Ilhan Mohammed Raja

Cypress, TX 77433
832-571-5823 ilhan.raja@icloud.com

Texas A&M - Computer Science
Stanford University - Center of Professional Development - Adv Computer Security Certificate

Software Projects

http://yungraj.com http://github.com/YungRaj/

Objective

- Aspires to become a kernel engineer, firmware developer, CPU architect engineer, software security engineer for OS vulnerability research or general software engineer for iOS/macOS/etc

Research

- The BIOS and UEFI specification, Boot Chains for the iPhone, Nintendo Switch (Coreboot, Tianocore, Clover, Chameleon, hekate, ReiNX, etc)
- Kernel Engineering (XNU, IOKit, BSD, and Linux) as well kernel reverse engineering for security purposes for iOS, macOS, and Nintendo Switch (Tegra X1)

Programming Languages

Proficient in Python, C, C++, Objective C, ARM Assembly, x86, Java
Moderate in JavaScript, Swift, Bash scripting and ACPI Machine Language
Trying to learn Rust

Experience

Security Research Freelance - DataFlow Security
October 2020 - January 2021

- Reverse engineer baseband firmware for iOS iPhones that are Qualcomm Hexagon, Intel, and ARM based, explore implementations of telecommunications for signal modulation, transmission/reception of signals, OFDMA, CDMA, TDMA, FDMA, AT commands, analyze strings inside the firmware, external references to them and identify basic C functions like memcpy, strcpy, etc, and call sites to them
- Reverse engineer userspace processes and kernel extensions inside iOS for Bluetooth, Cellular etc. Read whitepapers about the proprietary Bluetooth protocols for AppleWatch and AirPods, identify them in reverse engineering efforts, and come up with ways to fuzz the tools using Frida, etc.
- Write a full python tool to pull down iOS firmwares and kernelcaches and binary diff them using a IDA plugin tool named diaphora, symbolicate them using diaphora, update the symbolicated names to the exported sqlite database and diff sqlite database, update the diaphora source code to automate portions of the diffing process, use iometa to create a list of symbolicated C++ metaclass information, build an IDA database that is symbolicated for the diff'ed kernelcache versions (some helper idapython scripts to dump symbols, etc) and finally a loadable diff database into IDA that can partially identify kernel security patches
- Write a tool named diffie-hellcache that takes a kernelcache and a symbol file dumped using IDA pro, or executed on a jailbroken TFP0 device, dumps the kernel, calculates the kernel slide and base, and parses the Mach-O metadata to print load commands, symbols inspect kernel memory and memory inside processes and identify library load addresses, etc, patch find symbols if we want to look for them like kerntask, kernproc, realhost, etc using xrefs, datarefs, etc. dynamically finds offsets for kernel structures like task, proc, ipc_port, etc by using assembly level patchfinding. Every ARM64 instruction is encoded using a special 32 bit sized struct. Find kalloc and IOMalloc variable sized calls inside the kernelcache dynamically. It assembles and disassembles ARM64 machine code and produces IR that is compatible with LLVM IR/SSA/CFG. This tool is effectively a PE framework for iOS

Apple Internship - Firmware Security Engineering
August 2017 - June 2018

- Use the new hash verification tool named eficheck to check customer binaries for malware in the wild using Python and C and create a production framework in Python to detect malicious payloads coming from customers
- Fix Coverity, the static analysis tool to parse EFI code and detect real buffer overflows, information leaks, etc in shipping code
- Fuzz NVRAM variables stored on the flash chip using an IOKit and userland interface
- Develop malware for EFI by fuzzing low level implementations of DHCP, HFS+, APFS (unsuccessful), PCI, SPI, etc
- Triage and fix existing bugs found from fuzzing low level implementations
- Exploit vulnerabilities found in EFI file system implementations to bypass firmware passwords on MacEFI machines
- Inject x86 payloads into existing MacEFIFirmware to detect DMA buffer vulnerabilities and other security issues in hardware/firmware implementation
- Create full fuzzing infrastructure using the simics x86 emulator for NVRAM variables

Developing for BIOS and UEFI - Software Development
May 2014 - October 2015

- Use x86/ARM64 assembly and C to write lower level startup machine code for modern machines
- Conform to the specification of the firmware interfaces and study development of projects such as Clover and Chameleon
- Review open source UEFI firmware through existing reference implementations such as CoreBoot + Tianocore (ACPI and EDK2 development kit)

iOS Reverse Engineering and Jailbreaking
September 2015 - May 2017

- Reverse engineering iOS kernel, binaries and firmware dumps using IDA Pro, Hopper, radare2, class-dump, cycript, etc through dynamic and static analysis of assembly level instructions for x86/ARM, interface definitions and API/ABI's
- Inject into system processes and applications using the Mobile Substrate library using the Theos development tools and reverse engineering exploration (http://github.com/DHowett/theos)
- Implemented a Mobile Substrate tweak for the Snapchat application that notifies the user if a Snapchat streak is going to be over NOTE: StreakNotify was planned into the Phantom Lite project (@CokePokes on Twitter) http://cydia.saurik.com/package/com.yungraj.streaknotify

OSX86
October 2013 - May 2014

- Participated in a hobby project dedicated to installing OS X on a non-Apple Intel and AMD machines called OSX86
- Worked with lower level subsystems and API's in OS X and iOS such as Mach, IOKit, dyld, Mach-O, launchd, macf, sandbox, kauth, BSD, libkern, osmfk, libSystem, SpringBoard, mutex/semaphores, paging, hfs+, GCD, UIKit/Foundation, and Core Foundation to support the ecosystem
- Vastly skilled in Unix-based environments (bash/shell scripting)
- Supported the community by providing advanced technical support in an IRC chat
- Shared full set instructions to install OS X with binary patches, setup instructions and shared ACPI patches and tables such as the DSDT, SSDT and others on github (http://github.com/ilhanraja/XPS-13-9350-OS-X)
- Wrote a MachO binary parser before interning at Apple, it analyzes Objective C metadata, load commands, symbol tables, code signatures (verifies), etc

Application Development (iOS and Android)
May 2013 - January 2015

- Use Xcode and Android Studio to write applications in Java and Objective C(++)
- Utilize the frameworks such as UIKit, SpriteKit, SceneKit, Foundation, CoreFoundation, AppSupport, BulletinBoard, Security, GLKit, SpringBoardServices , CoreGraphics, CoreAnimation and many more to build complete products
- Developed This is the End for the iPhone (github)

Extracurriculars

TAMU CyberSecurity Club
November 2016 - June 2017

- Collectively learn reverse engineering, software security through the static and dynamic analysis tools such as OllyDbg, IDA Pro, radare2, Hopper, and the command line utility class-dump

Service

The Citizen's Foundation
30 hours: Feb 2012 - March 2014

- Actively participated as a member of the local chapter of a national Non Government Organization to provide funds for schools in the underprivileged parts of Pakistan
- Volunteered for the local charity and enticed donations to those interested in the organization

Education

Cypress Ranch High School

Texas A&M University Class of 2021

- Computer Science and Engineering

Stanford University

- Center of Professional Development - Advanced Computer Security Certificate (online @ scpd.stanford.edu)
- Cryptography, Network Security, Mobile Security, SQL Injection/Buffer Overflows, XSS attacks, etc

Publications read

- MacOS and iOS Internals, MacOS and iOS Internals, Volume I: Volume II: Volume III
- The A64 ISA

- The EFI specification