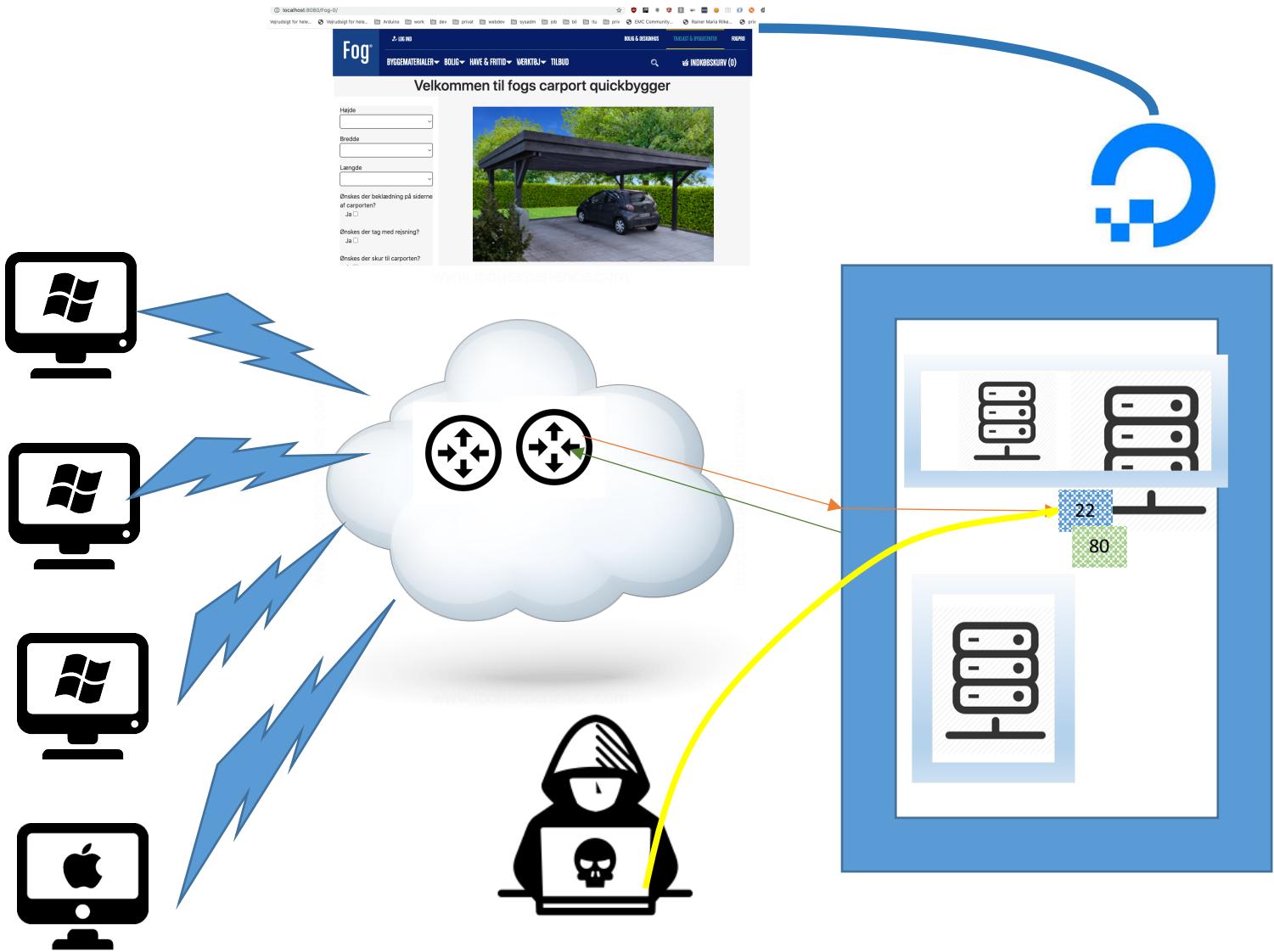




# Operativsystemer Linux Digital Ocean





## Velkommen til fogs carport quickbygger

Højde

Bredde

Længde

Ønskes der beklædning på siderne af carporten?

Ja

Ønskes der tag med rejsning?

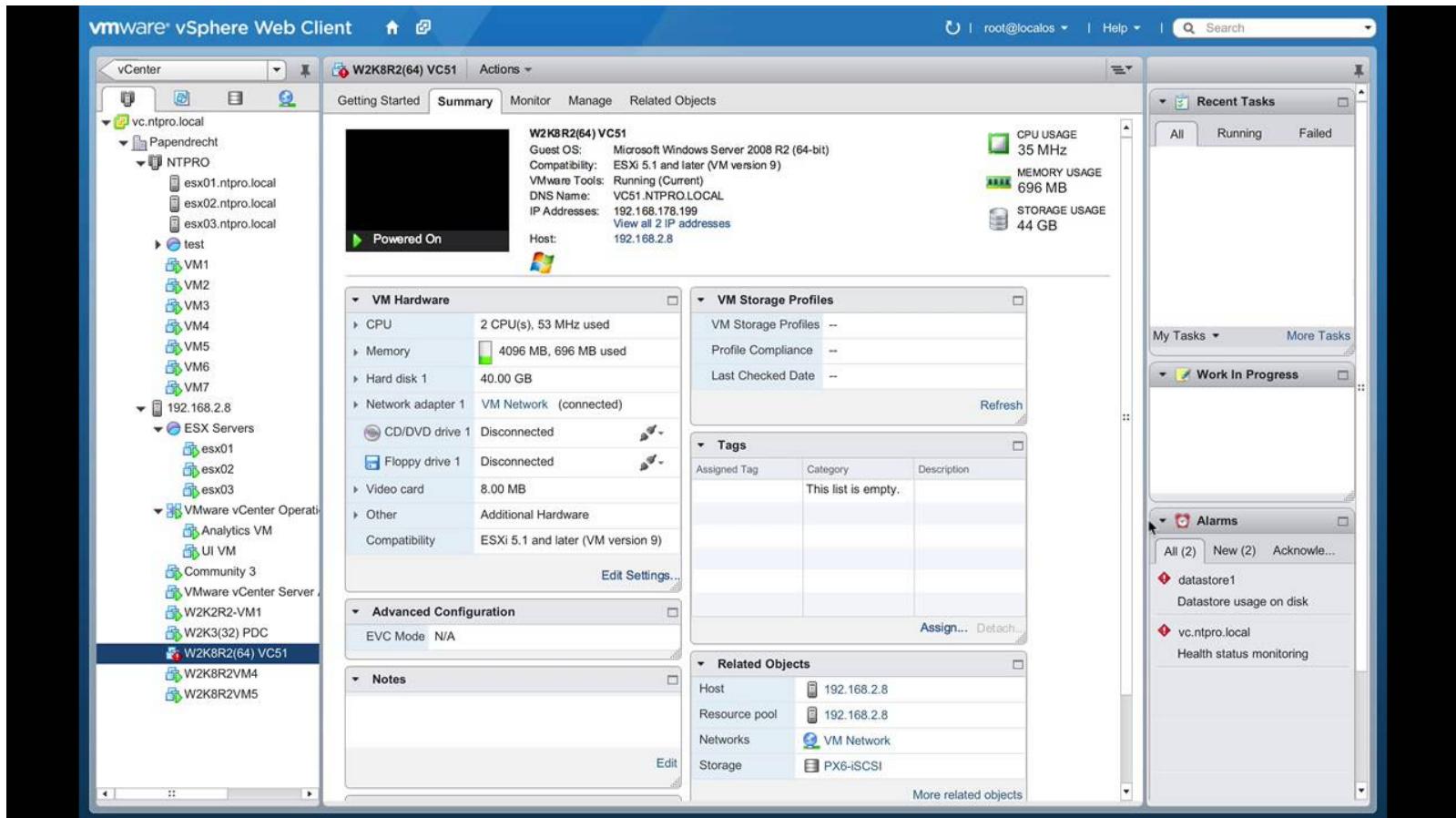
Ja

Ønskes der skur til carporten?

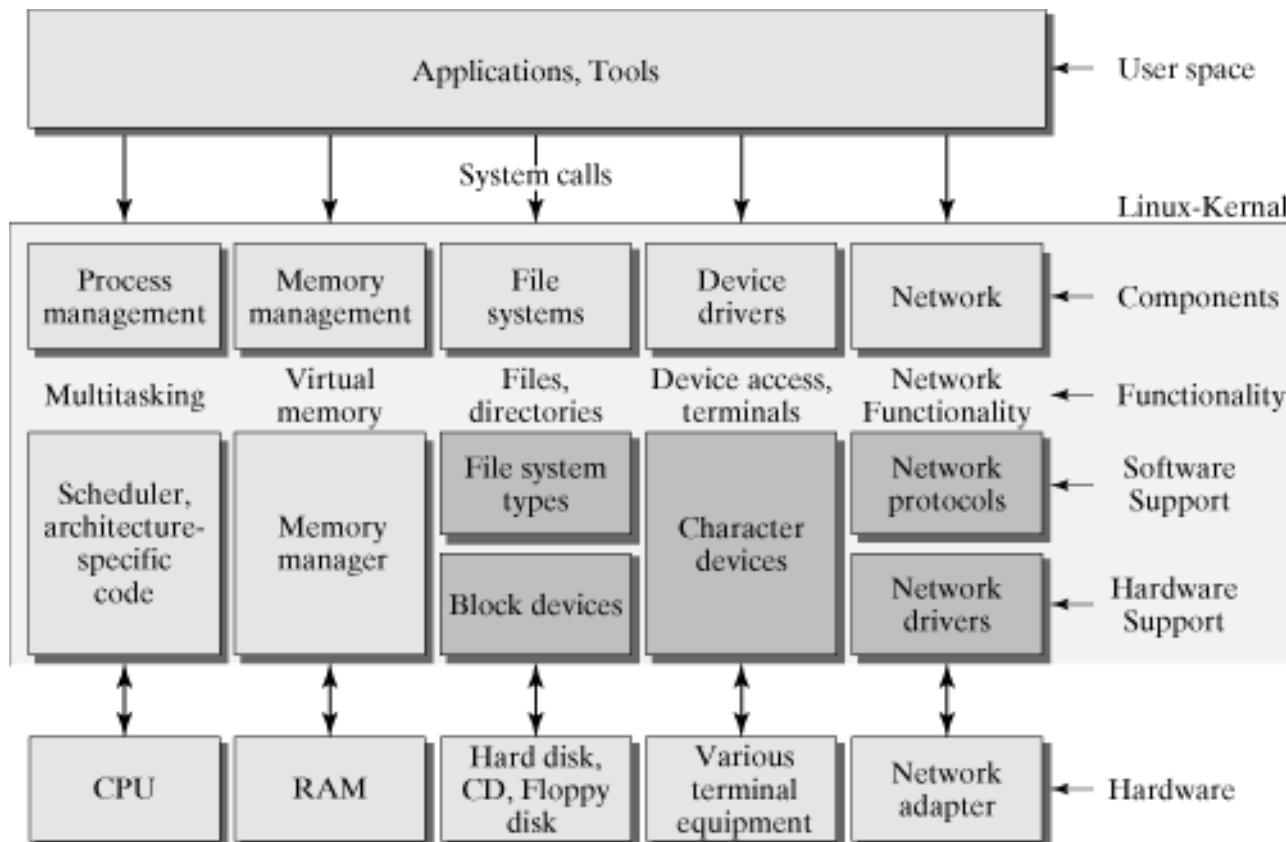
Ja

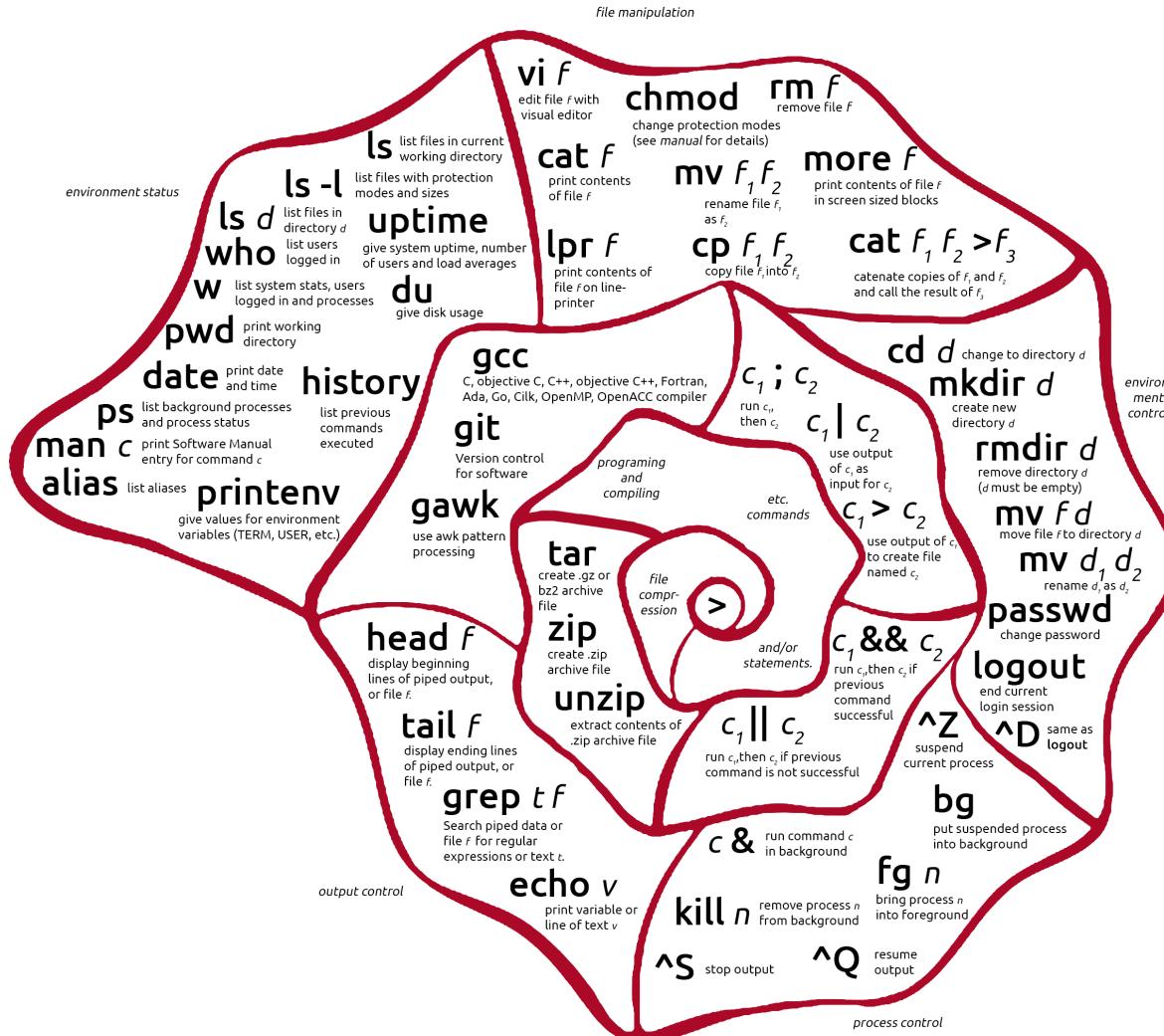
**Bestil**

# System landskab – Hypervisor og virtuelle maskiner



# Computer System Structure







# Digital Ocean, SSH og nøgler

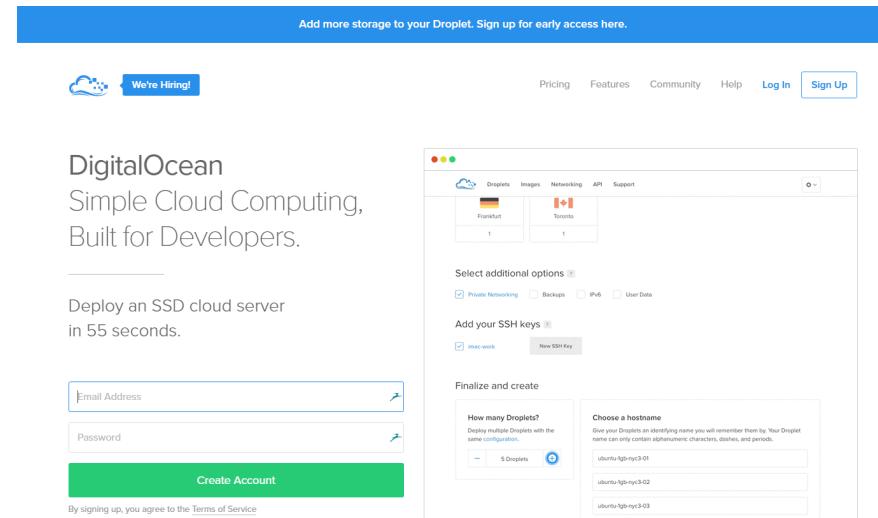
Linux in the cloud

# Digital Ocean

There are many providers which offer a Linux machine in the cloud.

For this set of exercises we recommend access to an Ubuntu Linux machine at digital ocean.

One machine can have several users logged in at the same time, and the cheapest version should do just fine.



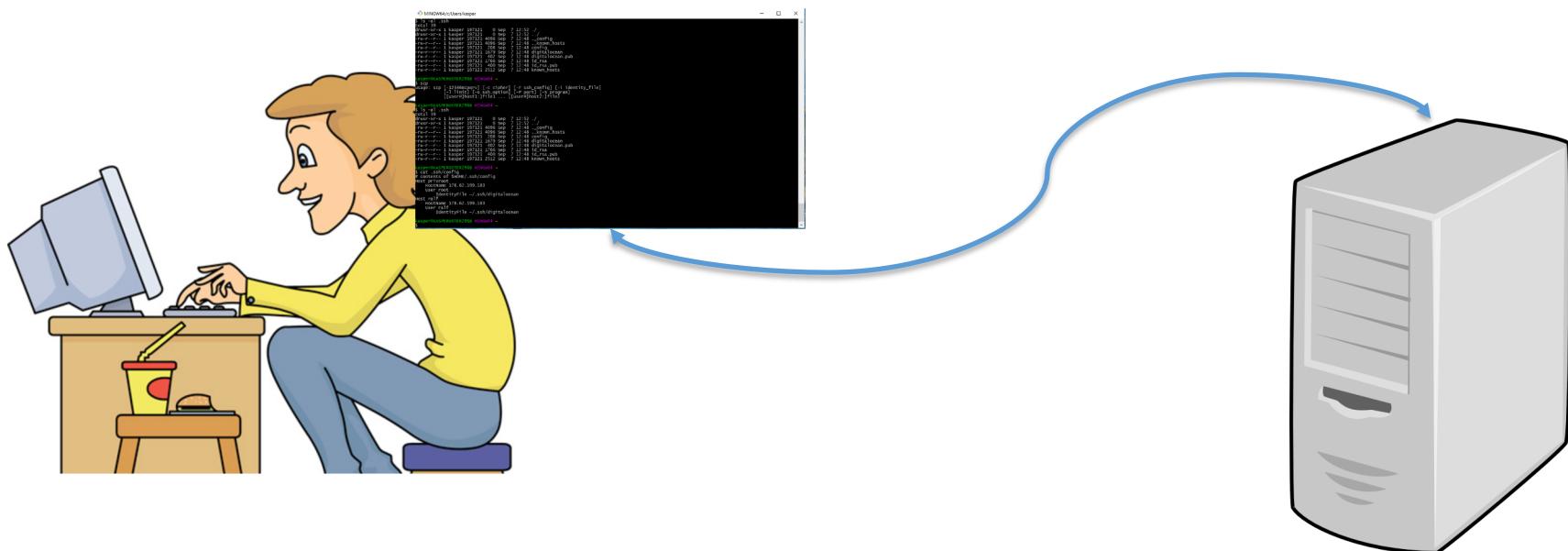
The image shows the DigitalOcean website interface. At the top, there's a blue header bar with the text "Add more storage to your Droplet. Sign up for early access here." Below the header, there's a navigation bar with links for Pricing, Features, Community, Help, Log In, and Sign Up. The main content area features the DigitalOcean logo and the tagline "Simple Cloud Computing, Built for Developers." It includes a section for deploying an SSD cloud server in 55 seconds, with fields for Email Address and Password, and a prominent green "Create Account" button. To the right, there's a "Select additional options" section with checkboxes for Private Networking, Backups, IPv6, and User Data, and a "SSH keys" section where a key named "iMac-work" is listed. At the bottom, there's a "Finalize and create" section with a "How many Droplets?" dropdown set to "5 Droplets" and a "Choose a hostname" input field containing "ubuntu-1gb-nyc3-01", "ubuntu-1gb-nyc3-02", and "ubuntu-1gb-nyc3-03".

# Connecting to a linux server

Connecting to a server using ssh is a two step procedure.

First you run the git-bash program to allow you to have a “shell” program to work on your own computer using a text-only interface

Second, you use the shell to connect to the digitalocean server – the command to do this is called ssh (secure shell).



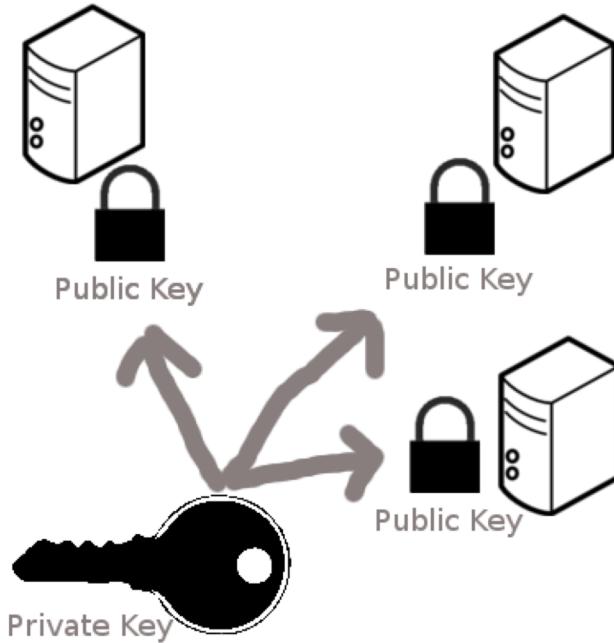
# Public and private key



Private Key



Public Key



- You store the private key on **your** computer
  - The typical place is in your root directory in the folder named ".ssh".
  - The private key is normally called id\_rsa, and the public one called id\_rsa.pub
- The public key is placed on the **remote** computer
  - The public key is placed in the .ssh/authorized\_keys
- You can place the public key on many computers

# Lav nøglepar

I git-bash :

```
MINGW64:/c/Users/vagrant
```

```
vagrant@DESKTOP-F245E69 MINGW64 ~
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/vagrant/.ssh/id_rsa):
```

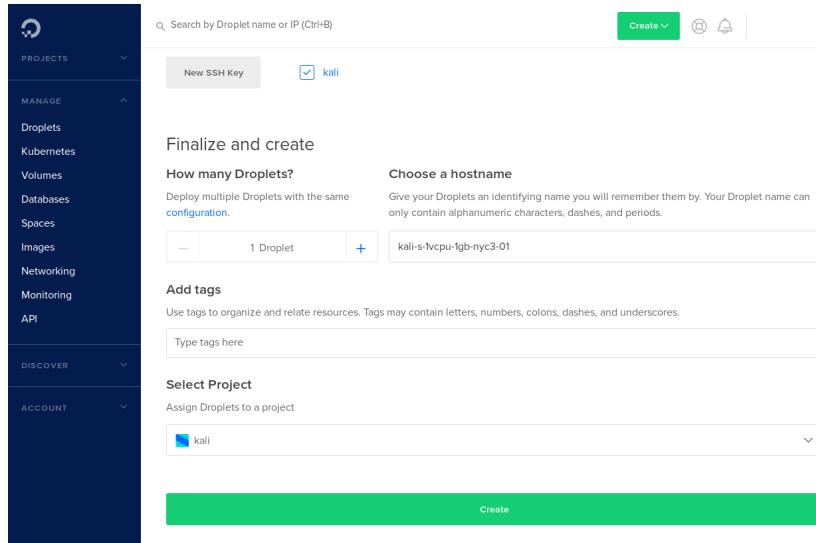
- Nu skal du "catte" indholdet af id\_rsa.pub:

```
$ cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAAB3NzaC1yc2EAAAQABAAABgQDF8r0VqIr56PCJhAd5L+Ru5mBFTLcYLso7p7X3XgNnWe0BmWb8HxiNaK4XDPE+22f/ptXCwdtWkr3JaagDTjHFyD941+
ap179tZWrjTzqo3r9Gr0DYLUVlZu01j/+KyAqhPB1wku0fHCix0bYb00o7pPWDIAwbNaUYcMtKgtiwEELsosTILTpFPu8itWaJV6WmQnf2ndvpdfAb+f7qDaNHQRbeN6EZ
VwmFHu4repPA/1A6Ssk1c3LgykiVRVw/v1LbGZikudv4BhAmMH7hQQKp1zRKKs2RLrUx4R/poLOL8QjJFjtqKsGzxYMB7zk3PaQidKni4aZZZVye04ejJCeI3FZ3z7FO
TjcmsFLDKv7sLR770V3phNAbFwveX4+4A0NQLsH6r+DGftVawV+GYa5j1aC7bzZh1jccWVfgXNiPEF0ILJqMYgtXb1jbqR2z9LIXeduP+bqpR5YdZYldUxHg6tCKGDGtC7
CPsLI1nJfcLqD6ILGeCYg2txm7c4s0U= vagrant@DESKTOP-F245E69
```

- Kopiér og indsæt når du laver din droplet

# Droplet – new or existing?

- Ny droplet



- Eksisterende

- Øvelse. Følg denne vejledning  
<https://www.digitalocean.com/docs/droplets/resources/lost-ssh-key/#enable-password-authentication>

# Logging in using ssh keys

- From git-bash
- `ssh username@ip-address`
- If you saved your key in a file other than `id_rsa`, then you must tell the file name with the key (using `-i`):
- `ssh -i filepathofprivatekey username@ip-address`

# The ssh config file

- If you are logging in and out of the server very often you get tired of remembering the ip-address number and other parameters
- You can have a file named config in the .ssh folder
- Having an ssh file allows us to log in as:
- ssh ralfpriv

```
Host ralfpriv
  HostName 95.85.40.235
  User ralf
  IdentityFile ~/.ssh/digitalocean
```



## Linux Commands

# File Management

In Linux are three basic types of files

- **Ordinary Files** – An ordinary file is a file on the system that contains data, text, or program instructions. In this tutorial, you look at working with ordinary files.
- **Directories** – Directories store both special and ordinary files. For users familiar with Windows or Mac OS, UNIX directories are equivalent to folders.
- **Special Files** – Some special files provide access to hardware such as hard drives, CD-ROM drives, modems, and Ethernet adapters. Other special files are similar to aliases or shortcuts and enable you to access a single file using different names.

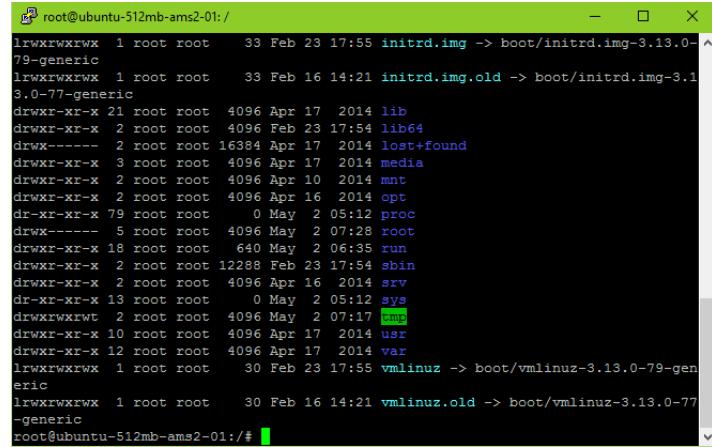
# File Management – Listing files

To list the files and directories stored in the current directory.

Use this command – **ls**

Here is the information about all the listed columns

- **First Column:** represents file type and permission given on the file. Below is the description of all type of files.
- **Second Column:** represents the number of memory blocks taken by the file or directory.
- **Third Column:** represents owner of the file. This is the Unix user who created this file.
- **Fourth Column:** represents group of the owner. Every Unix user would have an associated group.
- **Fifth Column:** represents file size in bytes.
- **Sixth Column:** represents date and time when this file was created or modified last time.
- **Seventh Column:** represents file or directory name.



```
root@ubuntu-512mb-ams2-01:/  
lrwxrwxrwx 1 root root    33 Feb 23 17:55 initrd.img --> boot/initrd.img-3.13.0-79-generic  
lrwxrwxrwx 1 root root    33 Feb 16 14:21 initrd.img.old --> boot/initrd.img-3.13.0-77-generic  
drwxr-xr-x 21 root root 4096 Apr 17 2014 lib  
drwxr-xr-x  2 root root 4096 Feb 23 17:54 lib64  
drwxr-xr-x  2 root root 16384 Apr 17 2014 lost+found  
drwxr-xr-x  3 root root 4096 Apr 17 2014 media  
drwxr-xr-x  2 root root 4096 Apr 10 2014 mnt  
drwxr-xr-x  2 root root 4096 Apr 16 2014 opt  
dr-xr-xr-x  79 root root     0 May  2 05:12 proc  
drwxr-xr-x  5 root root 4096 May  2 07:28 root  
drwxr-xr-x 18 root root   640 May  2 06:35 run  
drwxr-xr-x  2 root root 12288 Feb 23 17:54 sbin  
drwxr-xr-x  2 root root 4096 Apr 16 2014 srv  
dr-xr-xr-x 13 root root     0 May  2 05:12 sys  
drwxrwxrwt  2 root root 4096 May  2 07:17 tmp  
drwxr-xr-x 10 root root 4096 Apr 17 2014 usr  
drwxr-xr-x 12 root root 4096 Apr 17 2014 var  
lrwxrwxrwx 1 root root    30 Feb 23 17:55 vmlinuz --> boot/vmlinuz-3.13.0-79-generic  
lrwxrwxrwx 1 root root    30 Feb 16 14:21 vmlinuz.old --> boot/vmlinuz-3.13.0-77-generic  
root@ubuntu-512mb-ams2-01:/#
```

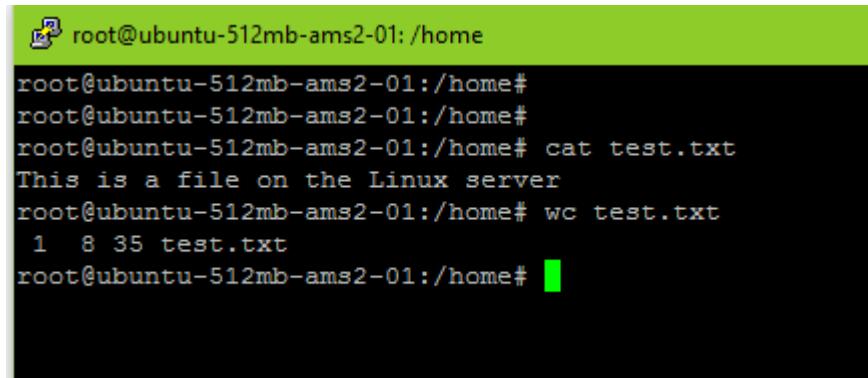
# Display content of a file

You can use **cat** command to see the content of a file.

```
cat test.txt
```

Count the numbers of words in a file is very easy just use

```
wc test.txt
```



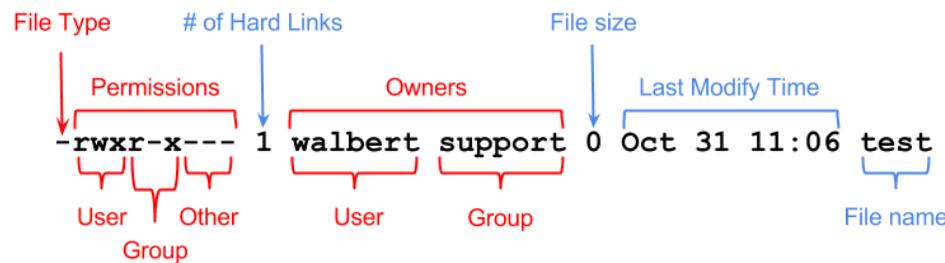
```
root@ubuntu-512mb-ams2-01:/home
root@ubuntu-512mb-ams2-01:/home#
root@ubuntu-512mb-ams2-01:/home# cat test.txt
This is a file on the Linux server
root@ubuntu-512mb-ams2-01:/home# wc test.txt
1 8 35 test.txt
root@ubuntu-512mb-ams2-01:/home#
```

## Directory Related Commands

- **cp** – Copy of file - `cp source_file destination_file`
- **mv** – Renaming - `mv old_file new_file`
- **rm** – Delete – `rm filename`
- **cd** – Change dir – `cd~ (home dir)` – `cd-` (last dir)
- **mkdir** – Create directory - `mkdir dirname`
- **rmdir** – Remove directory – `rmdir dirname`
- **pwd** – print working directory - `pwd`
- **cd** – Changing directory – `cd dirname`

# File permissions

If the command `ls -l` is given, a long list of file names is displayed. The first column in this list details the permissions applying to the file.



The **chmod** command changes the permission on a given file or directory.

**chmod** sets permissions in two ways.

- Using symbols
- Using octal values

Octal	Symbol	Permission
0	---	No Permissions
1	--x	Execute
2	-w-	Write
3	-wx	Write and Execute
4	r--	Read
5	r-x	Read and Execute
6	rw-	Read and Write
7	rwx	Read, Write, and Execute

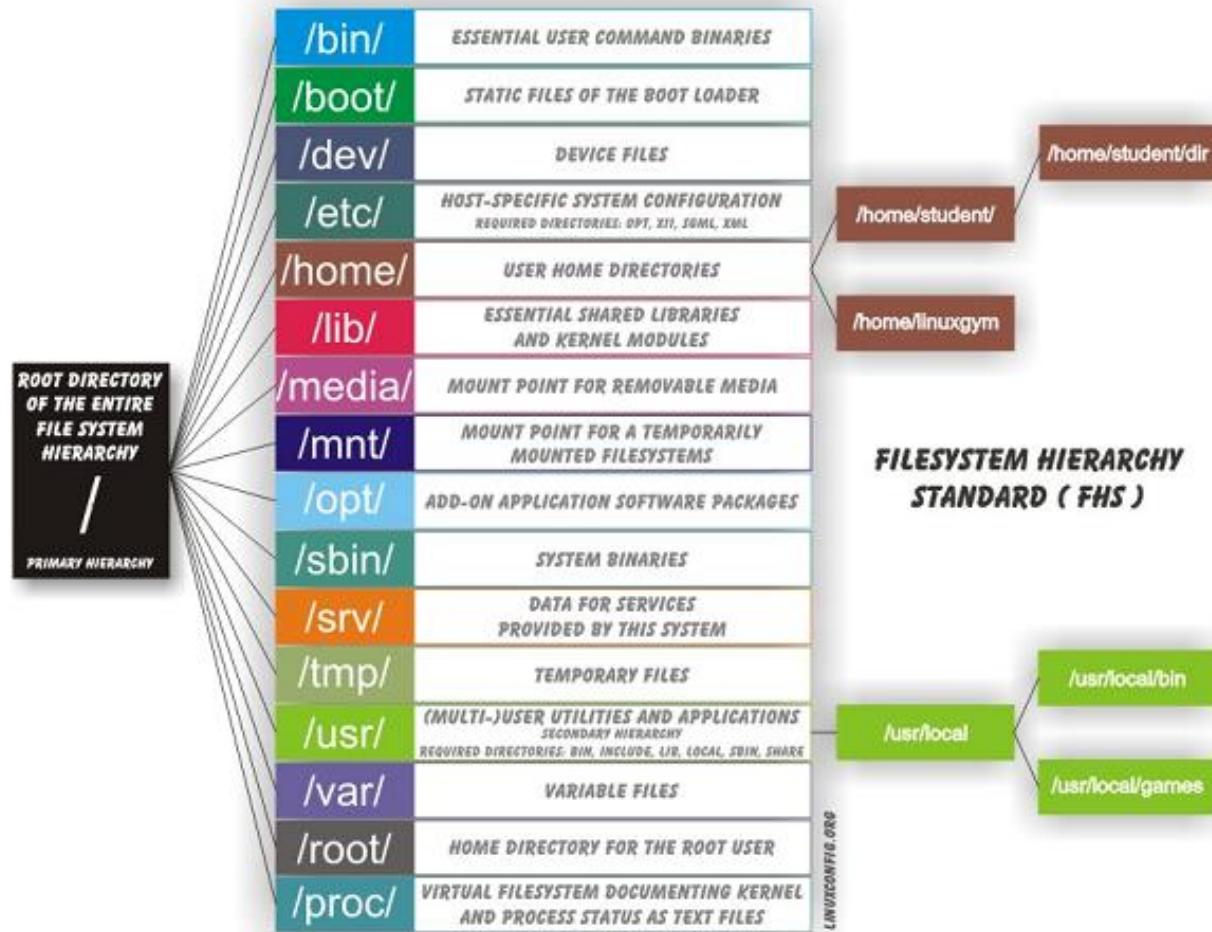
# Process Related Commands

- **ss**  
Obtain a listing of processes and their id's. Including the option aux will show all processes.
- **top**  
provides an ongoing look at processor activity in real time. It displays a listing of the most CPU-intensive tasks on the system, and can provide an interactive interface for manipulating processes.
- **netstat**  
Print network connections, routing tables, interface statistics, masquerade connections, and multicast memberships
- **pstree**  
shows running processes as a tree
- **kill**  
send signal to a process
- **who**  
who am I - Display information about the user

# Linux file system layout

This is a layout from a **Ubuntu** system.

Depending on the system admin, the operating system and the mission of the UNIX machine, the structure may vary, and directories may be left out or added at will.



# Subdirectories of the root directory

Directory	Content
/bin	Common programs, shared by the system, the system administrator and the users.
/boot	The startup files and the kernel, vmlinuz. In some recent distributions also grub data. Grub is the GRand Unified Boot loader and is an attempt to get rid of the many different boot-loaders we know today.
/dev	Contains references to all the CPU peripheral hardware, which are represented as files with special properties.
/etc	Most important system configuration files are in /etc, this directory contains data similar to those in the Control Panel in Windows
/home	Home directories of the common users.
/initrd	(on some distributions) Information for booting. Do not remove!
/lib	Library files, includes files for all kinds of programs needed by the system and the users.
/lost+found	Every partition has a lost+found in its upper directory. Files that were saved during failures are here.
/misc	For miscellaneous purposes.
/mnt	Standard mount point for external file systems, e.g. a CD-ROM or a digital camera.
/net	Standard mount point for entire remote file systems
/opt	Typically contains extra and third party software.
/proc	A virtual file system containing information about system resources. More information about the meaning of the files in proc is obtained by entering the command man proc in a terminal window. The file proc.txt discusses the virtual file system in detail.
/root	The administrative user's home directory. Mind the difference between /, the root directory and /root, the home directory of the root user.
/sbin	Programs for use by the system and the system administrator.
/tmp	Temporary space for use by the system, cleaned upon reboot, so don't use this for saving any work!
/usr	Programs, libraries, documentation etc. for all user-related programs.
/var	Storage for all variable files and temporary files created by users, such as log files, the mail queue, the print spooler area, space for temporary storage of files downloaded from the Internet, or to keep an image of a CD before burning it.



## Exercises

# Exercises

- Opret en bin-folder i dit homedir
  - Lav en fil – test.sh - med følgende indhold
  - Hvad skal der til for at eksekvere filen?
  - Modificér så filnavnet kun bliver myFile.<dato>
- Opret en "tmp"-folder i dit homedir
  - Hent cars.csv fra [github.com/cphstud/20m9596V2-uge5](https://github.com/cphstud/20m9596V2-uge5)
  - Flg kommando giver antal biler sorteret på bilmærke
    - cat cars.csv | cut -d\; -f2 | sort | uniq -c | sort -k1n
  - Modificer kommandoen så den i stedet sorterer på antal cylindre. Hvor mange biler har 6 cyl?
  - Brug grep til at finde ud af hvor mange Audi A6 der er i filen
  - Kan nogen forklare hvorfor grep "A6 3.0" cars.csv viser tre biler mens grep "A6 3.0" kun viser to?
- Find ud af om du har en .bashrc fil i dit home-dir (og hvis ikke så lav en ny)
- Tør du <https://gist.github.com/zachbrowne/8bc414c9f30192067831fafebd14255c> ?
- Hvis ja ..
  - Så ændre/tilføj alias til find så den (grep) ignorerer case.
  - Find ud af hvorfor whatismyip ikke virker og få den til at virke.
- Hvis nej
  - tilføj et alias "lt" som udfører "ls -ltra"
  - Tilføj et alias "psg" som udfører "ps -afe | grep -i "
  - Tilføj et alias for history
  - Tilføj et alias som grepper i din history
- Gå ned i /var/log og "kig" i syslog. Find en "mystisk" ip-adresse og tjek den med whois.

```
echo date  
sleep 3  
echo `date`  
echo `date +%d%m%y_%H%M`  
myDate=`date +%d%m%y_%H%M%S`  
echo "Today is $myDate"  
echo "Today is $myDate" > myFile.$myDate
```



## Resources

# Resources – Lynda.com og cisco's netacad

## Learn the Linux Command Line: The Basics

<https://www.lynda.com/Linux-tutorials/Learn-Linux-Command-Line-Basics/435539-2.html>

<https://www.netacad.com/portal/learning>

The screenshot shows a web browser displaying a Lynda.com course page. The main content area shows a terminal window with a Linux command-line session. The user is installing packages and navigating through a file tree. A large play button icon is overlaid on the terminal window. Below the terminal, there's a navigation bar with 'Contents' and 'Notebook' tabs, and a search bar. To the right, there's an 'Overview' section with a thumbnail of the instructor, Scott Simpson, and course details: 'Updated 3/24/2016 Released 3/24/2016'. It also includes a description of the course, which covers the basics of working with the Linux command line using the Bash shell, focusing on practical Linux commands with examples that help you navigate through the file and folder structure, edit text, and set permissions. Scott also discusses some of the common command-line tools, such as grep, awk, and sed, and command-line editors such as the nano and Vim text editors. The course wraps up with a look at how to install and update software via the Linux command line. The bottom right corner shows statistics: 'Skill Level Beginner', '1h 43m Duration', and '61,194 Views'.