hackersvilla.xyz

# Firmware Obfuscation

Presenter – Sanchay Singh
@ Digital Hash Seminar, Microsoft Office
16 November 2024

# $ whoami_

hackersvilla.xyz

-> ***Founder of HackersVilla CyberSecurity***

-> *Security Consultant/Trainer at MakeIntern*

-> *Working as SME with Upgrad & SunStone*

-> *Trained Employees of KPMG, Cognizant, etc*

-> *Security Mentor at OWASP Delhi & BSides Noida*

-> *Speaker at BSides, Defcon Delhi, CRACCon, etc*

-> *Active part of NULL , CRAC, THM Delhi Chapter*


***Sanchay Singh***

*CYBERSECURITY EXPERT | CORPORATE TRAINER | PUBLIC SPEAKER*

*sanchayofficial*

*sanchayofficial@gmail.com*

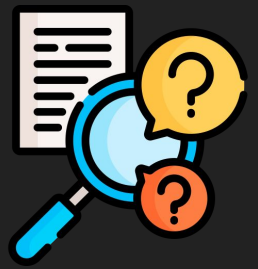

# My Journey


# Welcome to the Firmware Obfuscation Session

hackersvilla.xyz

# Prerequisites for Participants

- Basic to Intermediate level of Cybersecurity Knowledge
- A working laptop/system (or simply try this at your home)
- Curiosity and Enthusiasm




# Introduction


# What is Firmware?

- Embedded software that controls hardware components.
- Found in IoT devices, routers, printers, industrial control systems, etc.

hackersvilla.xyz

# Why is Firmware a Target in Cybersecurity?

- Stores critical instructions.
- Vulnerable to reverse engineering, tampering, and exploitation.


# What is Firmware Obfuscation?

- Technique to make firmware analysis harder by hiding or complicating its logic.
- Protects intellectual property and prevents exploitation.


# Importance of Firmware Obfuscation


# Key Objectives

- **Prevent Reverse Engineering:** Increases the difficulty for attackers.
- **Protect Intellectual Property:** Ensures proprietary code isn't stolen.
- **Enhance Device Security:** Safeguards against tampering and malware injection.

**Industries Benefiting from Firmware Obfuscation:**

Automotive, Healthcare, IoT, Industrial Systems.


# Techniques in Firmware Obfuscation


# 1. Control Flow Obfuscation

- Modifies the program's flow to mislead reverse engineers.
- Example: Using opaque predicates, dynamic function calls.


# 2. Data Obfuscation

- Encrypts or scrambles critical data.
- Example: Key storage mechanisms, configuration data encryption.


# 3. Code Encryption

- Encrypts entire sections of the firmware.
- Requires decryption at runtime, making static analysis difficult.


# 4. Dynamic Obfuscation

- Generates unique firmware for each device.
- Prevents mass exploitation of vulnerabilities.


# Threats and Challenges


# Common Attacks Against Firmware

- **Binary Reversing:** Using tools like Ghidra, IDA Pro.
- **Firmware Dumping:** Extracting firmware from devices via JTAG, UART, or SPI interfaces.
- **Injection of Malicious Code:** Injecting malware to alter device behavior.


# Challenges in Obfuscation

- **Performance Overhead:** Obfuscation can slow down device performance.
- **Compatibility Issues:** Obfuscated firmware might not work seamlessly across all devices.
- **Skilled Attackers:** Advanced attackers with sufficient resources can still deobfuscate firmware.


# Let's learn with a small Demo


# Best Practices for Secure Firmware Development


# Challenges in Obfuscation

- **Performance Overhead:** Obfuscation can slow down device performance.
- **Compatibility Issues:** Obfuscated firmware might not work seamlessly across all devices.
- **Skilled Attackers:** Advanced attackers with sufficient resources can still deobfuscate firmware.

# Best Practices

- **Implement Layered Security:** Combine obfuscation with secure boot and runtime integrity checks.
- **Frequent Updates:** Patch vulnerabilities regularly to stay ahead of attackers.
- **Threat Modeling:** Identify potential attack vectors during development.


# So....
# Questions?

hackersvilla.xyz

# Thank You

**PPT Available on**

*https://github.com/sanchayofficial/meetups-ppt*