

Dongliang Mu

Associate Professor
School of Cyber Science and Engineering
Mingde Building A309
Huazhong University of Science and Technology
Wuhan, Hubei, China 430043

Tel: +86 15171501908
dzm91@hust.edu.cn
mudongliangabcd@outlook.com
<https://mudongliang.github.io/about>
Google Scholar

Research Interests

My research focuses on **Software and System Security**. More specifically, my research interests span the areas of Software Failure Diagnosis, Vulnerability Reproduction, Vulnerability Fuzzing, and Binary Analysis. Currently, I am really interested in the vulnerability fuzzing, analysis and fixing of kernel programs (e.g., Linux Kernel).

Education

- 2014-2019 **Ph.D.** in Computer Science and Technology, *Nanjing University*
Adviser: [Professor Bing Mao](#)
- 2010-2014 **B.E.** in Computer Science and Technology, *Zhengzhou University*

Experiences

- 2020/08-now **Associate Professor**, *Huazhong University of Science and Technology*
- 2020/01-07 **Research Fellow**, *Pennsylvania State University*
Adviser: [Professor Xinyu Xing](#)
- 02/2018 **Organizer of 2018 Penn State Cybersecurity Competition**, *Pennsylvania State University*
HomePage : <https://psusecurity.github.io/>
- 2016-2020 **Research Assistant**, *Pennsylvania State University*
Adviser: [Professor Xinyu Xing](#)
- 2014-2016 **Graduate Research and Teaching Assistant**, *Nanjing University*
Adviser: [Professor Bing Mao](#)

Honors & Awards

- 07/2019 **Student Travel Grant of 14th ACM ASIA Conference on Computer and Communications Security**
- 10/2018 **Artificial Intelligence Scholarship at Nanjing University**
- 10/2018 **ACM CCS Outstanding Paper Award (Top 1)**
- 05/2017 **Student Travel Grant of 38th IEEE Symposium on Security and Privacy**

Publications

* means equal contribution

Conference Papers:

- P-9 **[ASE 2019] Mu, D.***, Guo, W.*, Cuevas, A., Chen, Y., Gai, J., Xing, X., Mao, B., Song, C., “RENN: Efficient Reverse Execution with Neural-Network-assisted Alias Analysis”, In Proceedings of the 34th IEEE/ACM International Conference on Automated Software Engineering, San Diego, CA, November 2019. **(CCF A)**
- P-8 **[AsiaCCS 2019] Chen, Y.***, **Mu, D.***, Sun, Z., Xu, J., Shen, W., Xing, X., Lu, L., Mao B., “Ptrix: Efficient Hardware-Assisted Fuzzing for COTS Binary”, In Proceedings of the 14th ACM ASIA Conference on Computer and Communications Security, Auckland, New Zealand, July 2019.

P-7

- [USENIX Security 2019] Guo, W., **Mu, D.**, Xing, X., Du, M., Song, D., “DEEPVSA: Facilitating Value-set Analysis with Deep Learning for Postmortem Program Analysis”, In Proceedings of the 28th USENIX Security Symposium, Santa Clara, California, August 2019. **(CCF A)**
- P-6 [CCS 2018] Guo, W., **Mu, D.**, Xu, J., Su, P., Wang, G., Xing, X., “LEMNA: Explaining Deep Learning based Security Applications”, In Proceedings of The 25th ACM Conference on Computer and Communications Security, Toronto, Canada, October 2018. **(CCF A, Outstanding Paper Award)**
- P-5 [USENIX Security 18] **Mu, D.**, Cuevas, A., Yang, L., Hu, H., Xing, X., Mao, B., Wang, G., “Understanding the Reproducibility of Crowd-reported Security Vulnerabilities”, In Proceedings of the 27th USENIX Security Symposium, Baltimore, Maryland, August 2018. **(CCF A)**
- P-4 [SecureCOMM 17] **Mu, D.**, Guo, J., Ding, W., Wang, Z., Mao, B., Shi, L., “ROPOB: Obfuscating Binary Code via Return Oriented Programming”, In International Conference on Security and Privacy in Communication Systems, Niagara Falls, Canada, October 2017.
- P-3 [SecureCOMM 17] Zhu, J., Zhou, W., Wang, Z., **Mu, D.**, Mao, B., “DiffGuard: Obscuring Sensitive Information in Canary Based Protections”, In International Conference on Security and Privacy in Communication Systems, Niagara Falls, Canada, October 2017.
- P-2 [USENIX Security 17] Xu, J., **Mu, D.**, Xing, X., Liu, P., Chen, P., Mao, B., “POMP: Postmortem Program Analysis with Hardware-Enhanced Post-Crash Artifacts”, In Proceedings of the 26th USENIX Security Symposium, Vancouver, Canada, August 2017. **(CCF A)**
- P-1 [CCS 16] Xu, J., **Mu, D.**, Chen, P., Wang, P., Xing, X., Liu, P., “CREDAL: Towards Locating a Memory Corruption Vulnerability with Your Core Dump”, In Proceedings of the 23rd ACM Conference on Computer and Communications Security, Vienna, Austria, October 2016. **(CCF A)**

Journal Papers:

- J-1 [TSE 2019] **Mu, D.**, Du, Y., Xu, J., Xu, J., Xing, X., Mao, B., “POMP++: Facilitating Postmortem Program Diagnosis with Value-set Analysis”, In IEEE Transactions on Software Engineering, 2326-3881, 2019. **(CCF A)**

Academic Service

Reviewer:

- [Oakland SP] IEEE Symposium on Security and Privacy: 2021 subreviewer
 [ACM CCS] ACM Conference on Computer and Communications Security: 2019, 2020 subviewer
 [USENIX Security] USENIX Symposium on Security: 2020 subreviewer
 [ACSAC] Annual Computer Security Applications Conference: 2019, 2020 subviewer
 [ISC] Information Security Conference: 2019 subreviewer
 [JSA] Journal of Systems Architecture: 2020 reviewer

Talks

- 7/2019 Facilitating Vulnerability Removal by Diagnosing Software Failures
InforSec Workshop, Wuhan, Hubei, China
- 7/2019 Ptrix: Efficient Hardware-Assisted Fuzzing for COTS Binary
AsiaCCS, Auckland, New Zealand
- 5/2019 Towards Facilitating the Removal of Software Defects
QiZhen Youth Forum in Zhejiang University, Hangzhou, Zhejiang, China
- 10/2018 From Physical Security to Cyber Security: How to forge data spoofing personalized auto insurance
GeekPwn China, Shanghai, China
- 8/2018 Understanding the Reproducibility of Crowd-reported Security Vulnerabilities
USENIX Security, Baltimore, USA

Research Projects

- 2018-2019 **Deep Learning Assisted Program Analysis** *Cyber Security Lab, Penn State University*
 • Develop deep learning assisted Value Set Analysis to facilitate Postmortem Program Analysis. [See P-7, P-9]

- 2017-2018 **Vulnerability Reproduction** *Cyber Security Lab, Penn State University*
- Perform an in-depth analysis on the reproducibility of crowd-reported security vulnerabilities. [See P-5]
- 2016-2017 **Analysis on Software Crashes** *Cyber Security Lab, Penn State University*
- Analyze core dumps caused by memory corruption vulnerabilities; locate the crash point; restore the stack trace; narrow down code segments carrying vulnerabilities. [See P-1]
 - Enhance a core dump with execution trace logged through Intel Processor Tracing; perform reverse execution and symbolic execution against the trace; pinpoint the root cause of software crash. [See P-2]
 - Leverage Value-set Analysis to improve the memory alias problem in the POMP, to achieve better effectiveness and efficiency. [See J-1]
- 2015-2016 **Obfuscation based ROP** *System Security Lab, Nanjing University*
- Propose an obfuscation scheme for binaries based on ROP (Return Oriented Programming), which aims to serve as an efficient and deployable anti-reverse-engineering approach. [See P-4]

Teaching

- Software Security, Instructor, Spring 2020
- Assembling Language, Instructor, Fall 2021

Open Source Projects

- 06/2016 **LinuxFlaw**
- Record all the memory error vulnerabilities we used for our Usenix Security 2018 [see P-5]. We not only disclose the detail of vulnerability reproduction but also try to create docker images about those vulnerabilities as possible as we can.
- 06/2016 **Source-packages**
- Source code for the vulnerable software in the LinuxFlaw
- 06/2016 **Dockerfiles**
- All the useful Dockerfiles and related tools in the LinuxFlaw
- 04/2016 **TraditionalMitigation**
- Summarize traditional mitigations in GCC to defend Memory Corruption Vulnerability
- 05/2017 **POMP**
- Leverage Intel PT to do reverse execution, and diagnose the root cause of software failure
- 06/2019 **DEEPVSA**
- Facilitate Value-set Analysis with Recurrent Neural Network for better Postmortem Program Analysis
- 12/2014 **Linux-insides**
- One book-in-progress about Linux Kernel and its insides.
- 12/2014 **Linux-insides-zh**
- Chinese Translation of [linux-insides](#). This upstream repo is a book-in-progress about Linux Kernel and its insides.

CVE Discovered

CVE ID	Vulnerability Type	Vulnerable Software	Vulnerable Version
CVE-2018-8816	Stack Exhaustion	perl	5.26.1
CVE-2018-8881	Heap buffer overflow	nasm	2.13.02rc2
CVE-2018-8882	Stack buffer overflow	nasm	2.13.02rc2
CVE-2018-8883	Global buffer overflow	nasm	2.13.02rc2
CVE-2018-10016	Division-by-zero	nasm	2.14rc0
CVE-2018-9138	Stack Exhaustion	binutils	2.29
CVE-2018-9996	Stack Exhaustion	binutils	2.29
CVE-2018-10316	Denial-of-Service	nasm	2.14rc0
CVE-2018-9251	Denial-of-Service	libxml2	2.9.8

Upstream Linux Kernel Bug Patches

Age	Kernel Commits
8/13/21	ipack: tpcizoo: fix memory leak in the tpcizoo_register
8/13/21	ipack: tpcizoo: fix many double free issues in tpcizoo_pci_probe
8/4/21	media: em28xx-input: fix refcount bug in em28xx_usb_disconnect
7/22/21	spi: meson-spicc: fix memory leak in meson_spicc_remove
7/22/21	media: dvb-usb: Fix error handling in dvb_usb_i2c_init
7/22/21	media: dvb-usb: fix uninit-value in vp702x_read_mac_addr
7/22/21	media: dvb-usb: fix uninit-value in dvb_usb_adapter_dvb_init
7/21/21	usb: hso: fix error handling code of hso_create_net_device
7/17/21	netfilter: nf_tables: fix audit memory leak in nf_tables_commit
7/15/21	usb: hso: fix error handling code of hso_create_net_device
7/7/21	ieee802154: hwsim: fix GPF in hwsim_new_edge_nl
6/22/21	ieee802154: hwsim: fix GPF in hwsim_set_edge_lqi
6/18/21	ieee802154: hwsim: Fix memory leak in hwsim_add_one
6/16/21	net: caif: modify the label out_err to out
6/14/21	net: usb: fix possible use-after-free in smsc75xx_bind
6/8/21	ieee802154: hwsim: Fix possible memory leak in hwsim_subscribe_all_others
6/2/21	media: dvd_usb: memory leak in cinergytz_fe_attach
5/21/21	ALSA: control led: fix memory leak in snd_ctl_led_register
5/17/21	misc/uss720: fix memory leak in uss720_probe
1/26/21	NFC: nci: fix memory leak in nci_allocate_device

Last updated: September 16, 2021

[Dongliang Mu de Blog](#)