Qiuchen Yan

https://www-users.cs.umn.edu/~yanxx297/ Email: yanxx297@umn.edu
https://github.com/yanxx297 Mobile : +1-651-235-4138

EDUCATION

University of Minnesota, Twin Cities Minneapolis, MN
Ph.D. in Computer Science, GPA 3.60 Sep. 2014 – May 2021 (anticipated)
Master of Science in Computer Science, GPA 3.625 Sep. 2012 – May 2014

Shandong University of Science and Technology Qingdao, China
Bachelor of Engineering in Computer Science, GPA 3.65 Sep. 2008 – July 2012

SKILLS

Programming Languages: C/C++, Python, OCaml, Java, HTML/CSS, X86 assembly
Systems & Tools: FuzzBALL(symbolic execution), Xed (Intel Pin), DWARF, VEX, Git, QEMU

COURSEWORK AND PERSONAL PROJECTS

Render helper 2017
Photoshop script that automates layer processing. Javascript.

Static BLOG/website template 2016
Hugo templates for my personal website and BLOG. HTML/CSS, Javascript.

Reproduce the Lucky Thirteen attack 2014
Implemented a timing side channel attack[i] to the TLS protocol. C, Shell script, POSIX.

Sybil attack study 2014
Surveyed about the Sybil attack in online social network and its state-of-art defense approaches, and collected data from real world Sybil communities in Sina Weibo. Weibo API for C/C++.

Encrypted address book for Android 2012
Designed and implemented an Android address book app that can send encrypted contact info via text message. Java(back-end), XML(front-end) and SQLite.

RESEARCH PROJECTS

Linux Kernel Vulnerability Verification by Symbolic Execution 2018 - present
- Reproduced a vulnerability in Linux kernel file system according to CVE.
- Improved the VEX based IR translator of FuzzBALL(a symbolic execution platform) to support Linux kernel.

Loop Summarization for Symbolic Execution (Darpa CGC) 2014 - 2015, 2018 - present
- Implemented a trace based loop summarization algorithm[ii] on a symbolic execution platform. 3000 lines of OCaml code.
- Contributed vulnerability detection code for team FuzzBOMB in Darpa CGC Qualification Event.

Fast & Automatic Emulator Testing System 2015 - 2018

Built an automatic test generator for 870 x86 instruction variations based on previous work[iii].

- Generated test aggregations each of which equivalent to 800 tests in original system but runs 200 times faster. Python and C. Decode input instructions by Xed. Tests written in x86 Assembly.
- Tested 39,685 historical versions of QEMU and identified 3 verified bugs.

Binary Level Type Inference 2013 – 2014

Built a static signedness inference system that achieves 96% true positive rate on GNU core utilities.

- Translated binaries to data flow graphs in static single assignment form, and divide it to signed and unsigned parts based on data flow and heuristics.
- Built with Vine and libdwarf. 10,000 lines of C/C++ code.

PUBLICATION

Qiuchen Yan, Stephen McCamant,"Fast PokeEMU: Scaling Generated Instruction Tests Using Aggregation and State Chaining," The 14th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (VEE'18)

Qiuchen Yan, Stephen McCamant, "Conservative Signed/Unsigned Type Inference for Binaries using Minimum Cut," Technical report

SERVICE

- Presented my work on The 14th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (VEE'18)
- Guest lectures on security related courses at the University of Minnesota.
- Presented and discussed recent papers on cryptography/security related seminars.
- Co-advised a under graduate student on a project related to program analysis.

i Nadhem J. Al Fardan and Kenneth G. Paterson. Lucky thirteen: Breaking the tls and dtls record protocols. In Proceedings of the 2013 IEEE Symposium on Security and Privacy, SP '13, pages 526–540, Washington, DC, USA, 2013. IEEE Computer Society.

ii Patrice Godefroid and Daniel Luchaup. Automatic partial loop summarization indynamic test generation. InProceedings of the 2011 International Symposium onSoftware Testing and Analysis, ISSTA '11, pages 23–33, New York, NY, USA, 2011.ACM.

iii Lorenzo Martignoni, Stephen McCamant, Pongsin Poosankam, DawnSong, and Petros Maniatis. 2012. Path-exploration lifting: hi-fi testsfor lo-fi emulators. InProceedings of the 17th International Conferenceon Architectural Support for Programming Languages and OperatingSystems, ASPLOS 2012, London, UK, March 3-7, 2012. 337–348.DOI:http://dx.doi.org/10.1145/2150976.2151012