How to Design Your Own Electronic Attack Device

Gao Shupeng (XiaoHuihui)
Senior Security Researcher, Baidu
Why make devices for attacking

**Authorized** attack testing (some Internal/External penetration testing projects)

Security research

Cool (hacker culture, spreading security knowledge)

Just for internal company security promotion and events
What will you learn

How to make a basic circuit board

Some common attack testing methods in the field of hardware security

I will come up with many new attack methods.

You can combine all this knowledge to try creating a circuit board that suits your own needs.
Interesting who enjoy crafting can give it a try

As security research, not useful:

- Waste of time
- Cannot be used for security research or making attack devices
Current

Circuit diagram

PCB

Finished product
Future

Circuitry might be software-defined

All functions to be implemented within powerful FPGAs

Only need to drag and drop the required CPUs and interface modules
We are going to learn

You need to:

● Develop a plan (what device to create, which main chips to use)
● Design a schematic diagram (how components connect) and PCB (Printed Circuit Board)

You no longer need to:

● Produce the PCB

You can:

● Component procurement and soldering by yourself.
● Or spend money, the factory will assist with component procurement and soldering.
Choose the circuit design software

Altium Designer ? KiCAD ?

LCEDA：free、online、full-service process、Most electronic components have footprint、open-source projects

Other software that you're more familiar with is fine too

It's just a tool

The most important：choosing which circuit design，new & effective ways
Goal: type-c 5V->3.3V, LED light

Solution: use AMS1117-3.3

Choose: selecting widely used components with high sales

Draw a schematic diagram
Draw a schematic diagram

- Refer to the official manual, online resources, open-source projects
- Use wires to correctly link the components.
- Adding "Nets"
- Wires with the same net name are connected together
Draw a schematic diagram

Adding LED, resistor, USB Type-C female connector

After connecting correctly, start to draw the PCB
• Add a Board Outline. The shape according to your needs

• Place components based on their functions

The blue lines represent the same network, indicating that they can be connected together
可以进行生产了
A case to solve security issues

As a researcher in IOT

We often need to dump firmware

Disassemble and assemble EMMC frequently

Repeated welding of EMMC is a very painful thing

Make an adapter?
It can unleash our imagination

Find new Chip solutions and channel attack methods.

To help us in security research and attack testing
Key Points & Research Directions

Creating tools to assist in security research: dump, sniffer

Monitoring and controlling channels for attack testing: WIFI, USB, HDMI

Miniaturization is essential

Capable of running a complete Linux system

Basic functionality: lithium battery power supply, 4G network, remote control, etc.
Creating a mini computer board

Used for:

Physical attacks, near-field attacks

● quite common in security services offered by various security companies.

● Research in Automotive Interface Security
  Compared to bulky laptops, it's much more convenient.

Attaching it to a drone

● Drones require small size and light weight.
We have a lot of experience in close-range attack testing.
Choosing a microcomputer

A full-fledged computer with complete Linux functionality, not like OpenWrt.

High performance, compact size,

Fewer unnecessary interfaces,

Best if it's a core module for easy expansion
Create a basic carrier board

Objective:

Expose the necessary interfaces, the validation version

The interfaces can be larger, miniaturization isn't a concern.

Refer to the CM4's official IO Board reference manual and CM4 module manual.

Interfaces: SD, USB, HDMI, power, LED, test pins
It's essential to connect GPIO_VREF, otherwise, CM4 won't function.

For SD card traces, aim for equal length adjustment.

Indicator lights can help us determine if everything is working properly.

For pins with uncertain functions, like Global and Boot, expose them in the test version.

Soldering connectors can be challenging, so it's better to have the factory handle the soldering.
Choose components, draw the schematic diagram, place components, connect them, product.

It seems work well

Just a validation version

Need to add features like 4G, USB, and charging.
More suitable for miniaturized attack tool

Add charging functionality (portable attack devices require power; power banks are too large).

Add 4G connectivity (for internet access and remote control).

Add USB ports (CM4 only has one USB port).
There are numerous charging circuits available. We just need:

- Charge the lithium battery
- Provide 5V power through boost functionality
- Have battery level indicators
- Handle high current and keep the circuit simple.

IP5306: 5V2A charging, 2.4A discharging, simple circuit.
Voltage step-down circuit

Digital circuits frequently require a 3.3V power.

Many options for voltage step-down circuits, DC-DC and LDO.

AMS1117? SY8088AAC?

For security researchers, it's important to opt for something simple and compact.

For instance, ME6210A33M3G (or similar models).

5V to 3.3V at up to 500mA, just need one capacitor.
Voltage step-down

If require multiple power supplies
Need a power management chip with high integration and simple circuit design
EA3036/EA3059
It support 3 to 4 channels of 5V step-down,
Adjust output voltages using resistors R1 and R2.

\[ V_{OUT1} = 0.6 \times \frac{R1}{R2} + 0.6 \text{ V} \]
About 4G module

Compact size is crucial.

Best to include serial and GPIO interface.

Different countries require supporting different frequency bands

Cat4 vs Cat1 200KB/s (so Cat1 is enough)

● Attack devices might not have human operators and may be delivered via delivery services.

● GPIO can enable remote power control of the core board to conserve energy or reboot to resolve system crashes.

● Serial ports allow remote shell operation to prevent connection loss due to various network issues.
About 4G module

Air780E:
Supports Asia, Europe, Africa, Australia, etc.
UART, GPIO, Lua scripting support, extensive Lua libraries.

Air700E:
Compact at only 10.5x13.5mm, TDD.

If faster CAT4 or global modules are needed:
EG25G, larger size and higher power consumption.
Air780E

Connect the UART of the 4G module to the UART of the Raspberry Pi.

Use GPIO to add control circuitry for managing the power and rebooting of the core module.
The USB Hub chip

Choose a compact and straightforward USB 2.0 hub.

Opt for the SL2.1s.

It only needs a crystal oscillator and a 5V power supply.
Used as an attack testing device
Why H616? Cheap

Still a bit large

The performance for WiFi-related attacks is average
The final version

To reduce the size:

- The Ethernet module is too large and often unnecessary for most tasks.
- To facilitate soldering, we'll switch to a double-sided layout, placing components on both sides of the board.
- Transition to using a 4-layer PCB design.

Add a high-performance WiFi card to enhance our capabilities for WiFi-related attacks.
Choosing a WiFi Card

How to conduct effective WiFi attacks:

- Support 2.4G / 5G monitor mode
- Sniffer manager / controller / data frame (not only manager)
- AP mode and support 802.1x hostap
- High High transmit/receive power (Preferably with an Amplifier)
- Better to be driver-free under a high version kernel
- USB Interface
- Modules available for soldering
### Recommended chips for WiFi attacks

<table>
<thead>
<tr>
<th>Chipset</th>
<th>Interface</th>
<th>Standard</th>
<th>Maximum Channel Width</th>
<th>Linux In-Kernel Driver</th>
<th>AP Mode</th>
<th>Monitor Mode</th>
<th>Recommended For Linux</th>
</tr>
</thead>
<tbody>
<tr>
<td>MT7921AU</td>
<td>USB3</td>
<td>WiFi 6E</td>
<td>160</td>
<td>✓ 5.16+</td>
<td>✓</td>
<td>✓</td>
<td>Yes</td>
</tr>
<tr>
<td>Mediatek MT7922au</td>
<td>USB3</td>
<td>WiFi 6E</td>
<td>160</td>
<td>✓ 5.16+</td>
<td>✓</td>
<td>✓</td>
<td>[4]</td>
</tr>
<tr>
<td>Realtek RTL8852cu</td>
<td>USB?</td>
<td>WiFi 6E</td>
<td>160</td>
<td>✓ [6]</td>
<td>✓</td>
<td>✓</td>
<td>No</td>
</tr>
<tr>
<td>Realtek RTL8832cu</td>
<td>USB3</td>
<td>WiFi 6E</td>
<td>160</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>No</td>
</tr>
<tr>
<td>RTL8812AU</td>
<td>USB3</td>
<td>WiFi 6E</td>
<td>80</td>
<td>✓ 5.18+</td>
<td>✓</td>
<td>✓</td>
<td>No</td>
</tr>
<tr>
<td>Mediatek MT7921au</td>
<td>USB3</td>
<td>WiFi 6E</td>
<td>80</td>
<td>✓ 5.18+</td>
<td>✓</td>
<td>✓</td>
<td>Yes</td>
</tr>
<tr>
<td>Realtek RTL8852bu</td>
<td>USB?</td>
<td>WiFi 6</td>
<td>80</td>
<td>✓ - avoid [2]</td>
<td>✓</td>
<td>✓</td>
<td>bad driver</td>
</tr>
<tr>
<td>Realtek RTL8832bu</td>
<td>USB3</td>
<td>WiFi 6</td>
<td>80</td>
<td>✓ - avoid [2]</td>
<td>✓</td>
<td>✓</td>
<td>bad driver</td>
</tr>
<tr>
<td>Realtek RTL8814au</td>
<td>USB3</td>
<td>WiFi 6</td>
<td>80</td>
<td>✓ - avoid old driver</td>
<td>✓</td>
<td>✓</td>
<td>bad driver</td>
</tr>
<tr>
<td>Mediatek MT7662u</td>
<td>USB2</td>
<td>WiFi 5</td>
<td>80</td>
<td>✓ 5.9+ [6]</td>
<td>✓</td>
<td>✓</td>
<td>No</td>
</tr>
<tr>
<td>Mediatek MT7612u</td>
<td>USB3</td>
<td>WiFi 5</td>
<td>80</td>
<td>✓ 4.19+</td>
<td>✓</td>
<td>✓</td>
<td>Yes</td>
</tr>
<tr>
<td>Realtek RTL8812bu</td>
<td>USB3</td>
<td>WiFi 5</td>
<td>80</td>
<td>✓ 6.2+ [3]</td>
<td>✓</td>
<td>✓</td>
<td>Yes</td>
</tr>
<tr>
<td>Realtek RTL8812cu</td>
<td>USB3</td>
<td>WiFi 5</td>
<td>80</td>
<td>✓ 6.2+ [3]</td>
<td>✓</td>
<td>✓</td>
<td>No</td>
</tr>
<tr>
<td>Realtek RTL8812au</td>
<td>USB3</td>
<td>WiFi 5</td>
<td>80</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>No</td>
</tr>
<tr>
<td>Mediatek MT7610u</td>
<td>USB2</td>
<td>WiFi 5</td>
<td>80</td>
<td>✓ 4.19+</td>
<td>✓</td>
<td>✓</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**MT7921AU**
Good performance with minimal packet loss.

**MT7612U**
Low kernel requirements.

**RTL8812AU**
Complex driver compilation.

https://github.com/morrownr/USB-WiFi/blob/main/home/USB_WiFi_Chipsets.md
**WIFI attack methods**

**Access Only:**

Simply accessing the target network. Scan, ARP spoofing can often be effective, especially when there's no attack surface on the web side.

**Traditional Methods:**

Deauth attack and capturing handshake packets

**Innovative Approaches:**

- 802.1x PEAP hash cracking: Enterprises with higher security requirements often utilize 802.1x WiFi authentication.
- Collecting MAC addresses, MAC to IPv6 address, direct attacks through IPv6.
- Monitor mode: Decrypting 802.11 frames online, pure monitoring mode, operating silently without detection.
WIFI card module

17.8x27x2.5mm

Need to draw the footprint yourself.

Higher transmission power

Improved performance
About draw PCB

- Determine Component Placement Based on Functionality
- Place Larger Components First
- Connect All Components Together
If there's no space for circuit wiring:

- Avoid areas with existing connections.
- Use via stitching to route signals through holes.
- Increase the number of PCB layers.
● Powerful sniffer WiFi card
● Remote control and power on/off via 4G
● Battery-powered with charging
● HDMI output
● Multiple USB ports
● Compatible with Raspberry Pi CM4 and other cheaper Pis
● Only 40x55 mm
How to create a Linux system circuit board

- CM4 modules are expensive and need connectors to connect to carrier boards.
- Their shape cannot be customized easily (for example, to make them smaller).
- CPU is a chip itself, we can design the circuit board based on the manual and recommended circuits.

Which manufacturer's CPU to choose?

**Allwinner** offers some open-source projects for their CPUs.
We can also refer to online resources like Orange Pi schematics for guidance.
Choose the processor

Getting Started:
Allwinner F1C100s/F1C200s/V3s/V3x
(Internal RAM, Simple Circuit)

High Performance:
Allwinner H2/3/5 H6/H616
About F1C200s

CPU: Less than $2, includes 64MB RAM

Small size

Moderate performance, about 1/3 of Raspberry Pi Zero

(V3s is recommended, slightly larger but has similar performance to Pi Zero)

First, we'll create a schematic based on online resources and the CPU manual.
Place the major components

Place others

Connect
Hardware

- Just 30x30mm in size, costing less than $4.
- Capable of running a complete Linux operating system.
- Supports booting from SPI Flash or TF card.
- Utilizes a 2-layer circuit board, keeping production costs low.
- All components are placed on the front side for easy soldering.
Software

With the hardware ready, you'll also need corresponding software:

- Compile the bootloader
- Compile the kernel
- Modify the device tree
- Create the root file system (rootfs)
- Flash the image

We've produced 10 of these for everyone, with the HITB logo printed on the back as a commemoration.

If anyone is interested, feel free to reach out to me after the session to obtain one.
Real attacks require running abundant security software, demanding High CPU Performance

**Scan & vul tools :**
- Tcpdump, tshark, nmap, masscan, sqlmap, Hydra

**WIFI & sniffer**
- bttercap, aircrack-ng, hostapd-wpe, kismet

**Other tools :**
- Frp, nps, Metasploit, hashcat, john, ......
Allwinner H616

H616 > H2/3/5

- CPU : Quad-Core ARM Cortex-A53
- USB : USB2.0: 1x OTG + 3x Host
- Performance : one-third of Raspberry Pi 4

Open-source projects on oshwhub for reference

Added memory traces, requires a 6-layer board.

High memory frequency, need equal-length traces.

6-layer PCBs are expensive.

https://github.com/YuzukiHD/YuzukiChameleon
Test

Validation:

● The core board works properly
● Full Debian operating system is functional
● High-performance WiFi adapter can be connected for security testing

Replicate previous work

So we can add 4G module and high-performance WiFi adapter
About the Software

- Compile the complete Armbian system (taking reference from the H616 CB1 module design, you can also use CB1's compilation configuration).

- Write the compiled image to the SD card. Kernel version: 6.2.16.

- Alternatively, directly use the image from the H616 CB1 module.
Security Testing and Research on Other Interfaces

- USB (network, HID, protocol analyze)
- WIFI
- 4G/LTE
- Ethernet
- HDMI
- Automotive Related (CAN, Ethernet, LVDS, GSML)
- SDR
- FPGA
CREATING A USB ANALYSIS DEVICE

USB protocol analyzers are often quite expensive (due to USB 2.0 speeds and FPGA usage).

How to create a simple analyzer for:

● Protocol analysis
● Keyboard logging

Are there simpler solutions?
The RP2040 chip:

- Priced at just $0.5.
- Dual ARM Cortex-M0+ Microcontroller
- The 8 PIOs (programmed input–output)
- It can achieve the same frequency as the CPU and doesn't consume CPU resources.
- Commonly employed for oscilloscopes, signal generators

Based on open-source projects like pico_usb_sniffer, USB2.0 Low / Full speed sniffing.
RP2040 Schematic

- Need to be very small in size, which is why RP2040 solution was chosen
- Reference circuit diagram: hardware-design-with-rp2040.pdf
- Remove unnecessary pins
- The goal is to fit inside a USB connector
How to Become Smaller

- Removing Unused Interfaces
- Retaining Only USB D+ D- Pins
- Using 4-Layer PCB
- Utilizing Small Footprint Components, such as 0402
Function Verification

Keyboard Receiver
Or other USB devices

USB D+ D-
Sniffer

PIO 0
PIO 1

Host

RP2040 Chip
Become a Keyboard Logger

- Successful Debugging, Keeping Only the Chip, Removing Debug USB Pins
- Only 14mm x 9mm
- You Can Place It Anywhere
- For Example, Inside a Keyboard Receiver
Tiny Chip with WiFi

The ESP32-S3:

- Added native USB compared to ESP32-C3
- Similar to RP2040 but without PIO pins, adds WiFi and Bluetooth
- Used for handshake packet detection and deauth attacks before

Getting smaller! (We've been making devices smaller)

Any new methods?

- Remote control badusb!
- Network MITM!
Become Smaller

- Refer to the official schematic on the right
- For a smaller size, chips instead of modules are needed
- Similar to RP2040, use small-package components
- Utilize built-in storage! Though it's a bit more expensive
- Hardware design is done, how to turn it into an attack testing device
Remote Controllability HID Device

- ESP32-S3 can be flashed with CircuitPython firmware, supporting HID device emulation
- A BadUSB device with remote control capability can be created
- Inserted into a data cable, similar to the diagram on the right
- USB HUB chip needs to be added, such as SL2.1s, for normal data cable function and enhanced concealment

```python
# Circuit Playground HID Keyboard
import board
import usb_hid
from adafruit_hid.keyboard import Keyboard
from adafruit_hid.keyboard_layout_us import KeyboardLayoutUS
```
USB Cable for Traffic Monitoring

- HID device emulation achieved via TinyUSB
- What more can TinyUSB do? Network emulation!
- We can employ the ESP-IDF Framework with native TinyUSB support
- Add relevant TinyUSB functionality in compilation options
- Activate Network Bridge feature and configure Auto-Connect WiFi
- Note: Only configurations matching the diagram below ensure optimal compatibility
- We've obtained a data cable that can capture network traffic!
- Compatible with Android, MacOS, Windows, Linux
- According to the operating system's routing rules, wired network takes priority
- Very responsive – network card works just 2 seconds after insertion

Attacker-controlled WiFi
Http Cookie、DNS......

Recognized as an Ethernet adapter

Maybe a charging cable

10x17 mm

Automatically connect
Anything Smaller?

- Remember the Air700E 4G module we mentioned before?
- The circuit is simple, the PCB is just 15x12mm
- It also supports USB RNDIS device emulation
- Plugged into a computer, it generates a USB Ethernet network card
- The 4G signal coverage is much wider

How to control traffic?

- Use SRSLTE to generate a 4G base station
- Sign up for a private APN service with an operator, then all 4G traffic will flow through your server
Capturing HDMI Data

- Previously used FPGA solutions, which were large and expensive
- Now there are better options, like MS2109, MS2130/2131
- Single Chip + Flash, USB 3.0 Support
- Designing a Very Compact HDMI to USB Device
- Higher Power Requirement, we can using EA3036/3059
Capturing Ethernet Data

Regarding Ethernet capture, it has been around for quite a while, like the Throwing Star LAN Tap.

However, it's necessary to capture both directions simultaneously, and the speed is low, unable to send data packets.

A dual Ethernet card can be used for bridging to achieve Ethernet monitoring.

The RTL8152 chip can be utilized, small in size, and driver-free.

Using analog switch chips like CH440 to control the communication direction of Ethernet.
Automotive Security

Opensource Project : USB CAN

Chip: STM32F072+TJA1051T/3

Can be integrated into the previous attack testing circuit board

Convenient for remote CAN signal analysis and debugging
Automotive Security

- Video transmission in vehicles does not use HDMI or DP
- Utilizes GMSL and FPDLink High-Speed signals
- Often, we face difficulties procuring screens for analyzing Vehicle IVI Systems (Expensive)
- Our Automotive Security Simulation team wishes to simulate camera data
- So we can create relevant video encoding and decoding circuit boards to assist in our research

Some OpenSource Project

fpdlink2raspi PCB
Applications in Other Security Work

Our company recently had a cybersecurity awareness month event. We created an ESP32+display device named "Boss is Coming" for this occasion.

Using WiFi sniffing, we measured the signal strength of the boss's MAC address. We distributed these devices for soldering activities, aiming to enhance interest in security among everyone.

To demonstrate the insecurity of the 802.1x encryption and the resulting issues of password leakage and cracking, we created a "Sheep Wall" device that performs hash sniffing and cracking based on 802.1x.
Summary
Sharing examples of using circuit board production for security research and testing

Key points: simple solutions, miniaturization, new attack approaches

Everyone, please refrain from using them for malicious purposes.

I started delving into this field just three months ago, so it's not too difficult and doesn't require much time

No Q&A, if any question, mail to gaoshupeng@baidu.com
THANK YOU!