USENIX Security '24 Artifact Appendix: D-Helix: A Generic Decompiler Testing Framework Using Symbolic Differentiation

Muqi Zou
PurSec Lab, Purdue University

A   Artifact Appendix

A.1   Abstract

This document contains information about our artifacts which we make available via a GitHub page: https://github.com/purseclab/D-helix/tree/691c162ec5d4a967adae395546edc13eb3285798. The GitHub repository is BSD 3-clause licensed and contains source code and scripts for setting up the environment for D-helix. The experiments run on Ubuntu 20.04.

A.2   Description & Requirements

A.2.1   Security, privacy, and ethical concerns

There are no ethical concerns.

A.2.2   How to access

Source code and documentations can be accessed at https://github.com/purseclab/D-helix/tree/691c162ec5d4a967adae395546edc13eb3285798

A.2.3   Hardware dependencies

We run D-helix on a 112-core Intel(R) Xeon(R) Gold with 1 TB of physical memory. The minimum hardware requirement for D-helix is the same as the minimum requirements for Ubuntu 20.04.

A.2.4   Software dependencies

We run D-helix on Ubuntu 20.04, with GCC (version 11.1.0), Clang (version 16.0.0), and Python3 (version 3.8). clang3.8, angr, prompt, and Ghidra are required to install from the following links:

https://releases.llvm.org/download.html
https://github.com/angr/angr-dev
https://github.com/sysrel/PROMPT
https://github.com/NationalSecurityAgency/ghidra/tree/Ghidra_10.0_build

A.2.5   Benchmarks

All 2004 programs shown in Table 4 can be found under the directory named dataset of https://github.com/purseclab/D-helix/tree/691c162ec5d4a967adae395546edc13eb3285798.

A.3   Set-up

A.3.1   Installation

All necessary steps to install D-helix have been included in the README in https://github.com/purseclab/D-helix/tree/691c162ec5d4a967adae395546edc13eb3285798. Users need to first git checkout angr, prompt, and Ghidra to specific commits and then patch them using the provided patches.

A.3.2   Basic Test

Once following the installation and running instructions description in the README, we expect to see a file named *diff_result* to show the result of SYMD-IFF and a directory named *correct_result* to show the result of Tuner. We give samples of these two results with the same name in https://github.com/purseclab/D-helix/tree/691c162ec5d4a967adae395546edc13eb3285798

A.4   Version

Based on the LaTeX template for Artifact Evaluation V20231005. Submission, reviewing and badging methodology followed for the evaluation of this artifact can be found at https://secartifacts.github.io/usenixsec2024/.