Modern day automated security analysis depends on Control Flow Graphs (CFG) that are assumed to be complete for sourceless executables. Often, these CFGs are flawed due to a lack of resolving indirect control flow. To increase automated analysis of executables, we must increase the completeness of CFG’s by resolving these indirect control locations.

On average executable programs contain nearly 91 unresolved control flow locations, accounting for nearly 50% of all indirect control flow in an executable. This is a major analysis flaw.

- Resolving around 25% more indirect control flow locations on average
- Ability to resolve constant propagation across functions and resolved structs in memory
- Created a Python framework for extending with more modern resolving methods like Andersen Analysis (future work)