Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Μεταπτυχιακό Πρόγραμμα Σπουδών

Πληροφοριακά και Επικοινωνιακά Συστήματα

Μεταπτυχιακή Διατριβή

Τεχνικές ανάπτυξης web εφαρμογών για κρυπτογράφηση από-άκρο-σε-άκρο

Γεώργιος Ντόντος

Επιβλέπων Καθηγητής
Κωνσταντίνος Λιμνιώτης

Δεκέμβριος 2017

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Μεταπτυχιακό Πρόγραμμα Σπουδών
Πληροφοριακά και Επικοινωνιακά Συστήματα

Μεταπτυχιακή Διατριβή

Τεχνικές ανάπτυξης web εφαρμογών για κρυπτογράφηση από-άκρο-σε-άκρο

Γεώργιος Ντόντος

Επιβλέπων Καθηγητής
Κωνσταντίνος Λιμνιώτης

Η παρούσα μεταπτυχιακή διατριβή υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων για απόκτηση μεταπτυχιακού τίτλου σπουδών στα Πληροφοριακά και Επικοινωνιακά Συστήματα

από τη Σχολή Θετικών και Εφαρμοσμένων Επιστημών του Ανοικτού Πανεπιστημίου Κύπρου

Δεκέμβριος 2017

ΛΕΥΚΗ ΣΕΛΙΔΑ

Περίληψη

Η αδυναμία των κλασικών τεχνικών ασφαλείας για την πλήρως αποτελεσματική προστασία των δεδομένων στις εφαρμογές ιστού έχει οδηγήσει σε πληθώρα περιστατικών διαρροής εμπιστευτικών δεδομένων, αναδεικνύοντας έτσι την ανάγκη της κρυπτογράφησης των δεδομένων από-άκρο-σε-άκρο, έτσι ώστε τα δεδομένα να τηρούνται κρυπτογραφημένα ακόμα και στον εξυπηρετητή ιστού. Το γεγονός αυτό, παράλληλα με την αδυναμία των κλασικών κρυπτογραφικών τεχνικών να παρέχουν κρυπτογράφηση από-άκρο-σε-άκρο και ταυτόχρονα να καλύπτουν τις ανάγκες λειτουργικότητας των εφαρμογών, έχουν οδηγήσει την ερευνητική κοινότητα να στραφεί προς νέες τεχνολογικές προσεγγίσεις. Οι προσεγγίσεις αυτές ενσωματώνουν νέες ειδικές κρυπτογραφικές τεχνικές και δίνουν τη δυνατότητα δημιουργίας λειτουργικών εφαρμογών, ενώ παράλληλα επιτυγχάνουν κρυπτογράφηση από-άκρο-σε-άκρο.

Στη παρούσα διατριβή πραγματοποιείται λεπτομερής ανάλυση των χαρακτηριστικών μίας νέας πλατφόρμας που έχει προταθεί για την αποτελεσματική κρυπτογράφηση από-άκρο-σε-άκρο, της πλατφόρμας Mylar. Ειδικότερα, μελετήθηκε ο τρόπος λειτουργίας της, με έμφαση στις απαιτήσεις ασφαλείας που ικανοποιεί. Η πλατφόρμα μελετάται τόσο σε θεωρητικό επίπεδο με ανάλυση των ειδικών κρυπτογραφικών χαρακτηριστικών της, όσο και σε πρακτικό επίπεδο με ενσωμάτωσή της σε μία υπάρχουσα εφαρμογή ιστού. Σε αυτήν την κατεύθυνση, πραγματοποιήθηκε, μέσω εκτέλεσης κατάλληλων σεναρίων χρήσης της εφαρμογής, αποτίμηση της απόδοσης αυτής με την ενσωμάτωση των χαρακτηριστικών ασφαλείας της πλατφόρμας Mylar, σε σχέση με την αρχική απλή εκδοχή της εφαρμογής η οποία δεν παρέχει υπηρεσίες ασφαλείας. Μέσα από τη μελέτη αυτής της νέας τεχνολογικής προσέγγισης δίνεται θετική απάντηση στα ερωτήματα που αφορούν στη δυνατότητα δημιουργίας εφαρμογών ιστού, οι οποίες να προστατεύουν πλήρως τα ευαίσθητα δεδομένα των χρηστών τους από-άκρο-σε-άκρο και παράλληλα να είναι σε υψηλό βαθμό λειτουργικές.

Summary

Due to the large number of data breach incidents, traditional security techniques seem that they are not fully adequate for data protection in web applications and, thus, the need for end-to-end encryption of data – so as to ensure that web servers have access only to encrypted data - becomes extremely important. This, along with the insufficiency of classical cryptographic techniques to simultaneously achieve end-to-end encryption as well as high functionality, has led the research community to study new technological approaches. These approaches incorporate new specific cryptographic techniques that allow the development of functional applications while achieving end-to-end encryption.

This thesis focuses on Mylar, a recent open source research platform which supports end-to-end encryption for web applications. More precisely, a detailed analysis of the features of the Mylar platform and its functionality is presented, whilst its security characteristics are also discussed. The underlying cryptographic features of Mylar are fully described, whereas the platform is also studied at a practical level via integrating it into an existing web application. In this direction, effort has been put on evaluating the performance of a web application which is enriched with Mylar security features, compared to the initial version of the application without security services. Our study indicates that such a new technology approach suffices to protect sensitive end-user data via an end-to-end encryption, while simultaneously being fully functional.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή αυτής της διατριβής κ. Κωνσταντίνο Λιμνιώτη για την αμέριστη συμπαράσταση και την καθοριστική συμβολή του στην εκπόνηση της παρούσας διατριβής.

Περιεχόμενα

1 Εισαγωγή 1

1.1 Αντικείμενο και δομή της διατριβής 3

2 Βασικές αρχές κρυπτογραφίας 5

2.1 Κρυπτογραφικοί στόχοι και ορολογία 6

2.1.1 Κρυπτογραφικοί στόχοι 6

2.1.2 Βασικοί όροι 7

2.1.3 Ασφάλεια 8

2.2 Κρυπτογραφικά συστήματα 9

2.2.1 Συμμετρική κρυπτογράφηση 9

2.2.2 Ασύμμετρη κρυπτογράφηση 13

2.2.3 Συναρτήσεις κατακερματισμού (Hash - MAC functions) 15

2.2.4 Ψηφιακή υπογραφή 16

2.2.5 Ψηφιακά πιστοποιητικά 16

3 Ειδικές κρυπτογραφικές τεχνικές 19

3.1 Ασφάλεια (επικρατούσα τάση) 19

3.2 Ασφάλεια (κρυπτογράφηση) 21

3.2.1 Κοινή χρήση δεδομένων 21

3.2.2 Δυνατότητα υπολογισμών και αναζητήσεων επί κρυπτογραφημένων δεδομένων ... 22

4 Πλατφόρμα Mylar: Αρχιτεκτονική και Λειτουργία 28

4.1 Αρχιτεκτονική 29

4.1.1 Δομικά στοιχεία λογισμικού του Mylar 29

4.2 Ασφάλεια 30

4.3 Κύριες οντότητες 32

4.3.1 Στατικές Principals 34

4.4 Λειτουργία 35

4.4.1 Κοινή χρήση δεδομένων (data sharing) 35

4.4.2 Αναζήτηση (Searching) 40

5 Πρακτική εφαρμογή του Mylar: Η περίπτωση του kChat 45
5.1 kChat 46
5.2 kChat + Mylar = EncChat (Ενσωμάτωση) 48
5.3 Υποδομή και εγκατάσταση 50
5.4 Κρυπτογράφηση από-άκρο-σε-άκρο 52
5.5 Αξιολόγηση της απόδοσης 54
5.5.1 Σύνδεση σε δωμάτιο 55
5.5.2 Αποστολή μηνύματος 56
5.5.3 Πρόσκληση χρήστη 57
5.5.4 Αναζήτηση μηνύματος 58
5.5.5 Χωρητικότητα 60

6 Επίλογος 63
6.1 Σύνοψη 63
6.2 Συμπεράσματα 64

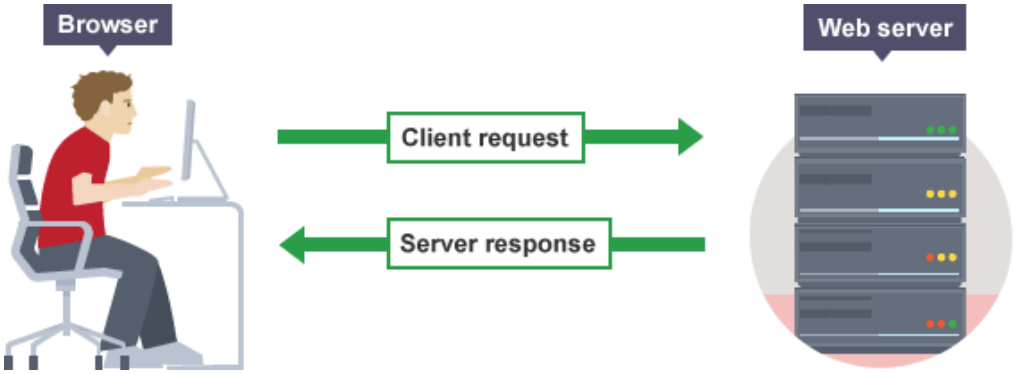
Βιβλιογραφία 67

A Διεπαφή εφαρμογών του Mylar A-1

B Υλικό σεναρίων και αναλυτικά γραφήματα B-1
B.1 Ψευδο-κώδικας σεναρίων από το VUGen B-2
B.2 Λίστες παραμέτρων σεναρίων B-6
B.3 Αναλυτικά γραφήματα εκτελέσεων B-9
B.4 Στοιχεία της συλλογής μηνυμάτων (Robomongo) B-16

Κεφάλαιο 1
Εισαγωγή

Η ολοένα αυξανόμενη χρήση εφαρμογών ιστού (web applications) έχει συχνά ως αποτέλεσμα τη διαρροή εμπιστευτικών κρίσιμων δεδομένων, χωρίς μάλιστα πολλές φορές αυτό να το γνωρίζουν οι χρήστες των εφαρμογών. Πράγματι, παρά τις διάφορες τεχνικές προστασίας των δεδομένων που εφαρμόζονται στις εν λόγω εφαρμογές, υπάρχει πληθώρα περιστατικών παραβίασης δεδομένων τα οποία οφείλονται σε ποικίλα είδη επιθέσεων ασφαλείας – είτε από «απλές» επιθέσεις κοινωνικής μηχανικής, είτε από προηγμένες επιθέσεις ασφαλείας που εκμεταλλεύονται ευπάθειες σε πρωτόκολλα ασφαλείας, ιδίως δε σε όχι ορθές υλοποιήσεις αυτών.



Εικόνα 1.1: Αρχιτεκτονική πελάτη-εξυπηρετητή (client-server) σε εφαρμογή ιστού.

Στις εφαρμογές ιστού χρησιμοποιείται η αρχιτεκτονική πελάτη-εξυπηρετητή (client-server) (Εικόνα 1.1). Για την ασφάλεια των εφαρμογών αυτών αξιοποιούνται κρυπτογραφικές τεχνικές – αφού η έλλειψη κρυπτογράφησης θα καθιστούσε κάθε τέτοια εφαρμογή πλήρως ευάλωτη στον οποιονδήποτε αποκτούσε πρόσβαση στο φυσικό μέσο μετάδοσης των δεδομένων. Δύο κρυπτογραφικές προσεγγίσεις είναι αυτές που μπορούν να υιοθετηθούν: στην πρώτη, που είναι και η απλούστερη και συνηθέστερη, τα δεδομένα αποστέλλονται κρυπτογραφημένα και στη συνέχεια αποκρυπτογραφούνται στον εξυπηρετητή, ώστε να μπορούν να είναι εύκολα επεξεργάσιμα και να μπορεί ο εξυπηρετητής ιστού (web server) να ικανοποιεί τα αιτήματα του χρήστη. Στη δεύτερη τα δεδομένα αποθηκεύονται κρυπτογραφημένα και δεν αποκρυπτογραφούνται ούτε από τον εξυπηρετητή ιστού.

Στην πρώτη περίπτωση, η προστασία των δεδομένων εστιάζεται στο να αποτρέψουμε τους επίδοξους εισβολείς να τα προσεγγίσουν, το οποίο είναι ένα πολύ δύσκολο εγχείρημα. Αν και η στρατηγική αυτή είναι η καθιερωμένη μέθοδος προστασίας, σε πολλές περιπτώσεις τα αποτελέσματα της δεν είναι τα αναμενόμενα [12]. Επίσης, πρέπει να σημειωθεί ότι στην περίπτωση αυτή, ακόμη και αν έχει διασφαλιστεί ότι κάθε υποκλοπέας δεν είναι σε θέση να πλήξει την ασφάλεια της εφαρμογής, τα δεδομένα είναι διαθέσιμα στον εξυπηρετητή – γεγονός που σε κάποιες περιπτώσεις ενδεχομένως εγείρει ζητήματα ασφαλείας και προστασίας προσωπικών δεδομένων (π.χ. αν πρόκειται για ευαίσθητα δεδομένα υγείας των χρηστών της εφαρμογής).

Στη δεύτερη περίπτωση έχουμε την κρυπτογράφηση από άκρο-σε-άκρο (end-to-end encryption). Με την τεχνική αυτή, τα δεδομένα παραμένουν προστατευμένα ακόμα και αν υποκλαπούν από τον εξυπηρετητή ή πραγματοποιηθεί οποιαδήποτε άλλη αθέμιτη πρόσβαση και επεξεργασία επ' αυτών, αφού ούτε ο εξυπηρετητής έχει δυνατότητα ανάγνωσης επ' αυτών. Αν και αυτή η προσέγγιση φαίνεται αναγκαία για την προστασία των δεδομένων, εν τούτοις δεν χρησιμοποιείται ακόμη ευρέως: ο κύριος λόγος είναι ότι δημιουργεί προβλήματα στη λειτουργικότητα των εφαρμογών, όπως στην ευχερή ανάκτηση των δεδομένων από τον χρήστη, στην κοινή χρήση των δεδομένων (data sharing) αλλά και στην εκτέλεση υπολογισμών σε αυτά. Οι κλασικές, ευρέως χρησιμοποιούμενες, κρυπτογραφικές τεχνικές, δεν επαρκούν στο να παρέχουν κρυπτογράφηση από-άκρο-σε-άκρο και ταυτοχρόνως, να διασφαλίζουν το ότι η εφαρμογή μπορεί να παρέχει όλες τις ανωτέρω λειτουργίες.

Τα παραπάνω έχουν οδηγήσει την ερευνητική κοινότητα να στραφεί προς νέες τεχνολογικές προσεγγίσεις, οι οποίες ενσωματώνουν νέες ειδικές κρυπτογραφικές τεχνικές και δίνουν τη δυνατότητα δημιουργίας λειτουργικών εφαρμογών, ενώ παράλληλα επιτυγχάνουν κρυπτογράφηση από-άκρο-σε-άκρο – η οποία θεωρείται βέλτιστη προσέγγιση ως προς τη διασφάλιση της εμπιστευτικότητας των δεδομένων. Μία τέτοια, πρόσφατη, προσέγγιση, που ακούει στο όνομα Mylar και πρόκειται για μία ερευνητική πλατφόρμα ανοιχτού κώδικα (open source) από ερευνητές στο Πανεπιστήμιο MIT (Massachusetts Institute of Technology), είναι και το κύριο αντικείμενο παρούσας διατριβής.

1.1 Αντικείμενο και δομή της διατριβής

Αντικείμενο της παρούσας διατριβής θα είναι η μελέτη αυτής της νέας τεχνολογικής προσέγγισης που προτείνεται μέσω της πλατφόρμας Mylar. Στόχος της διατριβής είναι η λεπτομερής ανάλυση των χαρακτηριστικών της πλατφόρμας και του τρόπου λειτουργίας της, όπως και των θεμάτων ασφαλείας που ενδεχομένως ανακύπτουν. Η πλατφόρμα μελετάται τόσο σε θεωρητικό επίπεδο με ανάλυση των ειδικών κρυπτογραφικών χαρακτηριστικών της, όσο και σε πρακτικό επίπεδο με ενσωμάτωσή της σε μία υπάρχουσα εφαρμογή. Έμφαση επίσης θα δοθεί στον προσδιορισμό του πρόσθετου κόστους που εισάγει για τις εφαρμογές η ενσωμάτωση αυτής της πλατφόρμας, προκειμένου να αποτιμηθεί κατά πόσον η ενίσχυση της ασφάλειας που επιτυγχάνεται δημιουργεί προβλήματα στην απόδοση της εφαρμογής. Για την εν λόγω αποτίμηση θα χρησιμοποιηθεί η εφαρμογή συνομιλιών (chat application) kChat. Ουσιαστικά, απώτερος σκοπός της διατριβής είναι η διερεύνηση των ερωτημάτων που αφορούν στην δυνατότητα δημιουργίας εφαρμογών ιστού, οι οποίες να προστατεύουν πλήρως τα ευαίσθητα δεδομένα των χρηστών τους από-άκρο-σε-άκρο και παράλληλα να είναι λειτουργικές.

Ειδικότερα, η δομή της διατριβής έχει ως εξής:

Στο Κεφάλαιο 2 παρουσιάζονται οι βασικές αρχές της κλασικής κρυπτογραφίας. Αρχικά περιγράφονται οι κρυπτογραφικοί στόχοι και οι βασικοί κανόνες ασφάλειας. Στη συνέχεια παρουσιάζονται τα είδη των κλασικών αλγορίθμων κρυπτογράφησης και γίνεται μια μικρή περιγραφή του πρότυπου αλγορίθμου AES, ο οποίος είναι ένας από

τους αλγόριθμους που χρησιμοποιούνται από την πλατφόρμα Mylar. Επιπλέον, περιγράφονται οι ψηφιακές υπογραφές και τα ψηφιακά πιστοποιητικά τα οποία είναι απαραίτητα εργαλεία της κρυπτογραφίας.

Στο Κεφάλαιο 3 αρχικά αναλύονται οι κύριοι τρόποι προστασίας των δεδομένων και αναδεικνύεται η ανάγκη για κρυπτογράφηση από-άκρο-σε-άκρο. Στη συνέχεια παρουσιάζονται ειδικές κρυπτογραφικές τεχνικές που δίνουν την δυνατότητα υπολογισμών και αναζητήσεων στα κρυπτογραφημένα δεδομένα.

Στο Κεφάλαιο 4 παρουσιάζεται η πλατφόρμα Mylar και γίνεται ενδελεχής ανάλυση των χαρακτηριστικών της, του τρόπου λειτουργίας και ενσωμάτωσής της καθώς και των κρυπτογραφικών τεχνικών που χρησιμοποιεί. Ιδιαίτερη αναφορά γίνεται στο ειδικό κρυπτογραφικό σύστημα MK (Multi-Key Searchable Encryption) [30] και στον τρόπο λειτουργίας του.

Στο Κεφάλαιο 5 παρουσιάζεται και αποτιμάται η πρακτική εφαρμογή της πλατφόρμας Mylar στην εφαρμογή kChat. Αρχικά μελετάται ο τρόπος ενσωμάτωσης του Mylar στο kChat και αναδεικνύεται η ευκολία και απλότητά του. Στη συνέχεια παρουσιάζονται η μεθοδολογία αλλά και τα αποτελέσματα των σχετικών μετρήσεων που έγιναν ως προς την απόδοση των βασικών λειτουργιών της τροποποιημένης εφαρμογής, με εκτέλεση ενδεικτικών σεναρίων χρησιμοποίησης της εφαρμογής μέσω κατάλληλων εργαλείου λογισμικού. Από την ανάλυσή μας καταδεικνύεται ότι το πρόσθετο υπολογιστικό κόστος είναι στην πράξη, για το χρήστη, αμελητέο. Επιπλέον, παρουσιάζονται οι μετρήσεις που αφορούν τις απαιτήσεις σε χωρητικότητα της τροποποιημένης εφαρμογής λόγω της κρυπτογράφησης των δεδομένων.

Τέλος, στο Κεφάλαιο 6 πραγματοποιείται μία σύνοψη της διατριβής, με καταγραφή των βασικών συμπερασμάτων της και των πιθανών μελλοντικών ερευνητικών κατευθύνσεων.

Κεφάλαιο 2
Βασικές αρχές κρυπτογραφίας

Η κρυπτογραφία είναι ένας κλάδος της επιστήμης της κρυπτολογίας, ο οποίος ασχολείται με την μελέτη της ασφαλούς επικοινωνίας. Η κρυπτογραφία παρέχει τεχνικές μετατροπής ενός μηνύματος σε ακατάληπτη μορφή με κύριο στόχο την εμπιστευτικότητα του μηνύματος, έτσι ώστε η ανάγνωση του μηνύματος να είναι δυνατή μόνο από αυτούς που έχουν την εξουσιοδότηση.

Η χρήση της ξεκινά από την αρχαιότητα, με απλές τεχνικές που βασίζονται κυρίως σε απλές αντικαταστάσεις ή/και αντιμεταθέσεις των συμβόλων του μηνύματος, και συνεχίζεται μέχρι σήμερα όπου υπάρχει πληθώρα σύνθετων, διαρκώς εξελισσόμενων, κρυπτογραφικών αλγορίθμων. Η ραγδαία εξέλιξη των τηλεπικοινωνιών και εξάπλωσή τους σε τεράστιο εύρος εφαρμογών ανέδειξε την ιδιάζουσα αξία της έννοιας του απορρήτου των επικοινωνιών και κατέστησε την κρυπτογραφία αντικείμενο έντονου ερευνητικού ενδιαφέροντος.

Το εύρος εφαρμογών της κρυπτογραφίας είναι εξαιρετικά μεγάλο. Εφαρμογές της βρίσκουμε στην ασφαλή επικοινωνία στο διαδίκτυο όπως το ηλεκτρονικό ταχυδρομείο, και οι ηλεκτρονικές συναλλαγές. Συναντάται επίσης σε μηχανισμούς πρόσβασης όπου πιστοποιείται η ταυτότητα του χρήστη, όπως κατά την σύνδεση σε έναν υπολογιστή, σε

μία εφαρμογή ή ακόμα και στο μηχάνημα αυτόματων αναλήψεων (ATM) μίας τράπεζας. Η κρυπτογραφία συναντάται επίσης σε διαδικασίες που απαιτούν την ανωνυμία αλλά διασφαλίζουν την ακεραιότητα της διαδικασίας όπως ηλεκτρονικές δημοπρασίες και ηλεκτρονικές ψηφοφορίες. Τέλος, η κρυπτογραφία χρησιμοποιείται στα δίκτυα τηλεφωνίας, στα ασύρματα δίκτυα, όπως και σε στρατιωτικά και διπλωματικά δίκτυα.

Παράλληλα με την κρυπτογραφία αναπτύχθηκε και ο κλάδος της κρυπτανάλυσης, δηλαδή η μελέτη μαθηματικών τεχνικών που στοχεύουν στην καταστρατήγηση των κρυπτογραφικών μεθόδων, προκειμένου να πληγεί η ασφάλεια της πληροφορίας. Η κρυπτογραφία μαζί με την κρυπτανάλυση αποτελούν την επιστήμη της κρυπτολογίας.

2.1 Κρυπτογραφικοί στόχοι και ορολογία

Στη συνέχεια θα περιγραφούν οι στόχοι της κρυπτογραφίας, οι βασικοί κρυπτογραφικοί όροι και οι βασικοί ορισμοί κρυπτογραφικής ασφάλειας.

2.1.1 Κρυπτογραφικοί στόχοι

Αν και κύριος στόχος της κρυπτογραφίας παραμένει η εμπιστευτικότητα, η εξέλιξη της διεύρυνε τους στόχους της. Έτσι σήμερα οι κύριοι στόχοι της είναι οι παρακάτω.

1. **Εμπιστευτικότητα:** Η διατήρηση της πληροφορίας μακριά από μη εξουσιοδοτημένους χρήστες.

2. **Ακεραιότητα:** Η διασφάλιση του ότι η πληροφορία δεν έχει παραποιηθεί από μη εξουσιοδοτημένους χρήστες.

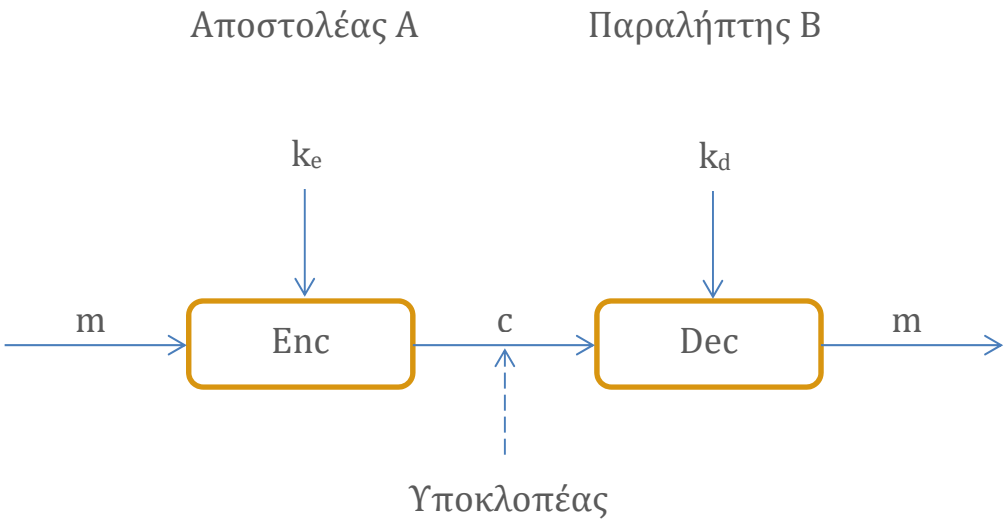
3. **Πιστοποίηση:** Η διασφάλιση ότι ο αποστολέας και ο παραλήπτης μπορούν να εξακριβώνουν τις ταυτότητές τους καθώς και την πηγή και τον προορισμό της πληροφορίας.

4. **Μη αποποίηση:** Η διασφάλιση ότι κανείς δεν θα μπορεί να αποποιηθεί προηγούμενες απόπειρες ή ενέργειές του.

2.1.2 Βασικοί όροι

Στη συνέχεια θα περιγραφούν οι βασικοί κρυπτογραφικοί όροι μέσω της περιγραφής ενός τυπικού κρυπτογραφικού συστήματος για την ασφαλή αποστολή ενός μηνύματος.

Αρχικά ο αποστολέας Α, μέσω ενός μετασχηματισμού Enc και μιας ποσότητας k_e, η οποία ονομάζεται κλειδί κρυπτογράφησης, μετατρέπει ένα αρχικό μήνυμα m σε ένα κρυπτογραφημένο μήνυμα ή αλλιώς κρυπτοκείμενο c, όπου c = Enc(k_e, m). Η διαδικασία αυτή καλείται κρυπτογράφηση. Το κρυπτοκείμενο μπορεί πλέον να σταλεί στον εξουσιοδοτημένο παραλήπτη του Β μέσα από ένα δημόσιο κανάλι μετάδοσης στο οποίο όλοι (ακόμα και οι επίδοξοι υποκλοπείς) έχουν πρόσβαση. Ο παραλήπτης Β του κρυπτογραφημένου μηνύματος c εφαρμόζει με τη σειρά του έναν αντίστροφο μετασχηματισμό Dec, στον οποίο υπεισέρχεται ένα κλειδί αποκρυπτογράφησης k_d, αποτέλεσμα του οποίου είναι το αρχικό μήνυμα m = Dec(k_d , c). Αυτή ακριβώς είναι η διαδικασία της αποκρυπτογράφησης. Το γενικό διάγραμμα λειτουργίας ενός κρυπτογραφικού συστήματος απεικονίζεται στο παρακάτω Σχήμα 2.1.



Σχήμα 2.1: Γενικό διάγραμμα λειτουργίας κρυπτογραφικού συστήματος.

Στην περίπτωση συμμετρικού συστήματος ισχύει ότι $k_e = k_d$, ενώ στην περίπτωση ασύμμετρου συστήματος τα k_e, k_d είναι το δημόσιο και ιδιωτικό κλειδί του παραλήπτη αντίστοιχα.

2.1.3 Ασφάλεια

Μια βασική αρχή που αφορά τον σχεδιασμό των κρυπτογραφικών συστημάτων οφείλεται στον Auguste Kerckhoffs (1883) [20]. Σύμφωνα με την αρχή αυτή, η ασφάλεια ενός κρυπτογραφικού συστήματος θα πρέπει βασίζεται μόνο στη μυστικότητα του κλειδιού, και θα πρέπει να παραμένει ασφαλής ακόμα και αν αποκαλυφθούν όλες οι άλλες λεπτομέρειες της σχεδίασής του. Η αρχή αυτή βασίστηκε στο γεγονός ότι η αλλαγή και αναδιανομή του κλειδιού είναι πολύ πιο εύκολη διαδικασία από την αλλαγή και αναδιανομή του αλγόριθμου.

Το 1949 ο Claude Shannon, στη θεμελιώδη για την κρυπτογραφία εργασία του [34] ορίζει την έννοια του απόλυτα ασφαλούς κρυπτογραφικού συστήματος. Ορίζει ένα σύστημα κρυπτογράφησης ως απεριόριστα ασφαλές (unconditionally secure) αν το παραγόμενο κρυπτοκείμενο, ανεξαρτήτως του μεγέθους του, δεν περιέχει πληροφορία αρκετή ώστε να προσδιοριστεί μονοσήμαντα το αρχικό μήνυμα, ακόμα και αν ο επίδοξος υποκλοπέας διαθέτει απεριόριστη υπολογιστική ισχύ. Η πρακτική όμως υλοποίηση ενός τέτοιου συστήματος δεν είναι δυνατή. Εναλλακτικά, ο Shannon εισήγαγε τις έννοιες της σύγχυσης (confusion) και της διάχυσης (diffusion) ως κύρια κριτήρια που πρέπει να πληρούνται από ένα καλό κρυπτογραφικό σύστημα και παράλληλα πρότεινε ένα δίκτυο αντικατάστασης-αντιμετάθεσης για την ικανοποίηση, αντίστοιχα των κριτηρίων αυτών. Τα κριτήρια αυτά εξακολουθούν να παίζουν σημαντικό ρόλο στη σχεδίαση κρυπτογραφικών συστημάτων [36].

Στο πλαίσιο καθορισμού ενός μοντέλου αποδείξιμης ασφάλειας (provable security) ενός αλγορίθμου, προκειμένου να καταδειχθεί ένας μαθηματικός τρόπος αποτίμησης και θεμελίωσης της «πραγματικής ασφάλειας» ενός κρυπτοσυστήματος, οι Goldwasser-Micali εισάγουν το 1984 τον όρο του σημασιολογικά ασφαλούς (semantically secure) κρυπτογραφικού συστήματος. Σε ένα τέτοιο σύστημα, η πληροφορία που πιθανώς περιέχει το παραγόμενο κρυπτοκείμενο σχετικά με το αρχικό, δεν είναι δυνατό να εξαχθεί μέσω της βέλτιστης κρυπταναλυτικής τεχνικής και με τη χρήση υπολογιστικής ισχύος πολύ μεγαλύτερης από αυτή που διαθέτει ένας επιτιθέμενος. Επιπλέον αποδεικνύουν ότι η σημασιολογική ασφάλεια είναι ισοδύναμη με έναν άλλο ορισμό ασφάλειας που ονομάζεται μη διακρισιμότητα σε επιθέσεις επιλεγμένου αρχικού κειμένου (IND-CPA). Ουσιαστικά, ένα κρυπτογραφικό σύστημα είναι IND-CPA ασφαλές αν γνωρίζοντας το κρυπτοκείμενο c και επιπροσθέτως έχουμε την ειδικότερη γνώση ότι

το μήνυμα μπορεί να λάβει μόνο μία εκ δύο συγκεκριμένων τιμών m_0 και m_1, η γνώση του c – σε συνδυασμό με τη γνώση του αλγορίθμου που χρησιμοποιήθηκε – δεν πρέπει να επιτρέπει ποτέ το να θεωρείται ένα εκ των δύο μηνυμάτων m_0 και m_1 περισσότερο πιθανό από το άλλο ως προς το ότι μπορεί να είναι αυτό το αρχικό μήνυμα.

Στη συνέχεια θα περιγραφούν οι διάφορες κατηγορίες κρυπτογραφικών συστημάτων και τα σημαντικότερα από αυτά.

2.2 Κρυπτογραφικά συστήματα

Τα κρυπτογραφικά συστήματα διακρίνονται σε δύο βασικές κατηγορίες.

1. **Συστήματα συμμετρικής κρυπτογράφησης ή συστήματα ιδιωτικού κλειδιού:** στα οποία το κλειδί κρυπτογράφησης συμπίπτει με το κλειδί αποκρυπτογράφησης, δηλαδή ισχύει $k_e = k_d$.

2. **Συστήματα ασύμμετρης κρυπτογράφησης ή συστήματα δημόσιου κλειδιού:** στα οποία γίνεται χρήση ζεύγους κλειδιών, επιλεγμένα ώστε η κρυπτογράφηση με το ένα να απαιτεί αποκρυπτογράφηση με το άλλο.

Τα πρώτα αναφέρονται και ως κλασικά συστήματα κρυπτογράφησης, σε αντιπαραβολή με τα νεότερα συστήματα δημόσιου κλειδιού που εμφανίστηκαν το 1976.

2.2.1 Συμμετρική κρυπτογράφηση

Οι αλγόριθμοι των συμμετρικών συστημάτων κρυπτογράφησης διακρίνονται σε δύο κατηγορίες συστήματα διακρίνονται σε δύο βασικές κατηγορίες.

1. **Αλγόριθμοι τμήματος (block ciphers):** στους αλγορίθμους αυτούς το αρχικό μήνυμα χωρίζεται σε τμήματα και στη συνέχεια το κάθε τμήμα κρυπτογραφείται ξεχωριστά.

2. **Αλγόριθμοι ροής (stream ciphers):** στους αλγορίθμους αυτούς η κρυπτογράφηση του αρχικού μηνύματος γίνεται μεταβάλλοντας ένα χαρακτήρα, στην πράξη bit ή byte, την φορά.

Αλγόριθμοι τμήματος

Οι αλγόριθμοι τμήματος χρησιμοποιούνται από τα περισσότερα κρυπτογραφικά συστήματα.

Σε έναν αλγόριθμο τμήματος το αρχικό μήνυμα χωρίζεται σε διαδοχικά τμήματα ίσου μεγέθους και το κάθε τμήμα κρυπτογραφείται ξεχωριστά. Μία τυπική τιμή για το μέγεθος του τμήματος σε αλγορίθμους αυτής της κατηγορίας είναι 128 bits. Η λειτουργία αυτών των αλγορίθμων είναι επαναληπτική. Ουσιαστικά το αρχικό τμήμα κρυπτογραφείται μέσα από διάφορους διαδοχικούς γύρους (rounds), όπου σε κάθε γύρο εκτελούνται οι ίδιοι μετασχηματισμοί και χρησιμοποιείται διαφορετικό τμήμα του κλειδιού. Κάθε ένας από τους γύρους περιέχει δομικές μονάδες για την εκτέλεση αντικαταστάσεων (S-Box) και αντιμεταθέσεων (P-Box) των bit του τμήματος με σκοπό την εισαγωγή μεγαλύτερης σύγχυσης και διάχυσης αντίστοιχα.

Οι κυριότεροι αλγόριθμοι τμήματος είναι οι DES (Data Encryption Standard) και AES (Advance Encryption Standard).

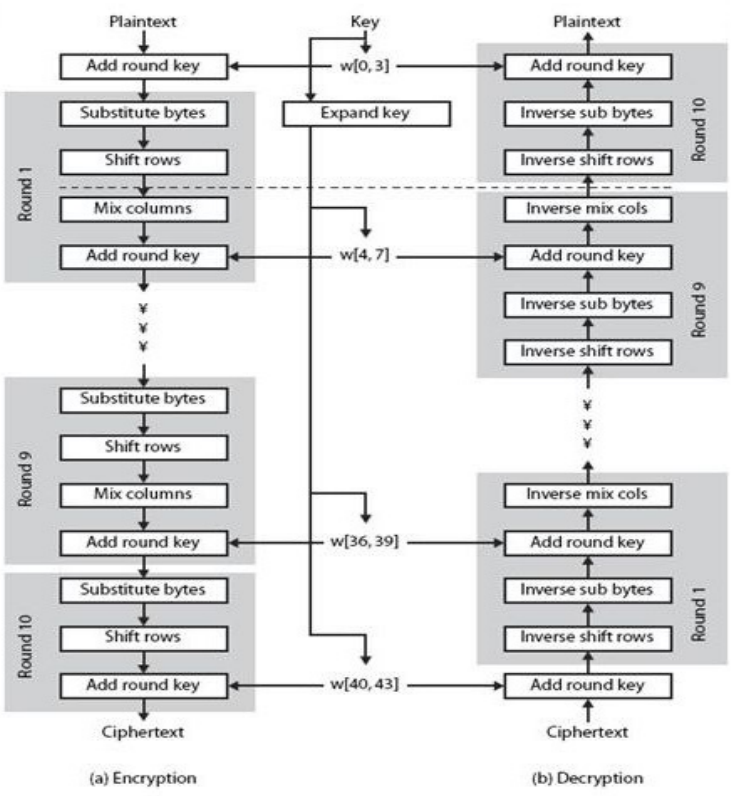
DES

Ένας από τους για πολλά χρόνια ευρέως χρησιμοποιημένους αλγόριθμους τμήματος είναι, ο Data Encryption Algorithm (DES). Ο DES καθιερώθηκε σαν πρότυπο κρυπτογράφησης το 1976 από το NIST (National Institute of Standards and Technology). Ο αλγόριθμος χρησιμοποιεί κλειδί μεγέθους 56-bits και πραγματοποιεί 16 γύρους κρυπτογραφώντας τμήματα των 64-bits. Ο DES παρέμεινε ασφαλής αλγόριθμος σε επιθέσεις εξαντλητικής αναζήτησης για περίπου 20 έτη. Το μέγεθος του κλειδιού του όμως δεν ήταν πλέον ικανό να αποτρέπει επιθέσεις εξαντλητικής αναζήτησης κλειδιού και, πλέον, δεν χρησιμοποιείται.

AES

Το 1997, ο NIST προσκάλεσε δημόσια για ορισμό νέου προτύπου κρυπτογράφησης που θα λάμβανε το όνομα Advanced Encryption Standard (AES), προς αντικατάσταση του DES. Ο διαγωνισμός ολοκληρώθηκε το 2000 με νικητή τον αλγόριθμο Rijndael ο οποίος θα αποτελούσε το AES. Η επιλογή έγινε τόσο με βάση την ασφάλεια, δηλαδή την αντοχή στις επιθέσεις που πραγματοποιήθηκαν και την θεωρητική άμυνα απέναντι σε μεθόδους γραμμικής και διαφορικής κρυπτανάλυσης, όσο και με βάση την ταχύτητα, την απλότητα και την ευελιξία του Rijndael.

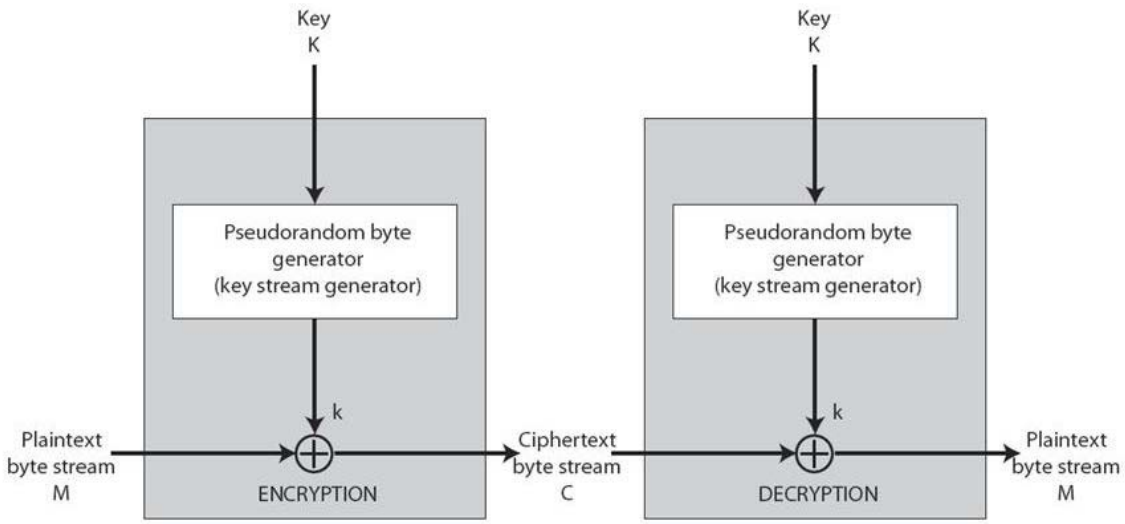
Ο AES κρυπτογραφεί τμήματα των 128-bit και έχει τη δυνατότητα επιλογής κλειδιού κρυπτογράφησης μεταξύ των 128-bits, 192-bitsκαι 256-bits. Η λειτουργία της κρυπτογράφησης περιγράφεται κατάλληλα με χρήση πινάκων από bytes μεγέθους 4x4. Ο αριθμός των γύρων (10 -12 - 14) εξαρτάται από το μέγεθος του κλειδιού. Τέλος σε κάθε γύρο, συμβαίνουν τα εξής: αντικατάσταση των bytes με τη χρήση S-box, ολίσθηση γραμμών, ανάμειξη δεδομένων σε κάθε στήλη του πίνακα και η πράξη XOR του πίνακα με το κλειδί του γύρου. Η λειτουργία του AES απεικονίζεται στην Εικόνα 2.2



Εικόνα 2.2: Η κρυπτογράφηση και αποκρυπτογράφηση στον αλγόριθμο AES (πηγή: Stallings, W., 2006. Cryptography and network security [36]).

Αλγόριθμοι ροής

Οι αλγόριθμοι ροής κρυπτογραφούν μεμονωμένους χαρακτήρες (bits ή bytes) του αρχικού μηνύματος, ένα κάθε φορά. Χρησιμοποιούν ένα κλειδί το οποίο παράγεται με τυχαίο τρόπο και έχει πολύ μικρό μέγεθος, ανεξάρτητο από το μήνυμα προς κρυπτογράφηση. Το κλειδί εισάγεται σε μία γεννήτρια ψευδοτυχαίας ακολουθίας (keystream generator) η οποία παράγει μια ροή από χαρακτήρες που ονομάζεται κλειδοροή (keystream). Στη συνέχεια κάθε χαρακτήρας της κλειδοροής προστίθεται, με πράξη XOR, σε ένα χαρακτήρα του αρχικού μηνύματος και προκύπτει ένας χαρακτήρας κρυπτογραφημένου κειμένου. Η αποκρυπτογράφηση λόγω της πράξης XOR είναι το ίδιο απλή, κάθε χαρακτήρας της κλειδοροής προστίθεται, με πράξη XOR, σε κάθε χαρακτήρα του κρυπτογραφημένου κειμένου και δίνει το αρχικό μήνυμα.



Εικόνα 2.3: Διάγραμμα αλγόριθμου ροής (πηγή: Stallings, W., 2006. Cryptography and network security. [36]).

Ο πλέον γνωστός αλγόριθμος ροής είναι το σημειωματάριο μιας χρήσης (one-time-pad) ο οποίος βασίζεται στον αλγόριθμο Vernam [39]. Το κλειδί του αλγορίθμου είναι μία τυχαία ακολουθία bits ίσου μεγέθους με το μήνυμα. Η τυχαιότητα της ακολουθίας και το μέγεθος της είναι απαραίτητες προϋποθέσεις για την ασφάλεια του αλγόριθμου. Ο Shannon [34] απέδειξε ότι ο αλγόριθμος του αυτός είναι απεριόριστα ασφαλής και παραμένει μέχρι σήμερα ο μοναδικός με αυτή την ιδιότητα. Αν και είναι ιδανικός

αλγόριθμος κρυπτογράφησης στην πράξη δεν μπορεί να εφαρμοστεί καθώς για μεγάλα μηνύματα απαιτούνται εξίσου μεγάλα κλειδιά με αποτέλεσμα τη δύσκολη παραγωγή και ανταλλαγή τους. Όλοι οι αλγόριθμοι ροής σχεδιάζονται με τέτοιο τρόπο ώστε να προσομοιάζουν τη λειτουργία του one-time-pad.

Αν και από τους αλγόριθμους ροής κανένας μέχρι τώρα δεν έχει οριστεί ως διεθνώς αποδεκτό πρότυπο κρυπτογράφησης μιας και θεωρούνται λιγότερο ασφαλείς από τους αλγόριθμους τμήματος, η εύκολη υλοποίηση τους σε υλικό (hardware) και η πολύ καλή απόδοση παράλληλα με τις ελάχιστες απαιτήσεις για ενδιάμεση αποθήκευση, τους καθιστούν πολύ χρήσιμους και απαραίτητους σε πολλές εφαρμογές, όπως οι τηλεπικοινωνίες.

Οι αλγόριθμοι ροής χρησιμοποιούνται σε πλήθος εφαρμογών τη σημερινή εποχή. Ένας από τους πιο σημαντικούς για πολλά χρόνια ήταν – και μάλιστα ακόμα χρησιμοποιείται σε ορισμένες περιπτώσεις - είναι ο RC4, ο οποίος χρησιμοποιήθηκε για πολλά χρόνια στο πρωτόκολλο SSL/TLS, αλλά και στα ασύρματα δίκτυα (WEP, WAP). Πλέον, για τον RC4 έχει διατυπωθεί ότι πρέπει να μη χρησιμοποιείται, λόγω κάποιων ευπαθειών που έχουν ανακαλυφθεί. Άλλοι γνωστοί αλγόριθμοι ροής είναι ο A5/1 ο οποίος χρησιμοποιήθηκε στο σύστημα GSM (κινητή τηλεφωνία) καθώς και ο E0 ο οποίος χρησιμοποιείται στο πρωτόκολλο Bluetooth.

2.2.2 Ασύμμετρη κρυπτογράφηση

Οι W.Diffie και M.Hellman το 1976 πρότειναν μια κρυπτογραφική μέθοδο διαφορετικής λογικής από αυτή της συμμετρικής κρυπτογράφησης [10]. Το μοντέλο αυτό ονομάστηκε κρυπτογράφηση δημόσιου κλειδιού ή ασύμμετρη κρυπτογράφηση. Στην ασύμμετρη κρυπτογράφηση κάθε χρήστης έχει ένα ζευγάρι κλειδιών k_e και k_d κατάλληλα επιλεγμένα έτσι ώστε το ένα αντιστρέφει το άλλο. Το k_e είναι διαθέσιμο σε όλους και ονομάζεται δημόσιο κλειδί ενώ το k_d παραμένει μυστικό και ονομάζεται ιδιωτικό κλειδί.

Όταν ο αποστολέας A θέλει να στείλει ένα μήνυμα στον παραλήπτη B τότε κρυπτογραφεί το μήνυμα m με τη χρήση του δημόσιου κλειδιού k_e του B, το οποίο είναι γνωστό σε όλους, και δημιουργεί το κρυπτοκείμενο C. Στη συνέχεια το C αποστέλλεται στον παραλήπτη B όπου και αποκρυπτογραφείται με τη χρήση του ιδιωτικού κλειδιού

k_d του Β. Η αποκρυπτογράφηση του C μπορεί να γίνει μόνο από τον Β, μιας και μόνο αυτός έχει στην κατοχή του το k_d. Η γενική λειτουργία του συστήματος απεικονίζεται στο Σχήμα 2.1, όπου σε αυτήν την περίπτωση, σε αντίθεση με τη συμμετρική κρυπτογράφηση, ισχύει $k_e \neq k_d$.

Το κύριο χαρακτηριστικό και πλεονέκτημα της ασύμμετρης κρυπτογράφησης είναι ότι η ασφαλής ανταλλαγή μηνυμάτων μπορεί να γίνει χωρίς καμία εκ των προτέρων ανταλλαγή μυστικής πληροφορίας, σε αντίθεση με την συμμετρική κρυπτογράφηση όπου είναι προαπαιτούμενη η ασφαλής ανταλλαγή του μυστικού κλειδιού.

Όμως, αυτό το χαρακτηριστικό τους κάνει ευάλωτους σε ένα συγκεκριμένο είδος επιθέσεων, στις οποίες ένας επίδοξος εισβολέας μπορεί να χρησιμοποιήσει το δημόσιο κλειδί κάποιου άλλου προσποιούμενος πως είναι αυτός. Η λύση δίνεται μέσω των ψηφιακών πιστοποιητικών που θα περιγράψουμε παρακάτω.

Όλοι οι αλγόριθμοι ασύμμετρης κρυπτογράφησης βασίζουν την ασφάλειά τους σε μαθηματικά προβλήματα από το χώρο της θεωρίας πολυπλοκότητας που είναι γνωστά για τη δυσκολία τους, όπως το πρόβλημα διακριτού λογαρίθμου και το πρόβλημα παραγοντοποίησης μεγάλων ακεραίων αριθμών. Χαρακτηριστικό αυτών των αλγόριθμων είναι το μεγάλο μήκος κλειδιού και η σχετικά μικρή ταχύτητα τους. Συγκριτικά με τους αλγόριθμους συμμετρικής κρυπτογράφησης μπορεί και να είναι ακόμα και 1000 φορές πιο αργοί.

Παρά τα μειονεκτήματά της η ασύμμετρη κρυπτογράφηση δίνει λύση σε σημαντικά προβλήματα που αφορούν στόχους της κρυπτογραφίας, όπως η πιστοποίηση, η ακεραιότητα και η μη αποποίηση. Αυτό γίνεται μέσω της δημιουργίας ψηφιακών υπογραφών τις οποίες θα περιγράψουμε παρακάτω.

Το σημαντικότερο όμως είναι ότι δίνει λύση στο καθολικό και πολύ σημαντικό πρόβλημα της συμμετρικής κρυπτογράφησης το οποίο είναι η διανομή του μυστικού κλειδιού. Έτσι, το μυστικό κλειδί ενός συμμετρικού αλγόριθμου κρυπτογραφείται μέσω ενός ασύμμετρου αλγόριθμου και διανέμεται με ασφάλειας στους επικοινωνούντες. Στη συνέχεια το μυστικό κλειδί χρησιμοποιείται μεταξύ των επικοινωνούντων για την ανταλλαγή μηνυμάτων μέσω του συμμετρικού αλγόριθμου, ο οποίος υπερτερεί σε απόδοση.

Στην πράξη σήμερα εφαρμόζεται συνδυασμός των μεθόδων ασύμμετρης και συμμετρικής κρυπτογράφησης, όπως στο πρωτόκολλο SSL και τον διάδοχο του TLS. Με τον τρόπο αυτό εκμεταλλευόμαστε τα προτερήματα και των δύο κρυπτογραφικών συστημάτων.

Ο πιο γνωστός και ευρέως χρησιμοποιούμενος αλγόριθμος αυτής της κατηγορίας είναι ο αλγόριθμος RSA, των Rivest-Shamir-Adleman [32]. Χρησιμοποιείται τόσο για κρυπτογράφηση, όσο και για δημιουργία ψηφιακής υπογραφής και είναι ο βασικός αλγόριθμος δημόσιου κλειδιού στο πρωτόκολλο SSL/TLS. Άλλοι αλγόριθμοι ασύμμετρης κρυπτογράφησης είναι οι El Gamal, DSA, Paillier, τα κρυπτοσυστήματα ελλειπτικών καμπυλών και άλλοι.

2.2.3 Συναρτήσεις κατακερματισμού (Hash - MAC functions)

Οι συναρτήσεις κατακερματισμού δέχονται σαν είσοδο μήνυμα οποιουδήποτε μεγέθους και επιστρέφουν μία έξοδο σταθερού και σχετικά μικρού μεγέθους. Το αποτέλεσμα ονομάζεται σύνοψη ή αποτύπωμα (message digest, fingerprint) του μηνύματος. Η ανάκτηση του αρχικού μηνύματος από το αποτύπωμά του είναι πρακτικά ανέφικτη, δηλαδή οι συναρτήσεις κερματισμού είναι μη αντιστρεπτές συναρτήσεις (one way functions). Επίσης, είναι πρακτικά ανέφικτο να βρεθούν δύο διαφορετικά κείμενα με το ίδιο αποτύπωμα.

Έτσι, ο αποστολέας, μέσω μια συνάρτησης κατακερματισμού, δημιουργεί ένα αποτύπωμα MD του μηνύματος το οποίο αποστέλλει μαζί με το μήνυμα στον παραλήπτη. Ο παραλήπτης με την σειρά του υπολογίζει και αυτός, μέσω της ίδιας συνάρτησης, το αποτύπωμα MD’ του μηνύματος που έχει παραλάβει. Συγκρίνοντας τα MD και MD’ μπορεί να καταλάβει αν το μήνυμα έχει παραποιηθεί κατά την μετάδοση του.

Με τις συναρτήσεις κατακερματισμού διασφαλίζεται η ακεραιότητα (integrity) του μεταδιδόμενου μηνύματος, δηλαδή το ότι το μήνυμα δεν παραποιήθηκε κατά τη μετάδοση του, όπως και η μη αποποίηση, δηλαδή ο αποστολέας του μηνύματος δεν μπορεί να ισχυρισθεί ότι έστειλε κάποιο άλλο μήνυμα και όχι αυτό.

Ειδική περίπτωση των συναρτήσεων κατακερματισμού είναι οι συναρτήσεις MAC (Message Authentication Code), στις οποίες υπεισέρχεται κάποιο μυστικό κλειδί.

Οι γνωστότερες συναρτήσεις κατακερματισμού είναι οι SHA-1, SHA-2, και SHA-3, με την τελευταία να έχει καθιερωθεί ως πρότυπο από το NIST.

2.2.4 Ψηφιακή υπογραφή

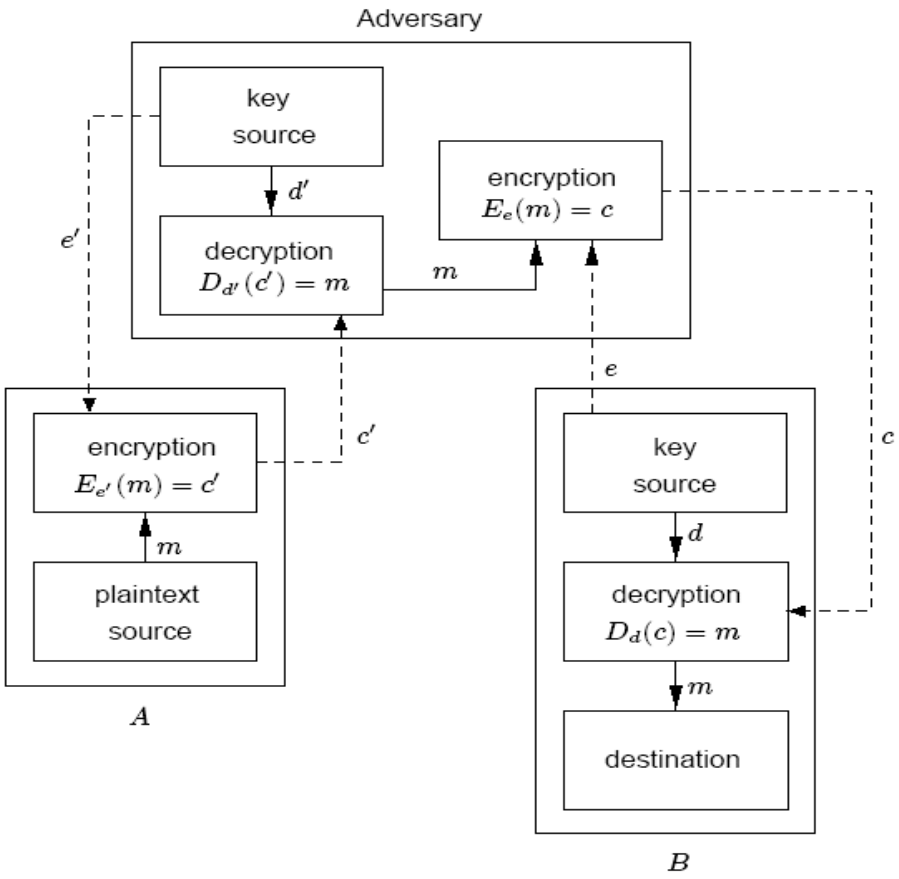
Με τις ψηφιακές υπογραφές μπορούμε να πετύχουμε την επαλήθευση της ταυτότητας του αποστολέα αλλά ταυτόχρονα και της ακεραιότητας του μηνύματος. Βασικά εργαλεία για την δημιουργία τους είναι οι ασύμμετροι αλγόριθμοι και οι συναρτήσεις κατακερματισμού.

Για τη δημιουργία της ψηφιακής υπογραφής αρχικά δημιουργείται ένα αποτύπωμα (MD) του μηνύματος. Το MD κρυπτογραφείται, με το ιδιωτικό κλειδί του αποστολέα. Έτσι προκύπτει έτσι η ψηφιακή υπογραφή (DS). Η υπογραφή επισυνάπτεται στο αρχικό μήνυμα και όλα μαζί αποστέλλονται στον παραλήπτη. Ο παραλήπτης με την σειρά του υπολογίζει και αυτός, μέσω της ίδιας συνάρτησης, το αποτύπωμα MD’ του μηνύματος που έχει παραλάβει. Στη συνέχεια αποκρυπτογραφώντας την υπογραφή DS με το δημόσιο κλειδί του αποστολέα παράγει το MD. Αν τα MD και MD’ συμπίπτουν, η ταυτότητα του αποστολέα επιβεβαιώνεται, ενώ ταυτόχρονα πιστοποιείται και η ακεραιότητα του μηνύματος.

Οι ασύμμετροι αλγόριθμοι που χρησιμοποιούνται συνήθως για την παραγωγή ψηφιακής υπογραφής είναι οι RSA και DSA.

2.2.5 Ψηφιακά πιστοποιητικά

Στην ασύμμετρη κρυπτογράφηση, η δυνατότητα χρήσης του δημόσιου κλειδιού από οποιονδήποτε δημιουργεί ένα σημαντικό πρόβλημα ασφάλειας. Αυτό φαίνεται ξεκάθαρα από το παρακάτω παράδειγμα, το οποίο απεικονίζεται στην Εικόνα 2.4.



Εικόνα 2.4: Επίθεση man-in-the-middle (πηγή: Menezes, A. et. al. Handbook of Applied Cryptography, CRC Press, 1996).

Ένας επίδοξος εισβολέας μπορεί να ξεγελάσει τον χρήστη Α ότι είναι ο Β, στέλνοντάς του το δικό του δημόσιο κλειδί e'. Έτσι, ο Α στέλνει μηνύματα κρυπτογραφημένα με το e' θεωρώντας ότι μιλά με τον Β. Συνεπώς, ο επιτιθέμενος μπορεί και αποκρυπτογραφεί όλα τα μηνύματα που στέλνει ο Α στον Β.

Επιπλέον, ο Β δεν μπορεί να αντιληφθεί την παρουσία του επιτιθέμενου, μια που αυτός συνεχίζει να του στέλνει κανονικά το μήνυμα m, κρυπτογραφημένο με το δημόσιο κλειδί του Β. Ο Β, λαμβάνοντας ένα μήνυμα, δεν μπορεί να ξέρει με σιγουριά ποιος του το έστειλε. Πρόκειται για την τυπική περίπτωση επίθεσης τύπου «man-in-the-middle».

Είναι φανερό ότι στην ασύμμετρη κρυπτογράφηση προκύπτει η ανάγκη πιστοποίησης της αντιστοιχίας μεταξύ των δημόσιων κλειδιών και των χρηστών.

Η λύση δίνεται μέσω των ψηφιακών πιστοποιητικών που εκδίδονται από μια έμπιστη τρίτη οντότητα, την αρχή πιστοποίησης (Certification Authority – CA). Η αρχή

πιστοποίησης εκδίδει και υπογράφει ψηφιακά πιστοποιητικά δημόσιων κλειδιών. Δηλαδή διασφαλίζει με τεχνικά μέσα ότι ένα δημόσιο κλειδί ανήκει σε μία (και μόνο μία) συγκεκριμένη οντότητα (και συνεπώς ότι η οντότητα αυτή είναι ο νόμιμος κάτοχος του αντίστοιχου ιδιωτικού κλειδιού).

Ένα ψηφιακό πιστοποιητικό περιέχει βασικές ομάδες πεδίων που αφορούν πληροφορίες της εκδότριας αρχής πιστοποίησης, την οντότητα που αφορά το πιστοποιητικό, το δημόσιο κλειδί της οντότητας και την ψηφιακή υπογραφή της εκδότριας αρχής πιστοποίησης.

Το πιο διαδεδομένο πρότυπο για τη σύνταξη ενός πιστοποιητικού είναι το X.509. Το X.509 δημοσιοποιήθηκε για πρώτη φορά το 1988 και είναι μέρος του ευρύτερου καταλόγου παροχής υπηρεσιών X.500. Το X.509 είναι ένα πολύ σημαντικό πρότυπο, γιατί μπορεί να χρησιμοποιηθεί σε πολλές υπηρεσίες. Το X.509 χρησιμοποιείται, μεταξύ άλλων, στα διαδεδομένα πρωτόκολλα ασφαλείας IPSec, SSL/TLS.

Οι αρχές πιστοποίησης συνεργάζονται μεταξύ τους και οργανώνονται με διάφορους τρόπους δημιουργώντας μία υποδομή δημόσιων κλειδιών (ΥΔΚ). Ένα μοντέλο οργάνωσης είναι το ιεραρχικό μοντέλο εμπιστοσύνης, στο οποίο μια αρχή πιστοποίησης μπορεί να πιστοποιείται από μία άλλη. Οι αρχές πιστοποίησης που βρίσκονται στην κορυφή της ιεραρχίας ονομάζονται κύριες (root CAs).

Στο πρότυπο X.509, υιοθετείται ιεραρχικό μοντέλο εμπιστοσύνης. Ένας χρήστης που αναγνωρίζει και αποδέχεται μία αρχή πιστοποίησης CA, αποδέχεται και τα πιστοποιητικά που έχουν εκδοθεί από οποιαδήποτε αρχή πιστοποίησης CA’ για την οποία υπάρχει ένα μονοπάτι εμπιστοσύνης με την CA.

Έτσι, η πρόσβαση ενός χρήστη Α στο δημόσιο κλειδί του Β, γίνεται μέσω του ψηφιακού πιστοποιητικού του Β, το οποίο περιέχει το δημόσιο κλειδί του. Το ψηφιακό πιστοποιητικό το έχει υπογράψει ψηφιακά μία αρχή πιστοποίησης CA και έτσι ο Α είναι σίγουρος ότι το πιστοποιητικό του Β είναι έγκυρο, μιας και εμπιστεύεται την CA, της οποίας το γνήσιο της υπογραφής μπορεί να επιβεβαιώσει.

Κεφάλαιο 3
Ειδικές κρυπτογραφικές τεχνικές

Η αποθήκευση δεδομένων σε εξυπηρετητές αποτελεί κοινό χαρακτηριστικό όλων των εφαρμογών ιστού. Αυτό γεννά και την ανάγκη για την προστασία τους. Η ανάγκη αυτή γίνεται ολοένα και μεγαλύτερη λαμβάνοντας υπόψη μας τα διαρκώς αυξανόμενα κρούσματα διαρροών εμπιστευτικών δεδομένων. Για παράδειγμα, σύμφωνα με την εξαμηνιαία αναφορά της Gemalto [12], το πρώτο εξάμηνο του 2017 διέρρευσαν περίπου 1.9 δισεκατομμύρια εγγραφές εμπιστευτικών δεδομένων, αυξημένες κατά 164% σε σχέση με το προηγούμενο εξάμηνο. Επιπλέον, στην ίδια αναφορά επισημαίνεται ότι μόνο το 4,6% αυτών αφορούσαν κρυπτογραφημένα δεδομένα, τα οποία δεν ήταν άμεσα χρήσιμα για τους εισβολείς.

3.1 Ασφάλεια (επικρατούσα τάση)

Για την ασφάλεια των δεδομένων χρησιμοποιούνται διάφορες τεχνικές. Μερικές από αυτές είναι έλεγχοι σε επίπεδο λειτουργικού συστήματος, επιβολή πολιτικών ασφάλειας, στατική όπως και δυναμική ανάλυση του πηγαίου κώδικα της εφαρμογής

[38], έλεγχοι σε επίπεδο δικτύου[24], αξιόπιστο υλικό [04] και άλλοι. Παρόλες αυτές τις προσπάθειες, η διαρροή των δεδομένων εξακολουθεί να συμβαίνει και να επιβαρύνει με τεράστιο οικονομικό κόστος τις εταιρείες που παρέχουν υπηρεσίες μέσω των εφαρμογών αυτών [40, 42], ενώ όταν η διαρροή αφορά προσωπικά δεδομένα μπορεί να υφίστανται εξαιρετικά δυσμενείς συνέπειες και οι ίδιοι οι χρήστες.

Στη συνέχεια θα αναφέρουμε μερικούς από τους συνηθέστερους λόγους διαρροής δεδομένων και τις δυσκολίες αντιμετώπισης τους.

Μια από τις πιο σημαντικές αιτίες είναι οι ευπάθειες του λογισμικού. Στις σύγχρονες εφαρμογές ιστού, το λογισμικό είναι σύνθετο και αυτό καθιστά εξαιρετικά δύσκολο το να κατασκευασθεί χωρίς ευπάθειες. Η χρήση της γλώσσας JavaScript και οι νέες πλατφόρμες ανάπτυξης όπως Node.js, AngularJS, Meteor κ.α. δίνουν νέες δυνατότητες στους προγραμματιστές, αλλά παράλληλα κάνουν και δυσκολότερο τον έλεγχο του λογισμικού για ευπάθειες. Οι κύριες μέθοδοι αντιμετώπισης των ευπαθειών αυτών είναι η στατική και η δυναμική ανάλυση του κώδικα της εφαρμογής. Όμως στην πραγματικότητα οι μέθοδοι αυτές δεν εφαρμόζονται λεπτομερώς και σε βάθος. Έτσι βλέπουμε στις πρώτες θέσεις της ετήσιας λίστας με τους 10 σημαντικότερους κινδύνους ασφάλειας εφαρμογών ιστού, που δημοσίευσε ο οργανισμός OWASP [26], να βρίσκονται κίνδυνοι που αφορούν ευπάθειες λογισμικού.

Η δεύτερη απειλή εμφανίζεται συχνότερα στο πλαίσιο της υπολογιστικής νέφους (cloud computing). Όλο και περισσότερες εταιρείες χρησιμοποιούν το νέφος - ουσιαστικά κάποια τρίτη εταιρεία - για την φιλοξενία των εφαρμογών ή/και των δεδομένων τους. Αυτό έχει ως συνέπεια τα δεδομένα τους να είναι άμεσα διαθέσιμα σε διαβαθμισμένο προσωπικό, π.χ. διαχειριστές, της τρίτης εταιρείας. Στις περιπτώσεις αυτές, τα δεδομένα είναι στην διάθεση αυτών των υπαλλήλων και δεν υπάρχει κάποιος μηχανισμός που να μπορεί να τους εμποδίσει [08, 37].

Από τα παραπάνω συμπεραίνουμε ότι η προστασία των δεδομένων, εστιάζοντας στο να αποτρέψουμε τους επίδοξους εισβολείς να τα προσεγγίσουν, είναι ένα πολύ δύσκολο εγχείρημα. Αν και τα αποτελέσματα αυτής της στρατηγικής πολλές φορές δεν είναι ικανοποιητικά, εξακολουθεί εν τούτοις να είναι η επικρατούσα επιλογή. Μια εναλλακτική στρατηγική, και οι νέες προκλήσεις που δημιουργούνται, παρουσιάζεται στη συνέχεια.

3.2 Ασφάλεια (κρυπτογράφηση)

Μια άλλη προσέγγιση είναι η κρυπτογράφηση από-άκρο-σε-άκρο. Με την τεχνική αυτή, τα δεδομένα κρυπτογραφούνται στο πρόγραμμα πλοήγησης (browser) του χρήστη και αποστέλλονται για αποθήκευση στον εξυπηρετητή όπου και παραμένουν κρυπτογραφημένα. Η αποκρυπτογράφησή τους γίνεται μόνο κατά την ανάκτησή τους από τον browser του χρήστη. Έτσι στην περίπτωση διαρροής απευθείας από τον εξυπηρετητή, τα δεδομένα είναι άχρηστα (μη αξιοποιήσιμα) για τους εισβολείς.

Η προσέγγιση αυτή δεν υιοθετείται ευρέως για τους εξής λόγους.

3.2.1 Κοινή χρήση δεδομένων

Μια κρίσιμη λειτουργία των εφαρμογών ιστού είναι η κοινή χρήση και ανταλλαγή δεδομένων μεταξύ των χρηστών. Για να επιτευχθεί αυτό, διατηρώντας ταυτόχρονα κρυπτογραφημένα τα δεδομένα στον εξυπηρετητή, τα κοινά δεδομένα θα πρέπει να κρυπτογραφούνται με ξεχωριστά κλειδιά, τα οποία να διανέμονται σε όλους τους χρήστες που μοιράζονται τα δεδομένα. Ωστόσο, το πλήθος των κλειδιών και η πολυπλοκότητα της πολιτικής που θα πρέπει να ακολουθηθεί καθιστά τη διαχείριση και διανομή κλειδιών δύσκολη και επισφαλή. Αυτό, στην καλύτερη περίπτωση, απαιτεί μεγάλες αλλαγές στον σχεδιασμό και την υλοποίηση της εφαρμογής. Μερικές εφαρμογές [11, 31, 43] ήδη χρησιμοποιούν την παραπάνω προσέγγιση. Το ενδιαφέρον είναι ότι στην αρχική σελίδα των παραπάνω αναφέρεται ρητά ότι είναι ευάλωτες απέναντι σε επιθέσεις μέσω JavaScript κώδικα. Μια τέτοια επίθεση θα είχε ως αποτέλεσμα, τη διαρροή του κλειδιού κρυπτογράφησης και, κατά συνέπεια, και των δεδομένων. Αυτό δείχνει και τη δυσκολία των εφαρμογών στη διαχείριση και διανομή των κλειδιών. Τελικά, η κρυπτογράφηση των δεδομένων καθιστά την κοινή χρήση τους ένα δύσκολο εγχείρημα και αυτή είναι μια από τις αιτίες που δεν χρησιμοποιείται ευρέως. Ο κυριότερος λόγος όμως είναι ο παρακάτω.

3.2.2 Δυνατότητα υπολογισμών και αναζητήσεων επί κρυπτογραφημένων δεδομένων

Μια κρίσιμη λειτουργικότητα για τις περισσότερες εφαρμογές είναι η δυνατότητα αναζήτησης. Υπάρχουν πολλές εφαρμογές, π.χ. ηλεκτρονικού ταχυδρομείου, ανταλλαγής μηνυμάτων, για τις οποίες θα ήταν εξαιρετικά σημαντικό να μπορούσαν να κρυπτογραφήσουν τα δεδομένα τους και, παράλληλα, να υποστήριζαν τη δυνατότητα αναζήτησης πάνω σε αυτά. Η αναζήτηση σε κρυπτογραφημένα δεδομένα έχει απασχολήσει ιδιαίτερα την κρυπτογραφική κοινότητα και έχουν δημιουργηθεί αρκετά συστήματα κρυπτογράφησης ειδικά για τον σκοπό αυτό. Μερικά από τα συστήματα αυτά θα περιγραφούν ξεχωριστά παρακάτω.

Η πλειονότητα των εφαρμογών δεν αποθηκεύει απλώς τα δεδομένα της στους εξυπηρετητές. Οι εφαρμογές βασίζουν τις κύριες λειτουργίες τους σε υπολογισμούς που εφαρμόζουν πάνω σε αυτά. Έτσι, κρυπτογραφώντας τα δεδομένα με κάποιο από τα παραδοσιακά συστήματα κρυπτογράφησης, ουσιαστικά τα καθιστούμε πρακτικώς μη συμβατά με υπολογισμούς πάνω σε αυτά.

Αν και υπάρχουν κρυπτογραφικά συστήματα που δίνουν την δυνατότητα υπολογισμών στα κρυπτογραφημένα δεδομένα, στην πλειονότητά τους μερικώς, κανένα από αυτά δεν είναι σε θέση να καλύψει όλες τις ανάγκες των υπαρχουσών εφαρμογών με πρακτικό τρόπο.

Στην κρυπτογραφική κοινότητα, ιδιαίτερα τα τελευταία χρόνια, έχουν παρουσιαστεί αρκετά κρυπτογραφικά συστήματα που παρέχουν την δυνατότητα υπολογισμού σε κρυπτογραφημένα δεδομένα. Τα συστήματα αυτά επιτυγχάνουν διάφορες επιδόσεις στις εξής τρεις περιοχές.

1. Λειτουργικότητα
2. Ασφάλεια
3. Απόδοση

Οι επιδόσεις των περισσοτέρων δεν είναι επαρκείς σε τουλάχιστο μία από αυτές τις περιοχές και δυστυχώς, κανένα δεν αρκεί από μόνο του να καλύψει όλο το φάσμα των εφαρμογών που διαθέτουμε σήμερα.

Στη συνέχεια παρουσιάζονται τα κυριότερα από αυτά τα συστήματα.

Ομομορφική κρυπτογράφηση.

Ένα σύστημα ομομορφικής κρυπτογράφησης κρυπτογραφεί τα δεδομένα με τέτοιο τρόπο ώστε να μπορούν να εκτελεστούν υπολογισμοί στα κρυπτογραφημένα δεδομένα χωρίς την γνώση του κρυπτογραφικού κλειδιού. Έτσι, έχοντας τις κρυπτογραφημένες ποσότητες $c1 = E(k, m1)$ και $c2 = E(k, m2)$ μπορούμε να υπολογίσουμε $το$ $E(k, f(m1, m2))$ χωρίς να χρειάζεται να αποκρυπτογραφήσουμε τα $c1, c2$. Εδώ η f υποδηλώνει κάποιο αυθαίρετο τελεστή.

Τα συστήματα κρυπτογράφησης που υποστηρίζουν μόνο πρόσθεση ή πολλαπλασιασμό ονομάζονται μερικώς ομομορφικά. Αυτά που υποστηρίζουν και τις δύο πράξεις, ουσιαστικά υποστηρίζουν οποιαδήποτε συνάρτηση, ονομάζονται πλήρως ομομορφικά.

Ο όρος ομομορφική κρυπτογράφηση χρησιμοποιήθηκε για πρώτη φορά το 1978 από τους Rivest-Shamir-Dertouzos, οι οποίοι έθεσαν το ζήτημα ύπαρξης ενός πλήρους ομομορφικού συστήματος εμπνευσμένοι από την ομομορφική ιδιότητα του συστήματος RSA, το οποίο είχε εισαχθεί την ίδια χρονιά.

Μερικά από τα μερικώς ομομορφικά συστήματα είναι τα παρακάτω

3. **RSA**. Σύστημα δημόσιου κλειδιού, ομομορφικό ως προς τον πολλαπλασιασμό. Ισχύει: $E(k, m1) * E(k, m2) = E(k, m1 * m2)$

4. **El Gamal**. Σύστημα δημόσιου κλειδιού, ομομορφικό ως προς τον πολλαπλασιασμό. Ισχύει: $E(k, m1) * E(k, m2) = E(k, m1 * m2)$

5. **Paillier**. Σύστημα δημόσιου κλειδιού, ομομορφικό ως προς την πρόσθεση. Ισχύει: $E(k, m1) * E(k, m2) = E(k, m1 + m2)$

Λόγω της περιορισμένης λειτουργικότητας που προσφέρουν, μιας και υποστηρίζουν μόνο μία πράξη, έχουν εξειδικευμένες εφαρμογές, όπως σε εφαρμογές ηλεκτρονικής ψηφοφορίας [02].

Το πρώτο πλήρως ομομορφικό σύστημα δημιουργήθηκε το 2009 από τον Gentry [13]. Η σημαντικότητα της εργασίας του ήταν τεράστια και αποτέλεσε τη βάση για την δημιουργία νέων πλήρως ομομορφικών συστημάτων.

Τα πλήρως ομομορφικά συστήματα προσφέρουν πλήρη λειτουργικότητα και θεωρητικά μπορούν να καλύψουν όλες τις ανάγκες υπολογισμού, συμπεριλαμβανομένης και της αναζήτησης, στα κρυπτογραφημένα δεδομένα. Δυστυχώς όμως, αν και έχει γίνει μεγάλη πρόοδος σχετικά με την απόδοσή τους, προς το παρόν απέχουν πολύ από το να ικανοποιούν πλήρως την απόδοση που απαιτεί μια τυπική εφαρμογή [01].

Property preserving encryption (PPE)

Στην κατηγορία αυτή ανήκουν τα συστήματα κρυπτογράφησης στα οποία οι κρυπτογραφημένες τιμές διατηρούν κάποια από τις ιδιότητες, π.χ. διάταξη, που είχαν οι αρχικές μη κρυπτογραφημένες. Ειδικές περιπτώσεις PPE είναι η order preserving encryption (OPE) και η deterministic encryption (DE). Τα κρυπτογραφικά συστήματα στην PPE διατηρούν την ιδιότητα της διάταξης ενώ στην DE την ιδιότητα της ισότητας. Προφανώς κάθε σύστημα OPE είναι και DE.

Οι Agrawal-Kiernan-Srikant-Xu (2004) ήταν οι πρώτοι που χρησιμοποίησαν τον όρο OPE, προτείνοντας ένα κρυπτογραφικό σύστημα OPE [03] για αριθμητικά δεδομένα, το οποίο δίνει την δυνατότητα σύγκρισης στα κρυπτογραφημένα δεδομένα.

Πιο πρόσφατα, οι Popa-Redfield-Zeldovich-Balakrishnan (2011) δημιούργησαν την βάση δεδομένων CryptDB [27], η οποία χρησιμοποιεί πέντε διαφορετικά κρυπτογραφικά συστήματα για την κρυπτογράφηση των δεδομένων. Από τα συστήματα αυτά, τα δύο ανήκουν στην κατηγορία των PPE και δίνουν την δυνατότητα για κάθε είδους αναζήτηση στα δεδομένα. Η απήχηση της CryptDB ήταν μεγάλη και επηρέασε πολλές νέες υλοποιήσεις [05].

Γενικότερα στα συστήματα DE δίνεται η δυνατότητα αναζήτησης λέξης στα κρυπτογραφημένα δεδομένα. Αυτό γίνεται κρυπτογραφώντας τον όρο αναζήτησης με το κλειδί κρυπτογράφησης των δεδομένων και, στη συνέχεια, εκτελώντας την αναζήτηση όπως και στα μη κρυπτογραφημένα δεδομένα. Ιδιαίτερα στα OPE υποστηρίζεται και η αναζήτηση με ερωτήματα εύρους τιμών.

Παρά την χρησιμότητα και την καλή απόδοσή τους, τα συστήματα PPE παρουσιάζουν σημαντικά προβλήματα ασφάλειας [16, 23]. Μια λύση είναι να χρησιμοποιούνται σε συνεργασία με κάποιο άλλο κρυπτογραφικό σύστημα μεγαλύτερης ασφάλειας, όπως έγινε στην περίπτωση της CryptDB.

Πρέπει επίσης να σημειωθεί ότι τα συστήματα PPE δεν είναι σημασιολογικά ασφαλή (semantically secure). Αυτό συμβαίνει γιατί γνωρίζοντας κάποιος τις κρυπτογραφημένες τιμές, μπορεί να βρει τη σχέση που έχουν οι αρχικές μη κρυπτογραφημένες.

Ειδικότερα τα συστήματα OPE, όπως και κάθε σύστημα DE, είναι ευάλωτα σε επιθέσεις ανάλυσης συχνοτήτων (frequency analysis). Επιπλέον, στην περίπτωση που το σύστημα είναι και δημοσίου κλειδιού, τότε γίνονται ευάλωτα και σε λεξικογραφικές επιθέσεις. Επιθέσεις στατιστικής ανάλυσης σε συστήματα PPE περιγράφουν οι Naveed-Kamara-Wright στη σχετική εργασία τους [23].

Ειδικά κρυπτογραφικά συστήματα με δυνατότητα αναζήτησης. (SE)

Στη συνέχεια παρουσιάζονται τα σημαντικότερα κρυπτογραφικά συστήματα που κατασκευάστηκαν με κύριο στόχο την δυνατότητα αναζήτησης στα κρυπτογραφημένα δεδομένα.

Το πρώτο σύστημα κρυπτογράφησης με σκοπό την δυνατότητα αναζήτησης κατασκευάσθηκε το 2000 από τους Song-Wagner-Perrig [35]. Το σύστημα είναι συμμετρικής κρυπτογράφησης και χρησιμοποιεί αλγόριθμο ροής.

Αν και δίνει την δυνατότητα αναζήτησης λέξης, η οποία μπορεί να επεκταθεί ώστε να υποστηρίζει και πολυπλοκότερες αναζητήσεις, όπως συζεύξεις και διαζεύξεις, η

απόδοσή του είναι καλή σε περιορισμένου μεγέθους δεδομένα (ο χρόνος αναζήτησης είναι γραμμικός του πλήθους των δεδομένων).

Επιπλέον, ενώ το σύστημα παρέχει σημασιολογική ασφάλεια, δεν παρέχει την ειδική ασφάλεια που θέλουμε να παρέχεται από κάθε σύστημα κρυπτογράφησης με δυνατότητα αναζήτησης σύμφωνα με τα πρότυπα ασφάλειας για τα συστήματα SE που τέθηκαν αργότερα.

Το 2003 ο Goh [14] προτείνει νέα πρότυπα ασφάλειας για τα συστήματα SE και εισάγει την προσέγγιση της χρήσης ασφαλών ευρετηρίων. Τα νέα αυτά πρότυπα αφορούν τη διάχυση πληροφορίας που παρουσιάζεται λόγω της αναζήτησης που εκτελείται στα δεδομένα. Η διάχυση αφορά το μοτίβο εμφάνισης των λέξεων αναζήτησης στα δεδομένα όπως και το μοτίβο των λέξεων αναζήτησης. Το σύστημα που κατασκευάζει δίνει την δυνατότητα αναζήτησης λέξης και ικανοποιεί τα μέχρι τότε πρότυπα ασφάλειας, όμως η απόδοση του συστήματος αν και καλύτερη από το σύστημα του Song, δεν είναι ικανοποιητική.

Το 2011 οι Curtmola-Garay-Kamara-Ostrovsky [09] προτείνουν νέα πρότυπα ασφάλειας για τα συστήματα DE, τα οποία είναι τα ισχυρότερα μέχρι σήμερα, και εισάγουν την προσέγγιση των ανεστραμμένων ευρετηρίων. Το σύστημα που κατασκευάζουν δίνει την δυνατότητα αναζήτησης λέξης, ικανοποιεί τα κριτήρια ασφάλειας που έχουν θέσει και έχει ικανοποιητική απόδοση.

Άλλα σημαντικά συστήματα SE είναι το σύστημα των Kamara-Papamanthou-Roeder [18], το οποίο βασίζεται στο σύστημα των Curtmola-Garay-Kamara-Ostrovsky [09], και το σύστημα των Cash-Jarecki-Jutla-Krawczyk-Roşu-Steiner [07] το οποίο υποστηρίζει πολύπλοκες αναζητήσεις όπως συζεύξεις, διαζεύξεις, αρνήσεις. Η απόδοση και των δύο συστημάτων είναι πολύ καλή και ικανοποιούν τα πρότυπα ασφάλειας συστημάτων SE που έχουν προτείνει οι Curtmola-Garay-Kamara-Ostrovsky.

Τα τελευταία έτη η εξέλιξη στον τομέα του SE είναι ραγδαία. Παρουσιάζονται νέα συστήματα των οποίων οι επιδόσεις συνεχώς βελτιώνονται και τείνουν να είναι αυτές που απαιτούνται για την λειτουργία μίας σύγχρονης εφαρμογής. Βέβαια, όλα έχουν ένα κοινό χαρακτηριστικό: προϋποθέτουν ότι τα δεδομένα είναι κρυπτογραφημένα με το ίδιο κλειδί. Το χαρακτηριστικό αυτό, όπως θα δούμε στη συνέχεια στην υπο-ενότητα

4.4.2, δημιουργεί πρακτικά προβλήματα σε εφαρμογές που υποστηρίζουν την κοινή χρήση των δεδομένων.

Κεφάλαιο 4
Πλατφόρμα Mylar: Αρχιτεκτονική και Λειτουργία

Η πλατφόρμα Mylar [28], η οποία αναπτύχθηκε το 2014 από ερευνητές του Πανεπιστημίου MIT, είναι σχεδιασμένη με σκοπό να προσδώσει, στις εφαρμογές ιστού που θα τη χρησιμοποιήσουν, ασφάλεια των δεδομένων τους, διατηρώντας παράλληλα τη χρηστικότητά τους σε υψηλό επίπεδο. Σχετικά με το πρώτο, τα δεδομένα κρυπτογραφούνται την στιγμή που αυτά «εγκαταλείπουν» το πρόγραμμα πλοήγησης (browser) του χρήστη και παραμένουν έτσι μέχρι και την αποθήκευσή τους, ενώ επιπλέον ελέγχει και επαληθεύει ότι ο κώδικας της εφαρμογής που φορτώνεται από τον εξυπηρετητή στον browser του χρήστη δεν είναι αλλοιωμένος. Σχετικά με το δεύτερο, δίνει την δυνατότητα αναζήτησης λέξης (keyword search) πάνω στα κρυπτογραφημένα δεδομένα, καθώς επίσης και της κοινής χρήσης αυτών από πολλούς χρήστες.

Για την επίτευξη των παραπάνω λειτουργιών το Mylar χρησιμοποιεί γνωστούς αλγόριθμους συμμετρικής και ασύμμετρης κρυπτογράφησης σε συνεργασία με ένα νέο κρυπτογραφικό σύστημα, το MK (Multi-Key Searchable Encryption) [30], το οποίο κατασκευάστηκε από τους δημιουργούς του Mylar, Popa-Zeldovich, και είναι αυτό που

δίνει την δυνατότητα αναζήτησης σε δεδομένα τα οποία είναι κρυπτογραφημένα με διαφορετικά κλειδιά . Το ΜΚ θα περιγραφεί στη συνέχεια στην υπο-ενότητα 4.4.2.

4.1 Αρχιτεκτονική

Για την υλοποίηση της πλατφόρμας Mylar χρησιμοποιήθηκε το πλαίσιο ανάπτυξης εφαρμογών ιστού (framework) Meteor [22]. Η επιλογή αυτή βασίστηκε στα χαρακτηριστικά που συνοδεύουν το συγκεκριμένο framework και τα οποία ταιριάζουν στο μοντέλο σχεδίασης του Mylar.

Το framework Meteor είναι ανοιχτού κώδικα (open source), έχει υλοποιηθεί στη γλώσσα προγραμματισμού JavaScript και αποτελείται από πακέτα λογισμικού (software packages). Εδώ πρέπει να αναφερθεί ότι το Meteor είναι εξαιρετικά δημοφιλής πλατφόρμα και χρησιμοποιείται από πολλές εφαρμογές.

Η δομή του Meteor σε πακέτα λογισμικού και ο ενιαίος τρόπος μεταφοράς των δεδομένων αποτέλεσαν μια πολύ καλή βάση για την υλοποίηση του Mylar. Επιπλέον, και κυριότερο, επιτρέπει την αποστολή των δεδομένων και της δομής της html σελίδας σε ξεχωριστά στρώματα (layers). Αυτό το χαρακτηριστικό είναι και το πλέον κρίσιμο για να εφαρμοστεί η λύση ασφάλειας των δεδομένων που προτείνει το Mylar. Με αυτόν τον τρόπο ο στατικός κώδικας μιας εφαρμογής ιστού μπορεί να ελέγχεται για αλλοιώσεις και παράλληλα να κρυπτογραφούνται τα ευαίσθητα δεδομένα των χρηστών της.

4.1.1 Δομικά στοιχεία λογισμικού του Mylar

Τα δομικά στοιχεία λογισμικού του Mylar είναι τα εξής:

1. **Επέκταση browser (browser extension).** Πρόγραμμα λογισμικού που προσθέτει ο χρήστης της εφαρμογής ιστού στο πρόγραμμα πλοήγησής του (browser). Το πρόγραμμα αυτό ελέγχει αν ο στατικός κώδικας της εφαρμογής που “φορτώνεται” στον browser του χρήστη είναι αλλοιωμένος. Ο έλεγχος γίνεται εξετάζοντας την ψηφιακή υπογραφή του στατικού κώδικα.

2. **Βιβλιοθήκη πελάτη (Client-side library).** Σύνολο πακέτων λογισμικού που εκτελούνται στον browser του χρήστη της εφαρμογής. Κύριος ρόλος τους είναι να παρεμβαίνουν στην διαδρομή των δεδομένων, πριν την αποστολή τους στον εξυπηρετητή ή όταν λαμβάνονται από αυτόν, με σκοπό την κρυπτογράφηση ή αποκρυπτογράφηση των δεδομένων αντίστοιχα. Επιπλέον, αναλαμβάνουν την διαδικασία σύνδεσης των χρηστών στην εφαρμογή όπως και την διαδικασία δημιουργίας νέων.

3. **Βιβλιοθήκη εξυπηρετητή (Server-side library).** Σύνολο πακέτων λογισμικού που εκτελούνται στον εξυπηρετητή της εφαρμογής. Αφορούν διαδικασίες σχετικές με την αναζήτηση λέξης επάνω στα κρυπτογραφημένα δεδομένα.

4. **Πάροχος ταυτότητας (Identity provider) (IDP).** Προαιρετική υπηρεσία, η οποία είναι απαραίτητη μόνο σε εφαρμογές που επιτρέπεται η κοινοχρησία δεδομένων και παράλληλα δεν μπορούν να πιστοποιήσουν την ταυτότητα των χρηστών τους. Η υπηρεσία αυτή πιστοποιεί την ταυτότητα ενός χρήστη και την κυριότητα του δημόσιου κλειδιού από αυτόν. Για την παροχή της υπάρχει το λεγόμενο υποσύστημα IDP του Mylar - διαφορετικά μπορεί να χρησιμοποιηθεί ένας έμπιστος εξωτερικός πάροχος.

Η χρήση των παραπάνω δομικών στοιχείων γίνεται μέσω της διεπαφής (application interface) (API) του Mylar. Η διεπαφή αποτελείται από ένα σύνολο συναρτήσεων που χρησιμοποιεί τις παραπάνω βιβλιοθήκες και είναι απαραίτητη για την ανάπτυξη ή ενσωμάτωση μια εφαρμογής στο Mylar. Μια γενική περιγραφή των κυριοτέρων συναρτήσεων της διεπαφής βρίσκεται στο Παράρτημα Α.

4.2 Ασφάλεια

Η χρήση της πλατφόρμας Mylar στην ανάπτυξη μιας εφαρμογής ιστού θα της παρέχει υψηλά επίπεδα ασφάλειας κάτω από κάποιες προϋποθέσεις.

Το Mylar ταξινομεί τις επιθέσεις για την υποκλοπή των δεδομένων της εφαρμογής στις παρακάτω κατηγορίες και υποθέτει ότι ο επιτιθέμενος έχει ήδη πρόσβαση στον εξυπηρετητή της εφαρμογής.

1. **Παθητική (passive).** Ο επιτιθέμενος υποκλέπτει τα δεδομένα που βρίσκονται στον εξυπηρετητή χωρίς όμως να επηρεάζει την εφαρμογή, δηλαδή χωρίς να επιδρά στις λειτουργίες ή στα δεδομένα της.

2. **Ενεργή (active).** Στην κατηγορία αυτή ο επιτιθέμενος μπορεί να επηρεάζει τις λειτουργίες της εφαρμογής. Η επίθεση μπορεί να περιλαμβάνει αλλαγή των δεδομένων της εφαρμογής, αποστολή κακόβουλου κώδικα στους χρήστες, υποκλοπή όλων των δεδομένων και ο,τιδήποτε άλλο έχει την δυνατότητα να κάνει ο εισβολέας. Όσο πιο αυξημένα είναι τα δικαιώματα που έχει ο εισβολέας, τόσο μεγαλύτερη η επίδραση του στην εφαρμογή.

Οι εγγυήσεις που προσφέρει το Mylar έναντι αυτών των μορφών επιθέσεων, οι οποίες καλύπτουν ένα μεγάλο φάσμα προβλημάτων ασφάλειας, ισχύουν υπό τις ακόλουθες προϋποθέσεις:

1. Οι χρήστες της εφαρμογής έχουν εγκαταστήσει και ενεργοποιήσει την επέκταση browser του Mylar. Αυτό είναι αναγκαίο για τον έλεγχο του κώδικα που φορτώνεται στον browser του χρήστη. Η προϋπόθεση αυτή είναι απαραίτητη για την προστασία απέναντι στην ενεργή μορφή επίθεσης.

2. Ο προγραμματιστής που αναπτύσσει την εφαρμογή ιστού θα πρέπει να χρησιμοποιήσει την διεπαφή εφαρμογών του Mylar με προσοχή ώστε να αποφευχθούν σφάλματα στον κώδικα τα οποία μπορεί να εκμεταλλευτεί ο επιτιθέμενος. Η συγκεκριμένη οδηγία δεν αφορά μόνο τη χρήση της διεπαφής αλλά τον συνολικό τρόπο ανάπτυξης της εφαρμογής.

3. Η υπηρεσία παροχής ταυτότητας (IDP) είναι αξιόπιστη και λειτουργεί ορθά, είτε αυτή παρέχεται από το υποσύστημα IDP του Mylar είτε από κάποιον τρίτο πάροχο. Και αυτή η προϋπόθεση αφορά στην προστασία απέναντι στην ενεργή μορφή επίθεσης και μόνο τις εφαρμογές που έχουν την ανάγκη αυτής τη υπηρεσίας.

Με τις παραπάνω προϋποθέσεις, οι εγγυήσεις που προσφέρει το Mylar ανά μορφή επίθεσης είναι οι εξής:

1. Εγγυήσεις σε παθητική επίθεση. Στα δεδομένα της εφαρμογής που έχουν σημανθεί αποκλειστικώς ως κρυπτογραφημένα (encrypted notation), το Mylar εγγυάται κρυπτογράφηση από-άκρο-σε-άκρο (end-to-end encryption), δηλαδή εξαιρετικά μεγάλη ασφάλεια. Όμως για τα κρυπτογραφημένα δεδομένα στα οποία έχει σημανθεί η δυνατότητα αναζήτησης (searchable notation), οι εγγυήσεις είναι ασθενέστερες. Όπως έχει αναφερθεί νωρίτερα η κρυπτογράφηση με δυνατότητα αναζήτησης (searchable encryption) ενέχει κάποιες αδυναμίες αφού διαχέονται πληροφορίες σχετικές με τα δεδομένα που αφορούν το μοτίβο των λέξεων αναζήτησης ή και το μοτίβο εμφάνισης των λέξεων της αναζήτησης στα δεδομένα [17]. Έτσι, λαμβάνοντας επιπλέον υπόψη ότι η μέθοδος κρυπτογράφησης δεδομένων στο Mylar έχει την ιδιότητα της μη διακρισιμότητας σε επιθέσεις επιλεγμένου αρχικού κειμένου (IND-CPA) και ότι η κρυπτογράφηση της λέξης αναζήτησης γίνεται με ντετερμινιστικό αλγόριθμο, το Mylar κατατάσσεται στη μέση της βαθμίδας ιεράρχησης που αναφέρουν οι Cash–Grubbs–Perry-Ristenpart [06] σχετικά με την διάχυση πληροφορίας.

2. Εγγυήσεις σε ενεργή επίθεση. Σε αυτή την περίπτωση το Mylar προσφέρει τις εξής εγγυήσεις Κατ' αρχάς, εγγυάται ότι ο κώδικας της εφαρμογής που θα εκτελεστεί στον browser του χρήστη δεν θα έχει αλλοιωθεί. Επιπλέον, εγγυάται ότι κατά την αναζήτηση του κλειδιού κρυπτογράφησης μιας κύριας οντότητας από τον browser του χρήστη, ο εξυπηρετητής θα του προσφέρει το σωστό κλειδί.

4.3 Κύριες οντότητες

Πριν προχωρήσουμε στην περιγραφή της λειτουργίας του Mylar είναι απαραίτητο να περιγράψουμε αναλυτικά την έννοια 'Κύρια Οντότητα' (principal), την οποία θα συναντήσουμε στη συνέχεια και η οποία έχει πολύ σπουδαίο ρόλο στη λειτουργία του Mylar.

Κάθε principal είναι μια οντότητα για τον έλεγχο της πρόσβασης στα δεδομένα της εφαρμογής και αντιστοιχεί σε οντότητες της εφαρμογής που είτε πρέπει να έχουν πρόσβαση σε εμπιστευτικά δεδομένα, π.χ. χρήστες ή ομάδες χρηστών, είτε περιέχουν εμπιστευτικά δεδομένα, π.χ. έγγραφα. Κάθε principal έχει:

1. όνομα

2. ένα ζεύγος κλειδιών e_{priv} - e_{pub} (ιδιωτικό – δημόσιο) ασύμμετρης κρυπτογράφησης(El Gamal)

3. ένα ζεύγος κλειδιών d_{priv} - d_{pub} (ιδιωτικό – δημόσιο) ασύμμετρης κρυπτογράφησης για ψηφιακή υπογραφή (DSA)

4. ένα κλειδί *s* συμμετρικής κρυπτογράφησης (AES)

5. ένα κλειδί *k* κρυπτογράφησης MK για δυνατότητα αναζήτησης

Στο εξής θα αναφέρονται ως μυστικά κλειδιά της principal, όλα τα παραπάνω κλειδιά εκτός των δημόσιων κλειδιών e_{pub}, d_{pub}.

Η επιλογή των οντοτήτων για τις οποίες θα δημιουργηθούν principals καθορίζεται από την λογική της εφαρμογής και την πολιτική ασφάλειας στα δεδομένα της και γίνεται στην φάση ανάπτυξης της εφαρμογής. Ο ρόλος τους είναι πολύ σημαντικός μιας και η πολιτική ασφάλειας εφαρμόζεται με την χρήση τους. Επιπλέον, είναι σημαντικό τα ονόματα των principals να αντικατοπτρίζουν εννοιολογικά την αντίστοιχη οντότητα. Η απαίτηση αυτή θα δικαιολογηθεί στην ανάλυση της πολιτικής πιστοποίησης στην υπο-ενότητα 4.4.1.

Η δημιουργία των principals γίνεται μέσω άλλων principals, π.χ. ο χρήστης Α δημιουργεί το αρχείο Β. Εδώ πρέπει να σημειώσουμε ότι, τα μυστικά κλειδιά της principal Β κρυπτογραφούνται με το δημόσιο κλειδί e_{pub} της principal Α που τη δημιουργεί. Δηλαδή δημιουργείται ένα «εμφωλιασμένο» κλειδί (wrapped key) και αποθηκεύεται στον εξυπηρετητή. Μοναδικές εξαιρέσεις είναι οι principals χρηστών, όπου το wrapped key δημιουργείται με το password του χρήστη και η ειδική περίπτωση των στατικών principals.

Η δημιουργία όλων των principals, εκτός των χρηστών και των στατικών υλοποιείται προγραμματιστικά μέσω της κλήσης της παρακάτω συνάρτησης της διεπαφής του Mylar:

$princ\_create(name, creator\_princ)$.

η οποία καλείται για την δημιουργία της principal *name* από την principal *creator_princ*.

Αντίστοιχα οι principals χρηστών δημιουργούνται με τη κλήση της συνάρτησης:

$create\_user(uname, password, auth\_princ)$

η οποία καλείται για την δημιουργία του χρήστη *uname*, ο οποίος πιστοποιείται από την principal *auth_princ*.

Οι στατικές principals αναλύονται στην επόμενη υπο-ενότητα 4.3.1.

4.3.1 Στατικές Principals

Ιδιαίτερη περίπτωση principals είναι οι στατικές, οι οποίες έχουν κύριο ρόλο να αντιπροσωπεύουν ομάδες οντοτήτων με κοινά δικαιώματα πρόσβασης στα δεδομένα. Επιπλέον, στις εφαρμογές που η δημιουργία των χρηστών γίνεται με ελεγχόμενο τρόπο, οι στατικές principals μπορούν να χρησιμοποιηθούν για την πιστοποίηση της ταυτότητας των χρηστών και της κυριότητας των δημόσιων κλειδιών τους, δηλαδή να παίξουν τον ρόλο του IDP.

Οι στατικές principals δεν είναι απαραίτητες σε όλες τις εφαρμογές. Η δημιουργία τους αποτελεί μέρος της παραμετροποίησης της εφαρμογής. Για τη δημιουργία τους πρέπει να εκτελεστεί, μέσω κάποιου εργαλείου γραμμής εντολών, η παρακάτω συνάρτηση (function) της διεπαφής του Mylar.

$princ\_create\_static(name, password)$

Η παραπάνω συνάρτηση θα δημιουργήσει μια νέα στατική principal με όνομα *name* και τα αντίστοιχα κλειδιά κρυπτογράφησης. Τα μυστικά κλειδιά είναι κρυπτογραφημένα με κλειδί το *password*. Το κλειδιά αποθηκεύονται στον πηγαίο κώδικα (source code) της εφαρμογής.

Η πρόσβαση στα μυστικά κλειδιά της στατικής principal γίνεται μόνο μέσω της στην παρακάτω συνάρτηση της διεπαφής του Mylar, δίνοντας το $password$ που χρησιμοποιήθηκε για την δημιουργία της principal $name$.

$princ\_static(name, password)$

Οι στατικές principals μαζί με το IDP, λόγω του ρόλου τους να πιστοποιούν του χρήστες της εφαρμογής, αναφέρονται και ως κύριες οντότητες αυθεντικοποίησης (authority principals).

4.4 Λειτουργία

Στην παράγραφο αυτή θα περιγραφεί ο τρόπος με τον οποίο επιτυγχάνεται η κοινή χρήση των κρυπτογραφημένων δεδομένων καθώς και η αναζήτηση επάνω σε αυτά.

4.4.1 Κοινή χρήση δεδομένων (data sharing)

Στο Mylar τα εμπιστευτικά δεδομένα κρυπτογραφούνται με το συμμετρικό σύστημα κρυπτογράφησης AES χρησιμοποιώντας το συμμετρικό κλειδί s της principal στην οποία ανήκουν. Στην περίπτωση που τα δεδομένα έχουν σημανθεί και ως «αναζητήσιμα» (searchable), κρυπτογραφούνται μία επιπλέον φορά με το σύστημα MK και την χρήση του κλειδιού k. Ο λόγος που γίνεται αυτό θα αποσαφηνισθεί παρακάτω στην υπο-ενότητα 4.4.2.

Άρα για να έχει κάποια principal πρόσβαση σε αυτά πρέπει να έχει πρόσβαση στα μυστικά κλειδιά της. Έτσι ανακύπτουν δύο βασικά θέματα, τα οποία γίνονται περισσότερο κατανοητά με το παρακάτω παράδειγμα.

Υποθέτουμε ότι οι χρήστες A και B πρέπει να έχουν πρόσβαση στα εμπιστευτικά δεδομένα του αρχείου C. Η απαίτηση αυτή σημαίνει:

1. Η εφαρμογή πρέπει να δώσει στους χρήστες A και B πρόσβαση στα δεδομένα του αρχείου C.

2. Η εφαρμογή πρέπει να εξασφαλίσει ότι η πρόσβαση που προσφέρει στους χρήστες A και B αφορά πράγματι το αρχείο C και όχι κάποιο άλλο. Δηλαδή να πιστοποιείται η ταυτότητα του αρχείου.

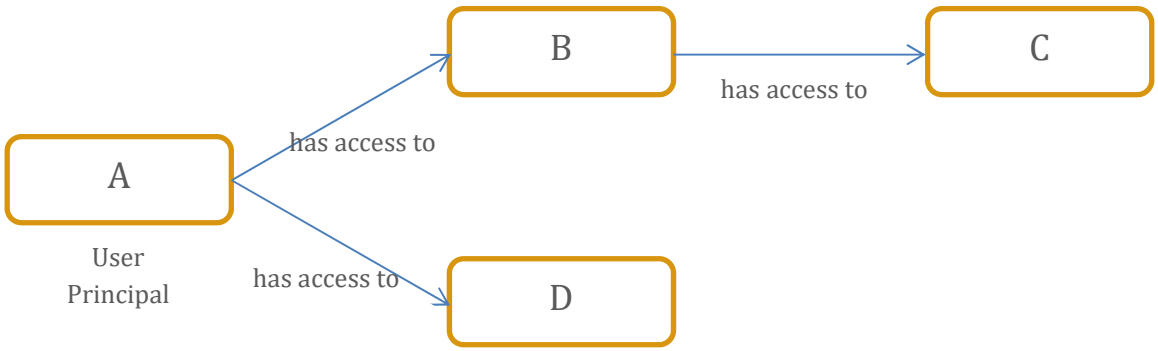
Έτσι, αρχικά συμπεραίνουμε ότι η κοινή χρήση των εμπιστευτικών δεδομένων μιας εφαρμογής απαιτεί από την εφαρμογή να καθορίσει και να εφαρμόσει μια πολιτική πρόσβασης, έτσι ώστε οι κατάλληλες principals, π.χ. χρήστες, να έχουν πρόσβαση στα εμπιστευτικά δεδομένα.

Επιπλέον, βλέπουμε ότι είναι αναγκαίο η εφαρμογή να βρίσκει την κατάλληλη principal είτε όταν πρέπει να κρυπτογραφήσει δεδομένα για αυτή, π.χ. ένα αρχείο, είτε όταν πρέπει να την κάνει προσβάσιμη σε κάποια άλλη principal, π.χ. σε έναν χρήστη. Άρα είναι απαραίτητη, σε ένα περιβάλλον ασφάλειας που οι ενεργές επιθέσεις δεν αποκλείονται, η πιστοποίηση των principals. Δηλαδή η εφαρμογή πρέπει να καθορίσει και να εφαρμόσει μια πολιτική πιστοποίησης.

Οι δύο αυτές πολιτικές αποτελούν τη πολιτική ασφάλειας της εφαρμογής σχετικά με την κοινοχρησία των δεδομένων και αναλύονται παρακάτω.

Πολιτική Πρόσβασης

Το Mylar δίνει την δυνατότητα σε μια εφαρμογή να καθορίσει την πολιτική πρόσβασης με την βοήθεια των principals δημιουργώντας μεταξύ τους σχέσεις πρόσβασης.



Σχήμα 4.1: Παράδειγμα Γράφου Προσβάσεων.

Η πολιτική πρόσβασης της εφαρμογής εκφράζεται με σχέσεις πρόσβασης μεταξύ των principals. Στο Mylar ορίζεται η σχέση πρόσβασης "$has\ access\ to$" μεταξύ των principals A και B ως εξής:

$A\ has\ access\ to\ B$ όταν η principal A έχει πρόσβαση στα μυστικά κλειδιά της principal B.

Η σχέση $A\ has\ access\ to\ B$ υλοποιείται κρυπτογραφικά ως εξής. Τα μυστικά κλειδιά της B κρυπτογραφούνται με το δημόσιο κλειδί e_{pub} της A, δημιουργείται δηλαδή ένα «εμφωλιασμένο» κλειδί (wrapped key). Με τον τρόπο αυτό εξασφαλίζεται η πρόσβαση της A στα μυστικά κλειδιά της B.

Επιπλέον πρέπει να σημειωθεί ότι η σχέση είναι μεταβατική. Δηλαδή, όταν ισχύει «$A\ has\ access\ to\ B$» και «$B\ has\ access\ to\ C$», τότε ισχύει και «$A\ has\ access\ to\ C$».

Η δημιουργία της σχέσης $A\ has\ access\ to\ B$ γίνεται με τους εξής τρόπους:

6. Κατά την δημιουργία της B από την A.

7. Όταν μια τρίτη principal C , η οποία έχει ήδη πρόσβαση στη B, δίνει πρόσβαση και στην A.

Τα παραπάνω υλοποιούνται προγραμματιστικά μέσω κλήσεων, αντίστοιχα, των παρακάτω συναρτήσεων της διεπαφής του Mylar.

1. $princ\_create(name, creator\_princ)$, η οποία καλείται για την δημιουργία της principal $name$ από την principal $creator\_princ$.

2. $granter.add\_access(grantee)$, η οποία καλείται για να δοθεί πρόσβαση στην principal $grantee$ στα δεδομένα της principal $granter$.

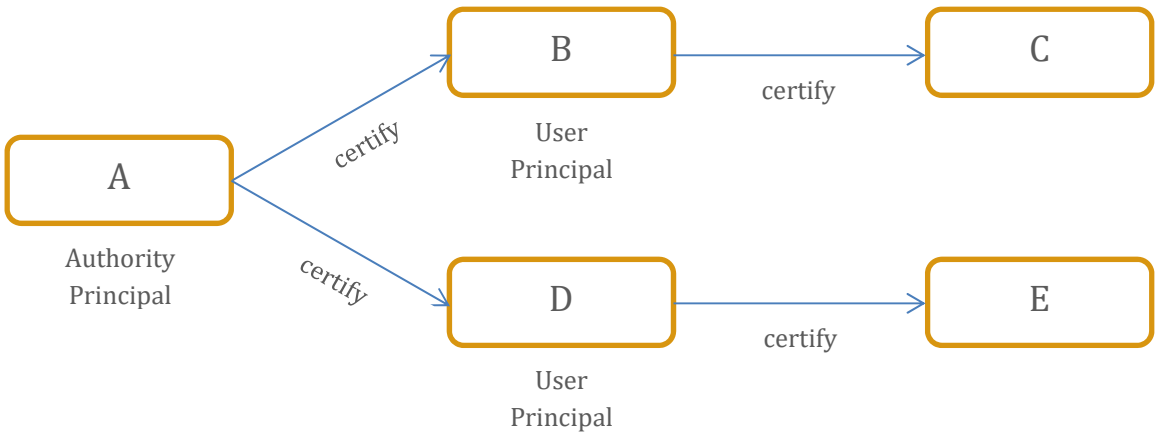
Οι σχέσεις $has\ access\ to$ σχηματίζουν ένα κατευθυνόμενο γράφο, τον Γράφο Πρόσβασης, ο οποίος απεικονίζει την πολιτική πρόσβασης της εφαρμογής, όπως π.χ. στο Σχήμα 4.1. Για παράδειγμα, για τις οντότητες του Σχήματος 4.1, η principal A έχει πρόσβαση στα δεδομένα της B, της D όπως και της C λόγω μεταβατικότητας της σχέσης. Η principal B έχει πρόσβαση μόνο στα δεδομένα της C. Σε κάθε τέτοιο γράφο,

λόγω του τρόπου δημιουργίας των σχέσεων, οι διαδρομές έχουν σαν κόμβο – πηγή (root node) κάποια principal χρήστη. Έτσι, λόγω της μεταβατικότητας της σχέσης, οι διαδρομές αυτές απεικονίζουν όλες τις principals που έχει πρόσβαση ο χρήστης που αντιστοιχεί στον κόμβο – πηγή. Σημειώνεται επίσης ότι σε κάθε μια τέτοια διαδρομή αντιστοιχεί και μια αλυσίδα «εμφωλιασμένων» κλειδιών (wrapped keys).

Έτσι, στο πλαίσιο λειτουργίας τη εφαρμογής, όταν μια principal A, π.χ. ένας χρήστης, ζητήσει πρόσβαση στα δεδομένα της principal B, π.χ. ενός αρχείου, η εφαρμογή θα αναζητήσει μια αλυσίδα wrapped keys που να ξεκινά από την principal της A και να φτάνει στην principal της B. Η ύπαρξή της εξασφαλίζει ότι η principal A μπορεί να έχει πρόσβαση στα κρυπτογραφημένα δεδομένα της principal B και επίσης δείχνει και τον τρόπο με τον οποίο θα γίνει αυτό.

Πολιτική Πιστοποίησης

Αντίστοιχα με την πολιτική πρόσβασης, το Mylar δίνει την δυνατότητα σε μια εφαρμογή να καθορίσει την πολιτική πιστοποίησης με την βοήθεια των principals δημιουργώντας μεταξύ τους σχέσεις πιστοποίησης.



Σχήμα 4.2: Παράδειγμα Γράφου Πιστοποίησης ως προς τις αλυσίδες πιστοποίησης, όπου η κάθε principal πιστοποιεί την επόμενη στην αλυσίδα.

Η πιστοποίηση της ταυτότητας των principals γίνεται μέσω της πιστοποίησης των δημόσιων κλειδιών τους.

Η πιστοποίηση των δημόσιων κλειδιών των principals χρηστών – που ουσιαστικά ισοδυναμεί με αυθεντικοποίηση των χρηστών - γίνεται κατά τη δημιουργία τους, με την βοήθεια των authority principals, δηλαδή του IDP ή κάποιων στατικών principals που έχουμε δημιουργήσει. Στην πρώτη περίπτωση, η εφαρμογή επικοινωνεί με το IDP, αποστέλλοντας απαραίτητες πληροφορίες για την ταυτοποίηση του χρήστη, όπως το όνομα, και τα δημόσια κλειδιά του και στη συνέχεια λαμβάνει από το IDP το ψηφιακό πιστοποιητικό με την υπογραφή του IDP. Στη δεύτερη περίπτωση, το ψηφιακό πιστοποιητικό δημιουργείται από την εφαρμογή και για τη υπογραφή του χρησιμοποιείται το ιδιωτικό κλειδί d_{priv} μιας στατικής principal. Η πρόσβαση στο ιδιωτικό κλειδί γίνεται όπως έχει περιγράφεται στην υπο-ενότητα 4.3.1 .

Η πιστοποίηση των δημόσιων κλειδιών όλων των άλλων principals γίνεται μέσω των principals που τις δημιουργούν. Δηλαδή, δημιουργείται από την εφαρμογή ένα ψηφιακό πιστοποιητικό που περιέχει τουλάχιστον το όνομα και τα δημόσια κλειδιά $e_{pub,}$ και d_{pub} της principal και τα οποία υπογράφονται ψηφιακά με το ιδιωτικό κλειδί d_{priv} της principal που τη δημιούργησε.

Έτσι στο Mylar ορίζεται τη σχέση πιστοποίησης "$certify$" μεταξύ των principals A και B ως εξής:

$A\ certify\ B$ όταν η principal A πιστοποιεί τα δημόσια κλειδιά e_{pub} και d_{pub} της principal B.

Τα παραπάνω υλοποιούνται προγραμματιστικά μέσω κλήσεων, αντίστοιχα, των παρακάτω συναρτήσεων της διεπαφής του Mylar.

1. $create\_user(uname, password, auth\_princ)$. Η οποία καλείται για την δημιουργία του χρήστη $uname$ με κωδικό πρόσβασης $password$ της οποίας τα δημόσια κλειδιά e_{pub}, d_{pub} υπογράφονται από την authority principal $auth\_princ$.

2. $princ\_create(name, creator\_princ)$. Η οποία καλείται για την δημιουργία της principal $name$ από την principal $creator\_princ$.

Οι σχέσεις $certify$ σχηματίζουν ένα κατευθυνόμενο γράφο, τον Γράφο Πιστοποίησης, ο οποίος απεικονίζει την πολιτική πιστοποίησης της εφαρμογής, όπως π.χ. στο Σχήμα 4.2.

Επιπλέον, λόγω του τρόπου δημιουργίας των σχέσεων, οι διαδρομές έχουν σαν κόμβο – πηγή (root node) κάποια authority principal. Ακόμα να παρατηρήσουμε ότι σε κάθε μια τέτοια διαδρομή αντιστοιχεί και μια αλυσίδα πιστοποιητικών.

Οι αλυσίδες πιστοποιητικών είναι παρόμοιες με του προτύπου X.509, σχετικά με τον τρόπο δημιουργίας των σχέσεων πιστοποίησης.

Έτσι, κατά την αναζήτηση μιας principal, η εφαρμογή μπορεί να ελέγξει για την ύπαρξη μιας αλυσίδας πιστοποιητικών που ξεκινούν από την principal και καταλήγουν σε μια authority principal. Η ύπαρξή της επιβεβαιώνει τη γνησιότητα του δημόσιων κλειδιών e_{pub}, d_{pub} της principal.

Ο παραπάνω έλεγχος από την εφαρμογή εξασφαλίζει ότι όταν μια principal A, π.χ. ένας χρήστης, ζητήσει τα δημόσια κλειδιά μια άλλης B, π.χ. ενός αρχείου, τότε η εφαρμογή θα του τα προσφέρει, αν αυτό είναι έγκυρο. Αυτό όμως που δεν εξασφαλίζεται είναι το ότι ο χρήστης A επέλεξε το σωστό αρχείο B. Η επιλογή παύει να είναι εύκολη από την στιγμή που τίποτα δεν αποκλείει την ύπαρξη και ενός άλλου αρχείου με το ίδιο όνομα B. Επιλέγοντας το λάθος αρχείο B, ο χρήστης A θα μοιραστεί πληροφορία, το πιθανότερο, με λάθος χρήστες. Για την αποφυγή τέτοιων προβλημάτων θα πρέπει ο σχεδιασμός της εφαρμογής να είναι τέτοιος που να δίνει την ευχέρεια στο χρήστη να διακρίνει σαφώς τις principals της εφαρμογής, ώστε να επιλέξει σωστά. Αυτό μπορεί να γίνει εμφανίζοντας στο χρήστη όχι μόνο το όνομα της principal αλλά και τα ονόματα των principals της αλυσίδας πιστοποιητικών που αντιστοιχεί στην οντότητα. Με τον τρόπο αυτό, και με την προϋπόθεση ότι τα ονόματα των principals αντικατοπτρίζουν εννοιολογικά της αντίστοιχες οντότητες, ο χρήστης έχει τη δυνατότητα να επιλέξει σωστά.

4.4.2 Αναζήτηση (Searching)

Τα κρυπτογραφικά συστήματα που υποστηρίζουν την αναζήτηση λέξεων σε κρυπτογραφημένα δεδομένα, ανεξάρτητα από τον αλγόριθμο κρυπτογράφησης που χρησιμοποιούν, έχουν ένα κοινό απαραίτητο χαρακτηριστικό στη διαδικασία της αναζήτησης. Για να αναζητήσουν μια λέξη σε κρυπτογραφημένα δεδομένα μετατρέπουν, μέσω του κλειδιού κρυπτογράφησης, την αρχική λέξη αναζήτησης σε μία λεκτική μονάδα αναζήτησης (search token) με τον οποίο τελικά γίνεται η εκτέλεση. Άρα,

για έναν χρήστη που θέλει να αναζητήσει μια λέξη σε N έγγραφα, τα οποία έχουν κρυπτογραφηθεί με διαφορετικά κλειδιά, λόγω κοινής χρήσης, θα πρέπει να δημιουργηθούν N search tokens. Εφαρμόζοντας ένα τέτοιο κρυπτογραφικό σύστημα στην πλατφόρμα του Mylar σημαίνει ότι η δημιουργία των N search tokens θα πρέπει να γίνει στον browser του χρήστη, μιας και μόνο εκεί υπάρχει μη κρυπτογραφημένη η λέξη αναζήτησης. Αυτό όμως θα ήταν αναποτελεσματικό για τους εξής λόγους:

1. Υπολογιστικό κόστος. Η αναζήτηση των κλειδιών κρυπτογράφησης γίνεται μέσω του Γράφου Πρόσβασης και είναι μια υπολογιστικά δαπανηρή διαδικασία. Εδώ πρέπει να προσθέσουμε και το υπολογιστικό κόστος μετατροπής της λέξης-κλειδί σε search token.

2. Δικτυακό κόστος. Η επιβάρυνση οφείλεται στην μεταφορά των κλειδιών κρυπτογράφησης στον browser του χρήστη για τη δημιουργία των search tokens, όπως και από την μεταφορά στον εξυπηρετητή, των search tokens, για την εκτέλεση της αναζήτησης.

Έτσι στην πλατφόρμα Mylar έχει ενσωματωθεί ένα κρυπτογραφικό σύστημα αναζήτησης, με κεντρική ιδέα την αποσυμφόρηση του browser του χρήστη και με σκοπό την αποτελεσματικότητα. Το σύστημα αυτό ονομάζεται Multi-key Searchable Encryption [30] και έχει δημιουργηθεί με σκοπό την χρήση του στο Mylar, χωρίς αυτό να είναι δεσμευτικό, και περιγράφεται παρακάτω.

Multi-key Searchable Encryption (MK)

Το MK βασίζεται στην εξής ιδέα: Η αναζήτηση από έναν χρήστη μιας λέξης, σε ένα σύνολο κρυπτογραφημένων δεδομένων, να καταστεί δυνατή με την δημιουργία μόνο μίας search token στον browser του χρήστη.

Περιγραφή

Στο MK ο κρυπτογραφικός αλγόριθμος βασίζεται στην θεωρία των ελλειπτικών καμπυλών και πρέπει να επισημανθεί ότι δεν περιλαμβάνεται συνάρτηση αποκρυπτογράφησης. Αυτό σημαίνει ότι τα δεδομένα που κρυπτογραφούνται με το MK δεν αποκρυπτογραφούνται. Έτσι, στις εφαρμογές που θα χρησιμοποιήσουν το MK,

προκύπτει η ανάγκη για παράλληλη κρυπτογράφηση των δεδομένων με κάποιο άλλο κρυπτογραφικό σύστημα ώστε να είναι δυνατή η αποκρυπτογράφησή τους. Για τον σκοπό αυτό στο Mylar χρησιμοποιείται το σύστημα συμμετρικής κρυπτογράφησης AES. Η συνεργασία αυτών των κρυπτογραφικών συστημάτων θα φανεί καθαρά στην επόμενη παράγραφο, όπου και περιγράφεται η λειτουργία της αναζήτησης.

Αναζήτηση στο MK.

Όπως και στα υπόλοιπα συστήματα κρυπτογράφησης που υποστηρίζουν αναζήτηση, έτσι και στο MK η αναζήτηση μιας λέξης σε κρυπτογραφημένα δεδομένα απαιτεί τη δημιουργία διαφορετικών search tokens, μίας ανά κλειδί κρυπτογράφησης. Στο MK όμως αυτό γίνεται στον εξυπηρετητή και η διαδικασία είναι η εξής.

Έστω ότι ο χρήστης U με κλειδί Uk εκτελεί μια αναζήτηση της λέξης w σε κρυπτογραφημένα δεδομένα με κλειδιά $k_1, \ldots, k_n$. Στον browser του χρήστη η λέξη w μετατρέπεται, με την χρήση του Uk σε ένα αρχικό search token, έστω tk_{Uk}^{w}. Η search token tk_{Uk}^{w} αποστέλλεται στον εξυπηρετητή. Ο εξυπηρετητής λοιπόν διαθέτει αυτό το ενιαίο search token και όχι τα ξεχωριστά tokens tk_{k1}^{w}, … , tk_{kn}^{w} τα οποία θα του επέτρεπαν, με κάποια τεχνική «κρυπτογράφησης με δυνατότητα αναζήτησης» (searchable encryption) να αναζητεί τις εν λόγω λέξεις στα κρυπτογραφημένα δεδομένα. Ωστόσο, βάσει της προσέγγισης που έχει υιοθετηθεί στο Mylar, ο εξυπηρετητής αναλαμβάνει να μετατρέπει τη tk_{Uk}^{w} σε διαφορετικές search tokens, έστω $\text{tk}_{k_1}^{w}, \ldots \text{tk}_{k_n}^{w}$ μία για κάθε κλειδί κρυπτογράφησης k_i, οι οποίες είναι τέτοιες ώστε μέσω αυτών να εκτελείται τελικά η αναζήτηση. Είναι σημαντικό να παρατηρηθεί ότι η μετατροπή γίνεται χωρίς την χρήση των κλειδιών k_i. Η μετατροπή γίνεται με την βοήθεια κρυπτογραφικών ποσοτήτων που ονομάζονται «διαφορές» (deltas), έστω $\Delta_{Uk \to k_i}$. Οι «διαφορές» υπολογίζονται στο browser του χρήστη μόνο μία φορά, όταν ο χρήστης αποκτά πρόσβαση στα δεδομένα μια οντότητας και στη συνέχεια αποθηκεύονται στον εξυπηρετητή. Η τεχνική κρυπτογράφησης που χρησιμοποιείται είναι τέτοια ώστε οι «διαφορές» να εξαρτώνται μόνο από τα κλειδιά και όχι από τις λέξεις-μηνύματα που θα κρυπτογραφηθούν.

Με τον τρόπο αυτό, το MK ελαχιστοποιεί το κόστος υπολογισμού στον browser του χρήστη όπως και την επιβάρυνση του δικτύου. Η επιβάρυνση του εξυπηρετητή για την

μετατροπή της search token είναι πολύ μικρή, μιας και αυτό γίνεται χωρίς τα κλειδιά κρυπτογράφησης των δεδομένων.

Ενσωμάτωση στο Mylar

Αρχικά, κατά την ανάπτυξη της εφαρμογής, μαζί με τον ορισμό των οντοτήτων για τις οποίες θα δημιουργηθούν principals, καθορίζονται και τα δεδομένα των principals για τα οποία θα υπάρχει η δυνατότητα αναζήτησης. Αυτό γίνεται από τον προγραμματιστή με την σήμανση των δεδομένων ως «αναζητήσιμα» (searchable). Τα δεδομένα αυτά κρυπτογραφούνται δύο φορές, μία φορά όπως τα δεδομένα όλων των principals με το συμμετρικό σύστημα AES και μία ακόμα με το σύστημα ΜΚ. Το ΜΚ θα το χρησιμοποιήσει για να βρει τα δεδομένα αυτά και το AES για την αποκρυπτογράφησή τους.

Έτσι, με την δημιουργία κάθε νέας principal U, π.χ. ενός χρήστη, παράγεται, όπως έχει ήδη αναφερθεί στην ενότητα 4.3, το κλειδί κρυπτογράφησης, έστω Uk, το οποίο χρησιμοποιείται μόνο από το ΜΚ. Στη συνέχεια, κάθε φορά που η principal U αποκτά πρόσβαση σε κάποια άλλη principal A, π.χ. ένα έγγραφο, με αντίστοιχο κλειδί Ak, δηλαδή δημιουργείται η σχέση $U\ has\ access\ to\ A$, το αντίστοιχο εμφωλιασμένο κλειδί (wrapped key) που δημιουργείται περιλαμβάνει και το κλειδί Ak. Ακολούθως όταν ο χρήστης U συνδεθεί στην εφαρμογή, αν δεν είναι ήδη, στον browser του χρήστη ανακτάται το Ak από το wrapped key και υπολογίζεται η διαφορά $\Delta_{Uk\to Ak}$, η οποία αποθηκεύεται στον εξυπηρετητή. Ο υπολογισμός του delta συμβαίνει μόνο μία φορά για κάθε σχέση $has\ access\ to$. Οποτεδήποτε ο χρήστης U θέλει να αναζητήσει μία οποιαδήποτε λέξη w εντός του A, το token tk_{Uk}^{w} που θα δημιουργήσει είναι τέτοιο ώστε ο εξυπηρετητής, λαμβάνοντας αυτό από το χρήστη και γνωρίζοντας τη διαφορά $\Delta_{Uk\to Ak}$, να μπορεί να υπολογίσει το «ισοδύναμο» token με το οποίο θα αναζητά – με τεχνική searchable encryption – τη λέξη w εντός των δεδομένων του αρχείου με κλειδί Ak: αυτό ισχύει ανεξαρτήτως της λέξης w.

Στην περίπτωση επιτυχούς αναζήτησης, ο εξυπηρετητής επιστρέφει στον χρήστη τα δεδομένα με τη συμμετρικά κρυπτογραφημένη μορφή τους (AES) ώστε αυτά να μπορούν να αποκρυπτογραφηθούν.

Ασφάλεια

Αν και το ΜΚ είναι σημασιολογικά ασφαλές, όπως έχει ήδη αναφερθεί σε κάθε σύστημα SE έτσι και στο ΜΚ, υπάρχει εν τούτοις κάποια διάχυση πληροφορίας που αφορά το μοτίβο των λέξεων αναζήτησης ή και το μοτίβο εμφάνισης των λέξεων της αναζήτησης στα δεδομένα [17].

Η αδυναμία αυτή μαζί με την κακή χρήση της διεπαφής του Mylar από τον προγραμματιστή, μη ελέγχοντας και αποκρύπτοντας από τον τελικό χρήστη την πλήρη αλυσίδα των principals που πιστοποιούν μια οντότητα, μπορεί να οδηγήσει στην υποκλοπή των δεδομένων του χρήστη μέσω μιας λεξικογραφικής επίθεσης όπως αυτή που έγινε το 2016 από τους Grubbs-McPherson-Naveed-Rinstenpart-Shmatikov [15]. Οι δημιουργοί του Mylar απάντησαν [29] ότι είναι απαραίτητη η σωστή χρήση της διεπαφής του Mylar για την ασφάλεια απέναντι σε επιθέσεις αυτής της μορφής. Αυτό έχει ήδη τονισθεί στην υπο-ενότητα 4.4.1 και είναι μια αδυναμία του Mylar που πρέπει να λαμβάνεται πολύ σοβαρά υπόψη από τον προγραμματιστή που θα το χρησιμοποιήσει.

Κεφάλαιο 5
Πρακτική εφαρμογή του Mylar: Η περίπτωση του kChat

Οι κατασκευαστές της πλατφόρμας Mylar επέλεξαν για την πρώτη δοκιμή της, την εφαρμογή ανταλλαγής μηνυμάτων(chat application) kChat [19]. Η επιλογή μιας εφαρμογής ανταλλαγής μηνυμάτων βασίστηκε κυρίως στην ανάγκη που παρουσιάζουν αυτού του είδους οι εφαρμογές για κοινή χρήση εμπιστευτικών δεδομένων, αλλά και στη μεγάλη δημοφιλία τους. Έτσι, μια επιτυχημένη ενσωμάτωση σε μια τέτοια εφαρμογή θα αποδείκνυε και την μεγάλη πρακτική χρησιμότητα του Mylar.

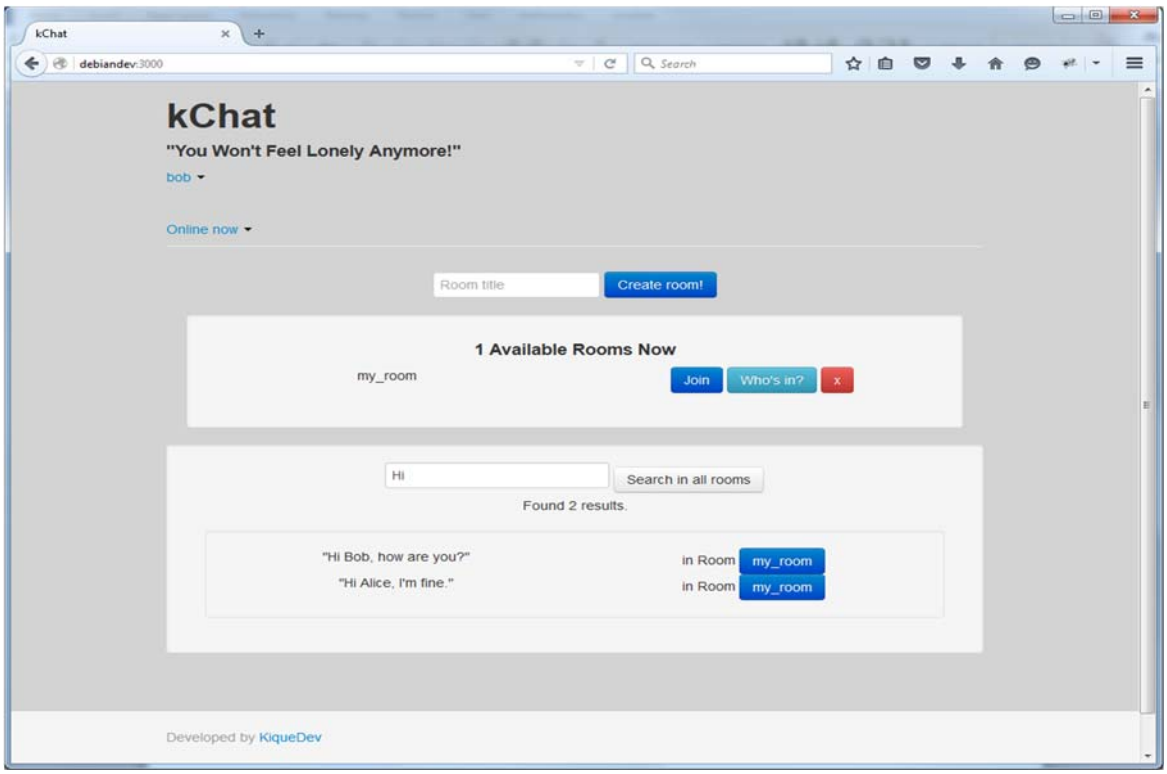
Στο κεφάλαιο αυτό θα περιγραφούν οι βασικές λειτουργίες του kChat, ο τρόπος που έγινε η ενσωμάτωση του Mylar και θα αναδειχθούν τα χαρακτηριστικά ασφαλείας που η ενσωμάτωση αυτή συνεισέφερε στην εφαρμογή. Επίσης θα μελετηθούν, με τη χρήση κατάλληλων εργαλείων λογισμικού, και οι επιπτώσεις της εν λόγω ενσωμάτωσης στην απόδοση των βασικών λειτουργιών του kChat.

5.1 kChat

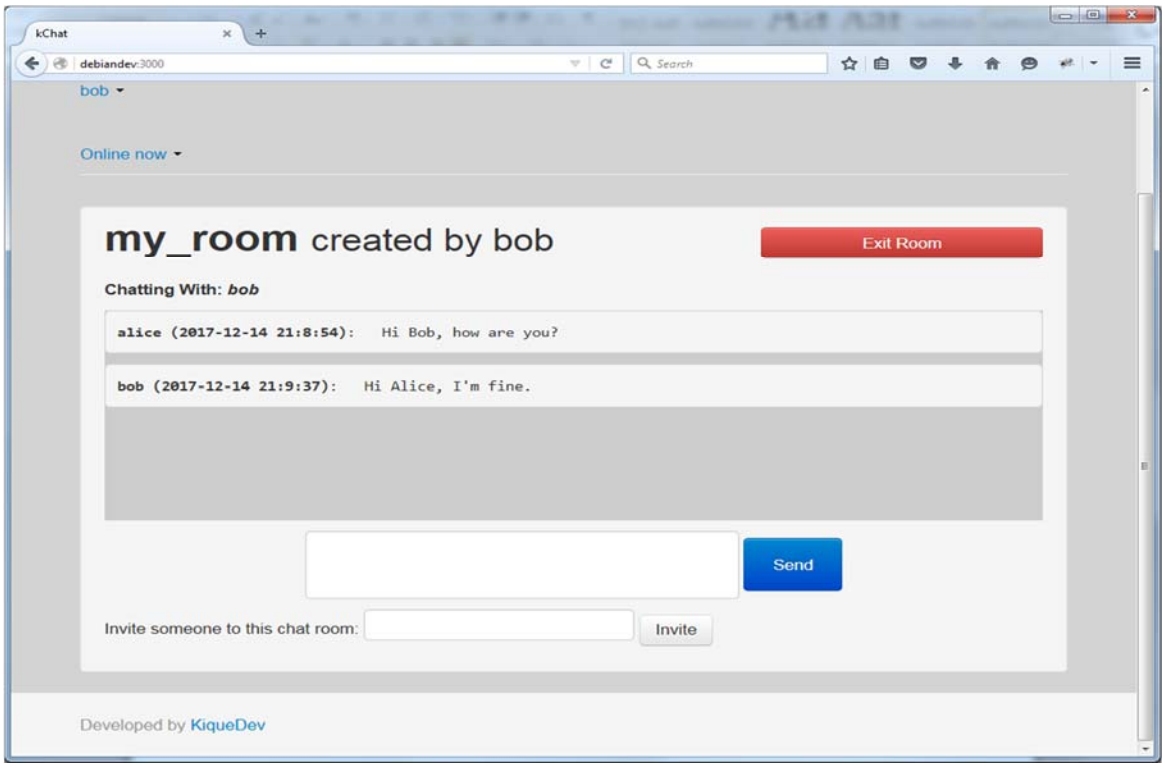
Το kChat είναι μια εφαρμογή ιστού, υλοποιημένη στην πλατφόρμα Meteor, η οποία έχει ως βασική της λειτουργία την ανταλλαγή μηνυμάτων μεταξύ των χρηστών της.

Αρχικά, ένας επισκέπτης της εφαρμογής kChat δημιουργεί ένα λογαριασμό χρήστη. Στη συνέχεια ο χρήστης έχει τη δυνατότητα δημιουργίας ενός ή περισσοτέρων «δωματίων επικοινωνίας» στα οποία, αφού συνδεθεί, θα μπορεί να ανταλλάξει μηνύματα με άλλους χρήστες που είναι συνδεδεμένοι στο ίδιο δωμάτιο. Στη διαθέσιμη αρχική εφαρμογή kChat δεν υπήρχε η λειτουργικότητα της πρόσκλησης ενός χρήστη σε ένα δωμάτιο. Αυτό ήταν απόρροια της σχεδιαστικής απόφασης που είχαν λάβει οι υλοποιητές της εφαρμογής, ώστε όλοι οι χρήστες να έχουν πρόσβαση σε όλα τα δωμάτια της εφαρμογής. Επιπλέον, η αρχική εφαρμογή δεν έδινε την δυνατότητα αναζήτησης των μηνυμάτων.

Οι κατασκευαστές του Mylar, πρόσθεσαν τις δύο αυτές λειτουργικότητες στην έκδοση του kChat που ενσωματώνει το Mylar. Έτσι στην έκδοση αυτή, μόνο οι προσκεκλημένοι χρήστες ενός δωματίου επικοινωνίας έχουν πρόσβαση στα μηνύματα του δωματίου αυτού και επιπλέον, ένας χρήστης μπορεί να αναζητήσει λέξεις στα μηνύματα των δωματίων στα οποία έχει πρόσβαση. Στα επόμενα, η έκδοση του kChat που ενσωματώνει το Mylar θα αναφέρεται ως EncChat, ώστε να είναι εύκολα διακριτές οι αναφορές στις δύο εκδόσεις. Οι παρακάτω εικόνες 5.1 και 5.2 είναι από το EncChat.



Εικόνα 5.1: Κύρια σελίδα του EncChat.



Εικόνα 5.2: Δωμάτιο επικοινωνίας του EncChat.

5.2 kChat + Mylar = EncChat (Ενσωμάτωση)

Όπως σε κάθε εφαρμογή που ενσωματώνει το Mylar, έτσι και στο kChat αρχικά καθορίζονται οι οντότητες της εφαρμογής για τις οποίες θα δημιουργηθούν οι αντίστοιχες principals. Στο kChat, αυτές είναι οι χρήστες και τα δωμάτια επικοινωνίας, μιας και τα δεδομένα – μηνύματα – κάθε δωματίου θα πρέπει να είναι προσβάσιμα μόνο από τους χρήστες που έχουν πρόσβαση στο δωμάτιο αυτό.

Οι κύριες προγραμματιστικές αλλαγές που έγιναν στο kChat, ώστε να καλούνται οι κατάλληλες συναρτήσεις της διεπαφής του Mylar, είναι οι ακόλουθες:

1. **Σήμανση δεδομένων.** Ειδική σήμανση προστίθεται σα εμπιστευτικά δεδομένα – μηνύματα – έτσι ώστε να κρυπτογραφούνται και παράλληλα να υπάρχει η δυνατότητα αναζήτησης σε αυτά. Αυτό γίνεται κατά την εκκίνηση της εφαρμογής μέσω των αντίστοιχων κλήσεων στις παρακάτω συναρτήσεις:

   $Messages.encrypted(\{"message": "roomprinc"\})$

   $Messages.searchable("message")$

2. **Δημιουργία χρηστών:** Η διαδικασία τροποποιήθηκε έτσι ώστε να καλείται η συνάρτηση

   $create\_user(uname, password, idp)$

   με την οποία δημιουργείται ο χρήστης $uname$ και η αντίστοιχη principal.

3. **Δημιουργία δωματίων επικοινωνίας.** Η διαδικασία τροποποιήθηκε έτσι ώστε να καλούνται οι συναρτήσεις:

   $princ\_current()$

   $princ\_create(roomtitle, princ\_current())$

Η πρώτη επιστρέφει την principal του χρήστη που την καλεί και χρησιμοποιείται από την δεύτερη για την πιστοποίηση της principal του δωματίου $roomtitle$ που δημιουργείται.

4. **Σύνδεση σε δωμάτιο.** Η διαδικασία τροποποιήθηκε έτσι ώστε καλεί την συνάρτηση:

   $princ\_lookup(room.name, room.creator, idp)$

   η οποία επιστρέφει την principal του δωματίου $name$, της οποίας τα κλειδιά καθίστανται διαθέσιμα στον χρήστη που την καλεί.

5. **Αποστολή μηνύματος.** Η διαδικασία τροποποιήθηκε έτσι ώστε καλεί την συνάρτηση:

$Messages.insert(\{message: msg, room: room.id, date: Date(), princ: room\_princ\})$

   η οποία καταχωρεί το μήνυμα msg, κρυπτογραφημένο με τα κλειδιά της principal του δωματίου $room\_princ$, στη βάση δεδομένων της εφαρμογής.

6. **Πρόσκληση χρήστη.** Η διαδικασία καλεί τις συναρτήσεις:

   $princ\_lookup(username, idp)$

   $room\_princ.add\_access(princ\_lookup(username, idp))$

   Η πρώτη επιστρέφει την principal του χρήστη $username$ και χρησιμοποιείται από την δεύτερη για να δοθεί πρόσβαση στον χρήστη $username$ στο δωμάτιο με principal $room\_princ$.

7. **Αναζήτηση.** Η διαδικασία αναζήτησης λέξης στα μηνύματα των δωματίων καλεί τις συναρτήσεις:

   $princ\_current()$

   $Messages.search(word, "message", princ\_current(), all, all)$

Η πρώτη επιστρέφει την principal του χρήστη που την καλεί και χρησιμοποιείται από την δεύτερη για την δημιουργία της λεκτικής μονάδας αναζήτησης που αντιστοιχεί στη λέξη *word* και την principal του χρήστη, την οποία θα στείλει στον εξυπηρετητή για την εκτέλεση της αναζήτησης.

Έτσι, όταν ένας χρήστης Α δημιουργήσει ένα δωμάτιο επικοινωνίας R, ταυτόχρονα δημιουργείται και η principal του δωματίου, η οποία έχει το ίδιο όνομα με αυτό του δωματίου και υπογράφεται από την principal του χρήστη. Στην συνέχεια όταν ο χρήστης συνδεθεί στο δωμάτιο και αποστείλει ένα μήνυμα, αυτό κρυπτογραφείται με τα κλειδιά της principal του δωματίου. Στην περίπτωση που ο χρήστης Α προσκαλέσει κάποιον άλλο χρήστη Β στο δωμάτιο R, η εφαρμογή δίνει πρόσβαση στον χρήστη Β στο δωμάτιο R. Τέλος, όταν ο χρήστης Α εκτελέσει μια αναζήτηση λέξης επί των μηνυμάτων, η εφαρμογή θα χρησιμοποιήσει την principal του Α για να δημιουργήσει τη λεκτική μονάδα αναζήτησης, την οποία θα στείλει στον εξυπηρετητή για την εκτέλεση της αναζήτησης.

Η ενσωμάτωση του Mylar στο kChat έγινε με πολύ λίγες προγραμματιστικές αλλαγές, οι οποίες αναφέρονται παραπάνω, χωρίς να επηρεάσει την εμφάνιση της εφαρμογής και τη διεπαφή με τον τελικό χρήστη.

5.3 Υποδομή και εγκατάσταση

Η υποδομή που χρησιμοποιήθηκε σε υλικό (hardware) είναι η εξής:

1. **Εξυπηρετητής:** Μία εικονική μηχανή με έναν επεξεργαστή (CPU) Intel i3-2120@3.30Ghz και 4GB μνήμη (RAM). Το λειτουργικό σύστημα είναι Linux και συγκεκριμένα Debian GNU/Linux 8.7 (jessie)

2. **Πελάτης:** Μία φυσική μηχανή με δύο επεξεργαστές (CPU) Intel i3-2120@3.30Ghz και 8GB μνήμη (RAM). Το λειτουργικό σύστημα είναι Windows 7 Professional.

Η υποδομή σε λογισμικό (software) που εγκαταστάθηκε είναι η κάτωθι:

1. **HPE LoadRunner 12.55 Community Edition:** Εφαρμογή δοκιμών λογισμικού της εταιρείας Micro Focus [21].

2. **Robomongo 1.0.0:** Εφαρμογή ανοιχτού κώδικα για την διαχείριση της βάσης δεδομένων MongoDB [33].

3. **WireShark 1.12.1:** Εφαρμογή ανοιχτού κώδικα για την ανάλυση και παρακολούθηση του δικτύου [41].

4. **VirtualBox 5.1.18:** Εφαρμογή ανοιχτού κώδικα για την δημιουργία και διαχείριση εικονικών μηχανών, της εταιρείας Oracle [25].

Σε όλες τις δοκιμές χρησιμοποιήθηκε το πρόγραμμα πλοήγησης Mozilla Firefox 42.0.

Εγκατάσταση

Για την εγκατάσταση των δύο εκδόσεων της εφαρμογής kChat, δημιουργήθηκε μέσω της εφαρμογής VirtualBox μία εικονική μηχανή, με τα χαρακτηριστικά που αναφέρθηκαν παραπάνω, η οποία έχει τον ρόλο του εξυπηρετητή. Για την EncChat είναι απαραίτητο να γίνει και η εγκατάσταση της πλατφόρμας Mylar. Ο πηγαίος κώδικας των εφαρμογών και της πλατφόρμας είναι διαθέσιμος στους παρακάτω συνδέσμους (τελευταία πρόσβαση: 13/10/2017).

kChat: https://github.com/KiqueDev/kChat.git

Mylar: git://g.csail.mit.edu/mylar

EncChat: git://g.csail.mit.edu/EncChat

Η εγκατάσταση έγινε μέσω της εφαρμογής Git με τις παρακάτω εντολές αντίστοιχα

$$git\ clone\ https://github.com/KiqueDev/kChat.git$$

$$git\ clone\ -b\ public\ git://g.csail.mit.edu/mylar$$

$$git\ clone\ git://g.csail.mit.edu/EncChat$$

Επιπλέον για την ενεργοποίηση της δυνατότητας αναζήτησης στο EncChat προστέθηκε το πακέτο λογισμικού search του Mylar με την εκτέλεση της εντολής

add search

στον φάκελο εγκατάστασης του Mylar.

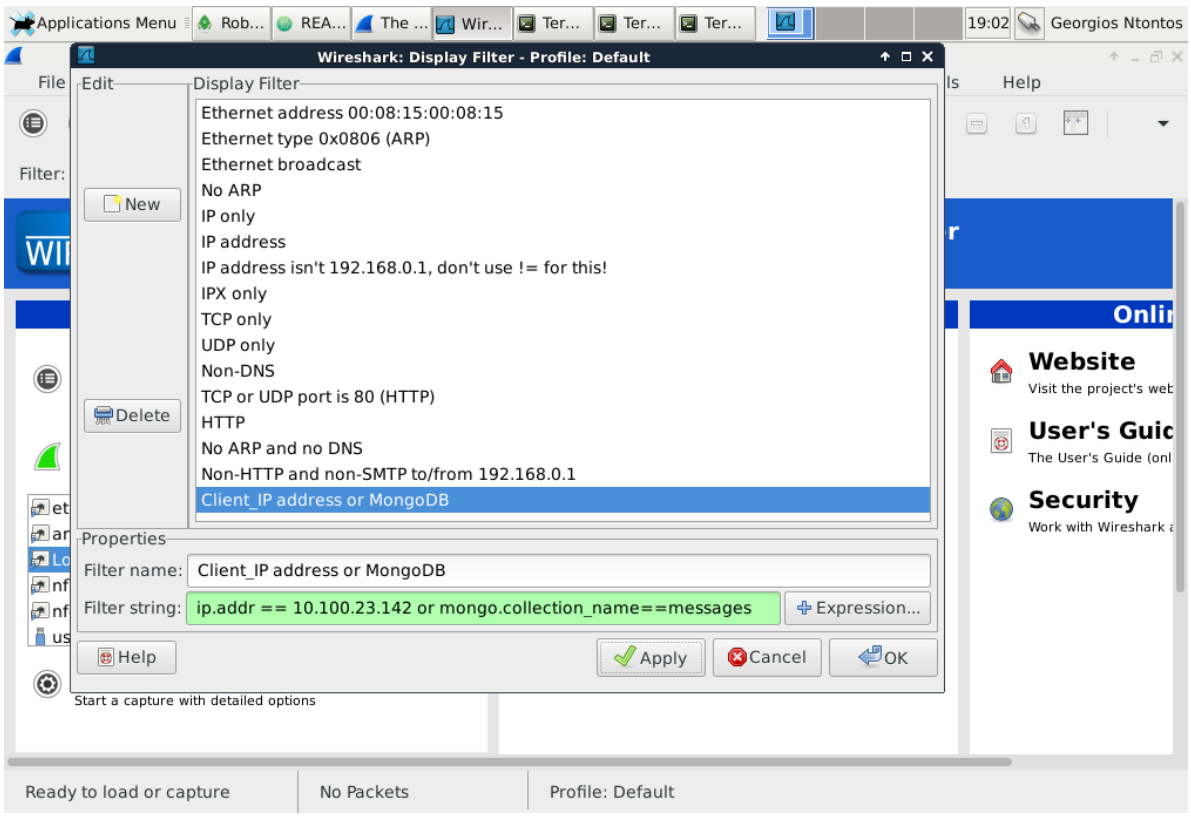
Οι δύο εκδόσεις του kChat έχουν υλοποιηθεί με την πλατφόρμα Meteor, η οποία χρησιμοποιεί την βάση δεδομένων MongoDB. Η πλατφόρμα Meteor και η βάση MongoDB εγκαθίστανται παράλληλα με τις εκδόσεις του kChat και δεν χρειάζεται ξεχωριστή εγκατάσταση.

5.4 Κρυπτογράφηση από-άκρο-σε-άκρο

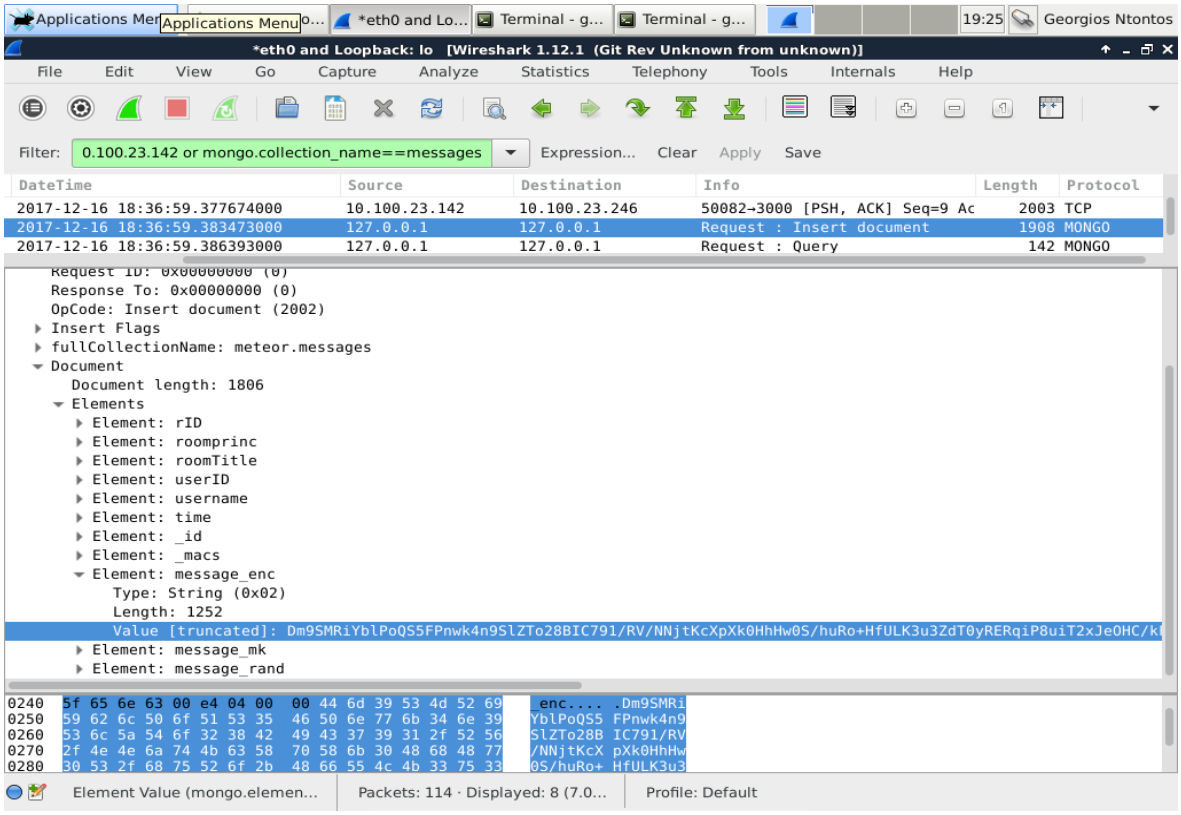
Για να διαπιστωθεί η κρυπτογράφηση από-άκρο-σε-άκρο, στο EncChat, χρησιμοποιήθηκε η εφαρμογή WireShark. Η εφαρμογή Wireshark είναι ένα ανοιχτού κώδικα λογισμικό, το οποίο χρησιμοποιείται για την ανάλυση και παρακολούθηση του δικτύου, και είναι διαθέσιμο για όλα τα κύρια λειτουργικά συστήματα, π.χ. Windows, Linux, Mac OS X.

Η εγκατάσταση του WireShark έγινε στον εξυπηρετητή και παραμετροποιήθηκε κατάλληλα ώστε να φιλτράρει τα πακέτα του δικτύου από και προς συγκεκριμένη διεύθυνση IP (IP address), αυτή του πελάτη, και παράλληλα τα πακέτα από και προς τη βάση δεδομένων MongoDB. Για τον σκοπό αυτό δημιουργήθηκε στο WireShark το φίλτρο που εμφανίζεται στην Εικόνα 5.4, το οποίο φιλτράρει τα πακέτα που αφορούν την διεύθυνση IP του πελάτη, 10.100.23.142, και αυτά που αφορούν τη βάση δεδομένων και συγκεκριμένα την συλλογή μηνυμάτων.

Η καταγραφή των πακέτων δεδομένων έγινε κατά την διάρκεια της εκτέλεσης όλων των βασικών λειτουργιών της εφαρμογής. Η ανάλυσή τους έδειξε ότι πράγματι τα δεδομένα - μηνύματα μεταφέρονται μόνο με την κρυπτογραφημένη μορφή τους. Ενδεικτικά στην Εικόνα 5.5 εμφανίζεται το πακέτο που αποστέλλεται στην βάση δεδομένων κατά την αποστολή μηνύματος. Η τιμή (value) είναι η κρυπτογραφημένη μορφή του μηνύματος που αποστάλθηκε.



Εικόνα 5.4: Φίλτρο πακέτων δεδομένων για συγκεκριμένη IP και την συλλογή μηνυμάτων.



Εικόνα 5.5: Πακέτο δεδομένων προς την MongoDB για την συλλογή μηνυμάτων.

5.5 Αξιολόγηση της απόδοσης

Στην ενότητα αυτή μελετάται η απόδοση του EncChat, σε σχέση με την αρχική, χωρίς υπηρεσίες ασφαλείας, εφαρμογή kChat. Αντιστοίχως, μελετώνται οι απαιτήσεις σε χωρητικότητα της βάσης δεδομένων του EncChat. Ειδικότερα, με σκοπό την αξιολόγηση της απόδοσης του EncChat μετρήθηκε η βασικότερη παράμετρος που επηρεάζει την απόδοση μιας εφαρμογής, ο χρόνος απόκρισης. Ο χρόνος απόκρισης μετρήθηκε στις βασικές λειτουργίες του EncChat, οι οποίες συνίστανται στις εξής: σύνδεση σε δωμάτιο (Join Room), αποστολή μηνύματος (Send Message), πρόσκληση χρήστη (Invite User) και αναζήτηση μηνύματος (Search).

Ειδικά για τις λειτουργίες της σύνδεσης σε δωμάτιο και της αποστολής μηνύματος, οι οποίες είναι διαθέσιμες και στο kChat, έγιναν αντίστοιχες μετρήσεις του χρόνου απόκρισης και στο kChat, ώστε να είναι δυνατή μια σύγκριση μεταξύ των δύο εκδόσεων, τουλάχιστον στις συγκεκριμένες λειτουργίες.

Οι μετρήσεις του χρόνου απόκρισης έγιναν με την εφαρμογή LoadRunner. Το LoadRunner αποτελείται από ένα σύνολο εργαλείων λογισμικού και χρησιμοποιείται για τη δοκιμή εφαρμογών και την αξιολόγηση της συμπεριφοράς τους.

Για κάθε μία από τις παραπάνω λειτουργίες δημιουργήθηκε και ένα σενάριο (script), με το εργαλείο VUGen, μέσω της εκτέλεσης του οποίου μετρήθηκε ο αντίστοιχος χρόνος απόκρισης. Με το VUGen η δημιουργία των σεναρίων γίνεται καταγράφοντας τις ενέργειες ενός πραγματικού χρήστη, τις οποίες στη συνέχεια προσομοιώνει κατά την εκτέλεση των σεναρίων.

Ο ψευδο-κώδικας των σεναρίων, με την μορφή που αυτός παράχθηκε από το εργαλείο VUGen, όπως και τα αναλυτικά γραφήματα των μετρήσεων του χρόνου απόδοσης στις εκτελέσεις των σεναρίων μαζί με τις λίστες τιμών των παραμέτρων που χρησιμοποιήθηκαν παρατίθενται στις υποενότητες Β.1, Β.3 και Β.2 αντίστοιχα.

Επιπλέον του χρόνου απόκρισης μετρήθηκαν και τα μεγέθη συγκεκριμένων συλλογών των βάσεων δεδομένων των δύο εκδόσεων, ώστε να φανεί η όποια επιβάρυνση σε χωρητικότητα επιφέρεται από την κρυπτογράφηση των δεδομένων. Οι μετρήσεις αυτές έγιναν μέσω της εφαρμογής Robomongo, η οποία χρησιμοποιήθηκε γενικότερα για την

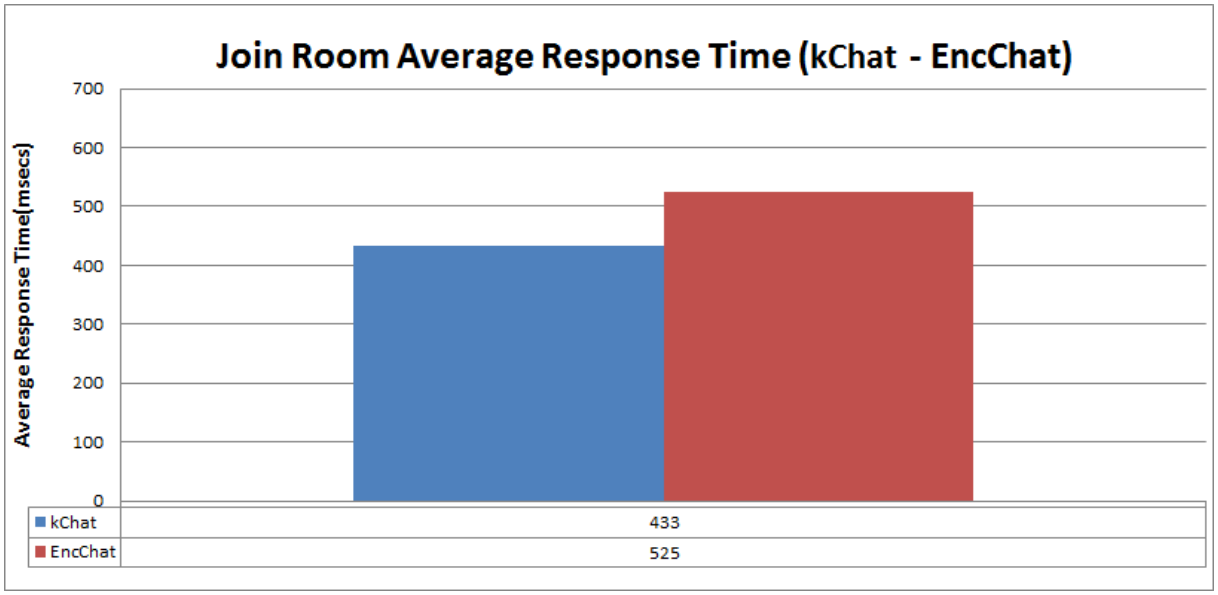
διαχείριση των βάσεων δεδομένων. Στην υπο-ενότητα Β.4 παρατίθενται ενδεικτικές εικόνες από την μέτρηση της συλλογής μηνυμάτων.

5.5.1 Σύνδεση σε δωμάτιο

Για την μέτρηση του χρόνου απόκρισης στη λειτουργία της σύνδεσης σε δωμάτιο δημιουργήθηκε το παρακάτω σενάριο.

Ένας χρήστης με πρόσβαση σε είκοσι διαφορετικά δωμάτια, αρχικά συνδέεται στην εφαρμογή και στη συνέχεια εκτελεί συνδέσεις κυκλικά σε κάθε ένα από αυτά. Οι συνολικές συνδέσεις που εκτελούνται είναι εκατό (100). Τέλος ο χρήστης αποσυνδέεται από την εφαρμογή.

Το σενάριο εκτελέστηκε και στις δύο εφαρμογές kChat και EncChat, με ακριβώς τις ίδιες τιμές παραμέτρων, και η μέση τιμή των χρόνων απόκρισης απεικονίζεται στην Εικόνα 5.6. Τα αρχικά δεδομένα που χρησιμοποιήθηκαν για την εκτέλεση του σεναρίου ήταν εκατό (100) χρήστες, εκατό (100) δωμάτια, είκοσι (20) μηνύματα ανά δωμάτιο, και με πρόσβαση του κάθε χρήστη κατά μέσο όρο σε τρία (3) δωμάτια.



Εικόνα 5.6: Μέσος χρόνος απόκρισης στην λειτουργία σύνδεσης σε δωμάτιο (kChat – EncChat).

Η πολύ μικρή επιβάρυνση που παρατηρείται στον χρόνο απόκρισης του EncChat οφείλεται στην αναζήτηση και μεταφορά των κλειδιών του δωματίου από τον εξυπηρετητή στο πρόγραμμα πλοήγησης του χρήστη καθώς και στην αποκρυπτογράφηση των μηνυμάτων του δωματίου. Η απόλυτη τιμή της είναι 92 msecs και αντιστοιχεί σε

ποσοστό επιβάρυνσης της τάξης του 21%. Η επιβάρυνση αυτή στην πράξη είναι τόσο μικρή που ουσιαστικά δεν γίνεται αντιληπτή από τον τελικό χρήστη. Εδώ πρέπει να τονίσουμε ότι στο EncChat οι γράφοι πρόσβασης και πιστοποίησης έχουν βάθος δύο (2) και είναι πολύ απλοί. Έτσι, σε μια εφαρμογή που θα απαιτούσε περισσότερο πολύπλοκες σχέσεις μεταξύ των principals, η αντίστοιχη μέτρηση ενδέχεται να διαφέρει.

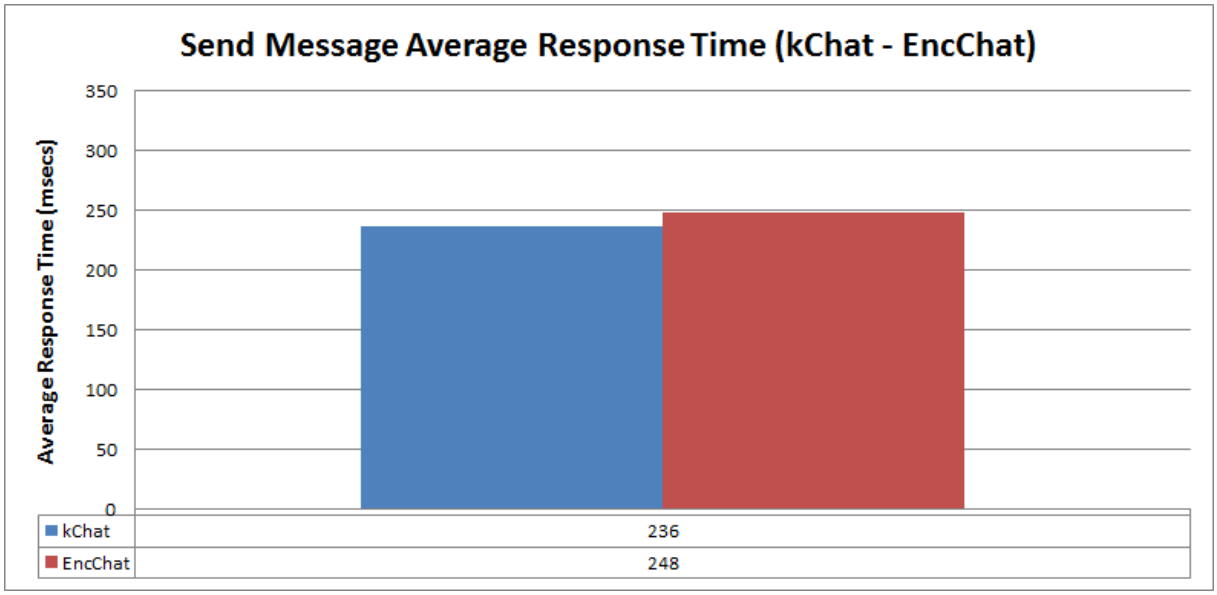
5.5.2 Αποστολή μηνύματος

Για την μέτρηση του χρόνου απόκρισης στη λειτουργία της αποστολής μηνύματος δημιουργήθηκε το παρακάτω σενάριο.

Ένας χρήστης, με πρόσβαση τουλάχιστο σε ένα δωμάτιο, αρχικά συνδέεται στην εφαρμογή και σε ένα δωμάτιο, και στη συνέχεια εκτελεί διαδοχικές αποστολές μηνυμάτων. Οι συνολικές αποστολές μηνυμάτων που εκτελούνται είναι εκατό (100). Τέλος ο χρήστης αποσυνδέεται από το δωμάτιο και την εφαρμογή.

Τα μηνύματα που χρησιμοποιήθηκαν στο σενάριο αυτό έχουν διαφορετικά μεγέθη, 5 έως 15 λέξεων. Το μέγεθος αυτό είναι αντιπροσωπευτικό για εφαρμογές αυτού του είδους. Και στις δύο εκτελέσεις (kChat και EncChat) του σεναρίου χρησιμοποιήθηκαν οι ίδιες τιμές παραμέτρων, δηλαδή ακριβώς τα ίδια μηνύματα.

Το σενάριο εκτελέστηκε και στις δύο εφαρμογές kChat και EncChat και η μέση τιμή των χρόνων απόκρισης απεικονίζεται στην Εικόνα 5.7.



Εικόνα 5.7: Μέσος χρόνος απόκρισης στην λειτουργία αποστολής μηνύματος (kChat – EncChat).

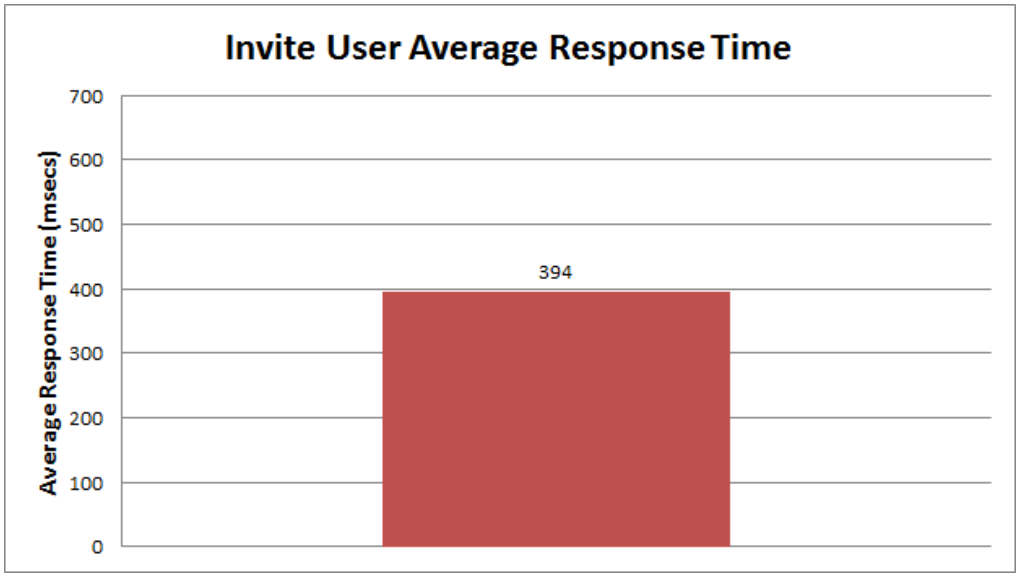
Η επιβάρυνση που παρατηρείται στον χρόνο απόκρισης του EncChat είναι ασήμαντη και δεν γίνεται αντιληπτή από τον τελικό χρήστη. Η λειτουργία αυτή επιβαρύνεται μόνο από την κρυπτογράφηση των δεδομένων η οποία είναι προφανώς ταχύτατη. Η απόλυτη τιμή της επιβάρυνσης είναι 12 msecs και αντιστοιχεί σε ποσοστιαία επιβάρυνση της τάξης του 5%.

5.5.3 Πρόσκληση χρήστη

Για την μέτρηση του χρόνου απόκρισης στη λειτουργία της πρόσκλησης χρήστη δημιουργήθηκε το παρακάτω σενάριο.

ένας (1) χρήστης, με πρόσβαση τουλάχιστο σε ένα δωμάτιο, αρχικά συνδέεται στην εφαρμογή και σε ένα δωμάτιο, και στη συνέχεια εκτελεί διαδοχικές προσκλήσεις χρηστών. Οι συνολικές προσκλήσεις χρηστών που εκτελούνται είναι εκατό (100). Τέλος ο χρήστης αποσυνδέεται από το δωμάτιο και την εφαρμογή. Τα αρχικά δεδομένα που χρησιμοποιήθηκαν για την εκτέλεση του σεναρίου ήταν εκατόν ένας (101) χρήστες, εκατό (100) δωμάτια, και με πρόσβαση του κάθε χρήστη κατά μέσο όρο σε τρία (3) δωμάτια.

Το σενάριο εκτελέστηκε στην εφαρμογή EncChat και η μέση τιμή του χρόνου απόκρισης απεικονίζεται στην Εικόνα 5.8.



Εικόνα 5.8: Μέσος χρόνος απόκρισης στην λειτουργία πρόσκλησης χρήστη (EncChat).

Κατά την διαδικασία αυτή η εφαρμογή πρέπει να αναζητήσει την principal του χρήστη στον γράφο πιστοποίησης και στη συνέχεια να την προσθέσει στον γράφο πρόσβασης. Οι ενέργειες αυτές, αν και είναι υπολογιστικά κοστοβόρες δεν δημιουργούν ουσιαστικό πρόβλημα στην αίσθηση του τελικού χρήστη. Όπως αναφέρθηκε και στο σενάριο σύνδεσης σε δωμάτιο, στο EncChat οι γράφοι πρόσβασης και πιστοποίησης είναι πολύ απλοί. Έτσι, σε μια εφαρμογή με περισσότερο πολύπλοκες σχέσεις μεταξύ των principals, η αντίστοιχη μέτρηση ενδέχεται να διαφέρει.

5.5.4 Αναζήτηση μηνύματος

Η μέτρηση του χρόνου απόκρισης της αναζήτησης έγινε λαμβάνοντας υπόψη δύο βασικές παραμέτρους που την επηρεάζουν: η πρώτη αφορά το πλήθος των μηνυμάτων και η δεύτερη την διασπορά τους σε διαφορετικά δωμάτια.

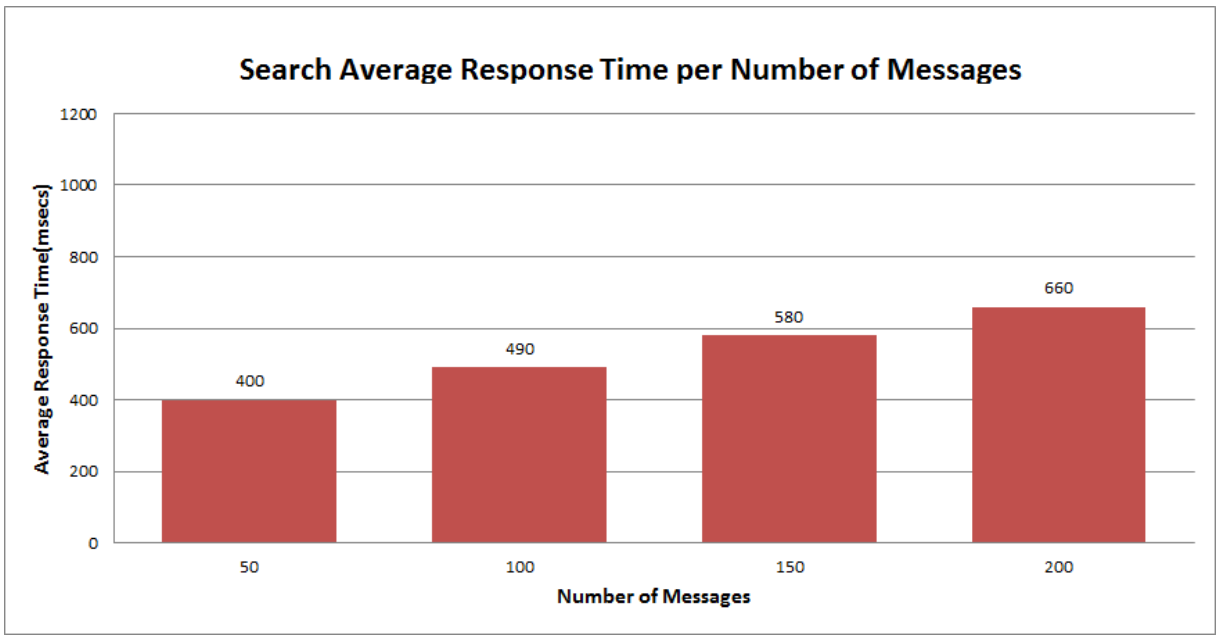
Αναζήτηση (παράμετρος το πλήθος των μηνυμάτων)

Με σκοπό να μετρηθεί η επίδραση αυτής της παραμέτρου δημιουργήθηκε το παρακάτω σενάριο, το οποίο εκτελέστηκε τέσσερις φορές από έναν χρήστη με πρόσβαση σε ένα δωμάτιο, του οποίου το πλήθος των μηνυμάτων ήταν μεταβλητό (50-100-150-200) σε κάθε εκτέλεση.

Ένας χρήστης, με πρόσβαση σε ένα δωμάτιο, αρχικά συνδέεται στην εφαρμογή, και στη συνέχεια εκτελεί διαδοχικές αναζητήσεις λέξεων. Οι συνολικές αναζητήσεις λέξεων που εκτελούνται είναι εκατό (100). Τέλος ο χρήστης αποσυνδέεται από την εφαρμογή.

Τα δεδομένα που χρησιμοποιήθηκαν, οι λέξεις αναζήτησης και τα μηνύματα (όχι το πλήθος τους) στα οποία έγινε η αναζήτηση, ήταν ακριβώς τα ίδια σε όλες τις εκτελέσεις του σεναρίου. Επιπλέον, όλα τα μηνύματα ανήκαν στο ίδιο δωμάτιο. Αυτό έγινε ώστε η μόνη παράμετρος που να διαφέρει στις εκτελέσεις των σεναρίων να είναι το πλήθος των μηνυμάτων.

Το γράφημα με τους μέσους χρόνους απόκρισης των εκτελέσεων του σεναρίου απεικονίζεται στην Εικόνα 5.9.



Εικόνα 5.9: Μέσος χρόνος απόκρισης ανά πλήθος μηνυμάτων στην λειτουργία αναζήτησης (EncChat).

Τα αποτελέσματα των δοκιμών δείχνουν μια σχεδόν γραμμική επιβάρυνση στον χρόνο απόκρισης, περίπου 90msecs ανά 50 μηνύματα. Η επιβάρυνση αυτή δεν επηρεάζει ουσιαστικά την λειτουργία της αναζήτησης και την αίσθηση του τελικού χρήστη.

Αναζήτηση (παράμετρος το πλήθος των δωματίων)

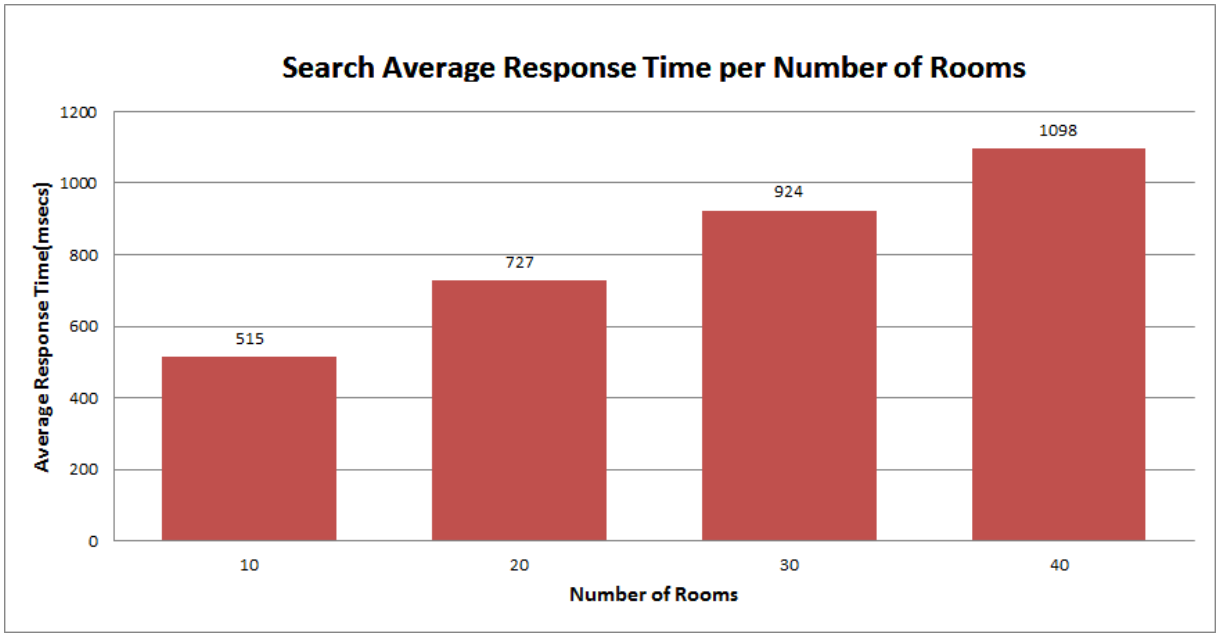
Η αναζήτηση στο EncChat, όπως και σε κάθε εφαρμογή που ενσωματώνει το Mylar, εκτελείται σε δεδομένα – μηνύματα τα οποία ανήκουν σε διαφορετικά δωμάτια επικοινωνίας και άρα έχουν κρυπτογραφηθεί με διαφορετικά κλειδιά. Έτσι, όταν ένας χρήστης αναζητά μια λέξη, η εφαρμογή μετατρέπει το λεκτικό αναζήτησης σε τόσα ισοδύναμα λεκτικά αναζήτησης όσα και τα δωμάτια στα οποία έχει πρόσβαση, και στη συνέχεια εκτελεί αναζητήσεις μέσω αυτών.

Άρα, το πλήθος των δωματίων στα οποία ο χρήστης έχει πρόσβαση είναι μία παράμετρος που επηρεάζει την απόδοση της αναζήτησης.

Με σκοπό να μετρηθεί η επίδραση αυτής της παραμέτρου χρησιμοποιήθηκε το προηγούμενο σενάριο – που δημιουργήθηκε για την αναζήτηση σε μεταβλητό πλήθος μηνυμάτων – το οποίο εκτελέστηκε από τέσσερις διαφορετικούς χρήστες, με τον κάθε ένα να έχει πρόσβαση σε διαφορετικό πλήθος δωματίων (10-20-30-40).

Τα δεδομένα που χρησιμοποιήθηκαν, οι λέξεις αναζήτησης, το πλήθος και τα μηνύματα στα οποία έγινε η αναζήτηση, ήταν ακριβώς τα ίδια σε όλες τις εκτελέσεις του σεναρίου. Αυτό έγινε ώστε η μόνη παράμετρος που να διαφέρει στις εκτελέσεις των σεναρίων να είναι το πλήθος των δωματίων.

Το γράφημα με τους μέσους χρόνους απόκρισης των εκτελέσεων του σεναρίου απεικονίζεται στην Εικόνα 5.10.



Εικόνα 5.10: Μέσος χρόνος απόκρισης ανά δωματίων στην λειτουργία αναζήτησης (EncChat).

Τα αποτελέσματα των δοκιμών δείχνουν μια σχεδόν γραμμική επιβάρυνση στον χρόνο απόκρισης, περίπου 200msecs ανά 10 δωμάτια. Αν και η επιβάρυνση αυτή, δεν είναι αμελητέα, η λειτουργία της αναζήτησης παραμένει σε αποδεκτά πλαίσια με μια σχετικά μικρή επίπτωση στην αίσθηση του τελικού χρήστη.

5.5.5 Χωρητικότητα

Με την ενσωμάτωση του Mylar σε μια εφαρμογή δημιουργούνται τέσσερις συλλογές στη βάση δεδομένων, οι οποίες είναι απαραίτητες για την αποθήκευση των δεδομένων που αφορούν την λειτουργία του.

Αυτές είναι οι εξής:

1. **certs:** Αποθηκεύονται τα πιστοποιητικά των principals - ουσιαστικά περιέχει όλη την πληροφορία του γράφου πιστοποίησης.

2. **princs:** Αποθηκεύονται στοιχεία των principals, όπως το όνομα και τα δημόσια κλειδιά.

3. **princtype:** Αποθηκεύονται οι τύποι των principals.

4. **wrapped_keys:** Αποθηκεύονται όλα τα «εμφωλιασμένα» κλειδιά και οι διαφορές (deltas) που δημιουργούνται από τις σχέσεις πρόσβασης, ουσιαστικά περιέχει όλη την πληροφορία του γράφου πρόσβασης.

Στο EncChat, για 100 χρήστες, 100 δωμάτια, και με πρόσβαση του κάθε χρήστη κατά μέσο όρο σε 3 δωμάτια, τα στοιχεία των συλλογών παρουσιάζονται στον Πίνακα 5.11.

Συλλογή	**Πλήθος εγγραφών**	**Μέγεθος (KB)**	**Μέσο μέγεθος εγγραφής (KB)**
Certs	200	134	0.67
Princs	200	64	0.32
Princtype	2	0.12	0.06
wrapped_keys	300	504	1.68

Πίνακας 5.11: Στοιχεία συλλογών της βάσης δεδομένων του EncChat, τα οποία αφορούν την λειτουργία του Mylar. Οι μετρήσεις έγιναν με την χρήση της εφαρμογής Robomongo.

Το μέσο μέγεθος εγγραφής σε κάθε μια από τις συλλογές πρέπει να θεωρηθεί σταθερό για όλες τις εφαρμογές που ενσωματώνουν το Mylar, μιας και εξαρτάται μόνο από τις κρυπτογραφικές μεθόδους που χρησιμοποιούνται από το Mylar.

Είναι φανερό ότι στο EncChat η επιβάρυνση σε χωρητικότητα από τις παραπάνω συλλογές είναι ασήμαντη. Επιπλέον, φαίνεται ότι ακόμα και σε εφαρμογές με χιλιάδες principals και με εξαιρετικά περισσότερο σύνθετες σχέσεις μεταξύ τους, της τάξης του εκατομμυρίου, η επιβάρυνση θα είναι σε απολύτως φυσιολογικά όρια.

Στη συνέχεια θα εξετάσουμε την επιβάρυνση που επιφέρεται στην χωρητικότητα από την κρυπτογράφηση των δεδομένων – μηνυμάτων. Για τον σκοπό αυτό μετρήθηκαν τα στοιχεία των αντίστοιχων συλλογών στο kChat και στο EncChat τα οποία παρουσιάζονται στον Πίνακα 5.12. Τα μηνύματα είχαν μεταβλητό μέγεθος, 5 έως 15 λέξεων και ήταν ακριβώς τα ίδια και στις δύο εφαρμογές.

	kChat		**EncChat**	
Πλήθος μηνυμάτων	**Μέγεθος (KB)**	**Μέσο μέγεθος εγγραφής (KB)**	**Μέγεθος (KB)**	**Μέσο μέγεθος εγγραφής (KB)**
500	99	0.20	395	0.79
1000	198	0.20	794	0.79
1500	301	0.20	1192	0.79
2000	404	0.20	1591	0.79

Πίνακας 5.12: Στοιχεία της συλλογής μηνυμάτων στις βάσεις δεδομένων των kChat και EncChat, ανά πλήθος μηνυμάτων. Οι μετρήσεις έγιναν με την χρήση της εφαρμογής Robomongo.

Παρατηρούμε ότι το μέγεθος της συλλογής στο EncChat είναι τετραπλάσιο του μεγέθους στο kChat. Αυτή είναι μια πολύ σημαντική επιβάρυνση και πρέπει να λαμβάνεται υπόψη πριν την ενσωμάτωση μια εφαρμογής στο Mylar. Η επιβάρυνση οφείλεται αποκλειστικά στη κρυπτογράφηση των δεδομένων και δεν επηρεάζεται από το πλήθος των principals και τις μεταξύ τους σχέσεις.

Κεφάλαιο 6
Επίλογος

6.1 Σύνοψη

Στην παρούσα διατριβή μελετήθηκε μια νέα τεχνολογική προσέγγιση στο πλαίσιο ανάπτυξης εφαρμογών ιστού. Αυτή η νέα πρόταση, που ακούει στο όνομα Mylar, ενσωματώνει τεχνικές οι οποίες επιτελούν κρυπτογράφηση από-άκρο-σε-άκρο (end-to-end encryption) χωρίς να στερούν από τις εφαρμογές που θα τις χρησιμοποιήσουν τις βασικές λειτουργικότητες της κοινής χρήσης των δεδομένων και της αναζήτησης. Συνεπώς, η αξιοποίηση του Mylar φαίνεται ότι επιλύει ταυτόχρονα πολλά προβλήματα ασφάλειας τα οποία αδυνατούν να αντιμετωπίσουν οι κλασικές κρυπτογραφικές τεχνικές. Ειδικότερα, με την πλατφόρμα του Mylar διασφαλίζονται ότι τα τα δεδομένα τηρούνται με ασφάλεια σε έναν εξυπηρετητή ιστού (web server) κατά τρόπο τέτοιο ώστε ούτε ο ίδιος ο εξυπηρετητής να μπορεί να τα αποκρυπτογραφήσει και, άρα, να τα διαβάσει, χωρίς ωστόσο να περιορίζονται οι λειτουργίες που μπορούν να επιτελέσουν οι χρήστες της αντίστοιχης εφαρμογής ιστού (web application) επί των δεδομένων αυτών. Ως μελέτη περίπτωσης για την ανάλυση του Mylar επελέγη μία εφαρμογή συνομιλίας (chat application), η kChat.

Ειδικότερα, πραγματοποιήθηκε αρχικά μια επισκόπηση των κλασικών μεθόδων που ακολουθούνται για την ασφάλεια των δεδομένων και των προβλημάτων που παρουσιάζουν, αναδεικνύοντας έτσι την ανάγκη για κρυπτογράφηση από-άκρο-σε-άκρο. Ακολούθως έγινε ανάλυση των προβλημάτων που προκύπτουν από την κρυπτογράφηση των δεδομένων, τα οποία αφορούν την κοινή χρήση και την εκτέλεση υπολογισμών και αναζητήσεων επί κρυπτογραφημένων δεδομένων, και παρουσιάστηκαν τα σύγχρονα κρυπτογραφικά συστήματα που προσπαθούν να προσφέρουν λύση σε αυτά.

Στο κύριο μέρος της διατριβής μελετήθηκε η πλατφόρμα Mylar και η πρακτική της εφαρμογή στην εφαρμογή kChat, η οποία οδηγεί στην τροποποιημένη ασφαλή εφαρμογή EncChat. Για την πλατφόρμα Mylar έγινε ενδελεχής ανάλυση των χαρακτηριστικών της, του τρόπου λειτουργίας και ενσωμάτωσής της καθώς και των κρυπτογραφικών τεχνικών που χρησιμοποιεί. Ιδιαίτερη αναφορά έγινε στο κρυπτογραφικό σύστημα MK και στο συγκριτικό πλεονέκτημά του, της διαχείρισης των λεκτικών αναζήτησης, σε σχέση με τα υπόλοιπα κρυπτογραφικά συστήματα αναζήτησης (Searchable Encryption – SE). Στην εφαρμογή EncChat, αρχικά μελετήθηκε ο τρόπος ενσωμάτωσης του Mylar και αναδείχθηκε η ευκολία με την οποία έγινε αυτό. Στη συνέχεια διαπιστώθηκε η κρυπτογράφηση από-άκρο-σε-άκρο των δεδομένων και έγιναν μετρήσεις στην απόδοση των βασικών της λειτουργιών της εφαρμογής. Τα αποτελέσματα των μετρήσεων είναι ενθαρρυντικά και καταδεικνύουν ότι η λειτουργικότητα της εφαρμογής είναι σε πολύ καλό επίπεδο. Ακόμη και στις περιπτώσεις που αυτά συγκρίνονται με τα αντίστοιχα της kChat, η επιβάρυνση που παρατηρείται είναι πολύ μικρή. Τέλος μετρήθηκε η επιβάρυνση σε χωρητικότητα που επιφέρεται από την κρυπτογράφηση των δεδομένων. Η επιβάρυνση αυτή, αν και είναι σοβαρή, είναι προβλέψιμη και δεν επηρεάζει την λειτουργικότητα της εφαρμογής.

6.2 Συμπεράσματα

Το γενικό συμέρασμα που προκύπτει από την παρούσα διατριβή είναι ότι η κρυπτογράφηση από-άκρο-σε-άκρο μπορεί να συμβαδίσει με τις βασικές λειτουργικότητες μιας εφαρμογής ιστού, δηλαδή την κοινή χρήση των δεδομένων και την αναζήτηση σε αυτά. Η πλατφόρμα Mylar είναι ένα μεγάλο βήμα προς την κατεύθυνση αυτή. Συνδυάζει κλασικές και σύγχρονες κρυπτογραφικές τεχνικές

εκμεταλλευόμενη τα αντίστοιχα προτερήματά τους, ενώ παράλληλα η ενσωμάτωσή της σε μια εφαρμογή γίνεται με εύκολο τρόπο. Όλα αυτά προσφέρονται μέσα από ένα δημοφιλές framework όπως το Meteor, δίνοντας έτσι την δυνατότητα να επωφεληθεί ένα μεγάλο πλήθος εφαρμογών.

Θα πρέπει επίσης να σημειωθεί ότι οι τεχνολογικές προσεγγίσεις της κρυπτογράφησης από-άκρο-σε-άκρο, οι οποίες διασφαλίζουν στο μέγιστο βαθμό την εμπιστευτικότητα των δεδομένων, αναμένεται να αποτελέσουν βασικό σχεδιαστικό «πυλώνα» για τα μελλοντικά συστήματα. Και αυτό όχι μόνο γιατί η ανάγκη για ασφάλεια διαρκώς αυξάνεται αλλά και γιατί οι απαιτήσεις για προστασία των δεδομένων ήδη κατά το σχεδιασμό των συστημάτων/εφαρμογών (data protection by design) αποτελεί πλέον νομική υποχρέωση στο πλαίσιο συμμόρφωσης με το θεσμικό πλαίσιο για την προστασία των προσωπικών δεδομένων: συγκεκριμένα, η αρχή της προστασίας των δεδομένων ήδη κατά το σχεδιασμό προβλέπεται στο νέο Γενικό Κανονισμό (ΕΕ) 679/2016 της Ευρωπαϊκής Ένωσης για την προστασία προσωπικών δεδομένων [44] ως μία εκ των υποχρεώσεων όσων πραγματοποιούν επεξεργασία προσωπικών δεδομένων. Προφανώς, η εμπιστευτικότητα από-άκρο-σε-άκρο αναμένεται να αποτελεί, σε ορισμένες περιπτώσεις, μία βασική επιλογή για την πλήρωση της ως άνω υποχρέωσης.

Ένα επόμενο βήμα που θα πρέπει να γίνει αφορά την δοκιμή του Mylar σε περισσότερο σύνθετες εφαρμογές και με πολύ μεγαλύτερο όγκο δεδομένων. Η υλοποίηση του Mylar στο framework Meteor και η ευκολία ενσωμάτωσής του δίνει πρόσφορο έδαφος για ένα τέτοιο εγχείρημα. Εξάλλου, η χρήση του Mylar σε εφαρμογές όπως είναι, για παράδειγμα, οι ιατρικές, μπορεί να είναι η πλέον βέλτιστη λύση για την αντιμετώπιση όλων των σοβαρών θεμάτων ασφάλειας και προστασίας προσωπικών δεδομένων που πηγάζουν από την κρισιμότητα των δεδομένων που υφίστανται επεξεργασία μέσω αυτών των εφαρμογών.

Αναμφίβολα, τεχνολογίες όπως αυτή του Mylar είναι σημαντικό να μελετηθούν ενδελεχώς, τόσο ως προς το είδος των εφαρμογών στις οποίες μπορούν να ενσωματωθούν όσο και ως προς τα ζητήματα ασφαλείας που αντιμετωπίζουν, αλλά και σε εκείνα που ενδεχομένως παραμένουν μη αντιμετωπίσιμα – όπως επισημάνθηκε εξάλλου και στο τέλος του Κεφαλαίου 4. Το σίγουρο είναι ότι οι κλασικοί συμβατικοί αλγόριθμοι κρυπτογράφησης δεν μπορούν να καλύψουν ικανοποιητικώς το σύνολο των απαιτήσεων ασφαλείας και, ως εκ τούτου, οι πιο προηγμένες κρυπτογραφικές τεχνικές

– όπως αυτές στις οποίες στηρίζεται το Mylar – αναμένεται να κερδίζουν συνεχώς έδαφος και να μας απασχολήσουν ερευνητικά τα προσεχή χρόνια.

Βιβλιογραφία

[01] Acar, A., Aksu, H., Uluagac, A.S. and Conti, M., 2017. A Survey on Homomorphic Encryption Schemes: Theory and Implementation. arXiv preprint arXiv:1704.03578.

[02] Adler, J.M., Dai, W., Green, R.L. and Neff, C.A., 2000, December. Computational details of the votehere homomorphic election system. In Proc. Ann. Intl Conf. Theory and Application of Cryptology and Information Security (ASIACRYPT).

[03] Agrawal, R., Kiernan, J., Srikant, R. and Xu, Y., 2004, June. Order preserving encryption for numeric data. In Proceedings of the 2004 ACM SIGMOD international conference on Management of data (pp. 563-574). ACM.

[04] Bajaj, S. and Sion, R., 2014. Trusteddb: A trusted hardware-based database with privacy and data confidentiality. IEEE Transactions on Knowledge and Data Engineering, 26(3), pp.752-765.

[05] BigQuery - Analytics Data Warehouse | Google Cloud Platform. Google. Available at: https://cloud.google.com/bigquery/ [Accessed November 26, 2017].

[06] Cash, D., Grubbs, P., Perry, J. and Ristenpart, T., 2015, October. Leakage-abuse attacks against searchable encryption. In Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (pp. 668-679). ACM..

[07] Cash, D., Jarecki, S., Jutla, C., Krawczyk, H., Roşu, M.C. and Steiner, M., 2013. Highly-scalable searchable symmetric encryption with support for boolean queries. In Advances in cryptology–CRYPTO 2013 (pp. 353-373). Springer, Berlin, Heidelberg.

[08] Chen, A., GCreep: Google Engineer Stalked Teens, Spied on Chats (Updated). Gawker. Available at: http://gawker.com/5637234/gcreep-google-engineer-stalked-teens-spied-on-chats [Accessed November 23, 2017].

[09] Curtmola, R., Garay, J., Kamara, S. and Ostrovsky, R., 2011. Searchable symmetric encryption: improved definitions and efficient constructions. Journal of Computer Security, 19(5), pp.895-934.

[10] Diffie, W. and Hellman, M., 1976. New directions in cryptography. IEEE transactions on Information Theory, 22(6), pp.644-654.

[11] Encrypted Pastebin - Keep your data private and secure! - Defuse Security. Available at: https://defuse.ca/pastebin.htm [Accessed November 24, 2017].

[12] Gemalto, N.V., Breach-Level-Index-Report-H1-2017-Gemalto. Data Breach Statistics by Year, Industry, More - Breach Level Index. Available at: http://breachlevelindex.com/assets/BLI-ebook-H1-2017/Breach-Level-Index-Report-H1-2017-Gemalto.html [Accessed November 21, 2017].

[13] Gentry, C., 2009, May. Fully homomorphic encryption using ideal lattices. In STOC (Vol. 9, No. 2009, pp. 169-178).

[14] Goh, E.J., 2003. Secure indexes. IACR Cryptology ePrint Archive, 2003, p.216.

[15] Grubbs, P., McPherson, R., Naveed, M., Ristenpart, T. and Shmatikov, V., 2016, October. Breaking web applications built on top of encrypted data. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (pp. 1353-1364). ACM.

[16] Grubbs, P., Sekniqi, K., Bindschaedler, V., Naveed, M. and Ristenpart, T., 2016. Leakage-Abuse Attacks against Order-Revealing Encryption. IACR Cryptology ePrint Archive, 2016, p.895.

[17] Islam, M.S., Kuzu, M. and Kantarcioglu, M., 2012, February. Access Pattern disclosure on Searchable Encryption: Ramification, Attack and Mitigation. In Ndss (Vol. 20, p. 12).

[18] Kamara, S., Papamanthou, C. and Roeder, T., 2012, October. Dynamic searchable symmetric encryption. In Proceedings of the 2012 ACM conference on Computer and communications security (pp. 965-976). ACM.

[19] kChat. Available at: https://github.com/KiqueDev/kChat [Accessed April 17, 2017].

[20] Kerckhoffs, A., 1883. La cryptographie militaire, ou, Des chiffres usités en temps de guerre: avec un nouveau procédé de déchiffrement applicable aux systèmes à double clef. Librairie militaire de L. Baudoin.

[21] LoadRunner. Micro Focus. Available at: https://software.microfocus.com/en-us/products/loadrunner-load-testing/overview [Accessed December 21, 2017].

[22] Meteor, Inc. Meteor: The fastest way to build. Available at: https://www.meteor.com [Accessed November 24, 2017].

[23] Naveed, M., Kamara, S. and Wright, C.V., 2015, October. Inference attacks on property-preserving encrypted databases. In Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (pp. 644-655). ACM.

[24] Nmap. Nmap: the Network Mapper - Free Security Scanner. Available at: https://nmap.org/ [Accessed December 21, 2017].

[25] Oracle VM VirtualBox. Available at: https://www.virtualbox.org/ [Accessed December 21, 2017].

[26] OWASP. OWASP Top Ten Project. Available at: https://www.owasp.org/index.php/Category:OWASP_Top_Ten_Project [Accessed November 20, 2017].

[27] Popa, R.A., Redfield, C., Zeldovich, N. and Balakrishnan, H., 2011, October. CryptDB: protecting confidentiality with encrypted query processing. In Proceedings of the Twenty-Third ACM Symposium on Operating Systems Principles (pp. 85-100). ACM.

[28] Popa, R.A., Stark, E., Helfer, J., Valdez, S., Zeldovich, N., Kaashoek, M.F. and Balakrishnan, H., 2014, April. Building Web Applications on Top of Encrypted Data Using Mylar. In NSDI (pp. 157-172).

[29] Popa, R.A., Stark, E., Helfer, J., Valdez, S., Zeldovich, N., Kaashoek, M.F. and Balakrishnan, H., 2016, November. Response to "Breaking web applications built on top of encrypted data" (CCS 2016) by P. Grubbs, R. McPherson, M. Naveed, T. Ristenpart and V. Shmatikov. Mylar. Available at: https://css.csail.mit.edu/mylar/security.html [Accessed May 24, 2017].

[30] Popa, R.A. and Zeldovich, N., 2013. Multi-Key Searchable Encryption. IACR Cryptology ePrint Archive, 2013, p.508.

[31] PrivateBin. Available at: https://privatebin.info/ [Accessed November 24, 2017].

[32] Rivest, R.L., Shamir, A. and Adleman, L., 1978. A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2), pp.120-126.

[33] Robomongo - native MongoDB management tool (Admin UI). Available at: https://robomongo.org/ [Accessed December 21, 2017].

[34] Shannon, C.E., 1949. Communication theory of secrecy systems. Bell Labs Technical Journal, 28(4), pp.656-715.

[35] Song, D.X., Wagner, D. and Perrig, A., 2000. Practical techniques for searches on encrypted data. In Security and Privacy, 2000. S&P 2000. Proceedings. 2000 IEEE Symposium on (pp. 44-55). IEEE.

[36] Stallings, W., 2006. Cryptography and network security: principles and practices. Pearson Education India.

[37] Tate, R., Why You Shouldn't Trust Facebook with Your Data: An Employee's Revelations. Gawker. Available at: http://gawker.com/5445592/why-you-shouldnt-trust-facebook-with-your-data-an-employees-revelations [Accessed November 23, 2017].

[38] Tenable.io. *Tenable™*. Available at: https://www.tenable.com/products/tenable-io [Accessed December 21, 2017].

[39] Vernam, G.S., 1926. Cipher printing telegraph systems: For secret wire and radio telegraphic communications. Journal of the AIEE, 45(2), pp.109-115.

[40] Weiss, N.E. and Miller, R.S., 2015. The target and other financial data breaches: Frequently asked questions. Congressional Research Service.

[41] Wireshark - Go Deep. Available at: https://www.wireshark.org/ [Accessed December 21, 2017].

[42] WordPress.Com Hacked, Attackers Gain Root Access to Servers. Information Security News, IT Security News & Expert Insights: SecurityWeek.Com. Available at:

http://www.security77week.com/wordpresscom-hacked-attackers-gain-root-access-servers [Accessed November 23, 2017].

[43] ZeroBin - because ignorance is bliss. Available at: http://sebsauvage.net/wiki/doku.php?id=php:zerobin [Accessed December 21, 2017].

[44] Κανονισμός (ΕΕ) αριθ. 679/2016 της 27ης Απριλίου 2016 για την προστασία των φυσικών προσώπων έναντι της επεξεργασίας των δεδομένων προσωπικού χαρακτήρα και για την ελεύθερη κυκλοφορία των δεδομένων αυτών. Available at: http://eur-lex.europa.eu/legal-content/EL/TXT/PDF/?uri=OJ:L:2016:119:FULL&from=EL [Accessed December 21, 2017].

Παράρτημα Α

Διεπαφή εφαρμογών του Mylar

Συνάρτηση	**Περιγραφή**
$idp\_config(url, pubkey)$	Επιστρέφει την principal που αντιστοιχεί στο IDP με δημόσιο κλειδί $pubkey$ και διεύθυνση url.
$create\_user(uname, password, auth\_princ)$	Δημιουργεί τον χρήστη $uname$ με κωδικό $password$, ο οποίος πιστοποιείται από την principal $auth\_princ$.
$login(uname, password)$	Συνδέει τον χρήστη $uname$ στην εφαρμογή.
$logout()$	Αποσυνδέει από την εφαρμογή τον χρήστη που την καλεί.
$collection.encrypted(\{field: princ\_field\}, \dots)$	Καθορίζει ότι το πεδίο $field$ στη συλλογή $collection$ θα κρυπτογραφείται.
$collection.searchable(field)$	Καθορίζει ότι το πεδίο $field$ στη συλλογή $collection$ θα έχει την δυνατότητα αναζήτησης.
$grantee.allow\_search(granter)$	Επιτρέπει στην principal $grantee$ να εκτελεί αναζήτηση στα δεδομένα της principal $granter$.
$collection.search(word, field, princ, filter, proj)$	Εκτελεί αναζήτηση της λέξης $word$ στα δεδομένα του πεδίου $field$, με τη χρήση του

	λεκτικού αναζήτησης που δημιουργείται μέσω της principal $princ$.
$princ\_create(name, creator\_princ)$	Δημιουργεί την principal $name$ η οποία πιστοποιείται από την principal που την δημιουργεί $creator\_princ$. Επιπλέον η $creator\_princ$ αποκτά πρόσβαση στην $name$.
$princ\_create\_static(name, password)$	Δημιουργεί μια στατική principal $name$ και τα αντίστοιχα κλειδιά κρυπτογράφησης. Τα μυστικά κλειδιά είναι κρυπτογραφημένα με κλειδί το $password$.
$princ\_static(name, password)$	Επιστρέφει τα μυστικά κλειδιά της στατικής principal $name$ με κωδικό $password$.
$princ\_current()$	Επιστρέφει την principal του χρήστη που την καλεί.
$princ\_lookup(name1, \ldots, namek, root)$	Επιστρέφει την principal $name1$, για την οποία η αλυσίδα $name1, \ldots, namek, root$ υπάρχει στον γράφο πιστοποίησης. Όπου $root$ κάποια authority principal.
$granter.add\_access(grantee)$	Δίνει πρόσβαση στην principal $grantee$ επί της principal $granter$

Πίνακας Α.1: Ψευδο-κώδικας της διεπαφής εφαρμογών του Mylar.

Παράρτημα Β
Υλικό σεναρίων και αναλυτικά γραφήματα

Η υπο-ενότητα Β.1 περιέχει τον ψευδοκώδικα των σεναρίων που εκτελέστηκαν για τις μετρήσεις του χρόνου απόκρισης στις διάφορες λειτουργίες των εφαρμογών, όπως αυτά παράγονται από το εργαλείο λογισμικού VUGen.

Η υπο-ενότητα Β.2 περιέχει τις λίστες τιμών των παραμέτρων που χρησιμοποιήθηκαν κατά την εκτέλεση των σεναρίων, ενώ η υπο-ενότητα Β.3 περιέχει τα αναλυτικά γραφήματα των μετρήσεων του χρόνου απόκρισης όλων των εκτελέσεων των σεναρίων.

Η υπο-ενότητα Β.4 περιέχει ενδεικτικές εικόνες από το εργαλείο λογισμικού Robomongo με τα στατιστικά των συλλογών μηνυμάτων στις εφαρμογές kChat και EncChat.

Β.1 Ψευδο-κώδικας σεναρίων από το VUGen

Σύνδεση σε δωμάτιο

```
// ***********************************************************************************************************
// ****  PLEASE NOTE: This is a READ-ONLY representation of the actual script. For editing please press the "Develop Script" button. ****
// ***********************************************************************************************************

Action()
{
  truclient_step("1", "Navigate to 'http://debiandev:3000/'", "snapshot=Action_1.inf");
  truclient_step("2", "Click on Sign in button", "snapshot=Action_2.inf");
  truclient_step("3", "Click on Username textbox", "snapshot=Action_3.inf");
  truclient_step("4", "Type TC.getParam('Users') in Username textbox", "snapshot=Action_4.inf");
  truclient_step("5", "Click on Password passwordbox", "snapshot=Action_5.inf");
  truclient_step("6", "Type *** in Password passwordbox", "snapshot=Action_6.inf");
  truclient_step("7", "Click on Chat Now! button", "snapshot=Action_7.inf");
  truclient_step("8", "Click on body", "snapshot=Action_8.inf");
  truclient_step("9", "For ( var i = 0 ; i < 100 ; i++ )", "snapshot=Action_9.inf");
  {
    lr_start_transaction("joinRoom");
    truclient_step("9.1", "Click on Join button", "snapshot=Action_9.1.inf");
    lr_end_transaction("joinRoom",0);
    truclient_step("9.2", "Click on Exit Room button", "snapshot=Action_9.2.inf");
  }
  truclient_step("10", "Click on usrExit", "snapshot=Action_10.inf");
  truclient_step("11", "Click on Log out", "snapshot=Action_11.inf");

  return 0;
}
```

Αποστολή μηνύματος

```
// ***********************************************************************************************************
// ****  PLEASE NOTE: This is a READ-ONLY representation of the actual script. For editing please press the "Develop Script" button.  ****
// ***********************************************************************************************************

Action()
{
  truclient_step("1", "Navigate to 'http://debiandev:3000/'", "snapshot=Action_1.inf");
  truclient_step("2", "Click on Sign in button", "snapshot=Action_2.inf");
  truclient_step("3", "Click on Username textbox", "snapshot=Action_3.inf");
  truclient_step("4", "Type TC.getParam('Users') in Username textbox", "snapshot=Action_4.inf");
  truclient_step("5", "Click on Password passwordbox", "snapshot=Action_5.inf");
  truclient_step("6", "Type *** in Password passwordbox", "snapshot=Action_6.inf");
  truclient_step("7", "Click on Chat Now! button", "snapshot=Action_7.inf");
  truclient_step("8", "Click on body", "snapshot=Action_8.inf");
  truclient_step("9", "Click on Join button", "snapshot=Action_9.inf");
  truclient_step("10", "For ( var i = 0 ; i < 100 ; i++ )", "snapshot=Action_10.inf");
  {
    truclient_step("10.1", "Click on Chatting With textbox", "snapshot=Action_10.1.inf");
    truclient_step("10.2", "Type TC.getParam('Messages') in Chatting With textbox", "snapshot=Action_10.2.inf");
    lr_start_transaction("sendMessage");
    truclient_step("10.3", "Click on Send button", "snapshot=Action_10.3.inf");
    lr_end_transaction("sendMessage",0);
  }
  truclient_step("11", "Click on usrExit", "snapshot=Action_11.inf");
  truclient_step("12", "Click on Log out", "snapshot=Action_12.inf");

  return 0;
}
```

Πρόσκληση χρήστη

```
// ***********************************************************************************************************
// ****  PLEASE NOTE: This is a READ-ONLY representation of the actual script. For editing please press the "Develop Script" button.  ****
// ***********************************************************************************************************

Action()
{
  truclient_step("1", "Navigate to 'http://debiandev:3000/'", "snapshot=Action_1.inf");
  truclient_step("2", "Click on Sign in button", "snapshot=Action_2.inf");
  truclient_step("3", "Click on Username textbox", "snapshot=Action_3.inf");
  truclient_step("4", "Type TC.getParam('Users') in Username textbox", "snapshot=Action_4.inf");
  truclient_step("5", "Click on Password passwordbox", "snapshot=Action_5.inf");
  truclient_step("6", "Type *** in Password passwordbox", "snapshot=Action_6.inf");
  truclient_step("7", "Click on Chat Now! button", "snapshot=Action_7.inf");
  truclient_step("8", "Click on body", "snapshot=Action_8.inf");
  truclient_step("9", "Click on Join button", "snapshot=Action_9.inf");
  truclient_step("10", "For ( var i = 0 ; i < 100 ; i++ )", "snapshot=Action_10.inf");
  {
    truclient_step("10.1", "Click on Invite User textbox", "snapshot=Action_10.1.inf");
    truclient_step("10.2", "Type TC.getParam('Invited_Users') in Invite User textbox", "snapshot=Action_10.2.inf");
    lr_start_transaction("Invite");
    truclient_step("10.3", "Click on Invite button", "snapshot=Action_10.3.inf");
    lr_end_transaction("Invite",0);
  }
  truclient_step("11", "Click on Exit Room button", "snapshot=Action_11.inf");
  truclient_step("12", "Click on usrExit", "snapshot=Action_12.inf");
  truclient_step("13", "Click on Log out", "snapshot=Action_13.inf");

  return 0;
}
```

Αναζήτηση

```
// ***********************************************************************************************************
// ****  PLEASE NOTE: This is a READ-ONLY representation of the actual script. For editing please press the "Develop Script" button.  ****
// ***********************************************************************************************************

Action()
{
  truclient_step("1", "Navigate to 'http://debiandev:3000/'", "snapshot=Action_1.inf");
  truclient_step("2", "Click on Sign in button", "snapshot=Action_2.inf");
  truclient_step("3", "Click on Username textbox", "snapshot=Action_3.inf");
  truclient_step("4", "Type TC.getParam('Users') in Username textbox", "snapshot=Action_4.inf");
  truclient_step("5", "Click on Password passwordbox", "snapshot=Action_5.inf");
  truclient_step("6", "Type *** in Password passwordbox", "snapshot=Action_6.inf");
  truclient_step("7", "Click on Chat Now! button", "snapshot=Action_7.inf");
  truclient_step("8", "Click on body", "snapshot=Action_8.inf");
  truclient_step("9", "For ( var i = 0 ; i < 100 ; i++ )", "snapshot=Action_9.inf");
  {
    truclient_step("9.1", "Click on Search Word textbox", "snapshot=Action_9.1.inf");
    truclient_step("9.2", "Type TC.getParam('Search_Words') in Search Word textbox", "snapshot=Action_9.2.inf");
    lr_start_transaction("Search");
    truclient_step("9.3", "Click on Search in all rooms button", "snapshot=Action_9.3.inf");
    lr_end_transaction("Search",0);
  }
  truclient_step("10", "Click on usrExit", "snapshot=Action_10.inf");
  truclient_step("11", "Click on Log out", "snapshot=Action_11.inf");

  return 0;
}
```

Β.2 Λίστες παραμέτρων σεναρίων

Σύνδεση σε δωμάτιο

	Παράμετροι	
	Users	**Rooms**
Τιμές	usr1	room1 … room20

Πίνακας Α.2: Παράμετροι εκτέλεσης του σεναρίου σύνδεσης σε δωμάτιο.

Αποστολή μηνύματος

	Παράμετροι	
	Users	**Messages**
Τιμές	usr101	How are you today Alice? Alice, please call me as soon as possible. Thanks for asking, I am fine, how are you. Hello everybody, this is my first message in this chat. These are your friends from childhood, through youth. Who goaded you on, demanded more proof.

Πίνακας Α.3: Παράμετροι εκτέλεσης του σεναρίου αποστολής μηνύματος.

Πρόσκληση χρήστη

	Παράμετροι		
	Users	**Rooms**	**Invited_Users**
Τιμές	usr1	room1	usr2 … usr101

Πίνακας Α.4: Παράμετροι της εκτέλεσης του σεναρίου πρόσκλησης χρήστη.

Αναζήτηση

Αναζήτηση (παράμετρος το πλήθος των μηνυμάτων)

	Παράμετροι				
	Users	**Rooms**	**Messages**	**Number of messages**	**Search_Words**
Τιμές	usr1	room1	This is a test message to check the encrypted search.	50	This, is, a, test, message, to, check, the, encrypted, search
	usr1	room1	This is a test message to check the encrypted search.	100	This, is, a, test, message, to, check, the, encrypted, search
	usr1	room1	This is a test message to check the encrypted search.	150	This, is, a, test, message, to, check, the, encrypted, search
	usr1	room1	This is a test message to check the encrypted search.	200	This, is, a, test, message, to, check, the, encrypted, search

Πίνακας Α.5: Παράμετροι των τεσσάρων εκτελέσεων του σεναρίου αναζήτησης με μεταβλητό πλήθος μηνυμάτων.

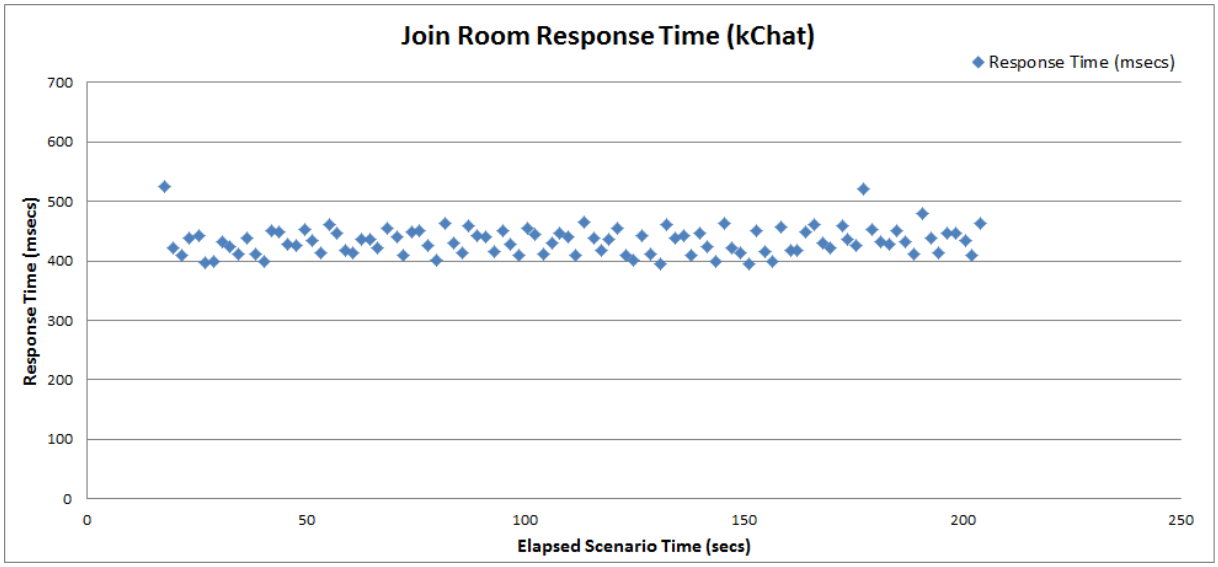
Αναζήτηση (παράμετρος το πλήθος των δωματίων)

	Παράμετροι				
	Users	**Rooms**	**Messages**	**Number of messages**	**Search_Words**
Τιμές	usr1	room1 … room10	This is a test message to check the encrypted search.	120	This, is, a, test, message, to, check, the, encrypted, search
	usr2	room11 … room30	This is a test message to check the encrypted search.	120	This, is, a, test, message, to, check, the, encrypted, search
	usr3	room31 … room60	This is a test message to check the encrypted search.	120	This, is, a, test, message, to, check, the, encrypted, search
	usr4	room61 … room100	This is a test message to check the encrypted search.	120	This, is, a, test, message, to, check, the, encrypted, search

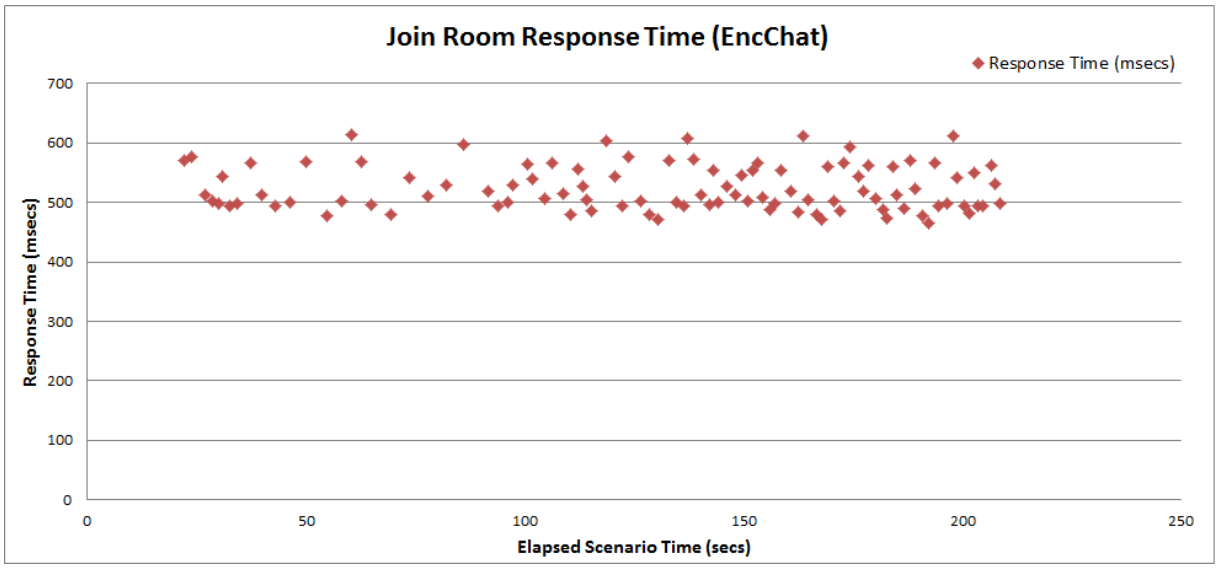
Πίνακας Α.6: Παράμετροι των τεσσάρων εκτελέσεων του σεναρίου αναζήτησης με μεταβλητό πλήθος δωματίων.

Β.3 Αναλυτικά γραφήματα εκτελέσεων

Σύνδεση σε δωμάτιο

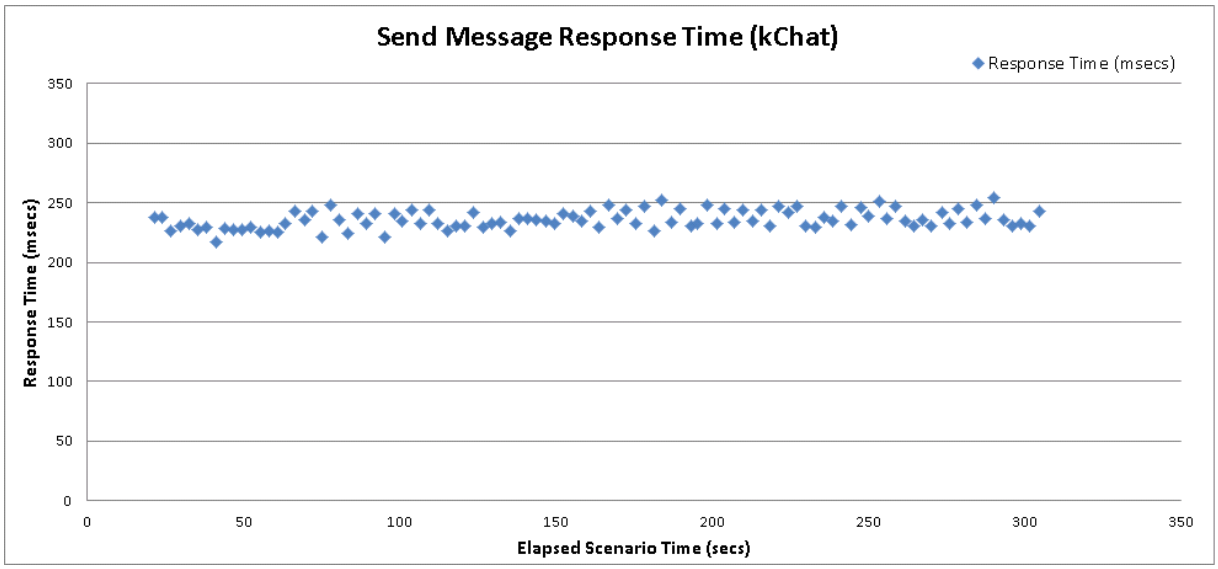


Εικόνα Α.7: Γράφημα χρόνου απόκρισης στο σενάριο σύνδεσης σε δωμάτιο (kChat)

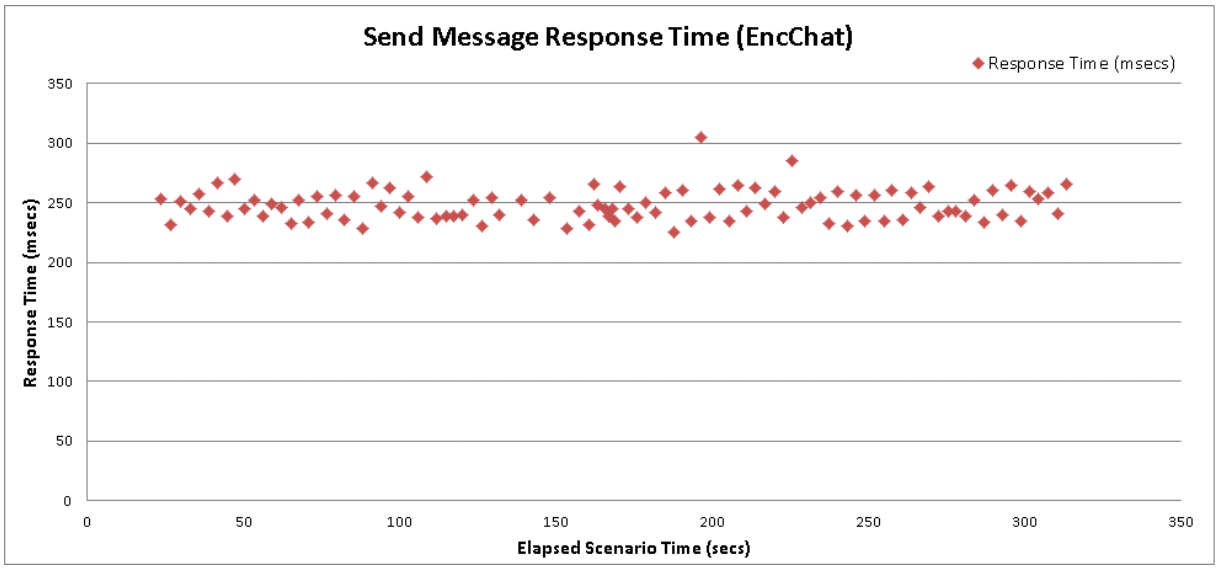


Εικόνα Α.8: Γράφημα χρόνου απόκρισης στο σενάριο σύνδεσης σε δωμάτιο (EncChat)

Αποστολή μηνύματος

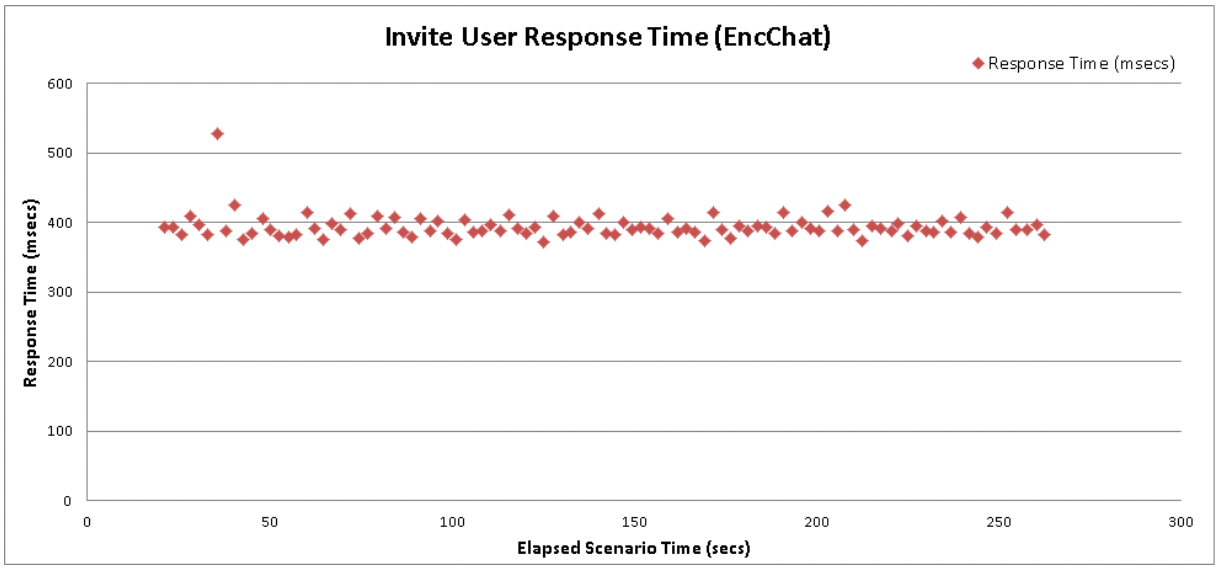


Εικόνα Α.9: Γράφημα χρόνου απόκρισης στο σενάριο αποστολής μηνύματος (kChat)



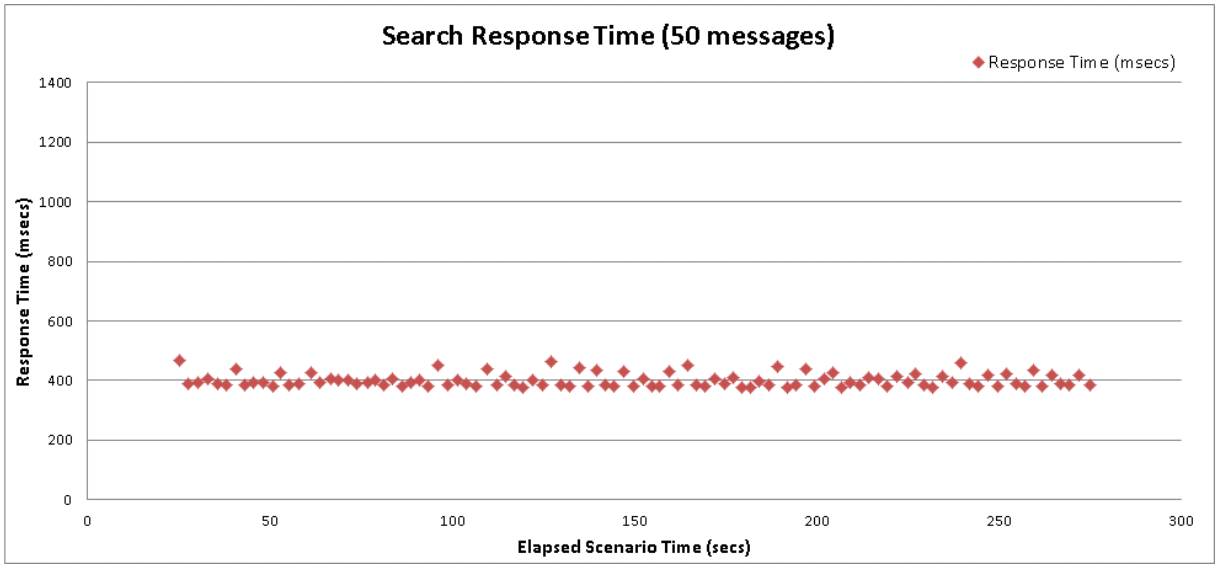
Εικόνα Α.10: Γράφημα χρόνου απόκρισης στο σενάριο αποστολής μηνύματος (EncChat)

Πρόσκληση χρήστη

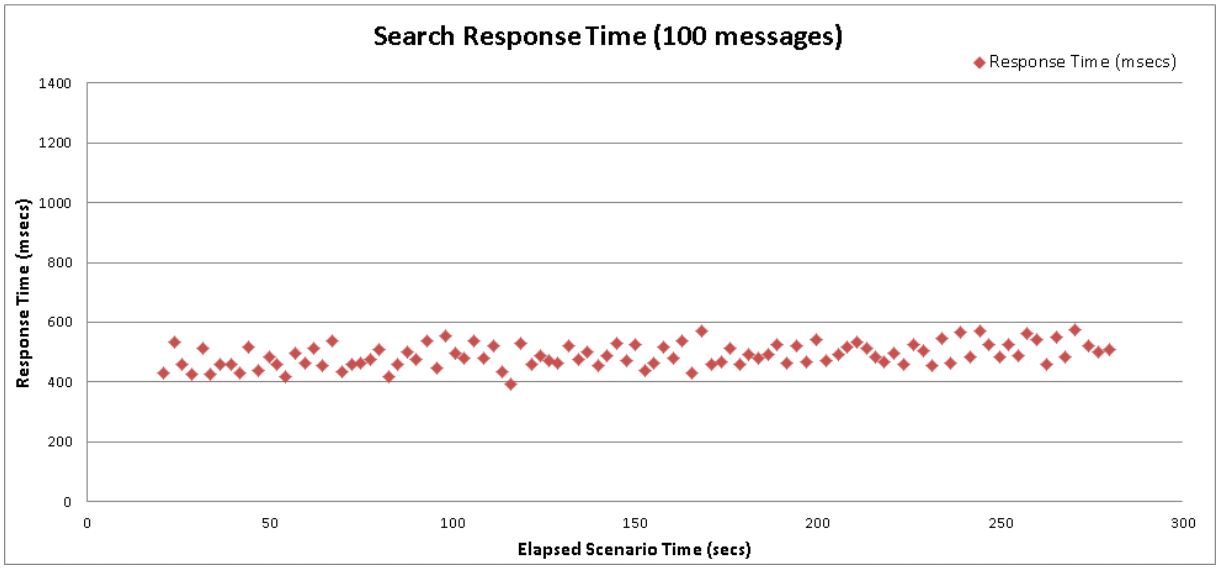


Εικόνα Α.11: Γράφημα χρόνου απόκρισης στο σενάριο πρόσκλησης χρήστη (EncChat)

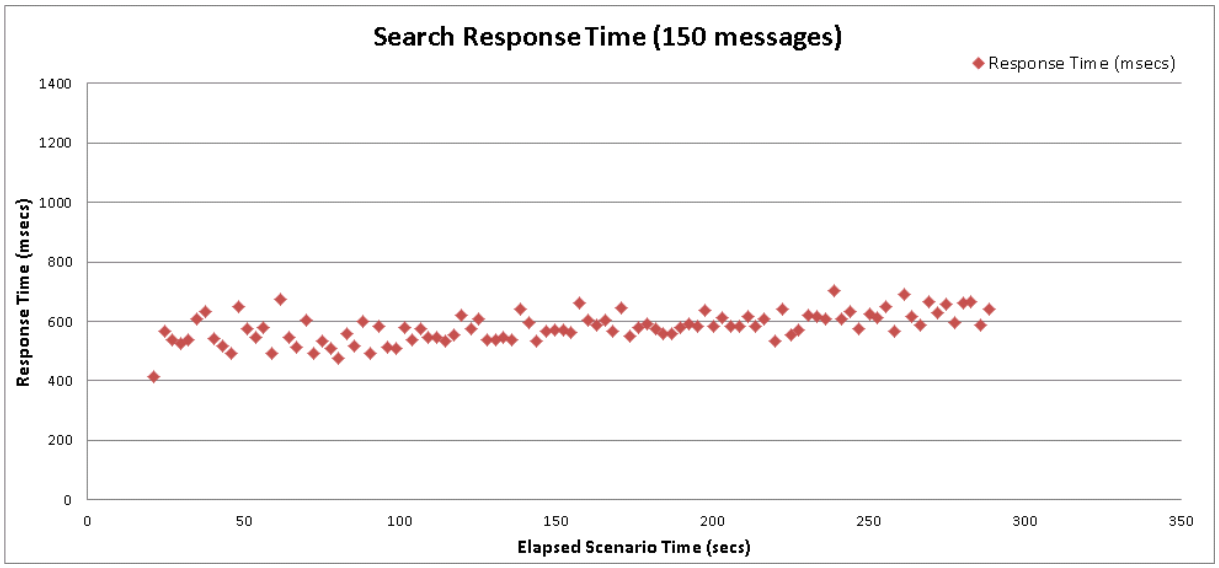
Αναζήτηση



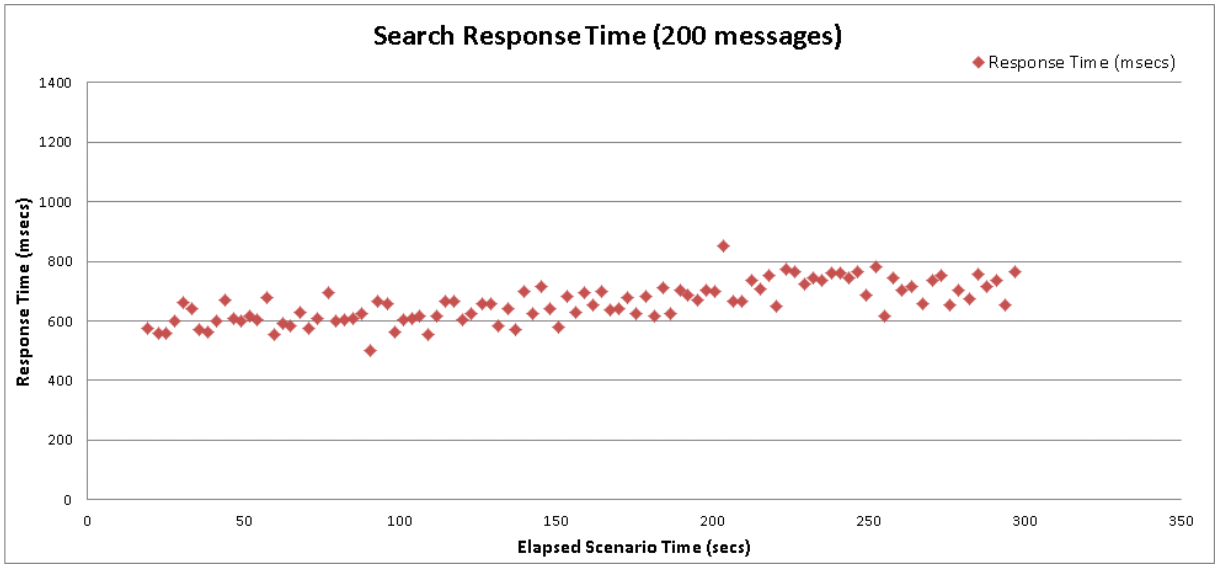
Εικόνα Α.12: Γράφημα χρόνου απόκρισης στο σενάριο αναζήτησης (EncChat) (50 messages)



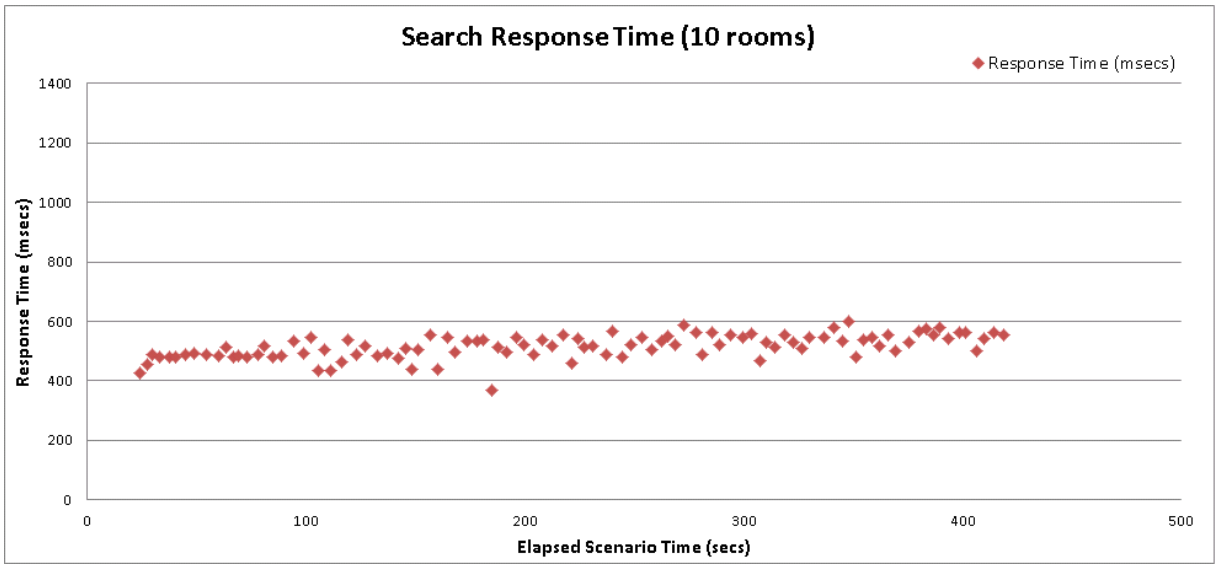
Εικόνα Α.13: Γράφημα χρόνου απόκρισης στο σενάριο αναζήτησης (EncChat) (100 messages)



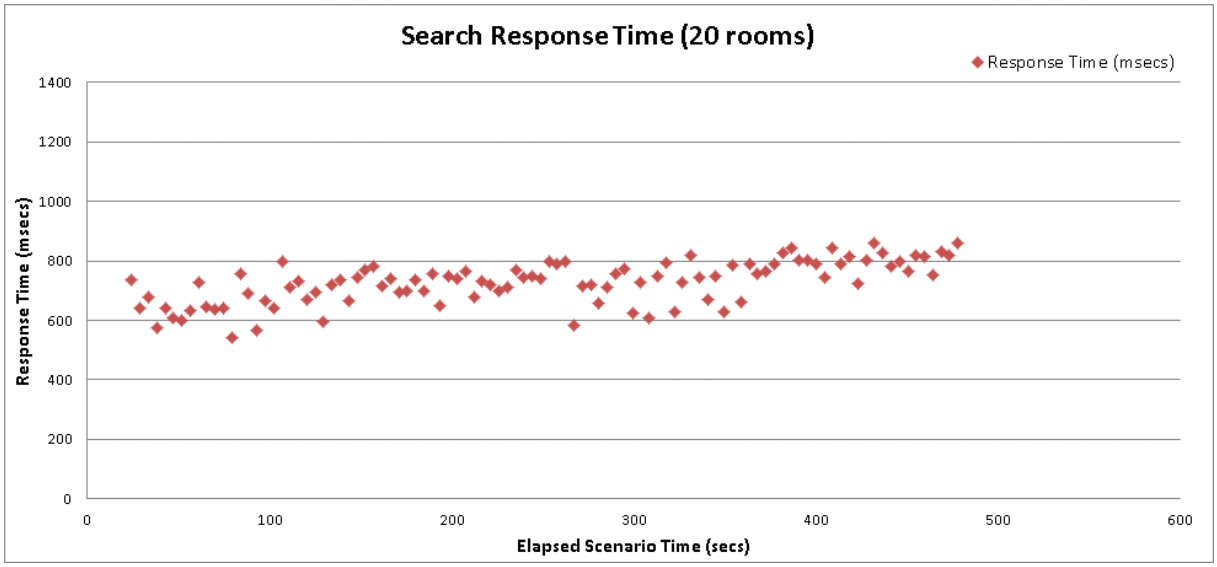
Εικόνα Α.14: Γράφημα χρόνου απόκρισης στο σενάριο αναζήτησης (EncChat) (150 messages)



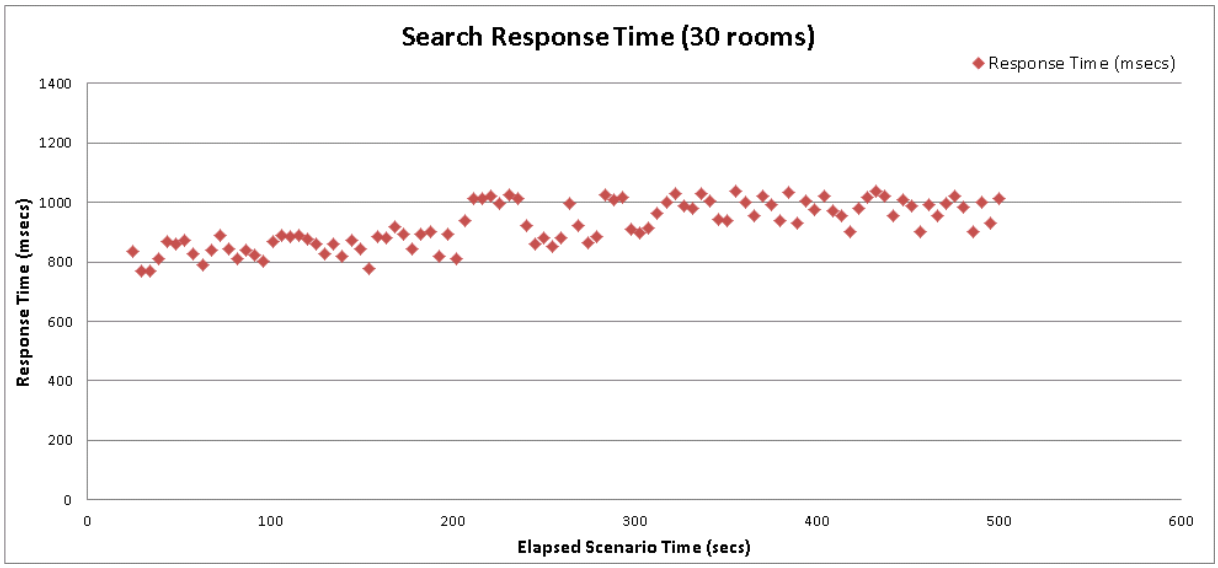
Εικόνα Α.15: Γράφημα χρόνου απόκρισης στο σενάριο αναζήτησης (EncChat) (200 messages)



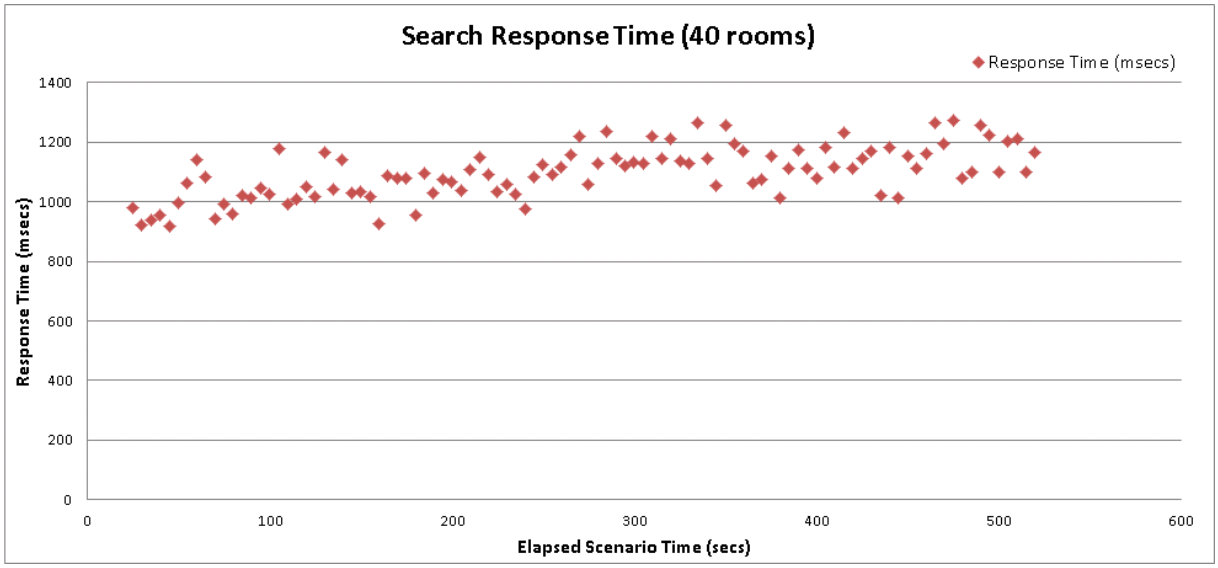
Εικόνα Α.16: Γράφημα χρόνου απόκρισης στο σενάριο αναζήτησης (EncChat) (10 rooms)



Εικόνα Α.17: Γράφημα χρόνου απόκρισης στο σενάριο αναζήτησης (EncChat) (20 rooms)

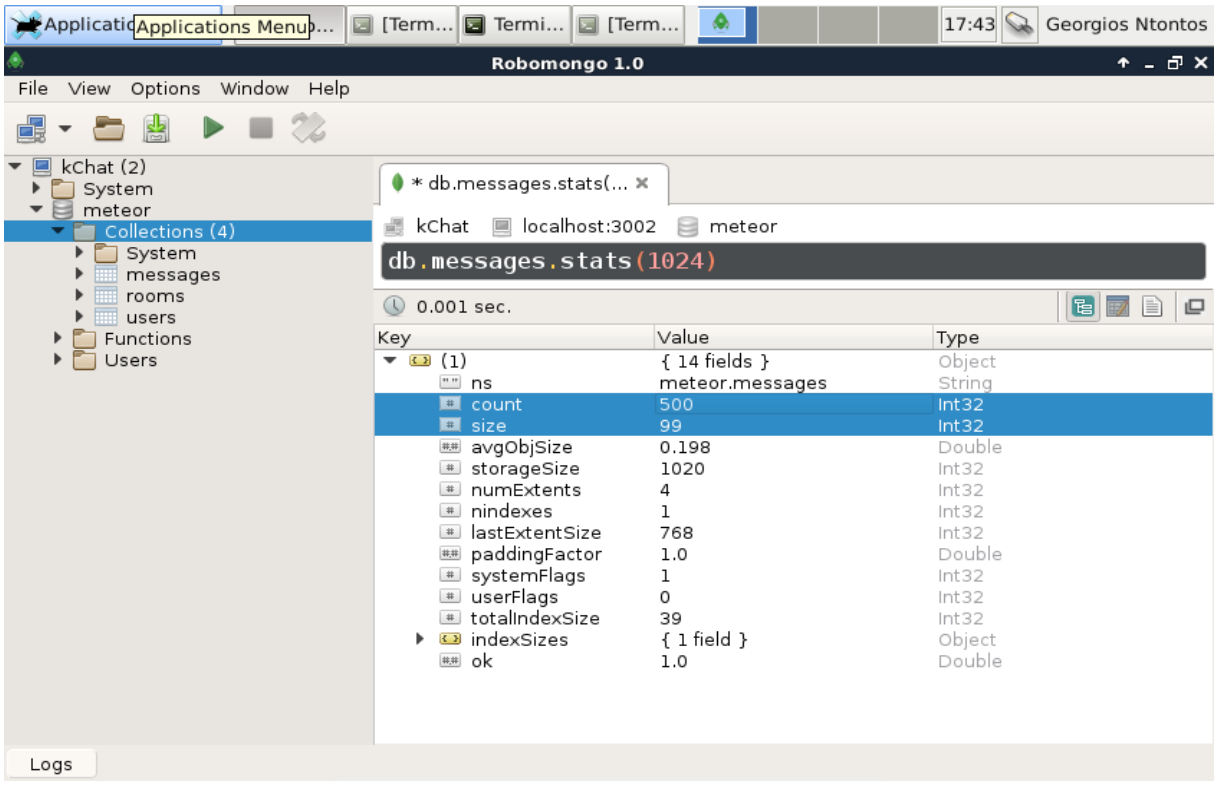


Εικόνα Α.18: Γράφημα χρόνου απόκρισης στο σενάριο αναζήτησης (EncChat) (30 rooms)

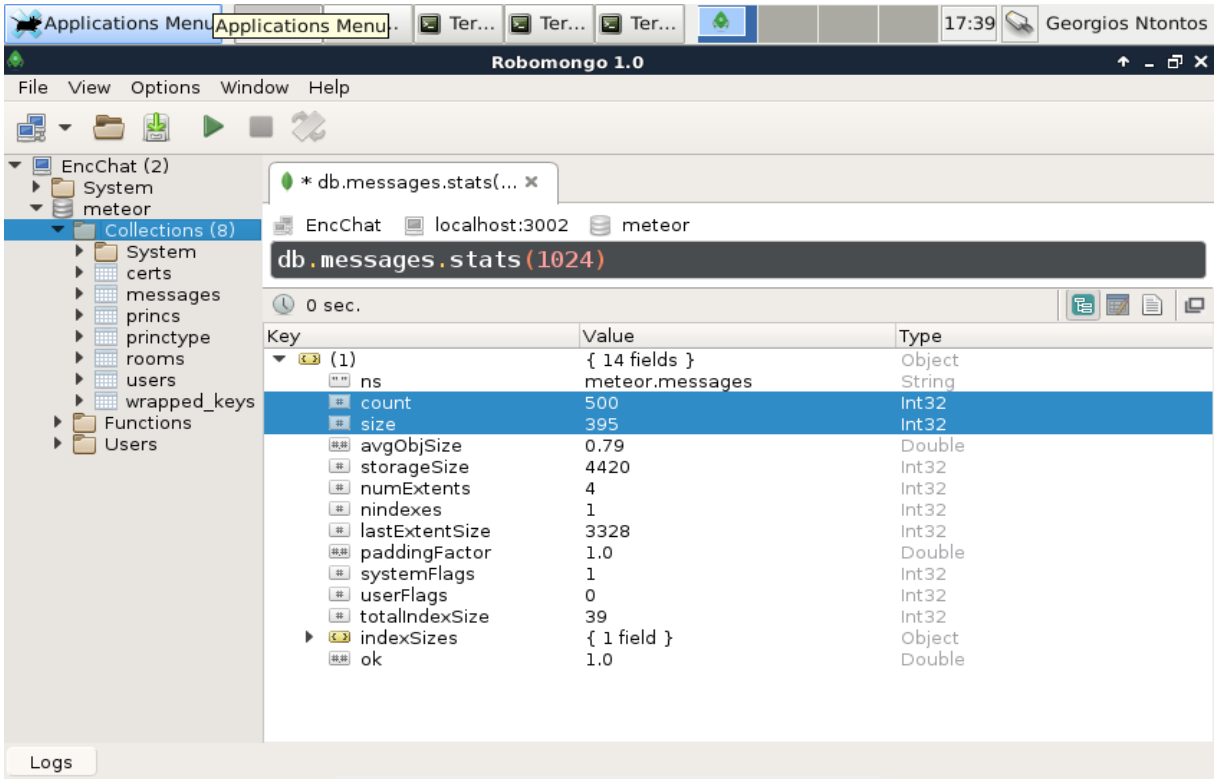


Εικόνα Α.19: Γράφημα χρόνου απόκρισης στο σενάριο αναζήτησης (EncChat) (40 rooms)

B.4 Στοιχεία της συλλογής μηνυμάτων (Robomongo)



Εικόνα Α.20: Στατιστικά της συλλογής μηνυμάτων(messages) στο kChat από το Robomongo.



Εικόνα Α.21: Στατιστικά της συλλογής μηνυμάτων(messages) στο EncChat από το Robomongo.