**PROBLEMATICHE DA AFFRONTARE**

* **Importanti**
  + ~~Diagrammi UML da fare~~
    - ~~Dei componenti (da aggiustare)~~
  + ~~protezione attributi utilizzati per l'autorizzazione~~
  + ~~protezione del ruolo di amministratore~~
  + separation of duties per i ruoli
    - amministratore de che?
    - Più ruoli privilegiati? (amministratore che entra sulla web app, gestore db, gestore account)
  + Distinzione nel DB tra utente amministratore e “utente” per la connessione con la webapp
    - Avranno permessi diversi? (query GRANT etc.)
  + Aggiungere limitazioni per indirizzi IP per l’accesso al DB
  + ~~Sicurezza sessione (contiene l’attributo id\_utente e isAdmin, anche gli oggetti visualizzati)~~
  + ~~AAServices nel Control (diagramma package)~~
  + ~~Mettere nel powerpoint la scadenza del token~~
* **Se rimane tempo**
  + Questione dell’accountability (INSERIRE LOGGING DELL’ACCESS PROXY)
  + ROBUSTEZZA PASSWORD AUTH0?

**STRUTTURA DEL PPT**

Problem Statement / Requisiti di alto livello

Architettura della soluzione = diagramma dei connettori / architettura logica / deploy?

Sintesi a punti dei principali ambiti di sicurezza affrontati

Identification ed Authentication

* roba teorica = token etc
* il sequence di autenticazione con le servlet

Authorization

* Risorse da proteggere
* Che tipo di autorizzazione è
* Sequence con il flusso di autorizzazione
* File delle policy (e hashing)
* DBAttributeFinder?

Database Access

* Come ci si autentica al database
* Come proteggiamo la comunicazione con il DB
* Come proteggiamo il DB (cifratura del DB)?

Assessment Sicurezza

[riccardo]

Req di alto livello

Nist

Parte di deploy

Threat modeling

Pen testing nmap