Remote Access to your Workstation

For security reasons, access to your Lorentz workstation is only possible within the Lorentz Institute intranet. Remote access can occur either securing your connection via an intermediate step called *SSH tunneling* (AKA *port forwarding*), through the Lorentz Institute VPN service or via the Lorentz Institute Remote Workspace.

Following are some examples that demonstrate the concept of SSH tunnelling. For alternative methods of connection, please see the relevant documentation.

SSH tunneling

By means of an SSH tunnel you can transport any arbitrary data over an encrypted SSH connection. Members of the Lorentz Institute can use this technique to gain remote shell access to their workstation across our firewall which would prevent access otherwise.

How does it work?

You must have an ssh client installed on your personal device – e.g. laptop, PC – in order to establish a *tunnelled* connection.
The Lorentz Institute has a dedicated server (SSH server) ready to listen to any (authenticated) client connections.
Once a client-server connection is established, a given application contacts the SSH client on a chosen port on which the client is listening.
The SSH client in turns forwards all encrypted application data to the server which finally communicates with the actual application server.

For remote ssh connections to your IL workstation, the steps above can be summarised into the following. Establish an ssh client-server to our SSH server and instruct your SSH client to forward any new SSH-connection data that will be sent to an arbitrary port number to go via our SSH server. The server will then relay this information to the SSH server running on your workstation.

Example 1

Establish an SSH connection to `workstation.lorentz.leidenuniv.nl` via our SSH server `styx.lorentz.leidenuniv.nl`

```
ssh -o ProxyCommand="ssh -W %h:%p username@styx.lorentz.leidenuniv.nl" username@workstation.lorentz.leidenuniv.nl
```

⚠️ For connections that will use the DISPLAY environment variable (think of any application with a GUI), add the option `-X` to your SSH commands.

Example 2

As in *Example 1* but this time using your client ssh configuration file usually located at $HOME/.ssh/config on GNU/Linux systems

```
# cat $HOME/.ssh/config
Host workstation.lorentz.leidenuniv.nl workstation
     ProxyCommand /usr/bin/ssh -W %h:%p styx.lorentz.leidenuniv.nl
     User username
```

Once this configuration is in place, a simple `ssh workstation` will get you to your workstation.

Example 3

Establish a web browser connection to a jupyter notebook on workstation.lorentz.leidenuniv.nl port YYYY. Configure your local $HOME/.ssh/config as below

```
Host styx
    HostName styx.lorentz.leidenuniv.nl
    LocalForward YYYY localhost:YYYY

Host marisXX
    HostName marisXX
    ProxyJump maris
    LocalForward YYYY localhost:YYYY
```

You are now ready to tunnel your connections through `xmaris` and visualize your notebook at http://localhost:YYYY.

NOTE: For this to work your workstation must have OpenSSH v7.3+.

Example 4

⚠️ Users are encouraged to use maris OoD interface to launch jupyter notebook.

Same situation as in Example 3 but this time you sit behind your laptop at home.

```
Host lorentz
     HostName ssh.lorentz.leidenuniv.nl
     User <your-IL-username>
     IdentityFile ~/.ssh/id_rsa

Host maris
     HostName xmaris.lorentz.leidenuniv.nl
```

```
    ProxyJump lorentz
    IdentityFile ~/.ssh/id_rsa
    User <your-IL-username>

Host marisXX
    HostName marisXX.lorentz.leidenuniv.nl
    ProxyJump maris
    IdentityFile ~/.ssh/id_rsa
    User <your-IL-username>
    LocalForward YYYY localhost:YYYY
```
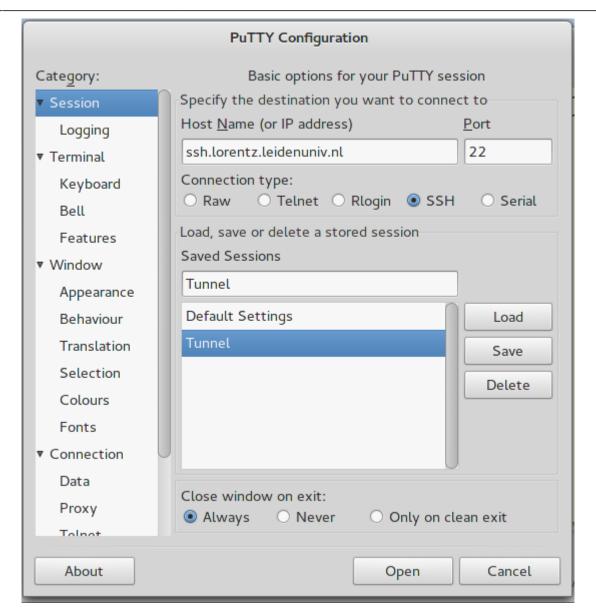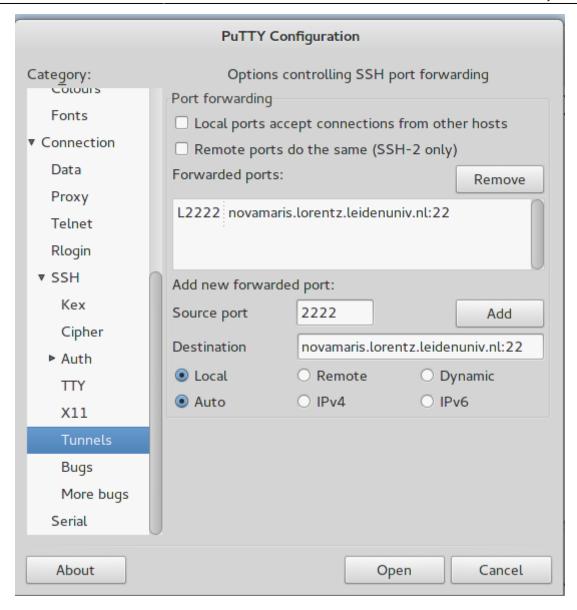
Example 5: Using Putty

In the snapshots that follow, please replace all occurrences of `novamaris` with `xmaris`.

You will need to open two putty sessions. The first one opens a tunnel, the second one uses it. For the sake of clarity let us call the first session `Tunnel' and the second one `Tunnel_use'. In this example we will establish a connection to the Maris headnode `novamaris` through an ssh tunnel on `ssh.lorentz.leidenuniv.nl`.
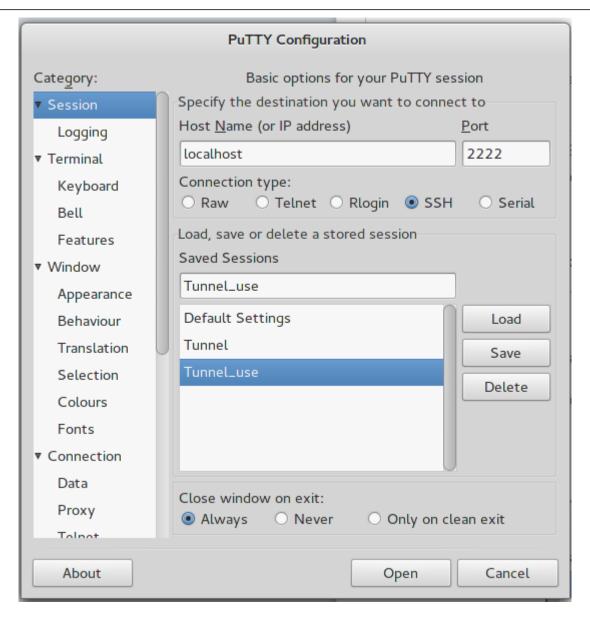
Session 1: Tunnel

Open putty and create a session called Tunnel, then set it according to the snapshots below

PuTTY Configuration

Category:

Colours
Fonts
▼ Connection
   Data
   Proxy
   Telnet
   Rlogin
  ▼ SSH
     Kex
     Cipher
    ▶ Auth
     TTY
     X11
     Tunnels
     Bugs
     More bugs
   Serial

Options controlling SSH port forwarding

Port forwarding

☐ Local ports accept connections from other hosts
☐ Remote ports do the same (SSH-2 only)

Forwarded ports: [Remove]

L2222 novamaris.lorentz.leidenuniv.nl:22

Add new forwarded port:

Source port: 2222 [Add]

Destination: novamaris.lorentz.leidenuniv.nl:22

◉ Local ○ Remote ○ Dynamic
◉ Auto ○ IPv4 ○ IPv6

[About] [Open] [Cancel]

Please note the settings in the port forwarding panel. We use an arbitrary port (2222), but you can choose any numbers above 1024 provided they are not currently in use. By pushing `Open', a terminal will appear so that you can authenticate using your Lorentz institute credentials. Leave this terminal open and proceed with the creation of session `Tunnel_use'

Now push `Open' and a terminal will appear asking your authentication credentials on novamaris. Any other putty connection to localhost:2222 will ssh directly to novamaris.

VNC: GNU/Linux

For detailed instructions on how to set up a vnc session you are encouraged to follow these instructions.

Finally, take a look at x2go should you be interested.

VNC: Windows Users

Please read here.

Proxy Browsing

Read here

From:
https://helpdesk.physics.leidenuniv.nl/wiki/ - **Computer Documentation Wiki**

Permanent link:
https://helpdesk.physics.leidenuniv.nl/wiki/doku.php?id=institute_lorentz:institutelorentz_remoteaccess&rev=1611735275

Last update: **2021/01/27 08:14**