

The old and cruffy interface set

Living Room Lights

23:14 2C

North Tower:OFF

Since: 20221209 at 11:30

Fireplace

South Tower:OFF

Since: 20221209 at 08:04

North Table:OFF

Since: 20221209 at 08:04

South Table:OFF

Since: 20221209 at 08:04

Couch

All lights off

Sound

Sound

Current volume is 30

Return

Volume

1

2

3

4

5

6

7

8

9

Mute

Stop

Select Speakers

Select CBC Station

Select SiriusXM Station

Speakers

Return

Connect ALL Speakers

Connect HomePods

Connect Sonos Livingroom

Link Volume to System

Disconnect All Speakers

CBC Stations

Radio is off

Return

Victoria (PT)

Edmonton (MT)

Winnipeg (CT)

Toronto (ET)

Halifax (AT)

St. John's (NT)

SiriusXM Stations

Current station is SiriusXM Acoustic

Return

Specify Channel:

Play/Pause

Acoustic Guitar

BBC World Service

Living Room Blinds

Return

Blinds are CLOSED

Since: 20230324 at 19:15

3D Printer Heater Timer

Current Time: 23:21 | 3D Printer Temp: 24C

3D Printer Heating is ON

On for 4 hours and 0 minutes with Auto-Off at: 0005 on Saturday

Specify Duration (H:MM) or OFF:

Printer room temperature will be maintained between 23C and 23C while heating is active

Heater is OFF right now

Home Assistant

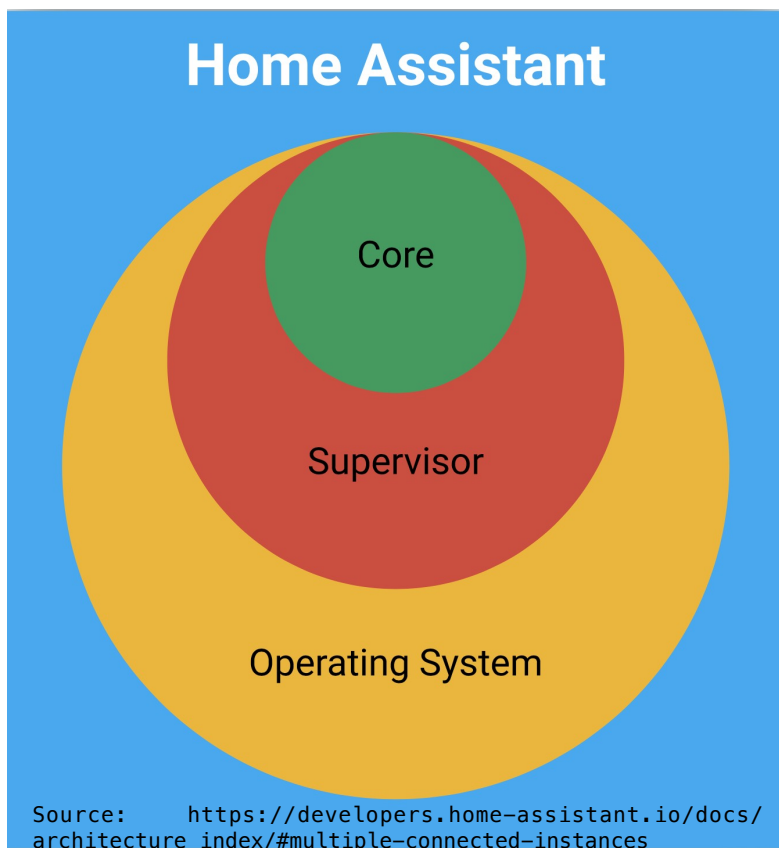
<https://www.home-assistant.io/getting-started/>

There are 4 basic "flavours" of Home Assistant (HA) installation:

- Home Assistant Operating System (HAOS [see * below])
- Home Assistant Container (Docker-centric)
- Home Assistant Supervised (Manual install)
- Home Assistant Core (Python Virtual Environment)

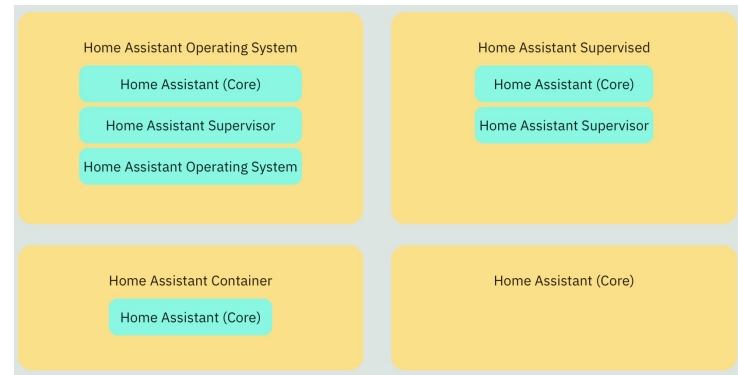
I chose the HAOS option on a Pi4, because everything that I read about installing HA in a Linux LXD/LXC Container suggested that it required significantly more experience than I currently have (especially given the time that I had available!)

My mid-term goal is to find a way to install HA safely and efficiently in an LXD/LXC Container so that I can propagate these to my family and friends and still be able to easily maintain them.



[*] Just to add some confusion to the mix, HAOS is also referred to by its former names, HASSOS and HASS.io, depending on the age of the documentation to which you are referring.

Source: <https://home-assistant-guide.com/guide/the-home-assistant-beginners-guide-part-1-setting-up-hass-io/>

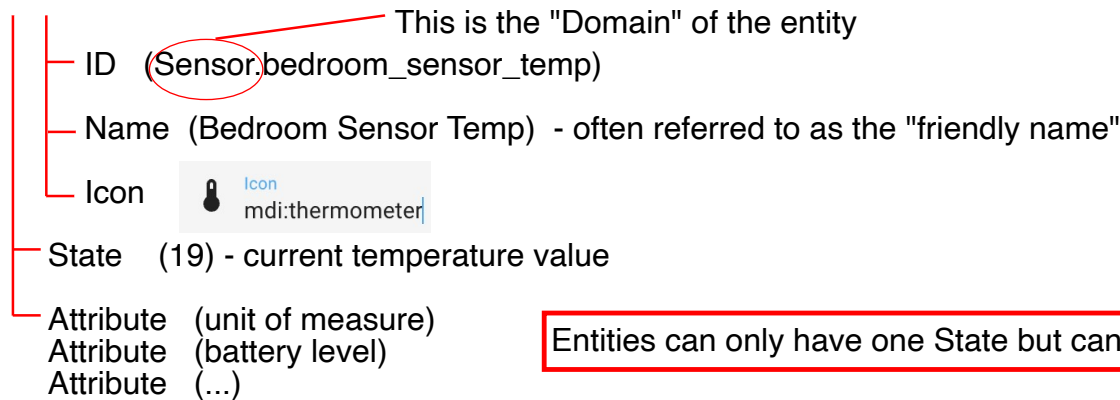


Basic Home Assistant (HA) Terminology

Each installation of Home Assistant (HA) is referred to as an "instance".

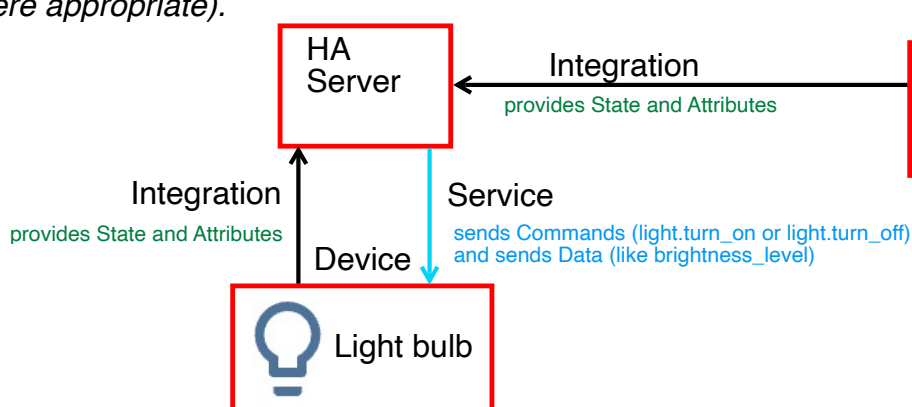
Device (a "thing" like a Motion Sensor - which can measure motion, temperature and light level)

Entity (an aspect of the Device like Temperature)



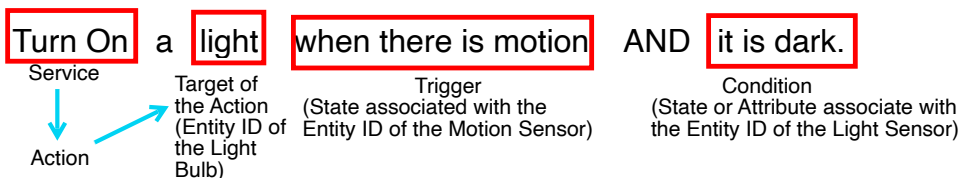
Integrations - A simple example

A Device is "Integrated" into HA and then through the Integration the Device is able to provide State and Attribute information to HA and through Services HA provides Commands and Data to control the Device (where appropriate).



Automations - A simple example

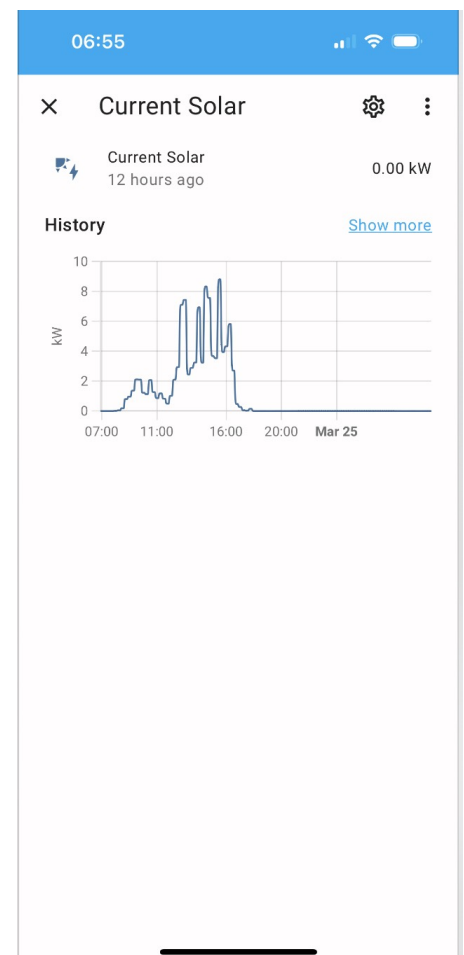
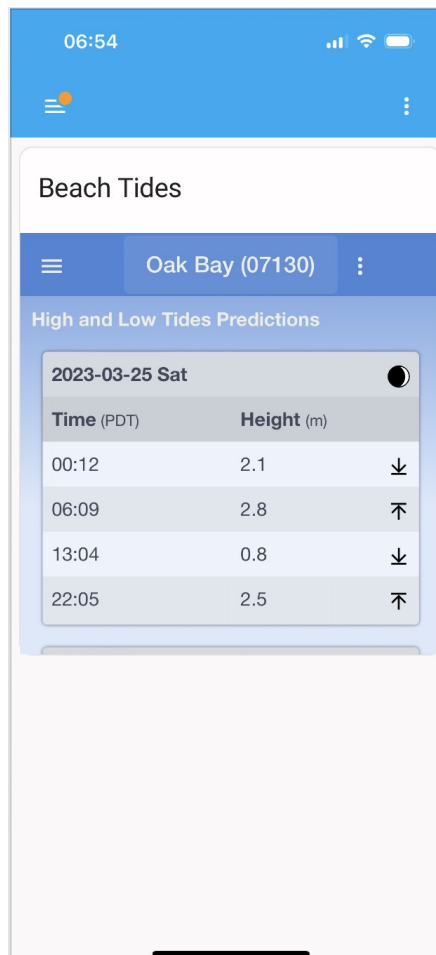
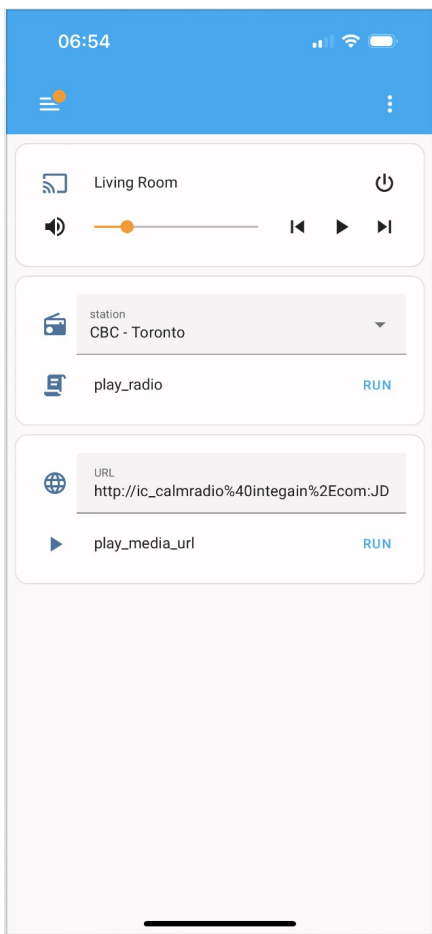
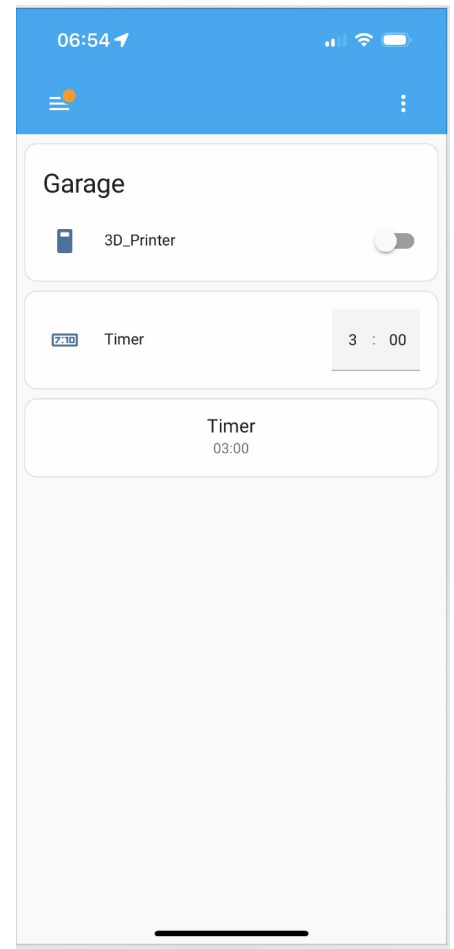
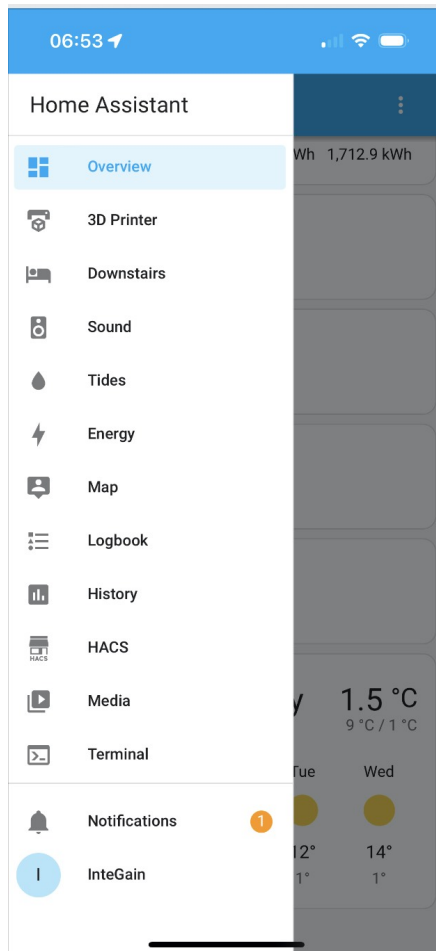
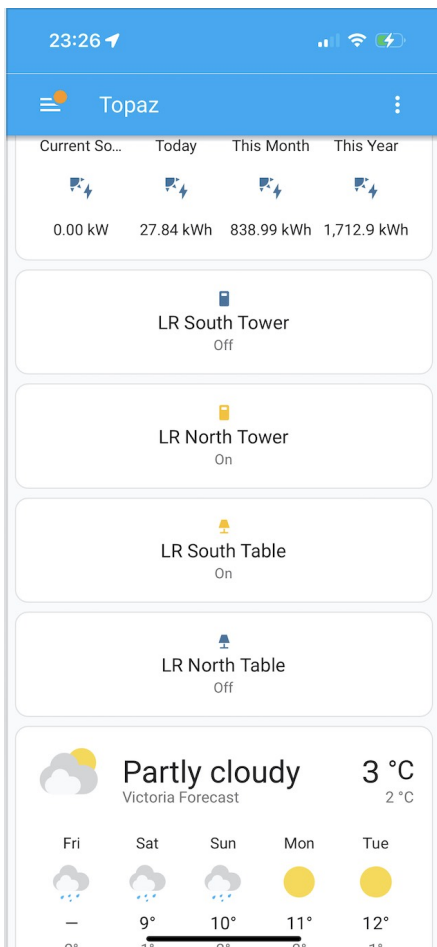
Assume that we want to turn on a light when there is motion detected AND it is dark. To do this, we will use an HA Automation.



The "Trigger" is the primary condition upon which the Action is dependent. So, in this instance, you might want to turn on the light when there is motion but it is not dark, but it is not likely that you would want to turn on the light when it is dark but there is no motion. The "Trigger" is a change in the "Platform State" (in this instance from off [no motion detected] to on [motion detected]). The "Condition" filters the Trigger (if desired) to a smaller set of circumstances.

In an Automation, an ACTION is taken on a TARGET based on a TRIGGER and (optional) associated CONDITIONS.

The current interface set



Installation

Installation is quick and easy using the instructions provided at <https://www.home-assistant.io/installation/raspberrypi> with one exception: I needed to download the file rather than using the URL pull because I received an error when trying to use the URL directly in BalenaEtcher. Since I am using a Mac, your mileage may vary.

Attention

Something went wrong. If it is a compressed image, please check that the archive is not corrupted.

```
/Applications/#IC
ADDED#/balenaEtcher.app/Contents/Resources/app,
writer.js:1
```

```
((()=>{var __webpack_modules__=[],
(__unused_webpack_module,exports,__webpack_re
{"use
strict":Object.defineProperty(exports,"__esModule",
{value:true});const
tslib_1=__webpack_require__(2);tslib_1.__exportSta
(__unused_webpack_module,__webpack_exports__
```

My BalenaEtcher error message

With the file downloaded, I simply pointed BalenaEtcher at it and within a few minutes I had a transferred image file.

The Home Assistant App is available on the Apple App Store and on Google Play. A Home Assistant instance is accessible from any web browser, too.

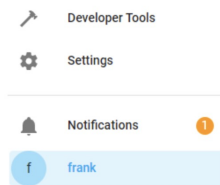
Individual users can be created, and screens (called "dashboards") can be created which are visible to everyone or only to administrators. You can also (apparently) create "views" which further restrict who can see what. I've not yet done this.

What I HAVE done:

- created Dashboards for my primary functions
- created working configurations for my lights
- integrated Flic buttons using Webhooks
- created a kiosk-style configuration (locked to the HA screens) on an Amazon Fire Tablet
- installed a couple of useful add-ons
 - HACS (Home Assistant Community Store)
 - Terminal and SSH
- pounded my head against a wall trying to get other things working...

In the future I want to...

- Integrate my EcoWitt weather station and temperature/humidity probes
- re-create the 3D Printer Heater control
- re-create the Hummingbird heater control
- re-create the living room blinds controller
- re-create the pond controller
- create a Bathroom humidity monitor to automatically control the exhaust fan
- integrate IotaWatt energy monitoring
- re-create the Sound system manager
- convert and integrate additional light switches around the house



YAML Lint tool
<https://www.yamllint.com>

Added SSH and Terminal

1. Select your **User Account** in the bottom left.
2. Enable **Advanced Mode** in the account options.
3. Now that **Advanced Mode** is enabled, we can install the **Terminal & SSH** application. Select **Settings** in the bottom left.
4. Select **Add-ons**.

In the bottom right, select **Add-On Store**.

Search for **Terminal & SSH**, then select the **Official** add-on.

Select the **Install** option.

When it's done installing, select **Start** in the bottom left. You can also enable any of the other options

After it's started, select **Open Web UI** in the bottom right.



Start on boot
Make the add-on start during a system boot ☒

Watchdog
This will start the add-on if it crashes ☐

Auto update
Auto update the add-on when there is a new version available ☐

Show in sidebar
Add this add-on to your sidebar ☐

[START](#)

Added HACS

Add SSH and Terminal (described above) then start it and Open Web UI (see above)
Paste the command below in the terminal window by pressing CTRL + SHIFT + V. If you have trouble pasting it in, type the command into the terminal window and press enter.

```
wget -O - https://get.hacs.xyz | bash -
```

HACS will be downloaded. When it's done downloading, run the command below to restart Home Assistant.

```
ha ha restart
```

After Home Assistant connects, select Settings, then Devices & Services.

In the bottom right, select Add Integration.

Search for HACS and select it.

If you agree with everything, select all options and then Submit.

Copy the code that Home Assistant provides and then select the link to sign into GitHub.

Sign in to GitHub, then paste in the code from the previous step.

If you'd like to proceed, select Authorize HACS.

HACS is now installed! It's best to reboot now by selecting Settings > System > Hardware > Reboot Host (top right corner).

After Home Assistant loads back up, HACS will be fully installed and ready to use!

Set up a new integration

Search integrations
hacs

HACS



Device Activation

Enter the code displayed on your device

□ □ □ □ - □ □ □ □

Continue

GitHub staff will never ask you to enter your code on this page.

Disable lock screen on Fire Tablet

<https://www.computerhilfen.de/english/fire-tablet-disable-the-lock-screen-no-root-required.html>
(full instructions are presented later in this document)

Fully-Kiosk App for FireOS

Source: <https://www.fully-kiosk.com> Info: <https://www.fully-kiosk.com/en/#faq-fire>
Download: <https://www.fully-kiosk.com/en/#download-box>
Need to manually download APK for FireOS

Force Fire Tablet to NOT update OS (which could break Fully-Kiosk)

Download the app: <https://github.com/mrhaydendp/Fire-Tools>
Mac Instructions: <https://github.com/mrhaydendp/Fire-Tools/blob/main/Setup-Instructions.md#macos>
Run the app: `./ui.sh Update`

Instructions for disabling lock screen on Fire Tablet

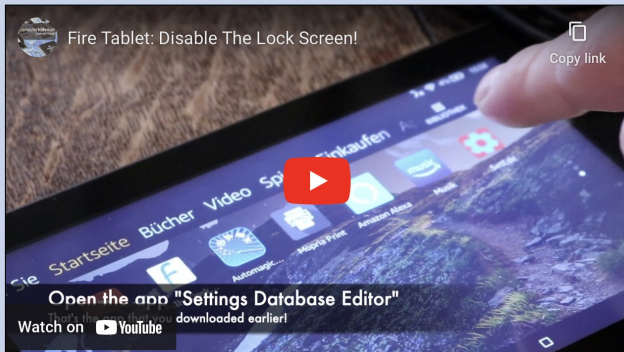
<https://www.computerhilfen.de/english/fire-tablet-disable-the-lock-screen-no-root-required.html>

Fire Tablet: Disable the Lock Screen (no root required!)

Computerhilfen > English tips
02/12/2019 (33552x read)

★★★★★ Vote! → ★★★★★ (1 votes, average: 4,00 out of 5)

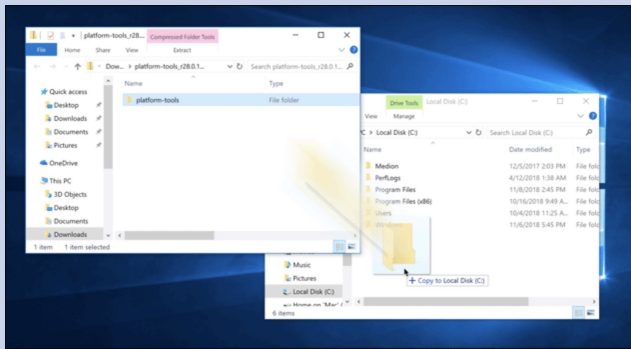
Amazon sells their Fire Tablets as cheap and nice tablets that could be perfect for a smart home monitor. Unfortunately they come with a lock screen that cannot easily be turned off in the settings. But don't worry, with the Google Android Developer Bridge you can change two important settings to keep the lock screen away!



This method also has some disadvantages: You cannot access the settings from the swipe-down menu anymore, the developer options are gone and there might be problems with updating apps from the Google Play Store. Luckily you can change the settings back – but then the lock screen will appear again.

Set up your ADB connection to the tablet

You need an ADB connection to your tablet: Google's „Android Debug Bridge“ is meant for developers. Here we use it to allow an app to change the tablet's settings. You can download the Google ADB tools on the Android Developer website; just make sure to take the „platform-tools“ directory and copy it directly to your Windows drive („C:\“):



- » ADB Download Windows
<https://dl.google.com/android/repository/platform-tools-latest-windows.zip>
- » ADB Download Mac OS
<https://dl.google.com/android/repository/platform-tools-latest-darwin.zip>
- » ADB Download Linux
<https://dl.google.com/android/repository/platform-tools-latest-linux.zip>

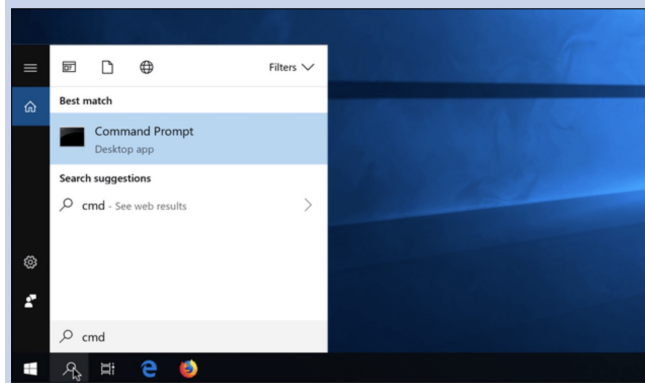
Now you have to enable the ADB connection on the tablet in the settings: Tap on settings and „Device Options“. To enable the „Developer Options“ you have to tap seven times onto the tablet's serial number.

After that, tap on „Developer Options“ and activate the ADB connection. Now you can connect the Tablet to your PC and open the command prompt. Use the command „cd“ to change to your platform-tools directory. Here you can run the ADB apps that will change settings on your tablet and help you to get rid of the Fire tablet's lock screen!

Disable the Fire Tablet's Lock-Screen!

To access the important setting on the Fire tablet, we are using the app „Settings Database Editor“. You can [download the app from the Google Play Store](#) (if you have it already installed on your Fire tablet) or from other websites like [APK Mirror](#). If you download the app from a website, not the Google Play Store, copy it onto a micro SD card to get it to the tablet and use an app like the „ES file explorer“ to open it. Make sure to allow „Apps from Unknown Sources“ in the tablets security settings!

Now connect your tablet to the PC and start the ADB bridge like we show in the video. Open the command prompt by typing „cmd“ in the Windows search bar and click on „Command prompt“:



To change to the „platform-tools“ directory (if you have copied it to C:\), enter this command:

```
cd c:\platform-tools
```

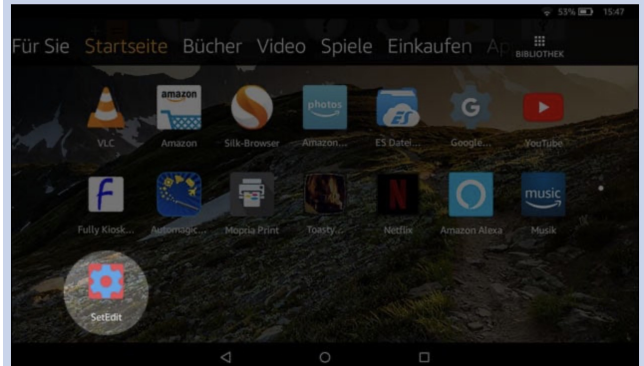
Now you can start the adb app and set up a connection to the tablet by entering these commands:

```
adb devices
```

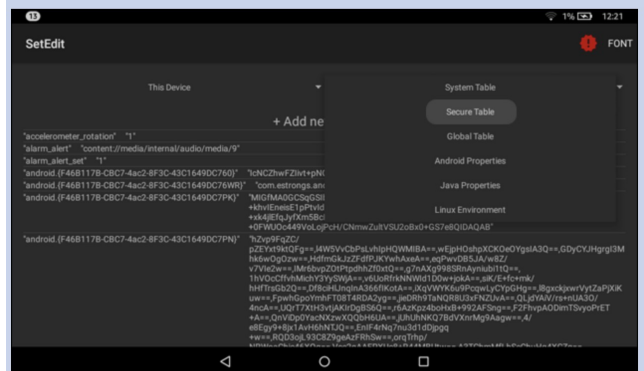
This will open the connection. You might have to grant access on the Fire device.

```
adb shell pm grant by4a.setedit22 android.permission.WRITE_SECURE_SETTINGS
```

This will allow the new app to change the Android-Settings that control the lock screen. You can now disconnect the Fire Tablet and start the app!



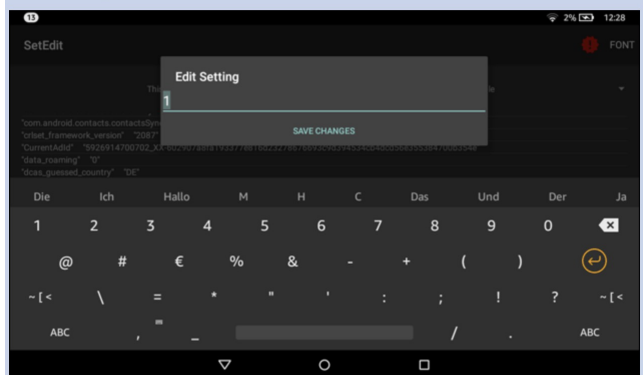
Now tap on „System Table“ in the upper right corner and select „Secure Table“ instead:



This will list all settings that can be changed here. Be very careful to not change anything else because it could damage your device!

Look for the line: „lockscreen_disabled“, „0“

Tap on that line and click on „Edit Value“: You can now change that settings value from „0“ to „1“ and save the setting.



Now select „Global Table“ instead of „Secure Table“ in the upper right corner and look for this setting:

„device_provisioned“, „1“

Edit this value from „1“ to „0“ and restart your tablet: Press the on/off button for several seconds until the Fire Tab asks you to shut down. When the tablet is powered off you can restart it by pressing the on/off button again for a few seconds.

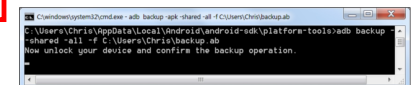
The Fire Tablet should start right away without the lock screen and show your installed apps: You are done! There might be a chance that Amazon is disabling your setting with an update: If you do not need an internet connection, you could forbid the tablet's internet access in your router's settings (if possible).

Backing Up Your Device

You can now use the adb backup command to back up your device. To back up everything, run the following command:

```
adb backup -apk -shared -all -f C:\Users\NAME\backup.ab
```

This backs up all your installed apps (APK files) and shared storage (SD card) data to the C:\Users\NAME\backup.ab file on your computer. Replace NAME in the command with your Windows username or provide another location for the backup file.



After running this command, you'll have to agree to the backup on your device. You can also encrypt the backup with a password here, if you like.

Instructions for putting an iOS device in Kiosk Mode:

<https://www.howtogeek.com/252670/how-to-put-an-ipad-into-kiosk-mode-restricting-it-to-a-single-app/>