LI CHENYANG

hduch1p@gmail.com · (+86) 137-778-38474 · Security Software Engineer · GitHub @Ch111p

Education Background

Hangzhou Dianzi University 09/2017 - 07/2021

- Computer Science and Technology Bachelor
- ACTIVITIES:
  - Core-member of Vidar-Team, A CTF Group in HDU. Responsible for Reverse Engineering Category
  - Co-planner of the 10th/11th HCTF Cyber Security Competition
- GPA: 3.09/5 (75.9/100)

Work Experience

Security Software Engineer, Security Assurance, **ByteDance Ltd.** 07/2021-Current

- Responsible for developing/maintaining program obfuscate tool
  - Security Compiler
    - Implemented our obfuscation algorithm/protect pass based on LLVM-IR/MIR, like Code Flatten, Virtual Machine Protect, String Encryption, Integrity Check, etc. All these algorithms are implemented on security core code in ByteDance's software.
    - As the only maintainer of this project for around 1.5 years, building all foundation things like CI/CD, automatic tests, and repository management.
  - ELF Executable Packer
    - Pack an ELF executable file into a static executable file. During runtime, it will load the raw LOAD segment from encrypted data, and return execution flow to raw logic. To increase security level, anti-debug algorithms are also implemented. Besides, the whole processes are obfuscated.
  - Static Library's Symbol Encrypt Tool
    - Rename the "modifiable symbol" of the static library, supporting both Mach-O and ELF static library. This tool is extremely useful when you want to hide symbols that are not expected as export symbols in a static library.
- Responsible for doing competitive product research about the obfuscation algorithms

Intern, Security Researcher, 404 Laboratory Team, **Knownsec Inc** 09/2020-12/2020

- Responsible for following up on the latest vulnerabilities, writing papers, and event analysis. Writing PoCs for vulnerabilities we found.
- Responsible for writing security tools
  - A httpd fuzzer based on AFL++, focus on IoT components like httpd.
    - Modify qemu and AFL++ to add some features in binary mode, like socket interface to interact, support fuzz multiple functions at one time, and trace the execution flow of processes created by system/execve.

Intern, Network Security Engineer, **Huawei Technologies Co., Ltd.** 07/2019-09/2019

- Responsible for using internal fuzzing tool to test products of Huawei and third-party open-source project
  - Found open-source's vulnerabilities, applied for CVEs like CVE-2020-19751/19752.
  - Solved tool's false positives problem when fuzzing some specific purpose like JVM.

Miscellaneous

- Computer Language: C(advanced), [C++,Python,(Assembly/LLVM IR?)](intermediate)
- Natural Language: Chinese(Native), English(Conversational), Japanese(Beginner)
  - Certification: TOEFL iBT(83/120), JLPT N2(159/180)