Lykke.Business

Preparing for Smart Contract Code Audit

Yury Glushkov, Head of Blockchain

February 2, 2020

Psychological Aspect

Lykke.Business

Audit can be uncomfortable and stressful

BUT

Auditor is not your foe

Uncertainty in your skills is not the reason of the audit

Successful audit, in reality, reduces the level of your responsibility (but you should still write your code in a way if you are holding the full responsibility for results)

Specification

Lykke.Business

Specification is mandatory for smart contracts

It is helpful if you write it in simple and understandable English

Smart Contracts methods should be treated as “black boxes”
with a description of what goes in, what goes out,
what should happen, what should not, and **why**

Diagrams can simplify the audit process greatly

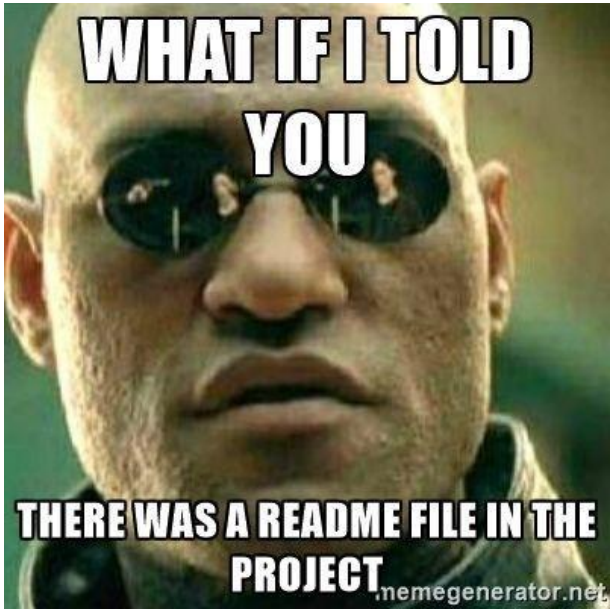
Your auditor is not Sherlock, really

README.md

Lykke.Business

Include the following information into README.md:

1. How to build smart contract locally
2. How to run a test suite
3. Inheritance graph description
4. Links to external resources that can help understand the project

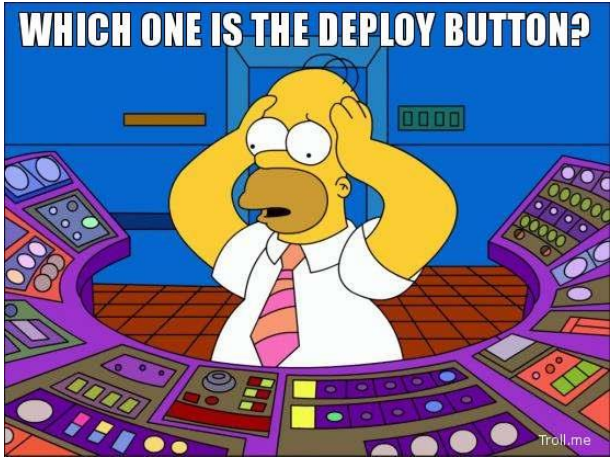


Deployment process

Lykke.Business

You should document deployment process well, so include:

1. Versions of compiler, deployment utilities, etc.
2. Order of deployment
3. Constructor and initialization parameters
4. The whole list of sacral knowledge that is important for deployment



Code Quality

Lykke.Business

Check if smarts contracts code:

- is written according to best practices
- is written according to official style guides
- checked with linters
- checked with existing security tools and analyzers
- contains comments for all not-straightforward logic

One hundred (100) percent test coverage

Sorry

Ice Age

Lykke.Business

Code base should be freezed

All dependencies versions should be freezed also

Have a Happy Audit!