Project 4 Specification Notes

<2015-04-09 Thu>

Contents

1 Dates

1.1 TODO Sprint #1

SCHEDULED: *<2015-04-06 Mon>–<2015-04-13 Mon>*
DEADLINE: *<2015-04-13 Mon 23:59>*

1.2 TODO Sprint #2

SCHEDULED: *<2015-04-13 Mon>–<2015-04-20 Mon>*
DEADLINE: *<2015-04-20 Mon 23:59>*

1.3 TODO Sprint #3

SCHEDULED: *<2015-04-20 Mon>–<2015-04-27 Mon>*
DEADLINE: *<2015-04-27 Mon 23:59>*

1.4 TODO Sprint #4

SCHEDULED: *<2015-04-27 Mon>–<2015-05-05 Tue>*
DEADLINE: *<2015-05-05 Tue 23:59>*

1.5 TODO All Deliverables

SCHEDULED: *<2015-05-05 Tue 23:59>*
DEADLINE: *<2015-05-05 Tue 23:59>*

1.6 TODO Final Presentation

SCHEDULED: *<2015-05-11 Mon 10:30>–<2015-05-11 Mon 12:30>*
DEADLINE: *<2015-05-11 Mon 10:30>–<2015-05-11 Mon 12:30>*

2 Design Choices

2.1 C++14

- Use `build.tamu.edu`
  - `gcc-4.9.2` (add `share/examples/bashrc` to your `~/.bashrc`)
  - linux `x86_64`

2.2 Big Integers

- We need to be able to store integers $\geq$ 4096 bits
- C++'s `uintmax_t` can only store integers $\leq$ (2^{64}-1) so we need a way of storing these big integers

2.2.1 The GNU Multiple Precision Arithmetic Library

2.3 Google Test

- TDD Proof
  - travis-ci build logs

2.4 travis-ci

- Requires committing and **pushing** [1] failing tests (so `travis-ci` builds and runs them)
- Gives us time-stamped builds and test runs for **every** commit and pull-request

2.5 git flow

- Makes following the git-flow branching model stupidly easy
- Use `feature/FEATURE_NAME` branches for new additions
  - keep these specific and small
- When a feature is done, create a pull request
  - allows travis to test if your branch builds
  - allows the rest of the group to discuss the feature

[1] Shouldn't be a problem except for Chris

3 Deliverables

3.1 `munchkinsteg`

- embed ciphertext into bmp
- extract ciphertext from bmp
- **MUST** support *at-least* 1-LSB and 2-LSB, but may include other modes
- report PSNR (peak signal to noise ratio) of stego-image

3.2 `toto`

- implements *at-least* 3 attacks on LSB image steganography systems

4 Requirements

4.1 Workflows

4.1.1 Agile

- 1-week sprints
  - At beginning of each week:
    - Choose features from product back-log to include in this sprint
  - At end of each week:
    - Unfinished tasks go back in back-log
    - Demonstrate sprint's result to TA
    - Submit to `CSNet`:
      - Backlogs
      - Burn-down Charts
      - Sprint Status Charts
- 4 scrums/week
  - Ask each group member (and record):
    1. "What have you done since last scrum meeting?"
    2. "What has impeded your work?"
    3. "What do you plan on doing between now and next scrum?"
  - At end of meeting:
    - Each team-member should update burn-down chart:
      - remaining effort for each task
      - status of tasks
- As soon as product is finished submit to `CSNet`

4.1.2 TDD

- Provide proof

4.2 Features

4.2.1 Encryption

- Encrypt to cipher-text
- Embed cipher-text in a `.bmp` image

4.2.2 Decryption

4.2.3 Crack