Tezos Smart contract Attack Vectors (Mindmap by @Sm4rty_)

1. Using source instead of sender for authentication
2. Transferring tez in a call that should benefit others
3. Performing unlimited computations
4. Needlessly relying on one entity to perform a step
5. Trusting signed data without preventing wrongful uses
6. Not protecting against bots (BPEV attacks)
7. Using unreliable sources of randomness
8. Using computations that cause tez overflows
9. Contract failures due to rounding issues
10. Re-entrancy flaws
11. Unsafe use of Oracles
12. Forgetting to add an entry point to extract funds
13. Calling upgradable contracts
14. Misunderstanding the API of a contract

Source: OpenTezos