Raspberry Fields Forever

Chris Johnson

Project Overview

- Deliverable:
  - Intentionally vulnerable Raspberry Pi image
- Capabilities:
  - Penetration testing training platform
  - Honeypot [stretch goal]
- Report:
  - Previous Works
  - Case Study: Developing the Raspberry Fields Forever Intentionally Vulnerable Platform
  - Penetration Testing the Raspberry Fields Forever Intentionally Vulnerable Platform
  - Case Study: Developing the Raspberry Fields Forever Honeypot [stretch goal]
  - Analysis of data collected by the Raspberry Fields Forever Honeypot [stretch goal]

Previous Work

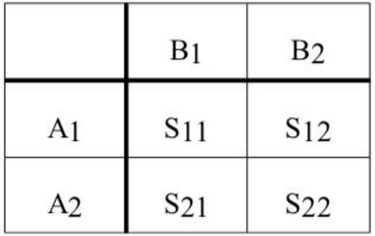
Research

- Is Attack Better Than Defense? Teaching Information Security the Right Way
- Evaluation of the Offensive Approach in Information Security Education
- Simulation Approaches in Information Security Education
- Shell We Play A Game? CTF-as-a-service for Security Education
- Guide for Designing Cyber Security Exercises
- Build Your Own Security Lab: A Field Guide for Network Testing
- Hacking Raspberry Pi
- A Simple Laboratory Environment for Real-World Offensive Security Education
- Penetration Testing Professional Ethics: A Conceptual Model & Taxonomy
- Professional Penetration Testing: Creating & Learning in a Hacking Lab (book)

Is Attack Better Than Defense? Teaching Information Security the Right Way

- “Our claim is that teaching offensive methods yields better security professionals than teaching defensive techniques alone.”
- Topics:
  - Concepts from psychology
  - Pedagogical sciences
  - Experimental method

Table 1. Examination scheme of a two-factorial design with independent variables A and B



	B_1	B_2
A_1	S_{11}	S_{12}
A_2	S_{21}	S_{22}

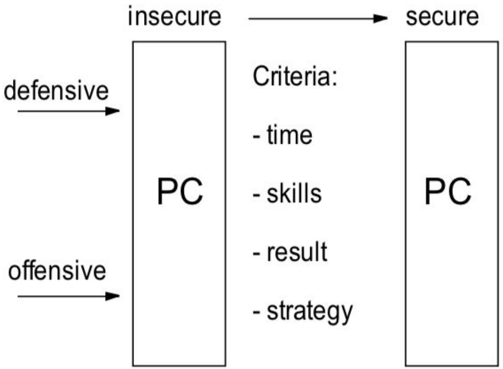


Figure 2: Experimental setup

Related Works

- Mark’s Pentest Challenge
  - “Small penetration testing challenge I set up on my Raspberry Pi for my classmates.”
  - Small variety of intentionally vulnerable web applications (DVWA, Mutillidae, WackoPicko, etc.)
- DV-PI
  - “The touch friendly ‘driving range’ for IoT penetration testing with your Kali-Pi.”
  - Moderate variety of intentionally vulnerable web applications
  - Moderate variety of intentionally vulnerable services
- RasPwn OS
  - “RasPwn was designed as a training tool and exists only to be attacked and pwned. Everything from the OS itself to the daemons and services to the web applications installed are all vulnerable to some degree.”
  - Large variety of vulnerable services (Bind, Samba, Apache2, Nginx, etc.)
  - Large variety of intentionally vulnerable web applications (DVWA, Mutillidae, WackoPicko, WebGoat, etc.)

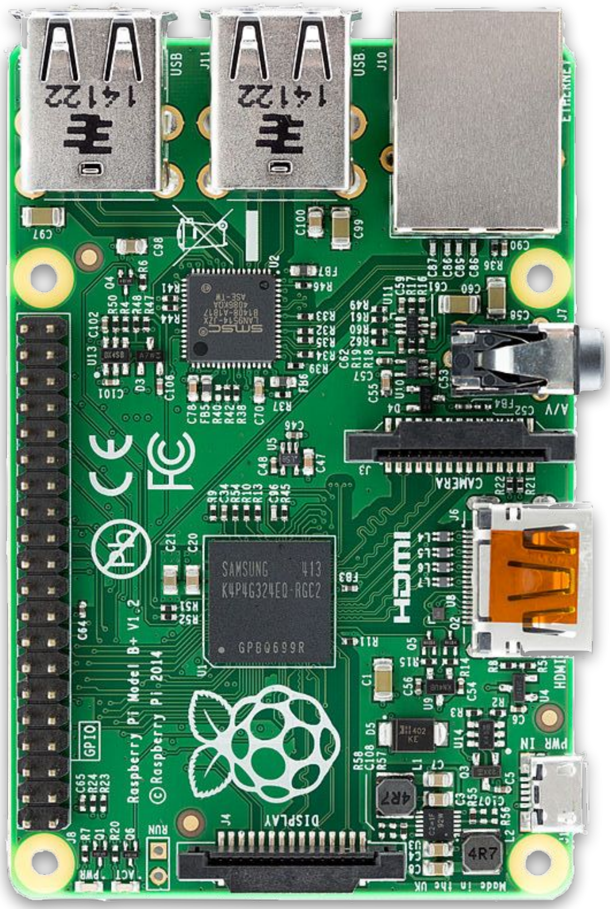
Common Thread between Related Works

- Poorly documented build process
  - Typically not described at all
- Low visibility
  - Severe lack of web presence compared to other intentionally vulnerable images
- Lack of walkthroughs
  - Higher barrier to entry
- Don’t take full advantage of Pi platform
  - “It wasn’t really designed for Pi, as it could run on any linux box. Today I might deploy it on an AWS EC2 simply for better connection and availability.” - Mark Szabó-Simon

Developing the Raspberry Fields Forever

Why Pi?

- ARM
  - Low power chips
  - Same CPU vulnerabilities as IOT devices
- Portable
  - Pentest from anywhere
  - Plug-and-play honeypot [stretch goal]
- Easily isolated
  - Ad-Hoc Wi-Fi
- Inexpensive
  - $30
  - Low barrier to entry

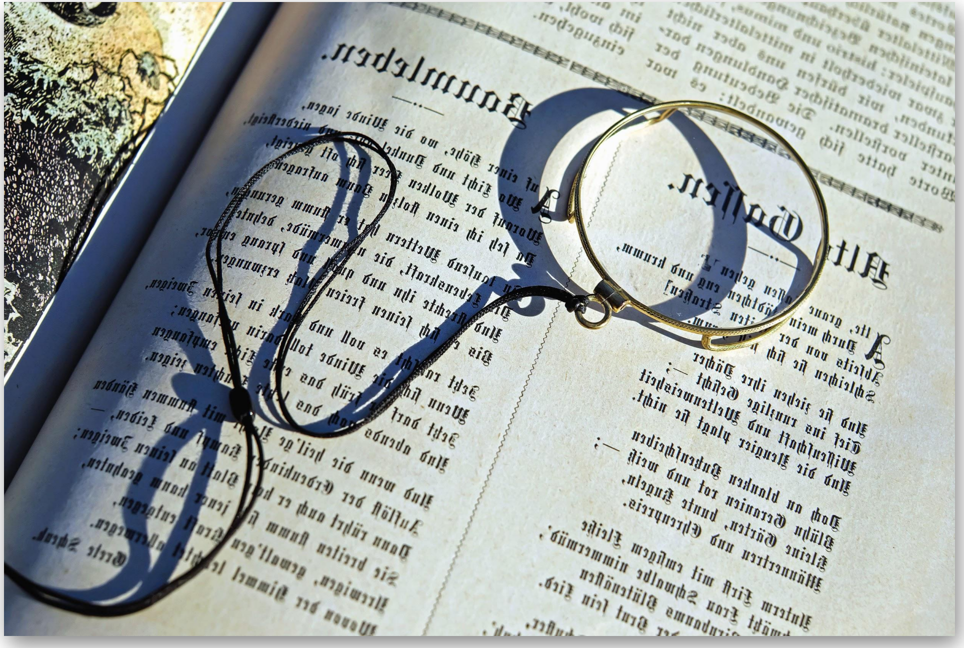


Plans for build

- Vulnerable services
  - Compatible Linux services from Exploit-DB
  - Custom service, written from scratch
    - [stretch goal]
- ARM Vulnerabilities
  - Available on Exploit-DB
- Application Vulnerabilities
  - Third Party intentionally vulnerable web application platforms
  - Custom web application, written from scratch
    - [stretch goal]

Case Study

- Lessons learned from reviewing other projects
- Step-by-step process of developing the Raspberry Fields Forever
- Lessons Learned
  - Unexpected challenges
  - Unexpected insights gained
- Walkthrough [stretch goal]



Penetration Testing

- Recon results
  - Vulnerability Scans
  - Nmap, Nikto, Vulscan, OpenVas, etc
- Automatic Exploit Testing
  - Armitage, etc
- Difficulty rating compared to non-Pi penetration testing images
  - Metasploitable 2, Metasploitable 3, Vulnhub VM's, etc.

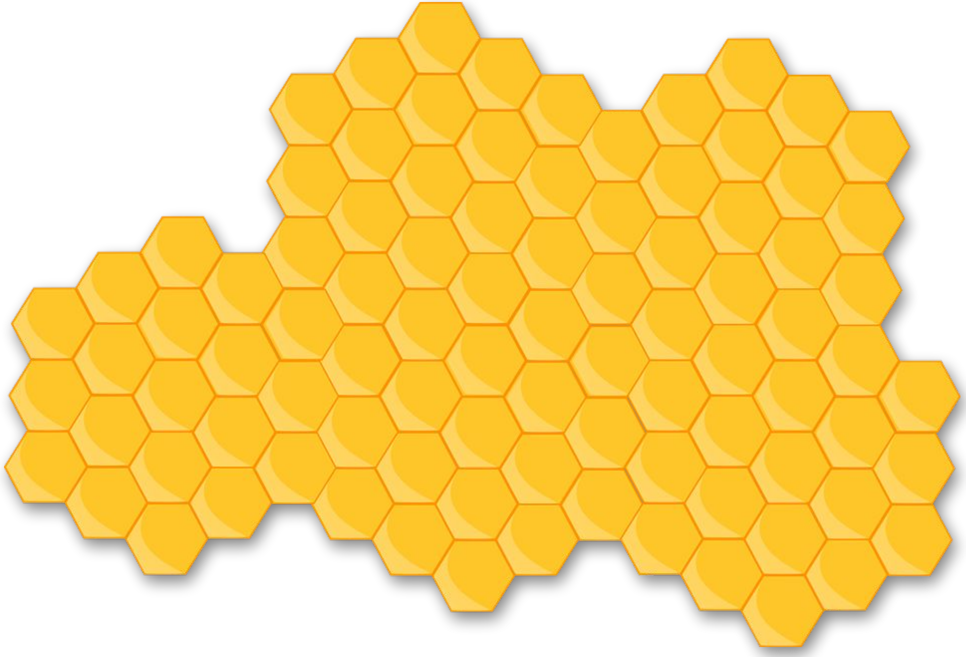
Honeypot

stretch goal

Research & Related Works

- Papers:
  - Intrusion Detection System Using Raspberry PI Honeypot in Network Security
  - Constructing Cost-Effective and Targetable Industrial Control System Honeypots for Production Networks
- Projects:
  - Cowrie
  - T-Pot

Case Study



- Technical configuration
- Build process
- Challenges

The Road Ahead

Anticipated Challenges

- Main Project:
  - Pi Compatibility
  - Pi Platform Resources
- Stretch Goals:
  - Developing vulnerable services
  - Developing vulnerable web apps
  - Writing an unbiased walkthrough
  - Complementing a penetration testing platform with honeypot functionality

Thank You