Xiaoguang Wang

xgwang9@hotmail.com · (+86) 136-8926-9881 · xgwang.info

Education

PhD candidate in Computer Science, Xi'an Jiaotong University, China 2010.9 – 2017.6

- *Thesis:* Enhancing System Software Security in Cloud Environment
- *Advisor*: Prof. Yong Qi (http://www.cs.xjtu.edu.cn/info/1267/1425.htm)

Visiting PhD student at Florida State University, Tallahassee, U.S. 2013.8 – 2015.8

- *Research Interests*: System security; Operating system; Virtualization; Compiler.
- *Advisor*: Dr. Zhi Wang (http://www.cs.fsu.edu/ zwang/)

B.Eng. in Software Engineering, Northwestern Polytechnic University, China 2006.9 – 2010.6

- Thesis: Design and implementation of a Guest OS communication module with shared memory. (Outstanding Bachelor's Thesis, NWPU)
- GPA: 85.8/100. (Rank: 16 out of 236)

Internship Experience

Samsung Research China (SRC), Xi'an, China 2016.11 – 2017.1

- Topic: Bare-metal operating system migration

Selected Publications

- **Xiaoguang Wang**, Yong Qi, Zhi Wang, Yue Chen, Yajin Zhou. "Design and Implementation of SecPod, A Framework for Virtualization-based Security Systems". IEEE Transactions on Dependable and Secure Computing (TDSC). (Accepted to appear)
- **Xiaoguang Wang**, Yong Qi. "Secure the Commodity Applications Against Address Exposure Attacks". The 22nd IEEE Symposium on Computers and Communications (ISCC'17). (Accepted to appear)
- **Xiaoguang Wang**, Yong Qi, Chi Zhang, Saiyu Qi, Peijian Wang. "SecretSafe: A Lightweight Approach Against Heap Buffer Over-read Attack". IEEE 41th Computer Software and Application Conference (COMPSAC'17). (Accepted to appear)
- **Xiaoguang Wang**, Yue Chen, Zhi Wang, Yong Qi, Yajin Zhou. "SecPod: A Framework for Virtualization-based Security Systems". Proceedings of the 2015 USENIX Annual Technical Conference (USENIX ATC '15).
- Yajin Zhou, **Xiaoguang Wang**, Yue Chen, Zhi Wang. "ARMlock: Hardware-based Fault Isolation for ARM". Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS). ACM, 2014.
- **Xiaoguang Wang**, Yong Qi, Yuehua Dai, Jianbao Ren, and Zhang Hang. "Protecting Outsourced Data Privacy with Lifelong Policy Carrying." In High Performance Computing and Communications (HPCC), 2013 IEEE 10th International Conference on, pp. 896-905. IEEE, 2013.
- **Xiaoguang Wang**, Yong Qi, Yuehua Dai, and Jianbao Ren. "Transparent Security-Sensitive Process Protection via VMM-Based Process Shadowing." In Computer Software and Applications Conference Workshops (COMPSACW'13), 2013 IEEE 37th Annual, pp. 115-120. IEEE, 2013.
- **Xiaoguang Wang**, Yong Qi, Yuehua Dai, Yi Shi, Jianbao Ren, Yu Xuan. "TrustOSV: Building Trustworthy Executing Environment with Commodity Hardware for a Safe Cloud", Journal of Computers, 9.10 (2014): 2303-2314.
- Jianbao Ren, Yong Qi, Yuehua Dai, **Xiaoguang Wang**, Yi Shi. "AppSec: A Safe Execution Environment for Security Sensitive Applications." Proceedings of the 11th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (VEE). ACM, 2015.
- Dai, Yuehua, Yong Qi, Jianbao Ren, Yi Shi, **Xiaoguang Wang**, and Xuan Yu. "A lightweight VMM on many core for high performance computing." In ACM SIGPLAN/SIGOPS Internation conference on Virtual Execution Environment (VEE), pp. 111-120. ACM, 2013.

Selected Research Projects

Runtime Code Execution Path Randomization Jan.2016 - Oct.2016

- Computer software is suffered from the vulnerability exploits. A malicious attacker could hijack the software with multiple techniques, such as ROP. Address Space Layout Randomization (ASLR) can raise the bar for such attacks, but it still can be bypassed by tricky attacks such as information leaks. A leaked pointer could be enough to de-randomize the code. In this project, we implemented the execution path randomization. Specifically, multiple copies of the code are maintained in the memory, and the code execution paths are randomly selected and executed. Therefore it makes attackers difficult to generate the malicious payload even if the attackers have de-randomized the code. We have built it with a modified compiler and a loader extension.

Compiler-based ASLR enhancement Aug.2014 - Apr.2015

- Address Space Layout Randomization (ASLR) is widely used to prevent attackers from exploiting the vulnerabilities. Current attacks use memory information leakage to guess out the process memory layout. Therefore, ASLR could be bypassed. In this project, we prevent the memory address leakage problem by hiding the code pointers and data pointers into the execute-only memory. We have implemented the source code compatible code translator as a LLVM extension.

Isolating the OS kernel page table update Dec.2013 - May.2014

- Commodity operating systems leverage the write $\oplus$ execute (W $\oplus$ X) on kernel address space to protect kernel memory from being violated (e.g. prevent kernel code to be modified or kernel data to be executed). This policy is enforced in kernel page tables and it is critical to a lot of kernel security applications. However, the page table itself is writable (kernel needs to frequently update page table for memory mapping), thus it is under attack. In this project, we seek to protect the page table as well as the memory mapping update. We isolate the kernel page table in a secure address space, and verify the memory mapping update in that space.

Transparent security-sensitive process protection with process-shadowing Jan.2012 - May.2012

- Out-of-VM (Virtual Machine) process sandboxing lacks transparency for inter-process communication (IPC) as the protected process is isolated by the VM from other running processes. On the other hand, sandboxing a process within an OS is not safe especially when the OS is not trusted (as malicious OS might compromise any software running on top of it). To achieve both strong VM-level isolation and transparent IPC, we built the shadow process execution system. It protects a running process by transparently migrating the running process into an isolated VM, at the same time, redirects all inter-process communication system calls back to the original VM. We built the prototype on KVM and it can transparently and on-demand migrate a running process.

Awards

- USENIX Annual Technical Conference (USENIX ATC 2015) Student travel grant Jul.2015
- Department of Computer Science, Florida State University, CS Expo, Best poster award. Dec.2013
- Excellent graduate student award at XJTU Oct.2012, Oct.2016
- Outstanding Bachelor's Thesis, NWPU May.2010

Skills

Linux kernel driver development. Excellent in C/C++/Assembly/Shell languages.
Familiar with perl/awk script languages. I used to use Java a lot.
IDE: vim Platform: Linux, Windows.
Fluent in English writing, speaking. Native language Chinese.

Hobbies

- Table tennis, Badminton, Hiking, Swimming, Harmonica, Traveling.