Secure Software Development – Seminar 3: Cryptography, Programming and Testing

Outline:

- Announcements
- Cryptography
- Seminar 3 Questions
- What is a Secure Programming Language?
- What testing tools should be used?

Announcements:

- Easter Holiday: 15.04.2022 – 18.04.2022
- No Office on Friday
- Office is Thursday 14.04.2022

Cryptography:

- Obscuring data by changing / encoding data
- Compare with Staganography → Kind of cryptography. It is the attempt to hiding information inside of pictures.
- Simplest cryptographic technique – Caesar's ciper
- Variation: ROT 13 → Kind of a Caesar cipher, rotate by 13 letters.
- ➔ Problem with symmetric Encryption: How to get the key secure to the destination (exchange)?
- ➔ Does the approaches meet the GDPR requirements? → Yes, it follows the CIA concept and follows the GDPR. But something to keep in mind are threats like **klickjacking**.
- ➔ Exploits happened like solarwinds, where databases were occupied by malicious, which creates a backdoor to the system. → APT: advanced persistent threat.

Seminar 3 – Reading & Questions

What did the solution mitigate?

- Ownership / Garbage Collector → Deals with buffer overflow errors. By trying to achieve a buffer overflow error the goal is to overflow the buffer an in this way to get access to the system by creating new memory spaces (for example in old C versions).
- Taint tracking → It internally marks data to get rid of the injection error. By using a part of a Software it gets checked and flagged.
- Faceted data → Deals with data leak. The idea is that the data is in different way access able by different users. Each user can only see the allowed information.

- What is a Secure Programming Language?
- Could Python be classed as a secure language? Justify your answer:
  → Question of the definition. But by being aware of the threats it is possible to create a secure code. As well it is as secure like Ruby and in some terms as Rust.
- Python would be a better language to crate operating systems than C. Discuss:
  - ➔ In terms of Security it is better. But Python is a high level type of programming language, and C is a middle level programming language. So a OS in Python would be slow. A better attempt would be to use Rust for an OS because it covers both aspects.

Testing:

- What kind of testing tools are available?
- → Unit: Helps at the class / function level. TDD – test driven development. If the code does not meet the function of the class then the unit test fails.
- → Integration: If classes are putting together.
- → System: Tests the hole system. Performing testing.
- → Linters: Check the design of the code.
- Assignment needs test evidence – record outputs and also test failures. How did you resolve them?

The Code Review Pyramid:

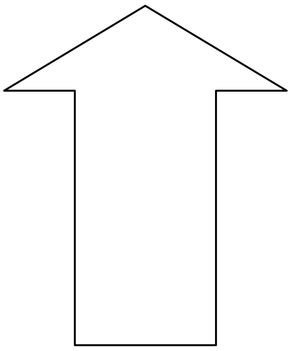
Code style

Tests

Documentation

Implementation Semantics

API Semantics



Gunnar Morling: https://www.mrling.dev/blog/the-code-review-pyramid

Reflection:

Aim to achieve:

- ■ Analysis of choice
- ■ Multiple perspectives
- ■ Critique of artefacts
- ■ Critique literature/action plan
  - ➔ 1-2 Thoughts

Next Week:

- No Seminar
- Open Office on Friday next 8.04.2022
- Review Reflection grid
- Remember the checklist – make sure you tick each requirement!