



Linux használata az ELKH Cloudon

Rusznák Attila
SZTAKI

Tartalomjegyzék

1. Csatlakozás Linux rendszerhez
2. Kötetek menedzselése
3. Jupyter telepítése





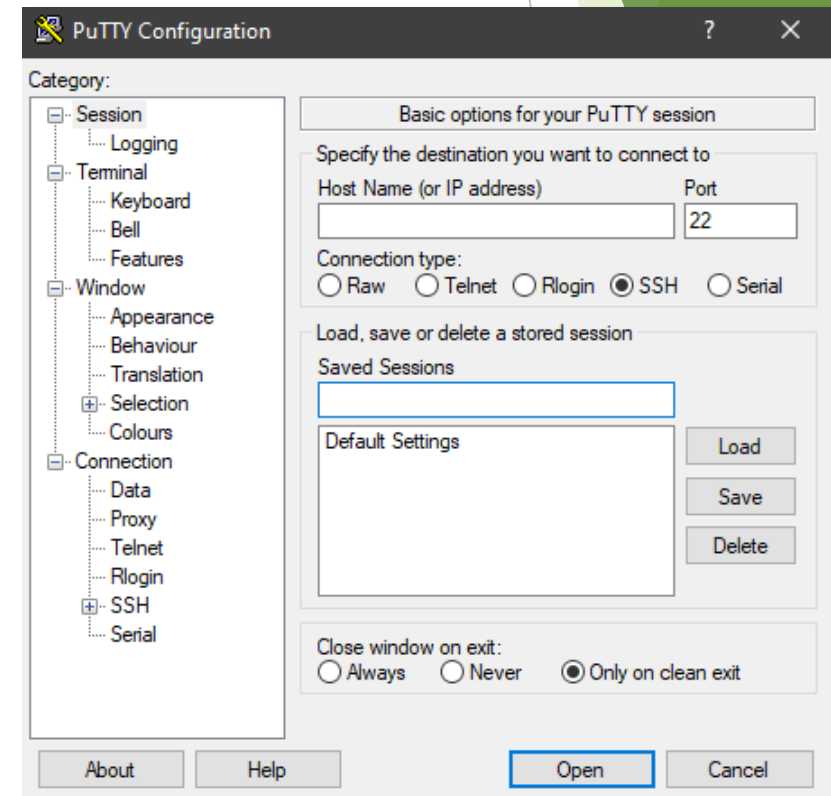
ELKH Cloud

Csatlakozás Linux rendszerhez

Csatlakozás Linux rendszerhez

Windows rendszerről történő csatlakozáshoz a következő lépéseket kövessük:

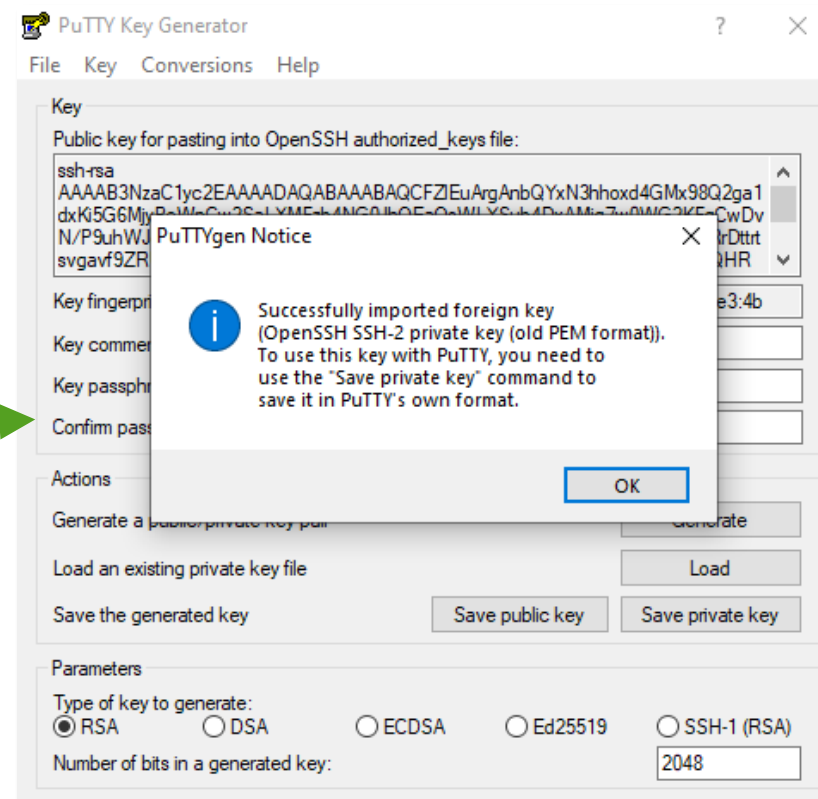
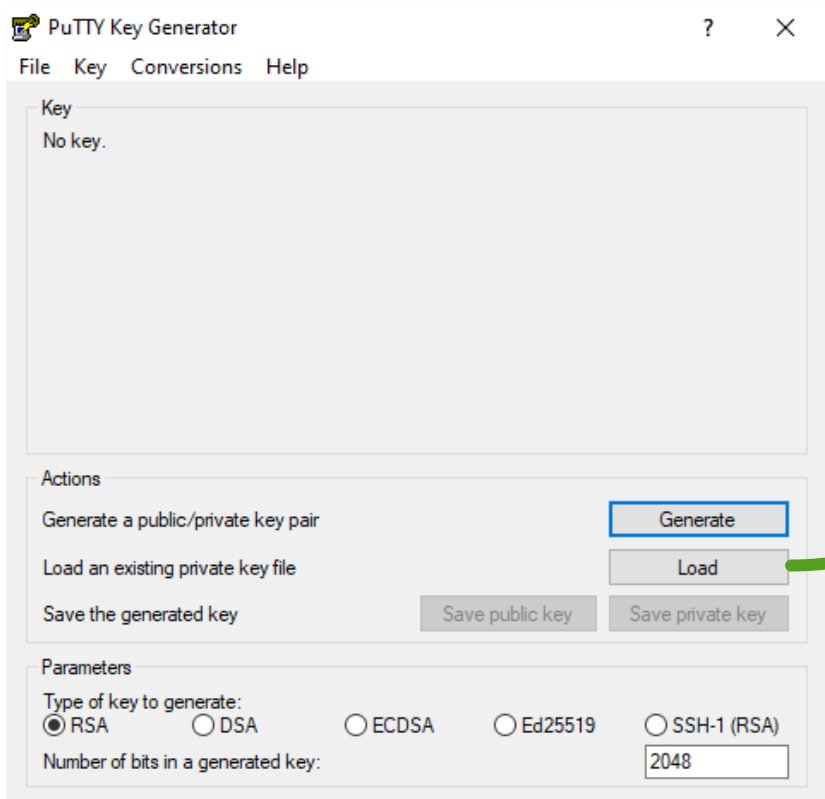
- ▶ **Generáljunk egy saját kulcspárt az ELKH Cloud-on** - 1. ea. 53. dia
 - ▶ Mentsük le a privát kulcsunkat a számítógépünkre
 - ▶ A virtuális gép létrehozásakor adjuk hozzá a kulcsunkat
- ▶ A megfelelő biztonsági csoportok hozzáadása a géphez
 - ▶ **Security Group-ok** beállítása (SSH) - 1. ea. 47. dia
- ▶ Külső IP cím hozzárendelése a géphez
 - ▶ **Floating IP** beállítása - 1. ea. 50. dia
- ▶ Töltsük le és telepítsük a **PuTTY** programot:
 - ▶ <https://www.putty.org>
- ▶ A programot elindítva adjuk meg a következőket:
 - ▶ **Host Name:** a virtuális gép külső IP címe
 - ▶ **Port:** a géphez kapcsolódó SSH port, alapból 22-es



Csatlakozás Linux rendszerhez

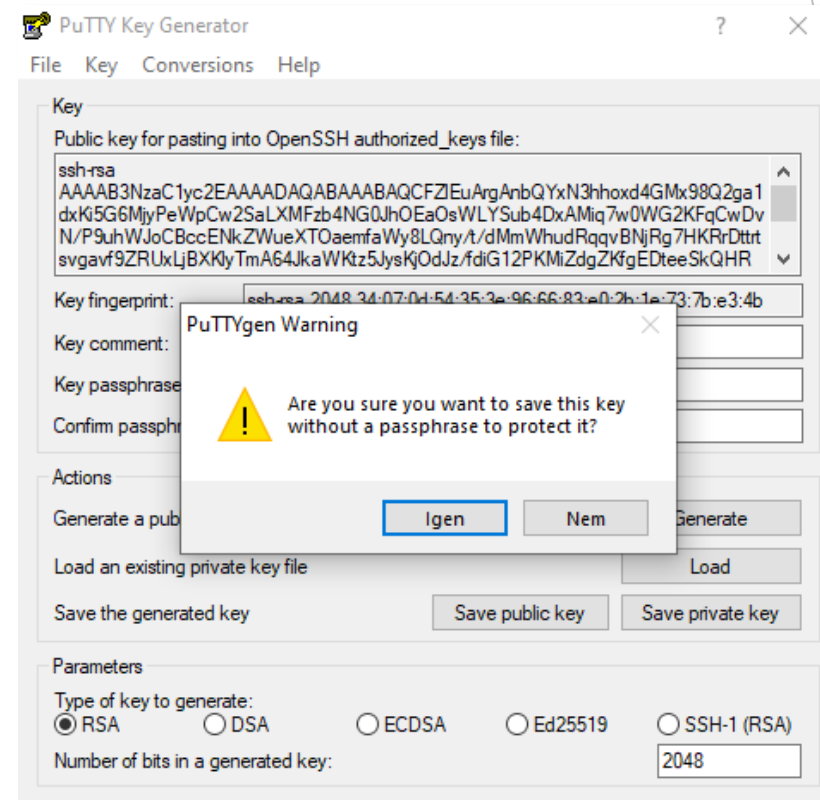
A letöltött privát kulcs kiterjesztése **pem**, amit át kell konvertálnunk:

- ▶ Ehhez indítsuk el a PuTTY Key Generator alkalmazást, majd tallózzuk be a kulcsot



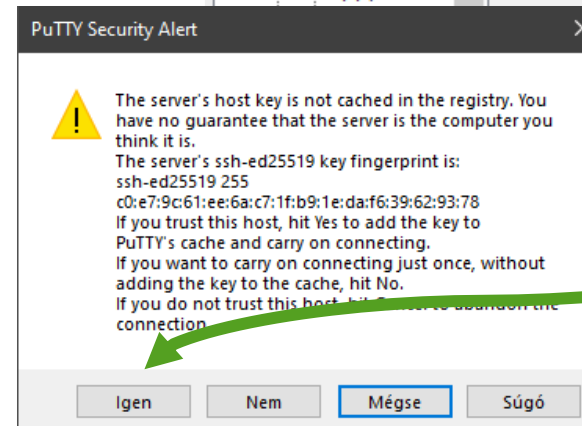
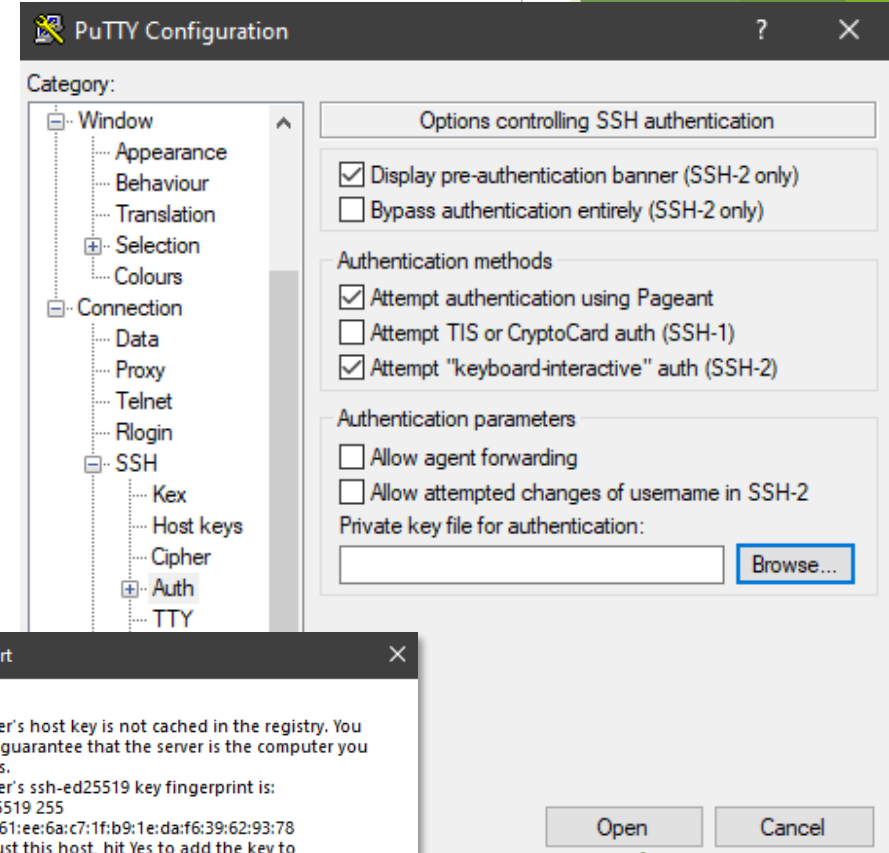
Csatlakozás Linux rendszerhez

- ▶ Kattintsunk a **Save private key** nyomógombra, majd válasszuk az Igen lehetőséget
- ▶ Mentsük el a privát kulcsunkat **ppk** kiterjesztéssel.
- ▶ Most már bezárhatjuk a kulcsgenerátort.



Csatlakozás Linux rendszerhez

- ▶ Lépünk vissza a PuTTY programba.
- ▶ Nyissuk le baloldalt az SSH menüpontot:
 - ▶ Ezen belül válasszuk az **Auth** almenüt
 - ▶ A **Browse** gombra kattintva tallózzuk be a **privát kulcsunkat**
 - ▶ A kulcs kiterjesztése **ppk**, ahogyan az előbb lementettük
- ▶ Kattintsunk az **Open** gombra
 - ▶ Első belépésnél kérni fogja a jelenlegi jelszót: **ubuntu**
 - ▶ Ezt követően kétszer meg kell adnunk egy új jelszót
 - ▶ A PuTTY ablaka be fog záródni (ez normális)
 - ▶ Csatlakozzunk ismét



Külső IP cím nélküli Ubuntu gép elérése

- ▶ A feladat végrehajtásához a következő beállítások szükségesek:
 - ▶ Egy Ubuntu virtuális gép létrehozása külső IP címmel (Floating IP, 1. ea. 50. dia)
 - ▶ Egy Ubuntu virtuális gép létrehozása külső IP cím nélkül
 - ▶ Mindkét gép legyen elérhető SSH-n keresztül (Security Group, 1. ea. 47. dia)
 - ▶ Kapcsolódjunk a külső IP-vel rendelkező gépre a PuTTY segítségével (3. dia)
- ▶ A külső IP-vel **nem** rendelkező gépet nyissuk meg a Cloud-on:
 - ▶ Állítsuk be a jelszó alapú hitelesítést:
`sudo nano /etc/ssh/sshd_config`
 - ▶ Keressük ki a következő utasítást és írjuk át **yes**-re:
`PasswordAuthentication yes`
 - ▶ Nyomjuk meg a **CTRL + X**-et, majd írjuk be: **y**
 - ▶ Végül indítsuk újra a gépen az SSH szolgáltatást:
`sudo service ssh restart`

```
GNU nano 2.9.3 /etc/ssh/sshd_config Modified
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication yes_
#PermitEmptyPasswords no

# Change to yes to enable challenge-response passwords (beware issues with
# some PAM modules and threads)
ChallengeResponseAuthentication no
```


Külső IP cím nélküli Ubuntu gép elérése

- ▶ A külső IP-vel rendelkező gépre csatlakozzunk a PuTTY segítségével:
 - ▶ Írjuk be a kapcsolódáshoz a következő kódot:
`ssh <felhasznalo>@<belso_ip>`
 - ▶ Például:
`ssh ubuntu@192.168.10.231`
- ▶ Csatlakozáskor meg kell erősíteni a kapcsolódási szándékunkat: `yes`
- ▶ Be kell írni a belső IP-vel rendelkező virtuális gép jelszavát (amit megadtunk)
- ▶ Ha szeretnénk kilépni az SSH kapcsolatból írjuk be: `logout`

```
Last login: Mon Oct 19 09:12:36 2020 from 192.168.10.232
ubuntu@ubuntu-test-2:~$ exit
logout
Connection to 192.168.10.231 closed.
ubuntu@ubuntu-test-1:~$ █
```



Kötetek menedzselése

Kötet csatolása Linux rendszerhez

- ▶ Miután készítettünk egy új kötetet, majd hozzácsatoltuk egy Ubuntu virtuális géphez, csatlakozzunk a számítógéphez, majd kövessük az alábbi lépéseket (1. ea. 39. dia).

A műveletekhez **root jogra** van szükségünk!

- ▶ Listázzuk a partíciókat:
`fdisk -l`

- ▶ Látjuk a csatolt kötetet:

Disk `/dev/vdc`: **1 GiB**, 1073741824 bytes, 2097152 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

```
ubuntu@proba:~$ sudo -i
root@proba:~# fdisk -l
Disk /dev/vda: 20 GiB, 21474836480 bytes, 41943040 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: B3B11BF8-A9BB-42DF-92CE-8AF0ACFEFFFF5

Device        Start      End  Sectors  Size Type
/dev/vda1    227328    41943006 41715679 19.9G Linux filesystem
/dev/vda14     2048     10239     8192    4M BIOS boot
/dev/vda15    10240    227327    217088  106M EFI System

Partition table entries are not in disk order.

Disk /dev/vdc: 1 GiB, 1073741824 bytes, 2097152 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
root@proba:~# _
```

Kötet csatolása Linux rendszerhez

- ▶ Partícionáljuk a merevlemezt:
`fdisk /dev/vdc`
- ▶ Új partíciót szeretnénk létrehozni:
Command (m for help): `n`
- ▶ A partíció típusa elsődleges (primary):
Select (default p): `p`
- ▶ A partíció száma:
Partition number (1-4, default 1): `1`

```
Welcome to fdisk (util-linux 2.31.1).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x32d235c2.

Command (m for help): n
Partition type
   p   primary (0 primary, 0 extended, 4 free)
   e   extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-2097151, default 2048): 2048
Last sector, +sectors or +size{K,M,G,T,P} (2048-2097151, default 2097151): 2097151

Created a new partition 1 of type 'Linux' and of size 1023 MiB.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

root@proba:~# _
```

Végül írjuk be a számokat, amiket kiír így a teljes merevlemez méretet használhatjuk.

Kötet csatolása Linux rendszerhez

- ▶ Formázzuk a merevlemez: `mkfs.ext4 /dev/vdc`
- ▶ Erősítsük meg a szándékunkat:
Proceed anyway? (y, N) `y`
- ▶ Készítünk egy könyvtárat a meghajtónak:
`mkdir /data`
- ▶ Végül csatoljuk a lemezt a könyvtárhoz:
`mount /dev/vdc /data`
- ▶ Ha szeretnénk leválasztani a kötetet, a következőképp tehetjük meg:
`umount /dev/vdc`

```
root@proba:~# mkfs.ext4 /dev/vdc
mke2fs 1.44.1 (24-Mar-2018)
Found a dos partition table in /dev/vdc
Proceed anyway? (y,N) y
Creating filesystem with 262144 4k blocks and 65536 inodes
Filesystem UUID: 3acc07b4-2d11-491e-8e7d-6e84b80549ec
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376

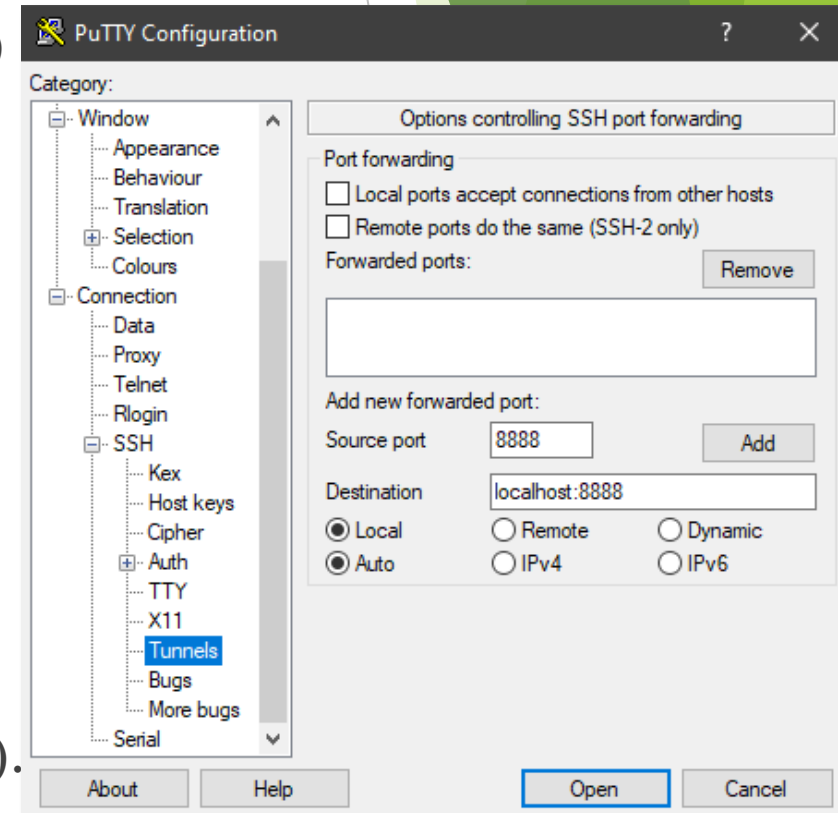
Allocating group tables: done
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
```



Jupyter telepítése

Jupyter telepítése Ubuntu-ra

- ▶ A virtuális gép létrehozását követően ellenőrizzük a következőket:
 - ▶ Rendelkezzen Floating IP-vel a gép (1. ea. 50. dia)
 - ▶ Engedélyezzük a szükséges portokat a Security Group-oknál (1. ea. 47.)
 - ▶ Rendelkezünk saját kulcspárral (1. ea. 53. dia)
- ▶ Windows környezetben indítsuk el a PuTTY programot:
 - ▶ Végezzük el azokat a beállításokat, mint a Linux csatlakozásnál
 - ▶ Tallózzuk be a privát kulcsunkat a **Connection** > **SSH** > **Auth** helyen
 - ▶ Az **SSH** > **Tunnels** fülön a **Source port**-hoz írjuk be a **8888**-as portot
 - ▶ A **Destination**-höz írjuk be a **localhost:8888**-as címet
 - ▶ Végül nyomjuk meg az **Add** gombot
- ▶ Ezekkel a beállításokkal csatlakozzunk a számítógépre (Open gomb).



Jupyter telepítése Ubuntu-ra

- ▶ Telepítsük fel az **Anaconda**-t:

- ▶ Lépünk be a tmp mappába:

```
cd /tmp
```

- ▶ Letöltjük a megfelelő telepítőt:

```
curl -O https://repo.anaconda.com/archive/Anaconda3-2020.07-Linux-x86_64.sh
```

- ▶ Végül elindítjuk a telepítést:

```
bash Anaconda3-2020.07-Linux-x86_64.sh
```

- ▶ Telepítéskor először el kell olvasni a felhasználási feltételeket, itt **ENTER**-eket kell nyomni. Ezután megkérdezi, hogy megfelel-e az alapértelmezett telepítési helynek a `/home/ubuntu/anaconda3` könyvtár. Végül megvárjuk, hogy befejeződjön a telepítés, majd írjuk be hogy **yes**, azaz inicializálja a telepítő az Anaconda-t.

- ▶ Aktiváljuk az Anaconda környezetet:

- ▶

```
source ~/.bashrc
```


Jupyter telepítése Ubuntu-ra

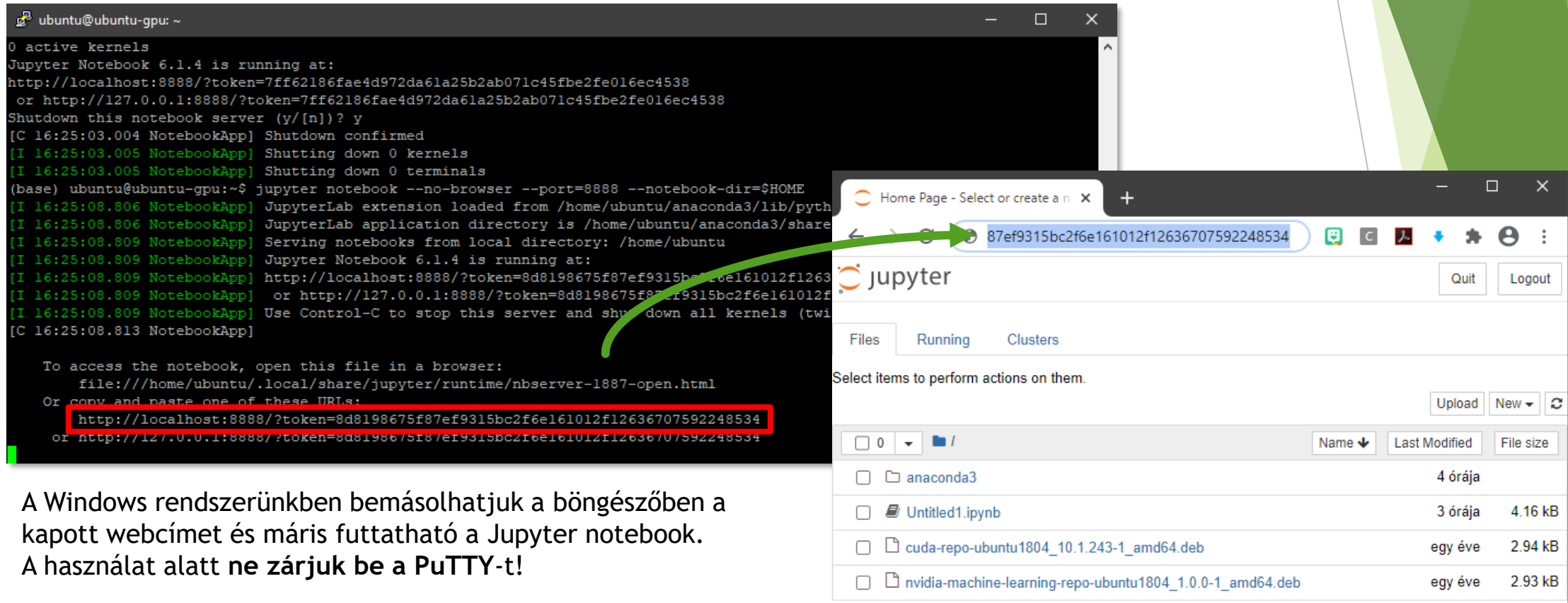
- ▶ Most már használhatjuk a conda parancsokat. Frissítsük fel a komponenseket:
 - ▶ `conda update conda`
 - ▶ `conda update anaconda`
 - ▶ `conda update python`

- ▶ Végül indítsuk el a Jupyter notebook-ot a következő utasítással:
 - ▶ `jupyter notebook --no-browser --port=8888 --notebook-dir=$HOME`

A port a PuTTY-ban beállított portszám lesz, míg a notebook könyvtár az alapértelmezett felhasználói mappa, azaz a `/home/ubuntu/`

Jupyter telepítése Ubuntu-ra

- ▶ Másoljuk ki az Ubuntu által legenerált címet



The image shows a terminal window on the left and a web browser on the right. The terminal window displays the output of the `jupyter notebook` command, including the URL `http://localhost:8888/?token=7ff62186fae4d972da61a25b2ab071c45fbe2fe016ec4538` and the instruction to copy and paste one of the URLs. The web browser shows the Jupyter notebook interface, with the URL `http://localhost:8888/?token=8d8198675f87ef9315bc2f6e161012f12636707592248534` in the address bar. A green arrow points from the terminal output to the browser address bar.

```
ubuntu@ubuntu-gpu: ~
0 active kernels
Jupyter Notebook 6.1.4 is running at:
http://localhost:8888/?token=7ff62186fae4d972da61a25b2ab071c45fbe2fe016ec4538
or http://127.0.0.1:8888/?token=7ff62186fae4d972da61a25b2ab071c45fbe2fe016ec4538
Shutdown this notebook server (y/[n])? y
[C 16:25:03.004 NotebookApp] Shutdown confirmed
[I 16:25:03.005 NotebookApp] Shutting down 0 kernels
[I 16:25:03.005 NotebookApp] Shutting down 0 terminals
(base) ubuntu@ubuntu-gpu:~$ jupyter notebook --no-browser --port=8888 --notebook-dir=$HOME
[I 16:25:08.806 NotebookApp] JupyterLab extension loaded from /home/ubuntu/anaconda3/lib/pyth
[I 16:25:08.806 NotebookApp] JupyterLab application directory is /home/ubuntu/anaconda3/share
[I 16:25:08.809 NotebookApp] Serving notebooks from local directory: /home/ubuntu
[I 16:25:08.809 NotebookApp] Jupyter Notebook 6.1.4 is running at:
[I 16:25:08.809 NotebookApp] http://localhost:8888/?token=8d8198675f87ef9315bc2f6e161012f12636707592248534
or http://127.0.0.1:8888/?token=8d8198675f87ef9315bc2f6e161012f12636707592248534
[I 16:25:08.809 NotebookApp] Use Control-C to stop this server and shut down all kernels (twi
[C 16:25:08.813 NotebookApp]

To access the notebook, open this file in a browser:
file:///home/ubuntu/.local/share/jupyter/runtime/nbserver-1887-open.html
Or copy and paste one of these URLs:
http://localhost:8888/?token=8d8198675f87ef9315bc2f6e161012f12636707592248534
or http://127.0.0.1:8888/?token=8d8198675f87ef9315bc2f6e161012f12636707592248534
```

Home Page - Select or create a n x +

87ef9315bc2f6e161012f12636707592248534

jupyter Quit Logout

Files Running Clusters

Select items to perform actions on them.

Upload New ↕

	Name	Last Modified	File size
<input type="checkbox"/>	0		
<input type="checkbox"/>	anaconda3	4 órája	
<input type="checkbox"/>	Untitled1.ipynb	3 órája	4.16 kB
<input type="checkbox"/>	cuda-repo-ubuntu1804_10.1.243-1_amd64.deb	egy éve	2.94 kB
<input type="checkbox"/>	nvidia-machine-learning-repo-ubuntu1804_1.0.0-1_amd64.deb	egy éve	2.93 kB

A Windows rendszerünkben bemásolhatjuk a böngészőben a kapott webcímet és máris futtatható a Jupyter notebook. A használat alatt ne zárjuk be a PuTTY-t!

A munka befejeztével a terminálban a **CTRL + C** kombinációval tudjuk leállítani a notebook szerveret.

Összefoglalás

- ▶ Csatlakozás Linux alapú távoli számítógépre
- ▶ Kötetek menedzselése Linux-on
- ▶ A Jupyter telepítése Linux rendszeren



A mai előadássorozat felépítése

- ▶ 9:00 - 9:20 Bevezetés
- ▶ 9:20 - 10:25 Az ELKH Cloud bemutatása
- ▶ 10:25 - 10:35 Kávészünet
- ▶ 10:35 - 11:15 Linux használata az ELKH Cloudon
- ▶ **11:15 - 12:00 Windows használata az ELKH Cloudon**