Toshiba Chromebook 2

this is a very nice device for only $329. it has a full hd 1080p 13in IPS screen, is only about 1.3 or 1.5kg and it looks pretty good for the price. the keyboard is okay too except for the fact that there currently does not seem to be swiss german or even german version (one that contains all the keys in the “right” places..). what's also very good is the sound quality.

battery life is far better than anything i had owned so far .. i have to admint though that my other notebook is already about 3 years old.

one thing though makes the notebook almost unusable.. chromeos :) at least for me.. it heavily depends on being connected to the internet and this at least in my case is not always given.. i often use my notebook in a datacenter at a customers where i can't simply plug it into the network and get access.. or on an airplane etc. plus despite being a gmail user i don't consier myself a heavy cloud user.. but i am a fulltime linux user and that's where the chromebook gets interesting.. it's one of the only notebooks available in stores that run a pre installed linux kernel.. and thanks to crouton it's quite easy to get a fully capable linux environment running on it.. i had this notebook running with standby working and everything an far less time than many regular windows notebooks..

It turns out that the chrooted crouton wasn't satisfying after all as some stuff was not working and it was just too tedious to work with. so my chomebook sat in the cellar for a while. i recently researched again and found that the chromebook's firmware lock had been found and that it is now possible to disable it and flash a new bios which then enables you to install whatever you want on the little device.

The new solution

in short, this is what i have done:

- disable bios flash lock
- flash MrChromebox full blown bios image
- install ubuntu-gnome from a usb stick like on any other pc
- fix some issues

bios flash lock

in order to flash the bios you need to disable the hardware lock.. this is very simple once you know where to look :) .. simply remove the 8 screws on the underside of the chromebook, then remove the two rubber feet which are further away from the hinges and remove the two additional screws from there. open up the cover starting from the hinges and then remove the large silver shield that is held in place with 6 screws. underneath of screw nr. 5 you will see a silver washer-like sticker on the mainboard whichcovers two copper contacts. remove that sticker, put some electrical tape over those two half-circles on the board and reassemble everything. you can leave screw 5 away. now your bios is no longer write protected and you can go ahead and follow the instructions on mrchromebox.tech

fix some issues

i wanted to use ubuntu-gnome, so i simply installed it like i would on any other notebook. there is however Gallium OS which is based on Xubuntu and runs out of the box on the chromebook as it already includes fixes for all possible problems. so in order to avoid fixing anything you can simply use that. however, since i wanted gnome and not xfce and since the disk space is rather limited, i did not siply stick to gallium and then install the ubuntu-gnome-desktop package but tried to fix the issues i had with ubuntu-ghome.

Sound

the sound card in the toshiba chrombeook 2 with Bay Trail cpu is a max98090 chip which does not seem to be supported. there is a PPA repository by Alberto Aguirre that looks promising.

i have installed his repo and then added some apt preferences to keep the kernel at an older version in order to stay with the patched one from aguirre:

# cat /etc/apt/preferences.d/pinKernel Package: linux-generic linux-headers-generic linux-image-generic linux-restricted-modules-generic Pin: version 4.10.0-30 Pin-Priority: 1001

Keyboard remapping

the keyboard needs remapping in order to get the multimedia keys to work.

The old solution

here follows my original post from just after i bought the chromebook just for reference.. i wasn't happy in the end with this setup and never really used my chromebook..

there is a second possibility called chrubuntu which aims at installing a directly bootable ubuntu running a patched ubuntu kernel rather than the chromeos kernel.. however this solution did not work for the toshiba chromebook 2 at the moment of wrinting because right now this chromebook does not support legacy boot which wold be needed for that.. there is a possibility to boot into chrubuntu with a few commands in chromeos, however the author describing this solution did not get the sound card nor the standby functionality running so far and i don't see any advantage so far over runnign crouton which runs ubuntu in its own chroot environment with everyting working flawlessly..

here is how i did my setup:

enable Developer mode

- before you begin it might be a good idea to create a recovery disk for your chromebook. you can do this later on on a second pc but why not do it now on your chromebook where it just works :) .. simply install the “chromebook recovery utility” app from the store and run it. you need a memory stick or sd card with at least a size of 4GB which you can dedicate to being the recovery disk.

- now enable developer (important: this will erase all your local data and settings) mode in order to get root access to your chromeos: press and hold esc. and reload and then press the power button to reboot the chromebook into recovery mode. once you are in recovery mode, press ctrl+d to continue and then select to enable developer mode. this might take a moment. after that your notebook will reboot and you will see a warning that developer mode has been enabled. you can either wait 30seconds for it to boot or press ctrl+d once more to boot.
- once rebooted, go through the setup wizard until you have a working internet connection and log in

disable rootfs verification

in order to get this to work i had to switch to the developer firmware first bz running this command as superuser in a command prompt:

```
chromeos-firmwareupdate --mode=todev
```

then reboot and disable rootfs verification (caution, whenever i ran this command a second time it somehow destroyed my crhome os and i had to recover from the usb stick! all settings where lost)

```
sudo /usr/share/vboot/bin/make_dev_ssd.sh --force --
remove_rootfs_verification
```

reboot again now whenever you want to change anyting on your root partition you can run

```
mount -o remount,rw /
```

to be able to writ to your root partition. this will be needed later on to autostart ubuntu and other stuff.

install crouton

- download crouton
- open a terminal with ctrl + alt + T. start a shell by typing “shell”
- run crouton. at best you first look at the available command line options by running

  ```
  sudo sh ~/Downloads/crouton -h
  ```

  so you can find what options suti you best. in my case at the time of writing this was

  ```
  sudo sh ~/Downloads/crouton -r trusty -t gnome,keyboard
  ```

  note the keyboard target. that's needed to get the function keys running.

- this will install a ubuntu trusty 14.04 lts version with a basic gnome environment. as the script tells you you can now start your ubuntu by typing

```
sudo startgnome
```

customize software

- crouton installs a very minimalistic gnome desktop and not much else. in order for it to look pretty and to have all the gnome tools you might have gotten used to you need to install additional packages. the easyest way to just get everyting that belongs to it you siply

  ```
  apt-get install gnome
  ```

  but that takes over 3GB of disk space and includes many games and stuff you probably don't ever need. \\instead you can run

  ```
  apt-cache depends gnome
  ```

  to see what packages get installed and then work your way from there to compile a list of what you really want installed on your system and what not. here's the list i've compiled:

  ```
  sudo apt-get install nano gnome-core desktop-base network-manager-gnome cheese file-roller gedit gnome-color-manager gnome-documents gnome-nettool nautilus-sendto seahorse totem vinagre alacarte avahi-daemon gimp gnome-media gnome-tweak-tool inkscape libreoffice-gnome libreoffice-writer libreoffice-calc libreoffice-impress sound-juicer tomboy shotwell tracker-gui xdg-user-dirs-gtk cups-pk-helper gedit-plugins gnome-applets gnome-shell-extensions gstreamer1.0-libav gstreamer1.0-plugins-ugly totem-plugins libgtk2-perl dia-gnome thunderbird synaptic update-manager-core thunderbird-gnome-support bash-completion vlc ubuntu-restricted-extras ttf-ubuntu-font-family software-center remmina
  ```

  this will use about 1.6GB of your precious disk space :) but it includes some more productivity stuff and VLC.

autostart crouton

source

- make sure your root filesystem is re-writeable.. see beginig of this page.
- now download crouton.conf and put it in the /etc/init/ folder
- make sure the parameters are matching your needs and installation
- touch a file called crouton.init in your downloads directory.

simulate middle mouse button

on a normal trackpad you would simply click the left and right mouse key at the same time to simulate a center key klick, but the chromebook has only one button. so this leaves us with the mac solution: tap the pad with one finger for a left click, two fingers for a right click and.. you might guess it.. three fingers for the center click… you can enable this by opening a command prompt and running this command

```
synclient TapButton3=2
```

however, this command needs to be executed each time you start gnome.. so here's how to do that automatically:

```
echo "synclient TapButton3=2" > ~/touchpad_settings.sh
chmod +x ~/touchpad_settings.sh
```

use of course your own user name instead of psuter. now add touchpad_settings.sh to your startup appilcations through te settings gui.

sidenote: updating ubuntu

if you need to update to a new releae simply use do-release-upgrade as youwould in any ubuntu installation.then log off and in chrome os run the crouton script to update its side of the installation:

```
sudo sh -e ~/Downloads/crouton -n <chroot_name> -u
```

where the chroot_name is uusally equal to the currently installed release name. in oder to keep it that way even after the upgrad we need to rename the chroot name by running

```
sudo edit-chroot -m <newrelease> <oldrelease>
```

OpenVPN with config file

chromeos comes with openvpn built in but it has a very limited gui which does not allow to use a config file and it does further not allow to use tls auth. but luckily it uses a standard openvpn binary behind the scenes, so that the binary can be called manually inorder to initiate the connection with a config file.

i have created a startup script that runs openvpn after iptables (and therefore the network) has been initiated. copy all certs and the config file to /home/vpn and name the configfile ovpn.conf

now copy paste this upstart script to /etc/init/openvpn.conf

```
description     "Start openvpn service with a config file from
/home/vpn/ovpn.conf"
```

```
author          "d-chromeos@psuter.ch"

start on started iptables
kill timeout 10

task

script
  openvpn --mktun --dev tun0
  openvpn --config /home/vpn/ovpn.conf --daemon --dev tun0 --cd /home/vpn
end script
```

From:
http://wiki.psuter.ch/ - **pswiki**

Permanent link:
http://wiki.psuter.ch/doku.php?id=toshiba_chromebook_2

Last update: **14.08.2017 23:50**