



Breaking Down Binaries

Navigating the Labyrinth of IoT Firmware Analysis

Edwin Shuttleworth



whoami

- Senior Security Researcher @ Finite State
 - ~6 years of startup life
- Chicago based
- Indoor boulderer
- Latest hyperfixation: Leatherworking





Outline

Why does any of this matter

The Flavors of Firmware

Firmware Acquisition

Unpacking

Analysis



Motivation





You can't opt out of IoT

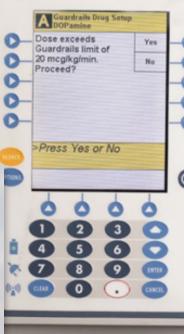
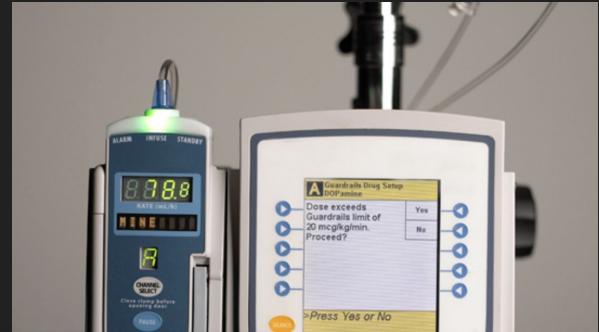


EFF

Street-Level Surveillance

Drones
License plate readers
Cell-site simulators
Body-worn cameras
Biometrics (face recognition)

Police databases
Gunshot detection
Camera networks
Social media monitoring
Real-time crime centers





Threat Surface

- Long update cycles
 - Even for FOTA, updates usually need to be tested to a greater extent
- Generally very behind compared to the desktop
 - Parker Wiksell - Vulnerability Trends in the Supply Chain DEF CON 28 IoT Village
 - Really rough OWASP Top 10

The image is a graphic titled "OWASP TOP 10 INTERNET OF THINGS 2018". It lists ten threats, each numbered 1 through 10, with a corresponding icon and a brief description.

Rank	Threat Category	Description
1	Weak, Guessable, or Hardcoded Passwords	Use of easily bruteforced, publicly available, or unchangeable credentials, including backdoors in firmware or client software that grants unauthorized access to deployed systems.
2	Insecure Network Services	Unneeded or insecure network services running on the device itself, especially those exposed to the internet, that compromise the confidentiality, integrity/authenticity, or availability of information or allow unauthorized remote control...
3	Insecure Ecosystem Interfaces	Insecure web, backend API, cloud, or mobile interfaces in the ecosystem outside of the device that allows compromise of the device or its related components. Common issues include a lack of authentication/authorization, lacking or weak encryption, and a lack of input and output filtering.
4	Lack of Secure Update Mechanism	Lack of ability to securely update the device. This includes lack of firmware validation on device, lack of secure delivery (un-encrypted in transit), lack of anti-rollback mechanisms, and lack of notifications of security changes due to updates.
5	Use of Insecure or Outdated Components	Use of deprecated or insecure software components/libraries that could allow the device to be compromised. This includes insecure customization of operating system platforms, and the use of third-party software or hardware components from a compromised supply chain.
6	Insufficient Privacy Protection	User's personal information stored on the device or in the ecosystem that is used insecurely, improperly, or without permission.
7	Insecure Data Transfer and Storage	Lack of encryption or access control of sensitive data anywhere within the ecosystem, including at rest, in transit, or during processing.
8	Lack of Device Management	Lack of security support on devices deployed in production, including asset management, update management, secure decommissioning, systems monitoring, and response capabilities.
9	Insecure Default Settings	Devices or systems shipped with insecure default settings or lack the ability to make the system more secure by restricting operators from modifying configurations.
10	Lack of Physical Hardening	Lack of physical hardening measures, allowing potential attackers to gain sensitive information that can help in a future remote attack or take local control of the device.



Types of Firmware



Linux-y Firmware

- Linux (and Android) is well understood and documented
- Most complicated relative to RTOS or bare metal
 - Full featured operating system
- Costs way less than you might expect
 - Sub \$3 Linux-on-a-business card (pre-covid)

Component	Price
F1C100s	\$1.42
PCB	\$0.80
8MB flash	\$0.17
All other components	\$0.49
Total	\$2.88





RTOS (Real Time OS) based firmware

- Single monolithic binaries
- May be closed source
- Highly configurable so existing research may or may not apply



```
## Starting application at 0x4010100000 ...

VxWorks 7 SMP 64-bit
Core Kernel version: 1.0.0.0
Build date: May 30 2014 10:51:05
Copyright Wind River Systems, Inc.
1984-2014

Board: Wind River Dev Kit MP8
CPU Count: 8
OS Memory Size: 1899MB
ED&R Policy Mode: Deployed

Adding 5290 symbols for standalone.

[vxworks]# i
```

NAME	TID	PRI	STATUS	PC	ERRNO	CPU #
tJobTask	40104cdbc0	0	PEND	401020c83c	0	-
tExcTask	40102a073c	0	PEND	401020c83c	0	-
tLogTask	40104d01d8	0	PEND	401020b0f0	0	-
tShell10	40105c1d30	1	READY	401021e008	0	0
ipcom_tick>	401057a990	20	PEND	401020c83c	0	-
tVxdbgTask	401057dc20	25	PEND	401020c83c	0	-
tNet0	40104d3b78	50	PEND	401020c2b4	0	-
ipcom_sysl>	40104c9810	50	PEND	401020d3d4	0	-
tNetConf	40105a6e40	50	PEND	401020c83c	0	-
miiBusMoni>	40104d5e08	252	DELAY	4010215640	0	-
ipcom_egd	4010583c20	255	DELAY	4010215640	0	-
tidleTask0	40102a2fb0	287	READY	401020c004	0	-
tidleTask1	40102a7220	287	READY	401020c00c	0	1
tidleTask2	40102ab490	287	READY	401020c004	0	2
tidleTask3	40102a2fb0	287	READY	401020c004	0	3
tidleTask4	40102b1700	287	READY	401020c004	0	4
tidleTask5	40102b2440	287	READY	401020c004	0	5
tidleTask6	40102a4620	287	READY	401020c004	0	6
tidleTask7	40102a4860	287	READY	401020c004	0	7

```
[vxworks]#
```



Bespoke Bare Metal Firmware

- You'll have to RE everything yourself
- Least complicated overall
 - No threads, no tasks, basically one big c file
- Very cheap → very prevalent
- Low power has security implications
 - Impacts on encryption and random number generation

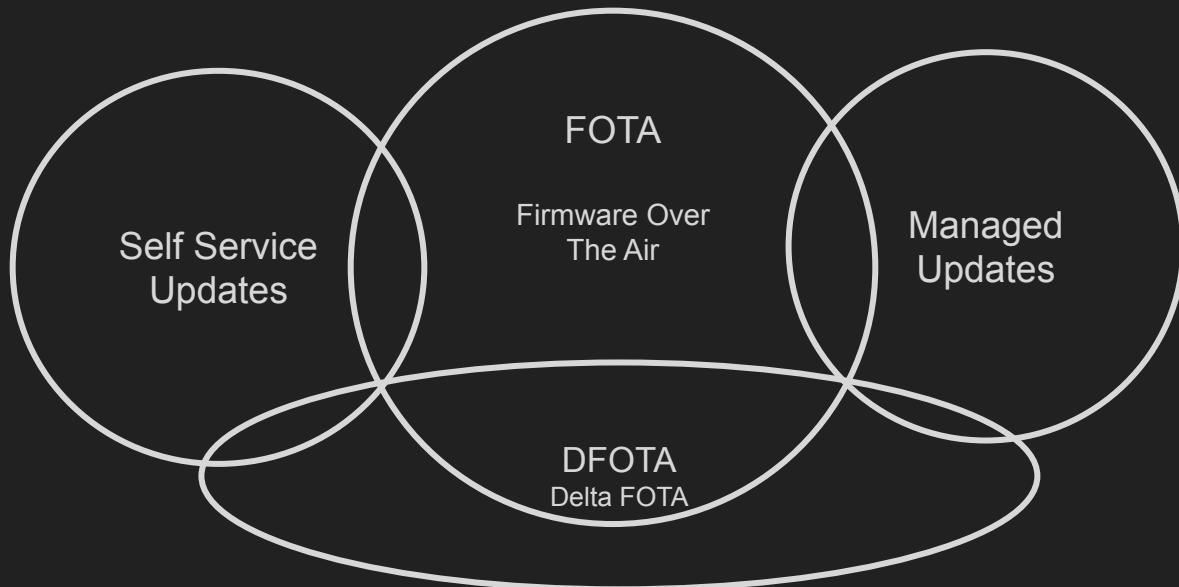




How to get firmware 101



How do updates happen anyways





Synology

Download Center

You can find the documents and files regarding the operating system, packages, desktop utilities and so on for your Synology product to enjoy the latest and versatile features.

Please select the product category and the corresponding model you use.

NAS DS120j

Search results for: DS120j

OS Version

Operating System Desktop Utilities Packages Documents Android Apps

Operating System

DSM 7.2.2

DSM is the operating system of DS120j.

[Download](#)

[MDS](#)

[Release Note](#)

[All Downloads](#)

Download pages

Blueskysea®

HOME

PRODUCTS

DOWNLOAD

NEWS

BLOG

SEARCH

EN

Home > Firmware

Firmware

Video Player

[B1W V20200707 Firmware\(solved iPhone Choppy Playback\)](#)



Firmware

[B2W V20191030 Firmware\(IP screen version\)](#)



User Manual

[B2W V20190726 Firmware\(TFT screen version\)](#)



[B1W Standard 20180528s Firmware](#)



Industries &
applications

Products &
solutions

Learning &
support

Where to buy

Firmware releases for all our products

Usually, you can choose between two tracks: **active** and **long-term support (LTS)**. In the active track, we continue to add features in addition to improving on cybersecurity and stability. In the LTS track, we don't add any features. The focus is to improve on cybersecurity and stability. If it has the features you need, we recommend that you choose the LTS track.

For information on new and upcoming releases, visit the [AXIS OS portal](#).



Other Online Exposure

Index of ftp://ftp.romsat.ua/pub/Lan/

[Up to higher level directory](#)

Name	Size	Last Modified
ASCENT	9/19/2018 20:00:00	
AURORA	9/20/2018 20:00:00	
BDCOM	8/11/2020 07:04:00	
CDATA	5/12/2019 20:00:00	
CeLAN Specs&Manuals	2/2/2016 19:00:00	
Comfast	2/4/2019 19:00:00	
ECView_Pro	7/21/2016 20:00:00	
EXTREME	10/30/2017 20:00:00	
Edge-Core Pictures	2/2/2016 19:00:00	
Edge-core Spec&Manuals	9/15/2020 04:16:00	
Firmware CeLAN	6/19/2014 20:00:00	
Firmware Edge-Core	2/2/2020 19:00:00	
IgniteNet	11/24/2016 19:00:00	
Juniper	10/30/2017 20:00:00	
MIMOSA	8/31/2016 20:00:00	
Marvo	4/10/2019 20:00:00	
OptoLink	5/28/2019 20:00:00	
PICOTEL	12/18/2019 19:00:00	
Prevail	3/22/2018 20:00:00	
RCI	8/21/2019 20:00:00	
SFP Datasheet	8/7/2019 20:00:00	
Skycom	4/9/2019 20:00:00	
TELMOR	7/30/2018 20:00:00	
TOTOLINK	9/11/2017 20:00:00	
WISI	10/8/2019 20:00:00	

Huawei firmware files found on update server

[huawei-fw-list.txt](#) [Raw](#)

```
1 =====
2 DO NOT WRITE ANY QUESTIONS IN COMMENTS
3 =====
4 This is not appropriate place for discussions. Keep this list FW-only.
5 I do NOT have any firmware files apart from published here or on 4pda. Please do not contact me for firmware files requests.
6
7
8 This is a list of files found on Huawei update server by brute-forcing URL parameters.
9 Some firmware files have changelogs. Just change file name to "changelog.xml" in the end of the URL.
10
11 Example:
12
13 File:
14 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v10149/f1/full/E3276Update_21.430.03.04.55_UTPS22.001.18.76.55_MAC22.
15
16 Changelog: http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v10149/f1/full/changelog.xml
17
18
19
20 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v10149/f1/full/E3276Update_21.430.03.04.55_UTPS22.001.18.76.55_MAC22.
21 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v16808/f1/full/BV7R2CUpdate_21.298.00.00.55.gz.bin
22 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v17769/f1/full/UTPS22.001.19.05.55_MAC22.001.19.05.55.exe
23 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v18041/f1/full/UTPS22.001.19.05.55_MAC22.001.19.05.55.exe
24 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v18042/f1/full/UTPS22.001.19.05.55_MAC22.001.19.05.55.exe
25 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v18050/f1/full/UTPS22.001.19.05.55_MAC22.001.19.05.55.exe
26 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v18807/f1/full/B710CUpdate_21.236.11.04.54.gz.bin
27 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v20245/f1/full/E3276Update_21.430.03.04.55.exe
28 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v21210/f1/full/B1NvB710CUPDATE_V200R001B180D35SP03C748.BIN
29 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v23646/f1/full/UTS1000Update_21.236.03.02.55_NEBUT_17.100.05.04.55.gz.
30 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v26016/f1/full/UTPS22.001.19.11.55_MAC22.001.19.11.55.exe
31 http://update.hicloud.com:8180/TDS/data/files/p9/s115/G345/g0/v26026/f1/full/E3372Update_21.300.05.00.55.exe
```



Academic Datasets

- Scrapers for large academic datasets:
 - <https://github.com/firmadyne/scraper/>
 - <https://github.com/WUSTL-CSPL/Firmware-Dataset>

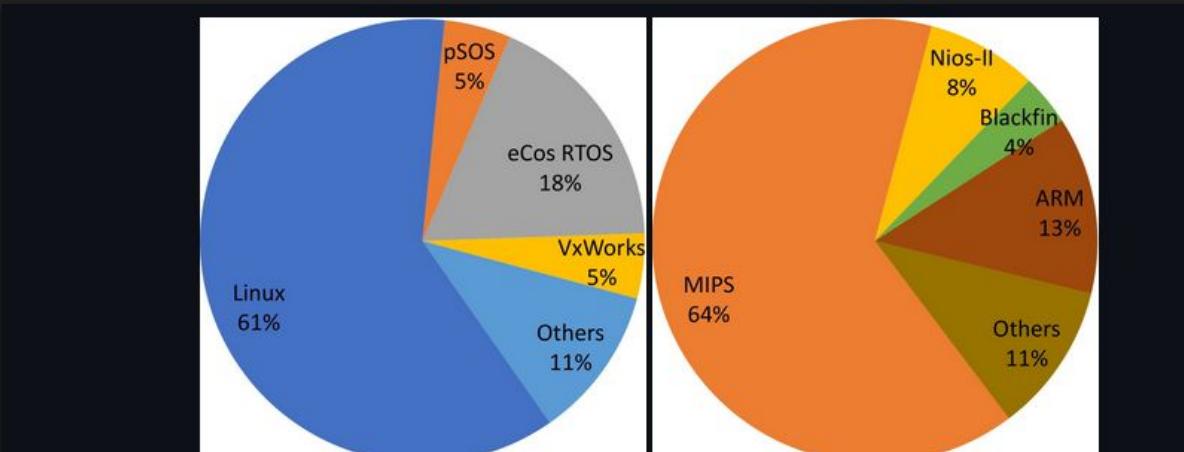


Fig.1. Firmware distribution in terms of OS (left) and architecture (right).

Graphs from <https://github.com/WUSTL-CSPL/Firmware-Dataset>



Dig Deeper with App Reverse Engineering

- Packaged with apps
- Update API/URL exposed via app
- Sometimes also exposes decryption keys and/or default passwords

```
Directory of [REDACTED]\assets\fota  
09/28/2024 12:38 PM <DIR> .  
09/28/2024 12:36 PM <DIR> ..  
09/28/2024 12:36 PM 79,461 app-[REDACTED]ref_dfu_package.zip  
06/25/2019 08:49 AM 78,824 BLE-rRF52832.bin  
06/25/2019 08:49 AM 141 BLE-rRF52832.dat  
09/28/2024 12:36 PM 106,564 DM5_L152.sim  
06/25/2019 08:49 AM 156 manifest.json
```

```
    private VerticalRangeSeekBar verticalBar;  
    private int width;  
    private String downLoadURL = "https://drive.google.com/file/d/[REDACTED]view?usp=sharing";  
    private boolean sendShutter = true;  
    private int HD Mode Style = 2;  
  
    /* JAD INFO: Access modifiers changed from: private */  
    public void stringRequestWithGet() {  
        Volley.newRequestQueue(getApplicationContext()).add(new StringRequest(1, [REDACTED], new Response.Listener<String>(this  
        {  
            this.this$0 = this;  
        }  
    }  
}
```

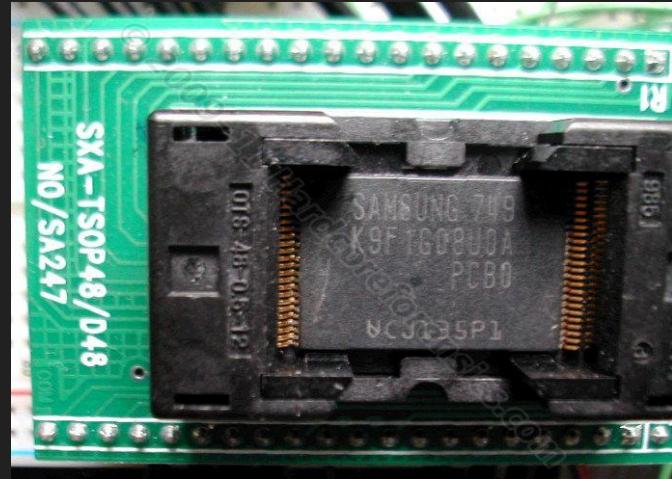
Example of Firmware Embedded in Android App

Examples of APIs and URLs in App Code



Other Methods

- Capture off the wire
- Pull it off a running device
- Read off the flash





Unpacking



Known Formats

- File systems
 - UBIFS, SquashFS, Ext*, JFFS2, YAFFS, etc
 - Usually you can either mount them (best case) or extract them with FOSS (less good)
- Container formats
 - tar, zip, etc
- Flashing formats
 - Intel Hex, Motorola SREC
- Esoterica
 - Many proprietary formats have unpackers written by the community
 - If you write something, you should post it, even if it's not perfect (I need to follow this advice)

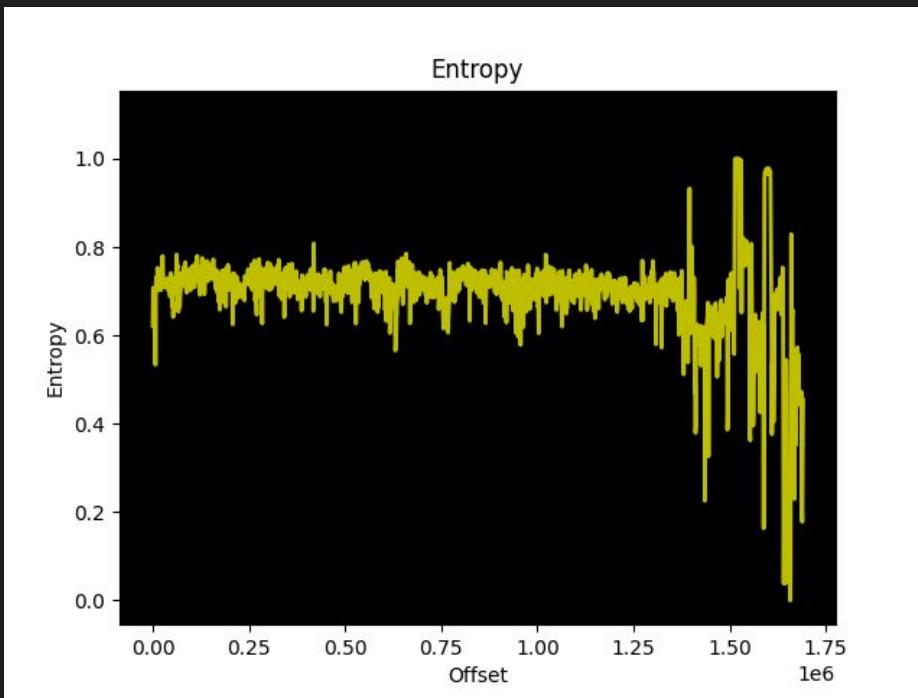


Unknown Formats

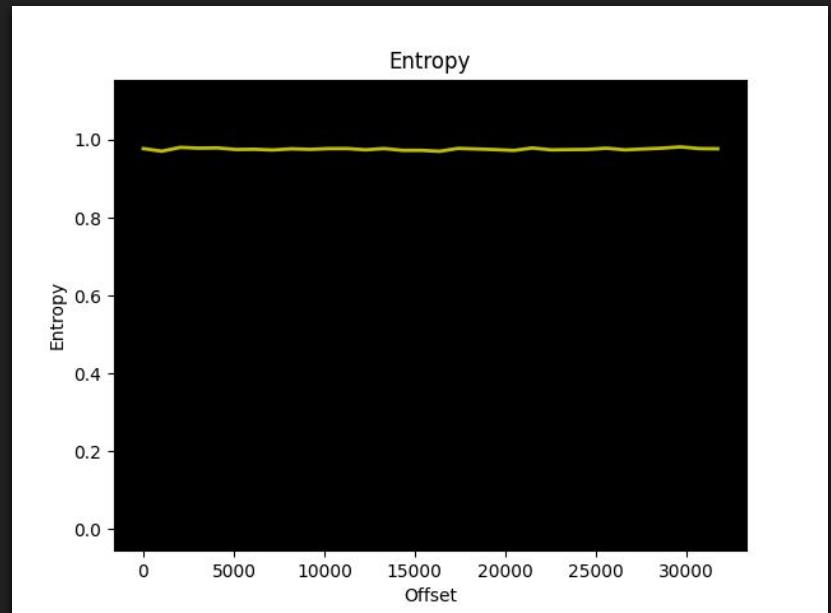
- Generic unpackers
 - Binwalk: <https://github.com/ReFirmLabs/binwalk/>
 - Currently maintained fork: <https://github.com/OSPG/binwalk>
 - Binwalk V3 in development!!! <https://github.com/ReFirmLabs/binwalk/tree/binwalkv3>
 - Unblob: <https://github.com/onekey-sec/unblob>
- Get Google'in for magic numbers
- Check entropy
 - ent
 - binwalk -E
 - unblob can do it too



A Moment on Entropy



Entropy graph of an uncompressed, unencrypted file



Entropy graph of a compressed file



Really Unknown Formats

- Make an unpacker
 - Kaitai.io is cool
 - Stare at it in a hex editor
- Think like an embedded systems engineer
 - Header?
 - File/Object Table?
 - List of Files/Objects?

```
/*
 * Legacy format image header,
 * all data in network byte order (aka natural aka bigendian).
 */
typedef struct image_header {
    uint32_t      ih_magic;        /* Image Header Magic Number */
    uint32_t      ih_hcrc;        /* Image Header CRC Checksum */
    uint32_t      ih_time;        /* Image Creation Timestamp */
    uint32_t      ih_size;        /* Image Data Size */
    uint32_t      ih_load;        /* Data Load Address */
    uint32_t      ih_ep;          /* Entry Point Address */
    uint32_t      ih_dcrc;        /* Image Data CRC Checksum */
    uint8_t       ih_os;          /* Operating System */
    uint8_t       ih_arch;        /* CPU architecture */
    uint8_t       ih_type;        /* Image Type */
    uint8_t       ih_comp;        /* Compression Type */
    uint8_t      ih_name[IH_NMLEN]; /* Image Name */
} image_header_t;
```



Analysis



Tips for Embedded Linux

- Network testing
 - Open ports and services?
 - shodan.io & censys can be used to get an idea of typical configurations
- Existing credentials, documented and... less documented
 - RTFM; RTWFM
- Software components
 - N-day CVEs
- Update mechanisms
 - CRCs, hashes, signing
 - Safe parsing of invalid formats



Some rad tools

- Emulation
 - QEMU: <https://gitlab.com/qemu-project/qemu>
 - Unicorn: <https://github.com/unicorn-engine/unicorn>
 - Firmadyne (2016): <https://github.com/firmadyne/firmadyne>
 - FirmAE (2020): <https://github.com/pr0v3rbs/FirmAE>
- End to end analysis with EMBA
 - Repo: <https://github.com/e-m-b-a/emba>
 - Unpacking, OS detection, Arch detection
 - Static and emulated dynamic analysis
 - Configuration analysis
 - Surface level SBOM + CVEs





Sandia Labs:
Reverse Engineering ICS Field Devices

The Conspiracy Board

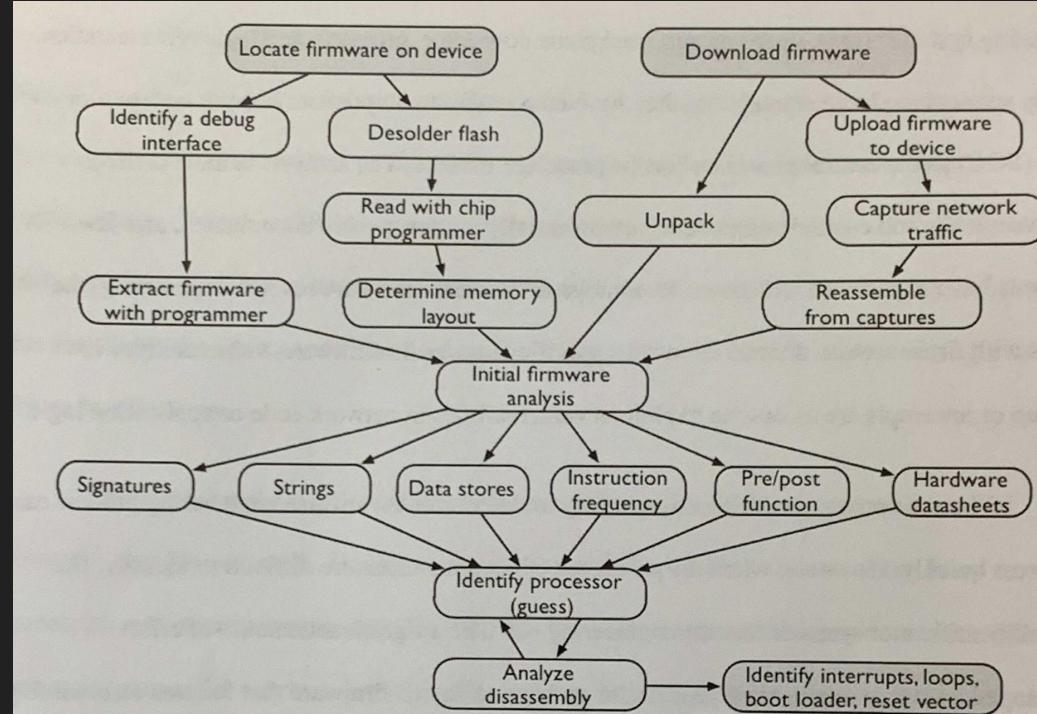


Figure 6. Process for reverse engineering embedded device firmware.

Getting the Firmware

We explored four methods of obtaining firmware—



RTOS and Bare Metal

- Identify architecture
 - Identify the chip it's running on (fccid.io may help)
 - binwalk -A
- Identify base addresses
 - Use the datasheet for the chip to start off
 - Intel Hex and SREC include addresses, you can import them right to Ghidra
 - Generic algorithm outlined in POC||GTFO 21:13 by EVM
- Identify entry/vector table
- Start looking for symbol tables/logging strings
- CMSIS System View Description (SVD)
 - Peripheral mapping for ARM Cortex-M



Questions?



Thank you!

Special thanks to Larry Pesce and Stephanie Pirman