

example, they may simply review lab reports. Note this is not an issue when parameterization individualizes the expected result for each student.

Labtainers allows designers to include keyed hashes of answers within the published files rather than the answers themselves. The `hash_equal` operator used in a `goals.config` file functions like the `string_equal` operator, except the comparison is made on a hash of the named result value, generated using the Lab Master SEED as the key.

Instead of creating a `goals.config` file directly, the designer creates a `goals.answers` file that contains the intended content of the `goals.config` file, but with the actual answers, e.g.,:

```
ipv4_count = matchany : hash_equal : _ipv4_count : answer=2029121
```

The `hash-goals.py` utility is then used to generate the `goals.config` file, replacing the plain text answers with the appropriate hashes.

It is intended that the `goals.answers` files will not be distributed, e.g., they would be maintained with the SimLab solutions repo.

6.3.4 Assessment Report

Evaluation of student results occurs on a *grading* container that starts when the instructor runs the `gradelab <lab>` command. A report is generated and displayed on the screen. A copy of the report is also placed in the `latainer_xfer` directory. Debugging your assessment configuration can be aided by using `gradelab -d <lab>`, which will start the grading container and give you a shell into it. From there, run the script named `instructor.py`. There is a log in `/tmp/instructor.log` in addition to diagnostics that might be generated on the terminal. See section 6.7 for additional information on debugging grading.¹⁹ By convention, all goals and boolean results whose symbolic names are not prefaced with an underscore (`_`) or an `cw_` (see 6.4), will have corresponding entries in the assessment report, located in the home directory in a file named `<lab name>.grades.txt`

6.3.5 Document the meaning of goals

Instructors will see descriptions of lab goals when they start the lab using `gradelab`. These descriptions are embedded within directives within the `goals.config` and `results.config` files. The descriptions are associated with symbolic names that immediately follow the documentation directives as described below:

```
# SUM:  -- The remainder of the line and comment lines that immediately
          follow are displayed independent of any goal symbols.
# DOC:  -- The remainder of the line and comment lines that immediately
          follow are displayed for the symbolic name that follows
          the comment lines.
# GROUP:-- The remainder of the line and comment lines that immediately
          follow are displayed for the group symbolic names that
          follows the comment lines. The group ends on a blank line, or
          new comment.
```

See existing labs for examples. Also see the 6.4 for additional directives.

¹⁹Be sure to run `stoplab` after use of the `-d` option to shut down the grading container when you are done with it