External Server Guide

Banco de Portugal's Microdata Research Laboratory (BPLIM)

October 2025

Contents

1 Access to the External Server

1.1 Upon Access Approval

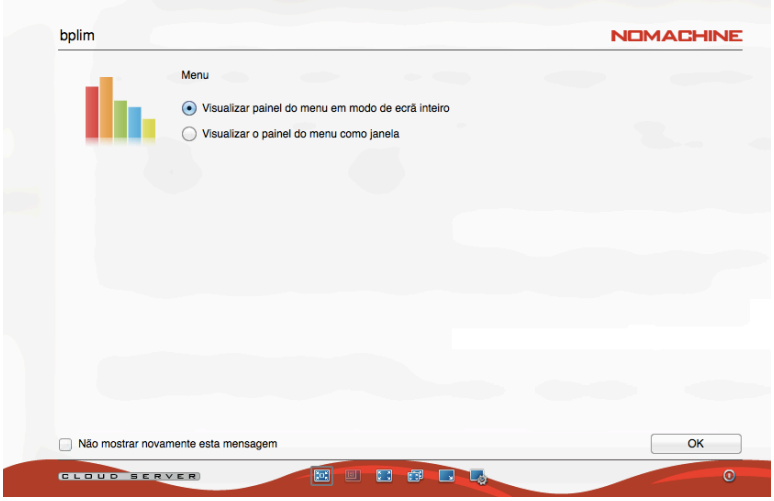
Once access is approved, you can connect to the external server using the **NoMachine** client. See Download, install and configure NoMachine client for detailed instructions.

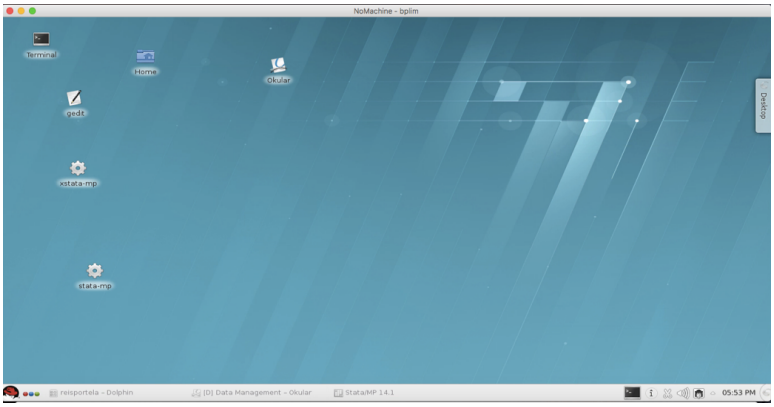
1.2 Password Policy

- The first password provided must be changed at your first login.
- Passwords expire after **60 days**. When this happens, the login window will prompt you for a new password.
- Passwords must comply with the rules described in Password requirements.

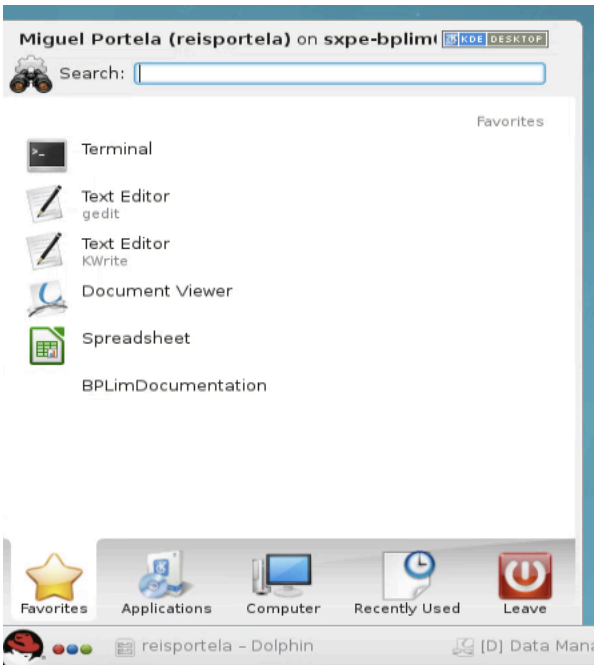
1.3 First Steps

1. When you start **NoMachine**, you will see the following three screens:



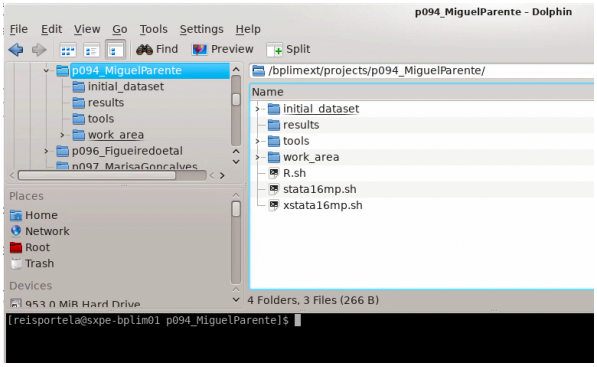


2. Select the "**Kickoff Application Launcher**" menu:



3. Then:

    1. Click on **Applications**.
    2. Select **BPLIM** and click on your project (e.g., `PXXX_name`).
    You will then see a graphical environment (the **Dolphin** file manager[1]):



You can display the command line (`Terminal`) alongside Dolphin by pressing **F4**.

4. Files with the `.sh` extension are scripts used to launch applications or enter an interactive environment.

[1] Dolphin is an intuitive and easy-to-use file manager. You can use it, for example, to browse directories or to create and delete files and folders (by right-clicking with the mouse). For more information about Dolphin, visit: https://userbase.kde.org/Dolphin.

For example, `stata_container.sh` starts the graphical version of Stata.[2]
You can run these scripts either by double-clicking them in Dolphin or by typing in the Terminal:

```
./stata_container.sh
```

5. Within your project folder, you will have access to the following directories:

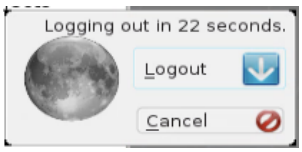
Directory	Purpose	Access
`initial_dataset`	Data sources provided by BPLIM	Read-only
`external`	Data provided by the researcher	Read-only
`intermediate`	Intermediate files	Read-only
`modified`	Modified data provided by BPLIM	Read-only
`results`	Output files generated by researchers	Read-write
`tools`	Project-specific analysis tools	Read-only
`work_area`	Temporary working directory	Read-write

Note: Your **work_area** folder also contains templates for both Stata and R. By default, these template files are read-only.

6. To reset and disconnect the remote desktop session, log out as shown below. After logging out, close the NoMachine window.[3]

[2]A container is a self-contained "box" that includes a program along with everything it needs to run - such as libraries, dependencies, and configuration settings. This ensures that the program behaves consistently across different computers, without requiring users to install or configure additional software. Containers also make it easy to install project-specific packages, which is considered good practice for ensuring transparency and reproducibility in research.
[3]Click the cross button in the upper-right corner to close.

Confirm before exiting by clicking **Logout**:[4]



- If you do not log out, your session will remain open until your next login. This may be useful to keep programs running, but note:
  - Leaving sessions open consumes server resources.
  - The recommended method for running programs overnight is **batch mode** (see discussion below).
  - If the server is rebooted for maintenance, your session will be closed and unsaved work will be lost. We strongly recommend saving your statistical programs at regular intervals.

2 Important Guidelines

2.1 Keep your home area tidy

- **Do not save files in your home area** (`/home/USER_LOGIN`).
  If you exceed its size limit, you will not be able to log in.
- Check the size of your project regularly. Open a Terminal and follow these steps:
  1. Move to the project folder:

     ```
     cd /bplimext/projects/PXXX_name/
     ```

  2. List the total project size:

     ```
     du -h
     ```

  3. Check folder sizes and list those $\geq$ 1 GB:

     ```
     du --max-depth=1 -h | sort -h | grep G
     ```

  4. Move to the `work_area` folder:

     ```
     cd work_area
     ```

  5. Repeat the size check in this folder:

     ```
     du --max-depth=1 -h | sort -h | grep G
     ```

  6. Identify duplicate or temporary files and remove them:

[4]Note that before exiting the server, you must ensure that all active programs are closed (unless they were launched in batch mode). Running programs in batch mode is appropriate for procedures that require significant computational resources, intensive calculations, and/or long processing times.

```
rm FILE_TO_DELETE
```

7. Compress large files or folders you are not currently using:
   - Compress a folder:

     ```
     tar -zcvf YOUR_FOLDER.tar.gz YOUR_FOLDER
     ```

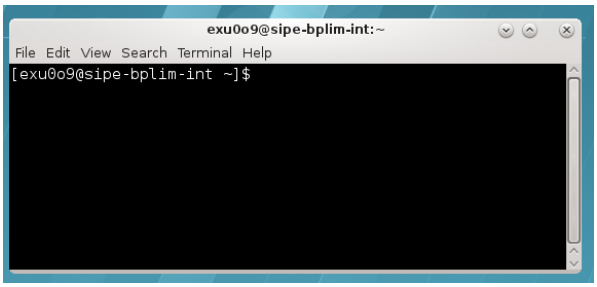
   - Compress an individual file:

     ```
     gzip YOUR_FILE
     ```

2.2 Using the Terminal

The Linux Terminal is a command-line interpreter. You can use the shell for many tasks, such as searching files and contents, organizing your workspace, and—most importantly—running programs in **batch mode**.

1. Access the Terminal from:
   Red Hat → Applications → System → Terminal



2. See Shell Commands for a list of frequently used commands.
3. If you use a non-English keyboard, the actual key mapping may differ from what you see on the screen. This mainly affects symbols.

   Example: on a Portuguese keyboard, `+` is on the `?` key, and `*` is on `SHIFT + ?`. This depends on your operating system.
4. Linux is **case-sensitive**. For example, `LS` and `ls` are different commands.
5. Use the **arrow keys** to scroll through previously entered commands.
6. Use the **Tab** key for automatic command-line completion.
7. Example: list elements within a folder in a human-readable format (`h`), long list (`l`), reverse order (`r`), sorted by modification time (`t`), including almost all files (`A`):

   ```
   ls -lArth
   ```

3 Statistical Software

3.1 Stata

3.1.1 Running Stata

Stata runs inside a container. You will find a launcher script named **`stata_container.sh`** in your project folder.

You can start Stata in any of the following ways:

- **Using the file manager** (Dolphin): double-click the `stata_container.sh` file to launch Stata.
- **Using the Terminal**:

  1. Open a Terminal in the project folder.
  2. Run:

  ```
  ./stata_container.sh
  ```

- **Manually opening the container**:

The project's container image is stored in the `tools/_container` directory. You can enter the container environment by running:

```
cd /bplimext/projects/PXXX_name
singularity shell tools/_container/CONTAINER_ID.sif
```

Then start Stata inside the container:

```
xstata-mp
```

Stata can be run in **non-graphical** modes

```
stata-mp
```

3.1.2 Ado-files

Ado-files are text files containing Stata programs. It is advisable to create and save your ado-files so results can be replicated later when running them on BPLIM datasets.

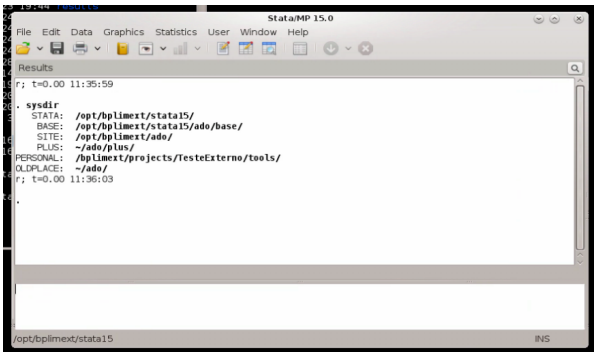
Stata looks for ado-files in several locations, typically organized as:

- **SITE** – system-wide ado-files
- **PLUS** – user-installed ado-files
- **PERSONAL** – user-created ado-files
- **OLDPLACE** – legacy location for ado-files

Ado-files created for your project can be found in the current directory (`.`). Specific ado-files requested from BPLIM will be placed in `/bplimext/projects/PXXX_name/tools`. To make sure Stata recognizes this directory, add the following line at the beginning of your `.do` file:

```
adopath + "/bplimext/projects/PXXX_name/tools"
```

The `sysdir` command within Stata will list all directories currently in use:



3.1.3 Temporary files

To manage Stata's temporary files:

1. Check the current temporary folder:

```
tempfile junk
display "`junk'"
```

2. It should display something like "/tmp/St98278.000001"
3. In case the temporary file is not in the path "/tmp", exit Stata and edit your `.bashrc` in your home directory (`cd ~`) with `kwrite` or `vi` and add:

```
export STATATMP="/tmp"
```

4. Apply the changes:

```
source .bashrc
```

5. Start a new Stata session (inside the container).

3.1.4 'batch' mode: an example using Stata

1. Open a **shell** in Linux and navigate to the directory containing the do-file you want to run (e.g., `prog1.do`):

```
cd /bplimext/projects/PXXX_name/work_area/
```

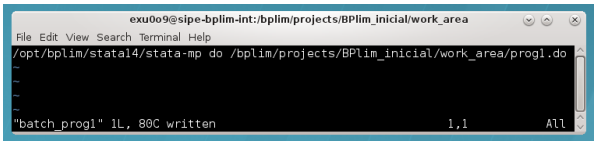
2. You may find it easier to use **Dolphin** (the File Manager) to browse your folder structure. In Dolphin, press **F4** to open an integrated Terminal. This allows quick navigation between folders and the ability to run shell commands in the same window.
3. Create a plain text file (ASCII) named, for example, `batch_prog1`.

4. Inside this file, write the execution command you would normally type in the shell. For example:

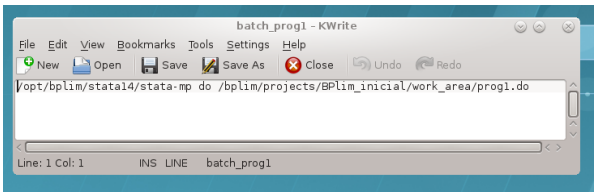
```
stata-mp do /bplimext/projects/PXXX_name/work_area/prog1.do
```

5. To create the batch file, you can use any text editor. For instance, with the command-line editor `vi`:

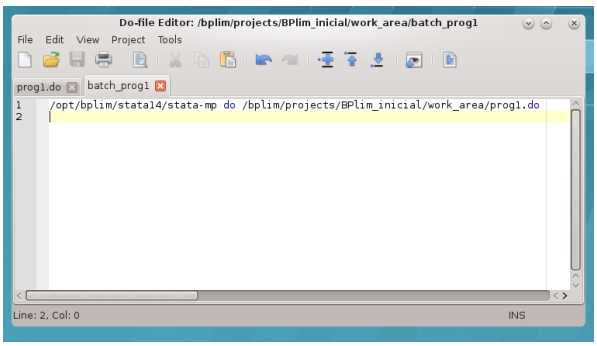
```
vi batch_prog1
```



6. The batch file can also be created using graphical editors such as **KWrite** or the Stata **Do-file Editor**:



or



7. You may add the extension `.txt` to the batch file name. Sometimes the Stata Do-file Editor does not recognize files without an extension (e.g., `batch`), but it does recognize `batch.txt`.

8. Once the batch file is created, make sure you are in the container environment and run the `.do` file in batch mode by typing in the Terminal:

```
at now -f batch_prog1
```

9. To explore more options for `at`, type `man at`. For example:

```
at now + 5 hours -f batch_prog1
```

or

```
at now + 4 minutes -f batch_prog1
```

This runs the Stata program 5 hours or 4 minutes from now, respectively. (`man` displays the Linux help manual.)

10. Type `top` in the Terminal to confirm the program is running.
11. Inside `top`, press `i` to hide irrelevant processes and reduce the output shown.
12. To terminate a running process in `top`:
    - Press `k` (for kill).
    - Enter the process number (PID, shown in the first column).
    - Then type `9` to force termination.
13. To exit `top`, press `q`.

Useful features of the `at` command

- `atq` — lists programs in the batch queue (`=` indicates the program is running, `a` indicates it is queued along with its scheduled execution time).
- `atrm <job_number>` — removes a program from the batch queue.
- Monitor progress by checking the log file with `tail`:

```
tail --f logcrc_may21.log
```

This continuously updates the last lines of the log without overwriting it.

3.1.5 Running programs in the background with `screen`

- `screen` is useful if you want to run Stata interactively and ensure the session is preserved even if your network connection drops. You can disconnect from NoMachine and later recover the session by typing:

```
screen --r
```

- Multiple `screen` sessions can run simultaneously. After reconnecting with NoMachine, list the running sessions:

```
screen -d
```

Then recover a specific session by typing:

```
screen -r <pid>
```

(replace with the actual process ID).

3.2 R

3.2.1 'interactive' mode

R is used within a container environment. You will find a launcher script named `r_container.sh` in the project folder. You can start R in the following ways:

- **Using the file manager** (Dolphin): double-click the `r_container.sh` file to launch RStudio inside the container.
- **Using the Terminal**:

  1. Open a Terminal in the project folder.
  2. Run:

  ```
  ./r_container.sh
  ```

Manually opening the container:

The container is in the `tools/_container` directory. Open a Terminal and type:

```
cd /bplimext/projects/PXXX_name
singularity shell tools/_container/CONTAINER_ID.sif
```

Once inside the container, then start RStudio

```
rstudio
```

IMPORTANT: Do **not** save your workspace image in your home folder when prompted (`Save workspace image? [y/n/c]`). If you need to keep the workspace, save it inside your project folder under `work_area`.

3.2.2 'batch' mode

Running R scripts in batch mode is recommended for long-running programs or computationally intensive tasks.

1. Open a **shell** in Linux and navigate to the directory containing the R script you want to run (e.g., `analysis.R`):

```
cd /bplimext/projects/PXXX_name/work_area/
```

2. You may find it easier to use **Dolphin** (the File Manager) to browse your folder structure. In Dolphin, press **F4** to open an integrated Terminal. This allows quick navigation between folders and the ability to run shell commands in the same window.
3. Create a plain text file (ASCII) named, for example, `batch_r_analysis`.
4. Inside this file, write the execution command you would normally type in the shell. For example:

```
Rscript /bplimext/projects/PXXX_name/work_area/analysis.R \
  > /bplimext/projects/PXXX_name/work_area/analysis.log 2>&1
```

The `> analysis.log 2>&1` redirects both standard output and error messages to a log file.

5. To create the batch file, you can use any text editor. For instance, with the command-line editor `vi`:

```
vi batch_r_analysis
```

6. The batch file can also be created using graphical editors such as **KWrite**:

```
kwrite batch_r_analysis
```

7. **Start the container environment** before creating the batch file, since R must run inside the container:

```
singularity shell ../tools/_container/CONTAINER_ID.sif
```

8. Once the batch file is created and you are inside the container environment, run the R script in batch mode by typing in the Terminal:

```
at now -f batch_r_analysis
```

9. To explore more options for `at`, type `man at`. For example:

```
at now + 5 hours -f batch_r_analysis
```

or

```
at now + 4 minutes -f batch_r_analysis
```

This runs the R program 5 hours or 4 minutes from now, respectively.

10. Type `top` in the Terminal to confirm the program is running. Look for processes named `Rscript` or `R`.
11. Inside `top`, press `i` to hide irrelevant processes and reduce the output shown.
12. To terminate a running process in `top`:
    - Press `k` (for kill).
    - Enter the process number (PID, shown in the first column).
    - Then type `9` to force termination.
13. To exit `top`, press `q`.

Useful features of the `at` command

- `atq` — lists programs in the batch queue (`=` indicates the program is running, `a` indicates it is queued along with its scheduled execution time).
- `atrm <job_number>` — removes a program from the batch queue.
- Monitor progress by checking the log file with `tail`:

```
tail -f analysis.log
```

This continuously updates the last lines of the log without overwriting it. Press `CTRL + C` to stop monitoring.

3.2.3 Running R programs in the background with `screen`

Similar to Stata, `screen` is useful for running R interactively while preserving the session if your network connection drops:

```
screen -S r_session
singularity shell tools/_container/CONTAINER_ID.sif
R
```

To detach from the screen session, press `CTRL + A`, then `D`.

To reattach to your R session:

```
screen -r r_session
```

List all running screen sessions:

```
screen -ls
```

3.3 Python

You will find a launcher script named **`python_container.sh`** in the project root. You can start the Python container by double-clicking this script in **Dolphin**.

Alternatively, start the container from the **Terminal**:

```
./python_container.sh
```

You can also start the container manually:

```
cd /bplimext/projects/PXXX_name
singularity shell tools/_container/CONTAINER_ID.sif
```

Once inside the container, you can use Python directly or launch a Jupyter Notebook:

```
jupyter notebook
```

A Jupyter Notebook will open in **Firefox**. Click New and select the **Python** kernel.

3.4 Julia

You will also find a launcher script named **`julia_container.sh`** in the project root. You can start Julia by double-clicking this script in **Dolphin**.

Alternatively, start Julia from the **Terminal**:

```
cd /bplimext/projects/PXXX_name
./julia_container.sh
```

You can also start the container manually:

```
cd /bplimext/projects/PXXX_name
singularity shell tools/_container/CONTAINER_ID.sif
```

Once inside the container, you can launch the Julia REPL or Jupyter:

```
# Julia REPL
julia

# or Jupyter Notebook
jupyter notebook
```

A Jupyter Notebook will open in **Firefox**. Click New and select the **Julia** kernel.

Alternatively, you can use **VS Code** to run Julia. Open VS Code from the terminal (in your project folder):

```
code
```

3.5 Updates to Commands and Packages

Requests for additional commands or packages, as well as updates to existing ones, must be submitted to the **BPLIM Team**.

3.6 Build a Container to Fine-Tune Your Statistical Packages

The server uses **Apptainer (formerly Singularity)** containers. To request one, please send the BPLIM Team the **definition file**. We will build the image and place it in your project's `work_area`. Detailed information about Apptainer/Singularity containers is available at https://sylabs.io/.[5] Additional notes are provided in the Appendix.

4 Allowed Outputs

Results can be exported to disk in the following formats (see the *Output Control Guide*):

1. **ASCII files** — e.g., log files
2. **Graphs** — export as `.png`
3. **CSV** — Comma-Separated Values, for use with MS Excel or similar
4. **TEX** — LaTeX format for integration into TeX documents

5 Requesting Outputs

Output files (e.g., log files, images) must be requested from the **BPLIM Team** at **bplim@bportugal.pt**. Researchers are not allowed to place or remove files on the server independently. All outputs must comply with the output control rules.

After validation, the requested results will be sent to you by email. The extraction process depends on whether you are working with modified data.

[5] Singularity is now called Apptainer. You can find more information here: https://apptainer.org.

5.1 Projects using modified data

1. Run the replication app successfully before requesting outputs. See the Replication App manual for detailed instructions.
2. Send an email to **bplim@bportugal.pt** with the subject line:

   PXXX_name: request replication

5.2 Projects NOT using modified data

1. Place all outputs you wish to extract in the `results` folder.[6]
2. Send an email to **bplim@bportugal.pt** with the subject line:

   PXXX_name: request for result extraction

6 User's Home Folder

1. Do **not** save files in your home folder: `/home/USER_ID/`
2. Regularly empty your **Trash** folder. If your disk usage exceeds the quota, you will not be able to log in. To clean the Trash via Terminal, type:

```
rm -rf ~/.local/share/Trash/*
```

7 Project Archival Policy

Projects that remain inactive for more than **two (2) years** will be archived.
Archived projects will no longer be directly accessible but can be reactivated upon request to the **BPLIM Team**.

8 Appendix

8.1 Basic Shell Commands on Linux

- `top`: List processes currently running on the server
  - Press `i` to hide background processes.
  - Press `h` to display the **help menu** for available options.
- `pwd`: Show the current working directory
- `cd`: Change directory

[6] You may only remove text files that do not contain data or any information that could allow identification. For every graph you request as an output, you must also provide the corresponding table required to replicate it. Graphs may only be exported in .PNG format - vector graphics are not permitted.

```
cd /bplimext/projects/PXXX_name/work_area/
```

- `cd ~`: Move to your home folder
- `cp`: Copy file(s) to a given path

```
cp prog1.do /bplimext/projects/PXXX_name/results
```

- `mv`: Move file(s) or rename file(s)

```
mv prog1.do /bplimext/projects/PXXX_name/results
```

- `rm`: Delete a file

```
rm /bplimext/projects/PXXX_name/results/prog1.do
```

- `mkdir`: Create a directory

```
mkdir programs
```

- `rmdir`: Delete an empty directory

```
rmdir programs
```

- `screen`: Start a session manager that allows running programs in the background and resuming them later

```
screen top
```

- `man`: Show the manual page for a given command

```
man ls
```

- `du -h`: Display disk usage of files and directories in human-readable format

```
du /bplimext/projects/PXXX_name/work_area/
```

- `df -h`: Show disk space utilization in human-readable format
- `vi`: View or edit ASCII text files (e.g., `.do` files, logs)
- `ghostscript`: Preview files with `.eps` or `.pdf` extensions

```
ghostscript /bplimext/projects/PXXX_name/results/file_name.pdf
```

- `okular`: View PDF files
- `find`: Search for files
  - Basic structure: `find /path options pattern`

```
find . -name "*.do"
```

  - Save search results to a file:

```
find . -name "*.do" > find_results.txt
```

  - Search for a string within filenames:

```
find . -name "*.do" | grep "analysis"
```

– Identify `.do` files containing the word `graph export`:

```
find . -name "*.do" -exec grep "graph export" '{}' \; -print
```

- `passwd`: Change your password
- **Exit a program**: Press `CTRL + C` to terminate the current process in the shell

8.2 Using the `vi` File Editor

1. Open a file in `vi` from the shell, for example:

   ```
   vi batch1.txt
   ```

2. Common shortcut keys in `vi`:

   - `i`: Insert text
   - `ESC`: Exit insert mode
   - `x`: Delete the character under the cursor
   - `dd`: Delete the current line
   - `10 dd`: Delete 10 lines
   - `yy`: Copy (yank) the current line
   - `p`: Paste the copied (yanked) text
   - `SHIFT + G`: Go to the last line
   - `gg`: Go to the first line
   - `ESC + :q!`: Quit without saving changes
   - `ESC + :w!`: Write (save) and overwrite the file
   - `ESC + :q`: Quit if no changes have been made

   For a more complete guide, see: https://www.cs.colostate.edu/helpdocs/vi.html

3. Easier alternative: use the `gedit` text editor for a graphical interface:

   ```
   gedit batch1.txt
   ```

8.3 Password Requirements

Rule	Value	Notes
Max. Password Lifetime	60 days	After 60 days the password will expire and must be changed at the next login. The password can be changed at any time using: (1) Red Hat icon → Applications → Settings → System Settings → Account Details, click *Change Password*; or (2) in the Shell type `passwd`.
Min. Character Classes	4	You should include at least 4 classes of characters in the password. For example: small letters, capital letters, numbers and punctuation marks. There are a total of five classes: - **A–Z** (capitals) - **a–z** (lowercase) - **0–9** (numbers) - **punctuation**: ␣ ! % & ( ) * + . , { } [ ] ~ " # $ ' - / \ ^ _ \ |‘ - **chars above 127**: (ã, á, ä, à, etc.; @, £, §, º, ª, «, »). **Note:** Using the same character 3+ times may require an additional class. Recommended: don’t repeat the same character more than twice consecutively.
Min. Length	8	The minimum size of the password is 8 characters (it may be higher if you repeat characters).

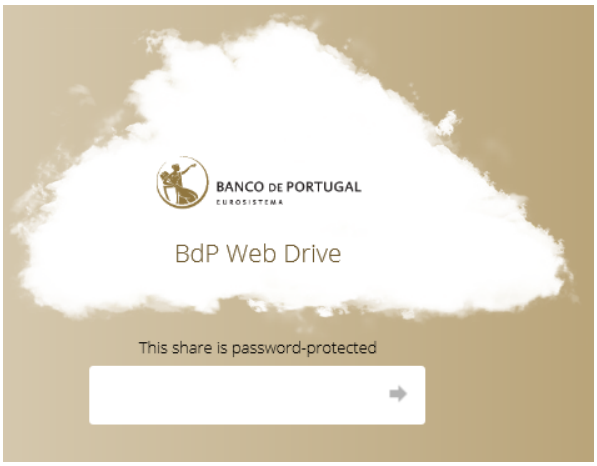
Rule	Value	Notes
Password History	7	You cannot reuse a password from the previous 7 passwords.
Max. Failures	6	If the user fails 6 consecutive times, the account will be locked for the time defined in *Lockout Time*.
Fail Interval	60 sec	Time interval to count attempts as consecutive. If more than 60 seconds have elapsed since the last attempt, the failure count resets to 1.
Lockout Time	600 sec	Time (10 minutes) during which the account will be locked if the maximum number of failed attempts is reached.

8.4 Download, Install and Configure NoMachine Client

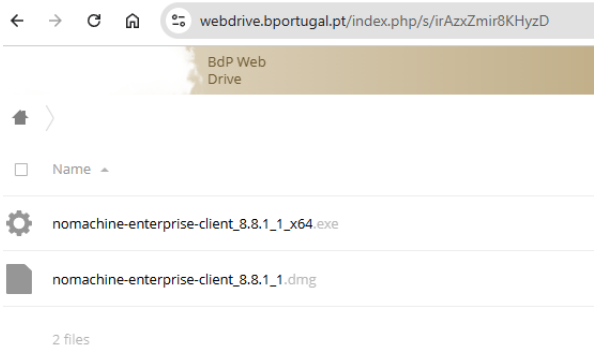
Step 1: Go to the following link and use the credentials provided by BPLIM to access the site:

Banco de Portugal Webdrive

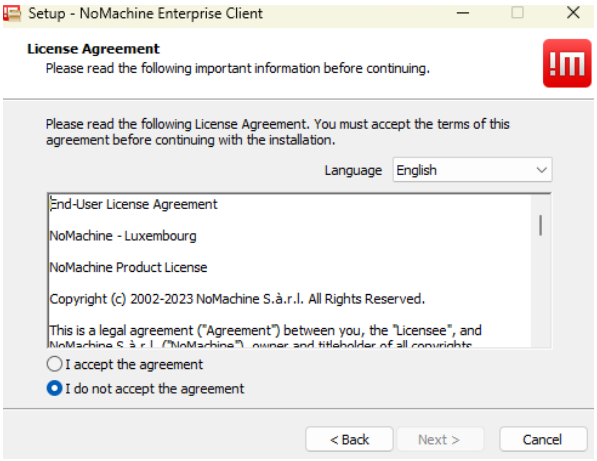
Note: sometimes the internet provider, *e.g.*, a University, may block access to this particular website. Please check with your provider in case you get an error while trying to use the link.

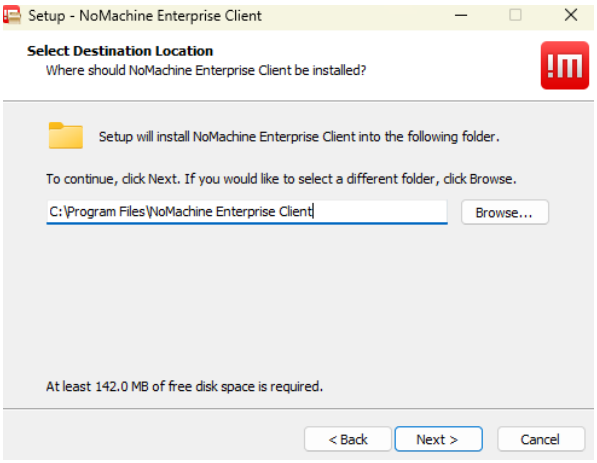


Step 2: Download the file with an extension compatible with your OS (Operating System).



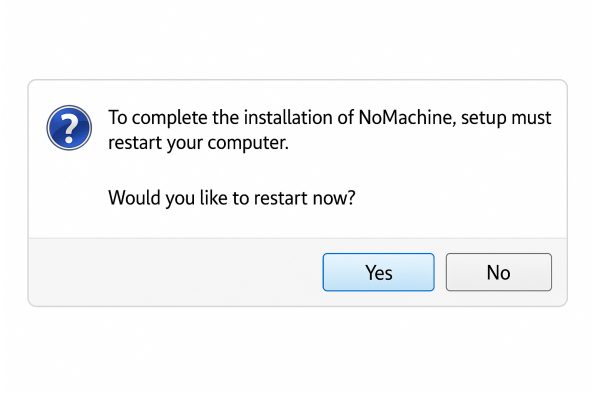
Step 3: Install 'NoMachine'.


Setup - NoMachine Enterprise Client
License Agreement
Please read the following important information before continuing.
Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.
Language English
End-User License Agreement
NoMachine - Luxembourg
NoMachine Product License
Copyright (c) 2002-2023 NoMachine S.à.r.l. All Rights Reserved.
This is a legal agreement ("Agreement") between you, the "Licensee", and
I accept the agreement
I do not accept the agreement
< Back
Next >
Cancel


Setup - NoMachine Enterprise Client
Select Destination Location
Where should NoMachine Enterprise Client be installed?
Setup will install NoMachine Enterprise Client into the following folder.
To continue, click Next. If you would like to select a different folder, click Browse.
C:\Program Files\NoMachine Enterprise Client
Browse...
At least 142.0 MB of free disk space is required.
< Back
Next >
Cancel

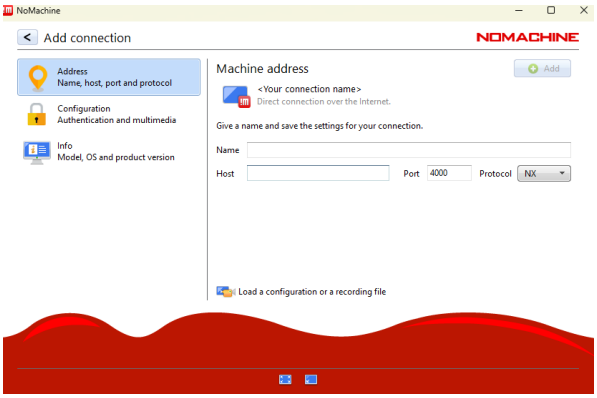

Setup - NoMachine Enterprise Client
NOMACHINE
ENTERPRISE CLIENT
Completing the NoMachine Enterprise Client Setup Wizard
Setup has finished installing NoMachine Enterprise Client on your computer.
Click Finish to exit Setup.
The network computing company
Finish

Step 4: Reboot your computer

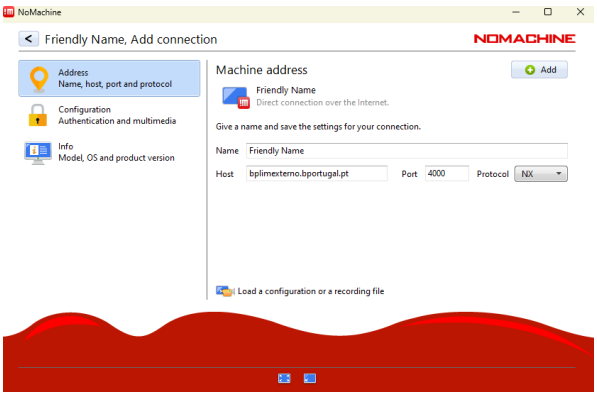


Step 5: NoMachine client access configuration.

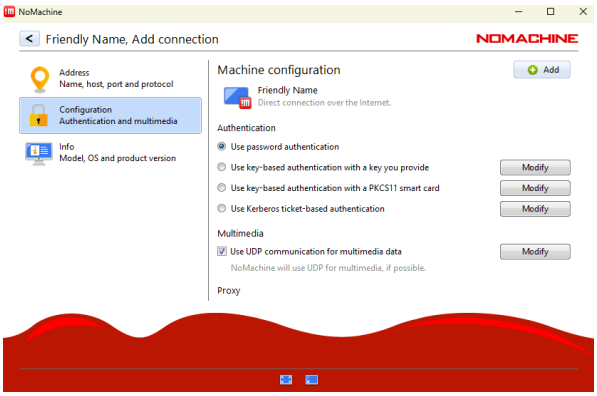
Step 5.1: Start 'NoMachine' and create a new connection.



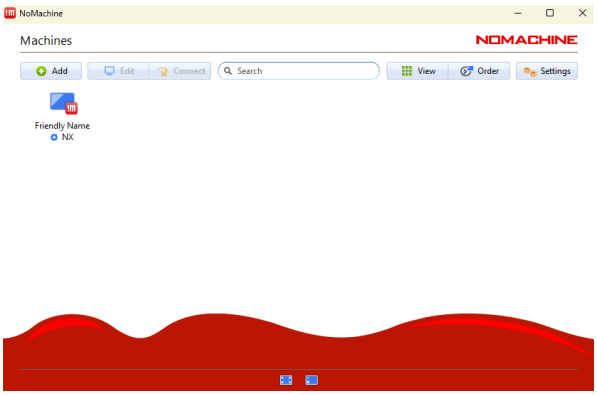
Step 5.2: Define the 'Host' as bplimexterno.bportugal.pt, 'Port' 4000, 'Protocol' NX and set a 'Friendly Name' for 'Name'.



Step 5.3: Use password authentication - with or without a proxy - according to the instructions provided by your network administrator or IT support. The proxy settings can be customized under Proxy in the bottom-right corner. After completing the configurations, click Add to create the connection.

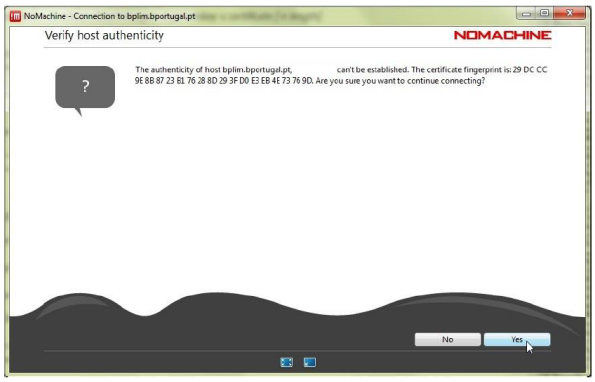


Step 5.4: Once the entry for `bplimexterno.bportugal.pt` has been created, connect:

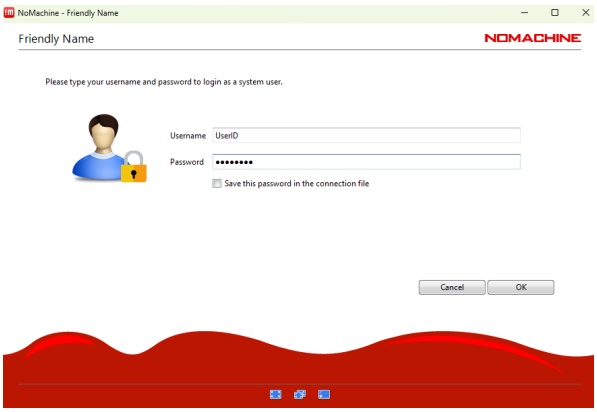


Step 5.5: Before the first effective connection, it may be necessary to accept the certificate from bplimexterno.bportugal.pt. You should verify that the "fingerprint" (verification code) is:

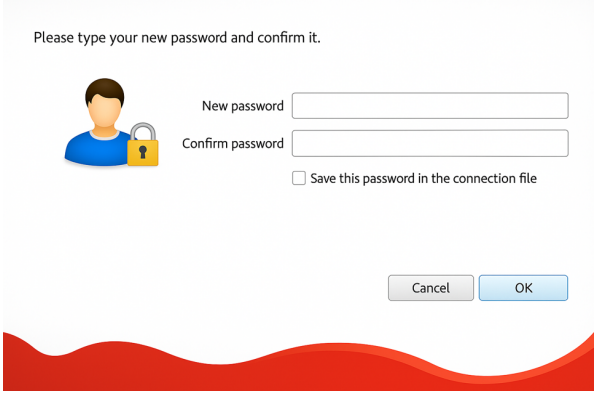
SHA256 ED 1B D9 E2 C2 F8 C6 08 1A 53 5F 97 DA 71 77 D9 D2 EE 7A 5F 9C 35 87 B3 19 F4 7E A1 CB 2C 68 0B



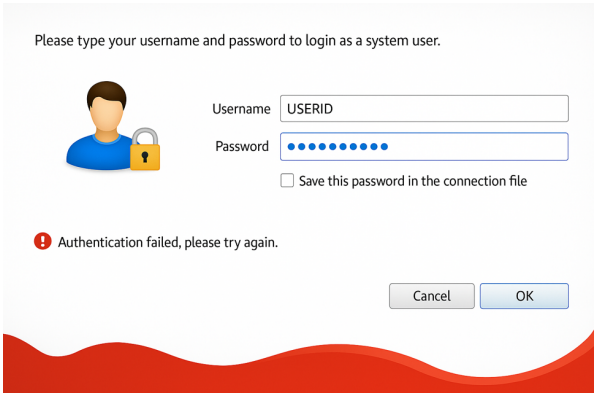
Step 5.6: Connect with the UserID (**case sensitive**) and password provided by Banco de Portugal:



Step 5.7: After the first successful login, it is necessary to change the password, which must comply with the Password Policy defined Section 1.2.

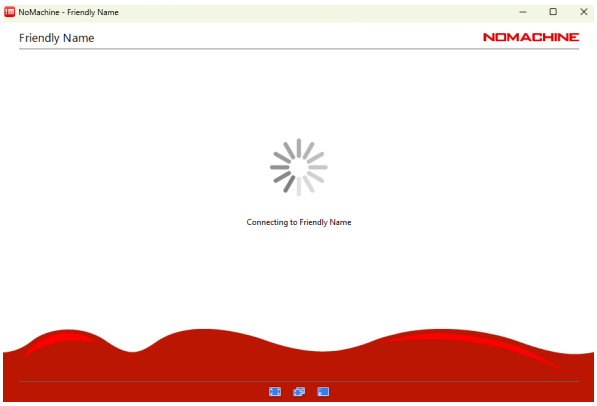


If the new password does not comply with the Password Policy, the original password provided by the Banco de Portugal will be re-requested. You get the message "Authentication failed, please try again." See Appendix 3 for details.

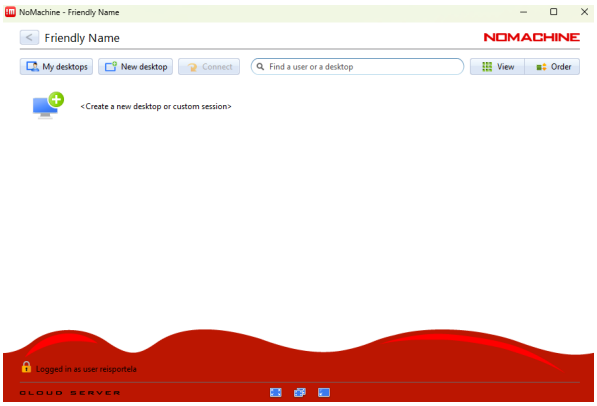


The NoMachine client does not tell you why the new password was not accepted – it is the responsibility of the user to verify that the new password is in compliance.

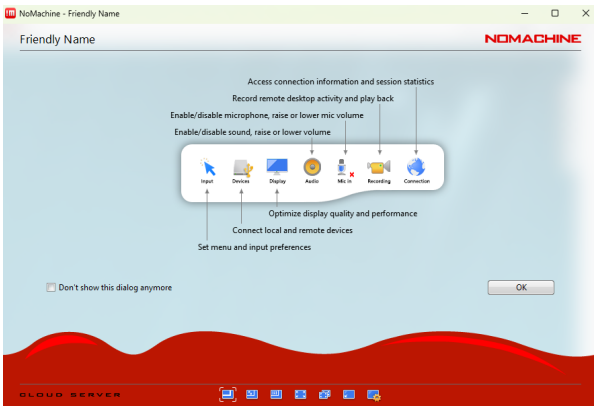
Step 5.8: Upon login success, the following screens should appear.

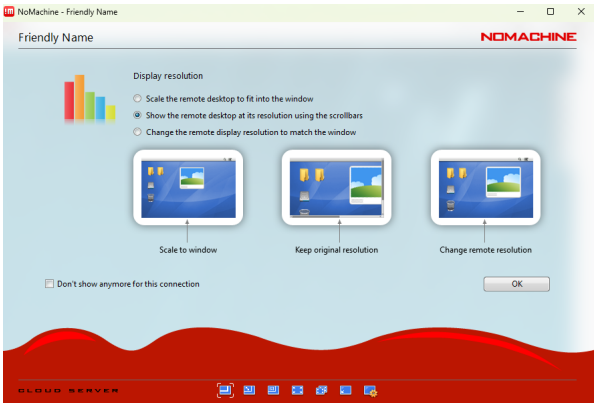


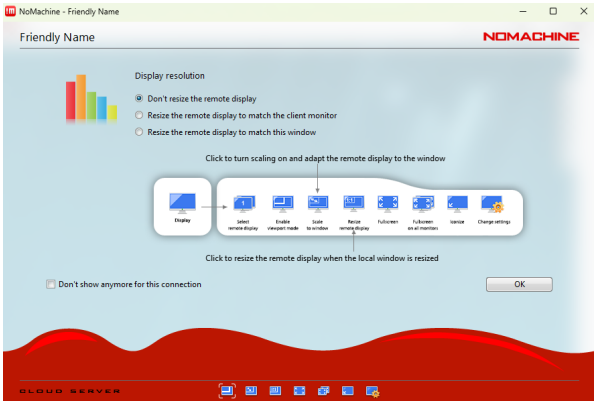
Create a new desktop.



Step 5.9: In the following screen define the settings of your monitor.

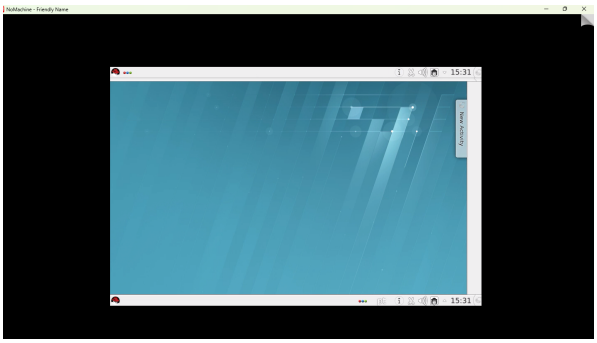




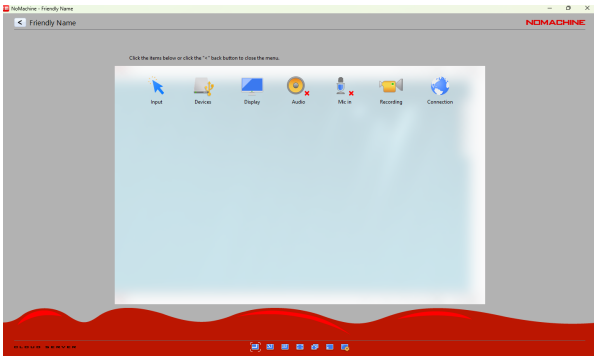


Step 5.10: Upon login success, the following screens should appear.

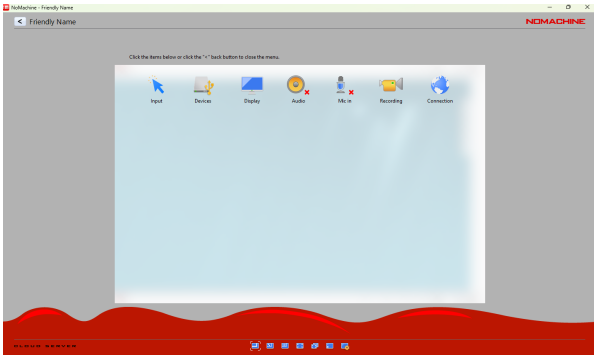
Once logged in and with access to a KDE session, click on the upper right corner of the KDE desktop, as shown below, to access the menu and then expand the screen as exemplified for greater ease of use.



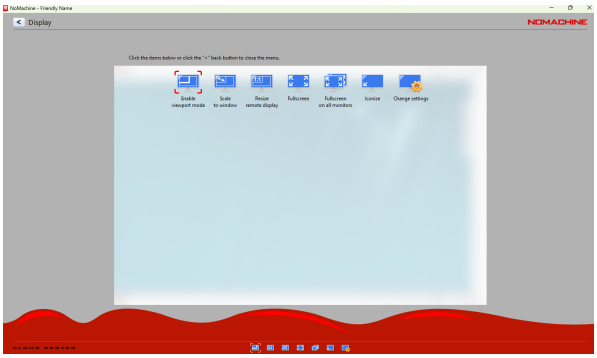
Step 5.11: You should see the following screen.



Step 5.12: Click ‘Display’.

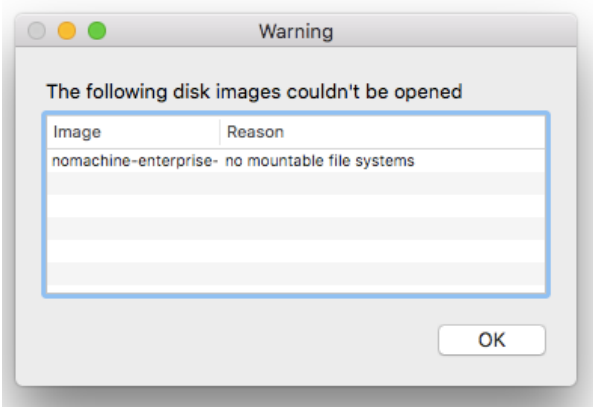


Step 5.13: Choose the option that best fits your monitor.



8.5 Frequently Asked Questions

1. **Mac users cannot install NoMachine and receive the error below:**



- Ensure your **macOS** is up to date.
- As a temporary solution, download the **NoMachine Enterprise Client** from the official website and run the installation file.

  NoMachine Client

2. **NoMachine authentication failure**

- This may happen due to a mismatched keyboard layout.
  For example, if you use a **Portuguese keyboard** but the system assumes a **US keyboard**, a password containing ç may be rejected as "wrong password."
  Verify your keyboard layout or change your password after the first login using:

  ```
  passwd
  ```

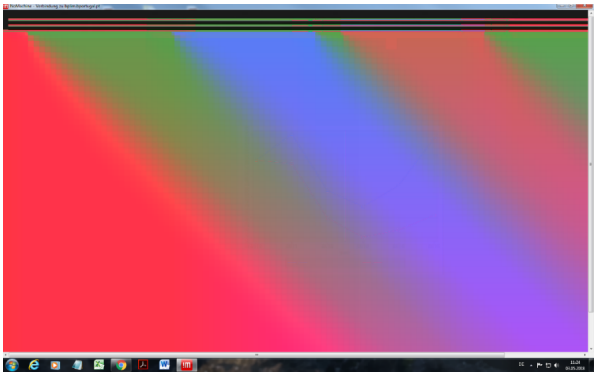
- If login fails with the error:
  "Could not connect to the server. Error is 138: Connection is timed out"
  check whether your network firewall is blocking the connection.
  Some university networks block external connections to BPLIM's server.
  Test from another location (e.g., your home network).

3. **User pressed 'Lock' instead of 'Log out' and cannot unlock**

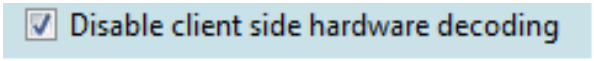
- Check that the keyboard layout is correct (e.g., PT or UK).

- Close the NoMachine session and start a new one. Before the final **Login** step, right-click and choose **Logout**, then double-click to reconnect.

4. **"Cannot see the screen in NoMachine"**

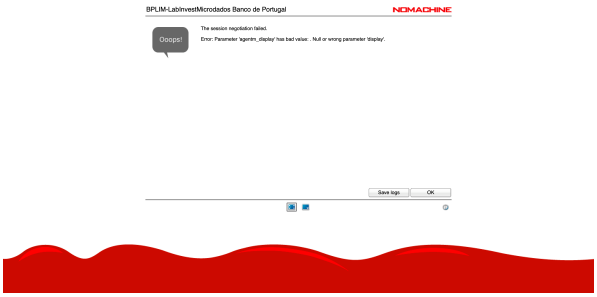


- **Option A**: Move your mouse to the top-right corner of NoMachine.
  A folded-sheet icon will appear. Left-click → **Display** → **Change settings** → enable **Disable client-side hardware decoding**.



- **Option B**: Close the NoMachine connection and start a new one. Before the final **Login** step, right-click and choose **Logout**, then double-click to reconnect.

5. **"Error: Parameter 'agentm_display' has bad value"**



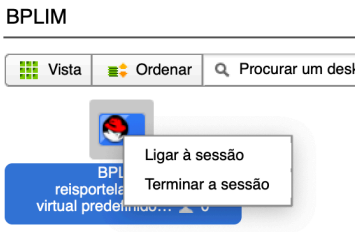
- This usually means your home folder is full (`/home/USER_LOGIN`).
  Do not save files in your home folder.

- Ask the BPLIM Team to free up space in your home directory.

6. **Session is frozen**

- From the first NoMachine screen, double-click the following icon:

- Then right-click the icon below and choose **Terminar sessão**:



7. **Visualizing LaTeX tables**

   If you want to preview a table exported to LaTeX as a PDF, create a simple file named `main.tex`:

   ```
   \documentclass{article}
   \begin{document}
     \input{your_table.tex}
   \end{document}
   ```

   Replace `your_table.tex` with the name of your table file. Compile it in the Terminal with:

   ```
   pdflatex main.tex
   ```

   This generates `main.pdf`, which you can view with:

   ```
   okular main.pdf
   ```

8.6 Version Control

The server runs GitLab.
If you need Git for your projects, please request access from the **BPLIM Team** (bplim@bportugal.pt).

From Wikipedia:

> *"Git is a distributed version-control system for tracking changes in any set of files, originally designed for coordinating work among programmers cooperating on source code during software development. Its goals include speed, data integrity, and support for distributed, non-linear workflows."*

First Steps

1. **Generate an SSH key**
   Open a **Terminal** in your home folder:

```
cd ~
```

Then type:

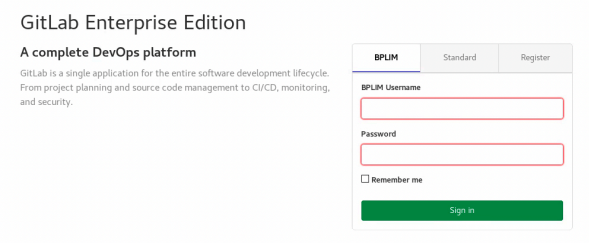
```
ssh-keygen -t rsa -C "BPLIM git"
cat ~/.ssh/id_rsa.pub
```

2. **Copy your SSH key**

Highlight the generated key in the terminal, right-click, and choose **Copy** to copy it to your clipboard.

3. **Access GitLab**

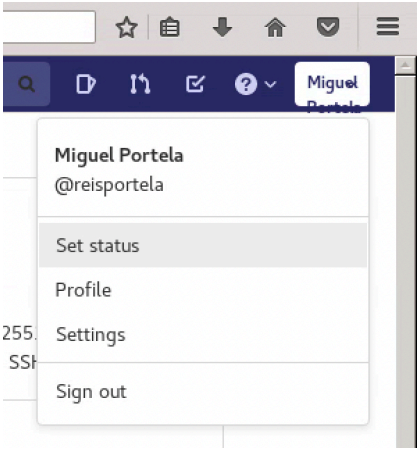
Open **Firefox** (Red Hat → Search → Firefox) and navigate to: https://vxpp-bplimgit.bplim.local/



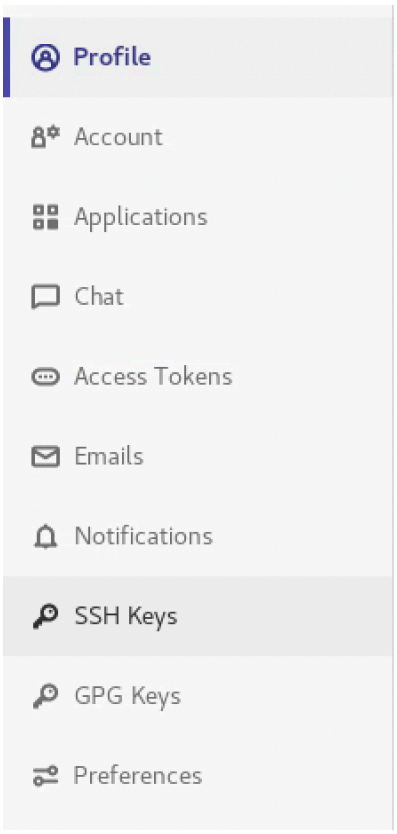
Log in with your external server credentials.

4. **Add your SSH key in GitLab**

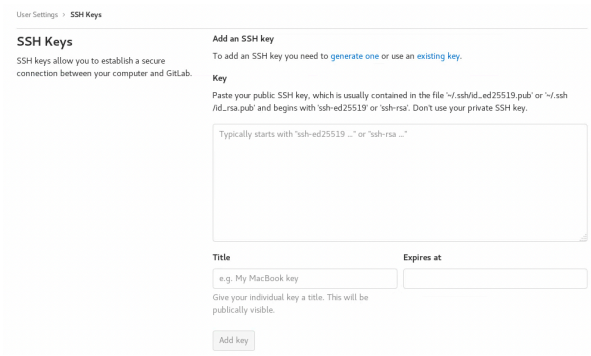
- Navigate to your profile (**Settings** in the top-right corner).



- In the left sidebar, click **SSH Keys**.



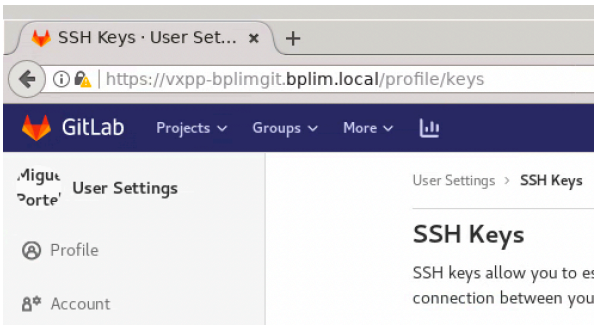
- Paste the contents of the clipboard in the text box on the top right corner under **Key**.

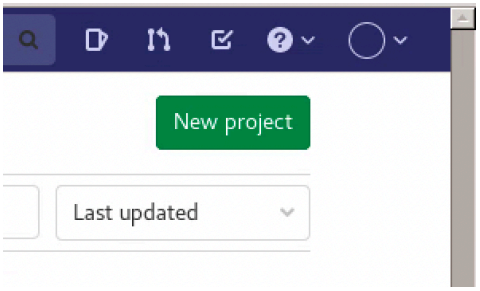


- Give a title, e.g., "BPLIM git", and click in **Add key**.

5. **Create a new project**

Go to **Projects** → **New project**, e.g., `scripts_P999` (replace `P999` with your project ID).





Configure Git by editing/creating the `.gitconfig` file in your home folder. You can use KWrite (Red Hat → Search → KWrite). Example:

```
[cola]
        spellcheck = false
[user]
        name = Investigador A
        email = investa@sxpe-bplim01.bplim.local
[gui]
        editor = kwrite
```

6. **Clone the project**

In the Terminal, move to your `work_area` and clone the repository:

```
cd /bplimext/projects/your_project_ID/work_area/
git clone git@vxpp-bplimgit.bplim.local:investa/scripts_P999.git
```

7. **Add** the file `.gitignore` available in folder `tools` of your project:

```
cd scripts_P999
cp /bplimext/projects/your_project_ID/tools/.gitignore .
```

8. **First commit & push**

```
git add *
git commit -a -m "First"
git push
```

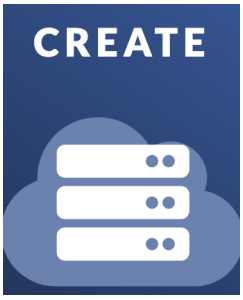
9. **Best practice**

Place all your scripts and code files in the `scripts_P999` folder. This ensures a structured and efficient workflow with version control.

8.7 Containers

8.7.1 Build Your Container

- Create a container definition using the template files available in our GitHub repository.
- Test your script and build the container using SylabsCloud (you can sign in with your GitHub account).
- Click **CREATE**:



- In the next step, upload your `.def` file or copy/paste its contents into the text box:

- Sylabs will validate your script. Once successful, the **Build** button will become available. Click it to start the build process.
- Monitor the build process at the bottom of the screen and check for any error messages.
- After the container is built successfully, send the **updated definition file** to the BPLIM Team.

8.7.2 Use the container in BPLIM's server

1. Open a **Terminal**.
2. Navigate to your project's container folder:

```
cd /bplimext/projects/PXXX_name/tools/containers
```

3. Start the container

```
singularity shell YOURPROJECTID.sif
```

4. The Terminal prompt will change to indicate that you are inside the container, showing: `Singularity`
5. Launch RStudio by typing `rstudio` (small caps)

6. Once inside RStudio, you will have access to the original folder structure of your project.

8.8 Jupyter Lab

Explore Jupyter lab:

> *"JupyterLab is a web-based interactive development environment for Jupyter notebooks, code, and data.*
> *It is flexible, allowing you to configure and arrange the interface to support diverse workflows in data science, scientific computing, and machine learning.*
> *JupyterLab is also extensible and modular, enabling plugins that add new components or integrate with existing ones."*

8.8.1 Starting JupyterLab

Run the following command in the Terminal:

```
jupyter lab --browser=firefox
```

8.8.2 Sample session