CH341a
The $1/$10/$20 Hacking Tool You Need in Your Toolbox

April 04, 2025

By Aaron Grothe
https://www.grothe.us

Introduction

CH341a?

This talk will be a quick intro to using a CH341a EEPROM BIOS USB Programmer

- Dump a rom
- Inspect the rom
- optionally modify the binary and push it to device

Introduction (Continued)

If you have questions/comments please feel free to ask them anytime. You don't have to hold them until the end of the talk.

If there are other resources similar to these that you think might be useful to people please let the group know.

Hopefully this will be an interactive and productive session.

Disclaimer #1: (Bricking Your Stuff)

Quick note: A lot of CH341a devices run at 5v and come with 1.8v adapters. Some chips want 3.3v or something else.

You can brick devices by doing stuff incorrectly, or even correctly possibly.

Use caution. Research the device you're working with a bit. Look up your flash chips.

You have been warned.

Disclaimer #2: (Legal)

I am not a Lawyer

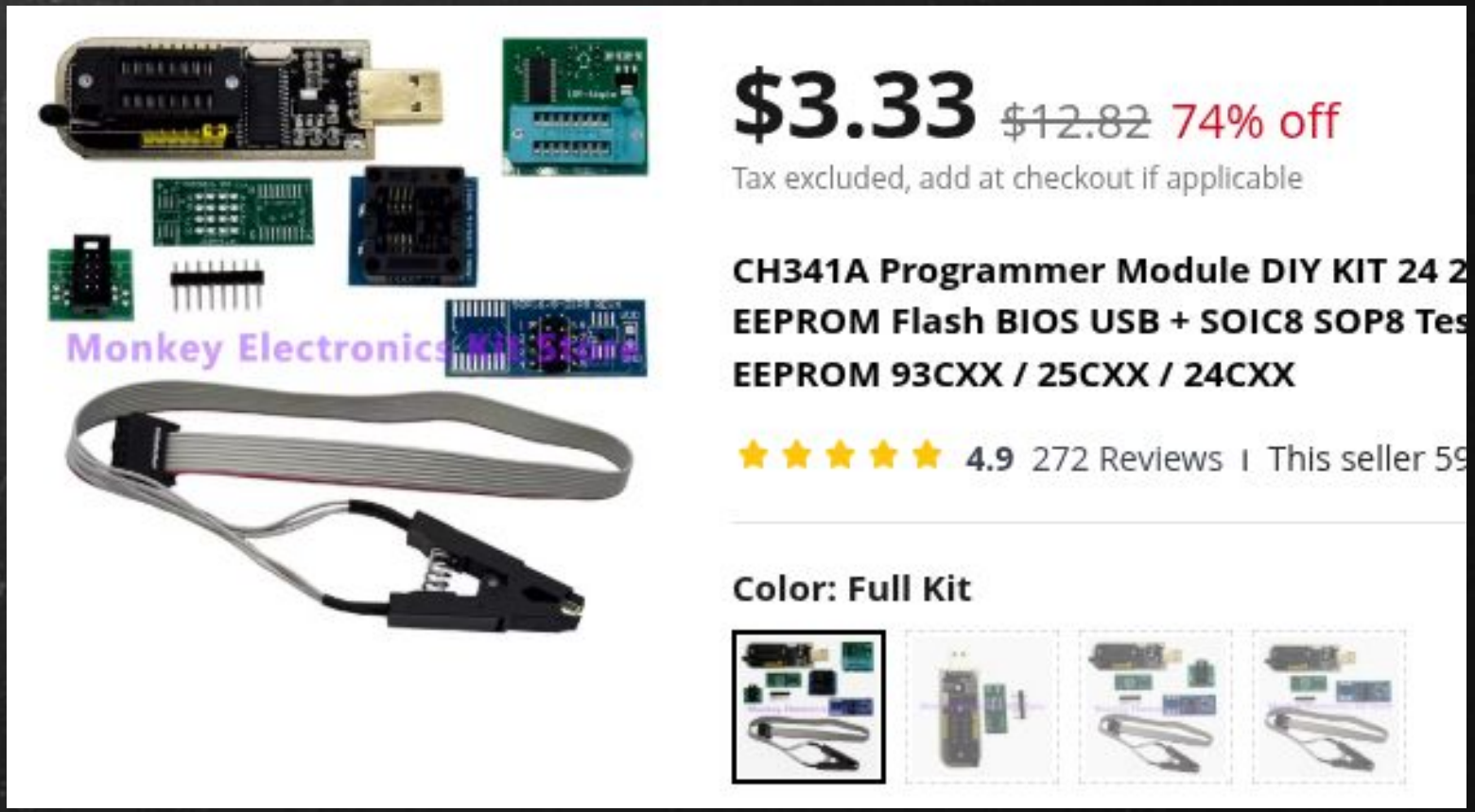
Is dumping a rom from a device and working with it legal? Believe so if you own the device, and it is for personal research purposes.

Keep in mind if you accept a Terms of Service for a device, you might be giving away some rights when you power it up.

Here's a CH341a on Amazon

Costs about $15.00 and is available the next day :-)

Or Aliexpress - if you're not in a Hurry



Costs about $5.00 if you're not in a hurry.

How does it Work?

The CH341a uses the SPI (Serial Peripheral Interface) Protocol to talk to the chip.

It sends the chip the right Opcodes to emit the firmware. This is what the CPU does when the device boots.

Works with almost any device that has NOR flash, which is a lot of them.

Device

We'll be looking at this device today

TPlink - TL-WR841N v 9.1

Router by tplink - basic mips based router

Yes from the TPlink :-)

Goals

- Understand how to use a CH341a to dump a rom
- Explore how to extract and inspect a rom
- Examine the rom
- Optionally modify the rom
- Explore how to push a modified rom back to a device

Tools (Part 1)

This is a quick list of some of the tools we'll be using today

(Primary)

- Flashrom - basic tool for dumping/pushing roms with CH341a
- Bvi - binary editor tool, vi like tool
- Binwalk - tool for extracting roms intelligently
- Diffoscope - very binary comparison tool

Tools (Part 2)

This is a quick list of some of the tools that might be helpful to you

(Secondary)

- IMSprog - Linux chip programmer for CH341a
- Binfmt_misc - linux tool to run other architectures via qemu

How to use it? High Level (Part One).

Quite simply you do the following

- Crack open the device
- Locate the NOR flash chips
- Confirm the voltage
- Put the clip on the Chip
- Dump the Rom (flashrom)

How to use it? High Level. (Part Two)

Quite simply you do the following

- Examine the rom
- Extract the rom (binwalk)
- Examine the extracted firmware
- Modify the firmware if desired
- Push the modified firmware back to the device

Crack Open the Device

Probably start online with a quick search for the device

ifixit.com - is a good place to start

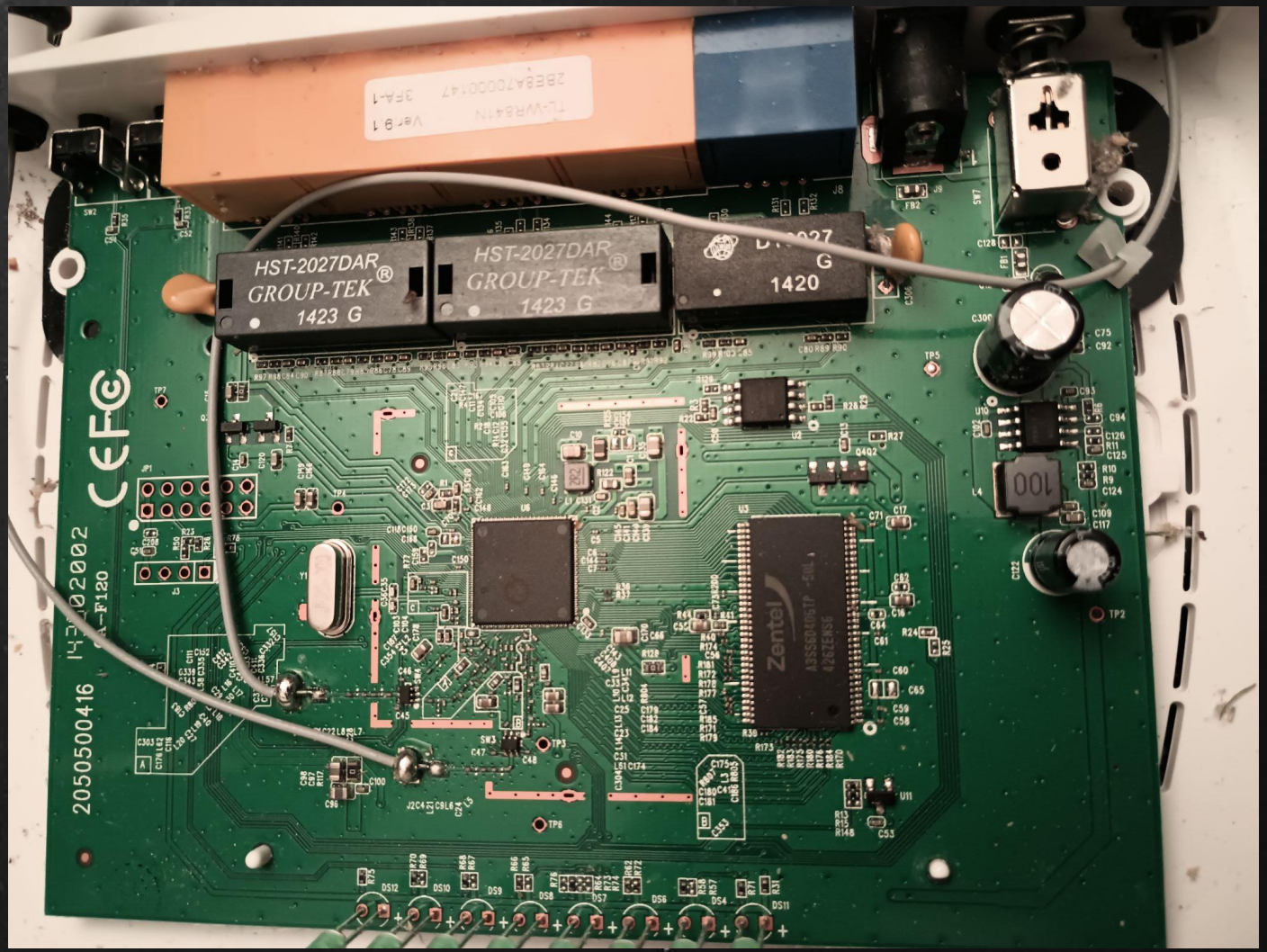
Youtube is another good source

- You can always "open" a device, but if you want to close it again you might want to do a bit of research

Screws can be hidden under rubber feet, labels etc.

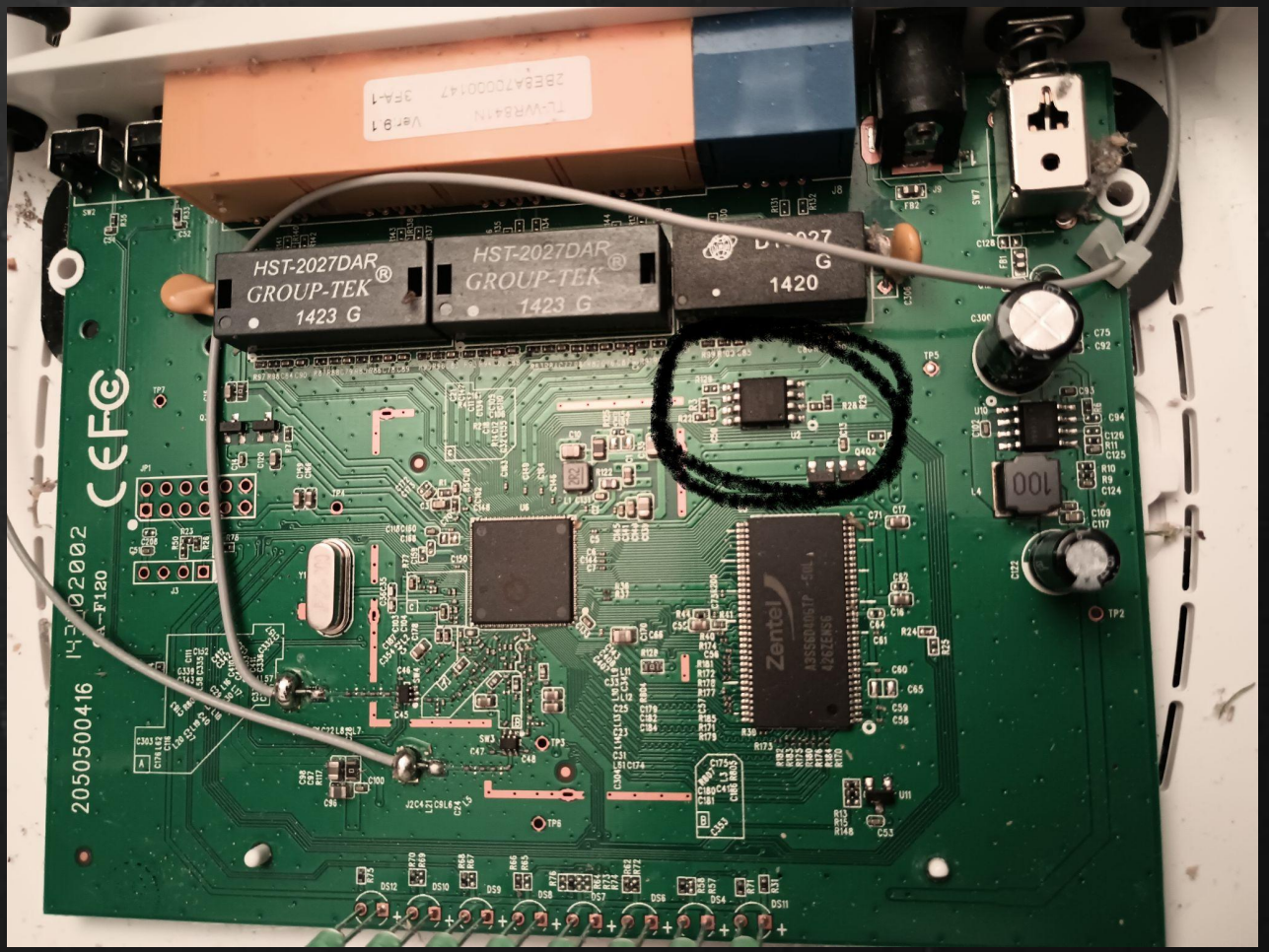
Locate the Flash Chips

Tplink



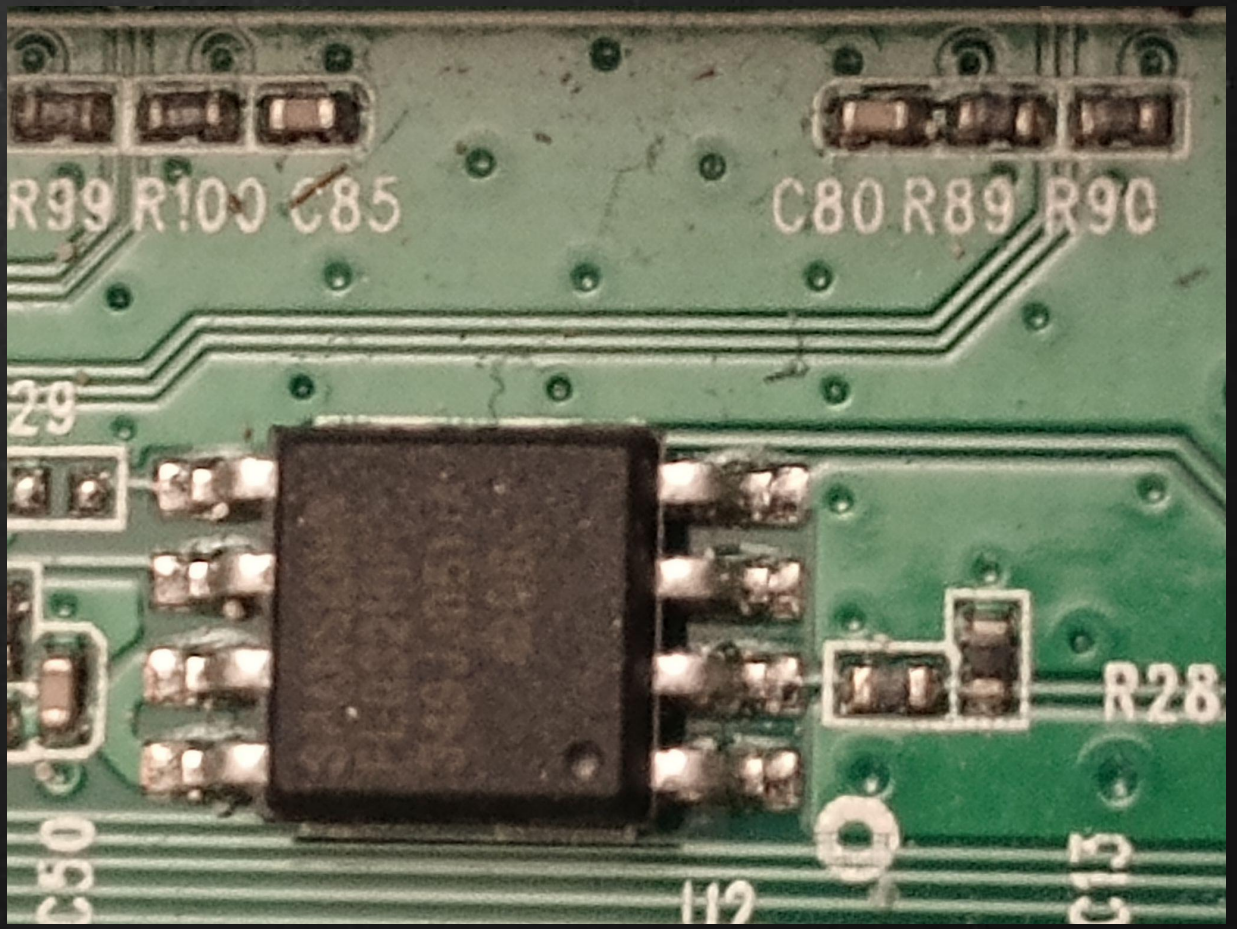
Locate the Flash Chips

NOR Flash chip circled



Locate the Flash Chips

Locate pin 1 - look for a triangle, or dot. Here it is on the tplink



Confirm the Voltage

There are two options here.

- Ignore it and hope for the best, how I started
- Break out a multi-meter and check the voltage on pins 1 and 5

Going with Option #2, it comes back with 3.3 v

Setting the Voltage

You'll get a series of adapters with your CH341a usually

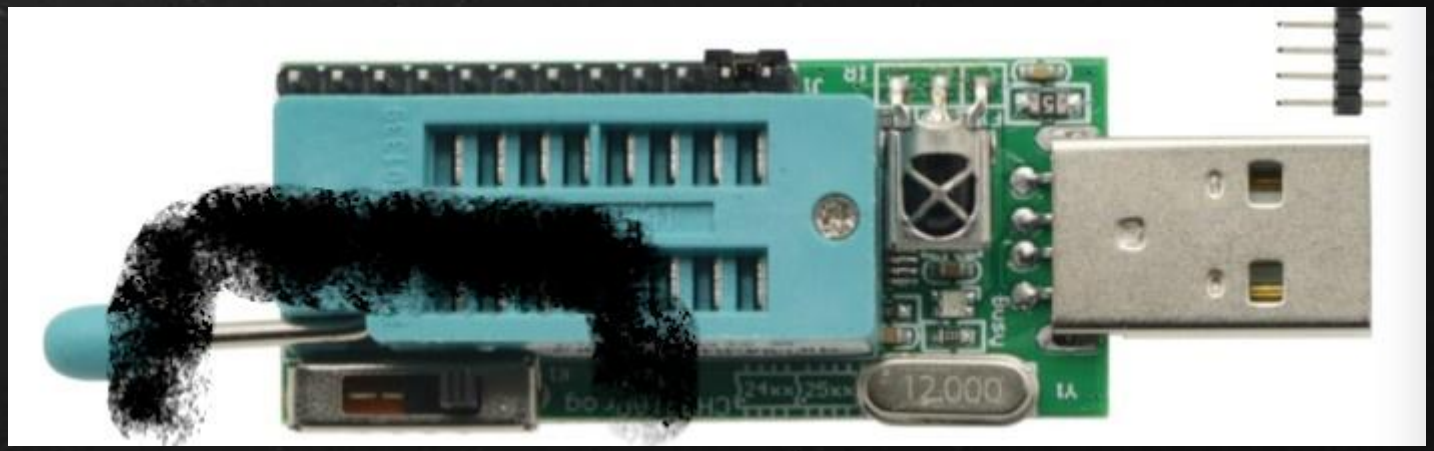
I bought a CH341a Progammer v1.7 which has adjustable voltage

Setting the Voltage

You'll get a series of adapters with your CH341a usually

I bought a CH341a Progammer v1.7 which has adjustable voltage

It is at the bottom of the board



Setting the Voltage

Selectable Voltages

- 5.0 v
- 3.3 v
- 2.5 v
- 1.8 v

You set the voltage by pushing it away from the usb-a interface

Accessing the Chip

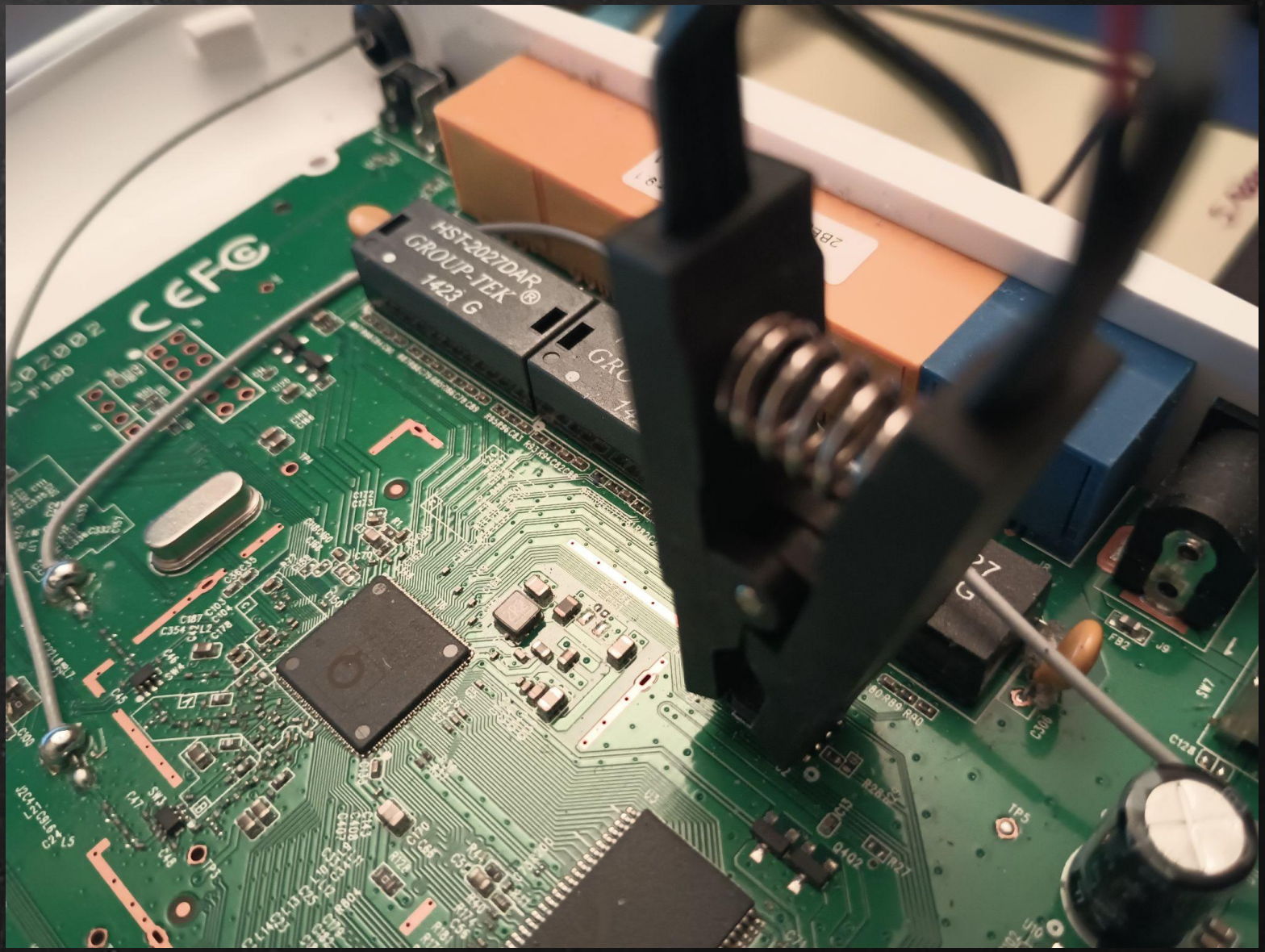
There are two ways to access the contents of the chip

- Desolder the chip and put it into the programmer
- Clip on top of the chip without having to do anything

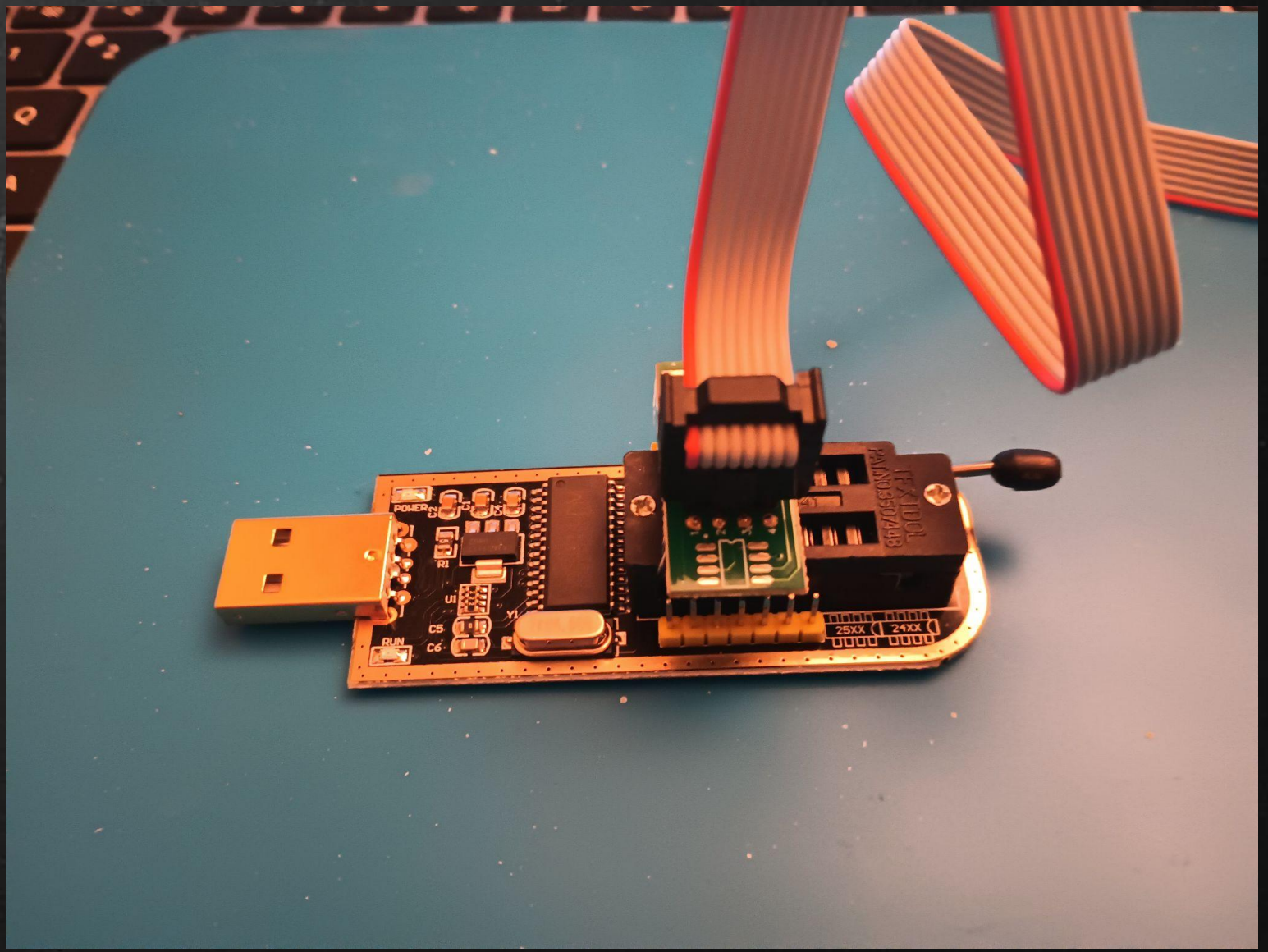
So we'll go with the second option

Your CH341a will come with cables and clips usually

Chip on The Clip



Ribbon Cable on the CH



Dump the Rom

We'll use flashrom to dump the rom

sudo flashrom -V -r tl841n.bin -p ch341a_spi

What this command does

- -V verbose
- -r read
- -p programmer name # ch341a_spi

This will take a couple of minutes usually

Dump the Rom

Part of the output

You can also try to follow the instructions here:
https://www.flashrom.org/contrib_howtos/how_to_mark_chip_tested.html
Thanks for your help!
Reading flash... read_flash: region (00000000..0x3fffff) is readable, reading range (00000000..0x3fffff).
done.

Make some Changes and dump again

Changed the default admin account and password to openwrt

And dumped the rom again

Changed rom named tl814n_changed.bin

Examine the Rom

We'll do strings on the rom to get an idea of what is in there.

$ strings tl841n.bin

Now we'll open the dump in bvi (binary vi)

$ bvi tl841nf.bin

Note: using regular vi will put an EOF character at the end of the file which will screw up attempted uploads.

Examine the Rom

Look for password - not set

We'll take a look at the other file now

$ bvi +/openwrt tl841n_changed.bin

So now we know where the username/password are set in the rom

We can change this and push it back to device to reset password

Examine the Rom

Screenshot of admin account and password for modified rom

```
003E4A30  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
003E4A40  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
003E4A50  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
003E4A60  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
003E4A70  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
003E4A80  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
003E4A90  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
003E4AA0  00 00 00 00 61 64 6D 69 6E 00 00 00 00 00 00 00  ....admin.......
003E4AB0  00 00 00 00 61 64 6D 69 6E 00 00 00 00 00 00 00  ....admin.......
003E4AC0  00 00 00 00 6F 70 65 6E 77 72 74 00 00 00 00 00  ....openwrt.....
003E4AD0  00 00 00 00 6F 70 65 6E 77 72 74 00 00 00 00 00  ....openwrt.....
003E4AE0  00 00 00 00 FF FF FE 20 00 00 00 00 00 00 00 00  ....... ........
003E4AF0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
003E4B00  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```

Extracting the Rom

We can use binwalk to extract all of the rom contents and make it a lot easier to use

$ binwalk -Me tl841n.bin

Options

-M “matryoshka” - recursively scan the file
-e “extract” - extract all the files

Examining the Firmware

So now we have an extracted copy of binary

$ cd _tl841n_changed.bin

Let's look for the html pages that make up the interface

$ find . -type f -name '*htm' -print

And viola the whole web interface is exposed

Examining the Firmware

Let's look for the password file

$ find . -type f -name 'passwd'

Vi the file

$ vi ./squashfs-root/etc/passwd

Examining the Firmware

Take a look at the shadow file

$ vi ./squashfs-root/etc/shadow

Let's google the shadow and get the password for this

So it is "sohoadmin"

Good to know

Examining the Firmware

Does the system use busybox?

```
$ find . -type f -name 'busybox'
```

```
$ cd squashfs-root/bin
```

Examining the Firmware

Ask file about the file type

$ file busybox

busybox: ELF 32-bit MSB executable, MIPS, MIPS32 rel2 version 1 (SYSV), dynamically linked, interpreter /lib/ld-uClibc.so.0, no section header

Examining the Firmware

Grab the version

$ strings busybox | grep v1

Output is as follows

BusyBox v1.01 (2013.11.29-02:54+0000) Built-in shell (msh)
klogd started: BusyBox v1.01 (2013.11.29-02:54+0000)
BusyBox v1.01 (2013.11.29-02:54+0000) multi-call binary

Examining the Firmware

Let's run the ls program from busybox to get an idea about the version, etc.

```
$ qemu-mips -L
/home/grothe/pres/laptop/_tl841n.bin.extracted/squashfs
-root ls -h
```

Output as follows

```
ls: cache '/etc/ld.so.cache' is corrupt
ls: invalid option -- h
BusyBox v1.01 (2013.11.29-02:54+0000) multi-call binary
```

Examining the Firmware

Being able to run the executables on our system can give us a way to further inspect the executables

Be VERY careful with this, as you're not running in a locked down environment

A potentially useful way to inspect, run parts of the system

Running the full environment inside qeum is outside the scope of this talk

Let's look at the Extracted System

Check out potential vulnerabilities against busybox

https://www.cvedetails.com/vulnerability-list/vendor_id-4282/product_id-7452/version_id-475435/Busybox-Busybox-1.01.html

So quite a few potential vulnerabilities are exposed to system

Diffing Roms

We'll do a quick comparison between the original rom and the rom where we've changed the password

$ diffoscope –html comparison.html _tl841n.bin.extracted _tl841n_changed.bin.extracted

$ google-chrome comparison.html

Quick overview of changes made to binary

Pushing Roms

Flashrom can also be used to push roms to the NOR chip if it writable, which most of the time it is

Note: the chip has to have been automatically detected by flashrom when dumping, can't do -c and override chipset

$ sudo flashrom -V -p ch341a_spi -w romfile

This is where things can get interesting.

$20/$50 Toolbox

$20 Toolbox (Jackson Toolbox)

$10 CH341a (Aliexpress)
$5 Multimeter (Harbor Freight)
$5 Device to break (Goodwill bytes)

$50 Toolbox (Grant Toolbox)

$15 CH341a v1.7 (adjustable voltage) (Amazon)
$15 Multimeter better (Amazon)
$20 TPlink 841n (Amazon)

Hard Lessons - Things I Wish I Knew B4

1. CH341a v1.7 (adjustable voltage is very nice)
2. Getting a known device like the TP-link 841n is a big help
3. You're going to let the smoke out of something
4. Buying a device with UART and a UART cable is not a bad idea
5. Binwalk is very awesome
6. Use the multimeter to check voltages
7. Use bvi or another tool that won't put a EOF character on an edited file

IoT

There are a lot of cheap IoT devices with these flash chips. You can learn a lot in the process.

Some devices such as the TL841n you can dump the Bios via Serial/UART, but with the CH341a you don't need to even power on the device to get the firmware

Other Devices

The CH341a programmer is a good cheap device but there are a couple of other tools out there that are pretty popular

EZP2023+
TL866-G3 T48

Both are supposed to be faster than the CH341a, but may not support the same chips

Both are available on Aliexpress as well :-)

EZP2023+ on Amazon

EZP2023 Programmer USB SPI 9 Adapters EZP2019 Upgraded Chip Writer Reader IC Test Clip 24 25 93 95 EEPROM Flash Bios Minipro Socket

Brand: MORIENZI

5.0 ★★★★★ (4) | Search this page

$32.98

prime

FREE Returns

Brand	MORIENZI
Media Type	Multiple values from ['Memory Stick', 'SDHC', 'MMCmicro', 'MMCmobile', 'Memory Stick PRO HG Duo', 'CFexpress', 'RS-MMC', 'SDXC', 'Mem...

TL866-G3 T48 on Amazon

Roll over image to zoom in

T48 TL866-3G Programmer Support 31000+ ICS for EPROM/MCU/SPI/Nor/NAND Flash/EMMC/IC Tester/ TL866CS TL866II Plus Replacement with 9 Adapter(T48 Programmer Host+ 9 Adapter)

Visit the ACEIRMC Store

4.6 ★★★★☆ (131) | Search this page

Amazon's Choice

50+ bought in past month

$72.99

Or **$12.17** mo (6 mo). Select from 2 plans

prime

FREE Returns

Color: **T48 Programmer host+ 9 Adapter**

$72.99	$84.99	1 option from $142.99	$61.99
FREE Delivery **Thursday**	FREE Delivery **Wednesday**		FREE Delivery **Tomorrow**

Brand ACEIRMC

Other Uses

If you do a youtube search for "ch341a bios apple", you'll see some interesting research being done on bypassing EFI locked passwords on macs

Lot of IoT devices are laid out similarly to the TL841n

Recovering devices is used by a lot of people

Adding binaries to a rom and pushing it to the device isn't covered in this talk, there is a link to it in the references section

So that is a Quick Overview

What we were hoping to go over today is as follows

- The CH341a is a very nice, very cheap tool
- You can do a lot with the firmware once you get it off the device
- A lot of IoT devices look a lot like the devices we're talking about today

Thank You & Questions

Thank you for Coming Today

The slides for the talk will be on my website https://www.grothe.us in the presentation section tonight/tomorrow

If you have any questions/comments please feel free to ask me at ajgrothe <at> gmail.com

Questions

Links - (Youtube Refs)

- Youtube: How is this Hacking Tool Legal - https://www.youtube.com/watch?v=X-Lzq7jAT8I
- Youtube: Don't use CH341a until you watch this! - https://www.youtube.com/watch?v=MMyDvb_v4uc
- YT: Bios Flash Programmers Compared - https://www.youtube.com/watch?v=wTt4wq2Y-zs
- YT: Using CH341a to recover a bricked motherboard - https://www.youtube.com/watch?v=FJ0rAM-N7tY

Links - Other Refs

- SNANDER: Another tool for processing roms - https://github.com/McMCCRU/SNANDer
- Hardware breakdown of the TL-WR841N - https://github.com/adamhlt/TL-WR841N
- Adding program to Device's rom TL-WR841N - https://github.com/JulianOzelRose/TL-WR841N-v14