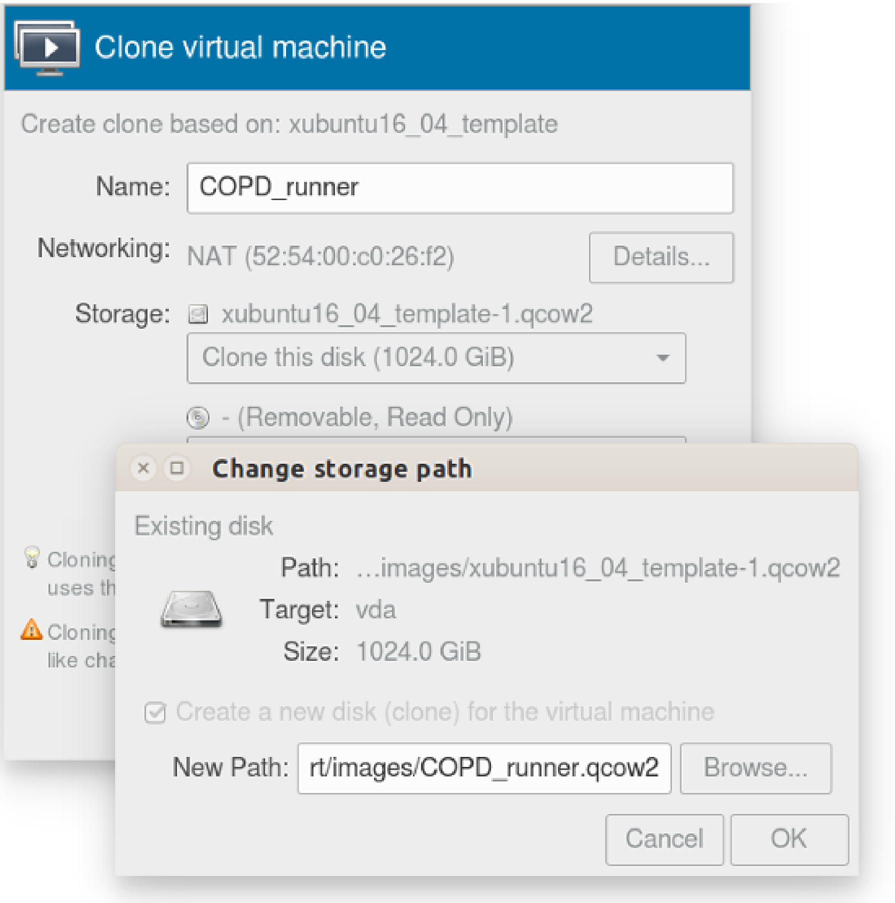
## Building a new Research VM

Document Author: arho.virkki@tyks.fi



Name: COPD_runner

Networking: NAT (52:54:00:c0:26:f2)

Storage: xubuntu16_04_template-1.qcow2

Clone this disk (1024.0 GiB)

- (Removable, Read Only)

Path: …images/xubuntu16_04_template-1.qcow2

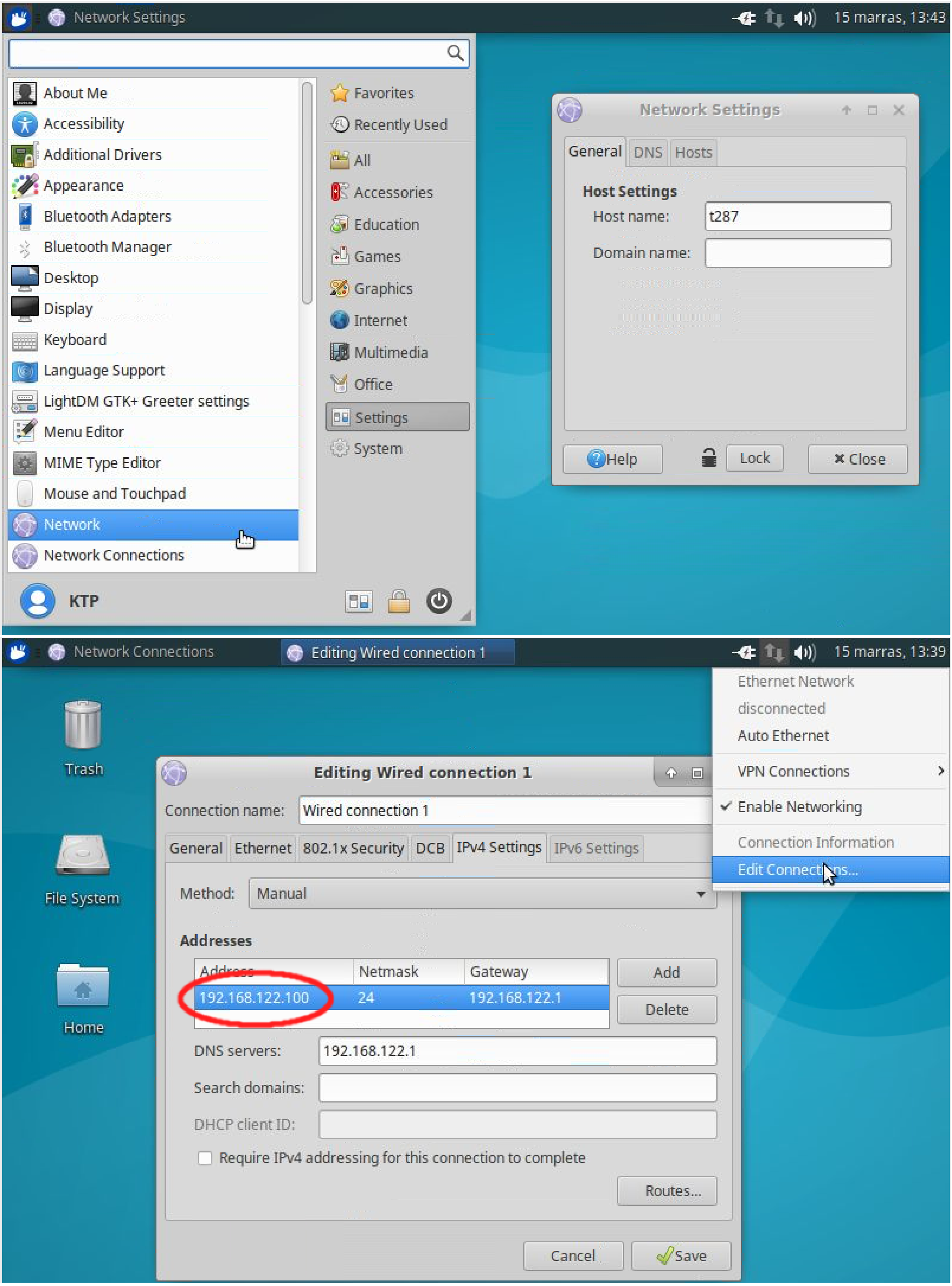
Target: vda

Size: 1024.0 GiB

☑ Create a new disk (clone) for the virtual machine

New Path: rt/images/COPD_runner.qcow2

**Clone an approproate virtual machine.** Then, log in as with the *ktp* user.



Network Settings

**Host Settings**

Host name: t287

Domain name:

Editing Wired connection 1

Connection name: Wired connection 1

Method: Manual

**Addresses**

| Address | Netmask | Gateway |
|---|---|---|
| 192.168.122.100 | 24 | 192.168.122.1 |

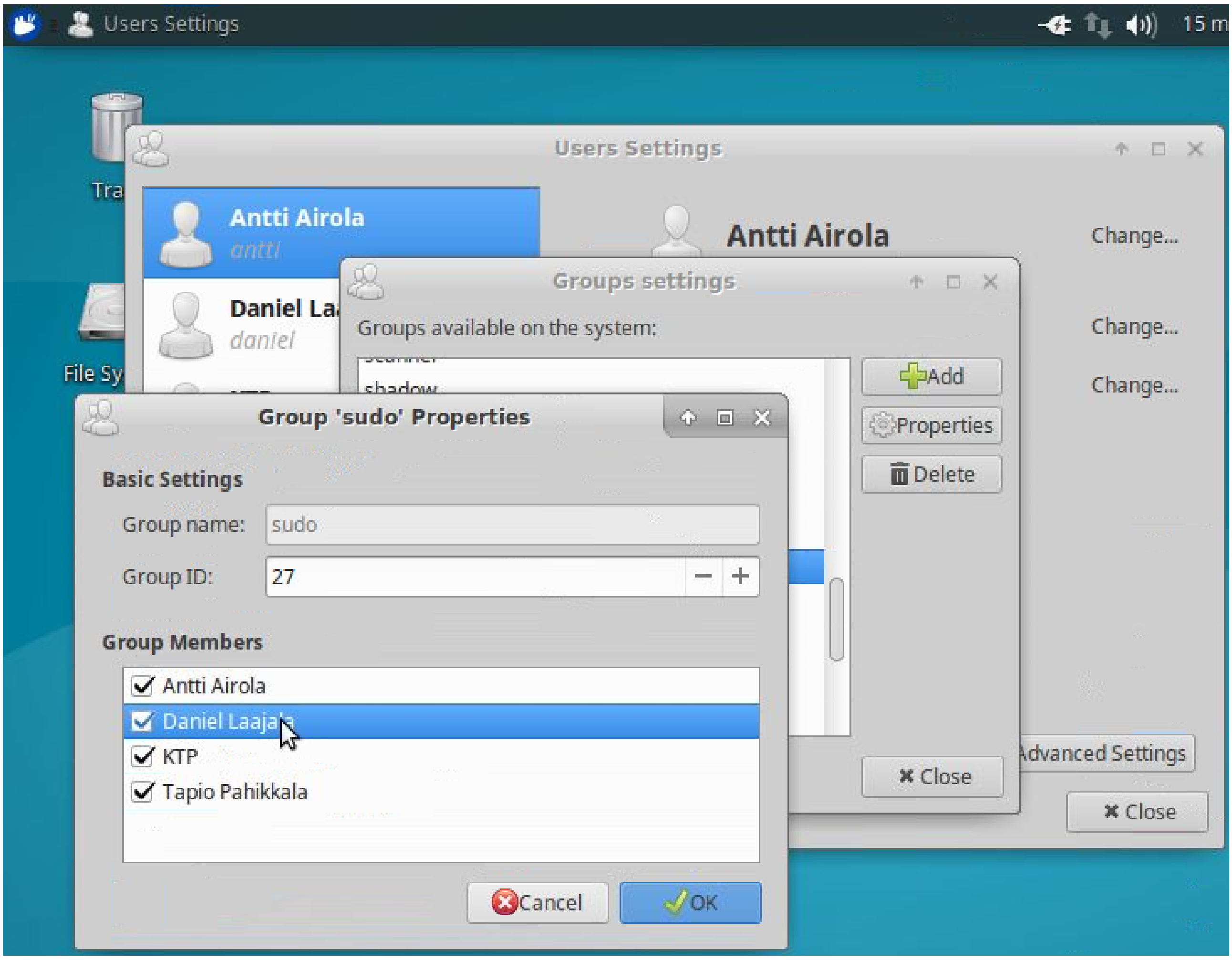
DNS servers: 192.168.122.1

Search domains:

DHCP client ID:

☐ Require IPv4 addressing for this connection to complete

**Configure the machine.** Configure a static IP and add it to the above list. Remember to refresh the MAC address, if needed (since it gets changed when machine is cloned). Then set machine name to the research project number (e.g. t287).



Basic Settings

Group name: sudo

Group ID: 27

Group Members

☑ Antti Airola
☑ Daniel Laajala
☑ KTP
☑ Tapio Pahikkala

**Manage users.** Add user accounts and add sudo (admin) right to the VM users.

### Adding public keys to Bastion

For testing purposes, we can also create a key for the ktp user:

```
ssh-keygen -C "Key pair for KTP User" -f KTP_Key
```

Copy the key to the bastion key pool:

```
scp KTP_Key* ktp@analytics.tyks.fi:
ssh ktp@analytics.tyks.fi
sudo sh -c "cat KTP_Key.pub >> /etc/ssh-pool/bastion_keys"
```

Now, connection to the machine can be made with

```
ssh <user>@192.168.122.<machine_number> -o ProxyCommand="ssh bastion@analytics.tyks.fi -W %h:%p -i
    <private_key>"
```

For example,

```
ssh ktp@192.168.122.100 -o ProxyCommand="ssh bastion@analytics.tyks.fi -W %h:%p -i ~/.ssh/KTP_Key"
```

### Note: The proxy command can also be stored in *.ssh/config*

```
Host 192.168.122.100
  IdentityFile /home/johndoe/Documents/Keys/My_VSSHP_Key
  ProxyCommand ssh bastion@analytics.tyks.fi -W %h:%p
```

### Copying data to External VMs

Open a port connection from in-house server (*ktpanalytics*) to the virtual machine's postgresql and leave it open in one terminal. In the following, change *192.168.122.<machine_number>* into the actual machine ip (like 192.168.122.*126*)

```
ssh ktp@192.168.122.<machine_number> -L 5432:localhost:5432 \
-o ProxyCommand="ssh bastion@analytics.tyks.fi -W %h:%p -i ~/.ssh/KTP_Key"
```

The database *research* and role *analyst* should exist in the VM template. If not, connect the research database and create the database and the user:

```
PGPASSWORD=ktp psql -d postgres -h localhost -U ktp
CREATE ROLE analyst LOGIN CREATEDB CREATEROLE PASSWORD 'analyst';
CREATE DATABASE research WITH owner analyst;
```

Then, copy the data into the virtual machine with the following command at the in-house server (*ktpanalytics*). Change *<schema_name>* into the actual schema containing the data to be exported.

```
pg_dump -U ktp -d ktp -h gradient.vsshp.net -n <schema_name> | \
PGPASSWORD=ktp psql -d research -h localhost -U ktp
```

Log in as 'ktp' to chech that the data is OK, and optionally rename the schema to something more generic to the researcher:

```
PGPASSWORD=ktp psql -d research -h localhost -U ktp

\dn
ALTER SCHEMA <old_schema_name> RENAME TO data;
GRANT USAGE ON SCHEMA data TO analyst;
GRANT SELECT ON ALL TABLES IN SCHEMA data TO analyst;
ALTER ROLE analyst SET search_path TO data,public;
CREATE EXTENSION adminpack;
```

Finally, check that the analyst can log in and view data:

```
PGPASSWORD=analyst psql -d research -h localhost -U analyst

\dt
SELECT count(1) FROM asiakas ;
```

### Connecting the VM with X2Go

### Adding Extra tools (to be included into the template)

PostgreSQL:

```
sudo apt-get install postgresql-9.5 libpq-dev
sudo -u postgres psql -c "CREATE ROLE ktp SUPERUSER LOGIN PASSWORD 'ktp'";
```

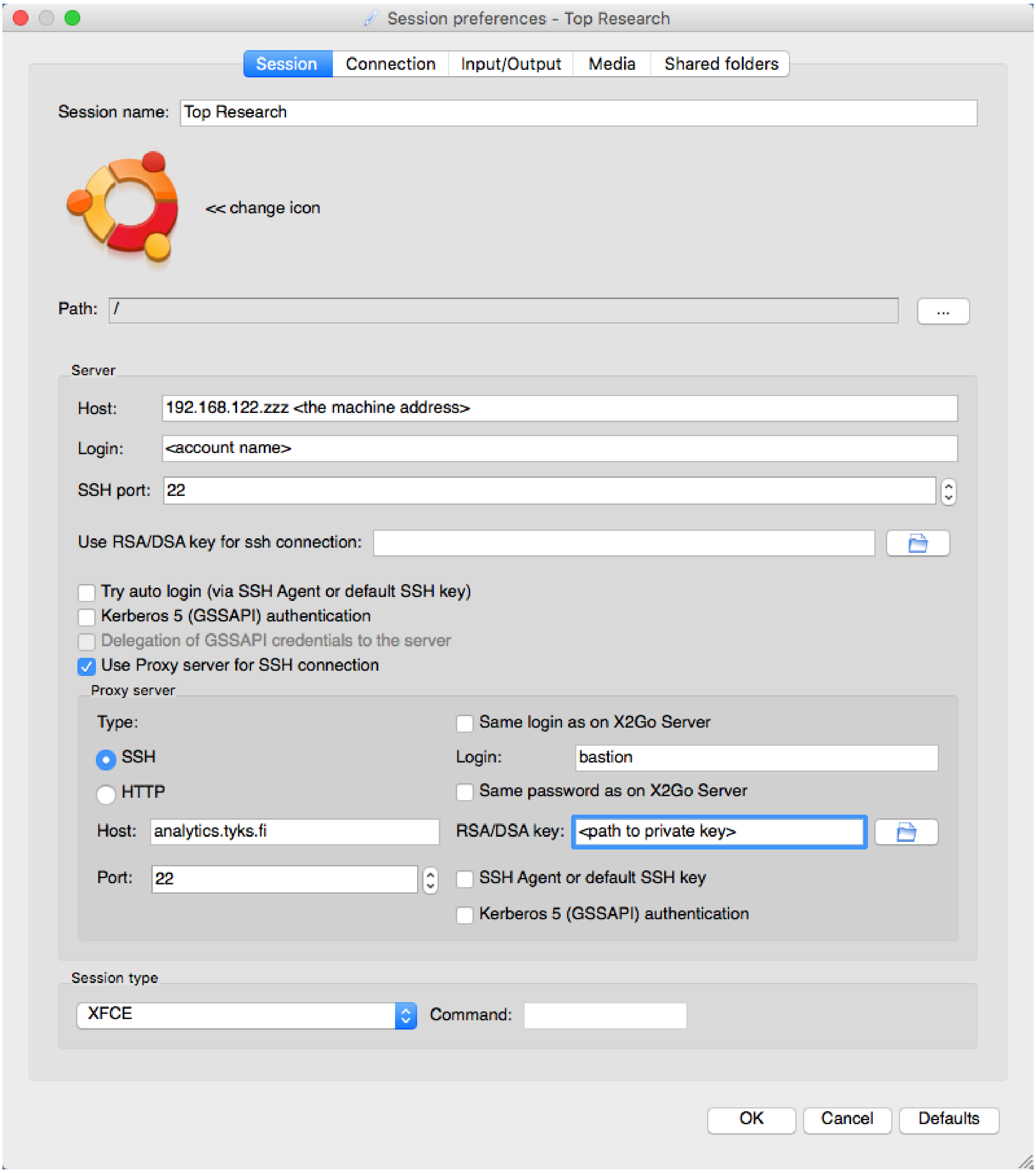
Java:

```
sudo apt-get install default-jdk
```

SQL GUI Tools:

```
# PgAdmin3
sudo apt-get install pgadmin3

# SquirrelSQL
wget
    http://sourceforge.net/projects/squirrel-sql/files/1-stable/3.7.1/squirrel-sql-3.7.1-standard.jar
sudo java -jar squirrel-sql-3.7.1-standard.jar
sudo ln -s /usr/local/squirrel-sql-3.7.1/squirrel-sql.sh /usr/local/bin/
wget https://jdbc.postgresql.org/download/postgresql-9.4.1212.jar
mv postgresql-9.4.1212.jar /opt/ktp/jar/
```



Session preferences - Top Research

Session name: Top Research

Path: /

Server

Host: 192.168.122.zzz <the machine address>

Login: <account name>

SSH port: 22

Use RSA/DSA key for ssh connection:

□ Try auto login (via SSH Agent or default SSH key)

□ Kerberos 5 (GSSAPI) authentication

□ Delegation of GSSAPI credentials to the server

☑ Use Proxy server for SSH connection

Proxy server

Type:

● SSH

○ HTTP

Host: analytics.tyks.fi

Port: 22

□ Same login as on X2Go Server

Login: bastion

□ Same password as on X2Go Server

RSA/DSA key: <path to private key>

□ SSH Agent or default SSH key

□ Kerberos 5 (GSSAPI) authentication

Session type

XFCE

Command:

Figure 1:

Editors:

```
sudo apt-get install vim-gtk3
```

R + RStudio:

```
sudo apt-get install r-core

sudo apt-get install gdebi-core
wget https://download2.rstudio.org/rstudio-server-1.0.44-amd64.deb
sudo gdebi rstudio-server-1.0.44-amd64.deb
```