ΑΣΦΑΛΕΙΑ ΠΛΗΡΟΦΟΡΙΑΚΩΝ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Τρίτη Εργαστηριακή Άσκηση

Ομάδα Φοιτητών :

Μπινιάκου Θεοφάνης icsd13126

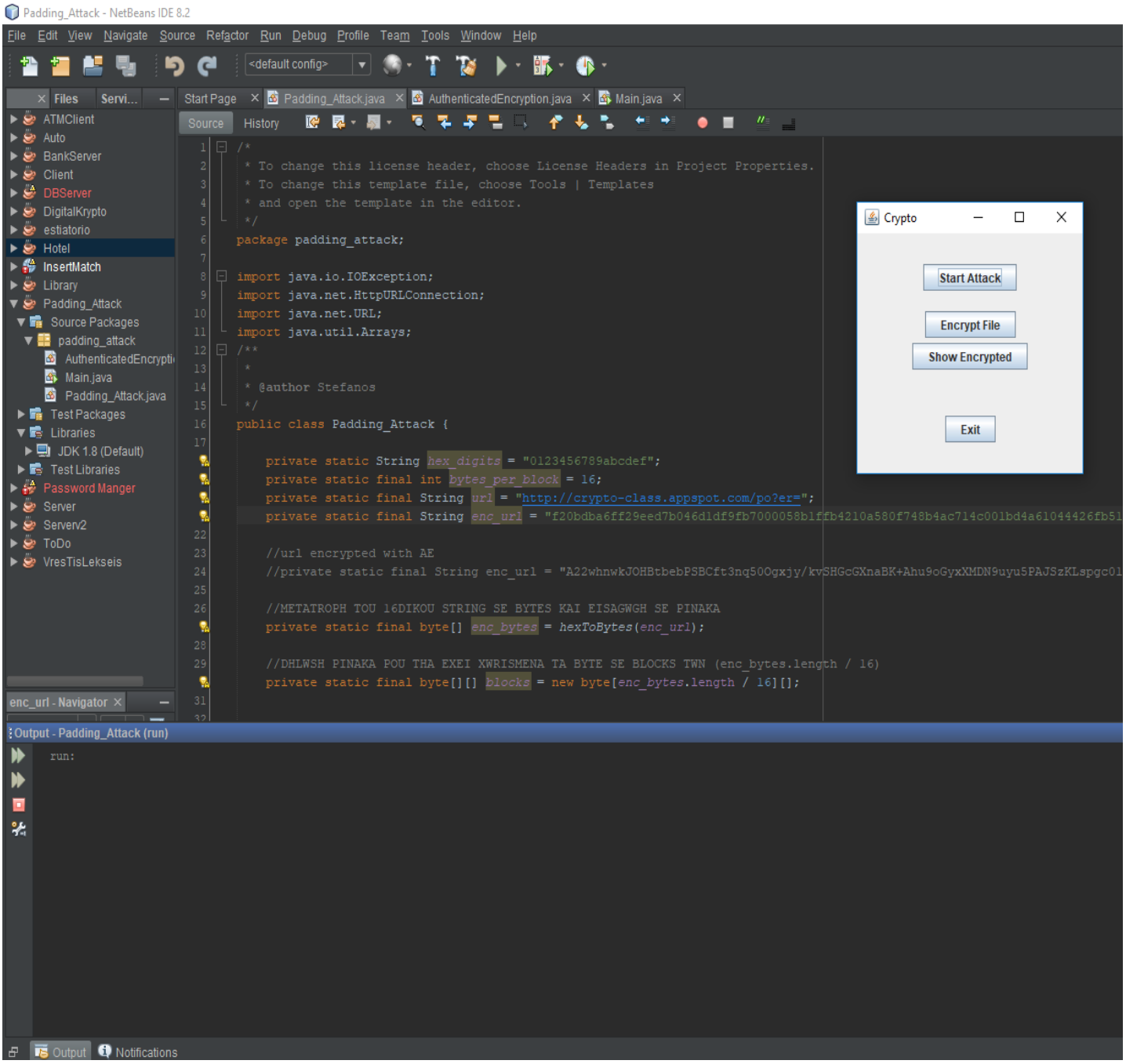
Τσότρας Στέφανος icsd13189

Κοντογεωργόπουλος Κωνσταντίνος icsd13080

A) Το αποτέλεσμα της επίθεσης είναι : The Magic Words are Squeamish Ossifrage.

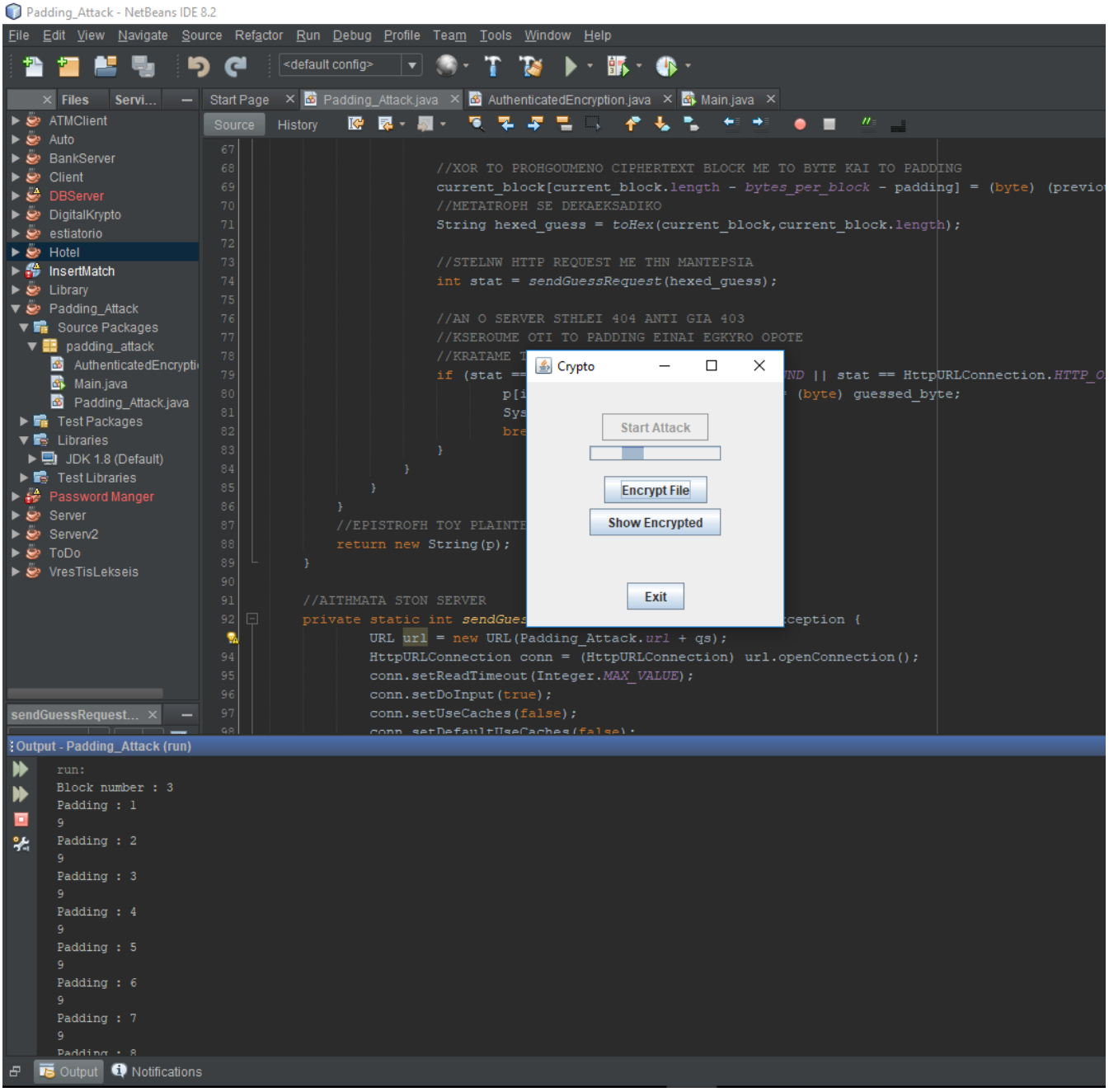
B) I) Επίθεση στον server

Έναρξη εφαρμογής.



Πατώντας Start Attack ξεκινάει ένα Thread το οποίο καλεί την μέθοδο στην οποία γίνεται η επίθεση, δίνοντας έτσι την δυνατότητα στο χρήστη να χρησιμοποιήσει τις υπόλοιπες λειτουργίες της εφαρμογής.

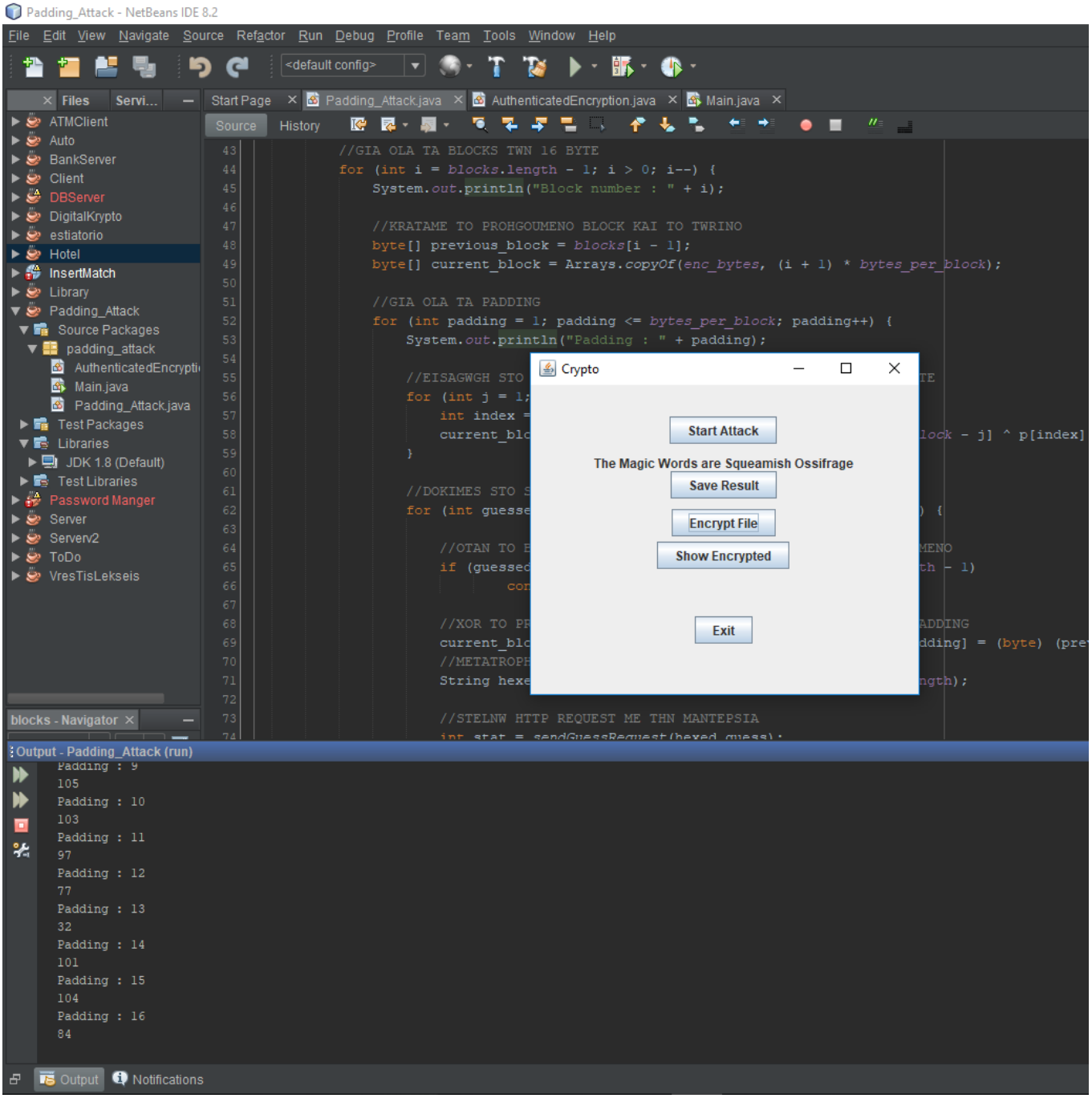
Γραφικό περιβάλλον : Μπάρα προόδου (ενεργοποιημένη όσο τρέχει το Thread), Κλείδωμα του κουμπιού start attack.



Στην κονσόλα βλέπουμε σε ποιο απ’τα blocks και με πιο padding γίνεται η μαντεψιά.

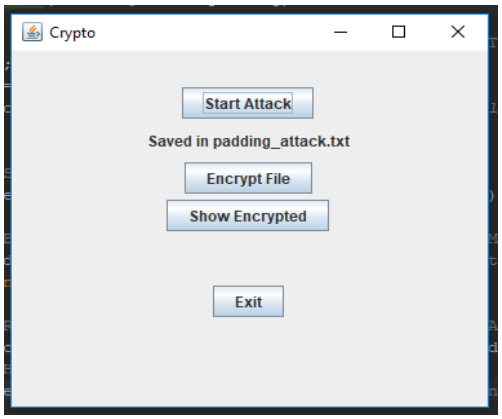
Όταν το Http Request που στείλουμε επιστρέψει 404 ξέρουμε ότι το padding ήταν έγκυρο και εμφανίζουμε το μαντεμένο byte.

Όταν η επίθεση τελειώσει εμφανίζεται σε ένα label το αποτέλεσμα και ο χρήστης έχει την επιλογή να σώσει το αποτέλεσμα σε αρχείο.



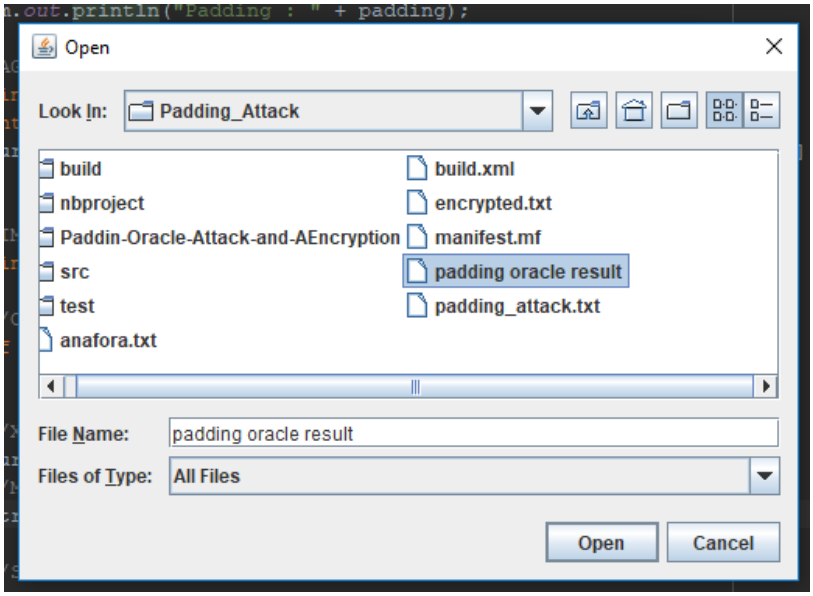
Αν επιλέξει να το σώσει εμφανίζεται το αρχείο που έγινε η αποθήκευση.

Αλλιώς μπορεί να ξανακάνει επίθεση η να επιλέξει την κρυπτογράφηση κάποιου text.



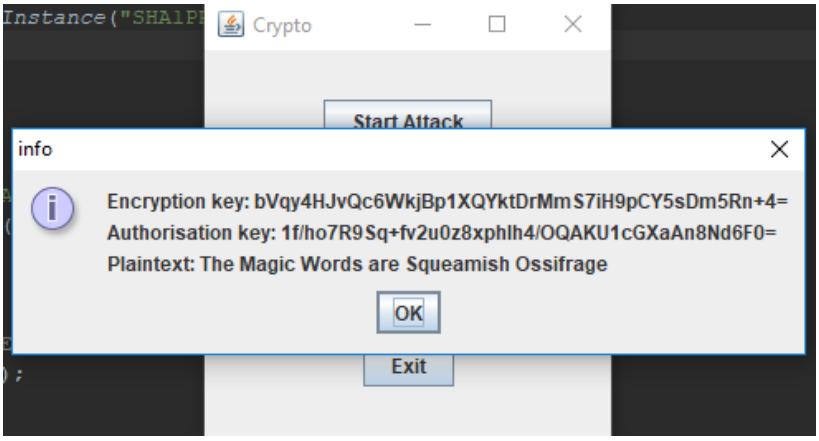
ii) Παραγωγή Αυθεντικοποιημένης κρυπτογράφησης.

Πατώντας Encrypt File ανοίγει ένα JFileChooser για να επιλέξεις το αρχείο που θες να κάνεις encrypt.



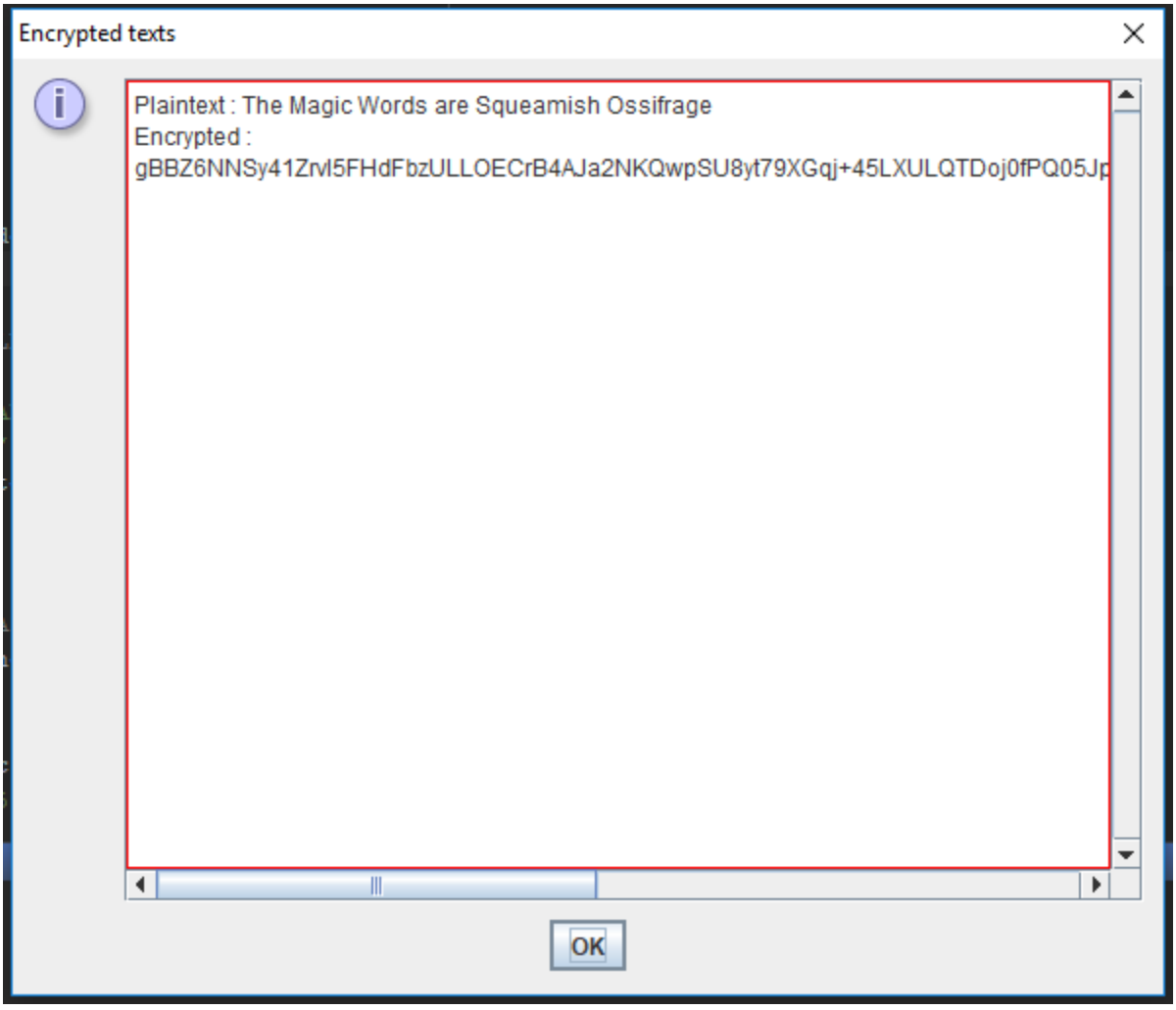
Αφού διαβαστεί το αρχείο δημιουργούνται δύο κλειδιά, το πρώτο για την κρυπτογράφηση του κειμένου και το άλλο για την δημιουργία του κώδικα αυθεντικότητας (MAC).

Εμφάνιση των κλειδιών και του κειμένου σε ένα showMessage Pane.

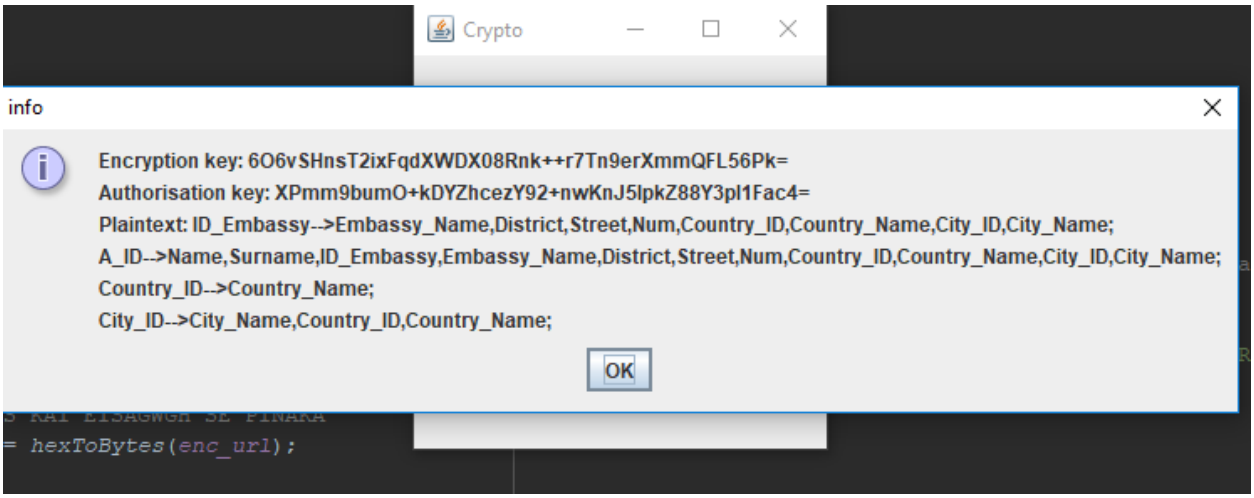


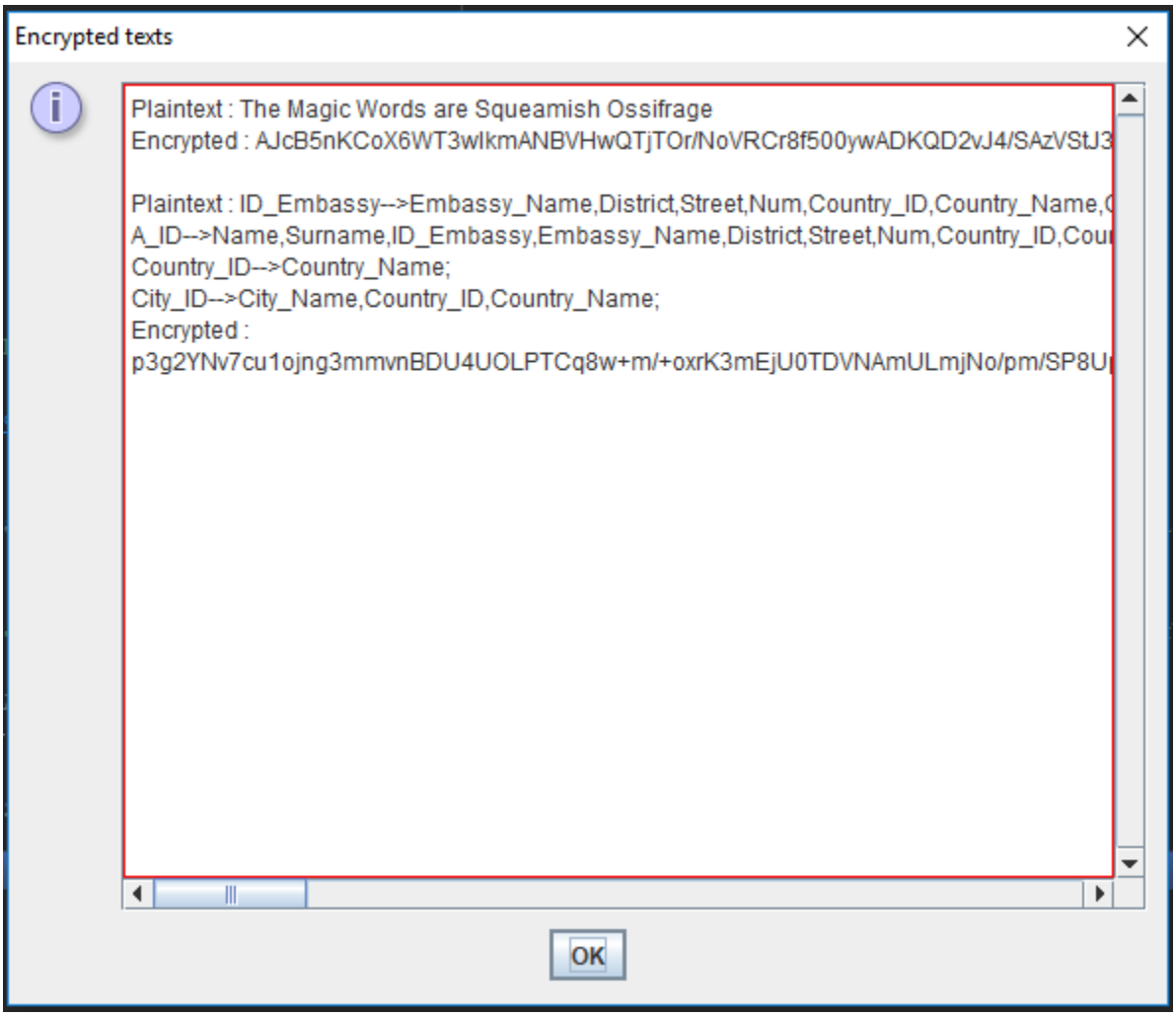
Πατώντας Ok γίνεται η κρυπτογράφηση και όταν τελειώσει σώζεται στην ‘’βάση’’(ένα txt αρχείο) και εμφανίζεται το αποτέλεσμα σε ένα JTextPane (ενθυλακωμένο σε ένα JScrollPane).

Το κουμπί Show Encrypted εμφανίζει ακριβώς το ίδιο αποτέλεσμα παίρνοντας την πληροφορία απτην ‘’βάση’’.



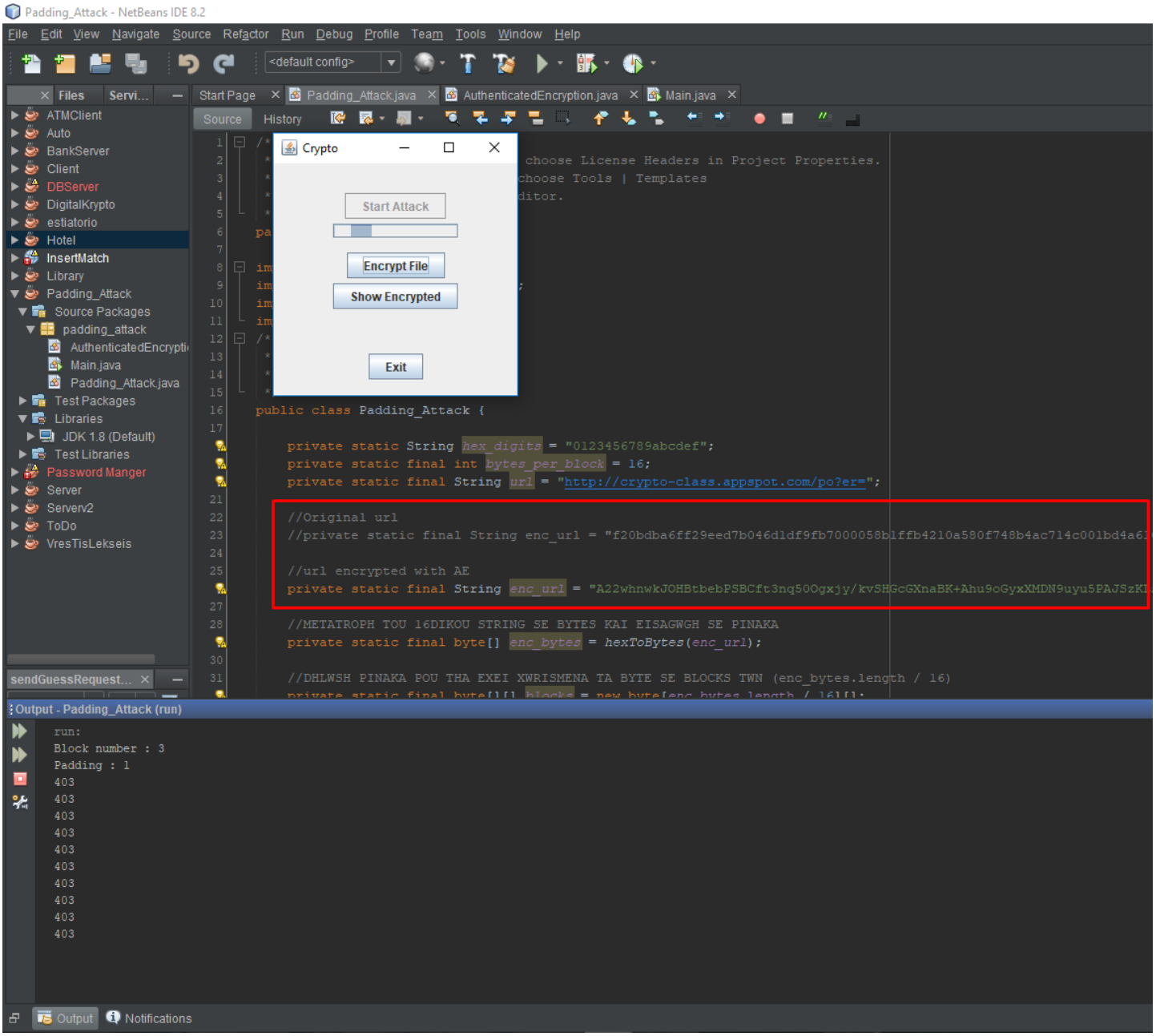
Μπορούμε να κρυπτογραφήσουμε οποιοδήποτε κείμενο θέλουμε κρατώντας και τα προηγούμενα κρυπτογραφημένα κείμενα μας.





iii) Αδυναμία επαλήθευσης του MAC όταν αλλοιώνεται το αυθεντικοποιημένο κρυπτοκείμενο.

Αλλοιώνοντας το κρυπτοκείμενο το request στέλνει μόνιμα 403 κάνοντας έτσι την αδύνατη την επίθεση.



Γ) Απαντήσεις στα ερωτήματα.

1. Είναι απαραίτητη η αποκρυπτογράφηση του πρώτου block του κρυπτοκειμένου; Αν ναι, γιατί;

Η κρυπτογράφηση του πρώτου block είναι απαραίτητη αφού για την αποκρυπτογράφηση του επόμενου block χρησιμοποιείται το πρώτο block.

2. Ποιος είναι ο μέγιστος αριθμός των http requests που χρειάζονται ώστε να αποκρυπτογραφηθεί ένα byte του plaintext στη συγκεκριμένη επίθεση;

Για την padding oracle επίθεση μπορείς να αποκρυπτογραφήσεις ένα byte την φόρα. Για κάθε byte χρειάζονται το πολύ 256 Http Requests.

3. Θα ήταν δυνατή η επίθεση αν ο server χειριζόταν με τον ίδιο τρόπο τα σφάλματα κατά την αποκρυπτογράφηση (μη έγκυρο padding) με τα σφάλματα που προκύπτουν από ένα αλλοιωμένο plaintext;

Όχι, γιατί αν ο server επέστρεφε 403 και για τα δύο σφάλματα δεν θα μπορούσαμε να ξεχωρίσουμε το έγκυρο padding και η επίθεση θα ήταν αδύνατη.

4. Θα ήταν δυνατή η επίθεση αν είχε χρησιμοποιηθεί AES σε CTR mode;

Στην περίπτωση του AES σε CTR mode τα extra byte του κρυπτογραφημένου plaintext αφαιρούνται οπότε δεν χρησιμοποιείται padding, κάνοντας την επίθεση αδύνατη.

5. Αν η αυθεντικοποίηση του κρυπτοκειμένου γινόταν με το σχήμα MAC-then-Encrypt κι όχι με το Encrypt-then-MAC που αναφέρθηκε στα ζητούμενα, θα μπορούσε ο server να παρέχει ασφάλεια; Αν ναι, με ποιόν τρόπο;

Ο server θα μπορούσε να παρέχει ασφάλεια με MAC-then-Encrypt, χειρίζοντας τα padding errors και τα MAC errors με τον ίδιο τρόπο.