

李清伟 (Lslightly)

✉ lqw332664203@mail.ustc.edu.cn · 🏠 [lslightly.github.io](https://github.com/lslightly) · 🆔 0009-0002-3888-2435 · 🌐 [Lslightly](#) · 知乎 · 📺

教育经历

中国科学技术大学，计算机专业，硕士，中国 2023 年 8 月 – 2026 年 6 月 (预计)
导师: 张昱 研究领域: 程序分析, 语言特征实证分析, 内存管理, 语言运行时优化

中国科学技术大学，计算机专业，本科，中国 2019 年 8 月 – 2023 年 6 月
GPA: 3.93/4.30 排名: 6/169

工作经历

学术经历

Go 语言程序的内存性能与安全问题实证研究 2023 年 10 月 – 2025 年 5 月

- 基于 CodeQL 实现批量代码仓库的内存性能相关模式扫描统计，以指导静态分析和编译优化。
- 人工分析总结内存安全问题模式。
- 针对切片表达式可能导致内存泄漏的问题模式编写 Linter 工具进行检测。

MEA2: a Lightweight Field-Sensitive Escape Analysis with Points-to Calculation for Golang 2023 年 9 月 – 2024 年 10 月

- 考虑 Go 语言的独特语言特性，在 LLVM IR 上实现更加精确的域敏感逃逸分析算法。在仅有额外 1% 编译时间开销下平均减少 7.9% 的堆分配点。
- 个人贡献: 过程间分析摘要设计，对象堆分配优化为栈分配实现。

DBI-Go: 动态插桩定位 Go 二进制的非法内存引用 2023 年 6 月

- 基于二进制插桩的内存引用分析。检测编译器逃逸分析算法错误导致的悬垂指针问题。
- 个人贡献为提供特定运行时函数的检测模式。

GoLLVM 内存管理的语义和优化 [📄 Lslightly/gollvm-heap2stack-invariant-verify](#) 2023 年 2 月 – 2023 年 5 月

- 基于 K 框架对 GoLLVM 编译器编译生成的 IR 进行语义建模
- 实现对编译器逃逸分析算法错误导致的悬垂指针错误检测。

技能

- 编程语言: 熟悉 Go/Python/C/C++(≤17) > 略熟悉 Rust/CodeQL > 了解 JS/TS/Haskell/OCaml/SystemVerilog
- 展示语言: 熟悉 \LaTeX Typst Markdown Excalidraw > 略熟悉 docx/pptx/剪映
- 开发环境:
 - VSCode+ChatGPT(Copilot/Cline)+OhMyZsh+PowerShell+Obsidain > LazyVim
 - 有使用 GitHub、GitLab 等团队协作工具的经验
- 框架: Go Toolchain(build/compile/runtime)/LLVM/...

其它