CS5231 Assignment Proposal

Members

1. Jeremy Heng (A0146789H)

Concept

Successful heap exploitation can hinge on multiple factors such as memory allocator implementation, chunk header sizes, fitting and coalescing strategies, , the magnitude of control an attacker possesses, and the availability of memory leaks. Crafting such an exploit can take extreme finesse and requires a large amount of effort in analysing and debugging.

We propose a tool to assist an exploit developer with the creation of such heap based exploits. Such a tool would:

1. Integrate with a debugger or dynamic analysis tool to track memory allocation function calls.
2. Generate a visualisation of memory space associated with the heap and its metadata over time.
3. Simulate `malloc()`, `free()`, or other memory allocation functions at a certain point in time to easily explore the effects.

Further enhancements could include:

1. A feature to suggest exploit techniques given the primitives available in the vulnerable binary. Some of the techniques are published with names such as House of Spirit, House of Mind, etc.

Progress Report

Every week, a progress report article will be posted on the project blog: https://nnamon.github.io/heapfriend/. The code during development is already open sourced in my github repository: https://github.com/nnamon/heapfriend.

Project Schedule

- Week 5 (Sept 11 - Sept 15) - Literature Survey: Perform a study on the existing literature and tools.
- Week 6 (Sept 18 - Sept 22) - Initial Design: Decide on an architecture, underlying technology, concretise the feature list.
- Week 7 (Sept 25 - Sept 29) - Alpha Development
- Week 8 (Oct 2 - Oct 6) - Alpha Development

- Week 9 (Oct 9 - Oct 13) - Alpha Release: First working prototype containing tracing and visualation features.
- Week 10 (Oct 16 - Oct 20) - Alpha Review/Beta Development
- Week 11 (Oct 23 - Oct 27) - Beta Development
- Week 12 (Oct 30 - Nov 3) - Beta Release/RC Development: Feature freeze on the application. Only quality of life enhancements and bug fixing from this point on.
- Week 13 (Oct 6 - Oct 10) - Beta Review/Release Candidate Development
- Week 14 (Oct 13 - Oct 17) - Release Candidate Release: The tool should fulfill the basic objectives as set out in the Concept section. The associated documentation should be complete.
- Week 15 (Oct 18) - Final Report Submitted