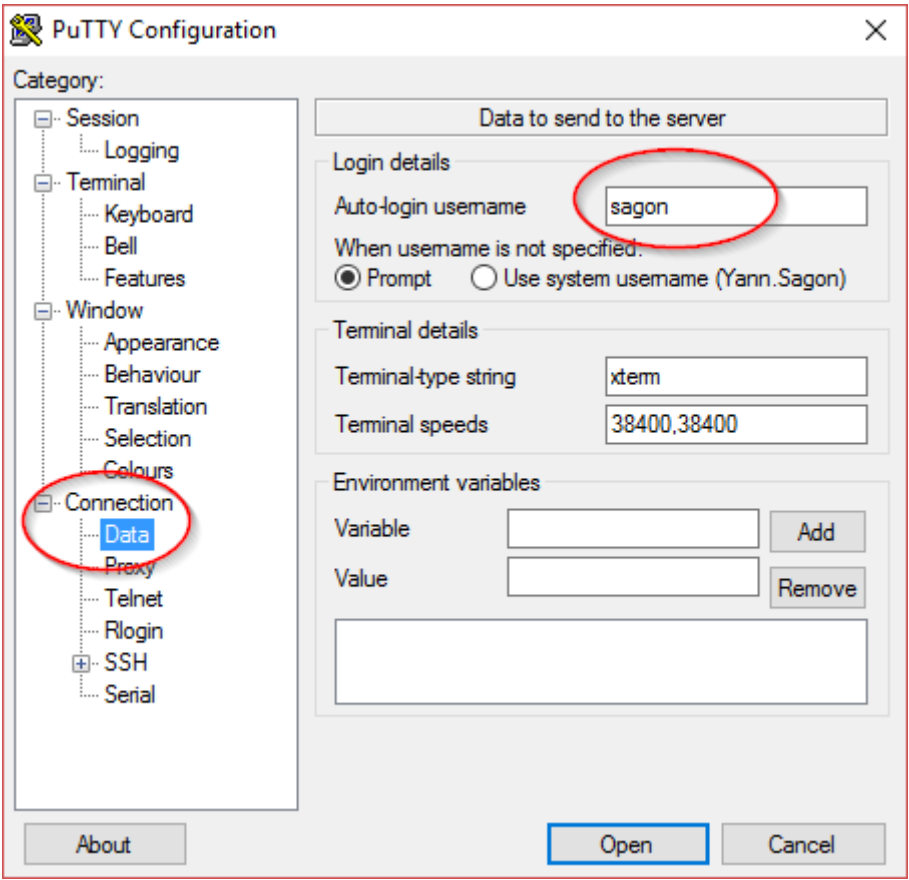
- ssh client integrated in PowerShell. Recent versions of Window 10 offer a ssh client. You can use it the same way you would use it on Linux (see [above](#))
- PuTTY which has been the *de facto* solution for years.
 - You can download it here [PuTTY](#).
 - You will find the needed information on the screenshots below.

N.B. : replace clusterhostname with the [login node](#) of the cluster you are willing to use.

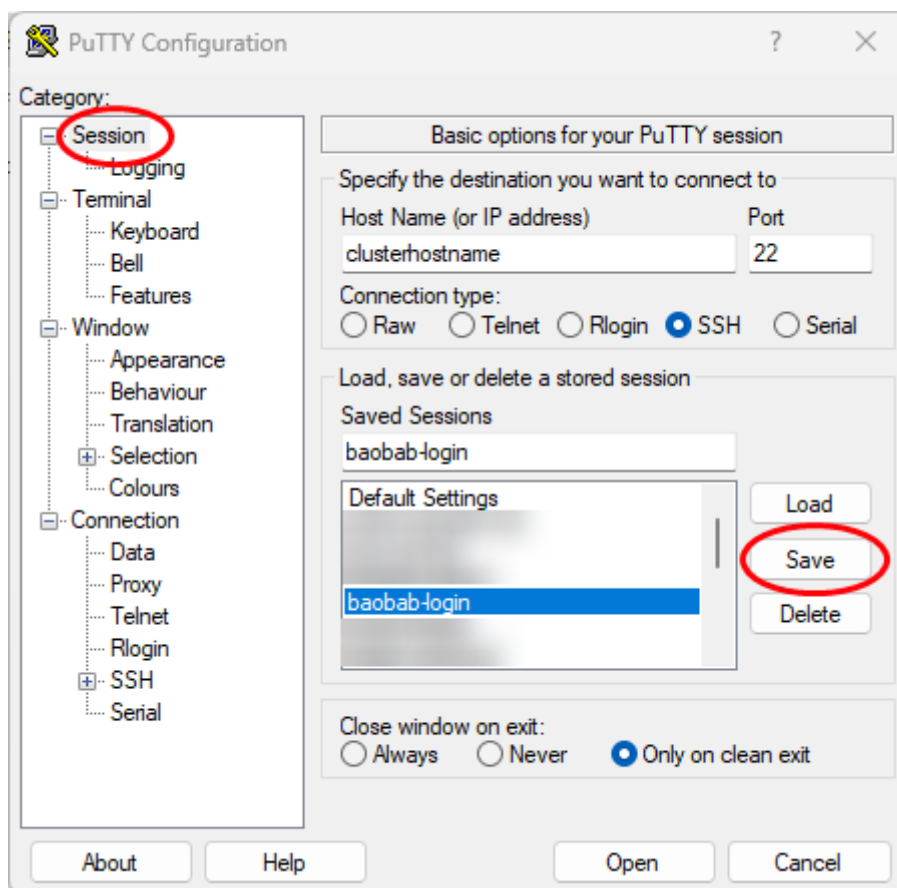
[PuTTY create session](#) :



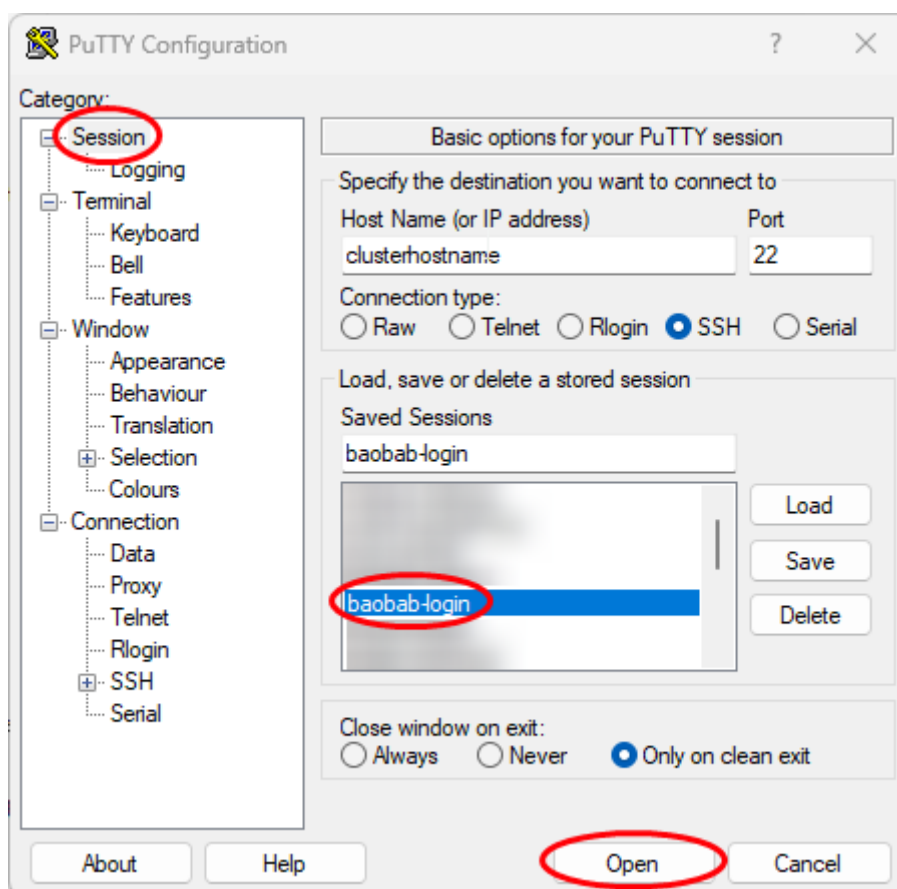
PuTTY enter username :



PuTTY save session :



PuTTY open session :



Once you open a session with PuTTY, you will be prompted for your password.

| Password:

Unlike Windows systems, Linux and Unix systems will **not display** * (star character) or any other character to indicate that you've entered something/anything in a password field - it simply stays totally blank as you type. Cursor won't blink, move, etc.

Just type your password and press enter, everything will be fine.

Access to the compute nodes

In case you need to access the compute nodes for purpose of debugging your software, using htop or other tool directly on the node, you need to go through the login node first and connect from there using ssh.

```
ssh cpu001
```

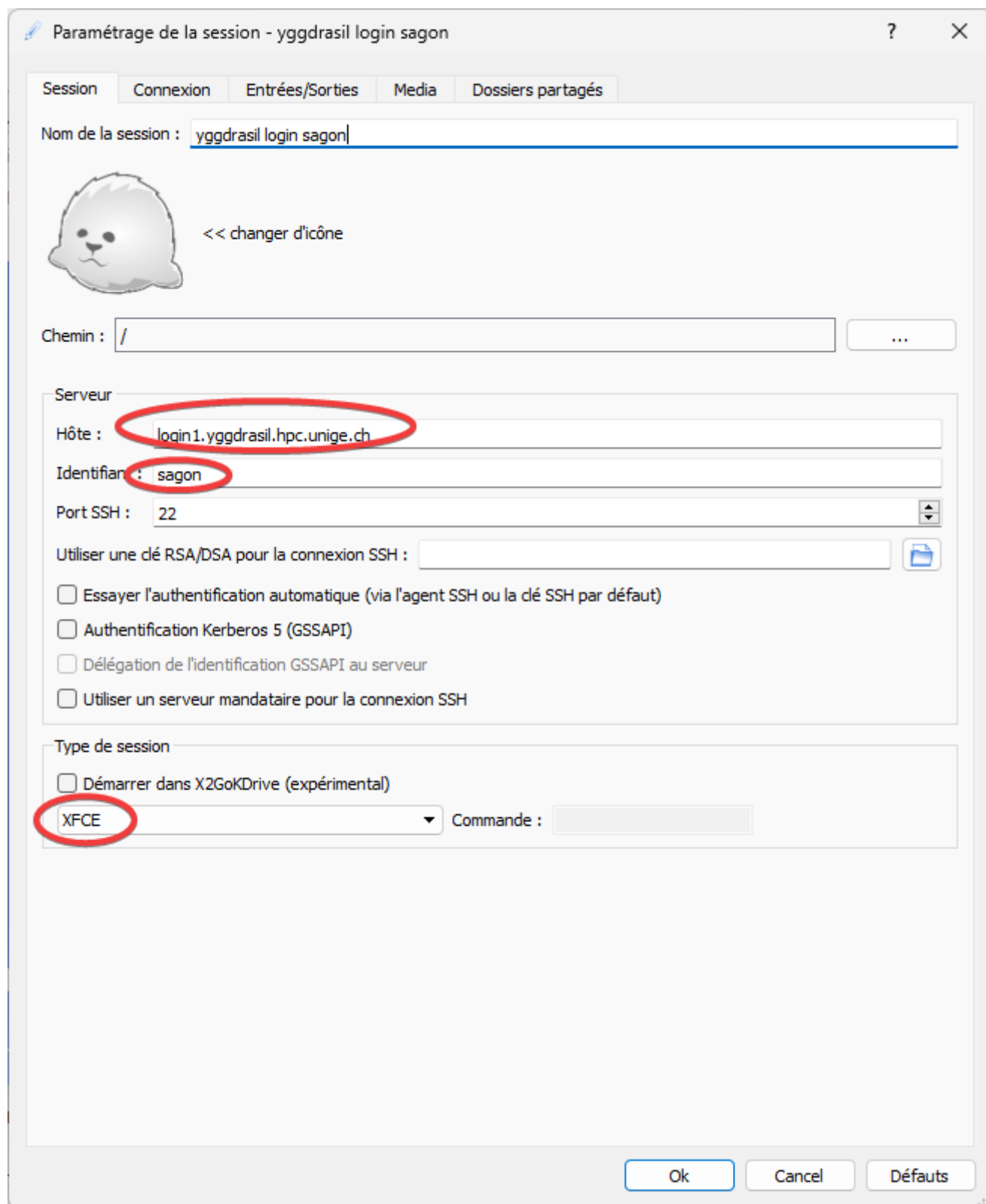
It is important to note that you **cannot** access directly the compute nodes unless you have a **RUNNING job** on it. As soon as your job is finished, you will be logged out from the compute node.

GUI access / Desktop with X2Go

It's possible to access Baobab or Yggdrasil using a graphical interface. We support only X2Go.

You need to install [X2Go client](#) on your computer and create a session following the screenshots below prior to connect to Baobab.

[X2Go create session](#) :



Once connected to the Linux desktop, you can launch lightweight (image viewer, etc.) applications directly.

If you need to launch a heavy graphical program such as Matlab for example, create a new interactive session (see the [Interactive jobs](#) section in the Slurm page) on a compute node. It is [forbidden](#) to use the login node for this purpose.

File transfer

If you need to transfer files from your computer to Baobab or Yggdrasil, you need to use the sftp protocol (scp).

From Linux

We suggest you use scp or a program like rsync

From Windows

We suggest you to use [FileZilla client](#) to transfer your files.



do not download the sponsored version (the name sponsored appears in the download link), this installer may include bundled offers that may be recognized as virus

[Filezilla create session :](#)

SSH tunnel and socks proxy

If you want running JupyterLab or VSCodeServer you may be interested by [OpenOnDemand](#)

The **login** nodes have a firewall that prevent incoming connection other than ssh.

If you need to access a service from the cluster, please follow the:

1. On your local machine, Save old ssh key and create a new one

```
$ mkdir ~/.ssh/old
$ mv ~/.ssh/* ~/.ssh/old
$ ssh-keygen
```

On the cluster, make sure you have not id_rsa key file (make a back up too)

2. Copy the rsa.pub in <https://my-account.unige.ch/main/home> (for Unige Account) <https://applicant.unige.ch/> (for Outsider Account) and wait for 5-10 min the synchronisation with AD is done. the following command on login node should print your public ssh key registered in the AD:

```
(baobab)-[alberta@login1 ~]$ /usr/bin/sss_ssh_authorizedkeys $USER
ssh-rsa [...]
```

Only for tunneling: Once the configured you MUST save this SSHpublicKey in your ssh_authorizedkeys on Cluster.

```
$ ssh-copy-id <ssh.pub> <users>@login1.<cluster>.hpc.unige.ch
```

3. On your local machine configure the proxyjump:

```
[alberta@localhost ~]$ cat .ssh/config_baobab

host baobab
    HostName login1.baobab.hpc.unige.ch
    User alberta

Host cpu*
    HostName %h
    User alberta
    ProxyJump baobab

Host gpu*
    HostName %h
    User alberta
    ProxyJump baobab

[alberta@localhost ~]$ cat .ssh/config_yggdrasil

host yggdrasil
    HostName login1.yggdrasil.hpc.unige.ch
    User alberta
Host cpu*
    HostName %h
    User alberta
```

```
ProxyJump yggdrasil
```

```
Host gpu*
  HostName %h
  User alberta
  ProxyJump yggdrasil
```

4. Alloc a test job and open a new tab on your local machine and try to connect on the allocated node:

On baobab:

```
(baobab)-[alberta@login1 ~]$ salloc --time=00:05:00
salloc: Pending job allocation 5574654
salloc: job 5574654 queued and waiting for resources
salloc: job 5574654 has been allocated resources
salloc: Granted job allocation 5574654
salloc: Waiting for resource configuration
salloc: Nodes cpu001 are ready for job
```

At the same time On your local machine, connect to the compute with selecting the right ssh config file (For this example: Baobab): (My first test was on cpu026 this is the message a i got)

```
[alberta@localhost .ssh]$ ssh -F .ssh/config_baobab cpu026
The authenticity of host 'cpu026 (<no hostip for proxy command>)' can't be
established.
RSA key fingerprint is SHA256:tKqp4nljL+EGVKl8T0VF2nS36DkHVFMPxQ0Pg/gKvg.
RSA key fingerprint is MD5:8f:75:c4:18:8a:75:f1:f1:19:4d:85:92:3b:b6:2a:e1.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'cpu026' (RSA) to the list of known hosts.
Last login: Tue Oct 24 10:49:29 2023
Installed: Thu Aug 17 14:40:08 CEST 2023
```

But working on cpu001 as well:

```
[alberta@localhost ~]$ ssh -F .ssh/config_baobab cpu001
Last login: Mon Oct 23 16:43:34 2023
Installed: Thu Aug 17 14:28:26 CEST 2023
(baobab)-[alberta@cpu001 ~]$
```

Alternative to using ProxyJump

ProxyJump doesn't work with host based authentication, this is the reason why you need to use a ssh key in the previous setup. The reason is because ProxyJump doesn't open a real ssh session and thus the ssh-keysign isn't run when you connect. Instead of using ProxyJump, you can proceed as follow:

```
Host baobab
  HostName login1.baobab.hpc.unige.ch
  User <youruser>
```

```
Host cpu*
  HostName %h
  User <youruser>
  ProxyCommand ssh -tt baobab ssh %r@%h
```

Usage: ssh cpu001

More Information on HPC-community forum:



<https://hpc-community.unige.ch/t/tutorial-ssh-tunneling-and-socks-proxy/1795>
<https://hpc-community.unige.ch/t/proxyjump-ssh-not-working-on-baobab/3078/15>

From:

<https://doc.ererearch.unige.ch/> - **eResearch Doc**

Permanent link:

https://doc.ererearch.unige.ch/hpc/access_the_hpc_clusters

Last update: **2025/07/10 14:42**

