CST Part IA: Numerical Method, SV 1
Joe Yan
2017-5-11

You are building a mobile phone app for KuDoS.

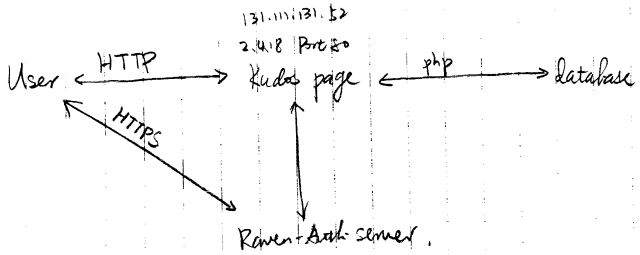
- Identify the subjects, people and principals involved in the system.
  A subject is a physical person involved in the project. E.g. A student or a supervisor under the meaning of a physical or natural person using the Kudos are all subjects to the Kudos.
  A person is a natural or legal person. Other than a subject which just means a natural person, a person can also be the department, company or other entities which has legal rights and obligations. E.g. The Churchill College can be a person of Kudos because it has the obligation to provide supervision rooms.
  A principle is a person, a (complex) role, an equipment, a communication channel or a compound of other principles.
  A role is an identity of someone or something in the project. E.g. A physical person can act as either a developer or a supervision to Kudos. It is possible that the same physical person has different roles.
  An complex role is the compound of multiple roles.
  An equipment can be the server Kudos running on or a printer which prints supervision work.
  A communication channel is the channel which the project uses to pass the information. E.g. the Internet the a communication channel for Kudos, the supervision booking information is passed on it.

- Where are there obligations of confidentiality?
  The obligation of confidentiality is the action of legally or morally protection to the secrets of someone else.
  E.g. Students have the obligation not to spread the material provided by the supervisor without permission. Supervisors have the obligation not to break the past paper answer policy to provide the student the answer directly which they do not have the right to access.

- Give examples of what might constitute each of the following in an implemented system: error, failure, accident, hazard, risk, safety.
  An error is a design flaw or a deviation from the intended state. E.g. If the design ignore the consistency issue when two students send two colliding booking requests to the system at the same moment, it is possible that the system sends two booking conformation email but actually only one booking was successful.
  A failure is a non-performance of the system. E.g. If the raven login authorisation server is down for some reason, it will cause a failure for student users which gives no respond for a login request. Student users now can not book or review their supervision information.
  An accident is an undesired, unplanned event resulting in a loss. E.g. A malicious software successfully format the disk of the supervision booking information. If there are no backup, this will cause a permanent booking information loss, students can only rebook everything.
  A hazard is a set of conditions on the system and the environment which can lead to a accident in the event of failure. E.g. The server which is running Kudos and supporting SSH use a weak password instead of set a key for remote login. There is a potential that

someone may crack in the server and do something undesired.
Risk is the probability of an accident combined with its danger and duration. Shortly, it is the expectation of the damage may happen to the system.
Safety is the freedom from accidents.

- Provide a threat model and a fault-tree analysis.



The connection between users and the Kudos page is not encrypted which may leak booking information.
The authorisation is based on the user device cookie which may allow a illegal login by cookie leak.
The *php*? request is used which can be potentially abused. E.g. php?/etc/passwd

  - Human failure
    - weak password or leaked password
    - forget to logout on a public device
    - print a very long fake pdf from a April Fool joke (unlikely)
  - Protocol failure
    - HTTP not encrypted
    - php? request abused
    - Other potential internet attack..
  - Device failure
    - the server is physically damaged
    - printer out of ink
  - Outside support failure
    - mid-man attack by weak college route protection
    - raven authorisation server down

- Give an examples for each of decoupling policy from mechanism; and defence in depth.
Decoupling policy from mechanism means separation of requirement and the technical implementation.
E.g. A security mechanism will set up the access control and then the specific student user can book supervisions in which group. However, a policy tells which user should be able to book supervisions.
Defence in depth is based on cheese model which means failure can only happen when a series of failures at the same point in all defence layers happen at the same time. This greatly reduces the probability of failure comparing with only one layer of defence.
E.g. A student can login by raven account this is one layer of defence. It is possible to set up a mechanism if the account is logged in from a suspicious IP address then Kudos will require a conformation message from a email to a student's personal email address (not using the university email because it is also based on raven). This is a two layers defence to a illegal login.