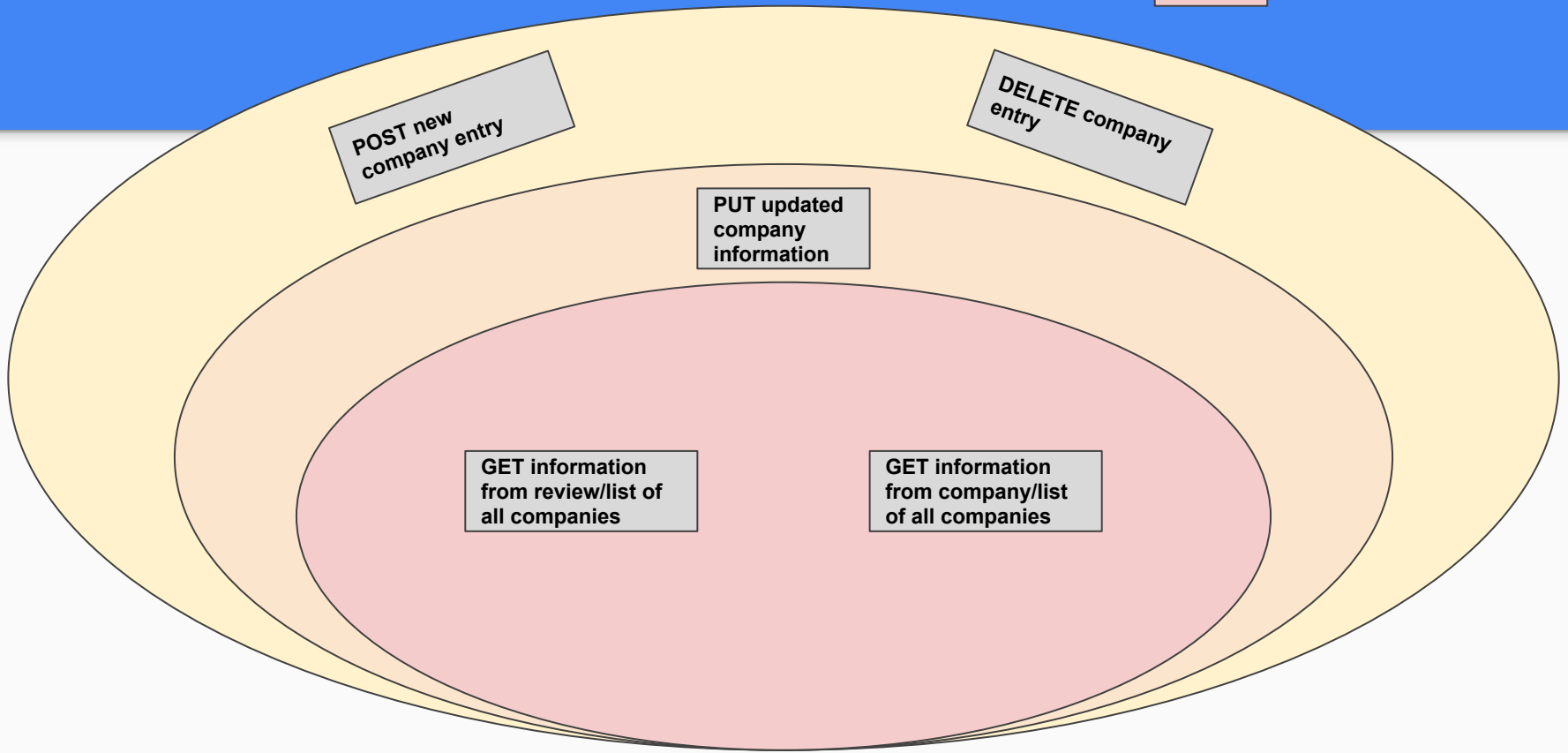
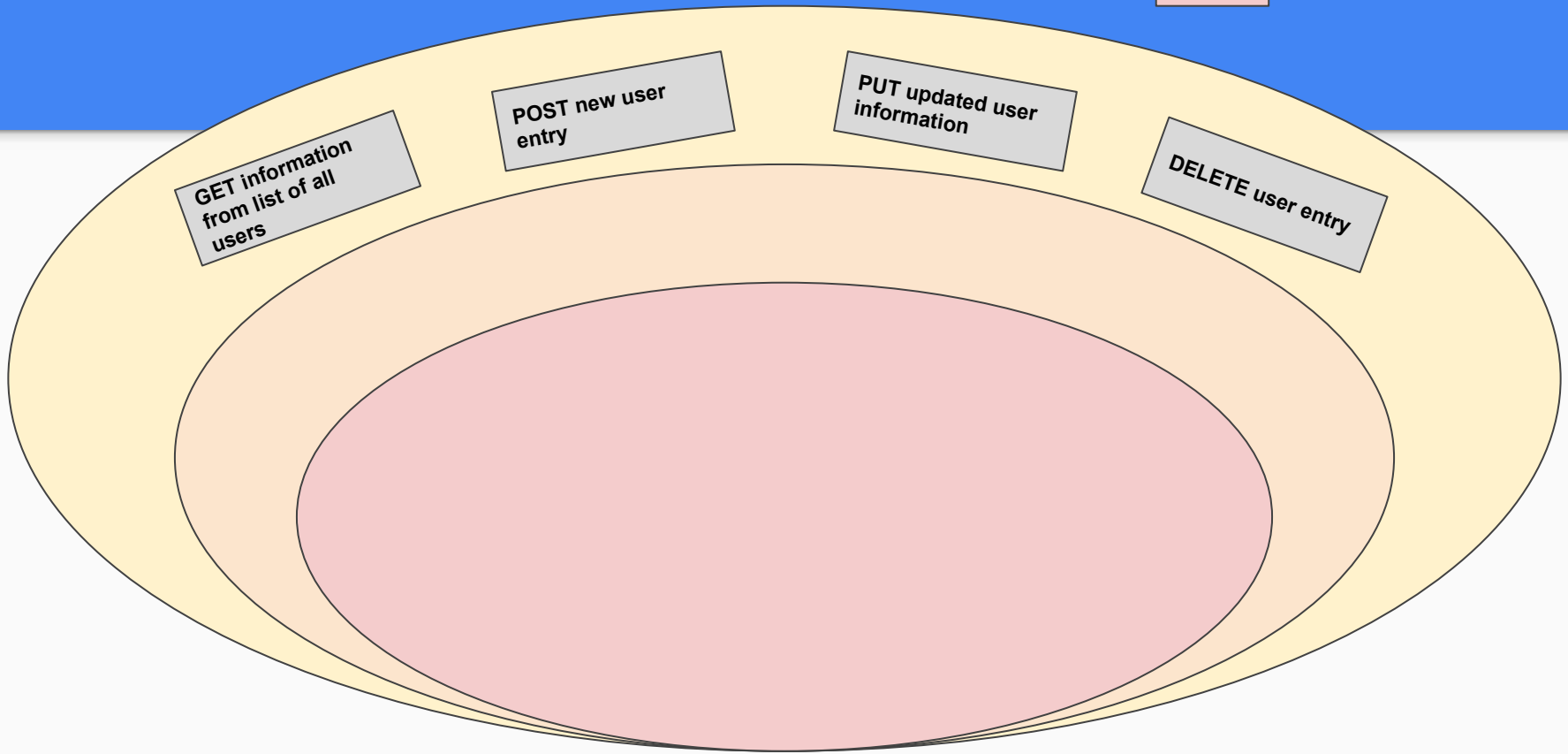
Week 4 task: Security and Access Control

Made by: Jere Pakkanen, Samu Peltonen, Miikka Ruohoniemi

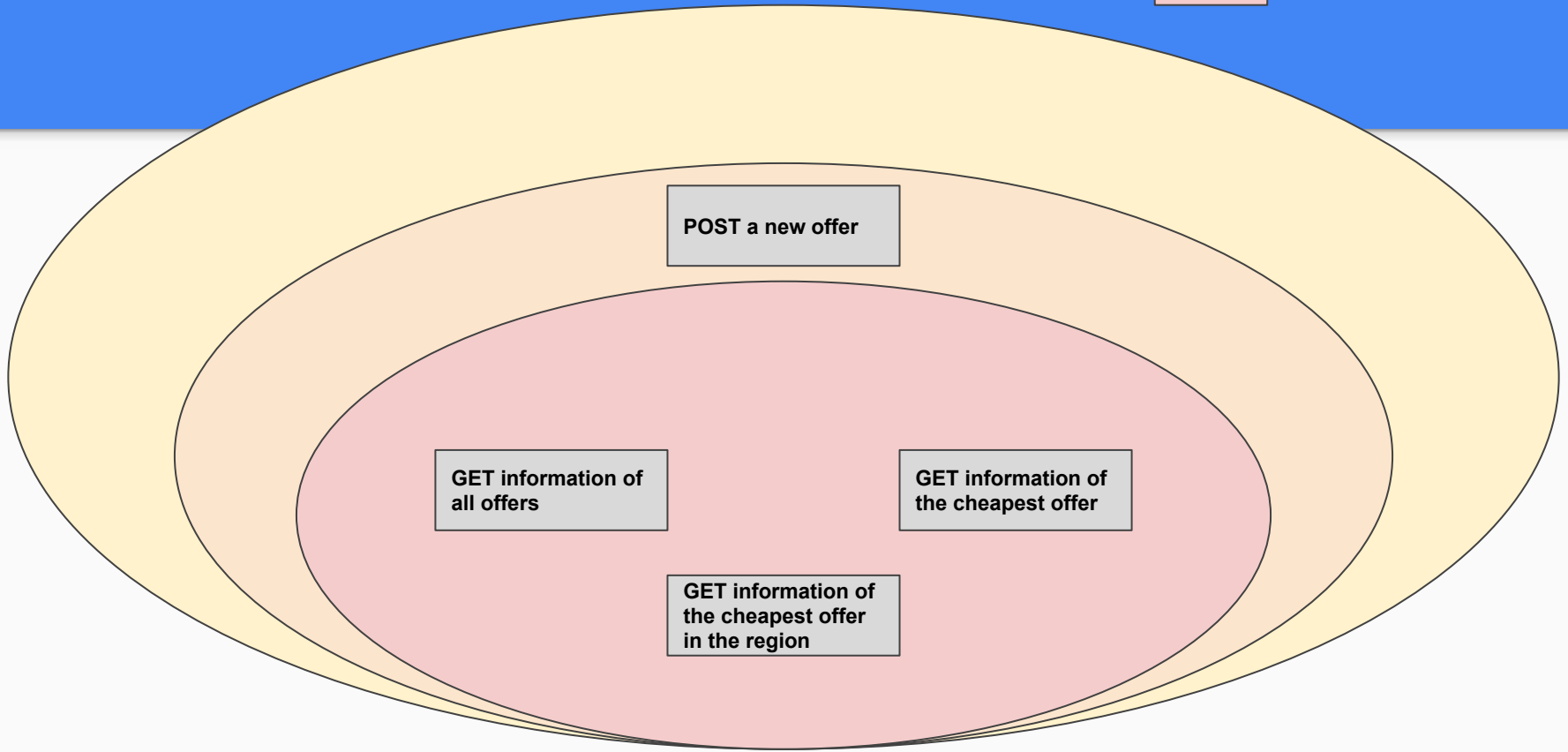
Company resource access control



User resource access control



Offer resource access control



Basic authentication description

- With every request (that is determined to require access checking), username and password are sent in the request header in plain text.
- Database checks if the credentials are valid, and if there is a user in the mock database, that corresponds to the username and password.
- If the database has the said user, that users role list is checked, if it has any roles, that are allowed to access the request. If there is at least one role that has the authority to access the request, server then proceeds to process the request.

Digest access authentication description

- If the authorization header starts with something else than “Basic” we use digest access authentication.
- We get username, realm, nonce, cnonce, cn, qop, response, **uri and opaque** in the header(bolded ones are not used in the server side).
- We get the password that corresponds the username from the database, if there is one. We then use this password and these bits of information to make a hash (in accordance to wikipedia article).
- We then proceed to check if the response is the same as the newly generated hash. If yes, we check if the user has the right to access the requested content. If again yes, we proceed with the request.

Problems

- Request is not determined dynamically. Only GET:/task4/webapi/users is usable by digest.
- Nonce etc. authentication header variables are not determined/checked server side, they just use client provided values to make the hash.
- Opaque is not checked.