ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

ΕΡΓΑΣΙΑ

Εργασία ανάπτυξης κώδικα

[Παράδοση 25/6/23]

Τίμια Ψηφιακά Ζάρια.

Δύο διαδικτυακοί παίκτες (ένας παίκτης και ο server) παίζουν ζάρια (από ένα ο καθένας). Ο μεγαλύτερος κερδίζει. Στην ισοπαλία δεν κερδίζει κανείς

[40%] Σχεδιάστε και υλοποιείστε ένα κρυπτογραφικό πρωτόκολλο που θα αποτρέπει ανέντιμη συμπεριφορά μεταξύ των παικτών και θα εξασφαλίζει το σωστό νικητή. Στην υλοποίηση: παράγεται από κάθε μέρος και εμφανίζεται στην διεπαφή χρήστη πρώτα το αποτέλεσμα της ρίψης. Κατόπιν εκτελείται το κρυπτογραφικό πρωτόκολλο. Κάθε μέρος με βάση το πρωτόκολλο εμφανίζει στην διεπαφή χρήστη αν είναι νικητής ή όχι.

[30%] Προσθέστε αυθεντικοποίηση server και από άκρο σε άκρο κρυπτογράφηση Server Authentication χρησιμοποιώντας το OpenSSL δημιουργήστε Certificate Authority (CA), CSR και ένα SSL certificate. Παραμετροποιήστε τον Apache προκειμένου να σερβίρει το πιστοποιητικό σας σε https και να ανακατευθύνει τα http σε https. Βεβαιωθείτε ότι το όλο το SSL certificate chain εμφανίζεται σωστά.

[30%] Επεκτείνετε την αυθεντικοποίηση και εισάγετε φόρμα εγγραφής χρήστη (όνομα, επίθετο, usename και password) και φόρμα login με username και password. Δημιουργήσετε σχεσιακή βάση δεδομένων με όνομα GDPR σε σύστημα ΒΔ δεδομένων και πίνακα users στη βάση που αποθηκεύει τα απαραίτητα δεδομένα για τη βασική λειτουργία Μηχανισμού Login με πεδία (columns) όνομα / επίθετο χρήστη, username, κωδικός χρήστη (password) ορίζοντας τις κατάλληλες εντολές σε γλώσσα ερωτημάτων SQL που απαιτούνται όχι μόνο για τη δημιουργία της βάσης αλλά και του πίνακα όσον αφορά τόσο τον καθορισμό του τύπου δεδομένων των πεδίων του πίνακα όσο και τον ορισμό κατάλληλου πεδίου που θα παίξει το ρόλο πρωτεύοντος κλειδιού αναζήτησης (primary key) εγγραφών (records) στον πίνακα. Θα εισάγετε στον πίνακα users με τις κατάλληλες εντολές SQL δύο (2) χρήστες, ο πρώτος θα έχει όνομα χρήστη τον αριθμό μητρώου σας και ο δεύτερος θα έχει όνομα χρήστη admin. Δώστε την ενδεδειγμένη λύση για την αποθήκευση του κωδικού του χρήστη στη βάση. Να παραθέσετε τις εντολές SQL ή γλώσσα προγραμματισμού υψηλού επιπέδου (συμπεριλαμβάνεται η χρήση framework) που απαιτούνται για το σωστό μετασχηματισμό του. Σε γλώσσα προγραμματισμού της επιλογής σας Java, Python, PHP και αν το επιθυμείτε με τη χρησιμοποίηση κάποιου framework (προτείνεται) θα δημιουργήσετε μια απλή web εφαρμογή μέσω της οποίας οι χρήστες που δηλώσατε θα μπορούν να κάνουν Login χρησιμοποιώντας κατάλληλο endpoint. Ο κώδικάς σας πρέπει να είναι

κατάλληλα διαμορφωμένος ώστε να μην δύναται να πραγματοποιηθεί επίθεση SQL injection.

[+20%] **Προαιρετικά:** Δημιουργία login και access μέσω jwt token. Προϋποθέτει την δημιουργία κλειδιών τα οποία θα παράγουν το token το οποίο θα γίνεται assign στο χρήστη και θα λήγει σύμφωνα με το configuration σας

Παραδοτέο:

- Report (max 25 σελίδες)
    - Περιγραφή του πρωτοκόλλου, σχόλια επί πληρότητας και ορθότητας του
    - Περιγραφή διαγράμματος ροής (flow chart/diagram)
    - Περιγραφή κώδικα

- Παραδοτέος Κώδικας
    - tar/zip/άλλο με txt αρχείο οδηγιών

Ομάδες των πολύ δύο φοιτητών

Γλώσσα Προγραμματισμού Backend → Συστήνονται Java, Python, PHP (προτείνουμε τη χρήση του framework java spring boot, python django, php symfony)

DB Backend αν χρειαστεί → επιλογή σας. Συστήνονται MySQL ή MariaDB ή PostgreSQL

Γλώσσα Προγραμματισμού Frontend Περιβάλλον web (όχι mobile app)

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ — ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Bit Commitment – Πως θα υλοποιούσαμε το Coin-flipping ? Εφαρμογή Cryptographic Hash Function

Βήμα 01. **Άννα** και **Bill** παράγουν ο κάθε ένας randomly generated string, "r_A" και "r_B".

Βήμα 02. **Άννα** επιλέγει ένα outcome του flipcoin(), πχ "head"

Βήμα 03. **Bill** στέλνει σε **Άννα** το r_B.

Βήμα 04. **Άννα** υπολογίζει $SHA2_{256}$("head" || r_A || r_B) = h_{commit}

Commits *A*

Βήμα 05. **Άννα** στέλνει σε **Bill** το h_{commit}

Βήμα 06. **Άννα** ρωτάει **Bill** : "head or tails"?

Βήμα 07. **Bill** λέει για παράδειγμα "tails".

Sends *B*

Βήμα 08. **Άννα** λέει σε **Bill** «κέρδισες»

Βήμα 09. **Άννα** στέλνει σε **Bill** το string ("head" || r_A || r_B)

Opens *A*

Βήμα 09. **Bill** ελέγχει αν η Άννα είπε αλήθεια // $SHA2_{256}$("head" || r_A || r_B)=h_2

Αλήθεια αν h_2== h_{commit}