

**Goal:** To generate a TV guide (EPG via Webgrab++) on a Raspberry Pi, every day, and then automatically transfer this guide, every day, from the Raspberry Pi to an iMac containing the TV application (EyeTV) in which the guide will then automatically load.

## 1. Prepare Raspberry Pi

---

### 1. Install OS on Pi (Debian version 11 bullseye)

<https://www.raspberrypi.com/news/raspberry-pi-os-debian-bullseye/>

Burn Image 2022-09-22-raspbian-bullseye-armhf-full.img on SD card with for example "balenaEtcher".

### 2. Install on-off knop scripts for RemotePi Board

This allows you to safely turn the Pi on and off without the risk of damaging the sd card.

<https://www.msldigital.com/pages/support-for-remotepi-board-plus-2015/>

### 3. Install Passwordless SSH Access

See <https://www.raspberrypi.com/documentation/computers/remote-access.html>

Setup SSH access in van Pi naar iMac (from IP<Pi> to IP<iMac>)

Test on Pi terminal with command =>

ssh [<usernamepi>@<ip-adress pi>](#)

### 4. Send Test file to iMac with scp command

Make testfile test.txt and put this into the Downloads folder of the Pi.

On Pi terminal type command =>

scp/home/pi/Downloads/test.txt <username iMac>@<IP-adress iMac>:/Users/<username iMac>/wg++

The file is now available on the iMac in directory ./wg++

Make a backup of the SD kaart.

The Raspberry Pi:



## 2. Install dotnet 6.0

---

We have the Pi 3B+ with Debian11 (Bullseye). That's ARM32 (32 bits OS).  
So we need the ARM32 software version, not ARM64.

Check with:

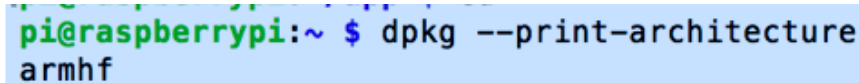
```
pi@raspberrypi:~$ uname -a
Linux raspberrypi 5.15.84-v7+ #1613 SMP Thu Jan 5 11:59:48 GMT 2023 armv7l GNU/Linux
```

"armv7l" is the indicator voor ARM32.  
"aarch64" is ARM64

OR check with:

```
pi@raspberrypi:~$ dpkg --print-architecture
armhf
```

armhf = ARM32  
aarch64 = ARM64



```
pi@raspberrypi:~$ dpkg --print-architecture
armhf
```

**Run dotnet-install.sh =>**

First read this to check dependencies.

<https://github.com/dotnet/core/blob/main/release-notes/7.0/linux-packages.md>

and read for install script..

<https://learn.microsoft.com/en-us/dotnet/core/install/linux-scripted-manual#scripted-install>

You can download the .NET script from:

<https://dot.net/v1/dotnet-install.sh>

Visit <https://dotnet.microsoft.com/download> to get **more info** about the installer.

Make it executable and Run this for dotnet 6.0:

```
sudo chmod +x ./dotnet-install.sh
```

And then install with:

```
./dotnet-install.sh --channel 6.0 --runtime dotnet
```

```
pi@raspberrypi:~/apps$ sudo chmod +x ./dotnet-install.sh
```

```
pi@raspberrypi:~/apps$ ./dotnet-install.sh --channel 6.0 --runtime dotnet
```

Add the .dotnet directory to \$PATH:

```
echo 'export DOTNET_ROOT=$HOME/.dotnet' >> ~/.bashrc
echo 'export PATH=$PATH:$HOME/.dotnet' >> ~/.bashrc
source ~/.bashrc
```

Add in .bashrc:

```
export PATH=$PATH:/home/pi/.dotnet
export DOTNET_ROOT=/home/pi/.dotnet
```

Then type:

```
pi@raspberrypi:~$ source ~/.bashrc
```

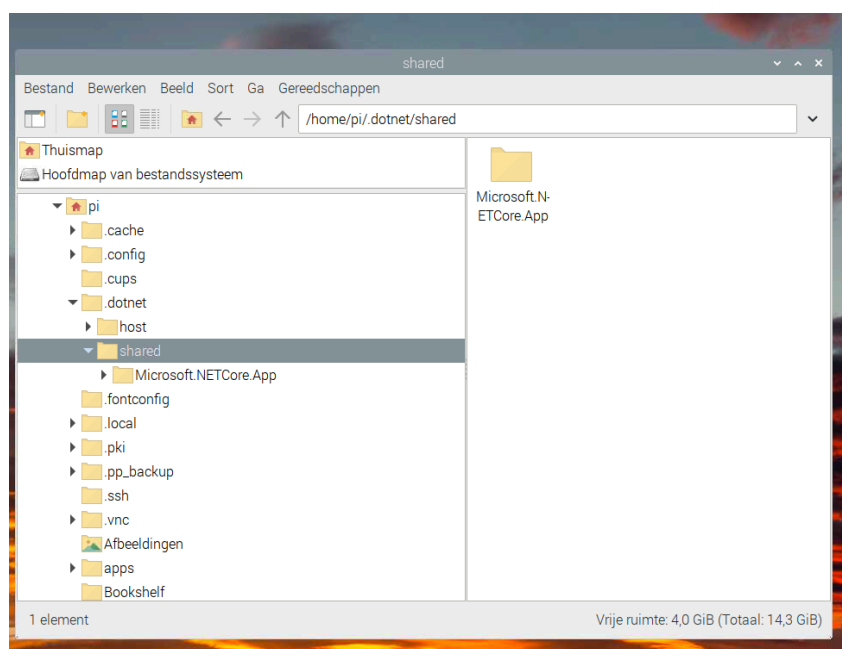
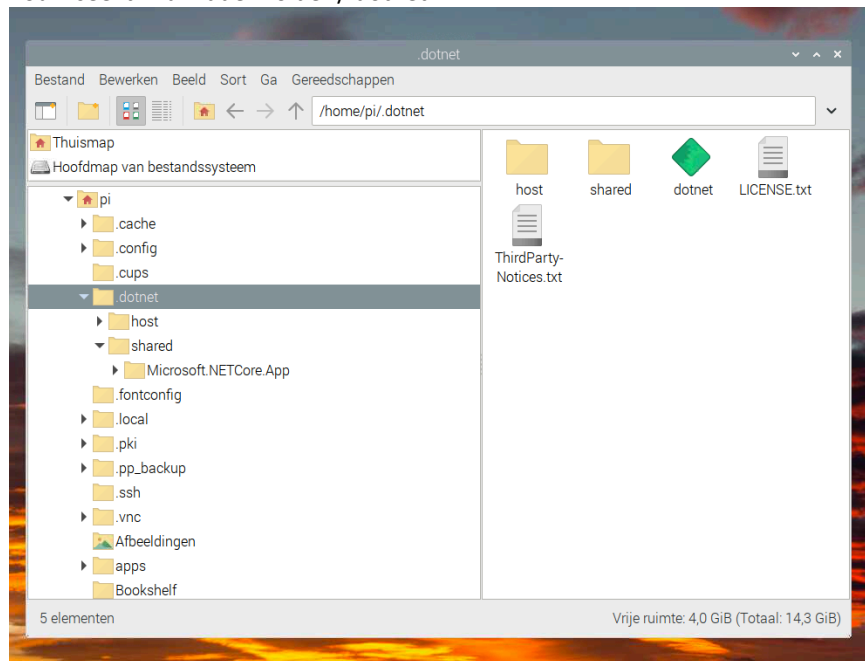
Test if .net is installed correct:  
dotnet --list-runtimes  
dotnet --info

Remark: “dotnet — version” does not work... (= error in Debian documentation) =>  
The documentation must be outdated.

```
pi@raspberrypi:~ $ dotnet --list-runtimes
Microsoft.AspNetCore.App 6.0.14 [/home/pi/.dotnet/shared/Microsoft.AspNetCore.App]
Microsoft.NETCore.App 6.0.14 [/home/pi/.dotnet/shared/Microsoft.NETCore.App]
```

OK!

You'll see it in a hidden folder /.dotnet



### 3. Install WebGrab (WG++)

---

Download WebGrab files =>

Full download link V4.2.2:

[http://webgrabplus.com/sites/default/files/download/SW/V4.2.2/WebGrabPlus\\_V4.2\\_install.tar.gz](http://webgrabplus.com/sites/default/files/download/SW/V4.2.2/WebGrabPlus_V4.2_install.tar.gz)

And download V5.0.1:

[http://webgrabplus.com/sites/default/files/download/SW/V5.0.1/WebGrabPlus\\_V5.0.1\\_beta\\_install.tar.gz](http://webgrabplus.com/sites/default/files/download/SW/V5.0.1/WebGrabPlus_V5.0.1_beta_install.tar.gz)

Put the install files van WG++ (V4.2 en V5.01) on de Pi in directory /home/pi/apps:

You can do this with an usb stick from your desktop PC or iMac or with the scp command.

We have:

WebGrabPlus\_V4.2\_install.tar.gz

WebGrabPlus\_V5.0.1\_beta\_install.tar.gz

Example of Copy from iMac to Pi with scp command on terminal of your PC/iMac:

```
iMac:<user>$ scp WebGrabPlus_V4.2_install.tar pi@<IP-adress Pi>:/home/pi/apps  
WebGrabPlus_V4.2_install.tar.gz
```

and

```
iMac:<user>$ scp WebGrabPlus_V5.0.1_beta_install.tar.gz  
pi@<IP-adress Pi>:/home/pi/apps WebGrabPlus_V5.0.1_beta_install.tar.gz
```

Install WG++ V 4.2.2 and also install dotnet 6.0 =>

See download links above.

Install 4.2.2 as it is a full install which has the bin.net.sh,ect files needed to run WebGrab.

You have to install V4.2.2 as it has all the files.

To install V4.2.2 extract the files and run the install.sh als volgt:

```
pi@raspberrypi:~/apps $ tar -zxvf WebGrabPlus_V4.2_install.tar.gz
```

```
pi@raspberrypi:~$ cd /home/pi/apps/.wg++
```

```
pi@raspberrypi:~/apps/.wg++ $ ./install.sh
```

```
pi@raspberrypi:~$ cd /home/pi/apps  
pi@raspberrypi:~/apps $ ls  
dotnet6 dotnet-install.sh dotnet-runtime-6.0.14-linux-arm.tar WebGrabPlus_V4.2_install.tar.gz WebGrabPlus_V5.0.1_beta_install.tar.gz  
pi@raspberrypi:~/apps $ tar -zxvf WebGrabPlus_V4.2_install.tar.gz  
.wg++/  
.wg++/bin.net/  
.wg++/bin.net/changelog.txt  
.wg++/bin.net/Microsoft.WindowsAzure.Storage.dll  
.wg++/bin.net/Newtonsoft.Json.dll  
.wg++/WebGrab++.config.example.xml  
pi@raspberrypi:~/apps $ cd /home/pi/apps/.wg++  
pi@raspberrypi:~/apps/.wg++ $ ls  
bin.net doc install.sh mdb rex run.net.sh site  
pi@raspberrypi:~/apps/.wg++ $ ./install.sh  
==> installing siteini.pack  
==> installing WebGrab++.config.xml  
==> installing mdb/mdb.config.xml  
==> installing rex/rex.config.xml  
==> DONE  
pi@raspberrypi:~/apps/.wg++ $
```

WebGrab will be installed in your /home/[username]/apps/.wg++ directory.

After this you can update to other beta versions.

Beta versions don't have a installer, just files.

You just replace the files in the [bin.net](#) directory. (=> /home/[username]/.wg++/bin.net folder).

After that you have to install the downloaded V5.0.1:

[http://webgrabplus.com/sites/default/files/download/SW/V5.0.1/WebGrabPlus\\_V5.0.1\\_beta\\_install.tar.gz](http://webgrabplus.com/sites/default/files/download/SW/V5.0.1/WebGrabPlus_V5.0.1_beta_install.tar.gz)

**Extract** the files and replace the ones in /home/[username]/.wg++/bin.net folder.

**V5.0.1 uses dotnet 6.0**

```
pi@raspberrypi:~/apps/.wg++ $ cd
```

```
pi@raspberrypi:~ $ cd /home/pi/apps
```

```
pi@raspberrypi:~/apps $ tar -zxvf WebGrabPlus_V5.0.1_beta_install.tar.gz
```

```
pi@raspberrypi:~/apps/.wg++ $ cd
pi@raspberrypi:~ $ cd /home/pi/apps
pi@raspberrypi:~/apps $ ls
dotnet6 dotnet-install.sh dotnet-runtime-6.0.14-linux-arm.tar WebGrabPlus_V4.2_install.tar.gz WebGrabPlus_V5.0.1_beta_install.tar.gz
pi@raspberrypi:~/apps $ tar -zxvf WebGrabPlus_V5.0.1_beta_install.tar.gz
.wg++/
.wg++/bin.net/
.wg++/bin.net/Azure.Core.dll
.wg++/bin.net/Azure.Storage.Common.dll
.wg++/bin.net/Azure.Storage.Files.Shares.dll
.wg++/bin.net/changelog.txt
```

**Finally:**

Configure WG++ to your own needs (see [the configuration howto](#)).

In WebGrab++.config.xml on your licence line, use h

After you run WebGrab once it will update your hardware id, you can remove the h option.

```
<license wg-username="username" registered-email="email"
password="password">h</license>
```

And Run the program. Execute next in a command line window.

```
pi@raspberrypi:~ $ cd /home/pi/apps/.wg++
```

```
pi@raspberrypi:~/apps/.wg++ $ ./run.net.sh
```

The EPG data is now loaded from the internet and the EPG guide is compiled.

There is now an EPG guide under ~/.wg++/

Copy this guide to the wg++ directory on your iMac for your TV application (EyeTV) and you're done!

**Test:** `pi@raspberrypi:~/apps/.wg++ $ ./run.net.sh`

[illegible]

The screenshot shows the EyeTV software interface on a Mac. At the top, the title bar reads 'EyeTV-programma's'. Below it, a navigation bar shows the date 'ma, 27. feb.' and the time '20:00'. The main window is divided into a sidebar on the left and a large grid on the right.

**Sidebar (Left):**

- BIBLIOTHEEK** (Library icon)
- Opnamen** (Recordings icon)
- Programmeringen** (Schedule icon)
- Zenders** (Channels icon)
- Programmagids** (Program Guide icon)
- EyeTV Plus** (EyeTV Plus icon)
- GEDEELD** (Shared icon)
- SLIMME GIDSEN** (Smart Guides icon)
- Bert** (Bert icon)
- AFSPEELIJSTEN** (Playlists icon)
- ZENDERS** (Channels icon)
  - Bert Favorieten
  - Bert Top 10

**Main Grid (Right):**

The grid displays a schedule of TV programs. The top row shows the date 'ma, 27. feb.' and the time '20:00'. The programs are listed in a grid format, with columns representing different time slots. The programs include:

- 100 MTV**: 2000 Catfish (UK) - Chantelle & Lewis
- 101 Net 5**: 1958 Lingo
- 102 RTL 4**: 2000 Goede Tijden, Slechte Tijden - Afllevering 6720
- 103 RTL 5**: 1931 Deurwaarders UK - Afllevering 23
- 104 SBS 6**: 1930 HLF8
- 105 FOX**: 1903 F The Uni - Pilot
- 106 NPO 1**: 2000 NOS Journaal
- 107 NPO 2**: 1957 Met het mes op Tafel
- 108 NPO 3**: 1925 First Dates
- 109 RTL 7**: 1959 Pawn Stars - Afllevering 36
- 110 RTL 8**: 1931 ER - No good deed goes unpunished
- 111 RTL 2**: 1944 Bar Rescue - Afllevering 27
- 112 TV Noord**: 2000 Noord Vandaag - Afllevering 484
- 113 Veronica / Dis...**: 1955 The Big Bang Theory - The Gyroscopic Collapse
- 114 Ziggo Sport Sel...**: 1930 Ziggo Sport Goals
- 2030 NCIS**: Higher Education
- 2030 Help, Mijn Man Is Klusser!**: Afllevering 3
- 2030 Big Brother**: Afllevering 43
- 2030 Chateau Meiland**
- 2102 The Equalizer**: Glory
- 2033 Van onschatbare waarde**
- 2033 Radar**
- 2025 Hunted VIPs**
- 2027 Sleepless**
- 2027 Chalet Girl**
- 2030 The Machines That Built The World**: Afllevering 3
- 2030 Noord Scoort**: Afllevering 8
- 2030 Veronica Offside**
- 2030 Rondo**
- 2100 Deliciousness**: Be-Tray-Al
- 2132 Ridiculousness**: Chanel and Sterling Dvill
- 2124 NCIS: Los Angeles**: Blood Bank
- 2130 112 Vandaag**: Afllevering 41
- 2135 Vandaag Inside**
- 2126 Bed & Breakfast**: Brabant en Limburg
- 2130 Nieuwsuur**
- 2129 Nachtdieren**: West-Brabant
- 2120 The Machines That Built The World**: Afllevering 4
- 2130 Noord Scoort**: Afllevering 8
- 2130 Huffers op Beeld**
- 2125 The Best FIFA Football Awards™ 2022**

## 4. Start daily job (generate EPG) with Crontab on Pi

---

### Cronjob on Pi:

7.40 hrs start epg.sh script (in /home/pi/ which generates the EPG guide using ./run.net.sh)  
10.10 hrs guide from Pi (in /home/pi/apps/.wg++) to iMac in dir ./wg++  
10.12 hrs guide from Pi to iMac desktop

### Launched control op iMac:

10.15 hrs launched.epgpi.plist (com.zerowidth.launched.epgpi.plist) starts (start script epgpi.sh in root dir <user> on iMac, which loads the EPG guide into EyeTV (with command open -a EyeTV guide.xml in dir ./wg++ on iMac).

### In detail:

#### epg.sh script (in /home/pi/) =>

```
pi@raspberrypi:~ $ nano epg.sh
#!/bin/bash
#
# EPG-Update script
# Rev. 25-2-2023
# B.L.J. Gaster

# PATH
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/sbin:/home/pi/apps/.wg++:/home/pi/.dotnet

# Commands to update EPG TV-guide
cd ~/apps/.wg++/
# Generate EPG guide
./run.net.sh
```

#### Crontab for user pi =>

```
pi@raspberrypi:~ $ crontab -e
# Crontab file for daily EyeTV guide update B.L.J. Gaster, 25
februari 2023
PATH=$PATH:/home/pi/.dotnet:/home/pi/apps/.wg++:/usr/bin:/home/pi

# EPG guide every day on 7.40 hrs
# Line below produces output with with logfile in tmp/cron
# 40 7 * * * /usr/bin/sh /home/pi/epg.sh >>/tmp/cron.log 2>&1
40 7 * * * /usr/bin/sh /home/pi/epg.sh

# Transport Guide to iMac for EyeTV
10 10 * * * /usr/bin/scp /home/pi/apps/.wg++/guide.xml
<username>@<IP-adress iMac>:/Users/<username>/.wg++/guide.xml

# Test transport to desktop on iMac
12 10 * * * /usr/bin/scp /home/pi/apps/.wg++/guide.xml
<username>@<IP_adress iMac>:/Users/<username>/Desktop/guide.xml

# Test every minute transfer to iMac. Logfile in /tmp/cron.log
# * * * * * /usr/bin/scp /home/pi/apps/.wg++/guide.xml <username
iMac>@<IP_adress iMac>:/Users/bljgaster/Desktop/guide.xml >>
/tmp/cron.log 2>&1
```

## Launched control on iMac:

At 10.15 hrs launched.epgpi.plist (com.zerowidth.launched.epgpi.plist) starts (start script epgpi.sh in root dir <user> on iMac, which loads the EPG guide into EyeTV (with command open -a EyeTV guide.xml in dir ./wg++ on iMac).

Create and edit the files below with the TextEdit.app on iMac.

**epgpi.sh =>** In directory /<user> on iMac.

```
#!/bin/bash
# EPG-Update script
# Rev. 25-2-2023 Load guide in EyeTV. Guide already loaded from Pi
# B.L.J. Gaster

# PATH
PATH=/usr/local/bin:/usr/local/sbin:~/bin:/usr/bin:/bin:/usr/sbin:/sbin:./wg++:/Library/Frameworks/Mono.framework/Versions/Current/bin

# Command for load EPG guide into EyeTV application
cd ~/.wg++/
open -a EyeTV guide.xml
```

**com.zerowidth.launched.epgpi.plist =>** In directory /<user>/Library/LaunchAgents on iMac.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist PUBLIC "-//Apple Computer//DTD PLIST 1.0//EN"
"http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>Label</key>
  <string>com.zerowidth.launched.epgpi</string>
  <key>Program</key>
  <string>/Users/<username>/epgpi.sh</string>
  <key>RunAtLoad</key>
  <true/>
  <key>StartCalendarInterval</key>
  <dict>
    <key>Hour</key>
    <integer>10</integer>
    <key>Minute</key>
    <integer>15</integer>
  </dict>
</dict>
</plist>
```

**Done! Now make a backup image of the Pi SD card.**



## 5. Background information

---

WG++

<http://webgrabplus.com/documentation/installation/raspberry-pi>

<http://www.webgrabplus.com/node/22>

.NET5 Install with script

<https://learn.microsoft.com/nl-nl/dotnet/core/install/linux-debian>

To check the list of dependencies, go to <https://learn.microsoft.com/dotnet/core/install>, select your operating system and check the "Dependencies" section.

Check dependencies (NET).

<https://github.com/dotnet/core/blob/main/release-notes/7.0/linux-packages.md>

**Download the .NET install script for ARM32 (Debian11 op Pi3B+) from**

<https://dot.net/v1/dotnet-install.sh>

See here for bashrc (add the .dotnet directory to \$PATH)

<https://learn.microsoft.com/en-us/dotnet/iot/deployment>

Install and use Microsoft Dot NET 6 with the Raspberry Pi

<https://www.petecodes.co.uk/install-and-use-microsoft-dot-net-6-with-the-raspberry-pi/>

Enable snaps on Raspberry Pi and install .NET Core SDK

<https://snapcraft.io/install/dotnet-sdk/raspbian>

.NET install in Linux without package management

<https://learn.microsoft.com/nl-nl/dotnet/core/install/linux-scripted-manual>

Download .NET 6.0.14 for ARM32 (Debian11 Bullseye op Pi 3B+) for manual install

<https://dotnet.microsoft.com/en-us/download/dotnet/thank-you/runtime-6.0.14-linux-arm32-binaries>

Visit <https://dotnet.microsoft.com/download> to get more info about the installer.

SCP info (copy file from iMac to Pi and v.v.)

<https://pimylifeup.com/scp-command-linux/>

<https://spellfoundry.com/docs/copying-files-to-and-from-raspberry-pi-and-mac/>

Crontab

[https://crontab.guru/#40\\_7\\_2\\_2\\_4](https://crontab.guru/#40_7_2_2_4)

<https://www.cyberciti.biz/faq/howto-linux-unix-start-restart-cron/>

Pi = 32 bit OS (use ARM32)

<https://forums.raspberrypi.com/viewtopic.php?t=305629>

## How to Remove Directories on Raspberry Pi

<https://linuxhint.com/remove-directories-raspberry-pi/>

Example of delete 2 unnecessary folders:

```
pi@raspberrypi:~ $ rm -rf /home/pi/apps/dotnet-test
```

```
pi@raspberrypi:~ $ rm -rf /home/pi/.dotnet/shared/Microsoft.AspNetCore.App
```

