

Serverhosting lokalt!

0. Lägg till följande på toppen av din node-express-server. Låt säga att den heter server.js. Lägg till på första raden i filen. Du bör också ändra så att servern binder till **port 80** och inte t.ex. 1337.

```
#!/bin/env node
```

1. Kopiera över mappen som innehåller allt du behöver för att köra din server. Använd **scp**. Följande är ett exempel om du vill kopiera över mappen hiscores till din IP address:

```
scp -r hiscores ubuntu@10.206.6.127:
```

2. Sedan använder du ssh till din server. Användarnamnet är ubuntu och lösenordet är raspberry.

```
ssh ubuntu@10.206.6.127
```

3. Sedan ska vi se till att filen server.js har exekveringsrättigheter:

```
chmod +x hiscores/server.js
```

4. Nu ska vi se till att systemd som ser till att servern hålles igång. Byt ut hiscores mot namnet på ditt projekt.

```
sudo vim /lib/systemd/system/hiscores.service
```

5. Klicka i för att börja skriva. Escape för att sluta. skriv :wq för att (w)rite och (q)uit.

6. Pasta in ungefär följande:

```
[Unit]
Description=hiscores server
After=network.target

[Service]
Type=simple
User=root
ExecStart=/usr/bin/node /home/ubuntu/hiscores/server.js
Restart=on-failure
WorkingDirectory=/home/ubuntu/hiscores
[Install]
WantedBy=multi-user.target
```

7. Skriv följande varje gång vi ändrar en .service fil:

```
sudo systemctl daemon-reload
```

8. Slå på så att datorn tar hand om att köra servern:

```
sudo systemctl enable hiscores.service
```

9. Slå på servern (från och med nu ska den startas av sig själv):

```
sudo systemctl start hiscores.service
```

10. Kolla att den körs:

```
sudo systemctl status hiscores.service
```

11. Du kan också nu stänga av servern om du vill:

```
sudo systemctl stop hiscores.service
```

12. Du kan också nu stänga av servern om du vill:

```
sudo systemctl stop hiscores.service
```

13. Du behöver kanske ladda ned och sätta upp din mongodatabas även på din raspberry.

<https://www.mongodb.com/developer/how-to/mongodb-on-raspberry-pi/>