

CentOS - Linux

Navn - Jonas Barigo Østergaard

Uddannelse - Datatekniker med speciale i Programmering

Instruktør - Simon Nicolas El Hanafi



CentOS 8.2

The Community ENTERprise Operating System

Indhold

Indledning	3
Dokumentation.....	4
Linux kerne og distribution.....	4
Desktop Miljø	5
CentOS og Ubuntu	5
Områder hvor Linux bliver anvendt.....	6
Beskrivelse af ændringer og installerede programmer	6
Dokumentation af opsætning, konfiguration og anvendte kommandoer	8
Konklusion	16
Logbog	17

Indledning

Formålet med denne opgave, er at få en introduktion til Linux og hvordan operativ systemet bliver anvendt.

I denne opgave har jeg fået stillet en case, hvori jeg skal konfigurere en pc med Linux systemet CentOS.

På selve computeren skal der desuden installeres en række programmer, som kan bruges i et udviklingsmæssigt sammenhæng. Disse programmer skal desuden installeres igennem terminalen, ved at bruge diverse kommandoer.

Udover at konfigurere Linux pc'en vil jeg i denne opgave også komme ind på hvad Linux kernen og hvad en Linux distribution er, samt hvilke forskelle der er på de to Linux distributioner: CentOS og Ubuntu.

Dokumentation

Linux kerne og distribution

Kernen eller kernel er en vigtig del i alle styresystemer. Kernens formål er at forbinde programmer og hardware sammen, desuden fordeler kernen også computerens ressourcer ud, hvis man har mere end et program åbent ad gangen. Der findes desuden flere forskellige typer af kerner, herunder Monolitiske kerner, Mikrokerner og Hybridkerner.

Monolitiske kerner er kerner, hvor alle funktionerne er tæt kodet sammen, som er afhængige af hinanden. Fordi kernen er så tæt integreret, er det også nødvendigt, at alle funktionerne er implementeret korrekt, hvis de ikke er det vil kernen være ustabil og man kan støde på fatale fejl. Hvis alle funktionerne derimod er programmeret korrekt, kan man opnå en kerne der er meget effektiv i det den kan.

Mikrokerner er en form for kerne, hvor man har valgt, at putte så lidt som muligt ind i computeren, og derved kun har basale funktioner i computerens kerne. Kernen er dog ikke kun i computeren, men ligger også på en server, som computeren tilgår, og på den måde har en fuldt virkende kerne. Denne form for kerne gør det lettere, at lave vedligeholdelser og teste nye udviklinger på kernen, da man har den separat for computeren.

Hybridkerner er lige som Mikrokerner, liggende både på en server og på computeren. Forskellen på Hybridkerner og Mikrokerner er, at man i Hybridkerner har mere funktionalitet i kernen på computeren. Desuden kan man se Hybridkerner som et kompromis mellem Monolitiske- og Mikrokerner.

Linux kernen er desuden en Monolitisk kerne, hvorimod Windows og MacOS kernerne er Hybridkerner. Linux kernen blev udgivet i 1991 af Linus Torvalds, hvorefter GNU operativ systemet adopterede Linux kernen. GNU var desuden ment som et gratis alternativ til UNIX. I dag er Linux mere end bare en kerne, det er et fællesskab af frivillige, hvis fokus er på fri software og dets udvikling.

En Linux distribution er et operativ system der er baseret på Linux kernen. Der findes mange Linux distributioner, som alle specialiserer sig inden for deres eget emne lige fra super computere til smartphones.

Desktop Miljø

Et desktop miljø (de) er et grafisk bruger interface (gui), der gør det muligt at anvende mus og keyboard til at styre computeren via grafik, så som ikoner, toolbars og widgets. Hvis man ikke har et desktop miljø, vil man kun kunne styre computeren fra en terminal, hvori man skriver kommandoer. Til Linux findes der forskellige Desktop Miljøer, som alle har deres eget fokus. Nogle desktop miljøer går op i at computeren skal bruge så få ressourcer som muligt mens andre prøver at give computeren et moderne look, uden at tage stor højde for ressource brug. Nogle af de mest populære desktop miljøer er:

- GNOME – Giver et moderne look, men bruger mange ressourcer.
- Xfce – Gammelt look men bruger få ressourcer.
- KDE – Højt tilpasseligt og bruger en moderat mængde ressourcer.
- LXDE – Fokus på så få ressourcer som muligt.
- Budgie – Moderne look, som bruger en moderat mængde ressourcer.

Nogle Linux Distributioner anvender desktop miljøer som standard, men man er ikke tvunget til at benytte sig af denne standard, man kan selv udskifte sit desktop miljø hvis man vil.

Generelt anvender man Desktop Miljøer til Linux Computere, men ikke til Linux Servers da disse er meget ressource intensive og man hellere vil bruge ressourcerne på noget andet end desktop miljøet.

CentOS og Ubuntu

CentOS er baseret på RHEL, Red Hat Enterprise Linux, og giver et 'enterprise-grade' operativ system gratis. CentOS 8 stoppede desuden udvikling i december 2021, men en gren kaldet CentOS Stream er fortsat i udvikling.

Siden CentOS er beregnet til virksomheder, bliver CentOS udviklet med henblik på stabilitet, derfor følger CentOS udgivelserne af RHEL, som den er baseret på, og har en lang tids support (LTS) på 10 år, desuden udkom der kun en ny større udgivelse mellem 2 – 5 år, dette skete fordi CentOS kun udgav helt stabile versioner. Desuden har CentOS en markedsandel på ca. 17% af alle Linux brugere.

Ubuntu er Baseret på Debian og findes i tre forskellige udgivelser:

- **Desktop** – Bruges på computere og anvendes som et typisk Linux styresystem. Denne udgivelse kommer med et GUI og almindelige programmer til tekstbehandling og video afspilning.
- **Server** - Denne udgivelse bruges til små servere og der bruges ikke noget GUI og dermed er der kun en terminal hvori man kan skrive kommandoer.
- **Core** – Bruges til Internet Of Things (IOT) og har fokus på sikkerhed.

Ubuntu er desuden anvendt af næsten 40% af alle Linux brugere og gør det dermed til det mest udbredte Linux styresystem. Ubuntu har desuden opdateringer ca. hver 6. måned, derudover tilbyder Ubuntu LTS på hver 4. udgivelse og LTS varer i 5 år. Da der er relativ kort opdateringscyklus, er Ubuntu også mindre stabil i forhold til andre Linux styresystemer. Dette medfører at Ubuntu er mindre egnet til virksomheder og mere egnet til brugere der vil have up-to-date udgivelser.

Områder hvor Linux bliver anvendt

Linux kan blive anvendt til næsten alt der har med teknologi at gøre. Linux kan anvendes i alt fra robotter og IOT til store servere der har massevis af data. Der findes næsten et tusinde forskellige Linux distributioner der alle har fokus på noget særligt, derfor er der også utroligt mange ting, som Linux kan blive brugt til.

Beskrivelse af ændringer og installerede programmer

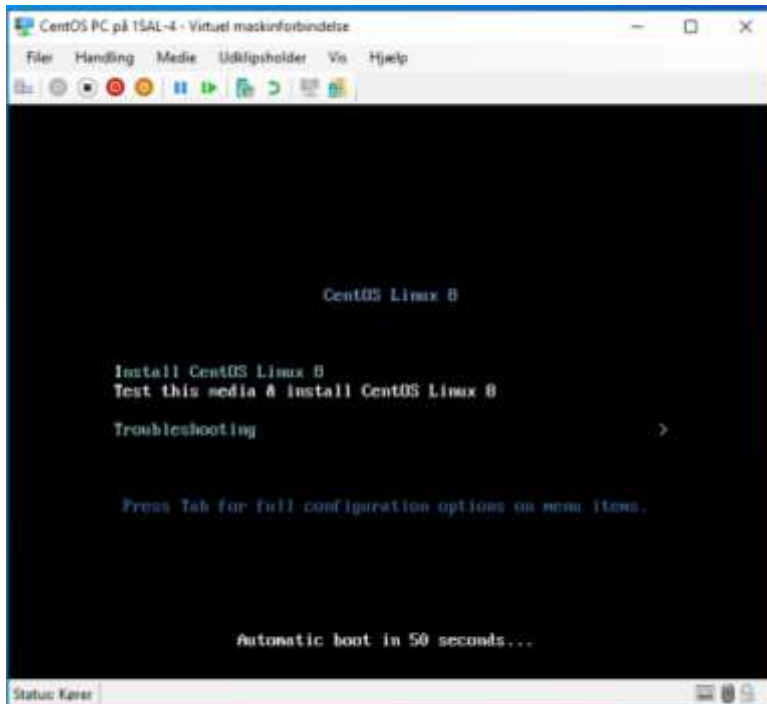
Til denne opgave har jeg deaktiveret SELinux, derudover har jeg skiftet mit mirror til vault.centos.org fordi CentOS 8 er ikke længere i udvikling. Dette skulle gøres, da der ikke var adgang til appstream, som sørger for at der kan installeres programmer på maskinen. Derudover har jeg ændret swappiness til 10 fra standard værdien. Jeg har installeret GNOME tweaks, og tilføjet minner og maksimer knapper, samt slået skrivebords ikoner, applications menu og status indicator til.

Derudover har jeg installeret:

- EPEL – Er en ekstra package med extra software. Da RHEL og CentOS er beregnet til kommercielt brug, er der en del software der ikke kommer med operativsystemet, derfor blev EPEL udviklet.
- Terminator – Er en terminal der kan have flere terminal vinduer i et vindue.
- Development Tools – er en række programmeringsværktøjer inkluderende compilers debuggers mm.
- Chrome Browser – En web browser udgivet af google.
- Samba – sørger for at Linux / Unix maskiner kan kommunikere med Windows maskiner på et netværk.
- Apache HTTP server – Er en open-source http webserver.
- MariaDB – Er en open-source database. Den er desuden lavet af de originale udviklere af MySQL.
- PHP – Et objekt orienteret og server side programmeringssprog.
- PHPMyadmin – Er et stykke software der bruges til at administrere MySQL over internettet.
- Python 3 – Python er et programmeringssprog der kan bruges til en række forskellige ting, herunder hjemmesider, automatisering, data analyse og data visualisering.
- Angular – Angular er en typescript baseret webapplikationsramme.
- React – Er et frontend javascript bibliotek, der bruges til at udvikle interaktive brugergrænseflader (UIs).
- Anaconda – Er en distribution af Python og R programmeringssprogene der bruges til beregningsvidenskab
- Wine – Er et stykke software der tillader windows programmer at blive kørt på Unix-like operativsystemer, så som Linux.
- VMWare Workstation Pro – Er software der bruges til at køre virtuelle computere.

Desuden har jeg oprettet en genvej til at åbne Terminator terminalen, genvejen er Ctrl + Alt + t. Jeg har også skiftet porten på apache serveren fra 80 til 8090, givet gruppen 'apache' ejerskab over mappen /var/www/html, givet mappen 755 rettigheder (read, write, execute) og tilføjet brugeren Admin til apache gruppen. Desuden har jeg oprettet en php fil ved navn phpinfo i mappen /var/www/html, hvor funktionen <?php phpinfo(); ?> står. Hvis denne fil åbnes I localhost (apache server), vil den udskrive information om php på apache serveren.

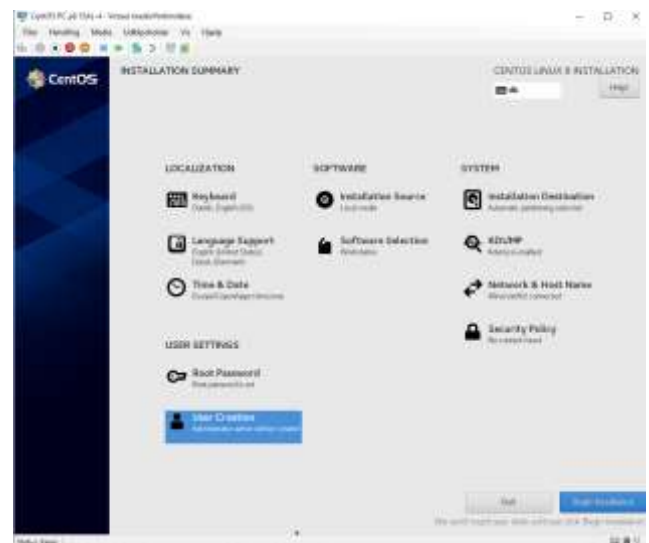
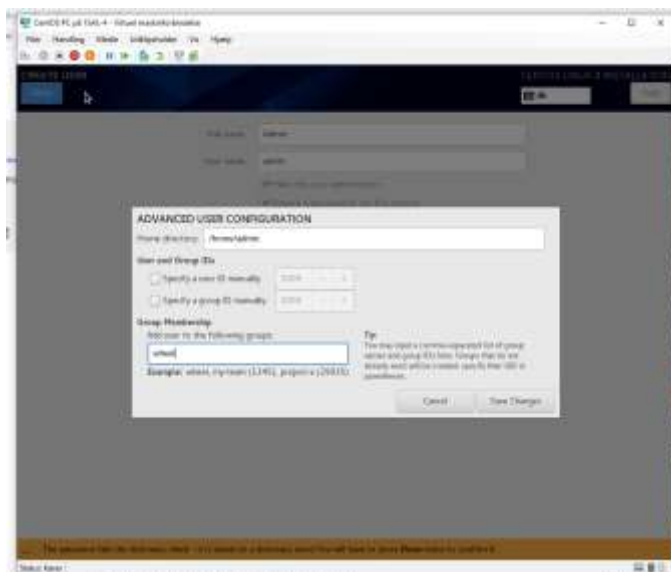
Dokumentation af opsætning, konfiguration og anvendte kommandoer



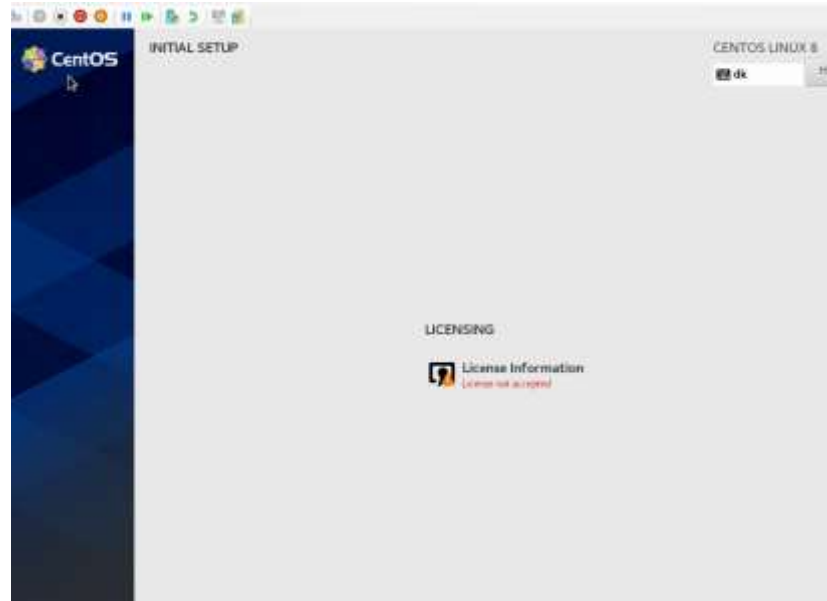
Når man starter CentOS første gang skal man konfigurere sit styresystem. Hvis man vil have en fuld konfiguration skal man trykke på tabulator knappen, ligesom vist på billedet. Her vil man kunne vælge hvilken slags CentOS version man vil benytte f.eks. workstation eller server. Derudover kan man også vælge nogle programmer der skal installeres så som GNOME DE eller Development Tools.

Derudover kan man konfigurere sprog, keyboard layout, tid og data, netværk, sikkerhed, Root Password, Bruger Oprettelse mm.

Under Bruger Oprettelse kan man desuden konfigurere om brugeren skal have sudo rettigheder, dette gøres ved at tilføje brugeren til gruppen wheel.



Når man har valgt sin konfiguration, vil styresystemet starte installations processen, som til sidst vil ende i en licens agreement prompt. Derefter skal man logge ind, hvorefter en hurtig opsætning begyndes. Endelig er man klar til at bruge sin CentOS maskine.



I CentOS finder man sine programmer under activities, her er terminalen en af de essentielle programmer for maskinen. I terminalen kan man udføre kommandoer til at udføre opgaver, så som at installere et program, eller oprette/ændre en fil.

```
[Admin@localhost ~]$ sudo nano /etc/selinux/config
```

Denne kommando bruges til at ændre i filen config, sudo betyder sudo rettigheder nano er et tekstredigerings program og /etc/selinux/config er stien til filen man vil ændre.

```
GNU nano 2.9.8 /etc/selinux/config

# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=enforcing
# SELINUXTYPE= can take one of these three values:
#   targeted - Targeted processes are protected,
#   minimum - Modification of targeted policy. Only selected processes are pr$
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted

[ Read 14 lines ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line
```

Her ses den åbnede fil med nano. I opgaven har jeg fået til opgave at deaktivere SELinux, dette gøres ved at ændre linjen SELINUX=enforcing til SELINUX=disabled. Herefter lukkes filen ved at trykke CTRL + x og gemme filen. I dette tilfælde skal man genstarte maskinen for at ændringerne træder i kraft. For at se om ændringen har fundet sted kan man bruge kommandoen sestatus i terminalen.

```
[Admin@localhost ~]$ sestatus
SELinux status: disabled
[Admin@localhost ~]$
```

Desuden skulle jeg også ændre swappiness til 10 fra 30. Dette gøres på samme måde blot med en anden fil, her skrives vm.swappiness=10.

```
GNU nano 2.9.8 /etc/sysctl.d/99-swappiness.conf

vm.swappiness=10

[ Read 1 line ]
^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text ^T To Spell  ^_ Go To Line
```

Man kan desuden se om ændringen er sket med kommandoen "sysctl vm.swappiness".

```
[Admin@localhost ~]$ sysctl vm.swappiness
vm.swappiness = 10
[Admin@localhost ~]$
```

Derefter skulle jeg opdatere CentOS, dette ville jeg kunne have gjort med en simpel "sudo yum update" kommando, men da CentOS stoppede udvikling i december 2021 virkede dette ikke længere. Jeg blev derfor nødt til at ændre metadata mirroret til vault.centos.org. Dette gøres ved at bruge følgende kommandoer:

```
[Admin@localhost ~]$ cd /etc/yum.repos.d/
[Admin@localhost yum.repos.d]$
```

```
[Admin@localhost yum.repos.d]$ sudo sed -i 's/mirrorlist/#mirrorlist/g' /etc/yum
.repos.d/CentOS-*
```

```
[Admin@localhost yum.repos.d]$ sudo sed -i 's|#baseurl=http://mirror.centos.org|
baseurl=http://vault.centos.org|g' /etc/yum.repos.d/CentOS-*
```

```
[Admin@localhost yum.repos.d]$ sudo yum update -y
```

```

Installed:
  kernel-4.18.0-348.7.1.el8_5.x86_64
  kernel-core-4.18.0-348.7.1.el8_5.x86_64
  kernel-devel-4.18.0-348.7.1.el8_5.x86_64
  kernel-modules-4.18.0-348.7.1.el8_5.x86_64
```

Complete!

```
[Admin@localhost yum.repos.d]$
```

Derefter kunne jeg bruge yum til at installere og opdatere som jeg ville.

For at installere Gnome Tweaks brugte jeg kommandoen "sudo yum install gnome-tweaks"

```
[Admin@localhost ~]$ sudo yum install gnome-tweaks
[sudo] password for Admin:
Last metadata expiration check: 0:00:53 ago on Tue 01 Mar 2022 02:16:11 PM CET.
Dependencies resolved.
=====
Package                                Arch    Version                               Repository    Size
=====
Installing:
  gnome-tweaks                          noarch  3.28.1-7.el8                         appstream     331 k
Installing dependencies:
  gnome-shell-extension-user-theme      noarch  3.32.1-22.el8_5                     appstream     27 k
Transaction Summary
=====
Install 2 Packages

Total download size: 359 k
Installed size: 1.4 M
Is this ok [y/N]: y
```

Her skrev jeg y for at bekræfte installationen, hvorefter installationen begyndte. Derefter kunne jeg åbne Gnome tweaks fra activities ligesom jeg åbnede terminalen.

Da jeg skulle installere Chrome Browseren via en RPM fil, skulle jeg først hente rpm filen og derefter installere den. Dette gøres ved en "wget {fil-link}" kommando der henter en fil fra et given link, efterfulgt af en "sudo yum localinstall {filnavn}" kommando for at installere filen.

```
[Admin@localhost ~]$ wget https://dl.google.com/linux/direct/google-chrome-stable_current_x86_64.rpm
[Admin@localhost ~]$ sudo yum localinstall google-chrome-stable_current_x86_64.rpm
```

Derudover skulle jeg også installere en apache server. Dette gjordes på følgende måde:

```
Admin@localhost:~$ sudo yum -y install httpd
```

Derefter blev jeg i min opgave bedt om at bruge "find" kommandoen til at finde filen httpd.conf fil. Dette gjorde jeg ved at bruge kommandoen "sudo find / -name httpd.conf".

```
[Admin@localhost ~]$ sudo find / -name httpd.conf
find: '/proc/1356/task/1356/net': Invalid argument
find: '/proc/1356/net': Invalid argument
find: '/run/user/1000/gvfs': Permission denied
/etc/httpd/conf/httpd.conf
/usr/lib/tmpfiles.d/httpd.conf
[Admin@localhost ~]$
```

Her fandt jeg mappen `"/etc/httpd/conf"` som `"httpd.conf"` ligger i. Derefter brugte jeg nano til at ændre i filen for at skifte port nummeret fra 80 til 8090.

Derudover skulle jeg give apache gruppen ejerskab over mappen `/var/www/html`. Dette gjorde jeg på følgende måde:

```
[Admin@localhost ~]$ sudo chown -R apache:apache /var/www/html
```

Derudover gav jeg også mappen 755 rettigheder (read, write, execute).

```
[Admin@localhost ~]$ sudo chmod -R 755 /var/www/html
```

Og tilføjede brugeren Admin til gruppen apache:

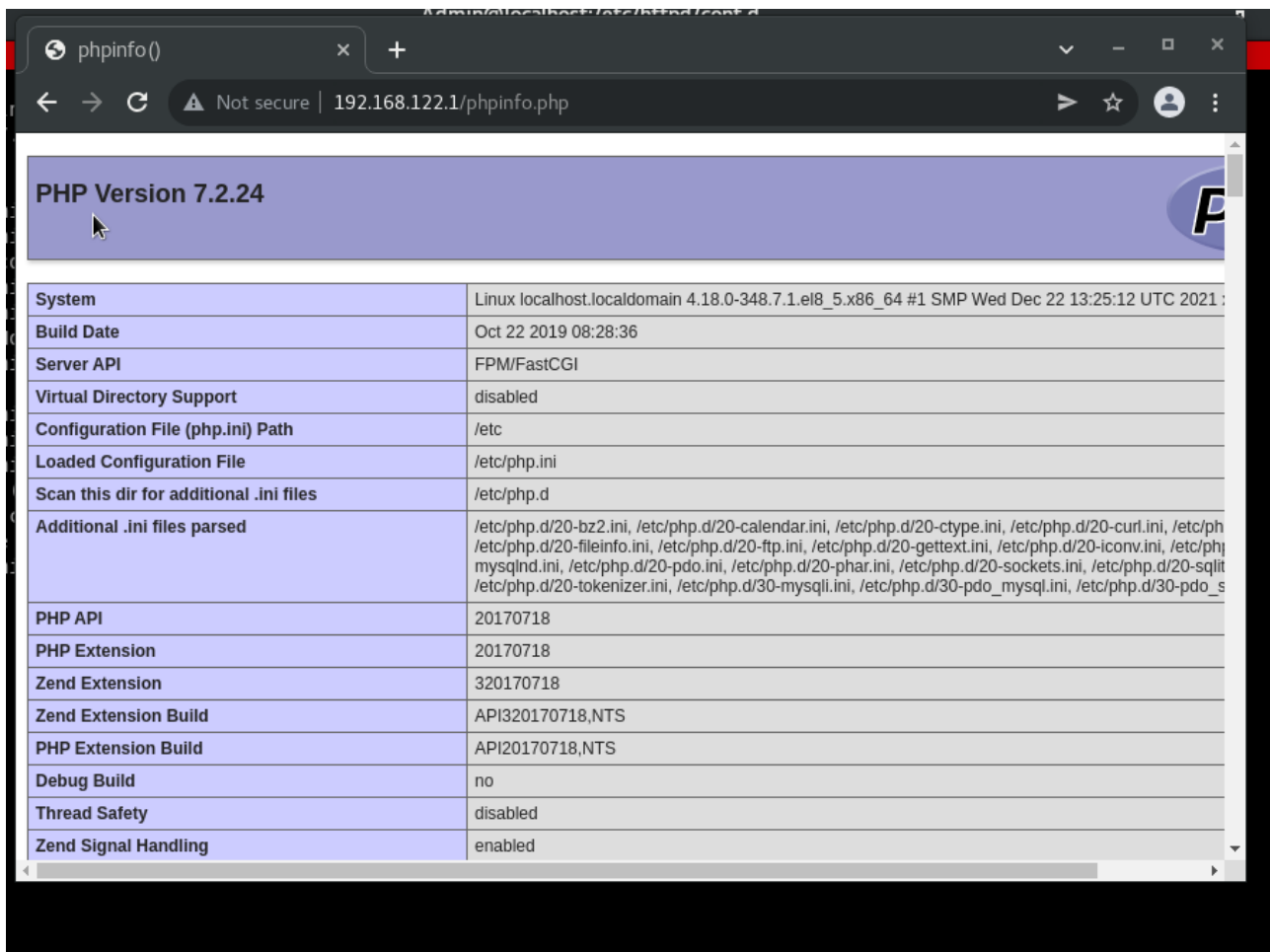
```
[Admin@localhost ~]$ sudo usermod -g apache Admin
```

Desuden oprettede jeg filen `phpinfo.php` i mappen `/var/www/html` efter at jeg havde installeret php. I denne fil skrev jeg funktionen `"php phpinfo();"` med nano kommandoen.

```
Admin@localhost: /var/www/html 80x24
[Admin@localhost ~]$ cd /var/www/html
[Admin@localhost html]$ touch phpinfo.php
touch: cannot touch 'phpinfo.php': Permission denied
[Admin@localhost html]$ sudo touch phpinfo.php
[Admin@localhost html]$ ls
phpinfo.php
[Admin@localhost html]$ sudo nano phpinfo.php
```

```
GNU nano 2.9.8
<?php phpinfo(); ?>
```

På localhost, der kører igennem apache, kunne jeg tilgå {ip-address}/phpinfo.php for at få denne side frem:



PHP Version 7.2.24

System	Linux localhost.localdomain 4.18.0-348.7.1.el8_5.x86_64 #1 SMP Wed Dec 22 13:25:12 UTC 2021
Build Date	Oct 22 2019 08:28:36
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/20-bz2.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-fileinfo.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-imagick.ini, /etc/php.d/20-ldap.ini, /etc/php.d/20-mbstring.ini, /etc/php.d/20-mcrypt.ini, /etc/php.d/20-mysqlnd.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-sqlite3.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/30-mysqli.ini, /etc/php.d/30-pdo_mysql.ini, /etc/php.d/30-pdo_sqlite.ini
PHP API	20170718
PHP Extension	20170718
Zend Extension	320170718
Zend Extension Build	API320170718,NTS
PHP Extension Build	API20170718,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled

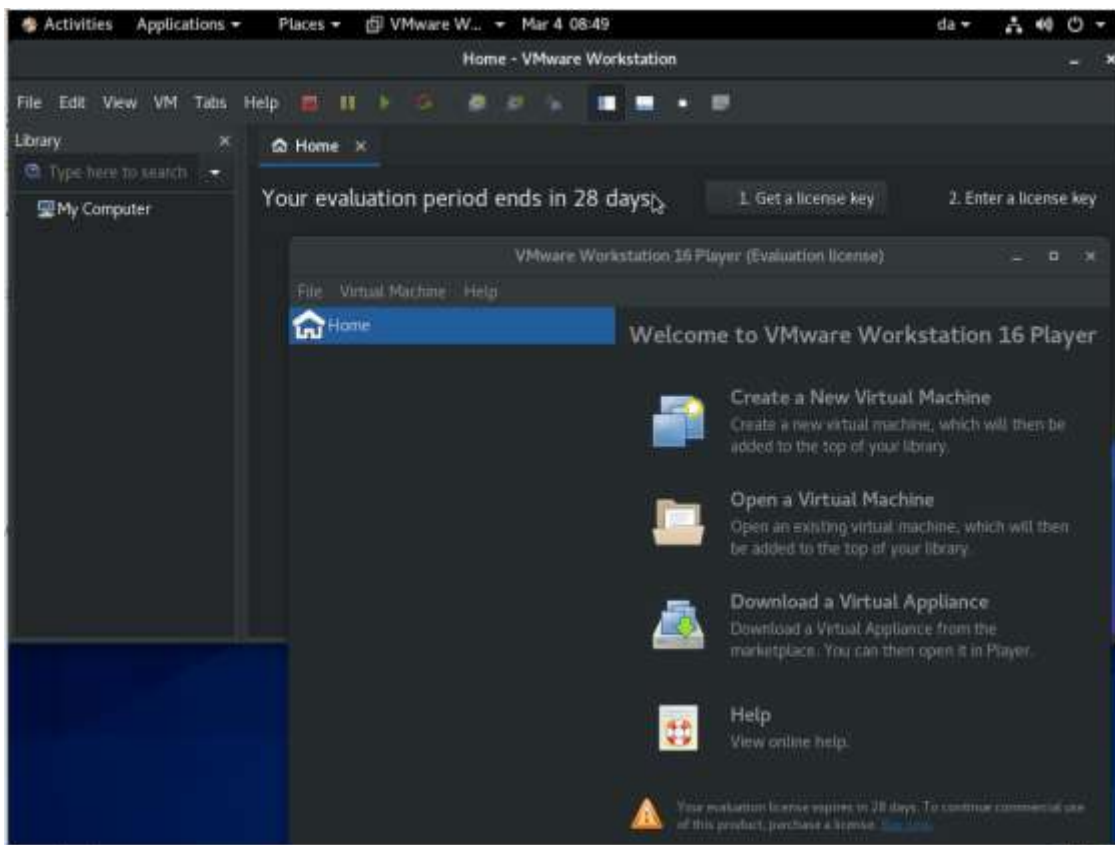
Denne side kommer frem fordi php filen bliver kørt, med andre ord er dette outputtet af funktionen `phpinfo()`;

Da jeg skulle installere VMWare løb jeg ind i nogle problemer da jeg skulle åbne programmet. Her ville vmmon & vmnet ikke compile og jeg kunne derfor ikke færdiggøre installationen af programmet.

Jeg endte med at finde en løsning på dette, ved at køre følgende script:

```
#!/bin/bash
VMWARE_VERSION=workstation-12.5.9
TMP_FOLDER=/tmp/patch-vmware
rm -fdr $TMP_FOLDER
mkdir -p $TMP_FOLDER
cd $TMP_FOLDER
git clone https://github.com/mkubeczek/vmware-host-modules.git
cd $TMP_FOLDER/vmware-host-modules
git checkout $VMWARE_VERSION
git fetch
make
sudo make install
sudo rm /usr/lib/vmware/lib/libz.so.1/libz.so.1
sudo ln -s /lib/x86_64-linux-gnu/libz.so.1
/usr/lib/vmware/lib/libz.so.1/libz.so.1
sudo /etc/init.d/vmware restart
```

Dette script cloner et github repo, som er nødvendigt for, at installere VMWare.



Konklusion

Jeg har I dette forløb lært hvordan man konfigurere CentOS, samt hvordan man anvender terminalen til, at installere programmer, oprette filer, redigere i filer, konfigurere installerede programmer, samt hvad de forskellige programmer jeg i denne opgave har skulle installere gør.

Derudover har jeg tilegnet mig viden omkring hvad, Linux er, og hvad det står for, samt nogle af de forskellige Linux distributioner der findes. Desuden har jeg også tilegnet mig viden omkring de forskellige typer af Kerner der findes, hvilke der bliver anvendt til Linux, samt hvad et Desktop Miljø er.

Logbog

28-02-2022

I dag startede jeg ud med at opsætte min centos pc og rode lidt rundt med terminalen. Derudover brugte jeg også en del tid på at gennemgå min c# kode og sammenligne den med andres.

01-03-2022

I dag nåede jeg til at installere apache serveren på min centos pc. Derudover brugte jeg igen tid på at sammenligne en af mine gamle opgaver, sql opgaven, med andre.

02-03-2022

I dag blev jeg færdig med at konfigurere centos pc'en. Jeg stødte især på problem med at få min apache server til at køre et php script i stedet for kun at vise det. Derudover stødte jeg også på problemer da jeg skulle installere vm-ware. Her blev jeg nødt til at køre et script fordi jeg ikke kunne compile vmmon og vmnet. Scriptet hentede nogle nødvendige filer fra et github repository. Scriptet skal køres hver gang min kernel bliver opdateret.

Derudover begyndte jeg på min rapport, hvor jeg skrev indledningen.

03-03-2022

I dag skrev jeg rapport. Her nåede jeg at skrive om Linux kerne og distribution, desktop miljøer, CentOS og Ubuntu, Områder hvor linux bliver anvendt og beskrivelse af ændringer og installerede programmer.

Desuden startede jeg på at skrive min dokumentation for konfigurerings og kommandoer.

04-03-2022

Jeg begyndte dagen med at skrive resten af min dokumentation færdig, samt at skrive min konklusion og færdiggøre denne Logbog.

Derefter var jeg færdig med CentOS opgaven og jeg besluttede mig for at uploade mine ting til github.