

---

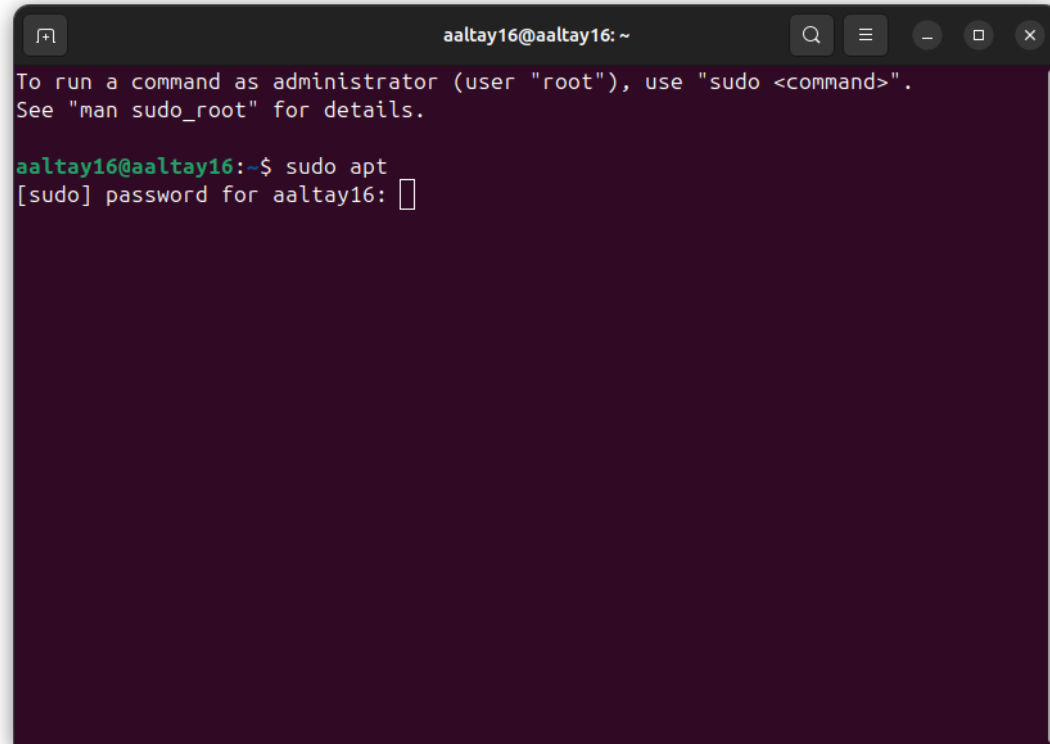
# Nettverk og Tjenester

*Oppdrag 1*

---

# Log og dokumentasjon

- I uke 37 startet vi opp Raspberry Pi og installerte nødvendige komponenter for å sette opp en personlig delt mappe via Samba, slik at filer enkelt kunne deles mellom Raspberry Pi og Windows.



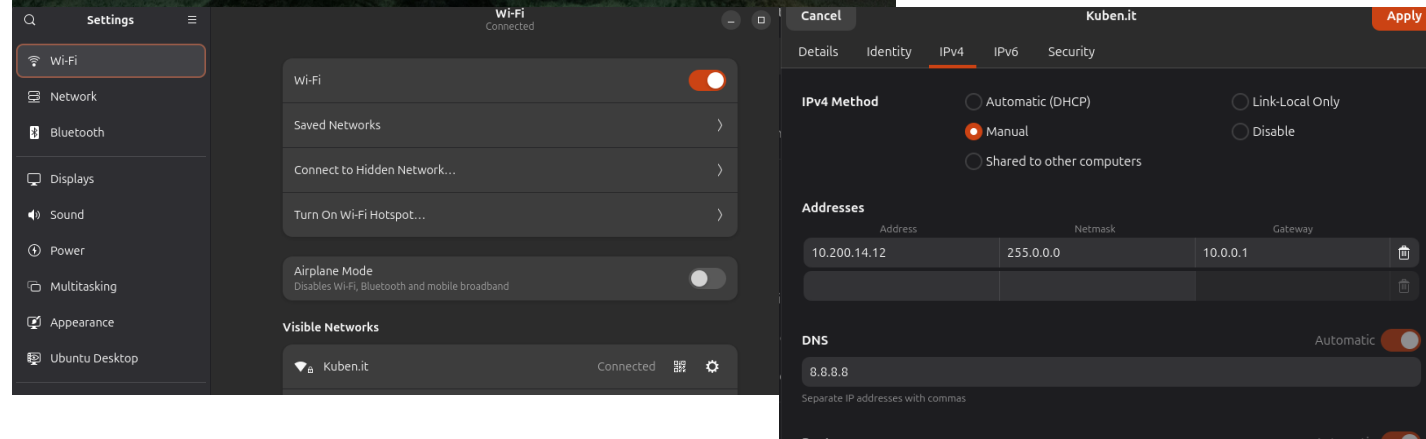
```
aaltay16@aaltay16: ~  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
aaltay16@aaltay16:~$ sudo apt  
[sudo] password for aaltay16: 
```

# Komme i gang

- Deretter fulgte jeg den oppgitte manualen og gjennomførte nødvendige oppdateringer og installasjoner for å sikre at systemet hadde alle komponenter som trengs for å konfigurere Samba.

Deretter måtte vi endre IP-adressen vår til en individuell og statisk adresse innenfor det lokale nettverket, slik at hver Raspberry Pi kunne identifiseres unikt og være tilgjengelig for fildeling uten konflikter.

```
aaltay16@aaltay16: ~  
: wlan0: <NO-CARRIER,BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel s  
ate DORMANT group default qlen 1000  
  link/ether 2c:cf:67:0b:01:54 brd ff:ff:ff:ff:ff:ff  
  altname wlx2ccf670b0154  
aaltay16@aaltay16:~$ ip a  
: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul  
qlen 1000  
  link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
  inet 127.0.0.1/8 scope host lo  
    valid_lft forever preferred_lft forever  
  inet6 ::1/128 scope host noprefixroute  
    valid_lft forever preferred_lft forever  
: eth0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN group  
default qlen 1000  
  link/ether 2c:cf:67:0b:01:53 brd ff:ff:ff:ff:ff:ff  
: wlan0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gro  
p default qlen 1000  
  link/ether 2c:cf:67:0b:01:54 brd ff:ff:ff:ff:ff:ff  
  altname wlx2ccf670b0154  
  inet 10.200.14.12/8 brd 10.255.255.255 scope global noprefixroute wlan0  
    valid_lft forever preferred_lft forever  
  inet6 fe80::2fd7:cd7e:7fcc:dd9a/64 scope link noprefixroute  
    valid_lft forever preferred_lft forever  
aaltay16@aaltay16:~$ S
```



# Nettverk

- Deretter testet jeg kommunikasjonen mellom enhetene ved å bruke ping, og fortsatte med å sette opp en Apache-webserver og koble sammen Raspberry Pi-terminalen med Windows CMD for å utforske samspillet mellom systemene.

```
aaltay16@aaltay16:~$ sudo apt install apache2
[sudo] password for aaltay16:
Waiting for cache lock: Could not get lock /var/
Waiting for cache lock: Could not get lock /var/
ld by process 4327 (unattended-upgr)
Waiting for cache lock: Could not get lock /var/
Waiting for cache lock: Could not get lock /var/
ld by process 4327 (unattended-upgr)
Waiting for cache lock: Could not get lock /var/
Waiting for cache lock: Could not get lock /var/
Executing: /usr/lib/systemd/systemd-sysv-install enable apache2
aaltay16@aaltay16:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: >
   Active: active (running) since Wed 2025-09-17 09:03:36 CEST; 1min 14s ago
   Invocation: 88d06318ae02427994f72b591fd96116
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 6373 (apache2)
    Tasks: 55 (limit: 8659)
     CPU: 207ms
   CGroup: /system.slice/apache2.service
           └─6373 /usr/sbin/apache2 -k start
             └─6376 /usr/sbin/apache2 -k start
               └─6377 /usr/sbin/apache2 -k start

Sep 17 09:03:36 aaltay16 systemd[1]: Starting apache2.service - The Apache HTTP>
Sep 17 09:03:36 aaltay16 apachectl[6372]: AH00558: apache2: Could not reliably >
Sep 17 09:03:36 aaltay16 systemd[1]: Started apache2.service - The Apache HTTP >
lines 1-16/16 (END)
```

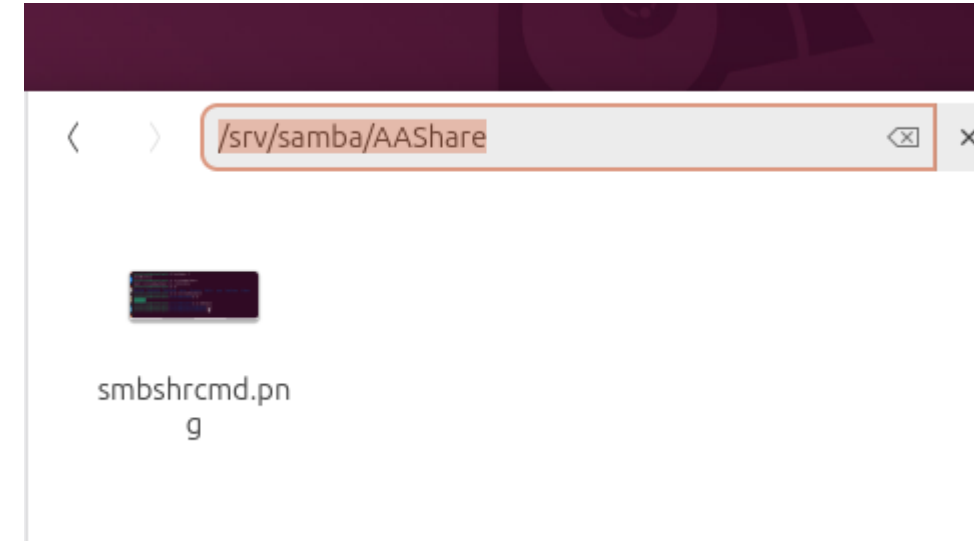


# Sambashare

- Sist, men ikke minst, forsøkte jeg å fullføre fildeling med Samba, men prosessen ble rotete og resultatet var langt fra oppgaven. Til tross for flere forsøk på å rette opp i feilene, fungerte ingenting som det skulle, og jeg fikk lite hjelp underveis. Derfor bestemte jeg meg for å factory resette hele Raspberry Pi og starte prosessen på nytt. Med erfaring fra første runde visste jeg nøyaktig hva jeg skulle gjøre og hvilke feil jeg skulle unngå, og kom raskt i gang igjen. Nå er jeg godt på vei, med et mer strukturert og stabilt oppsett.

**\*Ferdig\***

Nå som jeg endelig er ferdig, og litt mer selvstendig i prosessen, klarte jeg å få alt til å fungere. Jeg opprettet en helt ny delt mappe, konfigurerte den fra bunnen av, satte riktige tillatelser, fant IP-adressen og koblet den til nettverket – og nå fungerer fildelingen som den skal



# Github

- Etter at jeg var ferdig med Samba-oppsettet, gikk jeg videre til å klonе et GitHub repositorium og begynte å jobbe med versjonskontroll. Jeg startet med å skrive en enkel Python kode som viser systeminformasjon fra Raspberry Pi, og deretter logget jeg inn på GitHub og opprettet et eget repository som jeg skal bruke til å lagre og dele prosjekter og filer. Dette markerer overgangen fra lokal fildeling til skybasert samarbeid og dokumentasjon.



The screenshot shows the GitHub website's 'Get started' page. At the top, there are links for 'Set up in Desktop', 'HTTPS', and 'SSH'. Below these, a message says 'Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and a [CONTRIBUTING](#) file.' The main section is titled '...or create a new repository on the command line' and contains a code block with the following commands:

```
echo "# ScriptShrAA" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/AAltayy16/ScriptShrAA.git
git push -u origin main
```

Below this, there is another section titled '...or push an existing repository from the command line' with the following commands:

```
git remote add origin https://github.com/AAltayy16/ScriptShrAA.git
git branch -M main
git push -u origin main
```



The screenshot shows a Python script named 'Clone.py'. The script contains five lines of code that use the 'platform' module to print system information:

```
1 import platform
2
3 print("system:", platform.system())
4 print("machine:", platform.machine())
5 print("release:", platform.release())
```