IoT Hardware Hacking & Firmware Extraction

Overview

This repository provides resources and guides from two presentations on IoT hardware hacking and firmware extraction. The materials aim to help hardware security researchers, engineers, and enthusiasts understand, analyze, and secure IoT devices at the physical level, including techniques for reverse engineering, firmware dumping, and practical hacking defenses.[1][2]

Contents

- IoT Hardware Hacking**
  - Security overview of IoT device attacks and defenses
  - Typical device components (MCU, flash memory, radios, sensors)
  - Common attack surfaces: debug ports (UART/JTAG/SWD), external storage, wireless interfaces, leftover developer credentials
  - Tools and attack methods: UART logging, JTAGulator, OpenOCD, pyOCD, SPI programmers, fault/side-channel attacks
  - Defense checklist: disable/debug fusing, secure boot, key management, safe testing practices[2]

- Firmware Extraction Project**
  - Introduction to firmware extraction and its relevance for reverse engineering, security auditing, device repair
  - Tools:
    - Operating System: Ubuntu/Linux for flexibility and hardware interfaces
    - Flashrom (read/write/verify flash chips)
    - Binwalk (firmware analysis/extraction)
    - Hardware: CH341A USB programmer, FTDI USB-to-TTL, SOIC8 test clip, adapters, soldering tools
  - Practical steps:
    - Detect device (lsusb)
    - Install tools (`sudo apt install flashrom binwalk`)
    - Dump, verify, and analyze firmware (`flashrom`, `binwalk`)
    - Extract and inspect embedded files and code
  - Example: D-Link DIR-816 router firmware extraction workflow[1]

Getting Started

1. Review the slides in this repo for high-level concepts and technical details.
2. For hardware hacking labs, ensure test networks are isolated and testing is performed with permission as required by law.[2]
3. Install required open-source tools on a Linux system and gather necessary hardware interfaces before attempting firmware extraction.[1]
4. Follow the provided step-by-step guide for extracting and analyzing firmware from real devices.

Contributors

- Presented and compiled by Shruti Konde[1]

References and More Information
- For deeper dives into each topic, see the attached slide decks for visuals, diagrams, and examples.
- Feedback and suggestions for improvement welcome in issues or via the linked feedback form in the presentation.[1]

This README offers a concise description of both presentations, practical instructions, contributor information, and project context, making it ready for GitHub upload.[2][1]

[1](https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/attachments/18877943/2aeea68f-182e-4bcb-a15e-81c70b56767e/Project-presentation.pptx)
[2](https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/attachments/18877943/a4579e48-a037-44bf-9026-69d4d975d529/IoT-Hardware-Hacking.pptx)