Secure Software Development

Assignment D3: Defensive Reporting

Ehrenreich, Hadzic, Lamster, Nageler, Schwarzl, Weiser

11.11.2020

Winter 2020/21, www.iaik.tugraz.at/ssd

Timeline

- Deadline D3: **8.1.2021 - 23:59:59**
- Points: 15% (+ 5% bonus)

Since you're now almost a security expert ...

Rumors ...

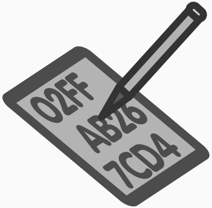
Since you’re now almost a security expert …

Since you're now almost a security expert ...

... let'z get in the shoes of a **software auditor**

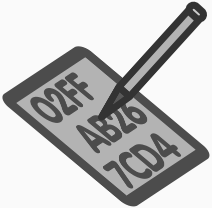
I DON'T CARE YOUR SSD SKILLS
I WILL FIND YOU, AND AUDIT YOU
imgflip.com

Code audit



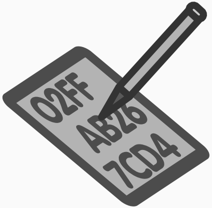
- You get an existing code base to audit

Code audit



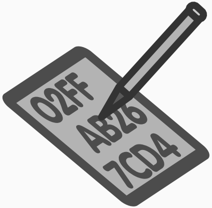
- You get an existing code base to audit
- You analyze it and report issues

Code audit



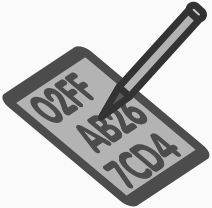
- You get an existing code base to audit
- You analyze it and report issues
- Our bug bounty program rewards you with points

Specification



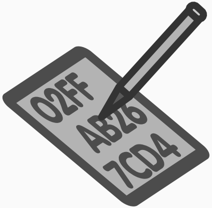
- Trudel - simple web server written in C

Specification



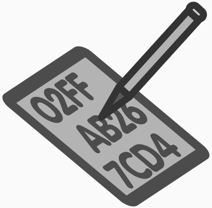
- Trudel - simple web server written in C
- Specification available as python web server

Specification



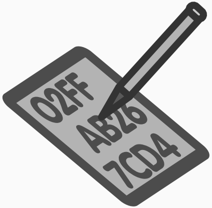
- Trudel - simple web server written in C
- Specification available as python web server
  - Start python web server inside Docker

Specification



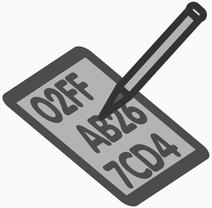
- Trudel - simple web server written in C
- Specification available as python web server
  - Start python web server inside Docker
  - Access Trudel at: `http://127.0.0.1:5000/`

Specification



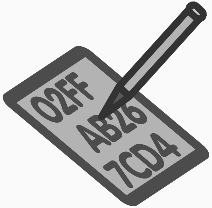
- Trudel - simple web server written in C
- Specification available as python web server
  - Start python web server inside Docker
  - Access Trudel at: `http://127.0.0.1:5000/`
  - Specification: `http://127.0.0.1:5000/swaggerui`

Specification

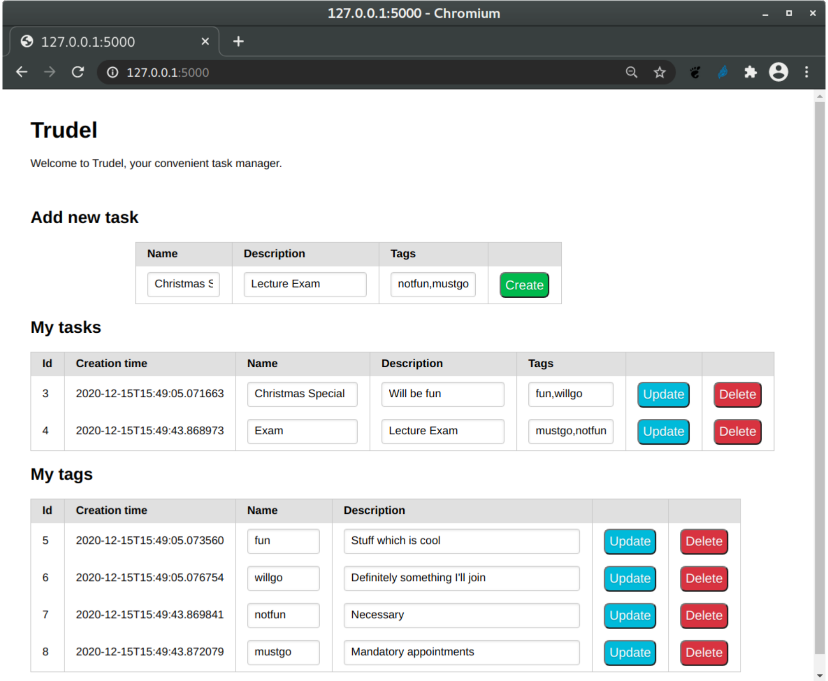


- Trudel - simple web server written in C
- Specification available as python web server
  - Start python web server inside Docker
  - Access Trudel at: `http://127.0.0.1:5000/`
  - Specification: `http://127.0.0.1:5000/swaggerui`
  - Specification: `http://127.0.0.1:5000/redoc`

Specification



- Trudel - simple web server written in C
- Specification available as python web server
  - Start python web server inside Docker
  - Access Trudel at: `http://127.0.0.1:5000/`
  - Specification: `http://127.0.0.1:5000/swaggerui`
  - Specification: `http://127.0.0.1:5000/redoc`
- You do not need to fully understand the spec in order to find some bugs



Trudel

Welcome to Trudel, your convenient task manager.

Add new task

Name	Description	Tags	
Christmas S	Lecture Exam	notfun,mustgo	Create

My tasks

Id	Creation time	Name	Description	Tags		
3	2020-12-15T15:49:05.071663	Christmas Special	Will be fun	fun,willgo	Update	Delete
4	2020-12-15T15:49:43.868973	Exam	Lecture Exam	mustgo,notfun	Update	Delete

My tags

Id	Creation time	Name	Description		
5	2020-12-15T15:49:05.073560	fun	Stuff which is cool	Update	Delete
6	2020-12-15T15:49:05.076754	willgo	Definitely something I'll join	Update	Delete
7	2020-12-15T15:49:43.869841	notfun	Necessary	Update	Delete
8	2020-12-15T15:49:43.872079	mustgo	Mandatory appointments	Update	Delete

Which Issues to Report?

- **Bugs & Vulnerabilities:** Memory corruptions and leaks, use after free, double free, use of uninitialized memory, format string vulnerability, integer overflow ...

Which Issues to Report?

- **Bugs & Vulnerabilities:** Memory corruptions and leaks, use after free, double free, use of uninitialized memory, format string vulnerability, integer overflow ...
- **Violations of defensive coding principles:** Hard-to-read and dangerous code, obviously bad naming, wrong use of asserts, implicit assumptions (undefined behavior, implementation-defined behavior)

Which Issues to Report?

- **Bugs & Vulnerabilities:** Memory corruptions and leaks, use after free, double free, use of uninitialized memory, format string vulnerability, integer overflow ...
- **Violations of defensive coding principles:** Hard-to-read and dangerous code, obviously bad naming, wrong use of asserts, implicit assumptions (undefined behavior, implementation-defined behavior)
- **Other issues:** all sorts of programming mistakes leading to

Which Issues to Report?

- **Bugs & Vulnerabilities:** Memory corruptions and leaks, use after free, double free, use of uninitialized memory, format string vulnerability, integer overflow ...
- **Violations of defensive coding principles:** Hard-to-read and dangerous code, obviously bad naming, wrong use of asserts, implicit assumptions (undefined behavior, implementation-defined behavior)
- **Other issues:** all sorts of programming mistakes leading to
  - Functional issues

Which Issues to Report?

- **Bugs & Vulnerabilities:** Memory corruptions and leaks, use after free, double free, use of uninitialized memory, format string vulnerability, integer overflow ...
- **Violations of defensive coding principles:** Hard-to-read and dangerous code, obviously bad naming, wrong use of asserts, implicit assumptions (undefined behavior, implementation-defined behavior)
- **Other issues:** all sorts of programming mistakes leading to
  - Functional issues
  - Program crash (segfault)

Which Issues to Report?

- **Bugs & Vulnerabilities:** Memory corruptions and leaks, use after free, double free, use of uninitialized memory, format string vulnerability, integer overflow ...
- **Violations of defensive coding principles:** Hard-to-read and dangerous code, obviously bad naming, wrong use of asserts, implicit assumptions (undefined behavior, implementation-defined behavior)
- **Other issues:** all sorts of programming mistakes leading to
  - Functional issues
  - Program crash (segfault)
  - Information leakage

Which Issues to Report?

- **Bugs & Vulnerabilities:** Memory corruptions and leaks, use after free, double free, use of uninitialized memory, format string vulnerability, integer overflow ...
- **Violations of defensive coding principles:** Hard-to-read and dangerous code, obviously bad naming, wrong use of asserts, implicit assumptions (undefined behavior, implementation-defined behavior)
- **Other issues:** all sorts of programming mistakes leading to
    - Functional issues
    - Program crash (segfault)
    - Information leakage
    - Errors/Warnings triggered by valgrind, asan, `-Wall`

Which Issues to Report?

- **Bugs & Vulnerabilities:** Memory corruptions and leaks, use after free, double free, use of uninitialized memory, format string vulnerability, integer overflow ...
- **Violations of defensive coding principles:** Hard-to-read and dangerous code, obviously bad naming, wrong use of asserts, implicit assumptions (undefined behavior, implementation-defined behavior)
- **Other issues:** all sorts of programming mistakes leading to
  - Functional issues
  - Program crash (segfault)
  - Information leakage
  - Errors/Warnings triggered by valgrind, asan, `-Wall`
- Missing documentation is **not** an issue you get points for

Getting points

- Report issues via Gitlab issue tracker

Getting points

- Report issues via Gitlab issue tracker
- Max. 4 points per issue

Getting points

- Report issues via Gitlab issue tracker
- Max. 4 points per issue
  - Up to 12 issues: max. 48 points (15%)

Getting points

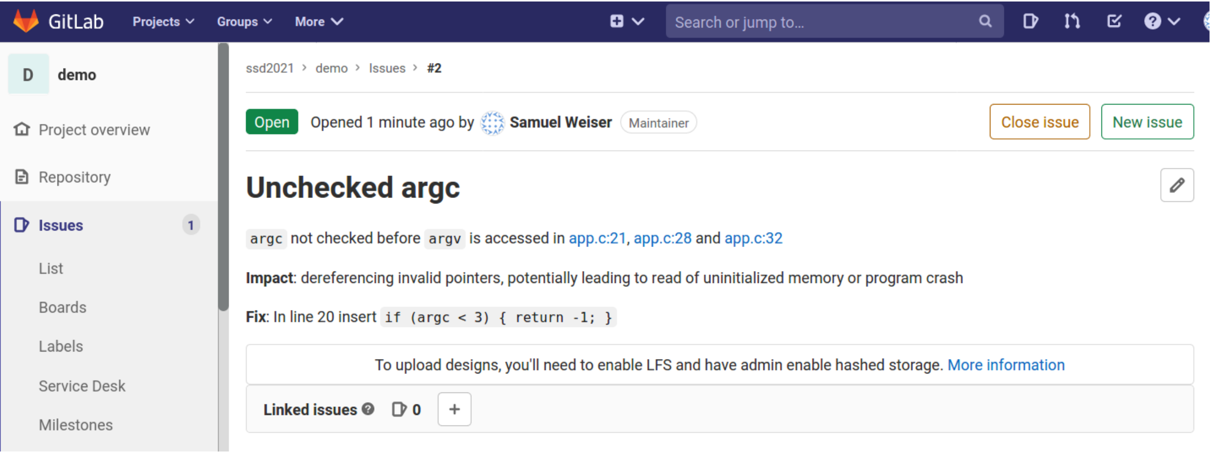
- Report issues via Gitlab issue tracker
- Max. 4 points per issue
  - Up to 12 issues: max. 48 points (15%)
  - Bonus: up to 15 issues, max. 48+12 points (15+5%)

Getting points

- Report issues via Gitlab issue tracker
- Max. 4 points per issue
  - Up to 12 issues: max. 48 points (15%)
  - Bonus: up to 15 issues, max. 48+12 points (15+5%)
  - If you report more than 15 issues, we only count the first 15 ones

Getting points

- Report issues via Gitlab issue tracker
- Max. 4 points per issue
  - Up to 12 issues: max. 48 points (15%)
  - Bonus: up to 15 issues, max. 48+12 points (15+5%)
  - If you report more than 15 issues, we only count the first 15 ones
  - We only count **open** issues. If you close an issue, we won't count it.



Opened 1 minute ago by Samuel Weiser Maintainer

Unchecked argc

`argc` not checked before `argv` is accessed in app.c:21, app.c:28 and app.c:32

Impact: dereferencing invalid pointers, potentially leading to read of uninitialized memory or program crash

Fix: In line 20 insert `if (argc < 3) { return -1; }`

To upload designs, you'll need to enable LFS and have admin enable hashed storage. More information

Rumors continue ...

Getting points – Criteria

- If criteria are not fulfilled → point deductions

Getting points – Criteria

- If criteria are not fulfilled → point deductions
- **Valid and Unique**: Duplicates only count once

Getting points – Criteria

- If criteria are not fulfilled → point deductions
- **Valid and Unique**: Duplicates only count once
  - Link **all** affected code locations in the same issue

Getting points – Criteria

- If criteria are not fulfilled → point deductions
- **Valid and Unique**: Duplicates only count once
    - Link **all** affected code locations in the same issue
- **Impact**: Nature of the issue, potential consequences

Getting points – Criteria

- If criteria are not fulfilled → point deductions
- **Valid and Unique**: Duplicates only count once
    - Link **all** affected code locations in the same issue
- **Impact**: Nature of the issue, potential consequences
    - E.g.: Integer overflow leading to memory corruption

Getting points – Criteria

- If criteria are not fulfilled → point deductions
- **Valid and Unique**: Duplicates only count once
  - Link **all** affected code locations in the same issue
- **Impact**: Nature of the issue, potential consequences
  - E.g.: Integer overflow leading to memory corruption
- **Concise**: Accurate issue description no longer than 100 words

Getting points – Criteria

- If criteria are not fulfilled → point deductions
- **Valid and Unique**: Duplicates only count once
  - Link **all** affected code locations in the same issue
- **Impact**: Nature of the issue, potential consequences
  - E.g.: Integer overflow leading to memory corruption
- **Concise**: Accurate issue description no longer than 100 words
  - E.g.: Hide URL behind markdown link → counts as one word

Getting points – Criteria

- If criteria are not fulfilled → point deductions
- **Valid and Unique**: Duplicates only count once
  - Link **all** affected code locations in the same issue
- **Impact**: Nature of the issue, potential consequences
  - E.g.: Integer overflow leading to memory corruption
- **Concise**: Accurate issue description no longer than 100 words
  - E.g.: Hide URL behind markdown link → counts as one word
  - Follow-up comments not needed. Attachments not needed.

Getting points – Criteria

- If criteria are not fulfilled → point deductions
- **Valid and Unique**: Duplicates only count once
  - Link **all** affected code locations in the same issue
- **Impact**: Nature of the issue, potential consequences
  - E.g.: Integer overflow leading to memory corruption
- **Concise**: Accurate issue description no longer than 100 words
  - E.g.: Hide URL behind markdown link → counts as one word
  - Follow-up comments not needed. Attachments not needed.
- **Propose a fix**

Getting points – Criteria

- If criteria are not fulfilled → point deductions
- **Valid and Unique**: Duplicates only count once
    - Link **all** affected code locations in the same issue
- **Impact**: Nature of the issue, potential consequences
    - E.g.: Integer overflow leading to memory corruption
- **Concise**: Accurate issue description no longer than 100 words
    - E.g.: Hide URL behind markdown link → counts as one word
    - Follow-up comments not needed. Attachments not needed.
- **Propose a fix**
    - Small fixes → Issue description

Getting points – Criteria

- If criteria are not fulfilled → point deductions
- **Valid and Unique**: Duplicates only count once
  - Link **all** affected code locations in the same issue
- **Impact**: Nature of the issue, potential consequences
  - E.g.: Integer overflow leading to memory corruption
- **Concise**: Accurate issue description no longer than 100 words
  - E.g.: Hide URL behind markdown link → counts as one word
  - Follow-up comments not needed. Attachments not needed.
- **Propose a fix**
  - Small fixes → Issue description
  - More complex fixes → Create new branch per issue and commit fix there, link it in issue

Live Demo

Any Questions?

Ducks ’n awards