Secure Communication Project

Rev=0.1

Σε αυτήν την εργασία, καλείστε να υλοποιήστε ένα σύστημα ασφαλούς επικοινωνίας μεταξύ δύο εφαρμογών client/server.

Α. φάση: Υλοποίηση δομικών στοιχείων

Στην πρώτη φάση θα πρέπει να υλοποιήσετε τα δομικά στοιχεία της εφαρμογής:

1. **Digest:** Υλοποιήστε μία εφαρμογή που θα δέχεται ως είσοδο ένα μήνυμα και θα υπολογίζει την σύνοψή του. Η σύνοψη θα υπολογίζεται με βάση το αλγόριθμο SHA-512.
2. **Συμμετρική κρυπτογραφία:** Υλοποιείστε μία εφαρμογή για κρυπτογράφηση/αποκρυπτογράφηση ενός μηνύματος με τον αλγόριθμο AES, σε mode ECB και padding PKCS5, με κλειδί AES 256-bits. Υλοποιήστε κατάλληλους ελέγχους για την ορθή λειτουργία της εφαρμογής. Πριν την κρυπτογράφηση προσθέστε στην αρχή του plaintext:
   a. το digest του κλειδιού κρυπτογράφησης
   b. το πραγματικό μέγεθος του plaintext

   Μετά την αποκρυπτογράφηση:
   a. Ελέγξτε ότι το αποκρυπτογραφημένο μήνυμα ξεκινάει με το digest του κλειδιού κρυπτογράφησης
   b. Ελέγξτε το πραγματικό μέγεθος του plaintext και επιστρέψτε τα αντίστοιχα bits του αποκρυπτογραφημένου μηνύματος.
3. **Ασύμμετρη κρυπτογραφία:** Υλοποιήστε μία εφαρμογή για κρυπτογράφηση/αποκρυπτογράφηση ενός μηνύματος με τον αλγόριθμο RSA και padding PKCS1, με κλειδί RSA 2048-bits. Υλοποιήστε όμοιους ελέγχους με την ορθή λειτουργία της συμμετρικής κρυπτογραφίας.
4. **Ασφαλής επικοινωνία:** Δημιουργήστε δύο εφαρμογές client/server που θα επικοινωνούν κρυπτογραφημένα με χρήση AES-256 bits, σε mode ECB και padding PKCS5. Θεωρείστε ότι το κλειδί είναι ήδη γνωστό και στις δύο οντότητες.

Β. φάση: Σχεδίαση και υλοποίηση της εφαρμογής

Σε αυτήν την φάση θα πρέπει να σχεδιάσετε και να υλοποιήσετε μία εφαρμογή ασφαλούς επικοινωνίας μεταξύ ενός client με τον server, αξιοποιώντας τα δομικά στοιχεία της πρώτης φάσης.

Αρχικά, η κάθε οντότητα έχει στην κατοχή της το δικό της ζευγάρι RSA κλειδιών (δημόσιο και ιδιωτικό) και το δημόσιο κλειδί της άλλης. Οι δύο οντότητες χρησιμοποιούν την ασύμμετρη κρυπτογραφία για να ανταλλάξουν ένα συμμετρικό κλειδί AES. Το συμμετρικό κλειδί χρησιμοποιείτε μετέπειτα για την ασφαλή επικοινωνία.

1. **Ανταλλαγή συμμετρικού κλειδιού:** Σχεδιάστε την διαδικασία με τον οποία ο server δημιουργεί ένα συμμετρικό κλειδί AES-256 bits και το στέλνει στον

client με χρήση ασύμμετρης κρυπτογραφίας RSA-2048 bits. Σχεδιάστε τα βήματα της επικοινωνίας και την μορφή των μηνυμάτων που ανταλλάσσουν. Έστω [Su, Sr] το δημόσιο/ιδιωτικό κλειδί του server και [Cu, Cr] το δημόσιο/ιδιωτικό κλειδί του client. Η κάθε οντότητα γνωρίζει το δικό της ζευγάρι κλειδιών και το δημόσιο κλειδί της άλλης. Ο server κρυπτογραφεί ένα μήνυμα m με βάση το: ECu(ESr(m)) = c. Ο client θα αποκρυπτογραφεί το μήνυμα με βάση το: DCr(DSu(c)) = m. Όπου m είναι το μήνυμα που στέλνει ο server με το AES κλειδί.

2. **Επικοινωνία με συμμετρικό κλειδί:** Αφού έχουν ανταλλάξει το συμμετρικό κλειδί, ο client θα στέλνει ένα μήνυμα στον server και θα τερματίζουν την επικοινωνία τους. Σχεδιάστε το βήματα της επικοινωνίας, την μορφή του μηνύματος και τον τερματισμό της εφαρμογής.

Παραδοτέα

Στην διάρκεια κάθε φάσης μπορείτε να ρωτάτε απορίες.
Με το τέλος της δεύτερης φάσης θα πρέπει να παραδώσετε την τελική υλοποίηση της εφαρμογής καθώς και μία αναφορά με την σχεδίαση και τα βασικά σημεία της υλοποίησης. Η αναφορά θα είναι εξολοκλήρου σε ηλεκτρονική μορφή (όχι σκαναρισμένα όχι φωτογραφίες).

Βαθμολογία

Η βαθμολογία του project θα γίνει στην τελική υλοποίηση και στην αναφορά που θα παραδώσετε:
40% για την σωστή υλοποίηση των δομικών στοιχείων της εφαρμογής
60% για την ποιότητα της αναφοράς και την σωστή υλοποίηση της τελικής εφαρμογής

Deadline

Με το τέλος του εξαμήνου (θα υπάρξει σχετική ανακοίνωση).

Σημαντικές παρατηρήσεις:

1. Θα χρησιμοποιήσετε γλώσσα προγραμματισμού **Python 2.7** σε πλατφόρμα **Linux**.
2. Η επίλυση απορίων σχετικά με τα παρακάτω προβλήματα γίνεται αποκλειστικά μέσω του forum στο courses (στα Threads: «**FINAL_A**, **FINAL_B**, **FINAL_REPORT**»). Όχι με προσωπικά e-mails.
3. Οποιαδήποτε άλλη ερώτηση σχετικά με το μάθημα πρέπει να απευθύνεται στο e-mail του διδάσκοντα.
4. Θα πρέπει να ολοκληρώσετε τον κώδικα σας σε όσο το δυνατόν λιγότερα **sources (π.χ tools.py, client.py, server.py).** Χρησιμοποιείστε σχόλια και τα απαραίτητα «prints» όπου χρειάζεται έξοδος.

5. Απαιτείται report σε ηλεκτρονική μορφή με **συγκεκριμένο format (ACM small standard format)**[1]
6. Block diagrams/visualizations που ενδεχομένως χρησιμοποιήσετε θα πρέπει να είναι **σχεδιασμένα σε υπολογιστή** (όχι στο χέρι).
7. Στο **ίδιο zip file** θα βρίσκεται και ο **κώδικας** μαζί με **όλα τα υπόλοιπα αρχεία** που χρειάζεται ο κώδικας σας για να τρέξει. Π.χ plaintext file, RSA keys κτλ. Στο ίδιο zip θα βρίσκεται επίσης και η **αναφορά** σε pdf.
8. Ο κώδικας θα πρέπει να τρέχει **"out-of-the-box"!** χωρίς καμία τροποποίηση.
9. Στην αρχή του κώδικα θα αναφέρονται όλα τα προσωπικά σας στοιχειά σας και το email σας σαν σχόλια καθώς και τα dependencies του κώδικα σας (αν υπάρχουν).
10. Όλοι οι κώδικες σας θα ελεγθούν για ομοιότητες.

[1] http://www.acm.org/publications/article-templates/acmsmall-word.zip