ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΙΓΑΙΟΥ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΑΣΦΑΛΕΙΑ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Τεύχος Εργαστηριακών Ασκήσεων

Μαρία Καρύδα
Αναπληρώτρια Καθηγήτρια
Τμήματος Μηχανικών Πληροφοριακών και Επικοινωνιακών Συστημάτων

Αναστασία Δούμα
ΕΔΙΠ Τμήματος Μ.Π.Ε.Σ

Αθανάσιος Μπαζάκας
Υποψήφιος Διδάκτορας Τμήματος Μ.Π.Ε.Σ

Μάρτιος 2020

ΑΣΚΗΣΗ 2:
ΥΛΟΠΟΙΗΣΗ ΜΕ ΧΡΗΣΗ ΚΑΤΑΛΛΗΛΩΝ ΚΛΑΣΕΩΝ ΤΟΥ JAVA API, ΜΗΧΑΝΙΣΜΩΝ ΑΥΘΕΝΤΙΚΟΠΟΙΗΣΗΣ, ΚΡΥΠΤΟΓΡΑΦΗΣΗΣ, ΣΥΝΟΨΗΣ ΚΑΙ ΨΗΦΙΑΚΗΣ ΥΠΟΓΡΑΦΗΣ

Περιγραφή

Σκοπός της συγκεκριμένης εργασίας είναι η εξοικείωση με το σύνολο των κλάσεων που παρέχονται από το Java API, με σκοπό την υλοποίηση μηχανισμών αυθεντικοποίησης, κρυπτογράφησης, σύνοψης και ψηφιακής υπογραφής. Οι μηχανισμοί αυτοί μπορούν να χρησιμοποιηθούν σε διάφορες εφαρμογές με απαιτήσεις ασφαλείας.

Καλείστε να αναπτύξετε μία εφαρμογή σε Java, όπου μέσω μίας απλής διεπαφής χρήστη (GUI) θα δίνεται η δυνατότητα στους χρήστες να εγγράφονται σε μια υπηρεσία ασφαλούς διαχείρισης και αποθήκευσης των στοιχείων των πιστωτικών καρτών. Ουσιαστικά η εφαρμογή που καλείστε να αναπτύξετε θα μπορούσε να αποτελεί ένα ηλεκτρονικό Wallet το οποίο θα επιτρέπει αποκλειστικά την αποθήκευση (σε κρυπτογραφημένη μορφή) και τη διαχείριση των στοιχείων των καρτών κάθε χρήστη (κυρίως πιστωτικών αλλά όχι μόνο) που χρησιμοποιεί στην καθημερινότητα του. Το μόνο που θα πρέπει να θυμάται ο χρήστης είναι ένας μοναδικός κωδικός, που θα χρησιμοποιηθεί για την είσοδο του στην εφαρμογή (Master password).

Η εφαρμογή σας θα πρέπει να αυθεντικοποιεί τους χρήστες της και να παρέχει μηχανισμούς διασφάλισης της εμπιστευτικότητας και της ακεραιότητας των αποθηκευμένων καρτών. Θα πρέπει να δίνει τη δυνατότητα σε ένα χρήστη (μέσω κατάλληλης γραφικής διεπαφής (GUI)) να δημιουργήσει νέο λογαριασμό ή σε περίπτωση που έχει ήδη λογαριασμό να αυθεντικοποιηθεί.

Κλειδιά της Εφαρμογής

Κατά την πρώτη εκτέλεση της εφαρμογής, θα πρέπει να παράγεται ένα ζεύγος ***δημόσιου*** και ***ιδιωτικού κλειδιού (RSA-2048)***, το οποίο θα πρέπει να διατηρείται αναλλοίωτο και κατά τις επόμενες εκτελέσεις της. Το δημόσιο κλειδί θα πρέπει να αποθηκεύεται σε ένα αρχείο που θα είναι διαθέσιμο και προσβάσιμο από όλους. Το ιδιωτικό κλειδί της εφαρμογής μπορεί να είναι αποθηκευμένο και αυτό σε ξεχωριστό αρχείο (όπως γνωρίζετε, δεν είναι ορθή πρακτική να αποθηκεύεται το ιδιωτικό κλειδί χωρίς να είναι προστατευμένο, αλλά είναι αποδεκτό στο πλαίσιο της παρούσας εκπαιδευτικής εργασίας). Εναλλακτικά, μπορείτε να το ορίσετε ως σταθερά στον κώδικα της εφαρμογής σας.

Εγγραφή και αυθεντικοποίηση του χρήστη

Ο χρήστης κατά τη διαδικασία της ***εγγραφής*** του στη εφαρμογή θα πρέπει να έχει τη δυνατότητα να εισάγει τα στοιχεία του (το ονοματεπώνυμο του και το email του) και τα επιθυμητά Username/Master Password για την χρήση της εφαρμογής.

Για να ολοκληρωθεί η διαδικασία της εγγραφής θα πρέπει το Username που επέλεξε ο χρήστης να είναι μοναδικό. Ακολούθως, η εφαρμογή θα παράγει τη σύνοψη (hash) του συνθηματικού του χρήστη. Για την παραγωγή της σύνοψης θα χρησιμοποιείται μαζί με το συνθηματικό και ένα διαφορετικό τυχαίο αλφαριθμητικό (salt) για κάθε χρήστη. Το αποτέλεσμα της σύνοψης θα κρυπτογραφείται με το δημόσιο κλειδί της εφαρμογής. Όλα αυτά τα στοιχεία (όνομα χρήστη, user name, salt, κρυπτογραφημένο συνθηματικό) θα

πρέπει να αποθηκεύονται σε σχετικό αρχείο που θα περιέχει τα στοιχεία όλων των χρηστών.

Ολοκληρώνοντας τη διαδικασία της εγγραφής του χρήστη, η εφαρμογή θα παράγει ένα ***συμμετρικό κλειδί (AES-256)*** για τον χρήστη, το οποίο θα το κρυπτογραφεί με το δημόσιο κλειδί της και θα το αποθηκεύει σε σχετικό αρχείο διαφορετικό για κάθε χρήστη.

Για να έχει τη δυνατότητα ένας χρήστης να εισέλθει στην εφαρμογή αφού έχει ολοκληρώσει την εγγραφή του θα πρέπει αρχικά να ***αυθεντικοποιηθεί***. Πιο συγκεκριμένα, ο χρήστης θα εισάγει το όνομα χρήστη (user name) και το συνθηματικό του. Αμέσως μετά, η εφαρμογή θα παράγει τη σύνοψη του συνθηματικού ακολουθώντας ακριβώς την ίδια διαδικασία που εκτέλεσε κατά τη διαδικασία της εγγραφής. Στη συνέχεια, θα αναζητά το συνθηματικό του χρήστη στο αρχείο που έχει αποθηκευμένα τα στοιχεία όλων των χρηστών. Θα το αποκρυπτογραφεί με το ιδιωτικό της κλειδί και θα συγκρίνει τις 2 συνόψεις που έχει δημιουργήσει. Αν ταιριάζουν οι συνόψεις τότε ο χρήστης θα αυθεντικοποιείται επιτυχώς από το σύστημα και θα έχει τη δυνατότητα να χρησιμοποιήσει τις λειτουργίες που παρέχονται από την εφαρμογή.

Λειτουργίες Εφαρμογής

Όσοι χρήστες έχουν ακολουθήσει την παραπάνω διαδικασία εγγραφής θα έχουν τη δυνατότητα να χρησιμοποιήσουν τις λειτουργίες της εφαρμογής. Για κάθε χρήστη η εφαρμογή θα πρέπει να δημιουργεί ένα κατάλογο (directory) με το username του στον οποίο θα φυλάσσονται τα διάφορα αρχεία που προκύπτουν από τις διαδικασίες της εφαρμογής. Θα πρέπει να υλοποιήσετε κατάλληλη διεπαφή, όπου ο χρήστης θα έχει τις παρακάτω δυνατότητες:

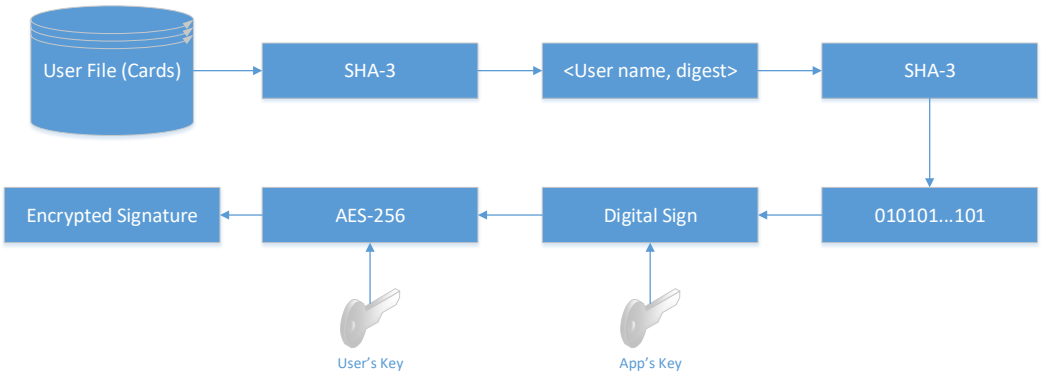
- **Προσθήκη στοιχείων κάρτας:** Ο χρήστης θα πρέπει να έχει τη δυνατότητα να εισάγει στοιχεία για κάθε κάρτα που διαθέτει. Τα στοιχεία που θα εισάγει ο χρήστης για κάθε κάρτα είναι ο τύπος της (Visa, Master Card, κλπ.), ο αριθμός της κάρτας, ο κάτοχος της κάρτας, η ημερομηνία λήξης, και ο αριθμός επαλήθευσης της (CVV/CVC2). Με την ολοκλήρωση της εισαγωγής των απαραίτητων στοιχείων της κάρτας θα πρέπει να διαμορφώνεται μια εγγραφή (αντικείμενο) με τα στοιχεία της και να γίνεται απευθείας και η κρυπτογράφησή της. Η κρυπτογράφηση θα γίνεται με το συμμετρικό κλειδί του χρήστη.
- **Εμφάνιση στοιχείων κάρτας:** Χρησιμοποιώντας το συμμετρικό κλειδί κρυπτογράφησης που παράγεται κατά την εγγραφή του χρήστη στην εφαρμογή, θα δίνεται η δυνατότητα στο χρήστη να αποκαλύπτει/αποκρυπτογραφεί τις αποθηκευμένες εγγραφές με τα στοιχεία καρτών. Ο χρήστης θα εισάγει τον τύπο κάρτας π.χ Visa και θα εμφανίζονται τα στοιχεία των καρτών αυτού του τύπου.
- **Τροποποίηση στοιχείων κάρτας:** Θα πρέπει να δίνεται η δυνατότητα στο χρήστη να τροποποιήσει μία υπάρχουσα εγγραφή. Για να το κάνει αυτό θα πρέπει να εισάγει τον τύπο και τον αριθμό της κάρτας που επιθυμεί να τροποποιήσει. Η λειτουργία της τροποποίησης θα πρέπει αυτόματα να πραγματοποιεί και την λειτουργία της αποκρυπτογράφησης ώστε να μπορεί ο χρήστης να δει το περιεχόμενο της εγγραφής που επέλεξε να τροποποιήσει.
- **Διαγραφή κάρτας:** Θα πρέπει να δίνεται η επιλογή ο χρήστης να διαγράφει μία κάρτα, εισάγοντας τον τύπο και τον αριθμό της.

Μπορείτε να προσθέσετε επιπλέον λειτουργίες σε περίπτωση που η υλοποίηση σας το απαιτεί. Σε κάθε περίπτωση όμως θα πρέπει να εξασφαλίζεται η λειτουργικότητα της εφαρμογής και να ικανοποιούνται οι απαιτήσεις της εργασίας.

Κατά το κλείσιμο της εφαρμογής θα πρέπει να κρυπτογραφούνται αυτόματα τα στοιχεία των καρτών που έχει αποκρυπτογραφήσει ο χρήστης και να φυλάσσονται σε σχετικό αρχείο.

Μηχανισμός Ακεραιότητας καρτών

Η εφαρμογή θα πρέπει να εξασφαλίζει την ακεραιότητα των στοιχείων που αποθηκεύουν οι χρήστες για τις κάρτες τους. Για το λόγο αυτό θα πρέπει να αναπτύξετε ένα μηχανισμό διασφάλισης των στοιχείων αυτών από μη-εξουσιοδοτημένη τροποποίηση. Ο μηχανισμός αυτός ακολουθεί το παρακάτω σχήμα:



Κατά το κλείσιμο της εφαρμογής θα πρέπει να υπολογίζονται οι συνόψεις όλων των αρχείων των χρηστών που περιέχουν τα στοιχεία των καρτών ξεχωριστά χρησιμοποιώντας τη μονόδρομη συνάρτηση κατακερματισμού SHA-3 (SHA3_256). Παράγωγο αυτής της διαδικασίας είναι τα ζεύγη <user name, digest>. Στη συνέχεια, το κάθε ζεύγος <user name, digest> θα υπογράφεται ψηφιακά από την εφαρμογή (μία ψηφιακή υπογραφή ανά χρήστη). Τέλος, η ψηφιακή υπογραφή θα κρυπτογραφείται με το συμμετρικό κλειδί του χρήστη και θα αποθηκεύεται σε κατάλληλο αρχείο.

Η αντίστροφη διαδικασία της επιβεβαίωσης θα πρέπει να ακολουθείται από την εφαρμογή μετά την αυθεντικοποίηση του χρήστη. Ο χρήστης θα πρέπει να ενημερώνεται εάν έχει γίνει κάποια μη-εξουσιοδοτημένη τροποποίηση των καρτών του με σχετικό μήνυμα.

Τροποποιήστε εσκεμμένα κάποιο αρχείο με τα στοιχεία των καρτών για κάποιο χρήστη και επιβεβαιώστε ότι ο μηχανισμός ακεραιότητας που υλοποιήσατε λειτουργεί σωστά.

Ερωτήσεις

Απαντήστε σύντομα στις παρακάτω ερωτήσεις:

- Ποιος ο λόγος της χρήσης του salt για την παραγωγή της σύνοψης ενός συνθηματικού;
- Ποιες είναι κατά την γνώμη σας οι αδυναμίες της συγκεκριμένης εφαρμογής; Περιγράψτε σύντομα τι ευπάθειες μπορεί να εκμεταλλευτεί ένας επιτιθέμενος. Προτείνετε μηχανισμούς που κατά την γνώμη σας μπορούν να βελτιώσουν την ασφάλεια που παρέχει η εφαρμογή.

Όπως ήδη αναφέρθηκε για την υλοποίηση της εφαρμογής θα πρέπει να γίνει αποκλειστική χρήση του API της Java.

Ο πηγαίος κώδικας θα αξιολογηθεί ως προς το αν υλοποιεί τα βασικά ζητούμενα της εκφώνησης, εκτελείται χωρίς να προκύπτουν σφάλματα λογισμικού (bugs), ακολουθεί

«καλές αρχές» προγραμματισμού (π.χ. σχολιασμό, στοίχιση, εύγλωττη ονοματοδοσία μεταβλητών, επαναχρησιμοποίηση κώδικα, κλπ).

Παραδοτέα
Τελική ημερομηνία παράδοσης της εργασίας: **Τετάρτη 15/04/2020**.

- Πηγαίος κώδικας της εφαρμογής **με τον απαραίτητο σχολιασμό** (ολόκληρο το project).
- **Αναφορά** που θα περιέχει οθόνες εκτέλεσης **ΟΛΩΝ** των λειτουργιών της εφαρμογής και **σχολιασμός των αποτελεσμάτων**. Οι οθόνες εκτέλεσης θα πρέπει να είναι **ευκρινείς**. Στην αναφορά θα πρέπει να επεξηγήσετε τις κλάσεις και μεθόδους Java που χρησιμοποιήσατε για την υλοποίηση της εφαρμογής. Θα πρέπει επίσης να περιγράψετε το τρόπο που δουλέψατε, να τεκμηριώστε αδυναμίες, περιττά βήματα ή βελτιώσεις που προτείνετε στην εφαρμογή.
- Στο τέλος της εργασίας θα πρέπει να αναφέρετε όλες τις πηγές που έχετε χρησιμοποιήσει (και σχετικούς συνδέσμους σε όσες αναφορές υπάρχουν).
- Αρχείο με το δημόσιο κλειδί της εφαρμογής.
- **Το σύνολο των φακέλων** και των αρχείων που θα δημιουργήσετε (Αρχεία που περιέχουν τα κλειδιά, directories, αρχεία που δημιουργήθηκαν για τους χρήστες, κτλ.).

Οδηγίες Παράδοσης
Για την 2η εργασία θα πρέπει να παραδοθεί ένα συμπιεσμένο αρχείο που θα περιέχει την αναφορά σε pdf αρχείο με όλα τα ζητούμενα της εργασίας (π.χ επεξήγηση/τεκμηρίωση κώδικα, οθόνες εκτέλεσης, απαντήσεις σε ερωτήματα κλπ.) καθώς και τα πρόσθετα αρχεία που ζητούνται στην ενότητα «παραδοτέα».

Το όνομα του αρχείου που θα παραδώσετε θα είναι ένα συμπιεσμένο αρχείο της μορφής AM1_Lab02.zip (π.χ. icsd17999_Lab02.zip – αριθμός μητρώου του υπεύθυνου της ομάδας).

Θα διορθωθούν μόνο οι ασκήσεις που πληρούν την παραπάνω περιγραφή.

Χρήσιμα Links
Java API: http://docs.oracle.com/javase/8/docs/api/