Author: Beni Selyem

System Call Sandboxing

Responsible Professor: Alex Voulimeneas

1. Introduction & Background

- **System call:** Talk to hardware through kernel (Figure 1)
- **Sandboxing:** Restrict system calls to minimal required set (Figure 2)
- **Problem:** Which calls to block and which ones to allow?
- **Solution:** Analyse applications, find out which calls are needed
- **Gap:** Static vs Dynamic analysis & Single vs Multi phase model (Figure 3)

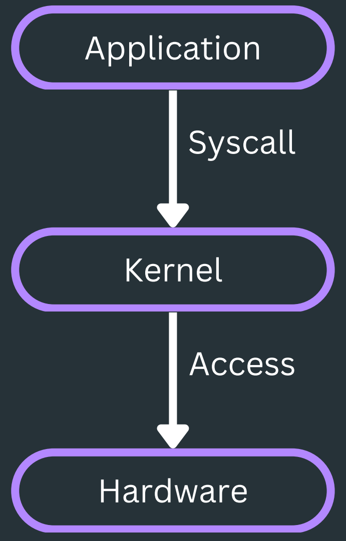

Figure 1: Regular syscall flow

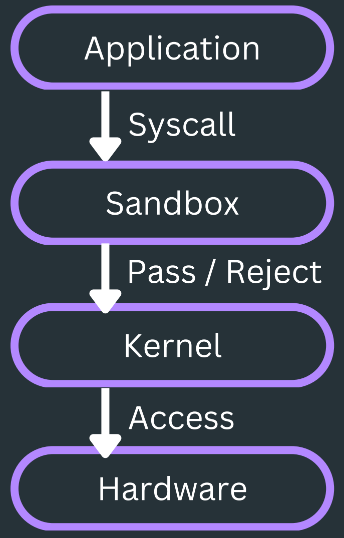

Figure 2: Sandboxed flow

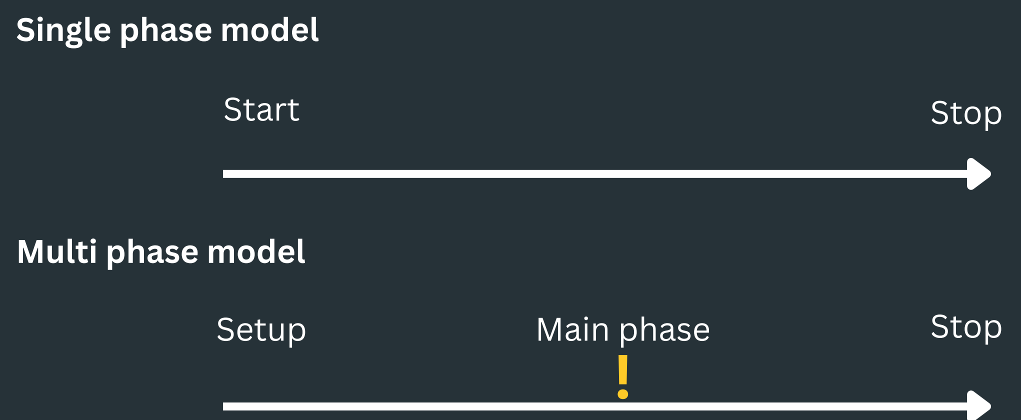

Figure 3: Single phase & multi phase model

2. Research questions & Contributions

1. How can dynamic analysis method be used to identify the used system calls?
2. What is the runtime and accuracy of single phase model static analysis tools?
3. What is the runtime and accuracy of multi phase model static analysis tools?

- Dynamic analysis tool
- Analysis of various static analysis solutions
- Comparison of dynamic vs static and single phase vs multi phase approaches

3. Dynamic tracer

- Gather list of system calls used
- Based on ptrace system call
- Traces multiple processes & threads
- Records process structure and system calls
- Requires exploration of many program states

4. Experiment setup

Test programs: ls, sqlite3, redis

SQLite redis

Analysis tools: Chestnut [1], Confine [4], temporal-specialization [3], sysfilter [2]

Measurements: Completion, Runtime, Accuracy

5. Results & Discussion

TODO: add results when available

6. Limitations & Conclusion

TODO: draw conclusions from results when available

References

[1] Claudio Canella, Mario Werner, Daniel Gruss, and Michael Schwarz. Automating seccomp filter generation for linux applications. In Proceedings of the 2021 on Cloud Computing Security Workshop, CCSW '21, page 139–151, New York, NY, USA, 2021. Association for Computing Machinery.

[2] Nicholas DeMarinis, Kent Williams-King, Di Jin, Rodrigo Fonseca, and Vasileios P. Kemerlis. sysfilter: Automated system call filtering for commodity software. In International Symposium on Recent Advances in Intrusion Detection, 2020.

[3] Seyedhamed Ghavamnia, Tapti Palit, Shachee Mishra, and Michalis Polychronakis. Temporal system call specialization for attack surface reduction. In Proceedings of the 29th USENIX Conference on Security Symposium, SEC'20, USA, 2020. USENIX Association.

[4] Seyedhamed Ghavamnia, Tapti Palit, Azzedine Benameur, and Michalis Polychronakis. Confine: Automated system call policy generation for container attack surface reduction. In International Symposium on Recent Advances in Intrusion Detection, 2020