

# MEedia Server on Ubuntu Desktop

February 12, 2011

## Contents

<b>1</b>	<b>Background</b>	<b>2</b>
<b>2</b>	<b>Objective</b>	<b>2</b>
<b>3</b>	<b>This Guide</b>	<b>2</b>
<b>4</b>	<b>Hardware and System Setup</b>	<b>2</b>
4.1	Install Ubuntu 9.10 (base system)	2
4.2	Disable GUI	3
4.3	Sound	3
<b>5</b>	<b>Software</b>	<b>4</b>
5.1	SSH (Remote Access)	4
5.1.1	Automatic Login	4
5.1.2	Desktop-like Access	4
5.2	NFS	4
5.2.1	RW-access Setup	5
5.2.2	Access using unfs3	5
<b>6</b>	<b>Media Software</b>	<b>5</b>
6.1	Music On Console	5
6.2	Stream Music	6
6.3	Streaming Video	6
<b>7</b>	<b>Sharing Software</b>	<b>6</b>
7.1	Torrent	6
7.2	OneSwarm	6
<b>8</b>	<b>Other Useful Information/Tools</b>	<b>7</b>
8.1	Fast and Simple Suspend	7
8.2	Screen Backlight	7
8.3	Reduce Disk Writes	7
8.3.1	Reduce writing of log-files	8
8.4	Convert Music	9
8.4.1	AAC/M4A to MP3	9

8.5 Backup Script . . . . .	10
8.6 Program List Backup . . . . .	11
8.7 Miscellaneous . . . . .	12

## 1 Background

I bought an *Asus Eee-PC 701 4G* in spring 2008 but I realised after some time it was too small to not be used "painfully", at least compared to my MacBook. Since noone is interested of buying that computer I wanted to create a media server of it. The disk space inside is only 4 gigabytes as well as the SD card, i.e. totally 8 gigabytes. However, that's not enough if you want to store music and films. So I bought a *Western Digital Extern HDD My Passport Essential 500GB* since it is small, quiet and does not require an external power source (it is USB-powered). The downside is that it is a bit more expensive than those larger in dimension.

As base system I first went for Ubuntu Server since I am familiar with Ubuntu and I know how to set up most needed software. But after a failure behind the steering wheel, I did a reinstallation and this time I used Ubuntu Desktop 9.10. I later updated it to 10.04 LTS.

## 2 Objective

Following the steps below will do mainly this:

- Allow connection to the server remotely using SSH and NFS.
- Install a terminal music player which will play through the soundcard of the Eee PC, i.e. locally, preferably connected to amplifier and speaker.
- Install OneSwarm so you can listen to streamed music via your browser and share files to your friends.

## 3 This Guide

You don't have to follow this guide from beginning til the end. Of course you'll need the base system, but additional functionality can be added at any time when you get the need for it.

## 4 Hardware and System Setup

### 4.1 Install Ubuntu 9.10 (base system)

Download and install Ubuntu 9.10 using UNetBootin to an USB memory stick. Since I have an SD-card I went for this configuration:

```
/dev/sda1 ext4 (3480 Mb), mounted as /
/dev/sda2 swap (518 Mb)
/dev/sdc1 ext4 (1999 Mb), mounted as /var
/dev/sdc2 swap (2023 Mb) (I don't know for what I will use this now)
```

I first had the external disk attached and it was supposed to be mounted automatically. However, it did not work as expected so I recommend to add it manually afterwards (add in `fstab` as `/media/Titan`). The way I mounted it physically is shown in figure 1.



Figure 1: Front and backside of the Eee Pc with the external drive fixed with adhesive tape! Be careful to have good tape that will not let the drive fall off (which happened to me once!).

## 4.2 Disable GUI

For some reason is the sound level muted on reboot, hence it is necessary to start Gnome, unmute and then play music. So the steps below may not be recommended at this moment.

`gdm` is the graphical login screen when Gnome starts. To disable (if you like), edit the following lines in `/etc/init/gdm.conf`:

```
start on (runlevel to start on (runlevel [3]
stop on runlevel [016] to stop on runlevel [0126]
```

To start X after login, run `startx`.

## 4.3 Sound

Test if sound does work with:

```
$ aplay -l
aplay: device_list:217: no soundcards found...
```

In this case it is not and may be solved by adding the user to the same group as those listed in `/dev/snd/`, probably `audio`. Fix with

```
$ sudo usermod -a -G audio hans
```

then logout and relogin. You should now be able to set volume with `alsamixer`.

## 5 Software

### 5.1 SSH (Remote Access)

Install the *SSH server* with:

```
$ sudo apt-get install openssh-server
```

If you do not want to be logged out automatically after inactivity, add the following to `/etc/ssh/sshd_config`:

```
KeepAlive yes
ClientAliveInterval 60
```

#### 5.1.1 Automatic Login

Based on guide at <https://help.ubuntu.com/8.10/serverguide/C/openssh-server.html>. Generate the keys on remote machine (not server), from a terminal prompt enter:

```
ssh-keygen -t dsa
```

This will generate the keys using a DSA authentication identity of the user. During the process you will be prompted for a password. Simply hit Enter when prompted to create the key.

By default the public key is saved in the file `~/.ssh/id_dsa.pub`, while `~/.ssh/id_dsa` is the private key. Now copy the `id_dsa.pub` file to the remote host and appended it to `~/.ssh/authorized_key`:

```
cat id_dsa.pub >> .ssh/authorized_key
```

Finally, double check the permissions on the `authorized_key` file, only the authenticated user should have read and write permissions.

Uncomment line in `/etc/ssh/sshd_config`:

```
AuthorizedKeysFile      %h/.ssh/authorized_key
```

To disable login without this key, also change to this:

```
PasswordAuthentication no
```

It will now not be possible to login with a password as before, but only with these keys. Don't forget to restart the SSH server:

```
$ sudo /etc/init.d/ssh restart
```

#### 5.1.2 Desktop-like Access

If you have *X* installed you can run it from another computer (similar to VNC) using

```
$ Xnest -geometry 800x600 :1 & DISPLAY=:1 ssh -X -Y -p 22
  USER@REMOTEIP gnome-session
```

## 5.2 NFS

Based on <http://ubuntuforums.org/showthread.php?t=249889>, this will install an NFS server which gives better transfer speeds than mounting as SSH. Since NFS uses the user ID as identification it should be the same on all machines accessing. This will not be the case if you use Mac since the first UID is 501, and on Ubuntu it is 1000. The NFS server `unfs3` is said to be able to handle this, but I could only get read access. If that's enough, the info is given below. But first the standard way.

### 5.2.1 RW-access Setup

Two options exist; either change the UID on one of the machines so they are the same (see `usermod` for more info), or use `anonuid=1000,anongid=1000` in the options of the share in the export file (may be a security risk). Install NFS server:

```
$ sudo apt-get install nfs-kernel-server nfs-common portmap
```

Add share to `/etc/exports`:

```
# Read-only for 192.168.1.1-255:
/media/Titan 192.168.1.0/24(rw,no_root_squash,async,
    no_subtree_check,insecure)
```

Update the file with `sudo exportfs -a` and then restart the server using `sudo /etc/init.d/nfs-kernel-server restart`.

### 5.2.2 Access using unfs3

In this case I'll use `unfs3` since it can handle mappings of user id's, good if your user on local machine has different UID than that on the server. Install with:

```
$ sudo apt-get install unfs3 portmap
```

Add share to `/etc/exports`:

```
# Read-only for 192.168.1.1-255:
/media/Titan 192.168.1.0/24(ro,async,insecure,map_static=/etc/nfs/
    hikari.map)
```

```
# ro = read-only, rw = read-write (however, it does not seem to work
    from my Mac)
# 'insecure' for use with Mac OS X.
# static mapping of UID/GID from http://ubuntuforums.org/showthread.php?p=4387032#post4387032
```

Use `id username` to see username's UID and GID. Then set up file `/etc/nfs/hikari.map` with mappings:

```
# NFS UID and GID mapping for Hikari
#      remote  server
uid    501     1000
gid    20      1000
```

Restart server:

```
$ sudo /etc/init.d/unfs3 restart
```

## 6 Media Software

### 6.1 Music On Console

Install *Music On Console Player*:

```
$ sudo apt-get install moc
```

To start the player (including the server), run `moop`.

This program plays the sound locally, i.e. through the speakers of the *Eee PC* or connect it to an amplifier. It plays FLAC, MP3, OGG and Wave. Information on how to convert AAC to playable format, read [8.4](#).

## 6.2 Stream Music

Enable “remote access” in *OneSwarm*, see section [7.2](#).

## 6.3 Streaming Video

Due to hardware limitations this may not work flawlessly.

Install VLC, FFMPEG and some codecs using

```
$ sudo apt-get install vlc ffmpeg libavcodec-unstripped-52
```

On server terminal, write

```
vlc --sout '#transcode{vcodec=mp4v,acodec=mpga,vb=800,ab=128,
  deinterlace}:rtp{mux=ts,dst=192.168.1.2,sap,name=Video}' -I dummy
  video.avi
```

On client machine (IP: 192.168.1.2), start VLC check the SAP announcements setting. Name “Video” will show up, click to start playing. However, it is streamed and pausing is not possible.

# 7 Sharing Software

## 7.1 Torrent

A small and text-based torrent software is *rTorrent*. Install by

```
$ sudo apt-get install rtorrent
```

User guide found at <http://libtorrent.rakshasa.no/wiki/RTorrentUserGuide>. When logged in to the server by SSH it is very convenient to use the program `screen`. It is then possible to initiate an rtorrent session, then log off (or perform other tasks) and let it continue. The command to use is

```
$ screen rtorrent
```

Hiding the “screen” is done by `Ctrl+a+d`. You may now disconnect the SSH session and resume it another time from any other computer. To resume the screen again, use

```
$ screen -r
```

## 7.2 OneSwarm

Download *OneSwarm*, current version is 0.6, from <http://oneswarm.cs.washington.edu/>. It requires *Java* and a browser, for example:

```
$ sudo apt-get install sun-java6-bin
```

It is possible to use the *X virtual frame buffer* which doesn't need to run *X.org*. The downside is that you can currently not add swarms manually (only automatically). But if you've set up remote access you can as always access your swarms. To install the *X virtual frame buffer*, run

```
$ sudo apt-get install xvfb
```

It is then possible to run OneSwarm from the terminal using

```
Xvfb :1 & env DISPLAY=:1 OneSwarm
```

Using *Xvfb* saves resources compared to using normal *X*. But if you need to manually add files, stop OneSwarm, start normal *X* with `startx`, open *Firefox* and then add files. Hopefully *OneSwarm* developers will implement functionality for adding files remotely soon.

## 8 Other Useful Information/Tools

### 8.1 Fast and Simple Suspend

Following the steps below, the suspend and resume of the Eee 701 is done in a few seconds.

In the power management settings (menu: System / Preferences / Power Management) you can set the computer goes to sleep (suspend) whenever the power-button is pressed. The default is to ask the user.

When you turn the computer back on, it will by default ask for password. It can be disabled by using `gconf-editor`. Run it from the terminal or using `Alt+F2`. Browse to `/apps/gnome-power-manager/lock`. Set (disable) the following:

```
suspend = false
use_screensaver_settings = false
```

Source: <http://superuser.com/questions/71465/how-to-disable-screen-locking-after-sleep-on-ubuntu-9-10>

### 8.2 Screen Backlight

Turn the backlight of the LCD on or off (NB: it will not turn on automatically by keyboard or mouse use!!):

```
$ sudo vbetool dpms on
$ sudo vbetool dpms off
```

### 8.3 Reduce Disk Writes

Make files written to `/tmp` only be in RAM instead, hence these will be lost when computer shutdowns or reboots. Add this line to `/etc/fstab`:

```
tmpfs          /tmp           tmpfs          defaults        0      0
```

### 8.3.1 Reduce writing of log-files

Reduce disk writes by putting the `/var/log` directory into RAM, and save it to disk only on halt or reboot, and eventually every `n`:th hour or so.

Add this line to `/etc/fstab`:

```
tmpfs      /var/log      tmpfs      defaults      0      0
```

Create the Upstart jobs `logsave` and `logload`, save to `/etc/event.d/`. `logsave` reads:

```
start on runlevel [06]
start on logsave
```

```
script
```

```
FILE=/var/log.tar.gz
cd /
if [ -f $FILE ]; then
    rm $FILE
fi
tar czf $FILE /var/log
```

```
end script
```

And `logload` reads:

```
start on runlevel 2
```

```
script
```

```
FILE=/var/log.tar.gz
cd /
tar xzf $FILE
```

```
end script
```

Eventually add command `/sbin/initctl emit logsave` to `crontab` (see section 8.5 for information).

Reduce what is logged by editing `/etc/syslog.conf`.

Current version of Ubuntu (9.04) will give a warning on login if the disk is almost full. If you use another version or want to set a special level for when the `/var/log` memory is “full”, put these lines in `/etc/bash.bashrc` which is run every time you log in:

```
PERCENT=$(df -h | grep /var/log | awk '{print $5}' | cut -d "%" -f1 -
)
WARNLEVEL=80
if [ $PERCENT -gt $WARNLEVEL ]; then
echo "WARNING: /var/log is $PERCENT % full!!!"
fi
```



## 8.4 Convert Music

Following based on information from [http://gimpel.gi.funpic.de/wiki/index.php?title=Howto:convert\\_aac/mp4\\_to\\_wav/mp3/ogg\\_on\\_Linux](http://gimpel.gi.funpic.de/wiki/index.php?title=Howto:convert_aac/mp4_to_wav/mp3/ogg_on_Linux).

### 8.4.1 AAC/M4A to MP3

Install `faad` (read AAC-files) and `lame` (MP3-converter) with the following command:

```
$ sudo apt-get install faad lame
```

Now convert files with

```
$ faad --quiet -o - inputfile.m4a | lame --quiet - outputfile.mp3
```

To convert many files at once, use this script:

```
#!/bin/bash
# Script to convert AAC/M4A to MP3-files since my version of MOC
# cannot
# play those files. It uses 'faad' to read the music stream and 'lame'
# to convert to MP3.
#
# Created by Hans, 2009-08-11.

# Define temporary file:
INFOFILE=infofile.tmp
REMOVE=true      # If true, remove after conversion

# Function to extract music information from a file:
function extract()
{
    grep $1 $INFOFILE | cut --delimiter=: -f2 | sed -e 's/^ //'
}

if [ $# -eq 0 ]; then
    echo "Usage: $0 file(s)"
    echo " -- or with 'find' --"
    echo " find -H . -name '*.m4a' -print0 | xargs -0 $0"
    exit 1
fi

# Loop through all files.
for FILE; do
    # Remove the extension:
    DIR=$(dirname "$FILE")
    F=$(basename "$FILE" .m4a)
    F=$(basename "$F" .aac)
    F=$DIR/$F
```

```

# Write track information to the infofile (2> redirects both
  stdout and stderr to file):
faad -i "$FILE" 2> $INFOFILE
# Extract titles etc:
TITLE=$(extract title)
ARTIST=$(extract artist)
#WRITER=$(extract writer)
ALBUM=$(extract album)
DATE=$(extract date)
GENRE=$(extract genre)
TRACK=$(extract track)

echo "$FILE..."
# Convert the song using faad and lame:
faad --quiet -o - "$FILE" | lame --quiet -h --tt "$TITLE" --ta
  "$ARTIST" --tl "$ALBUM" --ty "$DATE" --tn "$TRACK" --tg "
  $GENRE" - "$F.mp3";

# Remove the AAC-file:
$REMOVE && rm "$FILE"

done

# Remove the temporary info file:
rm $INFOFILE

exit 0

```

## 8.5 Backup Script

The external disk is also used as backup of important files. Backups should according to Linux Standard Base (LSB) be stored at `/mnt/backup`. Therefore I made a soft link to the external disk. The following script is based on script found at <https://help.ubuntu.com/8.10/serverguide/C/backup-shellscripts.html> and is run once a day (in my case below at 23:45). Put

```

# m h dom mon dow    command
45 23 * * * /usr/local/bin/sysbackup.sh

```

in the crontab file (edit it by running `sudo crontab -e`). The script, `sysbackup.sh` follows below and the filename of the backup file will contain the name of the day, hence only the last 7 days will be stored.

```

#!/bin/sh
#####
#
# Backup to NFS mount script.
#
#####
# Script hämtat från https://help.ubuntu.com/8.10/serverguide/C/backup
  -shellscripts.html

```

```

# What to backup.
backup_files="/etc /boot /home /root /usr/local"

# Where to backup to.
dest="/mnt/backup"

# Create archive filename.
day=$(date +%A)
hostname=$(hostname -s)
archive_file="$hostname-$day.tgz"

# Print start status message.
echo "Backing up $backup_files to $dest/$archive_file"
date
echo

# Backup the files using tar.
tar czf $dest/$archive_file $backup_files

echo
echo "Done"
date

ls -lhL $dest

#####
## Testa arkiv med:
# tar -tzvf /mnt/backup/eee-chan-Monday.tgz
## Återställ filen /etc/hosts till /tmp:
# tar -xzvf /mnt/backup/eee-chan-Monday.tgz -C /tmp etc/hosts
## Återställ allt:
# cd /; sudo tar -xzvf /mnt/backup/eee-chan-Monday.tgz

```

## 8.6 Program List Backup

Make a list of all programs (incl. libraries) to a file on the external drive<sup>1</sup>. In case of a reinstallation, it is easy to reinstall all programs automatically. The following script creates filenames in same manner as in sec 8.5. Save this script somewhere, for example as `/usr/local/bin/listinstalled.sh`, and make it run every now and then using `crontab` (see above).

```

#!/bin/bash
FILE=/mnt/backup/installedsoftware-$(date +%A).txt
dpkg --get-selections > $FILE

```

---

<sup>1</sup>Based on <http://www.cyberciti.biz/tips/linux-get-list-installed-software-reinstallation-restore.html>

To recover, run this as administrator:

```
$ dpkg --set-selections < /mnt/backup/installedsoftware-Friday.txt  
$ dselect
```

press “i” to install.

## 8.7 Miscellaneous

`bwm-ng` Bandwidth monitor

`l1tag` Music Tag Editor for console.

`/etc/motd.tail` Welcome message to show when logging in from terminal.