attify

Flare-On 6 CTF WriteUp (Part 8)

16.OCT.2019 . 4 MIN READ

This is the eighth part of the Flare-On 6 CTF WriteUp Series.

8 - snake

The challenge reads

> *The Flare team is attempting to pivot to full-time twitch streaming video games instead of reverse engineering computer software all day. We wrote our own classic NES game to stream content that nobody else has seen and watch those subscribers flow in. It turned out to be too hard for us to beat so we gave up. See if you can beat it and capture the internet points that we failed to collect.*

Different from others, challenge 8 deals with reversing a NES Rom named *snake.nes*. We will be using the Mesen emulator for running the Rom. Among other features, Mesen

Barun

Reverse Engineer with an interest in low level stuff and anything about security.

RECENT POSTS

1.

Flare-On 6 CTF WriteUp (Part 12)

20.OCT.2019

will be using the Mesen emulator for running the Rom. Among other features, Mesen supports debugging the assembly code which is integral for our purpose. The processor on the NES runs 6502 assembly. Without further ado, let's give the game a try.

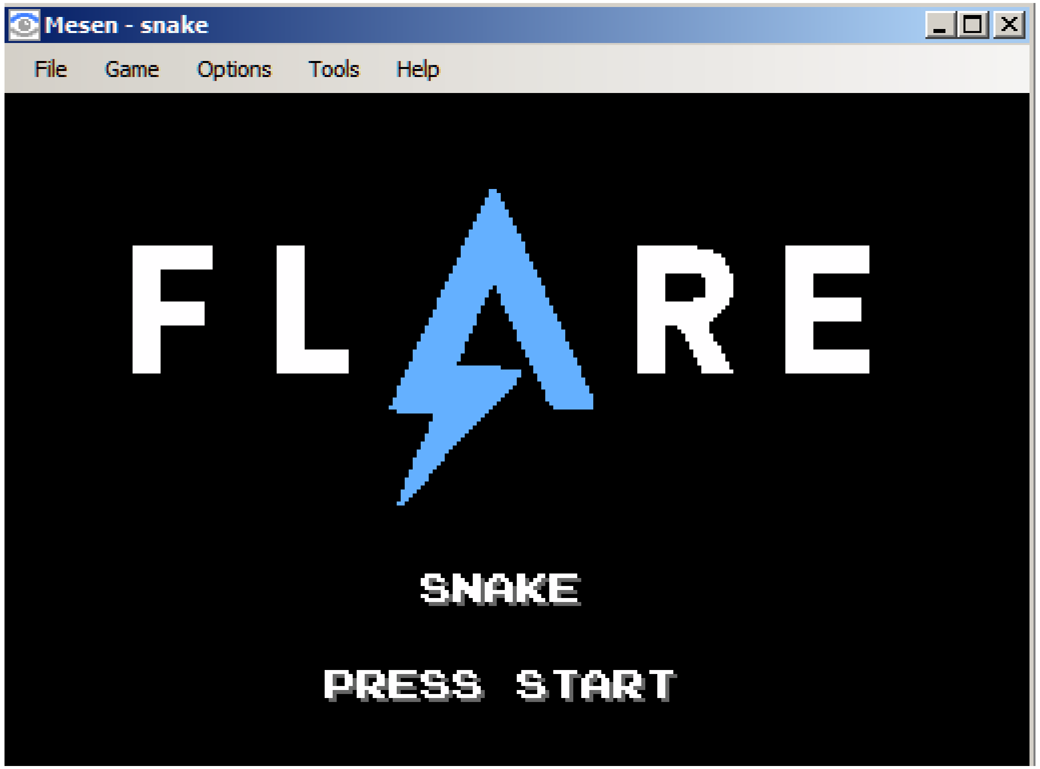


Figure 1: The starting window

2.

Flare-On 6 CTF WriteUp (Part 11)

19.OCT.2019

3.

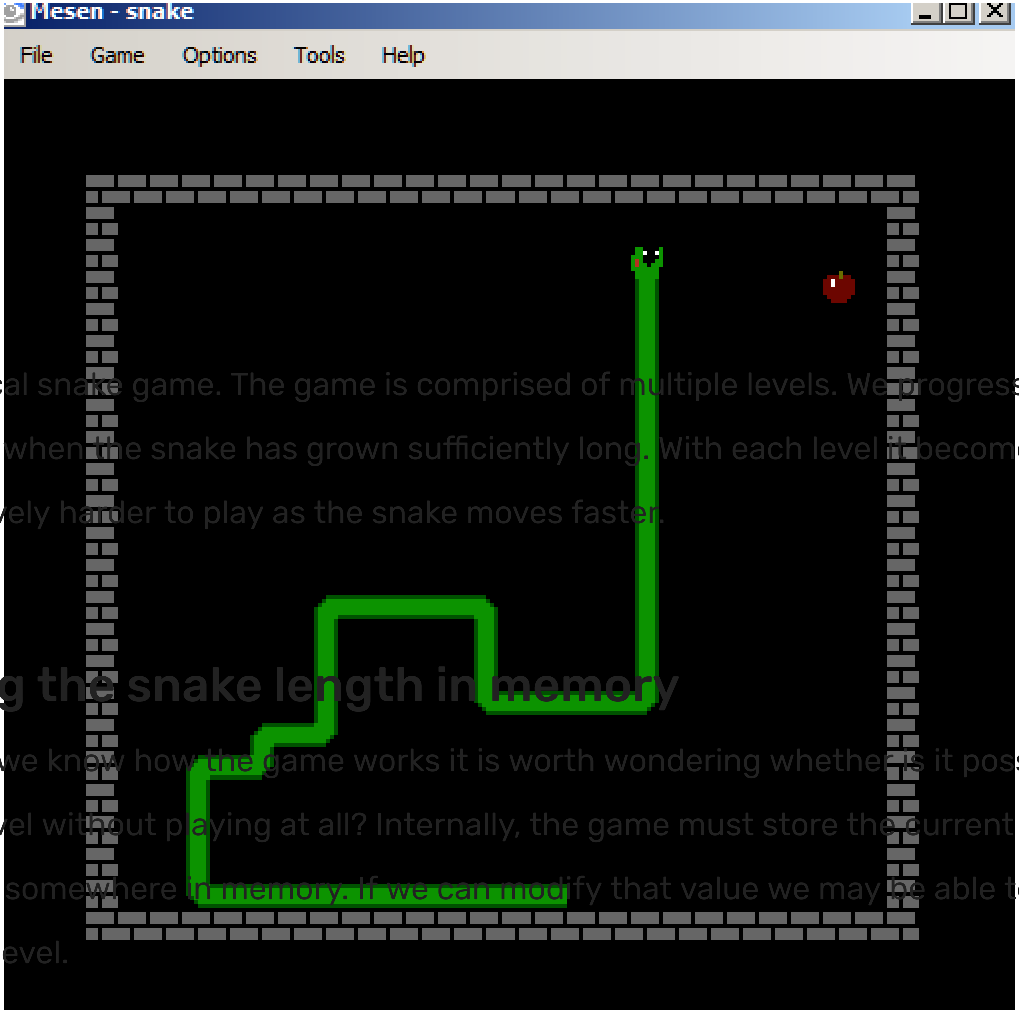
Flare-On 6 CTF WriteUp (Part 10)

18.OCT.2019

4.

Flare-On 6 CTF WriteUp (Part 9)

17.OCT.2019



It's a typical snake game. The game is comprised of multiple levels. We progress to the next level when the snake has grown sufficiently long. With each level it becomes progressively harder to play as the snake moves faster.

Finding the snake length in memory

Now that we know how the game works it is worth wondering whether is it possible to finish a level without playing at all? Internally, the game must store the current length of the snake somewhere in memory. If we can modify that value we may be able to bypass playing a level.

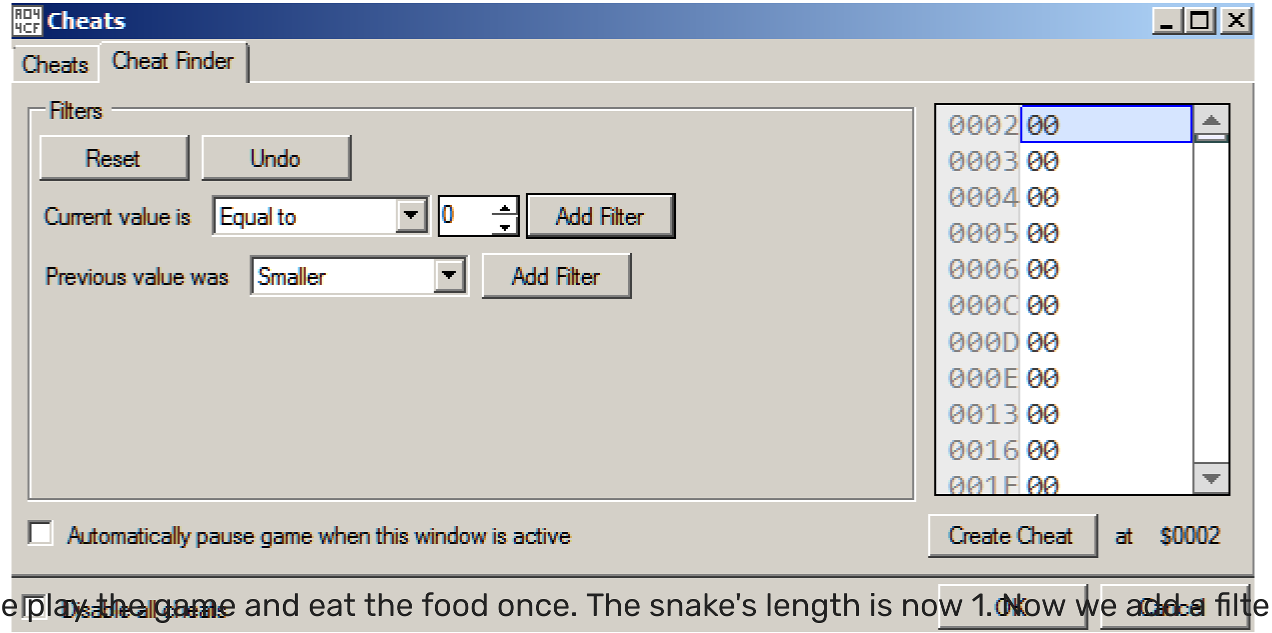
5.

Flare-On 6 CTF WriteUp (Part 7)

15.OCT.2019

Figure 2: A typical snake game

Mesen offers cheating functionality like the venerable Cheat Engine. Start a new Game and pause it immediately. Now go to Tools -> Cheats. Initially, the length of the snake is 0. In the Cheat Finder tab, we add a filter for Current value is Equal to 0.



Cheats

Cheats | Cheat Finder

Filters

Reset | Undo

Current value is: Equal to, 0 — Add Filter

Previous value was: Smaller — Add Filter

```
0002 00
0003 00
0004 00
0005 00
0006 00
000C 00
000D 00
000E 00
0013 00
0016 00
001F 00
```

☐ Automatically pause game when this window is active

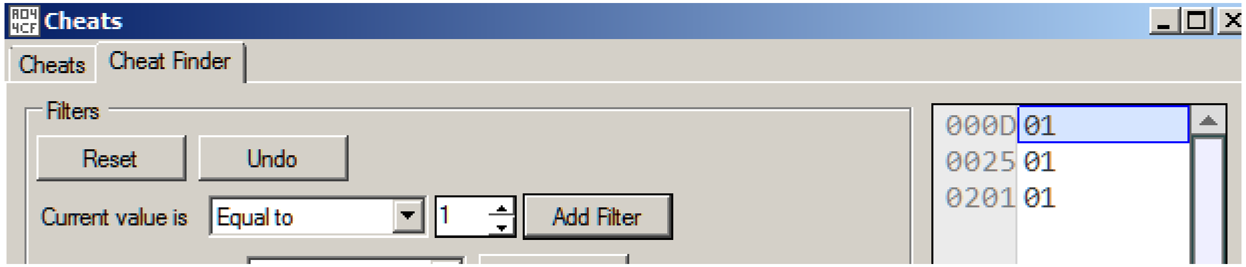
Create Cheat at $0002

☐ Disable all cheats

OK | Cancel

Figure 3: Using Cheat Finder

We play the game and eat the food once. The snake's length is now 1. Now we add a filter for current value is equal to 1. We get three possible memory locations.



Cheats

Cheats | Cheat Finder

Filters

Reset | Undo

Current value is: Equal to, 1 — Add Filter

```
000D 01
0025 01
0201 01
```

TAGS

- analog modulation
- Android
- android application security
- android hands on security and exploitation training
- android security
- Apktool
- application auditing
- application security auditing
- appsec usa
- appwatch
- arduino nano
- arm
- ARM binaries
- ARM course
- ARM exploitation book

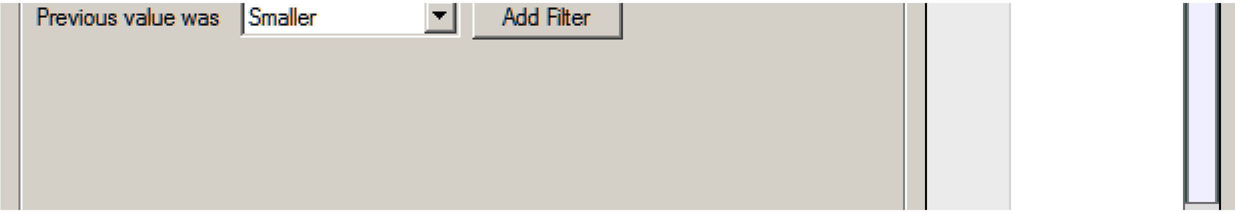
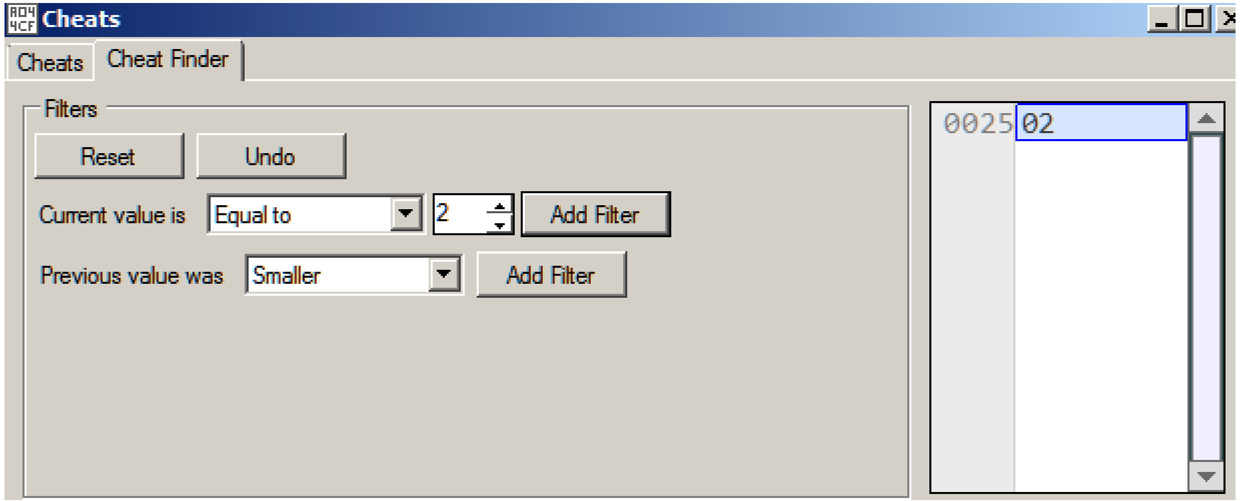

Previous value was Smaller

Figure 4: Three possible locations

Continuing in the same way, we just get a single hit when the snake's length is 2.



Current value is Equal to 2

Previous value was Smaller

```
0025 02
```

Figure 5: Snake length is stored at address 0x25

Thus 0x25 is the address of the memory where the length of the snake is stored. Now we need to locate the code that writes to this address. This can be done in Mesen by setting a Write Breakpoint. Open the Memory viewer in Debug view and navigate to address 0x25 where the snake length is stored.

ARM exploitation book

ARM exploitation video training

ARM gadgets

ARM Training attify

attify badge

attify training

best security practices

biggest iot attacks of all time

binwalk

blackberry pentesting

blackhat ble

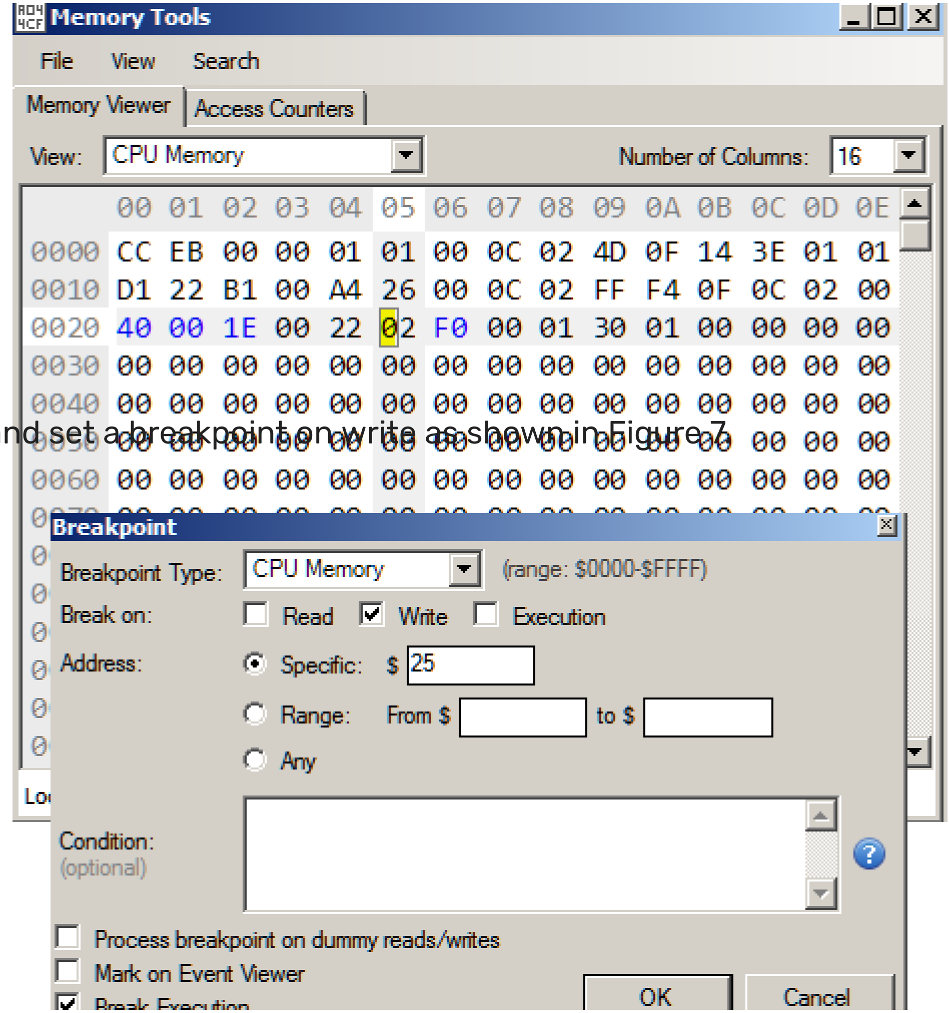
BLE attacks

BLE dangers

BLE hacking and exploitation

BLE security issues

BLE sniffing



View: CPU Memory Number of Columns: 16

```
      00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E
0000  CC EB 00 00 01 01 00 0C 02 4D 0F 14 3E 01 01
0010  D1 22 B1 00 A4 26 00 0C 02 FF F4 0F 0C 02 00
0020  40 00 1E 00 22 02 F0 00 01 30 01 00 00 00 00
0030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0040  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0050  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Breakpoint Type: CPU Memory (range: $0000-$FFFF)

Break on: □ Read ☑ Write □ Execution

Address: ● Specific: $ 25

○ Range: From $ to $

○ Any

Condition: (optional)

□ Process breakpoint on dummy reads/writes

□ Mark on Event Viewer

☑ Break Execution

Right click and set a breakpoint on write as shown in Figure 7.

BLE vulnerabilities

bleah

bluetooth technology

box

brut Exception

BtleJuice

capture radio traffic

career in cybersecurity

CCTV cameras

challenges in iot retail

chroot

cloud based mobile application security scanner

consulting

CTF

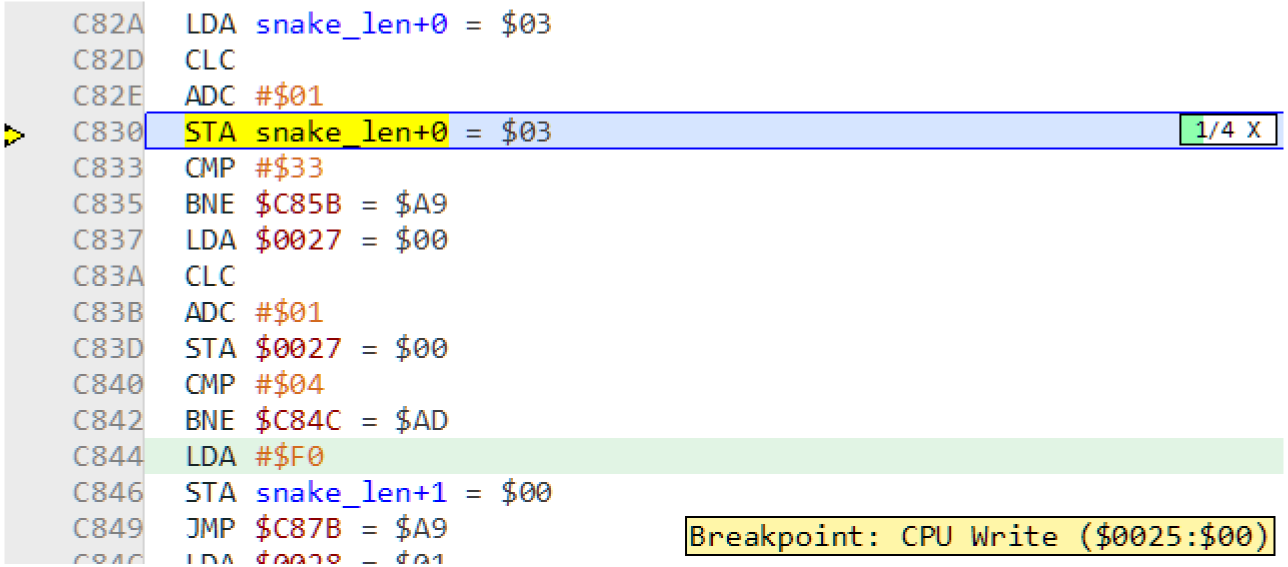
cyber attacks

cybersecurity

Damn Vulnerable iOS App

Figure 7: Setting a memory write breakpoint

We continue playing the game and just after the snake eats the food the breakpoint triggers.



```
 C82A  LDA snake_len+0 = $03
 C82D  CLC
 C82E  ADC #$01
>C830  STA snake_len+0 = $03                                    1/4 X
 C833  CMP #$33
 C835  BNE $C85B = $A9
 C837  LDA $0027 = $00
 C83A  CLC
 C83B  ADC #$01
 C83D  STA $0027 = $00
 C840  CMP #$04
 C842  BNE $C84C = $AD
 C844  LDA #$F0
 C846  STA snake_len+1 = $00
 C849  JMP $C87B = $A9             Breakpoint: CPU Write ($0025:$00)
 C84C  LDA $0028 = $01
```

Figure 8: The breakpoint hits

The code at C830 tried to write to the address at 0x25 which triggered the breakpoint. After incrementing the length it goes on to check if it equals 0x33. If not it jumps to C85B. Thus our snake has to be 0x33 units long in order to progress to the next level. We can set the memory to 0x33 to cheat our way to the next level, but there is an even easier way.

Recall, that the game is comprised of multiple levels. The code from C837 to C840

Damn Vulnerable iOS App

dangers of iot

DDoS attacks

devops

digital modulation

dumping memory

embedded hacking

expert

Exploit ARM devices

exploitation

exploiting ble

exploiting smart devices

firmadyne

firmware analysis toolkit

firmware emulation

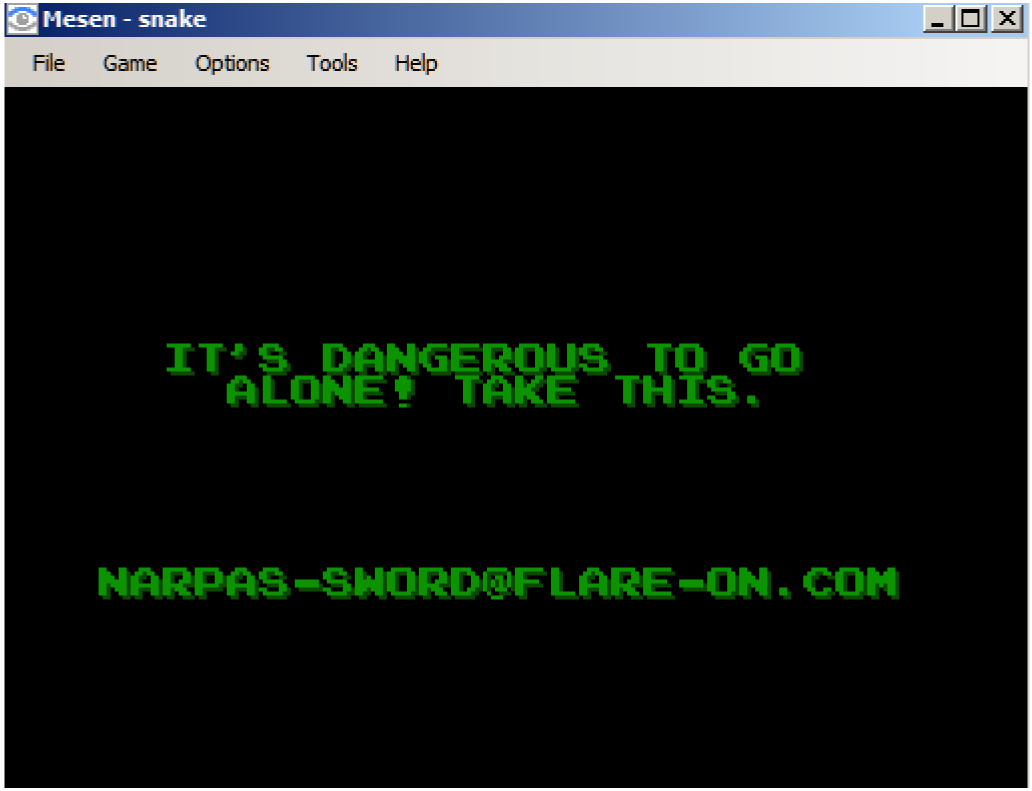
Firmware hacking

firmware reverse engineering

increments the current level when our snake is of length 0x33. At C840 the current level number is compared with 4 which implies there are that many levels. If our current level number is not 4, we jump to C84C or else we continue normally to C844.

Winning the game

If we set the Instruction Pointer to C844 we can bypass playing the game totally. This can be done in Mesen using "Set Next Statement" in the right click pop up menu. Jumping to the address and resuming execution we are pleasantly greeted with the flag.



IT'S DANGEROUS TO GO
ALONE! TAKE THIS.

NARPAS-SWORD@FLARE-ON.COM

Flare-on

frida

getting started with firmware hacking

GSMA

guide to ARM exploitation

hacked security IP cameras

hacked smart devices

hackers

hackfest

hacking smart devices

healthcare business protection against iot threats

healthcare cyber security

how can healthcare fight iot threats

How Mirai botnet infects your device

How Mirai works

how retail can prevent cyber

Figure 9: The flag!

Flag: `NARPAS-SWORD@FLARE-ON.COM`

CTF Flare-on Reversing

Comments

attacks

how to exploit ble

how to hack radio waves

how to protect iot devices

how to secure iot device

IDA internet of things

Internet of Things Security

internet security

ios application security

ios security iot

iot attacks

iot bots, malwares

iot device IoT Devices

IoT Exploitation

iot hacking iot hacks

IoT hacks on ARM devices

iot penetration testing

iot pentest

iot pentesting

iot security

IoT security guidelines

iot security training

iot threats

iot threats to healthcare industry

iotsecurity IP cameras

jtag jtag debugging

latest iot attacks

learn ARM exploitation

measures to prevent cyber attacks on healthcare organisations

Mirai Botnet

mirai history

mobile app

mobile application security

mobile application security testing

mobile security

monitor iot devices

Mozilla

network security in retail

ninja recon technique

NIST

offensive iot exploitation

ola cabs

owasp

owasp appsec

penetration testers

penetration testing

pentesting

pentesting mobile apps

phishing attacks

powerofcommunity

PrinterSecurity

privacy protection

profession professional

qemu quizup

radio communication protocol

radio coomunication

radio waves hacking

recent ARM attacks

recent cyber attacks

recent iot attacks

recent security camera attacks

attacks

retail iot Reversing

safety measures to protect privacy

sdr

secure coding guidelines

security

security cameras

security challenges in retail IoT

security in healthcare iot

security issue

security issues faced by e-retailers

security services

security training

security vulnerability

setup smart devices

smart user security

social networking

spi

steps to prevent iot attacks on healthcare

surveillance cameras hijacked

threat modeling

tools to exploit ble

training

uart

Understanding Mirai Botnet

virus

vulnerabilities discovered in popular IoT IP cameras

vulnerabilities in internet connected cameras

vulnerability

vulnerable ARM devices

What is mirai botnet?

why choose career in cybersecurity

writeups

xposed hooking zigbee

zigbee exploitation

zigbee security zwave

INSTAGRAM

You Might Be Interested In

ANDROID

Solving brut.android Exception

21.DEC.2014 /

CTF

Flare-On 6 CTF WriteUp (Part 12)

20.OCT.2019 /

CTF

Flare-On 6 CTF WriteUp (Part 11)

19.OCT.2019 /

Tags

Analog Modulation

Android

Android Application Security

Android Hands On Security And Exploitation Training

Android Security

Navigation

Home

Attify-Store

Public Classes

Consultation

Contact

Apktool

Application Auditing

Application Security Auditing

Appsec Usa

Appwatch

Arduino Nano

Arm

ARM Binaries

ARM Course

ARM Exploitation Book

ARM Exploitation Video Training

ARM Gadgets

ARM Training

Attify

Attify Badge

Attify Training

Best Security Practices

Biggest Iot Attacks Of All Time

Binwalk

Blackberry Pentesting

Blackhat

Ble

BLE Attacks

BLE Dangers

BLE Hacking And Exploitation

BLE Security Issues

BLE Sniffing

BLE Vulnerabilities

Bleah

Bluetooth Technology

Box

Brut Exception

BtleJuice

Capture Radio Traffic

Career In Cybersecurity

CCTV Cameras

Challenges In Iot Retail

Chroot

Cloud Based Mobile Application Security Scanner

Consulting

CTF

Cyber Attacks

Cybersecurity

Damn Vulnerable IOS App

Dangers Of Iot

DDoS Attacks

Devops

Digital Modulation

Dumping Memory

Embedded Hacking

Expert

Exploit ARM Devices

Exploitation

Exploiting Ble

Exploiting Smart Devices

Firmadyne

Firmware Analysis Toolkit

Firmware Emulation

Firmware Hacking

Firmware Reverse Engineering

Flare-On

Frida

Getting Started With Firmware Hacking

GSMA

Guide To ARM Exploitation

Hacked Security IP Cameras

Hacked Smart Devices

Hackers

Hackfest

Hacking Smart Devices

Healthcare Business Protection Against Iot Threats

Healthcare Cyber Security

How Can Healthcare Fight Iot Threats

How Mirai Botnet Infects Your Device

How Mirai Works

How Retail Can Prevent Cyber Attacks

How To Exploit Ble

How To Hack Radio Waves

How To Protect Iot Devices

How To Secure Iot Device

IDA

Internet Of Things

Internet Of Things Security

Internet Security

Ios Application Security

Ios Security

Iot

Iot Attacks

Iot Bots, Malwares

Iot Device

IoT Devices

IoT Exploitation

Iot Hacking

Iot Hacks

IoT Hacks On ARM Devices

Iot Penetration Testing

Iot Pentest

Iot Pentesting

Iot Security

IoT Security Guidelines

Iot Security Training

Iot Threats

Iot Threats To Healthcare Industry

Iotsecurity

IP Cameras

Jtag

Jtag Debugging

Latest Iot Attacks

Learn ARM Exploitation

Measures To Prevent Cyber Attacks On Healthcare Organisations

Mirai Botnet

Mirai History

Mobile App

Mobile Application Security

Mobile Application Security Testing

Mobile Security

Monitor Iot Devices

Mozilla

Network Security In Retail

Ninja Recon Technique

NIST

Offensive Iot Exploitation

Ola Cabs

Owasp

Owasp Appsec

Penetration Testers

Penetration Testing

Pentesting

Pentesting Mobile Apps

Phishing Attacks

Powerofcommunity

PrinterSecurity

Privacy Protection

Profession

Professional

Qemu

Quizup

Radio Communication Protocol

Radio Coomunication

Radio Waves Hacking

Recent ARM Attacks

Recent Cyber Attacks

Recent Iot Attacks

Recent Security Camera Attacks

Retail Iot

Reversing

Safety Measures To Protect Privacy

Sdr

Secure Coding Guidelines

Security

Security Cameras

Security Challenges In Retail IoT

Security In Healthcare Iot

Security Issue

Security Issues Faced By E-Retailers

Security Services

Security Training

Security Vulnerability

Setup

Smart Devices

Smart User Security

Social Networking

Spi

Steps To Prevent Iot Attacks On Healthcare

Surveillance Cameras Hijacked

Threat Modeling

Tools To Exploit Ble

Training

Uart

Understanding Mirai Botnet

Virus

Vulnerabilities Discovered In Popular IoT IP Cameras

Vulnerabilities In Internet Connected Cameras

Vulnerability

Vulnerable ARM Devices

What Is Mirai Botnet?

Why Choose Career In Cybersecurity

Writeups

Xposed Hooking

Zigbee

Zigbee Exploitation

Zigbee Security

Zwave