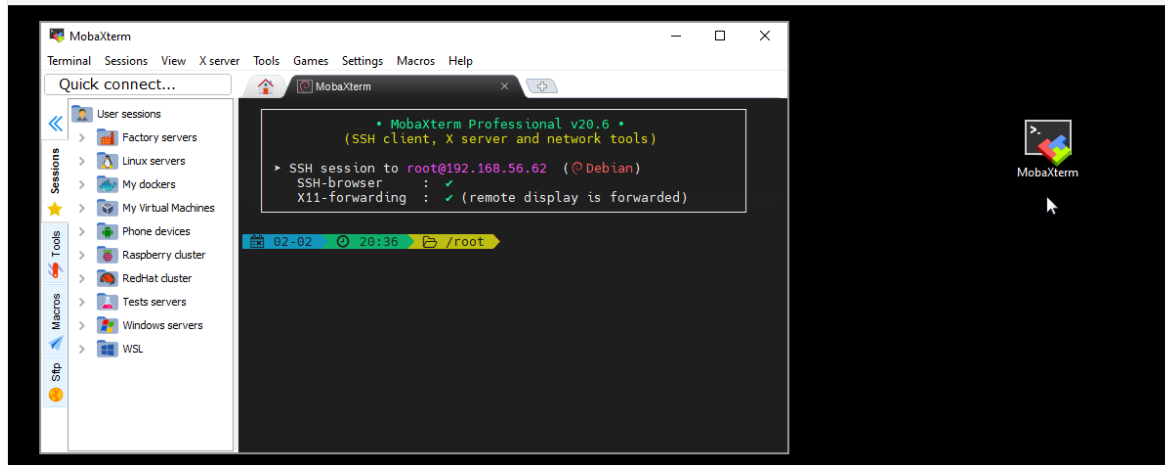


Access to the Linux server - WINDOWS

<https://mobaxterm.mobatek.net/download-home-edition.html>

MobaXterm

Enhanced terminal for Windows with X11 server, tabbed SSH client, network tools and much more



Home Edition

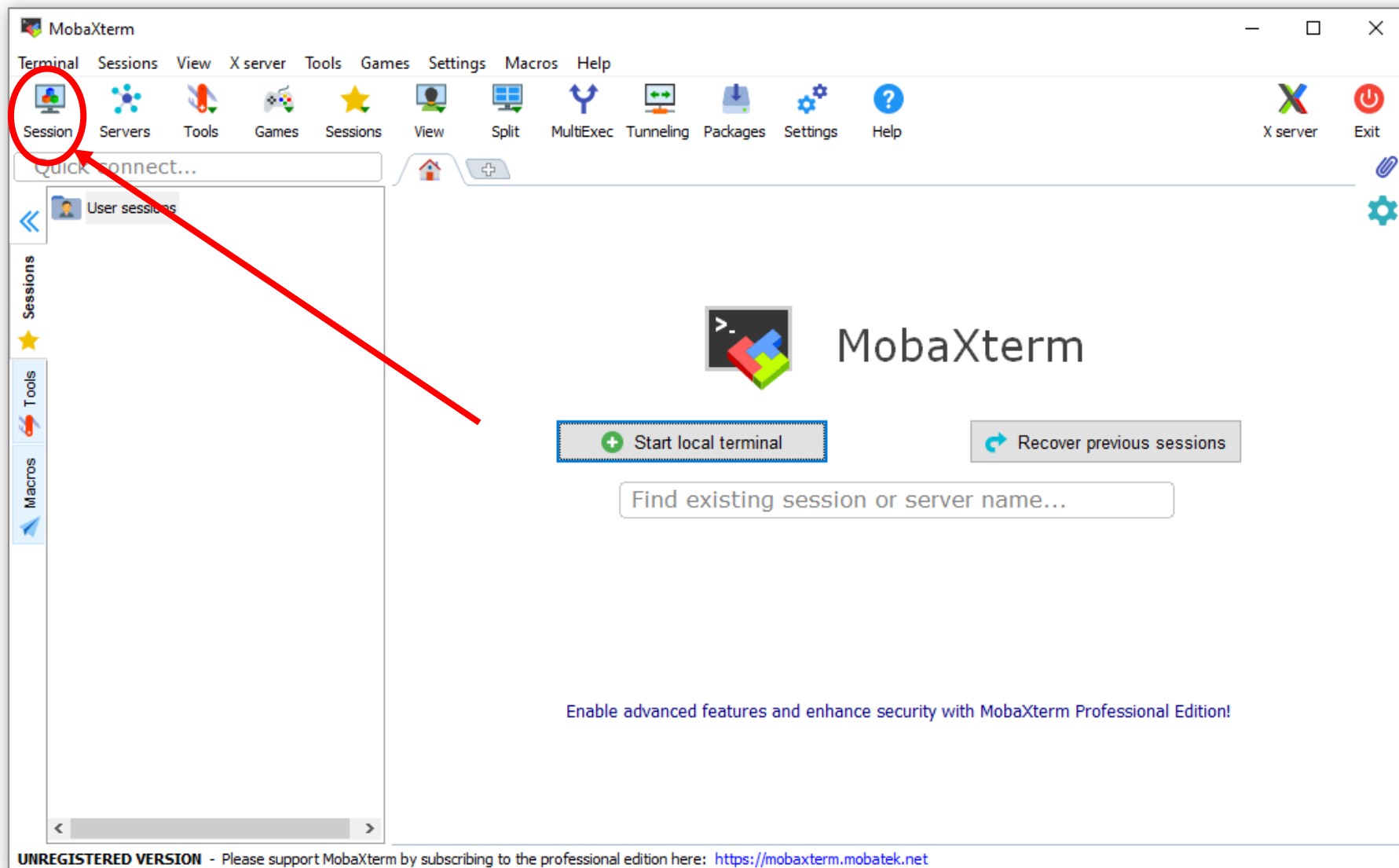
Free

Full **X server** and **SSH** support
Remote desktop (RDP, VNC, Xdmcp)
Remote terminal (SSH, telnet, rlogin, Mosh)
X11-Forwarding
Automatic SFTP browser
Master password protection
Plugins support
Portable and installer versions
Full documentation
Max. **12** sessions
Max. **2** SSH tunnels
Max. **4** macros
Max. **360** seconds for Tftp, Nfs and Cron

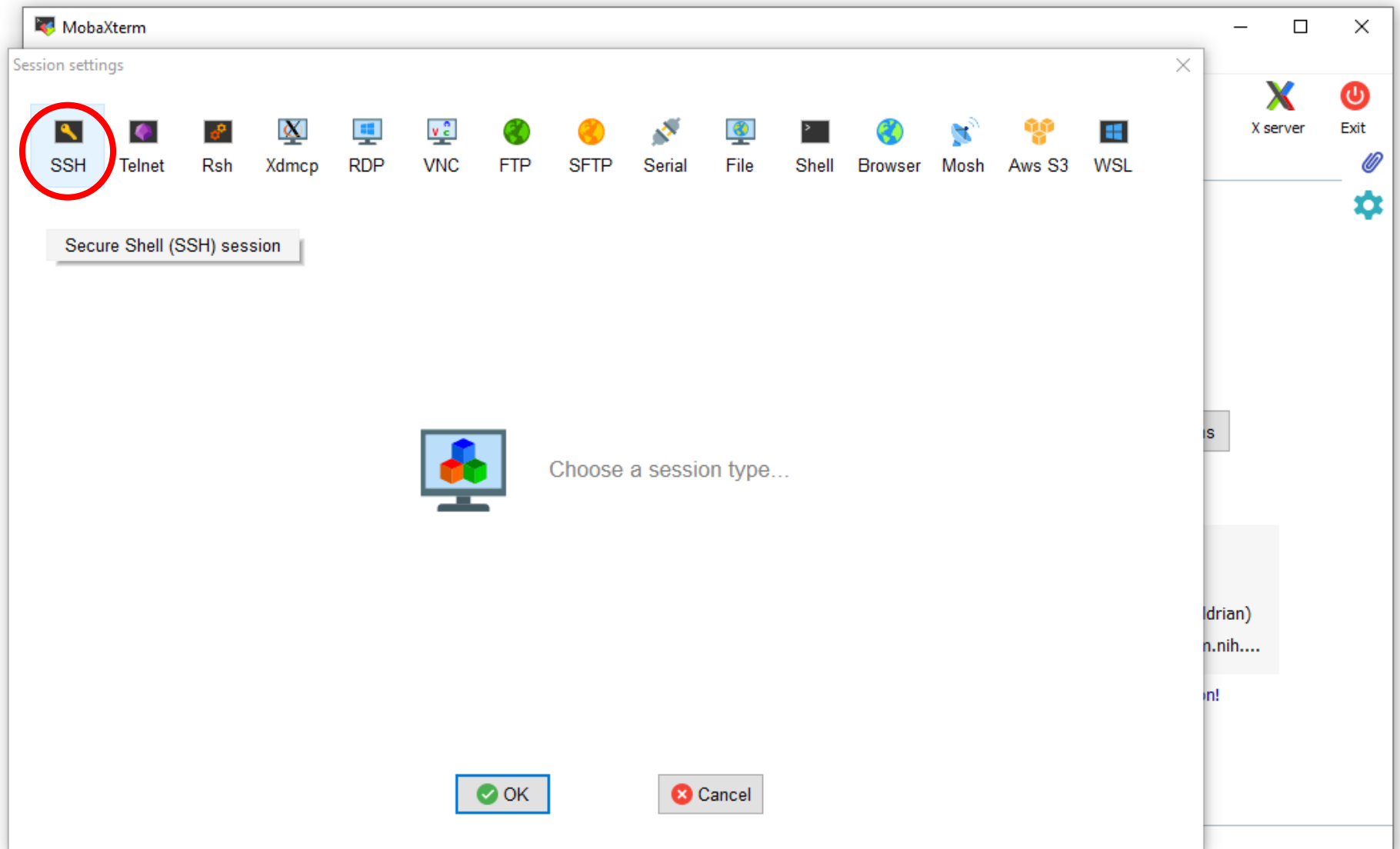


Download now

Access to the Linux server

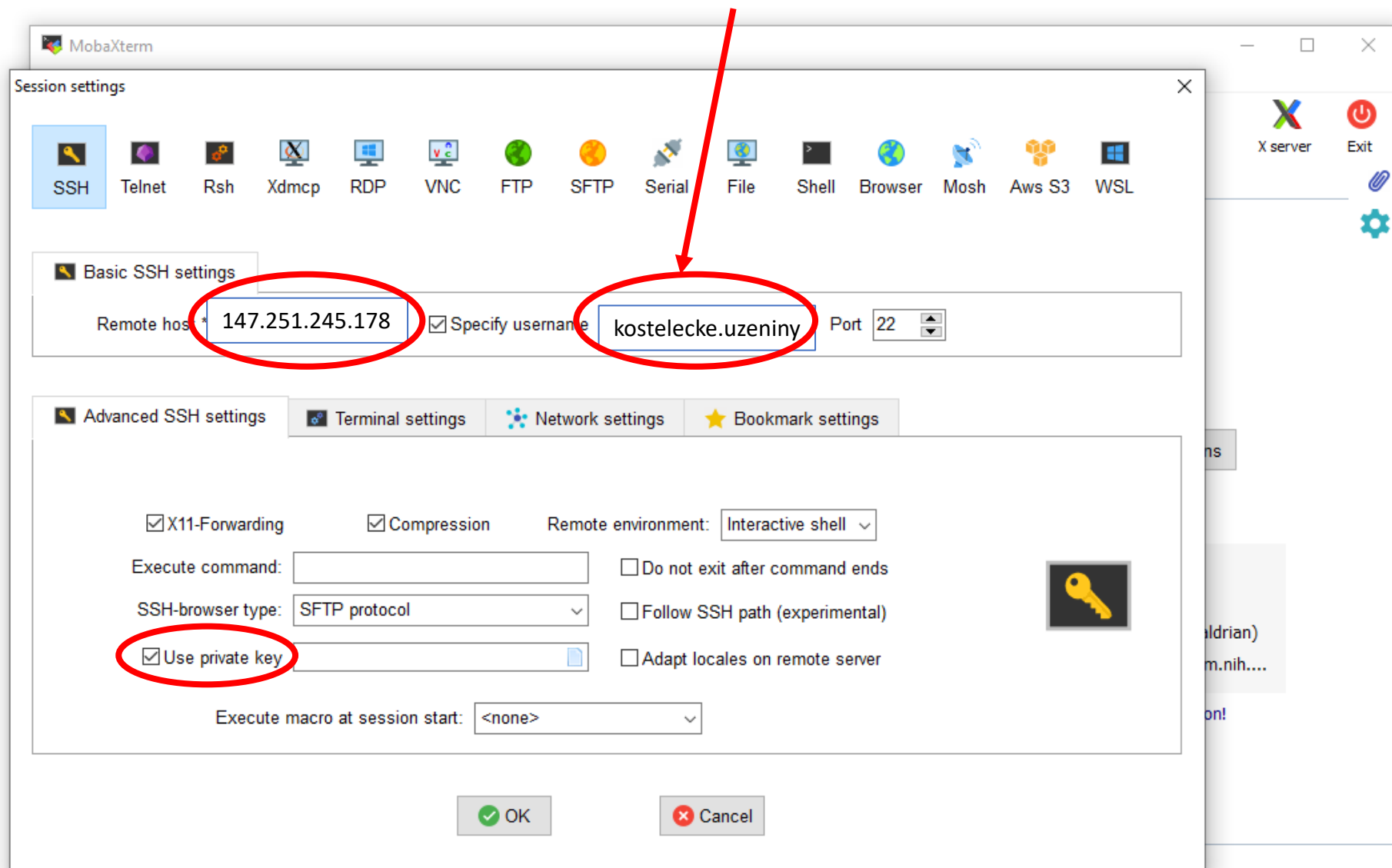


Access to the Linux server

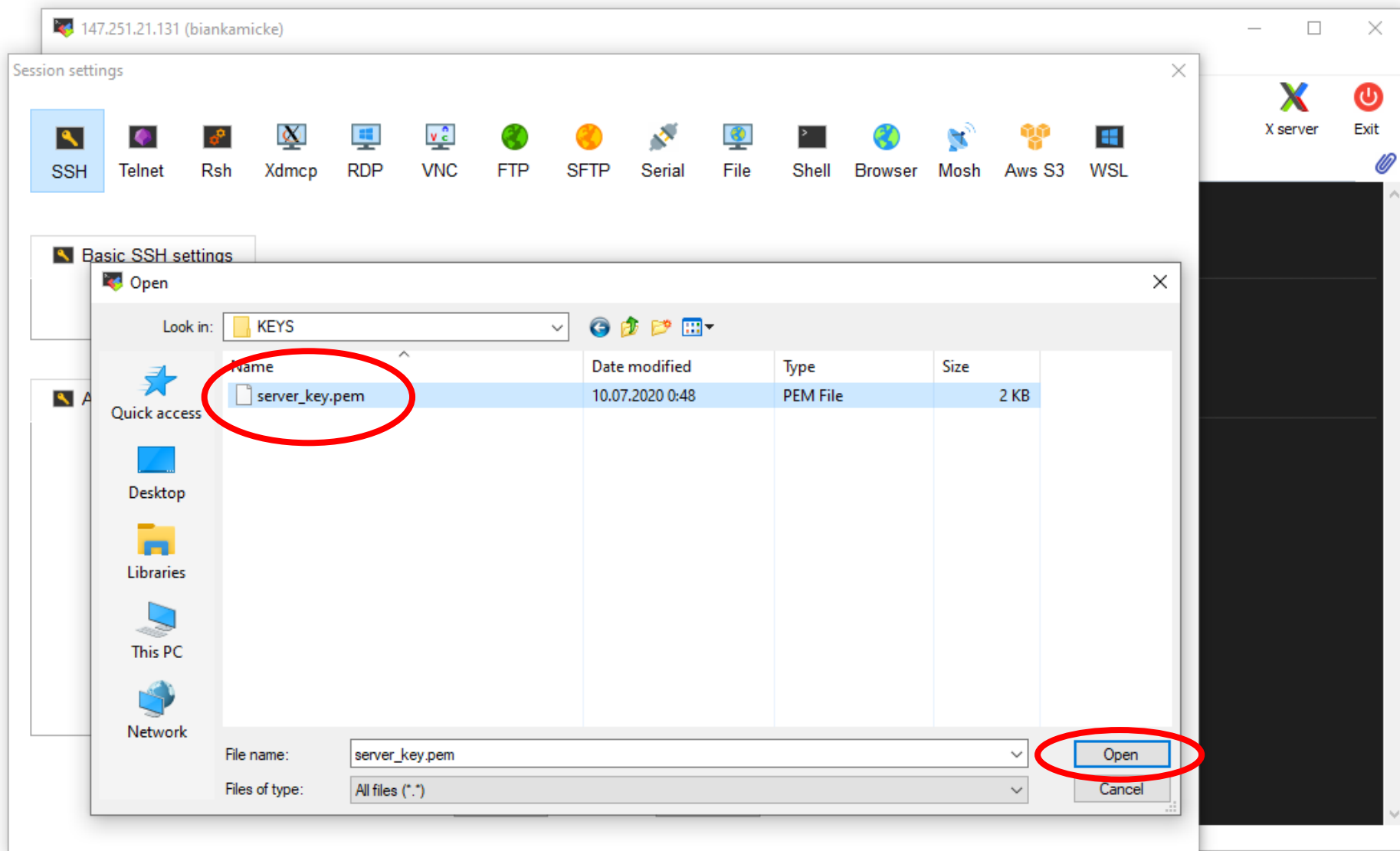


Access to the Linux server

The first part of email before „@“ – my example:
kostelecke.uzeniny@seznam.cz



Access to the Linux server



https://drive.google.com/file/d/14dWJ64UZu4fwG_fccGqMFdf5tvoFFI-s/view?usp=share_link

Access to the Linux server

Session settings

SSH Telnet Rsh Xdmcp RDP VNC FTP SFTP Serial File Shell Browser Mosh Aws S3 WSL

Basic SSH settings

Remote host * 147.251.245.178 ☒ Specify username kostelecke.uzeniny Port 22

Advanced SSH settings Terminal settings Network settings Bookmark settings

☒ X11-Forwarding ☒ Compression Remote environment: Interactive shell

Execute command: ☐ Do not exit after command ends

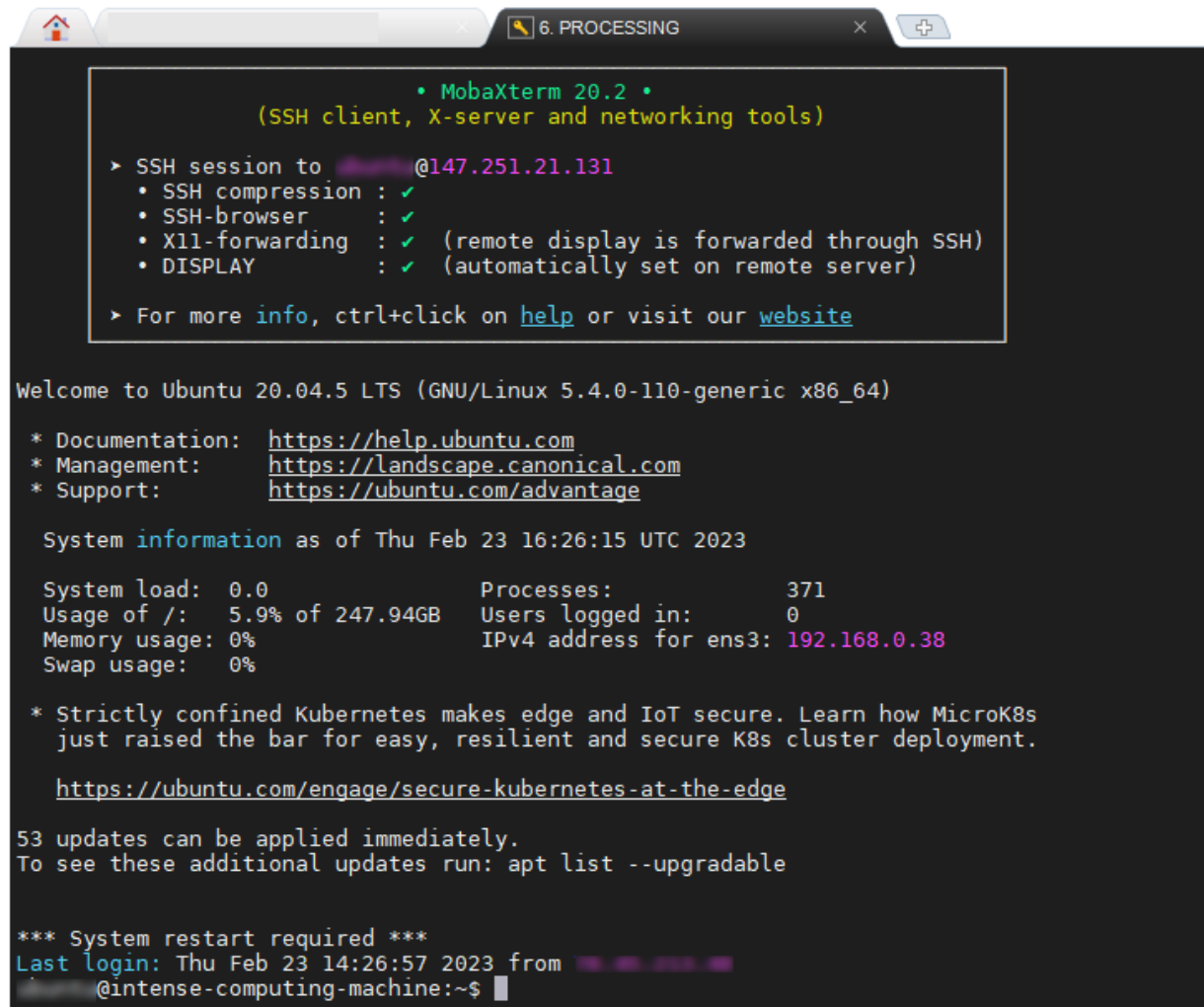
SSH-browser type: SFTP protocol ☐ Follow SSH path (experimental)

☒ Use private key D:\KEYS\KEYS\server_key.pem ☐ Adapt locales on remote server

Execute macro at session start: <none>

OK Cancel

Access to the Linux server



The image shows a terminal window titled "6. PROCESSING" with a dark background. At the top, it displays "MobaXterm 20.2" and "(SSH client, X-server and networking tools)". Below this, it shows the details of an SSH session to a server at IP 147.251.21.131. The session status is "OK", and various options like SSH compression, SSH-browser, X11-forwarding, and DISPLAY are all enabled. A message at the bottom of the terminal window states "53 updates can be applied immediately" and provides instructions on how to update the system using 'apt list --upgradable'. It also mentions that a system restart is required. The terminal prompt is '~\$'.

```
• MobaXterm 20.2 •
(SSH client, X-server and networking tools)

> SSH session to [redacted]@147.251.21.131
  • SSH compression : ✓
  • SSH-browser      : ✓
  • X11-forwarding   : ✓ (remote display is forwarded through SSH)
  • DISPLAY          : ✓ (automatically set on remote server)

> For more info, ctrl+click on help or visit our website

Welcome to Ubuntu 20.04.5 LTS (GNU/Linux 5.4.0-110-generic x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

System information as of Thu Feb 23 16:26:15 UTC 2023

System load:  0.0               Processes:            371
Usage of /:   5.9% of 247.94GB   Users logged in:     0
Memory usage: 0%               IPv4 address for ens3: 192.168.0.38
Swap usage:   0%

* Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
  just raised the bar for easy, resilient and secure K8s cluster deployment.

https://ubuntu.com/engage/secure-kubernetes-at-the-edge

53 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

*** System restart required ***
Last login: Thu Feb 23 14:26:57 2023 from [redacted]
[redacted]@intense-computing-machine:~$
```

Access to the Linux server - MAC



```
ssh -i server_key.pem <user_name>@147.251.245.178
```

- Open terminal and go to your **.ssh/** directory
- Create a file called **config**
- Add the following 1 line configuration in this file

```
Host * PubkeyAcceptedKeyTypes=+ssh-dss
```

```
p <path to your key> ~/.ssh/<name of your key>
```

Now you should be able to modify the permissions normally.

```
chmod 600 ~/.ssh/<your key's name>
```

Then ssh using WSL:

```
ssh -i ~/.ssh/<name of your key> <username>@<ip address>
```


Basic commands

pwd	Show current address	pwd
man	Show command manual	man chosen_command
cd	Change directory	cd directory_path
ls	List files and directories	-a or -all List hidden files -l or -list show details -h Human readable
mkdir	Create a directory	mkdir folder_Name

```
$ pwd  
/home/<username>
```

Create a directory called “carnaval”. Check what is inside your new directory.

Copying, moving and deleting

cp	Copy file or folder	<code>cp original_file copy_file</code>
mv	Move/rename file or folder	<code>mv file directory</code>
rm	Remove file or directory	<code>rm file1</code> <code>rm -r directory</code>
rmdir	Remove directory	<code>rmdir dir1</code>
touch	Create empty file	<code>touch file.txt</code>

Create three new files empty file called “rain.txt”, “people.txt” and “samba.txt”

Now move rain, people and samba to the carnaval!

Copying, moving and deleting

cp	Copy file or folder	<code>cp original_file copy_file</code>
mv	Move/rename file or folder	<code>mv file directory</code>
rm	Remove file or directory	<code>rm file1</code> <code>rm -r directory</code>
rmdir	Remove directory	<code>rmdir dir1</code>
touch	Create empty file	<code>touch file.txt</code>

Check what is inside the carnaval with the command “ls”.

The rain is killing it. Remove the rain from the carnaval.

See how this is fun?

Great. Now, lets stop with this party and get serious.

Change the name of the carnaval directory to work and remove the samba and the people.

Download a file from the internet

wget

wget -O sequences.zip

"<http://www.biomed.cas.cz/mbu/lbwrf/sequences.zip>"

unzip

unzip sequences.zip

File exhibition

cat	Exhibit and contatenate	<code>cat file.txt</code>
less	Read a file	<code>less file.txt</code>
more	Read a file	<code>more file.txt</code>
wc	Count the number of lines, characters and bytes of a file	<code>wc -l file.txt</code> [lines of the file] <code>ls wc -l</code> [how many files are there]
head	First 10 lines of the file	<code>head -n 21 file.txt</code>
tail	Last 10 lines of the file	<code>tail -n 15 file.txt</code>
sort	Order the lines of the file by the user's definition.	<code>sort names > names.sorted</code>

Check the “sequences” with the command less.

Use the key arrows in your keyboard to navigate through this file and the key “q” to exit the less command.

File exhibition

cat	Exhibit and contatenate	<code>cat file.txt</code>
less	Read a file	<code>less file.txt</code>
more	Read a file	<code>more file.txt</code>
wc	Count the number of lines, characters and bytes of a file	<code>wc -l file.txt</code> [lines of the file] <code>ls wc -l</code> [how many files are there]
head	First 10 lines of the file	<code>head -n 21 file.txt</code>
tail	Last 10 lines of the file	<code>tail -n 15 file.txt</code>
sort	Order the lines of the file by the user's definition.	<code>sort names > names.sorted</code>

Count the lines the fasta file.

Take a look in the first 16 lines of fasta file.

Take a look in the last 4 lines of fasta file.

File manipulation

>	Forward standar output	<code>program.pl > report.txt</code>
<	Modify the standar input to a file	<code>program.pl < gene.fasta</code>
	Allow the combination of commands	<code>ls wc -l</code>
>>	Adds the output to the end of the file.	<code>programa.pl >> relatorios</code>

Now we are going to generate a sample of our data_set. Take the 16 first lines of the file "sequences.fas" and forward to a new file called "small.fasta" in your work directory. If the file doesn't exist, the forward sign will create it automatically.

Add the last 12 lines from the file "sequences.fas" to your "small.fasta"

Use the pipe sigh "|" to combine the command ls and wc and check how many files are inside the work directory

Text manipulation

cat	Concatenate and exhibit	<code>cat text1 text2 > text1text2</code>
grep	Search the file line by line for defined expressions	<code>grep ">" genes.fasta</code>
uniq	Remove duplicated lines	<code>sort alfa uniq -c</code>
cut	Cut input files. Ideal for tables.	<code>cut -d " " f1 alfa</code>
awk	Programming language with resources for text manipulation	<code>awk -F '{print \$2 \$1}' table.csv</code>
sed	Used to manipulate and transform text.	<code>sed 's/t/u/g' dna.seq > rna.seq</code>

Lets inspect our sequences.

Use grep to search for the a pattern of your interest in your new "fasta" file. For example something from their header. "SampleID_8007"

Count how many times the motive appear in the header by "wc" command.

Choose a pattern, for example the start of the read name ">" and use grep to count how many times it show up in your

Text manipulation

cat	Concatenate and exhibit	<code>cat text1 text2 > text1text2</code>
grep	Search the file line by line for defined expressions	<code>grep ">" genes.fasta</code>
uniq	Remove duplicated lines	<code>sort alfa uniq -c</code>
cut	Cut input files. Ideal for tables.	<code>cut -d " " f1 alfa</code>
awk	Programming language with resources for text manipulation	<code>awk -F '{print \$2 \$1}' table.csv</code>
sed	Used to manipulate and transform text.	<code>sed 's/t/u/g' dna.seq > rna.seq</code>

Now to finish, lets see how many commands we learned, using the command history and forwarding the output to the file `commands.txt`

To inspect this file, use the commands:

```
cut -f 7 -d ' ' commands.txt | sort | uniq | wc -l
```

What does it show you?

What if you want to count how many times you used each one?