EECS 481 — Software Engineering

web.eecs.umich.edu/~weimerw/481/hw4.html

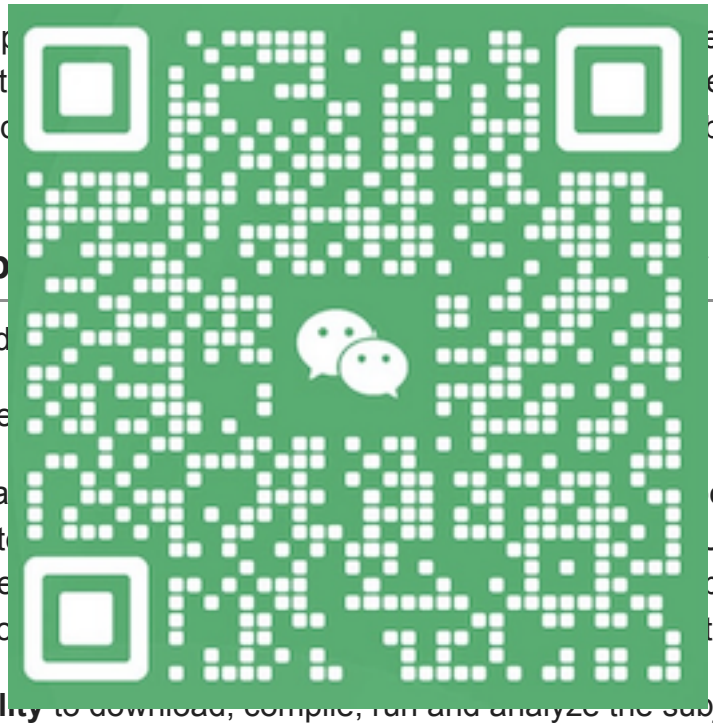
Homework Assignment #4 — Defect Detection

In this assignment you will use two different static analysis tools to automatically detect potential defects.

The first static analysis tool is GrammaTech's CodeSonar, which focuses on security issues, as well as memory, resource and concurrency defects. CodeSonar is a commercial tool used in activities such as DO-178B avionics certification; we have obtained an academic license for its use in this class.

The second static analysis tool is Facebook's Infer, which focuses on memory errors, leaks, race conditions, and API issues. Infer is open source.

You *may* work with a p e the same partner for all sub-component election feature. Only one partner need both do, nothing fatal happens.



Installing, Comp Code

Warning: Infer Is Hard

You should use the se

As an optional alterna ol does *not* run on the Windows Subsyst Ubuntu- or Linux-like interfaces. Headle ported to work very well. Officially, ho tion for the class.

It is **your responsibility** to download, compile, run and analyze the subject program and associated tools (or use the precompiled one: we recommend using the precompiled version since it is *known to work with the HW0 setup*). Getting the code and tools to work in some manner is **part of the assignment**. You can post on the forum for help and compare notes bemoaning various architectures (e.g., windows vs. mac vs. linux, etc.). Ultimately, however, it is your responsibility to read the documentation for these programs and tools and use some elbow grease to make them work.

The `lighttpd` webserver

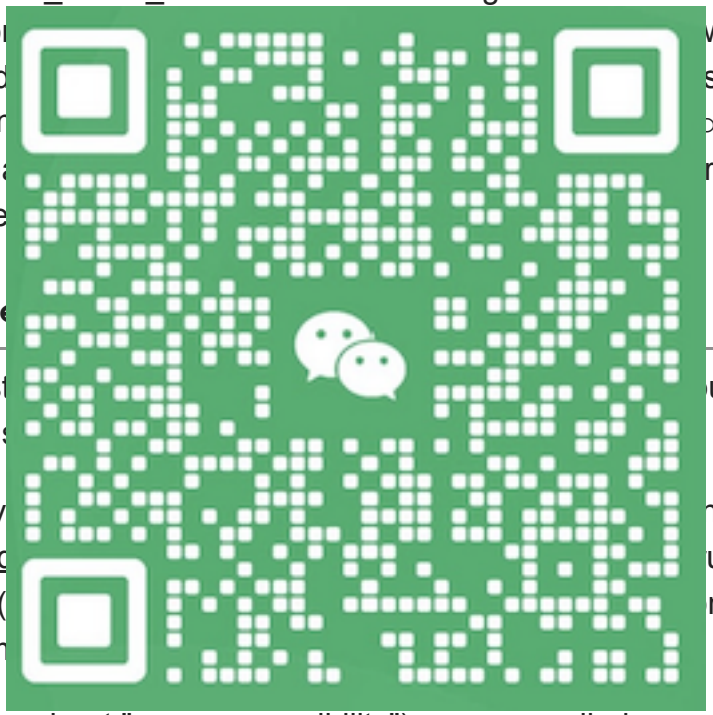
We will make use of the `lighttpd` webserver (pronounced "lighty"), version 1.4.17, as our primary subject program for this homework. A local mirror copy of `lighttpd-1.4.17.tar.gz` is available, but you can also get it from the original website. It is about

55,000 lines of code in about 90 files. While somewhat small for this class, some analysis tool licenses have LOC limits or scalability issues, so it was chosen as an indicative compromise.

While not as large or popular as `apache`, at various points `lighttpd` has been used by YouTube, xkcd and Wikimedia. Much like `apache`, old verisons of it have a number of known security vulnerabilities.

The Common Vulnerabilities and Exposures system is one approach for tracking security vulnerabilities. A CVE is basically a formal description, prepared by security experts, of a software bug that has security implications.

There are at least ten CVEs associated with `lighttpd` 1.4.17 tracked in various lists (such as cvedetails or mitre). For example, CVE-2014-2324 has the description "Multiple directory traversal vulnerabilities in (1) mod_evhost and (2) mod_simple_vhost in lighttpd before 1.4.35 allow remote attackers to read arbitrary files via a .. (dot dot) in the host name, related to request_check_hostname." You can dig into the information listed in, or linked from, a CVE (o where the bug is fixed!) to track down c st refers to source file `mod_evhost.c`, r ost.c, and request_check_hostn rmation when evaluating the whethe



Facebook's `Infe`

The `Infer` tool is a st ut running them. The primary website i

Unfortunately, some v nstall, despite their handy installation guic un on Windows Subsystem for Linux (nfiguration (instructions) is recom

Instead (but see above about "your responsibility"), a precompiled, *runs-on-the-HW0-setup* (`Ubuntu 16.04.2 LTS GNU/Linux 4.4.0-34-generic x86_64`) version of `Infer` is available locally here (warning: 265 MB; you will likely want to use `scp` to transfer the `.tar.gz` file to your HW0 setup and unpack it there). Once you have transferred and unpacked it, the main binary can be found at `infer-linux64-v0.13.0/infer/bin/infer`. You can use either the pre-compiled one or compile it yourself for full credit (any version at all of `Infer` is full credit).

While times will vary, some students have reported that running `Infer` on `jfreechart` can take five hours.

You can find `Infer`'s output in the `infer-out` folder.

GrammaTech's `CodeSonar`

GrammaTech's `CodeSonar` static analyzer is a commercial (not open source) tool for finding defects in program source code or binaries.

GrammaTech has generously provided an academic license so that students in this class can make use of their tool on a limited basis. This license includes a lines-of-code limit, so we have pre-run the analysis and made the results available for everyone to share (running it is very similar to Infer, see below). Analyzing additional code or trying to subvert this license runs the risk of ruining our relationship with that company and thus preventing me from giving future students this experience in subsequent semesters — doing so is thus a significant academic integrity violation.

`CodeSonar`'s output is designed to be *shared* among an organization's developers. As a result, the analysis is carried out once and then the reports are made available to everyone via a web interface. In this model the team might work together to triage and prioritize the defect reports, assigning some to one developer and some to another. We have already run `Cod`[redacted]`sults`.

Running `CodeSonar` [redacted]`d of running `make` one runs `codesonar` [redacted]`ut *you do **not** run them* (we have alread`[redacted]

```
$ cd lighttpd-1.4.17
$ sh configure
$ # DO NOT: codesona
```

Similarly, a Java proje[redacted]

```
$ cd jfreechart-1.5.
$ mvn compile
$ # DO NOT: codesona                                    scan
src/main/java/
```

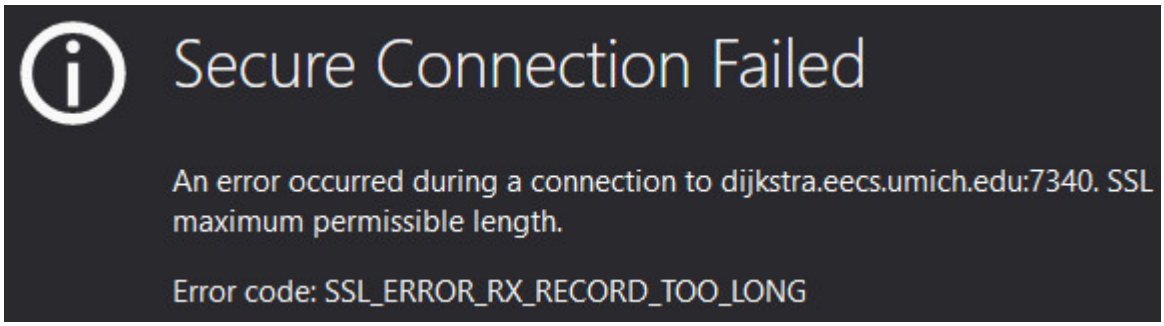
GrammaTech's `Co`[redacted]

Warning: CodeSonar Login

If you do not follow the CodeSonar login procedures properly, including having it email you a password, *it will appear to almost work but you won't be able to see any warnings*.

FAQ: Secure Connection Problem

If you receive a warning message like this:



Use one of these solutions to resolve it:

1. Carefully retype the URL so that it says `http` instead of `https` — **remove** the "s"! Even if you type http, some browsers will "correct" it to https.
2. Use Chrome instead of another browser like Firefox. (Some students report success with, horrors, Microsoft Edge.)

The `CodeSonar` reports can be found at this location — but you must connect with the UM VPN (students report Nord VPN does *not* work, etc.) or be on campus!. If you are registered with the co ssword", enter the username in the emai Login information will be emailed to you

For example, your ins tever name CodeSonar assigned; have an email with a set-your-passw

(The license agreeme ee the service.)

Additional Subje

We also make availab h as:

Note that the report re as one of those listed above) and ana

FAQ and Troubleshooting

In this section we detail previous student issues and resolutions:

1. **Question:** When I run `infer.exe run -- make` or `infer run -- mvn compile` I get errors like `InferModules__SqliteUtils.Error` or `Maven command failed`.

   Answer: The most common issue is that `Infer` does not always run well on Windows Subsystem for Linux (WSL) or similar shortcuts to get a Linux- or Ubuntu-like interface on another OS. We strongly recommend a headless Virtual Box setup (instructions).

2. **Question:** When I try to run Infer, I get `cannot execute binary file: Exec format error.`.

   Answer: One student reports: "Finally got it. Turns out I was using a 32 bit processor (i386) so even when I set up my vm as 64 bit, it couldn't run any x86-64 binaries. Fixed it by installing a 64 bit vdi. https://appuals.com/fix-cannot-execute-binary-file-exec-format-error-ubuntu/

3. **Question:** I see `Maven command failed: *** mvn compile -P infer-capture` when I try to run Infer.

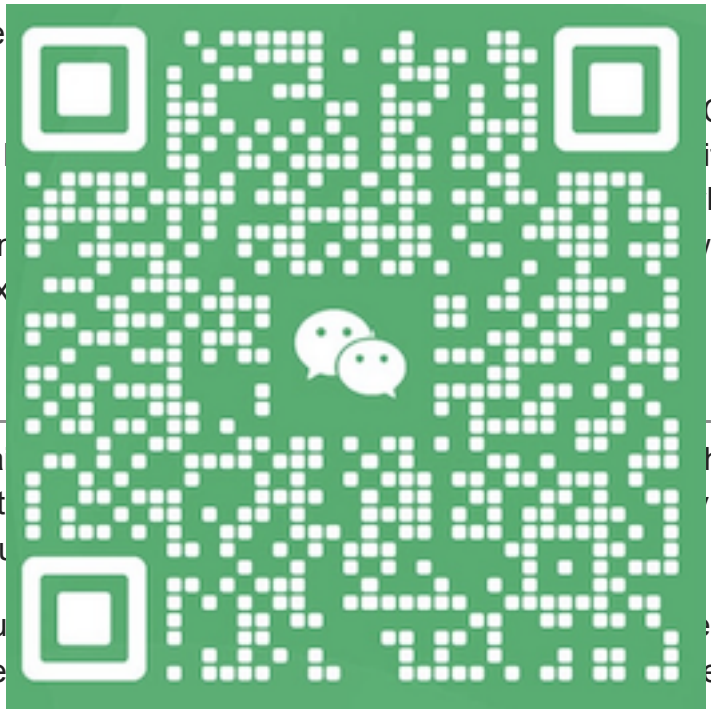
   Answer: Some students have seen success with:

   ```
   sudo apt-get install cobertura maven
   sudo apt-get install openjdk-8-jdk
   ```

   Others reported that "I ended up having to setup an Ubuntu 16.04 VM in VirtualBox".

4. **Question:** I see

   Answer: When CodeSonar allows you to navigate ity feature). You can resolve this hich involves having CodeSor arnings for `cpython`, for ex



Written Report

You must write a deta h these static analysis defect detect email address(es). In particu

- ([Reminder] You d with CodeSonar, eve eenshots from CodeSonar.)
- [Framing] Choose either "large software development organization" (e.g., the SQL Server group at Microsoft) or "small software development organization" (e.g., a dozen-person mobile app tech startup) — indicate your choice. You have been asked by your supervisor to evaluate these two tools and prepare a recommendation: which one, if any, should our organization use?
- [Setup] In a few sentences, describe your setup experiences with each applicable tool. (Yes, we know you did not directly set up CodeSonar.) This might include dependencies, installing it, runtime, etc.
  [1 point for description]

- [Usability] In a few sentences, compare and contrast your usability experiences with each tool. This might include locating the reports, navigating the report or documentation website, etc.
  [1 point for infer, 1 point for codesonar, 1 point for contrast, 1 point for other details]
- [Overall] Compare and contrast the quality and details of the reports generated by Infer and CodeSonar. At a high level, what did each tool do well? How might each tool be improved? Comment on defect report categorizations (e.g., Reliability, NULL_DEREFERENCE, Security, etc.). Did you observe any "duplicate" defect reports (i.e., the same underlying issue was reported in terms of multiple different symptoms) within the same tool? How much overlap did you observe between the issues reported by the two tools? What are the costs (in general, including developer time, monetary cost, risks, training, etc., and anything else mentioned at any point in class) associated with each tool?
  [4 points for infer, 4 points for codesonar, 1 point for categories, 1 point for duplicates, 1 point for overlap, 2 points for costs]
- [CVE] Choose two of the CVEs associated with `lighttpd`. For each tool, describe whether or not th E (or would otherwise have ch that at least one tool points o , separately, you should choose o E in some manner (if you fi finding security defects?

  Students a ment for this aspect of t u pick one CVE that Infer fi describe those, you can ge and one CVE that neither too edit. If you only pick one C will not get full credit. If yo ll credit (regretfully

  [2 point for each cv
- [lighttpd] Compare and contrast the defect reports produced by the tools for the `lighttpd` program. Which did you find more useful? Consider false positives, false negatives, and issues that you would consider to have high priority or severity. Include (copy-and-paste, screenshot, etc.) part of one report you found particularly noteworthy (good, bad, complex: your choice) and explain it.
  [3 point for compare/contrast, 1 point for inlined report and analysis, 2 point for other insights]
- [jfreechart] Compare and contrast the defect reports produced by the tools for the `jfreechart` program. (6 points, as above.)
- [additional] Choose an additional subject program. Compare and contrast the defect reports produced by the tools for that program. (6 points, as above.)