A. Artifact Appendix

A.1 Artifact check-list (meta-information)

- **Program:** The code repository for our framework along with the test suite.
- **Compilation:** Python for running the plotting scripts and the Lean4 toolchain, downloaded via `elan`
- **Run-time environment:** Any operating system that supports Docker.
- **Hardware:** Any x86-64 machine.
- **Output:** Key theorems of the paper will be built and shown to have no unsound axioms.
- **How much disk space required (approximately)?:** 10GB
- **How much time is needed to prepare workflow (approximately)?:** 1hr
- **How much time is needed to complete experiments (approximately)?:** 1hr
- **Publicly available?:** Yes
- **Code licenses (if publicly available)?:** MIT
- **Archived (provide DOI)?:** 10.5281/zenodo.11506328

A.2 Description

A.2.1 Hardware dependencies

None.

A.2.2 Software dependencies

Docker is necessary to run our artifact. The Docker image has all dependencies needed to compile our framework with Lean4.

A.3 Experiment workflow

Access the docker image `opencompl-ssa` from `https://zenodo.org/records/11506328`

```
$ docker load -i opencompl-ssa.tar
$ docker run -it siddudruid/opencompl-ssa
# | This clears the build cache,
# | fetches the maths library from the build cache,
# | and builds our framework.
$ cd /code/ssa && lake clean && lake exe cache get && lake build
```

To copy files for inspection from the docker container into the host, keep the container running, and in another shell instance, use the `docker cp` command to copy files from within the container out to the host:

```
$ docker container ls # find   ID
$ docker cp <CONTAINERID>:<PATH/INSIDE/CONTAINER> \
            <PATH/OUTSIDE/CONTAINER>
```

For more about `docker cp`, please see: (`https://docs.docker.com/engine/reference/commandline/cp/`)

A.4 Evaluation and expected results

On running `lake build`, the build succeeds with no errors. By running `grep -R "guard_msgs"`, one should see that every key theorem is guarded by a `guard_msgs`, which checks that the axioms used by the theorem does *not* include `sorry`. Check the following core theorems:

A.4.1 Core Framework Theorems

```
SSA/Core/Framework.lean
2408:#guard_msgs in #print axioms denote_rewritePeepholeAt
2444:#guard_msgs in #print axioms denote_rewritePeephole
```

A.4.2 Five Hardest Alive Examples

```
SSA/Projects/InstCombine/AliveHandwrittenLargeExamples.lean
54:#guard_msgs in #print axioms alive_DivRemOfSelect
219:#guard_msgs in #print axioms alive_simplifyMulDivRem805
313:#guard_msgs in #print axioms alive_simplifyMulDivRem805'
373:#guard_msgs in #print axioms alive_simplifyMulDivRem290
432:#guard_msgs in #print axioms alive_simplifyAndOrXor2515
548:#guard_msgs in #print axioms alive_simplifySelect764
```

A.4.3 Example Rewrites Shown in the Paper

```
SSA/Projects/PaperExamples/PaperExamples.lean
136:#guard_msgs in #eval eg₀val
```

A.4.4 Examples from FHE and Scf dialects

```
SSA/Projects/FullyHomomorphicEncryption/Rewrites.lean
107:#guard_msgs in #print axioms lhs
153:#guard_msgs in #print axioms p1

SSA/Projects/Scf/ScfFunctor.lean
488:#guard_msgs in #print axioms correct
542:#guard_msgs in #print axioms correct
596:#guard_msgs in #print axioms correct
```