ΑΝΩΤΑΤΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΠΕΙΡΑΙΑ Τ.Τ.
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
Τμήμα Ηλεκτρονικών Μηχανικών Τ.Ε.

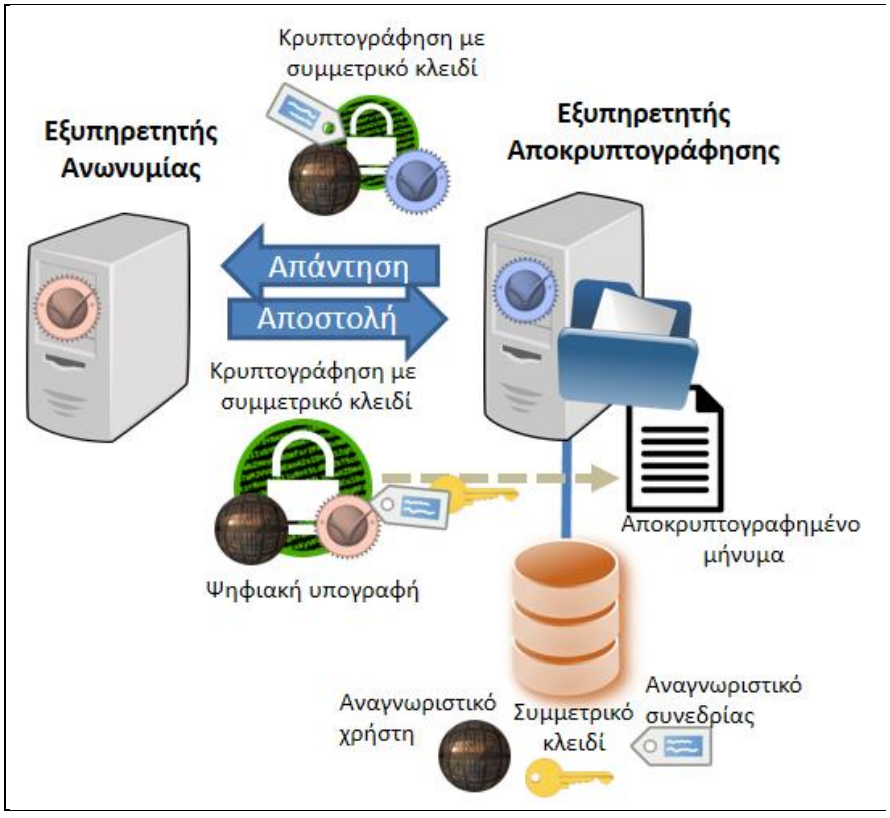
PIRAEUS UNIVERSITY of APPLIED SCIENCES
FACULTY OF ENGINEERING
Department of Electronics Engineering

Πρόγραμμα Μεταπτυχιακών Σπουδών
Διαδικτυωμένα Ηλεκτρονικά Συστήματα

Master of Science in
Internetworked Electronic Systems

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Αρχιτεκτονικές προστασίας ιδιωτικότητας για κινητό και φορετό υπολογισμό



Μεταπτυχιακός Φοιτητής: Χρήστος Χατζηγεωργίου, Α.Μ. 0007
Επιβλέπων: Χαράλαμπος Ζ. Πατρικάκης, Αναπληρωτής Καθηγητής

ΑΙΓΑΛΕΩ, ΣΕΠΤΕΜΒΡΙΟΣ 2017

ΑΝΩΤΑΤΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΠΕΙΡΑΙΑ Τ.Τ.
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
Τμήμα Ηλεκτρονικών Μηχανικών Τ.Ε.

PIRAEUS UNIVERSITY of APPLIED SCIENCES
FACULTY OF ENGINEERING
Department of Electronics Engineering

Πρόγραμμα Μεταπτυχιακών Σπουδών
Διαδικτυωμένα Ηλεκτρονικά Συστήματα

Master of Science in
Internetworked Electronic Systems

MSc Thesis

Privacy preserving architectures for mobile and wearable computing

Student: Christos Chatzigeorgiou, Reg. Nr. 0007
MSc Thesis Supervisor: Charalampos Z. Patrikakis, Associate Professor

ATHENS-EGALEO, SEPTEMBER 2017

ΠΕΡΙΛΗΨΗ

Την τελευταία δεκαετία υπάρχει μία ραγδαία αύξηση στη χρήση κινητών τηλεφώνων και ταμπλετών και πιο πρόσφατα σε φορετές συσκευές όπως καταγραφείς δραστηριότητας και έξυπνα ρολόγια. Η πρόσβαση στο διαδίκτυο είναι σχεδόν πανταχού παρούσα με ταχύτητες που επιτρέπουν ένα αρχείο βίντεο να μοιραστεί μέσα σε λίγα λεπτά από τη στιγμή της εγγραφής του. Εξαιτίας αυτών των τεχνολογιών, ένας τεράστιος όγκος δεδομένων δημιουργείται και ανταλλάσσεται. Αυτά τα δεδομένα συχνά περιέχουν ευαίσθητες πληροφορίες ή πληροφορίες αρκετές για τον εντοπισμό και την αναγνώριση κάποιου ατόμου. Αν και η προστασία αυτών των πληροφοριών από τρίτους είναι ανθρώπινο δικαίωμα, ο κόσμος του κινητού και φορετού υπολογισμού είναι γεμάτος από τρόπους παραβίασης της ιδιωτικότητας.

Αυτή η εργασία έχει στόχο την εξέταση και πρόταση λύσεων για την αντιμετώπιση παραβιάσεων ιδιωτικότητας που μπορούν να συμβούν καθώς οι περισσότεροι άνθρωποι τις φορητές τους συσκευές με συγκεκριμένους τρόπους. Η κρυπτογραφία είναι η βάση των περισσοτέρων αρχιτεκτονικών προστασίας ιδιωτικότητας που χρησιμοποιούνται σήμερα. Γι' αυτό το λόγο, οι πιο συχνά χρησιμοποιούμενοι αλγόριθμοι περιγράφονται στο δεύτερο κεφάλαιο. Επιπλέον, στο τρίτο κεφάλαιο, αναλύονται κάποιες από τις πιο συχνά χρησιμοποιούμενες αρχιτεκτονικές προστασίας ιδιωτικότητας. Στο τελευταίο κεφάλαιο παρουσιάζονται τεχνικές ενίσχυσης της ιδιωτικότητας συνδυάζοντας την κρυπτογραφία και τις αρχιτεκτονικές προστασίας της ιδιωτικότητας. Η εργασία κλείνει με τα συμπεράσματα και προτάσεις για περαιτέρω έρευνα.

ΛΕΞΕΙΣ – ΚΛΕΙΔΙΑ: ασφάλεια κινητών τηλεφώνων, κινητός υπολογισμός, κρυπτογραφία, προστασία ιδιωτικότητας, προσωπικά δεδομένα, φορετές συσκευές

ABSTRACT

In the last decade there is a rapid growth in the use of mobile phones and tablets and very recently a growth in the use of wearable devices such as activity trackers and smart watches. Internet access is almost ubiquitous today, and at speeds that allow a video file to be shared within a few minutes of its capture. Thanks to these technologies, a huge amount of data are generated and exchanged. These data often contain sensitive information or information sufficient so as to track and identify a person. Although protecting this information from third parties is an acknowledged human right, the world of mobile and wearable computing is full of ways to breach privacy.

This project aims to investigate and propose solutions to prevent privacy breaches that may occur due to the ways most people use their mobile devices. Cryptography is the basis of most privacy preserving architectures used today. For this reason, the most widely used algorithms are described in the second chapter of this project. Furthermore, in the third chapter, some of the most often used privacy preserving architectures are analyzed. In the last chapter privacy enhancing techniques are presented combining cryptography and privacy preserving architectures. Conclusions and further research propositions follow.

KEYWORDS: cryptography, mobile computing, mobile security, personal data, privacy protection, wearables

ΕΥΧΑΡΙΣΤΙΕΣ

Μέρος της εργασίας αυτής πραγματοποιήθηκε στα πλαίσια του ευρωπαϊκού ερευνητικού έργου TRILLION, H2020-FCT-2014. Στο σημείο αυτό κρίνεται σκόπιμη η απονομή ευχαριστιών σε όλους όσοι συνέβαλαν στην εκπόνηση αυτής της εργασίας μέσω των ιδεών τους.

Επίσης, θα ήθελα να ευχαριστήσω προσωπικά τον αναπληρωτή καθηγητή Χαράλαμπο Ζ. Πατρικάκη, για τη συνεργασία και την πολύτιμη συμβολή του στην ολοκλήρωση αυτής της εργασίας.

Τέλος θα ήθελα να ευχαριστήσω το Λάζαρο Τουμανίδη για την παροχή συμβουλών σε θέματα κώδικα που χρησιμοποιήθηκε στην εργασία.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΕΙΣΑΓΩΓΗ 10
ΚΕΦΑΛΑΙΟ 1: Εννοιολογικό πλαίσιο 13
1.1 (Ευαίσθητα) Προσωπικά δεδομένα 13
1.2 Ιδιωτικότητα 13
1.3 Κινητές και φορετές συσκευές 14
ΚΕΦΑΛΑΙΟ 2: Κρυπτογραφία – Αλγόριθμοι κρυπτογράφησης 17
2.1 Κλασσικά κρυπτοσυστήματα 18
2.2 Μοντέρνα κρυπτοσυστήματα 18
2.2.1 Κρυπτογραφία συμμετρικού κλειδιού 19
2.2.2 Κρυπτογραφία δημοσίου κλειδιού 24
2.2.3 Υβριδική κρυπτογραφία 26
ΚΕΦΑΛΑΙΟ 3: Αρχιτεκτονικές προστασίας ιδιωτικότητας 29
3.1 Κρυπτογράφηση από άκρο σε άκρο 29
3.2 Εικονικά Ιδιωτικά Δίκτυα 30
3.3 Ανώνυμη περιήγηση στο διαδίκτυο 32
3.4 Δρομολόγηση κρεμμυδιού (Onion routing) 32
3.5 Ανώνυμοι επαναποστολείς ηλεκτρονικής αλληλογραφίας 34
3.4.1 Επαναποστολείς Τύπου I [28] 34
3.4.2 Επαναποστολείς Τύπου II [29] 35
3.4.3 Επαναποστολείς Τύπου III [30] 35
3.4.4 Ψευδώνυμοι επαναποστολείς 35
3.6 Ανώνυμη αποστολή καταγγελιών 36
ΚΕΦΑΛΑΙΟ 4: Εφαρμογές προστασίας ιδιωτικότητας 37
4.1 Προστασία δεδομένων από παραβιάσεις μέσω διαδικτύου 37
4.2 Προστασία δεδομένων από παραβιάσεις των εφαρμογών 43
4.3 Προστασία δεδομένων από φυσικές παραβιάσεις της συσκευής 47
4.4 Προστασία στην επικοινωνία μεταξύ φορετών και φορητών συσκευών 48
ΚΕΦΑΛΑΙΟ 5: Πρόταση και υλοποίηση αρχιτεκτονικής προστασίας ιδιωτικότητας ... 49
5.1 Απαιτήσεις και προδιαγραφές 49
5.2 Αρχιτεκτονική και λειτουργία 49

5.3 Υλοποίηση 53
5.4 Χρήση σε περιβάλλον κινητού και φορετού υπολογισμού 59
5.5 Θέματα χρήσης και συμφωνία με ρυθμιστικό πλαίσιο 61
ΚΕΦΑΛΑΙΟ 6: Συμπεράσματα 62
ΒΙΒΛΙΟΓΡΑΦΙΑ – ΔΙΑΔΙΚΤΥΑΚΕΣ ΠΗΓΕΣ 65
ΠΑΡΑΡΤΗΜΑ 1 69
ΠΑΡΑΡΤΗΜΑ 2 85
ΠΑΡΑΡΤΗΜΑ 3 90

ΠΙΝΑΚΑΣ ΕΙΚΟΝΩΝ

Εικόνα 1.1 Apple iPhone 7 (a), [8] και Google Pixel (b), [9] 15
Εικόνα 1.2 Καταγραφέας καρδιακών παλμών (a), [10] και βηματισμού (b), [11] 16
Εικόνα 1.3 Καταγραφέας δραστηριότητας χειρός (a), [13], Apple Watch (b), [14] και έξυπνο ρολόι με Android (c), [15] 16
Εικόνα 2.1 Μοντέλο κρυπτοσυστήματος συμμετρικού κλειδιού 19
Εικόνα 2.2 Τρόπος λειτουργίας αλγορίθμου DES, [19] 21
Εικόνα 2.3 Λειτουργία (a) κρυπτογράφησης και (b) αποκρυπτογράφησης AES, [19] ... 23
Εικόνα 2.4 Το μοντέλο του κρυπτοσυστήματος δημοσίου κλειδιού 25
Εικόνα 2.5 Το μοντέλο κρυπτογράφησης του PGP, [16] 27
Εικόνα 2.6 Το μοντέλο αποκρυπτογράφησης του PGP, [16] 27
Εικόνα 3.1 Αποστολή δεδομένων με χρήση του Tor, [28] 34
Εικόνα 4.1 Σύγκριση πλοήγησης χωρίς και με χρήση του Tor 42
Εικόνα 5.1, Διάγραμμα ροής αποστολής δεδομένων από το χρήστη στον Εξυπηρετητή Ανωνυμίας 50
Εικόνα 5.2, Διάγραμμα ροής αποστολής δεδομένων μεταξύ των εξυπηρετητών 52
Εικόνα 5.3, Διάγραμμα ροής πρώτου τελικού σημείου ΕΑν 54
Εικόνα 5.4, Διάγραμμα ροής δεύτερου τελικού σημείου ΕΑν 56
Εικόνα 5.5, Διάγραμμα ροής πρώτου τελικού σημείου ΕΑπ 57
Εικόνα 5.6, Διάγραμμα ροής δεύτερου τελικού σημείου ΕΑπ 58
Εικόνα 5.7, Εφαρμογή Android η οποία στέλνει ανώνυμα δεδομένα χρησιμοποιώντας την αρχιτεκτονική που παρουσιάστηκε 60

Data is the pollution problem of the information age, and protecting privacy is the environmental challenge.

- Bruce Schneier, Data and Goliath

ΕΙΣΑΓΩΓΗ

Έχει περάσει μία δεκαετία από τότε που παρουσιάστηκε και κυκλοφόρησε το πρώτο iPhone, ένα έξυπνο κινητό τηλέφωνο όπου έθεσε τον πήχη στο πώς πρέπει να είναι ένα τηλέφωνο αυτής της κατηγορίας. Αυτό ήταν και η αιτία που σήμερα περισσότεροι από τους μισούς ανθρώπους στον πλανήτη έχουν στην κατοχή τους και χρησιμοποιούν κάποιο έξυπνο κινητό τηλέφωνο [1]. Η ιστορία όμως δε σταματάει εκεί. Η Apple, η εταιρία που ανέπτυξε και κυκλοφόρησε το iPhone, καθόρισε ακόμα μία φορά την αγορά των κινητών συσκευών με την κυκλοφορία του iPad το 2010. Πρόκειται για ένα είδος φορητού υπολογιστή χωρίς πληκτρολόγιο (ταμπλέτα). Οι χρήστες των ταμπλετών έχουν ήδη ξεπεράσει το ένα δισεκατομμύριο [2]. Χάρις στις ραγδαίες εξελίξεις στο χώρο των ηλεκτρονικών, που προσφέρουν σήμερα πληθώρα αισθητηρίων στο μέγεθος φακής που καταναλώνουν ελάχιστη ισχύ, ορισμένες φορητές συσκευές έγιναν και φορετές (wearable). Οι συσκευές αυτές φοριούνται ως επί το πλείστον στον καρπό ή στο στήθος, ανάλογα τη χρήση τους. Η λειτουργία μίας τέτοιας συσκευής μπορεί να είναι π.χ. απλώς η μέτρηση των καρδιακών παλμών αυτού που τη φοράει ή, σε πιο εξελιγμένες περιπτώσεις, να είναι ένα έξυπνο κινητό τηλέφωνο στο μέγεθος ενός ρολογιού χειρός, που θα πραγματοποιεί παράλληλα και όλες τις λειτουργίες του κινητού τηλεφώνου.

Με τη μαζική κυκλοφορία των φορητών συσκευών και την εξάπλωση των πρωτοκόλλων και υπηρεσιών 4ης γενιάς (4G), οι πάροχοι υπηρεσιών κινητής τηλεφωνίας έχουν αναβαθμίσει τα δίκτυά τους και έχουν μειώσει σε πολύ χαμηλά επίπεδα το κόστος πρόσβασης στο διαδίκτυο με τη χρήση αυτών των υπηρεσιών. Επίσης, η εγκατάσταση σημείων μέσα στην πόλη για ελεύθερη πρόσβαση στο διαδίκτυο για τους πολίτες από ιδιοκτήτες καταστημάτων ακόμα και από δήμους έχει διευκολύνει την πρόσβαση στο διαδίκτυο σχεδόν από κάθε σημείο του πλανήτη.

Συνέπεια της πολυπληθούς χρήσης των φορητών συσκευών, είτε φορετών είτε όχι, μαζί με την πανταχού παρούσα και γρήγορη πρόσβαση στο διαδίκτυο, είναι η δημιουργία και η διακίνηση τεράστιου όγκου δεδομένων καθημερινά από αυτές. Τα δεδομένα σε αρκετές περιπτώσεις είναι προσωπικά, δηλαδή δεδομένα που περιγράφουν και ταυτοποιούν κάποιο άτομο. Αυτά τα δεδομένα πλέον είναι πολύ εύκολο να συλλεχθούν, να υποστούν επεξεργασία και να ανακτηθούν για επιπλέον χρήσεις. Η διαγραφή τους όμως είναι δύσκολη. Σε έναν κόσμο όπου τα δεδομένα υπήρχαν μόνο σε αναλογική μορφή, ήταν εύκολο να καταστραφούν: αρκούσε να καταστραφεί φυσικά το μέσο που είχαν αποθηκευτεί. Αντίθετα, σήμερα, η διαγραφή των δεδομένων από το ψηφιακό μέσο αποθήκευσής τους δεν συνεπάγεται τη διαγραφή της πληροφορίας που περιέχεται στο μέσον, όπως έχει διαπιστώσει όποιος χρήστης των κοινωνικών δικτύων έχει προσπαθήσει να διαγράφει το προφίλ του σε κάποιο από αυτά. Σε κάποιες περιπτώσεις ούτε καν η καταστροφή του ίδιου του μέσου δεν είναι αρκετή για την καταστροφή του περιεχομένου [3].

Η χρήση φορητών (και φορετών) συσκευών αλλάζει τη ροή και τον όγκο της πληροφορίας με αποτέλεσμα να υπάρχουν νέες ανησυχίες και κίνδυνοι (risks) σχετικά με την ιδιωτικότητα. Στον "εκτός σύνδεσης" κόσμο, οι άνθρωποι έχουν καταφέρει να προστατεύουν τις προσωπικές τους πληροφορίες επιλέγοντας οι ίδιοι τι, πότε και με ποιόν μοιράζονται καθώς και με τη υιοθέτηση φυσικών εμποδίων τα οποία μπορούν να περιορίσουν την πληροφορία όπως για παράδειγμα ένας τοίχος που εμποδίζει τρίτους να ακούσουν μία μυστική ομιλία. Επίσης σημαντικό ρόλο για την προστασία της ιδιωτικότητας έχει παίξει το νομοθετικό πλαίσιο που έχει αναπτυχθεί μέχρι σήμερα σε όλο τον κόσμο, το οποίο όμως δεν έχει εκσυγχρονιστεί αρκετά γρήγορα ώστε να καλύπτει τη σύγχρονη πλήρως ψηφιακή μορφή των πληροφορικών.

Η παρούσα εργασία προσπαθεί να αντιμετωπίσει το θέμα της ιδιωτικότητας σε έναν κόσμο που έχει κατακλυστεί από φορητές συσκευές και εταιρίες που θέλουν να αποκτήσουν πρόσβαση σε προσωπικά δεδομένα των ανθρώπων, δίνοντας λύση μέσω αρχιτεκτονικών που είναι ικανές να προστατεύσουν την ιδιωτικότητα των ανθρώπων. Γίνεται μία προσπάθεια συλλογής των πιο πρόσφατων και συχνά χρησιμοποιούμενων τρόπων και των αρχιτεκτονικών πίσω από αυτούς για την προστασία των προσωπικών δεδομένων ώστε να υπάρχει μία βάση αναφοράς για μελλοντικά έργα.

Στο πρώτο κεφάλαιο ερμηνεύονται κάποιες έννοιες απαραίτητες για την κατανόηση του υπολοίπου μέρους της εργασίας.

Στο δεύτερο κεφάλαιο γίνεται μία ανασκόπηση στην κρυπτογραφία και στους πιο διαδεδομένους και συχνά χρησιμοποιούμενους αλγορίθμους. Η χρήση της κρυπτογραφίας έχει σημαντικό ρόλο στη διατήρηση της ιδιωτικότητας, διότι με αυτή είναι δυνατή η απόκρυψη της πληροφορίας από άτομα για τα οποία δεν προορίζεται αυτή η πληροφορία.

Στο τρίτο κεφάλαιο αναλύονται αρχιτεκτονικές οι οποίες έχουν σαν στόχο τη διατήρηση των προσωπικών δεδομένων κρυφή.

Στο τέταρτο κεφάλαιο εξηγούνται οι τρόποι με τους οποίους μπορούν να εφαρμοστούν οι αρχιτεκτονικές του προηγούμενου κεφαλαίου.

Στα παραπάνω πλαίσια, στο πέμπτο κεφάλαιο παρουσιάζεται μία ολοκληρωμένη αρχιτεκτονική προστασίας ιδιωτικότητας από άκρο σε άκρο, για συσκευές κινητού και φορετού υπολογισμού, μαζί με μία περίπτωση βασισμένη στο μοντέλο του πληθοπορισμού.

Τέλος, στο έκτο κεφάλαιο γίνεται συζήτηση για όλα όσα αναλύθηκαν στα προηγούμενα κεφάλαια.

Επειδή ο χώρος των υπολογιστών και της πληροφορικής είναι γεμάτος από τεχνικούς όρους που συναντώνται στη βιβλιογραφία στα αγγλικά, έχει γίνει μία προσπάθεια μετάφρασης αυτών των όρων στα ελληνικά. Για διευκόλυνση όμως του αναγνώστη σε περίπτωση αναζήτησης περαιτέρω πληροφοριών από τη βιβλιογραφία, κάθε φορά που δίνεται ένας ελληνικός όρος δίνεται ταυτόχρονα και ο αντίστοιχος αγγλικός. Στο Παράρτημα 3 υπάρχει πίνακας με όλους τους όρους συγκεντρωμένους.

Εκτός αν αναφέρεται διαφορετικά, οι εικόνες είναι δημιούργημα του συγγραφέα χρησιμοποιώντας υλικό χωρίς δικαιώματα, διαθέσιμο για χρήση και τροποποίηση.

ΚΕΦΑΛΑΙΟ 1:
Εννοιολογικό πλαίσιο

1.1 (Ευαίσθητα) Προσωπικά δεδομένα

Σύμφωνα με την Αρχή Προστασίας Δεδομένων Προσωπικού Χαρακτήρα [4], κάθε πληροφορία η οποία αναφέρεται και περιγράφει ένα άτομο είναι δεδομένα προσωπικού χαρακτήρα (ή προσωπικά δεδομένα). Η πληροφορία αυτή μπορεί να περιλαμβάνει τα στοιχεία αναγνώρισης του ατόμου (όνομα, επώνυμο, ηλικία, κατοικία, οικογενειακή κατάσταση κλπ.), τα φυσικά χαρακτηριστικά του, την εκπαίδευσή του, την εργασία του, την οικονομική του κατάσταση (έσοδα, περιουσιακά στοιχεία), τα ενδιαφέροντά του, τις δραστηριότητές του και τέλος τις συνήθειές του.

Κάποια όμως από τα προσωπικά δεδομένα είναι ευαίσθητα. Τα ευαίσθητα προσωπικά δεδομένα συμβάλλουν περισσότερο στο σχηματισμό της εικόνας της προσωπικότητας ενός ατόμου. Τα δεδομένα αυτά είναι η φυλετική ή εθνική προέλευση του ατόμου, τα πολιτικά του φρονήματα, οι θρησκευτικές ή φιλοσοφικές του πεποιθήσεις, η συμμετοχή του σε συνδικαλιστική οργάνωση, δεδομένα τα οποία αφορούν την υγεία του, η κοινωνική του πρόνοια, η ερωτική του ζωή, οι ποινικές διώξεις και οι καταδίκες του καθώς και η συμμετοχή του ατόμου σε συναφείς με τα ανωτέρω ενώσεις προσώπων.

1.2 Ιδιωτικότητα

Η ιδιωτικότητα είναι ένα βασικό δικαίωμα, απαραίτητο για την αυτονομία και την προστασία της ανθρώπινης αξιοπρέπειας που χρησιμεύει ως θεμέλιος λίθος πάνω στον οποίο χτίζονται άλλα ανθρώπινα δικαιώματα [5].

Η ιδιωτικότητα στην πληροφορία σημαίνει να έχει ο καθένας έλεγχο σχετικά με το πώς συλλέγονται και χρησιμοποιούνται τα προσωπικά του δεδομένα. Σήμερα, με τις ταχύτητες της δημιουργίας, ανταλλαγής και συλλογής των δεδομένων να είναι ανάλογες της ταχύτητας της εξέλιξης της τεχνολογίας, έχει προκύψει το θέμα της προστασίας των πολιτών και των προσωπικών τους δεδομένων στην παγκόσμια οικονομία της πληροφορίας.

Το πρόβλημα της προστασίας της ιδιωτικότητας έχει αντιμετωπιστεί από διάφορες οπτικές γωνίες και αναπτύχθηκαν κάποιες λύσεις οι οποίες στοχεύουν διάφορες πτυχές της ιδιωτικότητας. Οι λύσεις αυτές αφορούν

- στην ανωνυμία (κατάσταση κατά την οποία κάποιος δεν είναι αναγνωρίσιμος ανάμεσα σε ένα πλήθος υποκειμένων),

- στην αδυναμία σύνδεσης (για έναν παρατηρητή, δύο ενέργειες μέσα σε ένα σύστημα δεν είναι ούτε περισσότερο ούτε λιγότερο συσχετισμένες μετά από παρατήρηση σχετικά με τη γνώση που είχε γι' αυτές πριν την παρατήρηση),
- στη μη παρατηρησιμότητα (μία ενέργεια είναι μη παρατηρήσιμη αν κάνοντας αυτή την ενέργεια δε μπορεί να γίνει διάκριση σε σχέση με το να μην έχει γίνει η ενέργεια) και τέλος
- στην ψευδωνυμία (το ψευδώνυμο είναι ένα αναγνωριστικό ενός υποκειμένου διαφορετικό από την αληθινή του ταυτότητα, π.χ. συγγραφείς, καλλιτέχνες, κλπ.) [6].

1.3 Κινητές και φορετές συσκευές

Με τον όρο κινητός υπολογισμός εννοείται η ικανότητα χρήσης της τεχνολογίας με σκοπό την σύνδεση και χρήση κεντρικά ευρισκόμενης πληροφορίας ή/και λογισμικού εφαρμογής με τη χρήση μικρής, φορητής και ασύρματης συσκευής υπολογισμού και επικοινωνίας . Οι συσκευές αυτές περιλαμβάνουν:

- φορητούς υπολογιστές (laptops),
- ταμπλέτες (tablets),
- έξυπνα τηλέφωνα (smart phones),
- προσωπικούς ψηφιακούς βοηθούς (Personal Digital Assistants - PDAs) και
- συστήματα ψυχαγωγίας αυτοκινήτου (in-car entertainment).

Η χρήση των Προσωπικών Ψηφιακών Βοηθών (PDAs) έχει υποχωρήσει πλέον και τη θέση τους έχουν πάρει τα έξυπνα τηλέφωνα και οι ταμπλέτες που χρησιμοποιούνται καθημερινά κατά κόρον.

Στην αγορά των έξυπνων κινητών τηλεφώνων και των ταμπλετών κυριαρχούν δύο μεγάλες εταιρίες, η Apple και η Google. Η πρώτη είναι υπεύθυνη για το λειτουργικό σύστημα iOS το οποίο χρησιμοποιείται μόνο από συσκευές που σχεδιάζει και παράγει η ίδια, δηλαδή τα iPhones και iPads. Η Google σχεδιάζει και παράγει επίσης συσκευές που κάνουν χρήση του λειτουργικού συστήματος Android αλλά όχι αποκλειστικά: το τελευταίο το χρησιμοποιούν και άλλοι κατασκευαστές έξυπνων κινητών τηλεφώνων και ταμπλετών. Αυτά τα δύο λειτουργικά συστήματα είναι εγκατεστημένα στις 9 από τις 10 συσκευές που λειτουργούν σήμερα [7]. Στην Εικόνα 1.1 διακρίνονται τα τελευταία μοντέλα έξυπνων κινητών τηλεφώνων των δύο παραπάνω εταιριών στα οποία υπάρχει εγκατεστημένη η τελευταία έκδοση του iOS για το iPhone 7 της Apple (a) και του Android για το Pixel της Google (b). Για τα λειτουργικά αυτά συστήματα υπάρχει πλήθος προγραμμάτων λογισμικού ή *εφαρμογών* (*applications ή apps*, όπως συνηθίζεται να λέγονται) ικανών να καλύψουν τις ανάγκες των περισσότερων χρηστών.

Εικόνα 1.1 Apple iPhone 7 (a), [8] και Google Pixel (b), [9]

Οι φορετές συσκευές (wearable devices) είναι μία υποκατηγορία των φορητών (portable devices) ή κινητών συσκευών (mobile devices) και περιλαμβάνει συσκευές που φοριούνται κάτω από, πάνω από ή μαζί με τα ρούχα. Η σμίκρυνση των ηλεκτρονικών εξαρτημάτων και η ικανότητα ενσωμάτωσης τεχνολογιών επικοινωνίας (embedded electronics) έχει αποτέλεσμα οι φορετές συσκευές να είναι πλέον εύκολο και οικονομικά ρεαλιστικό να αποκτηθούν και να χρησιμοποιούνται όλο και συχνότερα. Οι πιο απλές από αυτές τις συσκευές χρησιμοποιούνται για τη μέτρηση καρδιακών παλμών με την πρόσδεση μίας ζώνης χαμηλά στο στήθος ή για τη μέτρηση βηματισμού και απόστασης με την πρόσδεσή τους στα κορδόνια του παπουτσιού όπως αυτές στην Εικόνα 1.2. Οι καταγραφείς δραστηριότητας φοριούνται στον καρπό και μετράνε και καρδιακούς παλμούς αλλά και βηματισμό. Επίσης κάποιες από αυτές διαθέτουν και οθόνη για να δείχνουν την ώρα αλλά και για να ειδοποιούν το χρήστη σε περίπτωση εισερχόμενης κλήσης στο κινητό του τηλέφωνο.

a b

Εικόνα 1.2 Καταγραφέας καρδιακών παλμών (a), [10] και βηματισμού (b), [11]

Τέλος υπάρχουν και τα έξυπνα ρολόγια τα οποία αν και μοιάζουν εξωτερικά με τους καταγραφείς δραστηριότητας όπως φαίνεται και στην Εικόνα 1.3, εσωτερικά τρέχουν ένα πλήρες λειτουργικό σύστημα και μπορούν να πραγματοποιήσουν σχεδόν κάθε λειτουργία που μπορεί να πραγματοποιήσει και ένα έξυπνο κινητό τηλέφωνο. Τα λειτουργικά συστήματα που κυριαρχούσαν στην αγορά το πρώτο τετράμηνο του 2017 ήταν

- το watchOS της Apple με το 57% του μεριδίου αγοράς,
- το Tizen, ένα λειτουργικό σύστημα ανοιχτού κώδικα το οποίο υποστηρίζεται από τη Samsung και την Intel, με το 19% της αγοράς και τέλος
- το Android Wear με το 18% της αγοράς, [12].

a b c

Εικόνα 1.3 Καταγραφέας δραστηριότητας χειρός (a), [13], Apple Watch (b), [14] και έξυπνο ρολόι με Android (c), [15]

Οι κινητές συσκευές που θα απασχολήσουν την παρούσα εργασία είναι τα έξυπνα κινητά τηλέφωνα και οι ταμπλέτες, αφού οι φορητοί υπολογιστές (αν και φορητοί) έχουν περισσότερα κοινά με τους προσωπικούς υπολογιστές όπως και οι φορετές συσκευές.

ΚΕΦΑΛΑΙΟ 2:
Κρυπτογραφία – Αλγόριθμοι κρυπτογράφησης

Η κρυπτογραφία είναι η επιστήμη η οποία με τη χρήση των μαθηματικών επιτρέπει την απόκρυψη και αποκάλυψη δεδομένων με σκοπό την αποθήκευση ευαίσθητης πληροφορίας ή/και τη μεταφορά της μέσω μη ασφαλούς μέσου με τέτοιο τρόπο ώστε να είναι αδύνατον να προσπελασθεί από άλλον εκτός από τον αποδέκτη της. Ο μετασχηματισμός ενός αρχικού μηνύματος σε μία ακατανόητη μορφή (κρυπτοκείμενο) ονομάζεται κρυπτογραφικός αλγόριθμος (ή κρυπταλγόριθμος) και η εφαρμογή αυτού ονομάζεται ***κρυπτογράφηση***. Κλειδί ονομάζεται ένας αριθμός που χρησιμοποιείται (μαζί με το αρχικό μήνυμα) ως είσοδος σε μία συνάρτηση κρυπτογράφησης. Αντίστοιχα, η αντίστροφη διαδικασία (εξαγωγή του αρχικού μηνύματος από το κρυπτογραφημένο κείμενο) ονομάζεται ***αποκρυπτογράφηση***. Το σύνολο των διαδικασιών κρυπτογράφησης και αποκρυπτογράφησης ονομάζεται ***κρυπτοσύστημα***. Η κρυπτογραφία έχει αναδειχθεί σε περιοχή ζωτικής σημασίας για την ασφαλή μετάδοση και αποθήκευση προσωπικών δεδομένων.

Ένα αξιοσημείωτο παράδειγμα αποτελούν οι διαβητικοί, οι οποίοι χρειάζεται να μετράνε καθημερινά τα επίπεδα του σακχάρου στο αίμα τους. Με τη χρήση συσκευών οι οποίες αποθηκεύουν τις μετρήσεις σε μία βάση δεδομένων, είναι δυνατή η ημερήσια παρακολούθηση του ασθενούς από το γιατρό από οποιοδήποτε σημείο ο τελευταίος έχει πρόσβαση στο διαδίκτυο.

Επιπλέον, καθημερινά εκατομμύρια άνθρωποι συνδέονται στον τραπεζικό τους λογαριασμό από το κινητό τους τηλέφωνο, προκειμένου να ενημερωθούν για το υπόλοιπο του λογαριασμού τους ή να πραγματοποιήσουν ηλεκτρονικά κάποια συναλλαγή.

Στα παραπάνω παραδείγματα, είναι ακριβώς η εφαρμογή της κρυπτογραφίας που κάνει δυνατή την ανταλλαγή των πληροφοριών και χωρίς να υπάρχει πιθανότητα κάποιος τρίτος να υποκλέψει τα ευαίσθητα προσωπικά δεδομένα (υγείας, οικονομικά) – τουλάχιστον όχι με ευκολία.

Ένας αλγόριθμος κρυπτογράφησης (ή ***κρυπταλγόριθμος***) μπορεί να είναι είτε δυνατός ή αδύναμος. Η δύναμη ενός αλγορίθμου μετράται από το χρόνο και τους πόρους που χρειάζονται για την ανάκτηση ενός μηνύματος χωρίς τη γνώση του κλειδιού (δηλαδή για την παραβίαση του κρυπτοσυστήματος).

Τα κρυπτοσυστήματα χωρίζονται στα ***κλασσικά*** κρυπτοσυστήματα και τα ***μοντέρνα*** κρυπτοσυστήματα. Στις επόμενες υποενότητες θα αναλυθούν αυτές οι δύο κατηγορίες.

2.1 Κλασσικά κρυπτοσυστήματα

Η κρυπτογραφία είναι τόσο παλιά όσο και η γραφή. Αρχικά βρήκε χρήση στη διασφάλιση της μυστικότητας των επικοινωνιών από κατασκόπους, αρχηγούς στρατευμάτων ή και διπλωμάτες.

Ένας από τους πιο γνωστούς κλασσικούς αλγορίθμους κρυπτογράφησης είναι ο Κώδικας του Καίσαρα. Πήρε το όνομά του από τον Ιούλιο Καίσαρα ο οποίος το χρησιμοποιούσε για την ανταλλαγή μηνυμάτων στρατιωτικού περιεχομένου . Πρόκειται για έναν κώδικα αντικατάστασης, όπου κάθε γράμμα του αρχικού μηνύματος αντικαθίσταται από ένα γράμμα, ορισμένες θέσεις πιο μετά, από το αρχικό στο αλφάβητο. Για παράδειγμα με μία ολίσθηση κατά 3, το γράμμα Δ θα αντικατασταθεί από το γράμμα Α. Παρόλο που δεν είναι γνωστό πόσο αποτελεσματικός ήταν ο κώδικας του Καίσαρα στον καιρό του, στις μέρες μας είναι πολύ εύκολο να παραβιαστεί [16].

Μία βελτίωση του κώδικα του Καίσαρα είναι ο αλγόριθμος κρυπτογράφησης Vigenere. Είναι ένας κώδικας αντικατάστασης αλλά σε αντίθεση με τον κώδικα του Καίσαρα ο οποίος μετατοπίζει κάθε γράμμα σταθερό αριθμό θέσεων, σε αυτόν τον κώδικα υπάρχει ένα κλειδί K αποτελούμενο από αλφαβητικούς χαρακτήρες το οποίο καθορίζει τον αριθμό των θέσεων κάθε γράμματος που θα γίνει η μετατόπιση. Το κάθε γράμμα του μηνύματος, προστίθεται με το γράμμα του κλειδιού και βγαίνει το γράμμα της λέξης του κρυπτοκειμένου. Στην περίπτωση που το κλειδί είναι μικρότερο του μηνύματος, το κλειδί επαναλαμβάνεται. Η γενική αλγεβρική περιγραφή του αλγορίθμου για την κρυπτογράφηση μηνυμάτων με χαρακτήρες που ανήκουν σε ένα σύνολο Σ μπορεί να γραφεί ως εξής:

$$C_i = E_K(M_i) = (M_i + K_{(i \bmod m)}) \bmod l$$

Και για την αποκρυπτογράφηση:

$$M_i = D_K(C_i) = (C_i - K_{(i \bmod m)}) \bmod l$$

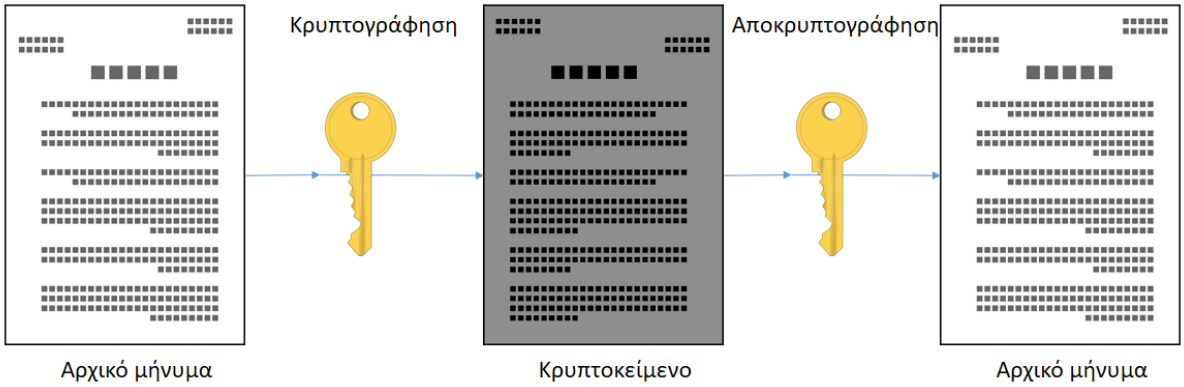
Όπου E_K η κρυπτογράφηση με χρήση του κλειδιού K, D_K η αποκρυπτογράφηση με χρήση του κλειδιού K, M_i ο i-οστός χαρακτήρας του μηνύματος M, C_i ο i-οστός χαρακτήρας του κρυπτοκειμένου C, l το μήκος του συνόλου Σ και m το μήκος του κλειδιού.

2.2 Μοντέρνα κρυπτοσυστήματα

Τα μοντέρνα κρυπτοσυστήματα χωρίζονται σε δύο μεγάλες κατηγορίες. Στα συστήματα συμμετρικού κλειδιού (ή συμμετρική κρυπτογραφία) και στα συστήματα δημοσίου κλειδιού.

2.2.1 Κρυπτογραφία συμμετρικού κλειδιού

Η συμμετρική κρυπτογραφία είναι εκείνη στην οποία η κρυπτογράφηση και η αποκρυπτογράφηση γίνεται με τη χρήση ενός κοινού (μυστικού) κλειδιού το οποίο είναι γνωστό και στον αποστολέα και στον παραλήπτη του μηνύματος. Ήταν ο μόνος τύπος κρυπτογραφίας που υπήρχε μέχρι και τη δεκαετία του 1970. Το μοντέλο του κρυπτοσυστήματος αυτού φαίνεται στην παρακάτω εικόνα.



Εικόνα 2.1 Μοντέλο κρυπτοσυστήματος συμμετρικού κλειδιού

Τα κρυπτοσυστήματα αυτά χωρίζονται σε δύο κατηγορίες: τα κρυπτοσυστήματα τμήματος (Block ciphers) και τα κρυπτοσυστήματα ροής (Stream ciphers) όπου στα πρώτα το μήνυμα χωρίζεται σε κομμάτια και μετά γίνεται η (από)κρυπτογράφηση και στα τελευταία γίνεται (από)κρυπτογράφηση σε κάθε χαρακτήρα του μηνύματος ξεχωριστά.

Αλγόριθμοι τμήματος που χρησιμοποιούνται συχνά είναι οι:

- AES,
- DES (3DES),
- Blowfish

Και συχνά χρησιμοποιούμενοι αλγόριθμοι ροής είναι οι:

- RC4,
- ChaCha

Παρακάτω αναλύονται κάποιοι από τους αλγόριθμους αυτούς.

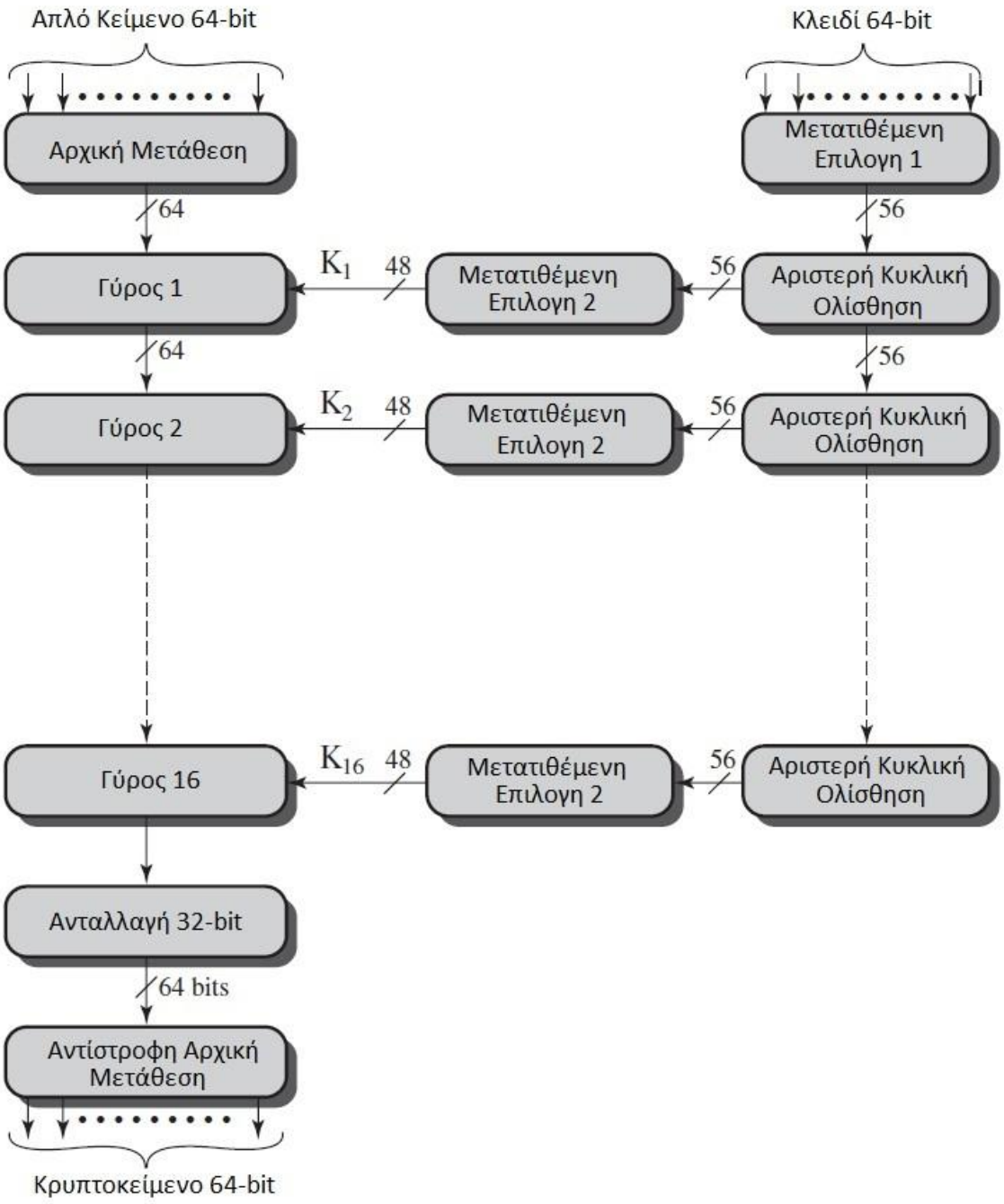
DES

Το Πρότυπο Κρυπτογράφησης Δεδομένων (Data Encryption Standard - DES) αναπτύχθηκε τη δεκαετία του 1970 από μία ομάδα στην IBM. Αιτία της ανάπτυξής του

ήταν η ζήτηση προτάσεων από το Γραφείο Προτύπων των ΗΠΑ για ένα κρυπτοσύστημα το οποίο θα μπορούσε να κρυπτογραφήσει ευαίσθητες πληροφορίες. Ύστερα από την κατάθεση της πρότασης της ομάδας της IBM όπως και την επέμβαση στην ανάπτυξη του αλγορίθμου από το Εθνικό Γραφείο Ασφάλειας των ΗΠΑ (NSA), ο αλγόριθμος αυτός προτυποποιήθηκε μέχρι το 2005 [17] όπου αντικαταστάθηκε διότι έπαψε να θεωρείται ασφαλής από το 1998 [18]. Πλέον χρησιμοποιείται μία πιο ασφαλής εκδοχή του, το τριπλό DES (triple DES - 3DES) όπου μετά την κρυπτογράφηση, το κρυπτοκείμενο κρυπτογραφείται πάλι με διαφορετικό κλειδί και το δεύτερο κρυπτοκείμενο κρυπτογραφείται μία ακόμα φορά με ένα τρίτο κλειδί.

Ο αλγόριθμος DES έπαιξε σημαντικό ρόλο στην προαγωγή της μοντέρνας κρυπτογραφίας, καθώς αποτέλεσε έναυσμα για την ακαδημαϊκή κοινότητα, να μελετήσει την κρυπτογραφία και να ασχοληθεί με θέματα κρυπτανάλυσης.

Όντας αλγόριθμος τμήματος, το μήνυμα χωρίζεται σε τμήματα (ή κομμάτια) των 64 bit στα οποία και εφαρμόζεται ξεχωριστά ο αλγόριθμος DES για την κρυπτογράφηση τους. Το μήκος του κλειδιού πρέπει επίσης να είναι 64 bit αν και στην πραγματικότητα μόνο τα 56 χρησιμοποιούνται. Τα υπόλοιπα 8 είναι bit ισοτιμίας και απορρίπτονται από τον αλγόριθμο.



Εικόνα 2.2 Τρόπος λειτουργίας αλγορίθμου DES, [19]

Η διαδικασία της κρυπτογράφησης γίνεται σε τρία στάδια. Στο πρώτο στάδιο το αρχικό μήνυμα περνάει από μία αρχική μετάθεση (Initial Permuation - IP) όπου τα bit αλλάζουν θέση μέσα στο τμήμα και προκύπτει η μεταβληθείσα είσοδος. Το δεύτερο στάδιο αποτελείται από 16 γύρους της ίδιας συνάρτησης η οποία αποτελείται από λειτουργίες μετάθεσης και αντικατάστασης. Στο τρίτο και τελευταίο στάδιο γίνεται πάλι μετάθεση, αντίστροφη από την αρχική, του οποίου η έξοδος είναι και το κρυπτοκείμενο.

Το κλειδί, αφού υποστεί και αυτό μία μετάθεση, περνάει επίσης από 16 γύρους της ίδιας συνάρτησης (αποτελούμενη από ολίσθηση και μετάθεση) όπου η έξοδος κάθε φορά τροφοδοτεί τον αντίστοιχο γύρο του μηνύματος με 48 bit.

Κατά την αρχική μετάθεση τα bit αλλάζουν θέση βάση ενός προκαθορισμένου πίνακα μετασχηματισμού. Ο αντίστοιχος πίνακας υφίσταται και για την τελική μετάθεση.

Σε κάθε γύρο από τους 16, το μήνυμα επεξεργάζεται σε δύο ίσα τμήματα των 32 bit, το δεξί και το αριστερό. Το δεξί τμήμα, το οποίο λαμβάνει τη θέση της εισόδου του αριστερού τμήματος στον επόμενο γύρο, περνάει από ένα πίνακα επέκτασης και μετάθεσης όπου έχει σαν έξοδο 48 bit. Σε αυτά τα bit γίνεται η πράξη XOR με τα bit του κλειδιού και περνάνε από μία συνάρτηση αντικατάστασης ώστε να γίνουν πάλι 32. Αφού γίνει μετάθεση των 32 αυτών bit, πραγματοποιείται η πράξη XOR με τα bit του αριστερού τμήματος και προκύπτει το δεξί τμήμα του επόμενου γύρου.

Για την αποκρυπτογράφηση, ακολουθείται η ίδια διαδικασία.

AES

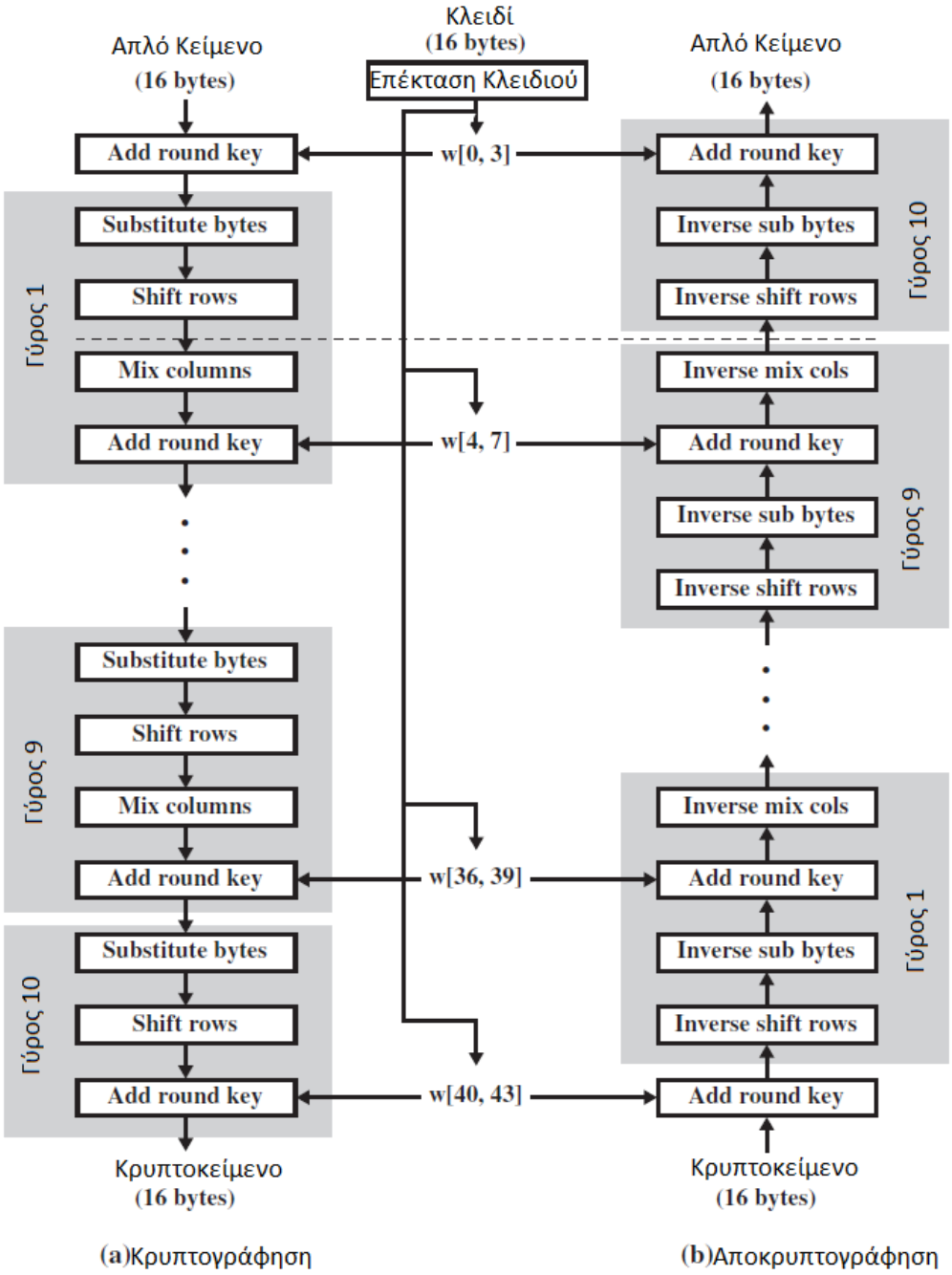
Βασισμένο στην οικογένεια αλγορίθμων Rijndael, το Προχωρημένο Πρότυπο Κρυπτογράφησης (Advanced Encryption Standard - AES) αναπτύχθηκε το 1998 από δύο Βέλγους κρυπτογράφους, τον Vincent Rijmen και τον Joan Daemen. Η ανάπτυξη έγινε στα πλαίσια της αντικατάστασης του DES από το Διεθνές Ινστιτούτο Προτύπων και Τεχνολογίας (National Institute of Standards and Technology - NIST) των ΗΠΑ [20].

Το πρότυπο αυτό χρησιμοποιεί τμήματα των 128 bit αλλά το μέγεθος κλειδιού διαφέρει ανάλογα την περίσταση και μπορεί να είναι 128, 192 ή 256 bit.

Όπως και στο DES, το αρχικό μήνυμα χωρίζεται σε επιμέρους τμήματα όπου κάθε τμήμα κρυπτογραφείται ξεχωριστά.

Βασικό στοιχείο του αλγορίθμου είναι ο πίνακας κατάστασης (state matrix), ένας πίνακας 4 x 4. Σε κάθε θέση αυτού αρχικά τοποθετούνται τα bytes του μηνύματος (ένα byte σε κάθε θέση) ενώ στη συνέχεια δημιουργούνται τα κλειδιά για τον κάθε γύρο, προερχόμενα από το αρχικό μυστικό κλειδί καθώς και ένα ακόμα το οποίο χρησιμοποιείται στο πρώτο στάδιο της κρυπτογράφησης .

Η κρυπτογράφηση περιλαμβάνει τέσσερις μετασχηματισμούς οι οποίοι εφαρμόζονται συγκεκριμένο αριθμό φορών (γύροι). Ο αριθμός των γύρων διαφέρει ανάλογα το μήκος του κλειδιού. Αναφορικά για ένα κλειδί μήκους 128 bits, εκτελούνται 10 γύροι, για κλειδί 192 bits 12 γύροι και τέλος για κλειδί 256 bits 14 γύροι.



Εικόνα 2.3 Λειτουργία (a) κρυπτογράφησης και (b) αποκρυπτογράφησης AES, [19]

Στον πρώτο μετασχηματισμό (SubBytes) γίνεται αντικατάσταση των bytes του πίνακα κατάστασης με βάση το Rijndael S-box. Στο δεύτερο μετασχηματισμό (ShiftRows) κάθε byte στη γραμμή του πίνακα κατάστασης ολισθαίνει αριστερά ένα συγκεκριμένο αριθμό θέσεων. Η πρώτη γραμμή μένει ως έχει ενώ η στη δεύτερη τα bytes ολισθαίνουν κατά μία θέση, στην τρίτη κατά δύο θέσεις και στην τέταρτη κατά 3 θέσεις. Κατά τον τρίτο μετασχηματισμό (MixColumns) τα στοιχεία του πίνακα κατάστασης παίρνουν τιμές οι οποίες είναι βασισμένες σε πολλαπλασιασμό και πρόσθεση του πίνακα κατάστασης, με πίνακα ο οποίος βασίζεται σε πολυώνυμα. Ο τελευταίος μετασχηματισμός

(AddRoundKey) περιλαμβάνει την πράξη XOR μεταξύ του πίνακα κατάστασης από τον και το κλειδί που έχει παραχθεί προηγουμένως .

Πριν την εκτέλεση των γύρων πραγματοποιείται ο μετασχηματισμός AddRoundKey. Ο κάθε γύρος, εξαιρουμένου του τελευταίου, περιλαμβάνει την εκτέλεση των μετασχηματισμών SubBytes, ShiftRows, MixColumns και AddRoundKey. Στον τελευταίο γύρο εκτελούνται οι μετασχηματισμοί SubBytes, ShiftRows και AddRoundKey.

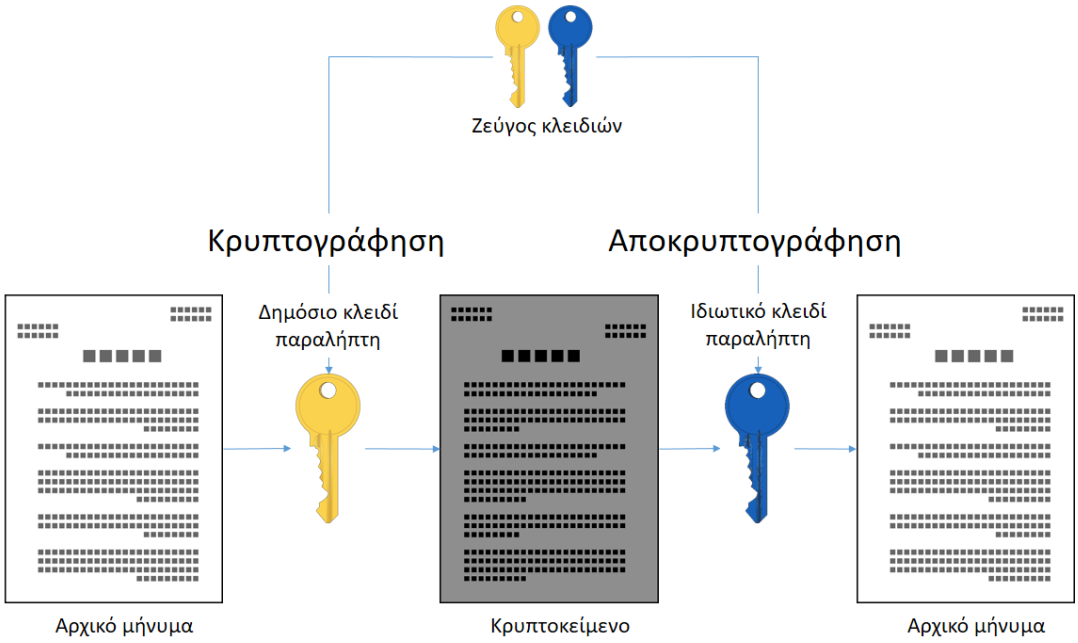
Το τελευταίο στάδιο του αλγορίθμου περιλαμβάνει την εκτέλεση του πρώτου, δεύτερου και τέταρτου μετασχηματισμού του προηγούμενου σταδίου.

Για την αποκρυπτογράφηση εφαρμόζονται οι αντίστροφοι μετασχηματισμοί και η σειρά που εφαρμόζονται σε κάθε γύρο είναι διαφορετική. Έτσι, στους πρώτους N-1 γύρους εφαρμόζονται οι μετασχηματισμοί InvShiftRows, InvSubBytes, AddRoundKey και InvMixColumns. Στον τελευταίο γύρο εφαρμόζονται οι InvShiftRows, InvSubBytes, AddRoundKey.

2.2.2 Κρυπτογραφία δημοσίου κλειδιού

Η κρυπτογραφία δημοσίου κλειδιού ή αλλιώς ασύμμετρη κρυπτογραφία είναι μία μορφή κρυπτοσυστήματος στο οποίο η κρυπτογράφηση και η αποκρυπτογράφηση πραγματοποιούνται με τη χρήση διαφορετικών κλειδιών· ένα δημόσιο κλειδί και ένα ιδιωτικό. Η ασύμμετρη κρυπτογραφία μπορεί να χρησιμοποιηθεί τόσο για κρυπτογράφηση όσο και για πιστοποίηση με τη χρήση κατάλληλων αλγορίθμων.

Η πληροφορία κρυπτογραφείται με το δημόσιο κλειδί και αποκρυπτογραφείται μόνο από το ιδιωτικό ενώ για την πιστοποίηση γίνεται η αντίστροφη διαδικασία, δηλαδή κρυπτογραφείται ένα μήνυμα με το ιδιωτικό κλειδί και αποκρυπτογραφείται με το δημόσιο. Έτσι ο παραλήπτης του μηνύματος ξέρει ότι ο αποστολέας έχει στην κατοχή του το ιδιωτικό κλειδί. Η αποτελεσματικότητα της ασύμμετρης κρυπτογραφίας έγκειται στη δυσκολία εξαγωγής του ιδιωτικού κλειδιού γνωρίζοντας το δημόσιο. Η δυσκολία υφίσταται επειδή οι κρυπταλγόριθμοι βασίζονται σε μαθηματικά προβλήματα τα οποία δε μπορούν να λυθούν εύκολα (παραγοντοποίηση ακεραίων, διακριτοί λογάριθμοι). Συνέπεια της δυσκολίας αυτής είναι και η χρήση περισσότερων πόρων για την εφαρμογή κρυπταλγορίθμων δημοσίου κλειδιού. Για το λόγο αυτό, οι αλγόριθμοι δημοσίου κλειδιού χρησιμοποιούνται συνήθως για την κρυπτογράφηση μικρών τμημάτων πληροφορίας. Μια τυπική χρήση είναι η δημιουργία ασφαλούς καναλιού για την ανταλλαγή ενός μυστικού κλειδιού όπου μετά εφαρμόζεται κρυπτογράφηση συμμετρικού κλειδιού στην πληροφορία.



Εικόνα 2.4 Το μοντέλο του κρυπτοσυστήματος δημοσίου κλειδιού

Η κρυπτογραφία δημοσίου κλειδιού βρίσκεται σε διάφορα πρότυπα του διαδικτύου εκ των οποίων το πιο συχνά χρησιμοποιούμενο είναι το πρωτόκολλο Ασφάλειας Επιπέδου Μεταφοράς (Transport Layer Security - TLS). Άλλα πρωτόκολλα τα οποία βασίζονται στην κρυπτογραφία δημοσίου κλειδιού είναι το SSH, PGP, GPG και Bitcoin.

RSA

Πρόκειται για έναν αλγόριθμο ο οποίος κάνει χρήση της κρυπτογραφίας δημοσίου κλειδιού. Παρουσιάστηκε το 1977 από τους Ron **R**ivest, Adi **S**hamir και Leonard **A**dleman [21] από τα ονόματα των οποίων βγήκε και το ακρωνύμιο RSA. Το 1977 κατατέθηκε πατέντα από το MIT η οποία έκανε χρήση του αλγορίθμου και τελικά δόθηκε το 1983 με ισχύ 17 χρόνια [22]. Η πατέντα χορηγήθηκε μόνο στις ΗΠΑ αφού είχε ήδη εκδοθεί η εργασία των RSA πριν την κατάθεση της πατέντας. Επίσης ο αλγόριθμος δόθηκε στο κοινό από την εταιρία RSA Security[1] (εταιρία την οποία ίδρυσαν οι εμπνευστές του αλγορίθμου) μερικές μέρες πριν τη λήξη της ισχύος την πατέντας.

Ο αλγόριθμος αυτός περιλαμβάνει τέσσερα στάδια: τη δημιουργία κλειδιών, το διαμοιρασμό του δημοσίου, την κρυπτογράφηση και την αποκρυπτογράφηση. Το τυπικό μήκος των κλειδιών κυμαίνεται από 1024 έως 4096 bit. Η διαδικασία για τη δημιουργία των κλειδιών έχει ως εξής:

1. Ο χρήστης διαλέγει τυχαία δύο πολύ μεγάλους πρώτους αριθμούς p, q και υπολογίζει το γινόμενό τους $n = p * q$.
2. Υπολογίζεται η συνάρτηση του Όιλερ $\varphi(n) = (p-1) * (q-1)$.

[1] https://www.rsa.com/en-us

3. Επιλέγεται ένας αριθμός $e > 1$ ώστε $e^{\varphi(v)} \equiv 1\ (mod\ n)$.
4. Υπολογίζεται αριθμός d ώστε $d \equiv e^{-1}(mod\ \varphi(n))$.

Το δημόσιο κλειδί περιλαμβάνει τους αριθμούς n και e ενώ το ιδιωτικό τους αριθμούς n και d.

Ο διαμοιρασμός του δημοσίου κλειδιού μπορεί να γίνει μέσω ενός οποιουδήποτε (ακόμα και μη ασφαλούς) καναλιού. Η εύρεση των p και q και συνεπώς του d είναι πρακτικά αδύνατη με αποτέλεσμα να μην υπάρχει κίνδυνος εύρεσης του ιδιωτικού κλειδιού (κλειδί αποκρυπτογράφησης).

Όσον αφορά την κρυπτογράφηση, το μήνυμα M μετατρέπεται σε έναν ακέραιο m τέτοιο ώστε $0 \leq m < n$ και ύστερα εφαρμόζοντας την εξίσωση $c \equiv m^e\ (mod\ n)$ παράγεται το κρυπτοκείμενο.

Η αποκρυπτογράφηση επιτυγχάνεται βρίσκοντας πρώτα τον ακέραιο m και στη συνέχεια με την αντίστροφη διαδικασία μετατρέπεται ο ακέραιος στο αρχικό μήνυμα M.

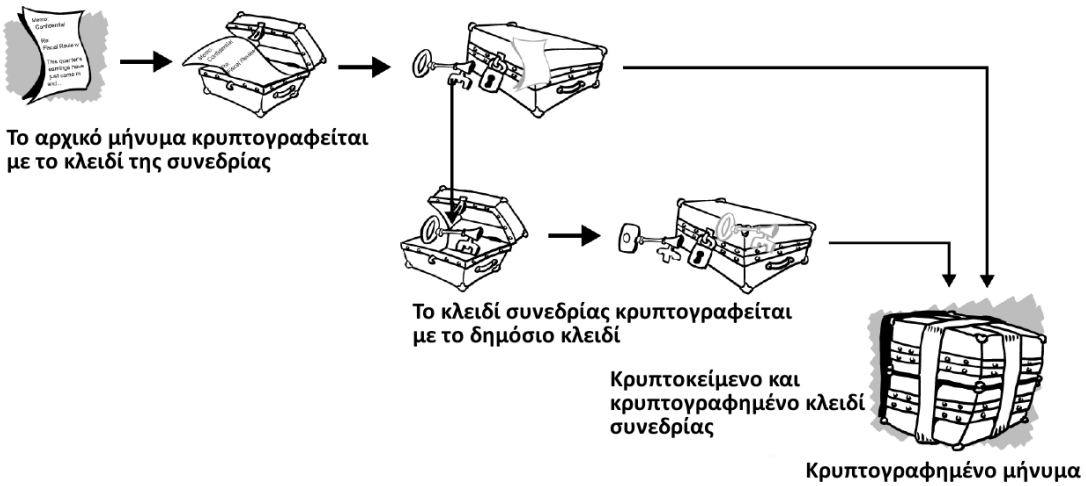
Στο σημείο αυτό αξίζει να παρατηρήσουμε ότι το μήκος του μηνύματος πρέπει να είναι το πολύ κατά μία μονάδα μικρότερο από τον αριθμό n. Αυτό συμβαίνει γιατί ο RSA δεν είναι αλγόριθμος τμήματος. Στην περίπτωση που χρησιμοποιηθεί σαν αλγόριθμος τμήματος και το αρχικό μήνυμα χωριστεί και κρυπτογραφηθεί με το ίδιο κλειδί, αυξάνεται ο κίνδυνος αποκρυπτογράφησής του από τρίτους. Αυτός είναι και ο λόγος για τον οποίο ο αλγόριθμος RSA χρησιμοποιείται ευρέως για την κρυπτογράφηση ενός κλειδιού και η εφαρμογή κάποιου αλγόριθμου συμμετρικού κλειδιού στην ίδια την πληροφορία.

2.2.3 Υβριδική κρυπτογραφία

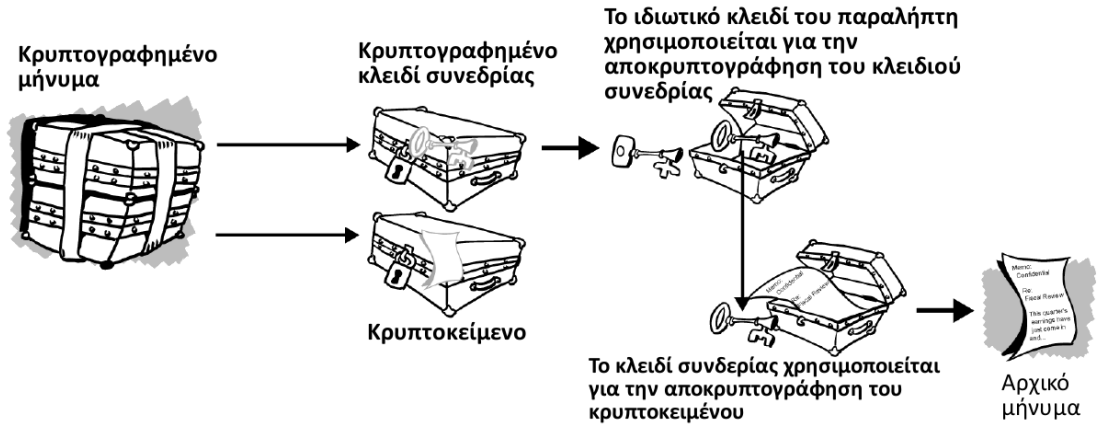
Ένα νομοσχέδιο του 1991 στις ΗΠΑ το οποίο ανέφερε ότι οι πάροχοι ηλεκτρονικών υπηρεσιών επικοινωνίας και οι κατασκευαστές προϊόντων ηλεκτρονικής επικοινωνίας έπρεπε να έχουν μία "κερκόπορτα" στα συστήματά τους ώστε να επιτρέπουν στην κυβέρνηση να αποκτήσει πρόσβαση στα δεδομένα επικοινωνίας, ήταν αυτό που οδήγησε τον Phil Zimmermann στην έκδοση του PGP (Pretty Good Privacy), ένα λογισμικό (τότε) το οποίο κρυπτογραφούσε μηνύματα [23]. Το 1997 προτάθηκε στο IETF η προτυποποίηση του PGP με όνομα OpenPGP η οποία έγινε αποδεκτή και έκτοτε το OpenPGP είναι πρότυπο σε ενεργή ανάπτυξη και χρησιμοποιείται για την ανταλλαγή μηνυμάτων ηλεκτρονικού ταχυδρομείου.

Το PGP συνδυάζει χαρακτηριστικά από την κρυπτογραφία συμμετρικού κλειδιού, την κρυπτογραφία δημοσίου κλειδιού καθώς και τις συναρτήσεις κατακερματισμού, για το λόγο αυτό θεωρείται υβριδικό κρυπτοσύστημα. Αρχικά το μήνυμα συμπιέζεται έτσι ώστε με αυτό τον τρόπο να ενισχύεται η ασφάλεια της κρυπτογράφησης. Στη συνέχεια, δημιουργείται ένα προσωρινό μυστικό κλειδί (κλειδί συνεδρίας) το οποίο θα χρησιμοποιηθεί μόνο για την κρυπτογράφηση αυτού του μηνύματος. Χρησιμοποιώντας κρυπτογραφία συμμετρικού κλειδιού, το μήνυμα κρυπτογραφείται με το μυστικό κλειδί. Αφού κρυπτογραφηθεί το μήνυμα, το μυστικό κλειδί κρυπτογραφείται κάνοντας χρήση

κρυπτογραφίας δημοσίου κλειδιού με το δημόσιο κλειδί του παραλήπτη. Με αυτό τον τρόπο διασφαλίζεται ότι το μυστικό κλειδί (και συνεπώς το κρυπτογραφημένο μήνυμα) θα μπορέσει να το αποκρυπτογραφήσει μόνο ο κάτοχος του ιδιωτικού κλειδιού. Τέλος, το κρυπτογραφημένο μήνυμα μαζί με το κλειδί αποστέλλονται στον παραλήπτη. Για την αποκρυπτογράφηση, ακολουθείται η αντίστροφη διαδικασία. Ο παραλήπτης χρησιμοποιεί το ιδιωτικό κλειδί για να αποκρυπτογραφήσει το προσωρινό μυστικό κλειδί και στη συνέχεια χρησιμοποιεί το τελευταίο για να αποκρυπτογραφήσει το κρυπτοκείμενο.



Εικόνα 2.5 Το μοντέλο κρυπτογράφησης του PGP, [16]



Εικόνα 2.6 Το μοντέλο αποκρυπτογράφησης του PGP, [16]

Εκτός από την ασφαλή μετάδοση πληροφορίας, το PGP χρησιμοποιείται και για την ψηφιακή υπογραφή του μηνύματος. Υπογράφοντας ένα μήνυμα, ο παραλήπτης μπορεί να είναι σίγουρος ότι τα δεδομένα δεν αλλοιώθηκαν από κάποιο τρίτο και ότι ο αποστολέας είναι αυτός που ισχυρίζεται ότι είναι.

Για τη δημιουργία της υπογραφής, το μήνυμα πρώτα περνάει από μία συνάρτηση κατακερματισμού . Αυτό γίνεται γιατί το μήκος του μηνύματος μπορεί να είναι πολύ μεγάλο, κάτι που θα δυσκόλευε στη μεταφορά της υπογραφής. Οπότε η περίληψη του

μηνύματος (message digest) που δημιουργείται από τη συνάρτηση κατακερματισμού έχει σταθερό και μικρό μήκος. Με την περίληψη του μηνύματος και το ιδιωτικό κλειδί του αποστολέα δημιουργείται η υπογραφή η οποία αποστέλλεται μαζί με τα υπόλοιπα δεδομένα. Ο παραλήπτης με την ίδια διαδικασία δημιουργεί την περίληψη του μηνύματος την οποία μετά συνδυάζει με το δημόσιο κλειδί του αποστολέα. Το αποτέλεσμα συγκρίνεται με την υπογραφή και πιστοποιείται η αυθεντικότητα του μηνύματος και του αποστολέα.

ΚΕΦΑΛΑΙΟ 3:
Αρχιτεκτονικές προστασίας ιδιωτικότητας

Η προστασία της ιδιωτικότητας είναι ένα θέμα το οποίο απασχόλησε αρκετούς ανθρώπους πριν την αύξηση της χρήσης των κινητών συσκευών. Αυτό είχε σαν συνέπεια την ανάπτυξη διαφόρων αρχιτεκτονικών για την προστασία της. Η χρήση αυτών των αρχιτεκτονικών αρχικά γινόταν στους προσωπικούς υπολογιστές και κάποιες από αυτές μεταφέρθηκαν και στις κινητές συσκευές. Στο υπόλοιπο κεφάλαιο περιγράφονται οι αρχιτεκτονικές αυτές.

3.1 Κρυπτογράφηση από άκρο σε άκρο

Σε πολλές εφαρμογές επικοινωνίας, εφαρμόζονται τεχνικές κρυπτογράφησης ώστε τα δεδομένα να μην είναι προσβάσιμα από τρίτους. Τα κλειδιά για αυτή την κρυπτογράφηση δίνονται από τον εξυπηρετητή ο οποίος μπορεί ανά πάσα στιγμή να αποκρυπτογραφήσει κάποια την επικοινωνία μεταξύ δύο ή και περισσότερων χρηστών του. Συνεπώς αν κάποιος τρίτος παραβιάσει τον εξυπηρετητή μπορεί εύκολα να αποκτήσει πρόσβαση σε πληροφορία που έχει ανταλλαχθεί μεταξύ αυτών. Μία τεχνική για να λυθεί αυτό το πρόβλημα είναι η κρυπτογράφηση από άκρο σε άκρο (End to End Encryption - EE2E) [24]. Πριν την αποστολή των δεδομένων στον εξυπηρετητή, κρυπτογραφούνται από τον αποστολέα και η αποκρυπτογράφησή τους είναι δυνατή με κλειδί που έχει στην κατοχή του μόνο ο παραλήπτης (ή οι παραλήπτες). Η επικοινωνία η οποία περιλαμβάνει την αποστολή των δεδομένων από τον αρχικό αποστολέα προς τον εξυπηρετητή και από τον εξυπηρετητή προς τον παραλήπτη μπορεί επίσης να είναι κρυπτογραφημένη με κλειδιά που έχει στην κατοχή του ο εξυπηρετητής.

Για την ανταλλαγή των κλειδιών μεταξύ του αποστολέα και του παραλήπτη χρησιμοποιούνται τεχνικές που κάνουν χρήση του PGP ή την ανταλλαγή κλειδιών Diffie-Hellman (Diffie-Hellman key exchange).

Παρόλο που η κρυπτογράφηση των δεδομένων μπορεί να είναι ισχυρή και να είναι πρακτικά αδύνατη η αποκρυπτογράφησή τους από κάποιον τρίτο, με την επίθεση ενδιάμεσου (man-in-the-middle attack) είναι δυνατή η πρόσβαση σε αυτά από κάποιον κακόβουλο. Κατά την ανταλλαγή των κλειδιών ο ενδιάμεσος ανταλλάσσει κλειδί με όλα τα άκρα και κάθε φορά που αποστέλλεται κάτι από το ένα άκρο στο άλλο το αποκρυπτογραφεί με το κλειδί του ενός άκρου, το κρυπτογραφεί με το κλειδί του άλλου άκρου και το προωθεί στο άλλο άκρο. Φυσικά αν ο ενδιάμεσος αποχωρήσει από το κανάλι επικοινωνίας, η επικοινωνία μεταξύ των δύο άκρων θα είναι αδύνατη. Η δημιουργία ενός ψηφιακού αποτυπώματος του κλειδιού (μέσω συνάρτησης

κατακερματισμού) και η σύγκριση αυτών (όχι απαραίτητα μέσω ενός ασφαλούς καναλιού) μετά την ανταλλαγή κλειδιών είναι ένας συνήθης τρόπος για να αποφευχθεί κάποια τέτοια επίθεση.

Όσο δυνατή και να είναι η κρυπτογράφηση στην ανταλλαγή των δεδομένων και η λήψη μέτρων για την αποφυγή επίθεσης ενδιάμεσου, υπάρχει πάντα ο κίνδυνος πρόσβασης σε αυτά πριν ή μετά την (από)κρυπτογράφησή τους. Αυτό μπορεί να γίνει αν κάποιος αποκτήσει πρόσβαση στη συσκευή. Τεχνικές για την αποφυγή πρόσβασης σε δεδομένα που βρίσκονται σε μια φορητή συσκευή θα αναλυθούν σε επόμενο κεφάλαιο.

3.2 Εικονικά Ιδιωτικά Δίκτυα

Το Εικονικό Ιδιωτικό Δίκτυο - ΕΙΔ (Virtual Private Network - VPN) είναι μία επέκταση ενός ιδιωτικού δικτύου και δίνει τη δυνατότητα στους χρήστες του, να έχουν απομακρυσμένη ασφαλή πρόσβαση σε ένα ιδιωτικό δίκτυο. Οι χρήστες μπορούν να στείλουν και να λάβουν δεδομένα με τον ίδιο τρόπο που θα γινόταν αν ήταν φυσικά συνδεδεμένοι στο ιδιωτικό αυτό δίκτυο.

Ένα ΕΙΔ επιτρέπει τη δημιουργία μίας σήραγγας, μία σύνδεση από τη συσκευή προς έναν εξυπηρετητή σε οποιοδήποτε σημείο του κόσμου μέσω του διαδικτύου. Οποιοδήποτε αίτημα πρόσβασης σε υπηρεσία ή ιστοσελίδα από μία συσκευή περνάει τη σήραγγα προς τον εξυπηρετητή και τελικά ο εξυπηρετητής είναι αυτός που πραγματοποιεί το αίτημα. Η απάντηση του αιτήματος γίνεται από την υπηρεσία προς τον εξυπηρετητή και μέσω της σήραγγας καταλήγει στη συσκευή. Τα δεδομένα που ανταλλάσσονται μέσω της σήραγγας είναι κρυπτογραφημένα.

Η ιστορία του ΕΙΔ ξεκινάει το 1996 τότε που ο Gurdeep Singh-Pall, μηχανικός λογισμικού της Microsoft, ανέπτυξε το Πρωτόκολλο Σήραγγας Ομότιμων Κόμβων (Peer-to-Peer Tunneling Protocol - PPTP) με στόχο την πιο ασφαλή και ιδιωτική σύνδεση μεταξύ ενός υπολογιστή και του διαδικτύου. Στις αρχές η χρήση των ΕΙΔ γινόταν κυρίως από επιχειρήσεις λόγω της ανάγκης των υπαλλήλων να προσπελάσουν αρχεία τα οποία βρίσκονταν στο ιδιωτικό δίκτυο της επιχείρησης από άλλη τοποθεσία. Με τον καιρό εμφανίστηκαν πάροχοι ΕΙΔ με σκοπό την χρήση τους από τους τελικούς χρήστες. Στις μέρες μας τα ΕΙΔ έχουν τρεις κύριες χρήσεις:

1. Σύνδεση σε απομακρυσμένα ιδιωτικά δίκτυα,
2. Παράκαμψη περιορισμών από το δίκτυο που έχει συνδεθεί κάποιος και τέλος
3. Απόκρυψη της δραστηριότητας του χρήστη στο διαδίκτυο.

Η πρώτη χρήση είναι και η πρωταρχική χρήση των ΕΙΔ η οποία δίνει τη δυνατότητα στους χρήστες να έχουν πρόσβαση σε αρχεία στο οικιακό τους δίκτυο ή/και στο δίκτυο της επιχείρησης όπου εργάζονται.

Είναι σύνηθες σε κάποιες επιχειρήσεις, πανεπιστήμια, ακόμα και δημόσια σημεία πρόσβασης στο διαδίκτυο (Hotspots) να εμποδίζουν την πρόσβαση σε ορισμένες ιστοσελίδες ή υπηρεσίες όπως το Facebook ή το YouTube. Τέτοια εμπόδια υπάρχουν και σε ορισμένες χώρες (π.χ. Κίνα, Αίγυπτος, Συρία κ.α.) όπου απαγορεύουν και εμποδίζουν την είσοδο σε κάποιες υπηρεσίες. Χρησιμοποιώντας κάποιο ΕΙΔ, ο χρήστης μπορεί να έχει πρόσβαση στις αποκλεισμένες υπηρεσίες, αφού η πρόσβαση στην υπηρεσία γίνεται μέσω του ΕΙΔ. Υπάρχουν όμως και οι αντίθετες περιπτώσεις, εκείνες στις οποίες επιτρέπεται η πρόσβαση σε κάποια υπηρεσία μόνο από μία (ή λιγοστές χώρες). Χαρακτηριστικό παράδειγμα είναι το Netflix, μία υπηρεσία μετάδοσης βίντεο κατά απαίτηση (Video on Demand – VoD), όπου επιτρεπόταν η πρόσβαση στις υπηρεσίες βίντεο μόνο σε όσους βρίσκονταν στις ΗΠΑ. Ακόμα και σε αυτές τις περιπτώσεις, χρησιμοποιώντας κάποιο πάροχο ΕΙΔ ο οποίος βρίσκεται στην χώρα στην οποία επιτρέπεται η πρόσβαση στις μπλοκαρισμένες υπηρεσίες, είναι δυνατή η χρήση τους από άλλες χώρες.

Ένας χρήστης μπορεί να συνδεθεί με το κινητό του τηλέφωνο σε ένα δημόσιο σημείο πρόσβασης στο διαδίκτυο και να πραγματοποιήσει μία αγορά με την πιστωτική του κάρτα. Κάποιος άλλος ο οποίος έχει συνδεθεί στο ίδιο σημείο πρόσβασης, μπορεί με εύκολο τρόπο να έχει πρόσβαση σε αυτή τη συναλλαγή και συνεπώς να μάθει τι ήταν αυτό που αγόρασε ο χρήστης ή ακόμα και πληροφορίες της κάρτας, ειδικά στην περίπτωση που η συναλλαγή δεν ήταν κρυπτογραφημένη. Με ανάλογο τρόπο μπορεί η κυβέρνηση μίας χώρας να ζητήσει τις ίδιες πληροφορίες από τον πάροχο διαδικτύου του χρήστη. Χρησιμοποιώντας κάποιο ΕΙΔ και εφόσον η πληροφορία μέχρι τον εξυπηρετητή του ΕΙΔ είναι κρυπτογραφημένη, πραγματοποιώντας την ίδια συναλλαγή η ανάκτησή της και η πρόσβαση στις πληροφορίες της δεν είναι εφικτές. Συνεπώς, με τη χρήση ΕΙΔ η δραστηριότητα ενός χρήστη είναι δυνατό να αποκρυφθεί.

Για τον έλεγχο της καλής τους λειτουργίας και την εύκολη συντήρηση, οι εξυπηρετητές καταγράφουν σε αρχεία ποιος είχε πρόσβαση σε αυτούς και πότε. Οπότε έχοντας κάποιος πρόσβαση σε αυτά τα αρχεία και στην κίνηση του χρήστη, μπορεί συνδυάζοντάς τα να σχηματίσει ολόκληρη την κίνηση ενός χρήστη. Φυσικά κάτι τέτοιο είναι αρκετά δύσκολο και χρονοβόρο, διότι απαιτείται η συνεργασία σωμάτων ασφαλείας διαφορετικών χωρών. Επιπροσθέτως, οι πάροχοι ΕΙΔ πολλές φορές διαλέγουν να εγκαταστήσουν τους εξυπηρετητές σε χώρες που υπάρχει αρκετή διαδικασία για να αποκτήσει κάποιος πρόσβαση σε αυτά τα αρχεία, δυσχεραίνοντας αυτόν που θέλει να έχει πρόσβαση. Τέλος, τα αρχεία καταγραφής διαγράφονται μετά από ορισμένο χρονικό διάστημα και σε αυτή την περίπτωση καθίσταται αδύνατος ο σχηματισμός κίνησης ενός χρήστη.

3.3 Ανώνυμη περιήγηση στο διαδίκτυο

Το 1998, δύο ερευνητές των εργαστηρίων της AT&T, παρουσίασαν μία αρχιτεκτονική για ανώνυμη περιήγηση στο διαδίκτυο με το όνομα Crowds (ή πλήθη) [25]. Σύμφωνα με αυτή την αρχιτεκτονική, τα αιτήματα προς τους εξυπηρετητές ιστού δρομολογούνται μέσα από έναν τυχαίο αριθμό κόμβων. Μέσω ενός προγράμματος στον υπολογιστή του χρήστη με το όνομα Jondo (από το αγγλικό John Doe που σημαίνει άγνωστος), ο χρήστης συνδέεται σε ένα δίκτυο από άλλους χρήστες που χρησιμοποιούν το εν λόγω πρόγραμμα. Το δίκτυο αυτό ονομάζεται πλήθος (ή crowd) εξ ου και το όνομα. Όταν ο χρήστης θελήσει να επισκεφθεί κάποια ιστοσελίδα, το πρόγραμμα αποφασίζει τυχαία και με ίση πιθανότητα τον επόμενο κόμβο και στέλνει το αίτημα. Ο επόμενος κόμβος αποφασίζει τυχαία αν θα προωθήσει το αίτημα σε άλλο κόμβο ή στον τελικό προορισμό. Η διαδικασία ακολουθείται για κάθε κόμβο που φτάνει κάποιο αίτημα μέχρι και τον τελικό προορισμό. Κάθε κόμβος καταγράφει τον προηγούμενό του ώστε να είναι δυνατή η επικοινωνία μεταξύ του χρήστη και του τελικού προορισμού.

3.4 Δρομολόγηση κρεμμυδιού (Onion routing)

Μία τεχνική για ανώνυμη επικοινωνία μέσω ενός δικτύου είναι αυτή της δρομολόγησης κρεμμυδιού [26]. Τα δεδομένα κρυπτογραφούνται και το αποτέλεσμα της κρυπτογράφησης κρυπτογραφείται πάλι κ.ο.κ. Έτσι, δημιουργούνται επίπεδα δεδομένων όπως και τα επίπεδα ενός κρεμμυδιού. Ύστερα τα δεδομένα μεταδίδονται μέσω κόμβων (δρομολογητές κρεμμυδιού) όπου κάθε κόμβος αποκρυπτογραφεί και από ένα επίπεδο αποκαλύπτοντας τον επόμενο κόμβο. Τα δεδομένα φθάνουν στον προορισμό όταν αποκρυπτογραφηθεί και το τελευταίο επίπεδο.

Αιτία της ανάπτυξης της δρομολόγησης κρεμμυδιού ήταν η ανάγκη της Κοινότητας Πληροφορίας των Ηνωμένων Πολιτειών (United States Intelligence Community) για προστασία των επικοινωνιών στο διαδίκτυο. Αναπτύχθηκε αρχικά στα μέσα της δεκαετίας του 1990 στο Ναυτικό Εργαστήριο Ερευνών των Ηνωμένων Πολιτειών (United States Naval Research Laboratory - NRL) και ύστερα από τον Οργανισμό Άμυνας Προηγμένων Ερευνητικών Προγραμμάτων.

Ο δημιουργός του μηνύματος που περιέχει τα δεδομένα, επιλέγει έναν αριθμό κόμβων από ένα κατάλογο μέσω των οποίων θα περάσει το μήνυμα. Η διαδρομή την οποία θα ακολουθήσει το μήνυμα ονομάζεται αλυσίδα και δημιουργείται αφού επιλεχθούν οι κόμβοι. Κανένας από τους κόμβους δε γνωρίζει αν ο προηγούμενός του είναι ο αποστολέας του μηνύματος ή ένας ακόμα κόμβος καθώς και πόσοι άλλοι κόμβοι υπάρχουν στην αλυσίδα, παρά μόνο ο τελευταίος γνωρίζει τη θέση του μέσα σε αυτή. Με αυτό τον τρόπο επιτυγχάνεται ανωνυμία. Ο αποστολέας πριν στείλει το μήνυμα, ανταλλάσσει μυστικά κλειδιά με κάθε κόμβο ξεχωριστά. Ύστερα το μήνυμα

κρυπτογραφείται πρώτα με το κλειδί που έχει ανταλλάξει με τον τελευταίο κόμβο. Στη συνέχεια το μήνυμα κρυπτογραφείται με το κλειδί που έχει ανταλλάξει με το προτελευταίο. Η διαδικασία συνεχίζεται μέχρι το μήνυμα να κρυπτογραφηθεί και με το κλειδί που έχει ανταλλαχθεί με τον πρώτο κόμβο. Όταν τελειώσει η κρυπτογράφηση, το μήνυμα αποστέλλεται στον πρώτο κόμβο, ο οποίος το αποκρυπτογραφεί και βλέπει τον επόμενο κόμβο όπου και το αποστέλλει. Ο επόμενος το αποκρυπτογραφεί και βρίσκει τον επόμενο μέχρι τελικά το μήνυμα να φτάσει στον τελικό κόμβο, ο οποίος το προωθεί στον τελικό του προορισμό. Η απάντηση σε αυτό το μήνυμα ακολουθεί την αντίστροφη πορεία. Πρώτα αποστέλλεται στον κόμβο που ήταν τελευταίος πριν και κρυπτογραφείται. Έπειτα αποστέλλεται στον προτελευταίο κόμβο και κρυπτογραφείται μέχρι να φτάσει στον πρώτο ο οποίος θα στείλει το μήνυμα στον αποστολέα του αρχικού μηνύματος. Ο αρχικός αποστολέας θα παραλάβει ένα μήνυμα κρυπτογραφημένο τόσες φορές όσοι και οι κόμβοι και για να το διαβάσει θα πρέπει πρώτα να το αποκρυπτογραφήσει.

Η διέλευση ενός μηνύματος μέσα από άλλους κόμβους μέχρι τον τελικό προορισμό είναι λογικό να δημιουργεί κάποια καθυστέρηση. Ακόμα μεγαλύτερη καθυστέρηση προκαλείται όταν ο ένας κόμβος είναι σε διαφορετική ήπειρο από τον προηγούμενο και τον επόμενό του.

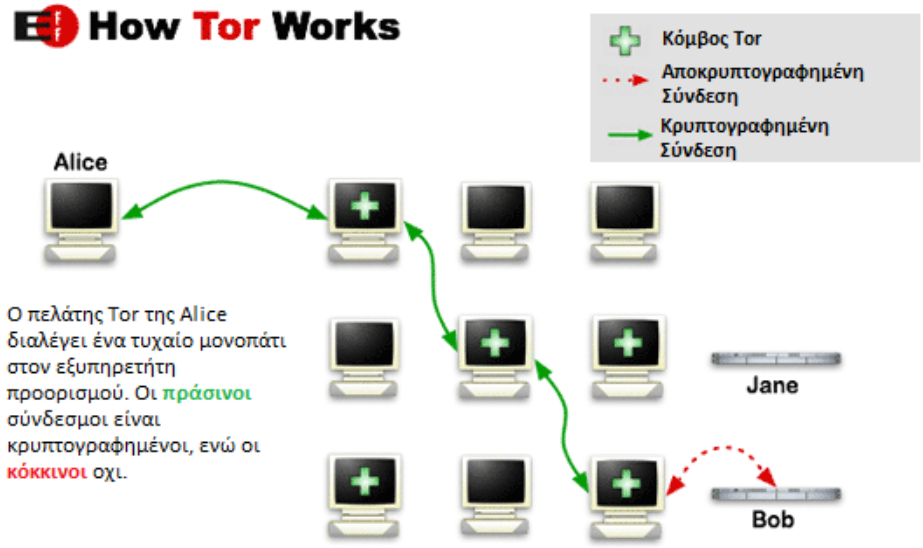
Η καθυστέρηση δεν είναι το μόνο αρνητικό της συγκεκριμένης δρομολόγησης. Αν κάποιος κακόβουλος συλλαμβάνει μηνύματα τα οποία αποστέλλονται από τον τελευταίο κόμβο προς τους τελικούς τους προορισμούς, μπορεί να αποκτήσει πρόσβαση σε ευαίσθητες πληροφορίες όπως κωδικοί, αριθμοί τραπεζικών λογαριασμών ή άλλες προσωπικές πληροφορίες. Η λύση σε αυτό το πρόβλημα είναι η χρήση κρυπτογράφησης από άκρο σε άκρο όπως το TLS πρωτόκολλο.

Ένας τρόπος για να αρθεί η ανωνυμία είναι η χρονική ανάλυση των μηνυμάτων. Παρακολουθώντας κάποιος την είσοδο του πρώτου κόμβου και την έξοδο του τελευταίου, συσχετίζοντας τη στιγμή που πήγε το μήνυμα στον πρώτο κόμβο με τη στιγμή που απεστάλη από τον τελευταίο, μπορεί να συμπεράνει και τον αποστολέα του μηνύματος. Η δρομολόγηση σκόρδου είναι μία παραλλαγή της δρομολόγησης κρεμμυδιού, η οποία κρυπτογραφεί πολλά μηνύματα μαζί ώστε να κάνει πιο δύσκολη μία τέτοια ανάλυση.

Η περίπτωση του Tor

Το πρόγραμμα Tor (The Onion Routing project) [27] είναι μία υλοποίηση της δρομολόγησης κρεμμυδιού που χρησιμοποιείται για ανώνυμη επικοινωνία. Η έκδοση άλφα του Tor λανσαρίστηκε το Σεπτέμβριο του 2002 και η πρώτη έκδοση για το κοινό το 2003. Ένα χρόνο μετά, παρουσιάστηκε η δεύτερη γενιά της δρομολόγησης κρεμμυδιού και την ίδια χρονιά ο πηγαίος κώδικας του Tor δημοσιεύθηκε με ελεύθερη άδεια. Στα τέλη του 2006 ιδρύθηκε ο μη-κερδοσκοπικός οργανισμός The Tor Project με σκοπό τη συντήρηση του Tor.

Όπως διακρίνεται και στην παρακάτω εικόνα, στην υλοποίηση του Tor οι κόμβοι που χρησιμοποιούνται είναι τρεις.



Εικόνα 3.1 Αποστολή δεδομένων με χρήση του Tor, [28]

3.5 Ανώνυμοι επαναποστολείς ηλεκτρονικής αλληλογραφίας

Οι ανώνυμοι επαναποστολείς ηλεκτρονικής αλληλογραφίας (anonymous remailers) είναι ηλεκτρονικοί εξυπηρετητές στους οποίους μπορεί ένας χρήστης να στείλει ένα ηλεκτρονικό μήνυμα το οποίο, αφού αφαιρεθεί η διεύθυνση του αποστολέα, θα προωθηθεί στον παραλήπτη. Το ηλεκτρονικό αυτό μήνυμα είναι γνωστό και ως ανώνυμο ηλεκτρονικό μήνυμα. Οι επαναποστολείς ηλεκτρονικής αλληλογραφίας χωρίζονται σε τέσσερις κατηγορίες.

3.4.1 Επαναποστολείς Τύπου I [29]

Γνωστοί και ως ανώνυμοι επαναποστολείς ηλεκτρονικής αλληλογραφίας Cypherpunk (Cypherpunk anonymous remailer). Αφού παραλάβουν ένα ηλεκτρονικό μήνυμα, αφαιρούν την ηλεκτρονική διεύθυνση του παραλήπτη από την επικεφαλίδα και προωθούν το μήνυμα. Συνήθως τα μηνύματα αυτά είναι κρυπτογραφημένα χρησιμοποιώντας το σχήμα PGP με το δημόσιο κλειδί του επαναποστολέα. Τα μηνύματα αποκρυπτογραφούνται στον επαναποστολέα πριν αποσταλούν στον παραλήπτη. Υπάρχει και η δυνατότητα χρήσης αλυσίδας επαναποστολέων το οποίο μειώνει την πιθανότητα να προσδιοριστεί ο αποστολέας και αυξάνει την ασφάλεια λόγω των περισσότερων επιπέδων κρυπτογράφησης. Τελικά προστέθηκε και η δυνατότητα απάντησης σε αυτούς τους επαναποστολείς χρησιμοποιώντας ένα τμήμα

απάντησης το οποίο ουσιαστικά δείχνει τη διεύθυνση του αποστολέα, η οποία όμως είναι κρυπτογραφημένη με τα κλειδιά των ενδιάμεσων επαναποστολέων. Ένα σημαντικό μειονέκτημα είναι ότι μετά την κρυπτογράφηση με το PGP διατηρείται το αρχικό μέγεθος του μηνύματος και έτσι είναι εύκολο να το ακολουθήσει κανείς μέσα στο δίκτυο παρατηρώντας το μέγεθός του.

3.4.2 Επαναποστολείς Τύπου II [30]

Οι επαναποστολείς Τύπου II (Type II remailers ή Mixmaster) βασίζεται στην ιδέα του mix-net από τον David Chaum. Το μήνυμα χωρίζεται σε κομμάτια ίσου μεγέθους (στο τελευταίο κομμάτι προστίθεται πληροφορία για να έχει ίδιο μέγεθος με τα υπόλοιπα). Κάθε κομμάτι κρυπτογραφείται και ακολουθεί μία συγκεκριμένη σειρά από επαναποστολείς μέχρι τον τελευταίο ο οποίος αναλαμβάνει να ανασχηματίσει το αρχικό μήνυμα από τα κομμάτια, αφού πρώτα τα αποκρυπτογραφήσει και τελικά το μήνυμα προωθείται στον παραλήπτη. Σε αντίθεση με του Τύπου I, δεν υπάρχει δυνατότητα απάντησης στο μήνυμα. Για την αποστολή του κάθε κομματιού στον τελευταίο επαναποστολέα, χρησιμοποιείται το SMTP. Η αδυναμία αυτών των επαναποστολέων επισημάνθηκε το 1990 από τους Pfitzmann & Pfitzmann αποδεικνύοντας ότι δεν παρέχονται οι απαραίτητες ιδιότητες αδυναμίας σύνδεσης (unlinkability).

3.4.3 Επαναποστολείς Τύπου III [31]

Ο επαναποστολέας Τύπου III (Type III remailer ή Mixminion) προσπαθεί να διορθώσει το πρόβλημα του τύπου II το οποίο είναι η απάντηση σε κάποιο μήνυμα. Δουλεύει με παρόμοιο τρόπο με τον Τύπου II με τη διαφορά ότι το μήνυμα από τον αποστολέα στον πρώτο επαναποστολέα είναι με SMTP πρωτόκολλο στου Τύπου II ενώ στου τύπου III είναι ένα δυαδικό μήνυμα κρυπτογραφημένο με SSL. Η απάντηση προς τον αρχικό αποστολέα του μηνύματος γίνεται με τη χρήση Τμήματος Απάντησης Ενός Χρήστη (Single-User Reply Blocks - SURBs).

3.4.4 Ψευδώνυμοι επαναποστολείς

Οι επαναποστολείς αυτού του τύπου, επιτρέπουν στον χρήστη να στείλει ηλεκτρονικά μηνύματα χρησιμοποιώντας κάποιο ψευδώνυμο με τη δυνατότητα να σταλεί πίσω στον αρχικό χρήστη κάποια απάντηση. Χρησιμοποιούν μία βάση η οποία περιέχει τις κατάλληλες οδηγίες για να επιστραφεί κάποιο μήνυμα στον αρχικό χρήστη. Το μειονέκτημα τους είναι ότι λόγω του ότι καταγράφονται στοιχεία για την ταυτότητα κάθε χρήστη, υπάρχει η δυνατότητα να εντοπιστεί είτε νόμιμα είτε παράνομα.

Χαρακτηριστικό παράδειγμα ψευδώνυμου επαναποστολέα είναι ο Επαναποστολέας Penet (Penet remailer) ο οποίος λειτούργησε από το 1993 μέχρι και το 1996. Το Σεπτέμβριο του 1996, η εκκλησία της Σαϊεντολογίας απαίτησε από το χειριστή του εξυπηρετητή να παραδώσει τα στοιχεία ενός χρήστη ο οποίος δημοσίευσε απόρρητα

έγγραφα αυτής μέσω του επαναποστολέα. Λόγω της κριτικής, των επιθέσεων και την ανικανότητα να εγγυηθεί η ανωνυμία των χρηστών, ο χειριστής τελικά έκλεισε τον εξυπηρετητή τον ίδιο μήνα.

3.6 Ανώνυμη αποστολή καταγγελιών

Η εταιρία Anderson Software σε συνεργασία με τον Eric Jacksch ανέπτυξε το σύστημα TipSoft SMS [32] το οποίο επιτρέπει στους πολίτες να πραγματοποιούν ανώνυμες καταγγελίες περιστατικών στα προγράμματα Crime Stoppers μέσω SMS χωρίς να αποκαλύπτεται ο τηλεφωνικός αριθμός. Επιπλέον, παρέχεται η δυνατότητα απάντησης σε κάποια καταγγελία από τα προγράμματα προς τον πολίτη. Αυτό επιτυγχάνεται χάρις στο διπλό ψευδώνυμο, μία προσέγγιση η οποία υιοθετήθηκε από το σύστημα. Τα μηνύματα SMS που περιέχουν την καταγγελία, στέλνονται σε έναν εξυπηρετητή ο οποίος βρίσκεται στον Καναδά και χειρίζεται από τον Jacksch. Σε αυτό τον εξυπηρετητή, ο αριθμός του τηλεφώνου κρυπτογραφείται και του ανατίθεται ένα ψευδώνυμο. Το περιεχόμενο του μηνύματος μαζί με το ψευδώνυμο προωθούνται σε ένα δεύτερο εξυπηρετητή τον οποίο χειρίζεται η εταιρία Anderson Software. Ύστερα αυτός ο εξυπηρετητής αναθέτει ένα δεύτερο ψευδώνυμο στο πρώτο και μαζί με το μήνυμα τα προωθεί στο σύστημα όπου έχουν πρόσβαση τα προγράμματα Crime Stoppers. Συνεπώς, τα προγράμματα έχουν πρόσβαση μόνο στο ψευδώνυμο και το περιεχόμενο του μηνύματος. Για την απάντηση, το πρόγραμμα ακολουθεί την αντίστροφη διαδικασία. Στέλνει την απάντηση στο δεύτερο εξυπηρετητή ο οποίος αντικαθιστά το δεύτερο ψευδώνυμο με το πρώτο και μαζί με το μήνυμα τα προωθεί στον πρώτο εξυπηρετητή. Σε αυτόν γίνεται αντικατάσταση του ψευδωνύμου με τον τηλεφωνικό αριθμό και τελικά το μήνυμα φτάνει στο χρήστη.

ΚΕΦΑΛΑΙΟ 4:
Εφαρμογές προστασίας ιδιωτικότητας

Με τις έρευνες να δείχνουν ότι οι άνθρωποι χρησιμοποιούν το κινητό τηλέφωνο και την ταμπλέτα κυρίως για να στείλουν γραπτά μηνύματα, να πλοηγηθούν στο διαδίκτυο ή να τραβήξουν φωτογραφίες [33], [34] και τις εταιρίες παροχής αυτών των υπηρεσιών να έχουν αόριστες πολιτικές απορρήτου σχετικά με το τι δεδομένα συλλέγουν και με ποιους τα μοιράζονται, προκύπτει η ανάγκη προστασίας των προσωπικών δεδομένων του χρήστη. Τα δεδομένα που συλλέγουν οι εταιρίες χρησιμοποιούνται για τη βελτίωση των υπηρεσιών τους αλλά και για διαφημιστικούς σκοπούς. Εκτός από τα δεδομένα που αποστέλλονται στο διαδίκτυο, πρόσβαση σε ευαίσθητα δεδομένα μπορεί να αποκτήσει κάποιος έχοντας στην κατοχή του τη συσκευή ενός τρίτου. Τα δεδομένα που υπάρχουν στη συσκευή είναι προσβάσιμα και από τις διάφορες εφαρμογές που τρέχουν σε αυτή.

Στη συνέχεια του κεφαλαίου γίνεται προσπάθεια εύρεσης τρόπων ώστε να διασφαλισθούν τα δεδομένα των χρηστών όσο το δυνατόν περισσότερο από τρίτους.

4.1 Προστασία δεδομένων από παραβιάσεις μέσω διαδικτύου

Καθημερινά χιλιάδες χρήστες ανταλλάσσουν μηνύματα μέσω ηλεκτρονικού ταχυδρομείου (e-mail) χρησιμοποιώντας το κινητό τους τηλέφωνο. Το περιεχόμενο των μηνυμάτων περιλαμβάνει κάθε είδους πληροφορία, η οποία πολλές φορές μπορεί να είναι ευαίσθητη.

Από τις πρώτες μέρες ύπαρξης του διαδικτύου μέχρι και σήμερα, ο τρόπος ανταλλαγής ηλεκτρονικών μηνυμάτων δεν έχει αλλάξει. Ο αποστολέας συντάσσει ένα μήνυμα χρησιμοποιώντας ένα πρόγραμμα στον υπολογιστή ή/και κινητό του και το στέλνει στον εξυπηρετητή του παρόχου της υπηρεσίας όπου και το αποθηκεύει. Ο παραλήπτης με τη χρήση ενός παρόμοιου προγράμματος ρωτάει τον εξυπηρετητή αν υπάρχει κάποιο μήνυμα γι' αυτόν και στην περίπτωση θετικής απάντησης του προωθεί το μήνυμα.

Οι πάροχοι υπηρεσιών ηλεκτρονικού ταχυδρομείου είναι πάρα πολλοί. Οι τρεις μεγαλύτεροι που ξεχωρίζουν και κατέχουν πάνω από το 60% του μεριδίου αγοράς είναι το Gmail, το Hotmail και το Yahoo Mail [35]. Πολλές εταιρίες και οργανισμοί χρησιμοποιούν αυτούς τους παρόχους για να προσφέρουν υπηρεσίες ηλεκτρονικού ταχυδρομείου στα μέλη τους, διότι το κόστος ανάπτυξης και συντήρησης μίας αυτόνομης υπηρεσίας είναι πολλές φορές μεγαλύτερο από την αγορά της υπηρεσίας από αυτούς τους παρόχους.

Οι πολιτικές απορρήτου των Google [36], Microsoft [37] και Yahoo [38] (των εταιριών που προσφέρουν τις ανωτέρω υπηρεσίες ηλεκτρονικού ταχυδρομείου) αναφέρουν ότι οι εταιρίες αυτές συλλέγουν πληροφορίες που αφορούν το χρήστη. Οι πληροφορίες αυτές είναι εκείνες που έδωσε ο ίδιος ο χρήστης κατά την εγγραφή του στην υπηρεσία (όπως όνομα, τηλέφωνο, διεύθυνση) αλλά και πληροφορίες που συλλέγονται κάνοντας χρήση της υπηρεσίας (π.χ. πληροφορίες συσκευής, τοποθεσία κλπ.). Οι λόγοι συλλογής αυτών των πληροφοριών ποικίλλουν και περιλαμβάνουν τη συντήρηση της υποδομής της υπηρεσίας, προστασία των χρηστών από επιθέσεις (για παράδειγμα το Gmail στέλνει μήνυμα στο χρήστη αν εισέλθει στην υπηρεσία από κάποια άλλη τοποθεσία ή άλλη συσκευή εκτός της συνηθισμένης), βελτίωση της υπάρχουσας υπηρεσίας κ.α. Σχετικά με την κοινή χρήση της πληροφορίας με τρίτους, όλες οι ανωτέρω εταιρίες αναφέρουν ότι έχουν το δικαίωμα να μοιραστούν τις πληροφορίες με εταιρίες συνεργάτες για την επεξεργασία των δεδομένων αλλά και για νομικούς λόγους. Μόνο η Google αναφέρει ότι θα μοιραστεί πληροφορίες με τρίτους (εκτός των παραπάνω περιπτώσεων) μόνο με τη συγκατάθεση του χρήστη. Η μόνη αναφορά στο περιεχόμενο των ηλεκτρονικών μηνυμάτων γίνεται από τη Microsoft η οποία δηλώνει ότι δεν το χρησιμοποιεί για διαφημιστικούς σκοπούς. Αξίζει να σημειωθεί ότι καμία εταιρία δεν αναφέρει το χρονικό διάστημα αποθήκευσης των πληροφοριών.

Η αποστολή και λήψη των μηνυμάτων μεταξύ των συσκευών και των εξυπηρετητών στις περισσότερες περιπτώσεις είναι κρυπτογραφημένη. Παρόλα αυτά, στον εξυπηρετητή και τις συσκευές τα μηνύματα παραμένουν σε αναγνώσιμη μορφή. Όπως διαπιστώθηκε από τις δηλώσεις απορρήτου, είναι στη διακριτική ευχέρεια της κάθε εταιρίας το αν και σε ποιους θα δώσει πρόσβαση σε αυτά τα δεδομένα. Εκτός από την οικειοθελή κοινοποίηση των δεδομένων, πρόσβαση μπορεί να υπάρξει και με ηλεκτρονική επίθεση στους εξυπηρετητές της εταιρίας οπότε τα δεδομένα θα βρεθούν στα χέρια τρίτων με παράνομες διαδικασίες.

Για την αποφυγή υποκλοπής των δεδομένων, υπάρχουν αρκετοί πάροχοι οι οποίοι προσφέρουν υπηρεσίες ηλεκτρονικού ταχυδρομείου με κρυπτογράφηση. Ο πιο γνωστός είναι ο ProtonMail. Ο συγκεκριμένος πάροχος κρυπτογραφεί τα μηνύματα του χρήστη με ένα ιδιωτικό κλειδί το οποίο βρίσκεται στον εξυπηρετητή του, ώστε να μπορεί ο χρήστης να αποκρυπτογραφεί τα μηνύματά του από οπουδήποτε συνδεθεί. Για την αποκρυπτογράφηση χρειάζεται και ένας κωδικός που γνωρίζει μόνο ο χρήστης. Συνεπώς αν κάποιος αποκτήσει πρόσβαση στα μηνύματα και στο κλειδί του χρήστη (είτε νόμιμα είτε παράνομα) δε θα είναι σε θέση να τα αποκρυπτογραφήσει. Όπως προκύπτει από την πολιτική απορρήτου [39], τα δεδομένα που συλλέγονται είναι ελάχιστα σε σχέση με τις άλλες εταιρίες. Άλλοι πάροχοι που προσφέρουν κρυπτογράφηση είναι οι Disroot, Tutanota, Mailbox.org κ.α.

Η αλλαγή παρόχου ηλεκτρονικού ταχυδρομείου δεν είναι βολική διότι πρέπει να αλλάξει η διεύθυνση ηλεκτρονικού ταχυδρομείου και θα πρέπει να ενημερωθεί πολύς κόσμος γι' αυτή την αλλαγή. Επίσης πολλοί πάροχοι δε μένουν ενεργοί για πολλά χρόνια και ο χρήστης αναγκάζεται να αλλάξει ξανά διεύθυνση. Μία εναλλακτική για αυτό είναι η

χρήση του PGP. Για να χρησιμοποιηθεί το PGP πρέπει ο χρήστης να φτιάξει ένα πιστοποιητικό το οποίο περιλαμβάνει ένα δημόσιο και ένα ιδιωτικό κλειδί με τα οποία γίνεται η (από)κρυπτογράφηση των μηνυμάτων. Το ιδιωτικό κλειδί το έχει μόνο ο χρήστης στην κατοχή του ενώ το δημόσιο μαζί με το πιστοποιητικό αναρτάται σε ένα αποθετήριο από όπου μπορούν οι υπόλοιποι χρήστες να το κατεβάσουν και να το προσθέσουν στην κλειδοθήκη τους. Ο χρήστης πλέον μπορεί να επιλέξει αν το μήνυμα που προτίθεται να στείλει θα κρυπτογραφηθεί. Για να γίνει αυτό πρέπει να έχει και ο άλλος χρήστης κάποιο κλειδί το οποίο θα χρησιμοποιήσει ο πρώτος για την κρυπτογράφηση. Σε κάθε μήνυμα που στέλνει ο χρήστης έχει τη δυνατότητα να επισυνάπτει και την υπογραφή του ώστε οι παραλήπτες του μηνύματος να βεβαιωθούν για την αυθεντικότητά του.

Το PGP μπορεί να χρησιμοποιηθεί με τους υπάρχοντες παρόχους ηλεκτρονικού ταχυδρομείου. Το μόνο που χρειάζεται είναι συνήθως κάποιο πρόσθετο στο πρόγραμμα ηλεκτρονικού ταχυδρομείου του χρήστη ώστε να μπορέσει να λειτουργήσει αυτό το χαρακτηριστικό. Για όσους χρησιμοποιούν το φυλλομετρητή τους για πρόσβαση στο ηλεκτρονικό ταχυδρομείο, υπάρχουν πρόσθετα και για αυτούς.

Τα άμεσα μηνύματα (instant messages) άρχισαν να χρησιμοποιούνται πολύ στα μέσα της δεκαετίας του 1990 όπου έκαναν την εμφάνισή τους τα πρώτα προγράμματα με γραφικό περιβάλλον για υπολογιστή (AIM, ICQ κ.α.). Από τότε η χρήση τους ολοένα και αυξάνεται και η είσοδος των έξυπνων κινητών στην αγορά βοήθησε την εξάπλωσή τους.

Τα προγράμματα άμεσων μηνυμάτων επιτρέπουν την ανταλλαγή σύντομων μηνυμάτων κειμένου σε πραγματικό χρόνο ανάμεσα σε δύο ή περισσότερους χρήστες. Εκτός από κείμενο οι εφαρμογές επιτρέπουν κλήσεις φωνής και εικόνας μέσω διαδικτύου (VoIP) και ανταλλαγή αρχείων.

Οι εφαρμογές που με τους περισσότερους χρήστες παγκοσμίως ανήκουν στην εταιρία Facebook και είναι οι Facebook Messenger και WhatsApp με 1.2 δισεκατομμύρια χρήστες έκαστη. Ακολουθούν το WeChat με 938 εκατομμύρια χρήστες και το Viber με 260 εκατομμύρια.

Όπως και στους παρόχους ηλεκτρονικού ταχυδρομείου, οι πολιτικές απορρήτου για τις ανωτέρω εφαρμογές αναφέρουν τη συλλογή πολλών προσωπικών δεδομένων του χρήστη (όνομα, ηλικία, εργασία, τοποθεσία, πληροφορίες συσκευής κ.α.) με κύριο σκοπό τη βελτίωση των υπηρεσιών. Ομοίως και ο διαμοιρασμός αυτών των δεδομένων είναι εφικτός με τρίτους (κυβερνήσεις, άλλες εταιρίες κλπ.). Σχετικά με το περιεχόμενο των μηνυμάτων, αναφορά στις πολιτικές απορρήτου γίνεται για το Viber και το WhatsApp όπου τα μηνύματα διαγράφονται από τους εξυπηρετητές όταν παραδοθούν στον παραλήπτη ή μετά το πέρας δύο εβδομάδων ή τριάντα ημερών αντίστοιχα αν δεν παραδοθούν. Στη δήλωση για το Viber αναφέρεται επίσης ότι τα δεδομένα του χρήστη δεν πωλούνται σε τρίτους (χωρίς όμως αυτό να σημαίνει ότι δε μοιράζονται σε τρίτους).

Στις εφαρμογές Viber και WhatsApp υπάρχει ενεργοποιημένη η δυνατότητα κρυπτογράφησης από άκρο σε άκρο και μάλιστα για το Viber εξηγούν και την υλοποίησή της [40]. Οι άλλες δύο εφαρμογές αρκούνται στην κρυπτογράφηση της επικοινωνίας μεταξύ εφαρμογής και εξυπηρετητή με το Facebook Messenger να έχει τη δυνατότητα ενεργοποίησης της κρυπτογράφησης από άκρο σε άκρο αν το θέλει ο χρήστης και μόνο για συγκεκριμένες συνομιλίες. Αξίζει να σημειωθεί ότι οι εφαρμογές που δεν υποστηρίζουν από προεπιλογή κρυπτογράφηση από άκρο σε άκρο είναι αυτές που στη δήλωση απορρήτου δεν κάνουν αναφορά στα μηνύματα.

Η εφαρμογή Viber έχει ακόμα δύο σημαντικά χαρακτηριστικά τα οποία βοηθάνε στην προστασία της ιδιωτικότητας. Το πρώτο είναι οι κρυφές συζητήσεις στις οποίες τα μηνύματα έχουν χρόνο αυτοκαταστροφής, προστατεύονται από προώθηση σε άλλους χρήστες και είτε απαγορεύουν τη λήψη στιγμιότυπου της οθόνης είτε προειδοποιούν τον άλλο χρήστη ότι ελήφθη κάποιο στιγμιότυπο. Το δεύτερο χαρακτηριστικό είναι η προστασία κάποιας συζήτησης με έναν τετραψήφιο κωδικό. Ορίζοντας κάποιο κωδικό, η συζήτηση δεν είναι πλέον ορατή στη λίστα με τις συζητήσεις και για την εμφάνισή της απαιτείται η εισαγωγή του κωδικού. Η μόνη άλλη εφαρμογή που περιλαμβάνει χρόνο αυτοκαταστροφής των μηνυμάτων είναι η Facebook Messenger.

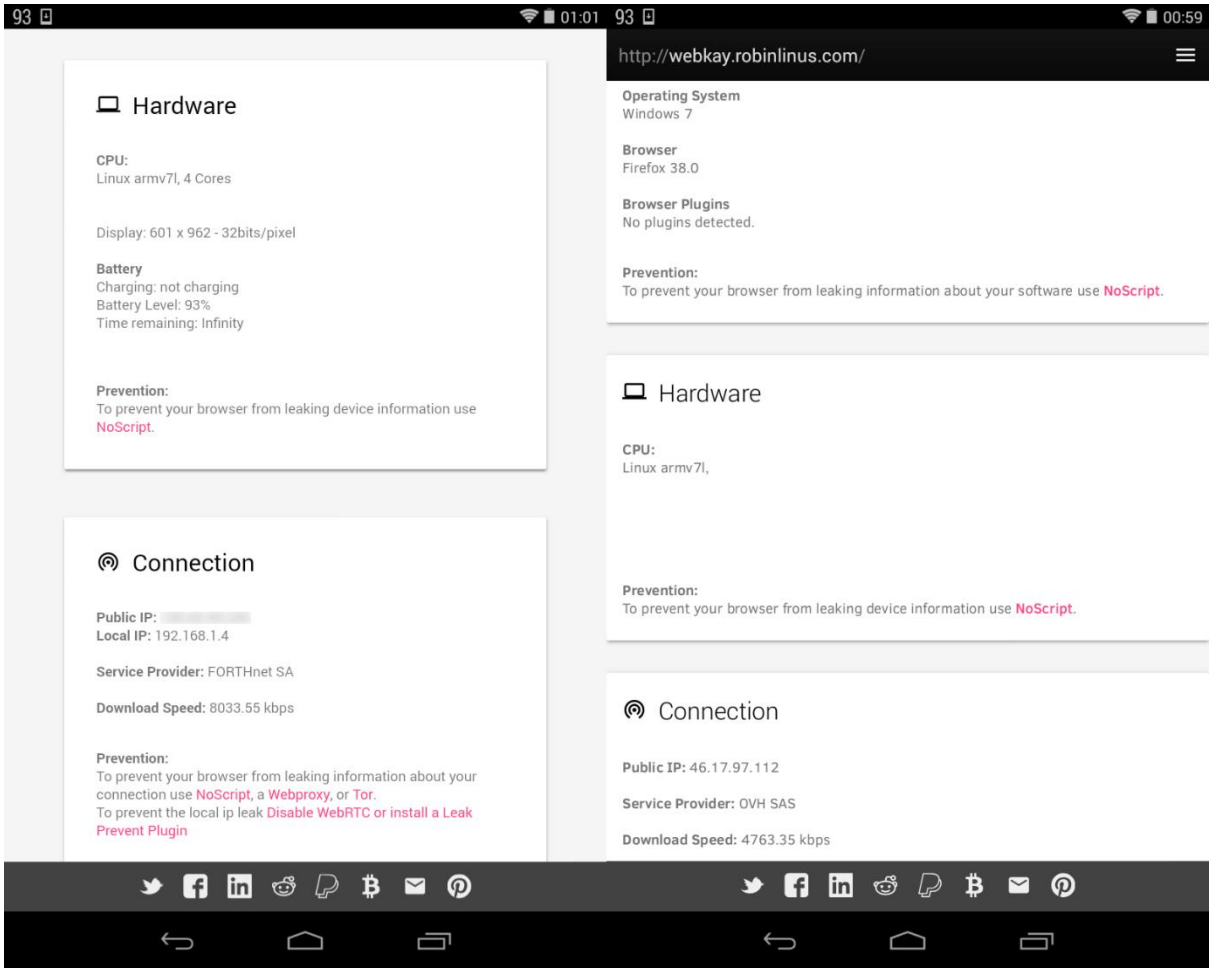
Μία εναλλακτική πλατφόρμα άμεσων μηνυμάτων η οποία δίνει βάση στην προστασία της ιδιωτικότητας και της ασφάλειας είναι η Signal η οποία συνίσταται και από τη διεθνή αμνηστία [41]. Η πλατφόρμα συλλέγει και αποθηκεύει όσες πληροφορίες είναι απαραίτητες για τη λειτουργία της. Οι πληροφορίες αυτές δε μοιράζονται με τρίτους εκτός αν υπάρχει συγκατάθεση του χρήστη ή αν απαιτείται νομικά [42]. Αυτό έχει άλλωστε αποδειχθεί, όταν η υπεύθυνη εταιρία για την πλατφόρμα είχε κληθεί από το δικαστήριο να μοιραστεί πληροφορίες σχετικές με έναν τηλεφωνικό αριθμό και το μόνο που μοιράστηκε ήταν πότε έκανε εγγραφή ο χρήστης στην πλατφόρμα και πότε την χρησιμοποίησε τελευταία φορά [43]. Για την κρυπτογράφηση από άκρο σε άκρο χρησιμοποιεί το πρωτόκολλο Signal, ένα ανοιχτό πρωτόκολλο ανεπτυγμένο από την ίδια εταιρία το οποίο χρησιμοποιούν και οι πλατφόρμες που αναφέρθηκαν προηγουμένως για τον ίδιο λόγο. Υποστηρίζει την ανταλλαγή μηνυμάτων κειμένου, αρχείων και κλήσεις ήχου και εικόνας όλα με κρυπτογράφηση. Επίσης υπάρχει η δυνατότητα ορισμού ορίου για την αυτοκαταστροφή των μηνυμάτων, παρεμπόδιση λήψης στιγμιοτύπου της οθόνης και κλείδωμα της εφαρμογής με κωδικό. Δεν υπάρχουν επίσημα στοιχεία σχετικά με την ποσότητα των χρηστών αλλά κρίνοντας από τις εγκαταστάσεις που έχει η εφαρμογή στο Android (5-10 εκατομμύρια), οι χρήστες είναι λίγοι συγκριτικά με τις άλλες εφαρμογές άμεσων μηνυμάτων.

Η πλοήγηση στο διαδίκτυο είναι από τους βασικούς λόγους χρήσης των έξυπνων τηλεφώνων και ταμπλετών. Για την πλοήγηση χρησιμοποιείται κάποιος φυλλομετρητής. Οι φυλλομετρητές που κυριαρχούν τον τελευταίο χρόνο στα κινητά τηλέφωνα και τις ταμπλέτες είναι ο Chrome της Google και ο Safari της Apple κατέχοντας σχεδόν το 90% του μεριδίου αγοράς [44].

Όταν ένας χρήστης επισκέπτεται μία ιστοσελίδα, μέσω του φυλλομετρητή είναι δυνατόν να αποκτήσει πρόσβαση σε πολλές πληροφορίες όπως πληροφορίες για το υλικό μιας συσκευής (αρχιτεκτονική επεξεργαστή, ποσοστό μπαταρίας κ.α.), για το λογισμικό (λειτουργικό σύστημα, εταιρία φυλλομετρητή κλπ.), πληροφορίες για τη σύνδεση, τοποθεσία. Εκτός από τις ιστοσελίδες και οι ίδιοι οι φυλλομετρητές συλλέγουν δεδομένα για το χρήστη. Η Google έχει κοινή πολιτική απορρήτου για όλες τις υπηρεσίες και τα προϊόντα της και έγινε αναφορά σε προηγούμενη ενότητα για τη δήλωση αυτή. Υπενθυμίζεται ότι κάνοντας χρήση των προγραμμάτων και των υπηρεσιών της, έχει δικαίωμα συλλογής πολλών ειδών δεδομένων. Και με την Apple τα πράγματα δεν έχουν μεγάλη διαφορά. Η εταιρία έχει το δικαίωμα να συλλέξει και να αποθηκεύσει οποιαδήποτε πληροφορία έχει τη δυνατότητα με χρήση των προϊόντων της και να τα μοιραστεί με τρίτους χωρίς τη συγκατάθεση του χρήστη [45].

Ευτυχώς η παρεμπόδιση συλλογής των πληροφοριών δεν είναι δύσκολη. Η πιο εύκολη λύση είναι η χρήση κάποιου φυλλομετρητή που δίνει βάση στην ιδιωτικότητα και είναι ανοιχτού κώδικα. Ένα τέτοιο παράδειγμα είναι ο Firefox Focus με μότο το “Πλοηγήσου σαν να μη βλέπει κανένας”. Για αυτούς που θέλουν περισσότερη ανωνυμία στο διαδίκτυο, η χρήση του Tor είναι απαραίτητη. Για τη χρήση του σε κινητά τηλέφωνα και ταμπλέτες αρκεί ο χρήστης να εγκαταστήσει έναν άλλο φυλλομετρητή και πλέον μπορεί να πλοηγηθεί με περισσότερη ανωνυμία.

Η ακόλουθη εικόνα δείχνει διαφορές στα δεδομένα που μπορούν να συλλεχθούν με τη χρήση Tor και χωρίς αυτή. Η δοκιμή έγινε με επίσκεψη σε μία ιστοσελίδα σχεδιασμένη για να δείχνει τι πληροφορίες είναι προσβάσιμες σε ένα φυλλομετρητή [46], χρησιμοποιώντας μία ταμπλέτα Nexus 7 [47] με λειτουργικό Android 4.4. Στο αριστερό μέρος της εικόνας φαίνεται η επίσκεψη στη σελίδα από το φυλλομετρητή Google Chrome. Οι πληροφορίες που δίνει ο Chrome σχετικά με τη συσκευή επαληθεύονται από τα χαρακτηριστικά της. Επίσης το επίπεδο της μπαταρίας επαληθεύεται από την ένδειξη στο πάνω αριστερό μέρος της οθόνης. Σχετικά με τις πληροφορίες της σύνδεσης, όχι μόνο φαίνεται η διεύθυνση IP που έχει δοθεί στο δρομολογητή από τον πάροχο (κρυμμένη στην εικόνα για λόγους ιδιωτικότητας) αλλά φαίνεται και η διεύθυνση IP που έχει πάρει η συσκευή μέσα στο ιδιωτικό δίκτυο. Γνωρίζοντας τη διεύθυνση IP είναι εύκολη η εύρεση του παρόχου σύνδεσης στο διαδίκτυο. Η ταχύτητα σύνδεσης στο διαδίκτυο που φαίνεται είναι αρκετά κοντά στην πραγματική. Στο δεξί μέρος της εικόνας η πρόσβαση έχει γίνει χρησιμοποιώντας Tor και έναν άλλο φυλλομετρητή. Ο τελευταίος αλλοιώνει τα δεδομένα σχετικά με τη συσκευή και το λογισμικό για διατήρηση της ανωνυμίας του χρήστη. Ο φυλλομετρητής στέλνει στην ιστοσελίδα ότι η συσκευή έχει Windows 7 και ο φυλλομετρητής είναι ο Firefox. Η αρχιτεκτονική της συσκευής έχει παραμείνει ίδια αλλά δεν αποκαλύπτονται άλλες πληροφορίες για τα χαρακτηριστικά της. Τέλος, η διεύθυνση IP και ο πάροχος που καταλαβαίνει η ιστοσελίδα είναι τελείως διαφορετική προερχόμενη από την Ολλανδία.



Εικόνα 4.1 Σύγκριση πλοήγησης χωρίς και με χρήση του Tor

Φυσικά υπάρχουν και τα αρνητικά σε αυτό. Η πλοήγηση είναι πιο αργή απ' ότι συνήθως επειδή η πληροφορία κρυπτογραφείται πολλές φορές και περνάει μέσα από περισσότερους εξυπηρετητές. Πολλές φορές τα αρχεία πολυμέσων (εικόνα, βίντεο κλπ.) παρακάμπτουν το Tor με αποτέλεσμα να χάνεται η ανωνυμία.

Για την αποφυγή της αργής ταχύτητας, είναι εφικτή η σύνδεση και σε κάποιο ΕΙΔ. Οι πάροχοι ΕΙΔ προσφέρουν ειδικές εφαρμογές για τη σύνδεση των κινητών συσκευών με τους εξυπηρετητές που συνεργάζονται. Τόσο το Android όσο και το iOS προσφέρουν τη δυνατότητα προσθήκης κάποιο παρόχου ΕΙΔ χωρίς τη χρήση ειδικής εφαρμογής μέσω των ρυθμίσεων του λειτουργικού συστήματος. Η ενεργοποίηση και απενεργοποίηση του ΕΙΔ γίνεται συνήθως με το απλό άγγιγμα ενός κουμπιού. Ενεργοποιώντας τη σήραγγα με κάποιον εξυπηρετητή ΕΙΔ, ολόκληρη η επικοινωνία της συσκευής από το διαδίκτυο γίνεται μέσω του εξυπηρετητή. Με αυτό τον τρόπο δεν προφυλάσσεται ο χρήστης μόνο από τις ιστοσελίδες που επισκέπτεται αλλά και από άλλες κακόβουλες ενέργειες όπως η αποκάλυψη της διεύθυνσης IP του χρήστη όταν κάνει κάποια κλήση μέσω μίας VoIP εφαρμογής (π.χ. Skype).

4.2 Προστασία δεδομένων από παραβιάσεις των εφαρμογών

Εκτός από τα δεδομένα που στέλνει ο χρήστης μέσω του διαδικτύου στα μηνύματα ηλεκτρονικού ταχυδρομείου, στα άμεσα μηνύματα και από την πλοήγηση, οι εφαρμογές που υπάρχουν στις κινητές συσκευές μπορούν να έχουν πρόσβαση σε ορισμένα από τα δεδομένα των συσκευών αυτών. Για τον περιορισμό των δεδομένων τα λειτουργικά συστήματα Android και iOS εφαρμόζουν κάποιες τεχνικές: ο αμμόλακκος (sandbox), η επιθεώρηση πριν τη δημοσίευση κάποιας εφαρμογής και ο περιορισμός στη Διεπαφή Ανάπτυξης Εφαρμογής (Application Programming Interface - API).

Ο αμμόλακκος είναι ένας μηχανισμός ο οποίος επιτρέπει σε κάθε εφαρμογή να εκτελείται σε ένα απομονωμένο περιβάλλον με ορισμένους περιορισμούς. Με αυτό τον τρόπο αποφεύγεται η αλληλεπίδραση μεταξύ των εφαρμογών αλλά και των πόρων του λειτουργικού συστήματος εκτός αυτών που τους έχει επιτραπεί. Χαρακτηριστικό παράδειγμα είναι ότι οι εφαρμογές δεν έχουν πρόσβαση σε όλο τον αποθηκευτικό χώρο της συσκευής αλλά μόνο σε αυτόν του περιβάλλοντός τους, εκτός αν επιτραπεί με άλλο τρόπο.

Οι εφαρμογές πριν δημοσιευτούν σε κάποιο αποθετήριο, περνάνε πρώτα από επιθεώρηση ώστε να ελεγχθεί η τήρηση των κατευθυντήριων γραμμών που ορίζει κάθε ένα. Παραβιάσεις στις πολιτικές, κακόβουλες εφαρμογές, εφαρμογές που κλέβουν τα δεδομένα του χρήστη δε φτάνουν ποτέ στο στάδιο της δημοσίευσης και ο χρήστης προστατεύεται. Ο τρόπος επιθεώρησης διαφέρει ανάμεσα στη Google και την Apple με την τελευταία να κάνει εξ' ολοκλήρου την επιθεώρηση από ανθρώπους. Η Google μέχρι και το 2015 έκανε τις επιθεωρήσεις μέσω αυτοματοποιημένων διαδικασιών με λογισμικό. Από τότε όμως άρχισε να εφαρμόζει και ανθρώπινη επιθεώρηση [48].

Η Διεπαφή Ανάπτυξης Εφαρμογής περιορίζει την κάθε εφαρμογή στα δεδομένα του χρήστη που μπορεί να έχει πρόσβαση. Από την άλλη, η ΔΑΕ μπορεί να δώσει δικαιώματα σε προσωπικά στοιχεία του χρήστη αλλά αυτό συνήθως γίνεται με τη συγκατάθεσή του. Για την πρόσβαση σε αυτά τα δεδομένα απαιτείται ο χρήστης να δώσει δικαίωμα στην εφαρμογή. Στο Android υπάρχουν δύο κατηγορίες δικαιωμάτων [49]:

- Τα κανονικά δικαιώματα στα οποία υπάρχει μικρός κίνδυνος παραβίασης της ιδιωτικότητα τους χρήστη και
- Τα επικίνδυνα δικαιώματα τα οποία περιλαμβάνουν πρόσβαση σε ιδιωτικά δεδομένα του χρήστη ή επηρεάζουν τα αρχεία του.

Από το Android 6.0 και μετά, ο χρήστης ερωτάται ξεχωριστά για κάθε ένα επικίνδυνο δικαίωμα που ζητάει η εφαρμογή και μπορεί να επιτρέψει ή όχι κάποιο αίτημα. Επίσης υπάρχει η δυνατότητα εκ των υστέρων άρνησης ή παροχής κάποιου επικίνδυνου δικαιώματος. Μέχρι πρότινος πριν την εγκατάσταση κάθε εφαρμογής παρουσιαζόταν στο χρήστη η λίστα με τα δικαιώματα που ήθελε κάθε εφαρμογή και μόνο αν παρείχε σε όλα τα δικαιώματα την πρόσβαση γινόταν εγκατάσταση της εφαρμογής. Στην

περίπτωση του iOS, η κάθε εφαρμογή πρέπει να ζητήσει δικαίωμα για πρόσβαση στα δεδομένα του χρήστη με τη διαφορά ότι ο χρήστης ερωτάται για κάθε δικαίωμα ξεχωριστά πριν η εφαρμογή χρειαστεί να αποκτήσει πρόσβαση σε κάποια δεδομένα. Ο πίνακας που ακολουθεί βασίζεται στον Kitchin [50] και έχει βελτιωθεί σύμφωνα με τα δικαιώματα που μπορεί να ζητήσει μία εφαρμογή για να αποκτήσει πρόσβαση σε προσωπικά δεδομένα [49], [51], [52].

Πίνακας 4.1 Διαθέσιμα δεδομένα σε έναν προγραμματιστή από τη ΔΑΕ

Τύπος δεδομένων	Ζητούμενες άδειες δεδομένων
Λογαριασμοί	Πρόσβαση σε λογαριασμούς ρύθμισης της συσκευής.
Εφαρμογές	UID, χρόνος εγκατάστασης, GIDs, τοποθεσία εγκατάστασης, τελευταία αναβάθμιση, άδειες, εκδοχή, εκκαθάριση μνήμης cache
Bluetooth	Ενεργοποίηση/ Απενεργοποίηση, διεύθυνση, ανάγνωση/ αλλαγή ονόματος, πληροφορίες συζευγμένων συσκευών (διεύθυνση, όνομα, τύπος), πληροφορίες συνδεδεμένων συσκευών, σύνδεση/ αποσύνδεση από/προς μια συσκευή.
Ημερολόγιο	Δυνατότητα πρόσβασης ανάγνωσης/εγγραφής σε δεδομένα του ημερολογίου του χρήστη.
Κάμερα	Λήψη φωτογραφιών και βίντεο.
Πληροφορίες Δικτύου Κυψέλης	Ταυτότητα συσκευές (IMEI/ MEID), τηλεφωνικός αριθμός, SIMID, IMSI, χώρα κυψελοειδούς δικτύου (IMCC), πάροχος κυψελοειδούς δικτύου (MCC+MNC), όνομα χειριστή κυψελοειδούς δικτύου, τύπος δικτύου, τύπος τηλεφώνου, ISO χώρας SIM, πάροχος SIM (MCC+MNC), όνομα παρόχου SIM, κατάσταση περιαγωγής
Επαφές	Ανάγνωση/Εγγραφή επαφών συσκευής.
Πληροφορίες Συσκευής	Πλακέτα, μάρκα, έκδοση κατασκευής, συσκευή, τύπος συσκευής, οθόνη, διεύθυνση MAC, κατασκευαστής, μοντέλο, πλατφόρμα λειτουργικού συστήματος, προϊόν, έκδοση SDK, επιτρεπόμενες άγνωστες πηγές.
Αρχεία	Πρόσβαση στα αρχεία συσκευής, διαμόρφωση αρχείων συστήματος, εισαγωγή/εξαγωγή αρχείων συστήματος, συνολικός χώρος δίσκου.
Δαχτυλικό Αποτύπωμα	Το δαχτυλικό αποτύπωμα του χρήστη.
Εργαλεία Υγείας	Ημερομηνία γέννησης, τύπος αίματος, φύλο, τύπος δέρματος, καρδιακοί παλμοί, βάρος, ύψος.
Τοποθεσία	Πρόσβαση σε δεδομένα GPS (ακρίβεια, γεωγραφικό πλάτος, μήκος, υψόμετρο, ταχύτητα), κατά προσέγγιση θέση με βάση την κυψελοειδή

	λήψη ή/και τη σύνδεση wifi.
Μικρόφωνο	Καταγραφή ήχου.
Δίκτυο	Υποστηριζόμενες συνδέσεις δικτύου (π.χ. WiFi, 3G, 4G), ενεργή σύνδεση δικτύου, αλλαγή ενεργού δικτύου
Τηλεφωνικές Κλήσεις	Ανάγνωση/Εγγραφή αρχείων κλήσεων, απευθείας κλήση αριθμού, προεπισκόπηση του κληθέντος αριθμού και ανακατεύθυνση/ακύρωση της κλήσης
Αισθητήρες	Πρόσβαση δεδομένων από οποιαδήποτε διαθέσιμο αισθητήρα μέσα στην συσκευή (π.χ. επιταχυνσιόμετρο, βαρόμετρο)
SMS/MMS	Αποστολή και λήψη δεδομένων SMS/MMS, ανάγνωση δεδομένων και αριθμών SMS/MMS
Συγχρονισμός	Ανάγνωση στατιστικών συγχρονισμού για έναν λογαριασμό (ιστορικό γεγονότων συγχρονισμού, ποσό συγχρονισμένων δεδομένων), έλεγχος εάν μια εφαρμογή είναι συγχρονισμένη με έναν λογαριασμό, συγχρονισμός μιας εφαρμογής με έναν λογαριασμό
WiFi	Λίστα αποθηκευμένων δικτύων (SSID, συχνότητα, συνθηματικό), ενεργές πληροφορίες δικτύου wifi (SSID, συχνότητα, IP, ταχύτητα, MAC, RSSI, BSSID), σάρωση δικτύων wifi (BSSID, SSID, αυθεντικότητα, οργάνωση κλειδιών, κρυπτογράφηση, εύρος καναλιών, συχνότητα, επίπεδο σήματος), σύνδεση/αποσύνδεση από/προς wifi

Από τον παραπάνω πίνακα μπορεί να γίνει μία προσπάθεια κατηγοριοποίησης των δεδομένων που μπορούν να αποκτηθούν από μία εφαρμογή ως εξής:

1. Δεδομένα που υπάρχουν στη συσκευή
2. Δεδομένα που έχει εισάγει ο χρήστης
3. Τεχνικά δεδομένα
4. Δεδομένα αισθητήρων

Δεδομένα που υπάρχουν στη συσκευή

Αυτά τα δεδομένα περιλαμβάνουν πληροφορίες που έχει εισάγει ο χρήστης στη συσκευή με τη χρήση άλλων εφαρμογών. Τέτοια είναι οι λογαριασμοί που έχει συνδέσει ο χρήστης με τη συσκευή του (ηλεκτρονικού ταχυδρομείου, κοινωνικής δικτύωσης κλπ.), δεδομένα από το ημερολόγιο του χρήστη, τις επαφές του, ιστορικό των τηλεφωνικών κλήσεων, τα SMS και MMS που υπάρχουν αποθηκευμένα και τέλος τα αρχεία που έχει αποθηκεύσει ο χρήστης στη συσκευή του.

Δεδομένα που εισάγει ο χρήστης

Οι εφαρμογές μπορούν να αποκτήσουν πληθώρα δεδομένων εφόσον οι χρήστες είναι διαθέσιμοι να αποκαλύψουν. Ιδιαίτερα στις εφαρμογές που επιτρέπουν τη δημιουργία προφίλ του χρήστη, δημιουργούνται πολλά δεδομένα. Τα δεδομένα αυτά συχνά περιλαμβάνουν ημερομηνία γέννησης, ονοματεπώνυμο, διεύθυνση ηλεκτρονικού ταχυδρομείου, φύλλο, αριθμό τηλεφώνου, εικόνα προφίλ και λογαριασμούς κοινωνικής δικτύωσης (π.χ. Facebook , Google+).

Τεχνικά δεδομένα

Τα τεχνικά δεδομένα περιέχουν κάθε είδους τεχνική πληροφορία σχετικά με τη συσκευή. Η τεχνική πληροφορία μπορεί να σχετίζεται με τον πάροχο κινητής τηλεφωνίας, όπου υπάρχει (όνομα, κωδικός χώρας, κωδικός παρόχου κ.α.). Επίσης μπορεί να περιέχονται πληροφορίες που αφορούν τη συσκευή όπως η μάρκα, το μοντέλο, ο τύπος συσκευής, η έκδοση του λειτουργικού συστήματος και το όνομα της συσκευής. Μπορούν ακόμα να εξαχθούν πληροφορίες σχετικές με το δίκτυο (τύπος δικτύου, διεύθυνση IP, αποθηκευμένα δίκτυα, δίκτυα εντός εμβέλειας κλπ.).

Δεδομένα αισθητήρων

Σχεδόν όλες οι κινητές συσκευές απαρτίζονται από μία πληθώρα αισθητήρων. Οι συνηθέστεροι αισθητήρες είναι το γυροσκόπιο και ο αισθητήρας επιτάχυνσης. Συνδυάζοντας αυτά τα δεδομένα με συγκεκριμένους αλγορίθμους μπορεί εύκολα να εξαχθεί συμπέρασμα για την κίνηση του χρήστη. Κάποιες συσκευές μπορούν να περιέχουν βιομετρικούς αισθητήρες όπως ο μετρητής καρδιακών παλμών και ο αισθητήρας δαχτυλικού αποτυπώματος. Δύο ακόμα πολύ συχνά χρησιμοποιούμενοι αισθητήρες είναι αυτός του μικροφώνου και της κάμερας. Αποκτώντας πρόσβαση σε αυτούς, μία εφαρμογή μπορεί να δει και να ακούσει το χρήστη.

Παρόλο που η Google και η Apple επιθεωρούν τις εφαρμογές και φροντίζουν ώστε να μην υπάρχουν εφαρμογές που κάνουν υπερβολές [53], αρκετές από τις δημοφιλείς εφαρμογές στέλνουν τα δεδομένα του χρήστη σε τρίτους [54]. Θα πρέπει λοιπόν οι χρήστες πριν την εγκατάσταση κάποιας εφαρμογής να δίνουν σημασία στα δικαιώματα που μπορεί να ζητηθούν και να είναι υποψιασμένοι. Αν για παράδειγμα μία εφαρμογή η οποία σαρώνει το χώρο για ασύρματα σημεία πρόσβασης και δίνει πληροφορίες για αυτά ζητήσει πρόσβαση στις επαφές του χρήστη, έχει και άλλους σκοπούς. Σε αυτή την περίπτωση η εφαρμογή είναι καλό να αντικατασταθεί από μία άλλη η οποία δε ζητάει παράλογα δικαιώματα. Επίσης χρησιμοποιώντας κανείς εφαρμογές ανοιχτού κώδικα μπορεί να είναι πιο ήσυχος για τα δεδομένα του αφού ο κώδικας των συγκεκριμένων εφαρμογών μπορεί να ελεγχθεί από τον καθένα (κάτι το οποίο γίνεται) και οποιαδήποτε υποψία για κοινοποίηση δεδομένων σε τρίτους γίνεται γνωστή. Τέλος, η αιτιολόγηση της ζήτησης δικαιωμάτων από τον προγραμματιστή της εφαρμογής σε κάποιο εμφανές

σημείο θα μπορούσε να βοηθήσει στην καλύτερη κρίση μίας εφαρμογής. Αυτό όμως γίνεται σπάνια διότι οι εφαρμογές στέλνουν δεδομένα σε τρίτους και αν το δηλώσουν θα χάσουν σε αριθμό εγκαταστάσεων.

4.3 Προστασία δεδομένων από φυσικές παραβιάσεις της συσκευής

Λαμβάνοντας τα παραπάνω μέτρα, ο όγκος των προσωπικών δεδομένων που κοινοποιείται σε τρίτους σίγουρα ελαττώνεται. Προσωπικά δεδομένα όμως μπορούν να διαρρεύσουν με το να αποκτήσει κάποιος τρίτος φυσική πρόσβαση στην κινητή συσκευή. Με την εφαρμογή των παρακάτω μέτρων, η πιθανότητα πρόσβασης μειώνεται.

Χρήση κωδικού κλειδώματος

Μπορεί να είναι προφανές αλλά σύμφωνα με μελέτες 1 στους 3 χρήστες δεν ασφαλίζει τη συσκευή του με κάποιου είδους κλείδωμα [55]. Αποφεύγοντας τη χρήση κλειδώματος, όποιος πάρει στα χέρια του τη συσκευή μπορεί πολύ εύκολα να αποκτήσει πρόσβαση στα περιεχόμενά της. Το κλείδωμα μπορεί να είναι ένα PIN, ένας κωδικός ή ακόμα και ένα μοτίβο. Υπάρχουν βέβαια ισχυρισμοί ότι το PIN είναι ασφαλέστερο από το μοτίβο [56]. Φυσικά η χρήση εύκολων PIN όπως “1234” ή τα γενέθλια του κατόχου δε δυσκολεύουν πολύ τον επίδοξο εισβολέα.

Κρυπτογράφηση της συσκευής

Η χρήση κωδικού για το ξεκλείδωμα της συσκευής είναι ένα πρώτο μέτρο για την προστασία της, αλλά δοθέντος αρκετού χρόνου μπορεί να παρακαμφθεί. Κρυπτογραφώντας τη συσκευή και τα δεδομένα της, η αποκρυπτογράφηση αυτών γίνεται μόνο με κωδικό ο οποίος δεν είναι εύκολο να παραβιαστεί. Το αρνητικό στην κρυπτογράφηση της συσκευής είναι ότι είναι μία χρονοβόρα διαδικασία αν και με την ισχύ των σύγχρονων επεξεργαστών πολλές φορές ο χρόνος είναι αμελητέος.

Εύρεση συσκευής και απομακρυσμένη διαγραφή δεδομένων

Σε περίπτωση απώλειας ή κλοπής της συσκευής, υπάρχει η δυνατότητα εντοπισμού της ώστε να ανακτηθεί. Αν η ανάκτηση είναι δύσκολη, η δυνατότητα διαγραφής των δεδομένων της απομακρυσμένα, προσφέρει ένα ακόμα επίπεδο ασφάλειας. Ακόμα κι αν κάποιος καταφέρει και παραβιάσει τον κωδικό κλειδώματος, διαγράφοντας τα δεδομένα που υπάρχουν καθιστά τη συσκευή άχρηστη. Τα μειονεκτήματα που μπορεί να υπάρξουν σε αυτό είναι ότι η διαγραφή γίνεται με τη χρήση συγκεκριμένης εφαρμογής όπου ο χρήστης δε μπορεί να είναι σίγουρος αν τα δεδομένα αποστέλλονται σε τρίτους καθώς και ότι η συσκευή πρέπει να είναι συνδεδεμένη στο διαδίκτυο.

4.4 Προστασία στην επικοινωνία μεταξύ φορετών και φορητών συσκευών

Η επικοινωνία των φορετών συσκευών με το κινητό τηλέφωνο ή την ταμπλέτα γίνεται με τη χρήση του πρωτοκόλλου Bluetooth. Παρόλο που το Bluetooth έχει μικρή εμβέλεια, μία επίθεση με στόχο τα δεδομένα που ανταλλάσσονται δεν είναι καθόλου απίθανη, ιδιαίτερα σε δημόσιους χώρους. Έχοντας αυτό υπόψη, από την έκδοση Bluetooth 2.1 και μετά έγιναν σημαντικές αλλαγές στην ασφάλεια.

Χρησιμοποιώντας αλγορίθμους οι οποίοι βασίζονται στον αλγόριθμο SAFER+, το Bluetooth υποστηρίζει εμπιστευτικότητα, πιστοποίηση και παραγωγή κλειδιών. Ο αλγόριθμος SAFER+ ήταν υποψήφιος για τη θέση του AES αλλά αποκλείστηκε επειδή ήταν πολύ αργός σε σχέση με τους άλλους. Η παραγωγή κλειδιών βασίζεται σε ένα PIN το οποίο πρέπει να εισαχθεί και στις δύο συσκευές. Πολλές φορές αυτό το PIN είναι προκαθορισμένο στη μία συσκευή λόγω έλλειψης διεπαφής εισόδου. Κατά τη σύζευξη των συσκευών δημιουργείται ένα μυστικό κλειδί το οποίο χρησιμοποιείται για την κρυπτογράφηση των δεδομένων. Η δημιουργία του κλειδιού γίνεται με τη χρήση του αλγορίθμου E22. Τέλος, ο κρυπταλγόριθμος E0 αναλαμβάνει την κρυπτογράφηση των πακέτων και την παροχή εμπιστευτικότητας.

Για διευκόλυνση των προγραμματιστών τόσο η Google όσο και η Apple προσφέρουν ΔΑΕ για την επικοινωνία των έξυπνων ρολογιών με τις κινητές συσκευές. Το πρόβλημα με αυτό είναι ότι στο υψηλότερο επίπεδο το κανάλι επικοινωνίας ελέγχεται από αυτές τις εταιρίες και σύμφωνα με τις πολιτικές απορρήτου που αναλύθηκαν παραπάνω, ο όγκος των πληροφοριών που συλλέγουν είναι μεγάλος. Επομένως δεν είναι δυνατό να γνωρίζει κανείς αν τα δεδομένα που περνάνε μέσα από αυτό το κανάλι δε συλλέγονται και κοινοποιούνται από αυτές τις εταιρίες. Ευτυχώς το λειτουργικό σύστημα αυτών των εταιριών δίνει πρόσβαση στη ΔΑΕ του Bluetooth και μπορεί ένας προγραμματιστής να αναπτύξει τη δική του βιβλιοθήκη.

Αρκετές εταιρείες που κατασκευάζουν καταγραφείς δραστηριότητας ή άλλες φορετές συσκευές προσφέρουν πρόσβαση στη ΔΑΕ για την ανάπτυξη εφαρμογών εκτός των επισήμων. Ορισμένες προσφέρουν πρόσβαση στο επίπεδο του Bluetooth [57], άλλες με χρήση έτοιμων βιβλιοθηκών δίνουν πρόσβαση απευθείας στα δεδομένα που καταγράφουν οι συσκευές [58], ενώ κάποιες δίνουν πρόσβαση στα δεδομένα αφού αυτά αποσταλούν στους εξυπηρετητές της εταιρίας.

Για την ασφαλή επικοινωνία των φορετών συσκευών με τις κινητές, είναι απαραίτητη η υλοποίηση βιβλιοθηκών από τους προγραμματιστές οι οποίες θα κάνουν χρήση της ΔΑΕ του λειτουργικού συστήματος. Με αυτό τον τρόπο εξασφαλίζεται η ασφάλεια στην επικοινωνία και η βεβαιότητα ότι τα ευαίσθητα δεδομένα που παράγονται από τις φορετές συσκευές δεν καταλήγουν σε τρίτους.

ΚΕΦΑΛΑΙΟ 5:
Πρόταση και υλοποίηση αρχιτεκτονικής προστασίας ιδιωτικότητας

5.1 Απαιτήσεις και προδιαγραφές

Ο μεγάλος όγκος των ευαίσθητων δεδομένων που υπάρχουν και ανταλλάσσονται στις κινητές και φορητές συσκευές έχει οδηγήσει στην αύξηση της προσοχής σε θέματα ιδιωτικότητας. Για την αντιμετώπιση αυτού του ευαίσθητου ζητήματος, στη συνέχεια του κεφαλαίου παρουσιάζεται μία αρχιτεκτονική η οποία εξυπηρετεί αυτό το σκοπό. Όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο της εργασίας, η ιδιωτικότητα του χρήστη μπορεί να προστατευθεί με διάφορους τρόπους. Ένας από αυτούς τους τρόπους είναι η χρήση της ανωνυμίας, δηλαδή της αφαίρεσης δεδομένων που προσδιορίζουν το χρήστη από την πληροφορία. Είναι βασικό να μπορεί ο χρήστης να εμπιστευθεί ένα σύστημα το οποίο του προσφέρει ανωνυμία ώστε να μη διστάσει να το χρησιμοποιήσει. Υπάρχουν όμως και περιπτώσεις όπου γίνεται κατάχρηση της δυνατότητας αυτής με καταστροφικές σε ορισμένες περιπτώσεις συνέπειες. Έτσι λοιπόν, πρέπει να υπάρχει η δυνατότητα άρσης της ανωνυμίας ώστε να είναι δυνατός ο εντοπισμός του ατόμου που κάνει κατάχρηση. Η δυσκολία έγκειται στην εξισορρόπηση της δυνατότητας ανωνυμίας και της δυνατότητας άρσης της καθώς αν δοθεί περισσότερη βάση στο πρώτο, αυξάνονται οι πιθανότητες κατάχρησης ενώ αν δοθεί περισσότερη βάση στο δεύτερο, θα είναι λιγότερο εύκολο για τους χρήστες να εμπιστευθούν ένα τέτοιο σύστημα.

5.2 Αρχιτεκτονική και λειτουργία

Η λύση εμπιστευτικότητας που παρουσιάζεται σε αυτή την εργασία αποτελείται από μία διαδικασία προστασίας της ιδιωτικότητας σε δύο στάδια. Κάθε στάδιο περιλαμβάνει έναν εξυπηρετητή με μοναδικό ρόλο στη διαδικασία. Ο πρώτος εξυπηρετητής ο οποίος χρησιμοποιείται για να αποκρύψει την ταυτότητα του χρήστη είναι ο ***Εξυπηρετητής Ανωνυμίας – ΕΑν (Anonymiser)***. Ο δεύτερος είναι ο ***Εξυπηρετητής Αποκρυπτογράφησης - ΕΑπ (Decrypter)*** και έχει το ρόλο της υποστήριξης κρυπτογράφησης από άκρο σε άκρο. Υπηρεσίες προώθησης ειδοποίησης (push notifications) χρησιμοποιούνται από τον πρώτο εξυπηρετητή ώστε να είναι δυνατή η απάντηση στο χρήστη. Τα δεδομένα που θέλει να στείλει ο χρήστης, θα σταλούν από

την κινητή ή φορετή συσκευή του και θα περάσουν από τους δύο εξυπηρετητές ώστε να καταλήξουν στον τελικό προορισμό.

Εικόνα 5.1, Διάγραμμα ροής αποστολής δεδομένων από το χρήστη στον Εξυπηρετητή Ανωνυμίας

Ο πρώτος εξυπηρετητής δέχεται ένα μήνυμα από την εφαρμογή που έχει εγκατεστημένη ο χρήστης στην κινητή του συσκευή. Το μήνυμα περιλαμβάνει το δημόσιο κλειδί του χρήστη, ένα συμμετρικό κλειδί, μία υπογραφή και τα κρυπτογραφημένα δεδομένα που θέλει ο χρήστης να στείλει. Προαιρετικά μπορεί να περιέχει μία λεκτική μονάδα (token) η οποία παρέχεται από την υπηρεσία ανταλλαγής μηνυμάτων μέσω του νέφους και ένα αναγνωριστικό συνεδρίας (Session ID) τα οποία χρησιμοποιούνται για την αποστολή απάντησης στο χρήστη. Η κρυπτογράφηση των δεδομένων επιτυγχάνεται χρησιμοποιώντας έναν αλγόριθμο συμμετρικού κλειδιού. Το συμμετρικό κλειδί για την κρυπτογράφηση δημιουργείται πριν την αποστολή των δεδομένων και είναι μοναδικό για κάθε συνεδρία. Το κλειδί αυτό χρησιμοποιείται για κάθε επόμενο μήνυμα. Επίσης κρυπτογραφείται χρησιμοποιώντας το δημόσιο κλειδί του εξυπηρετητή αποκρυπτογράφησης. Αυτή είναι μία κοινή μέθοδος για κρυπτογράφηση δεδομένων όπου το μέγεθός τους είναι μεγαλύτερο από το μήκος του δημοσίου κλειδιού και

χρησιμοποιείται συχνά σε κρυπτογραφικά πρωτόκολλα όπως το SSL/TLS και το PGP. Χρησιμοποιώντας αυτή τη μέθοδο, μόνο ο τελευταίος εξυπηρετητής είναι σε θέση να αποκρυπτογραφήσει το συμμετρικό κλειδί ώστε τελικά να αποκρυπτογραφήσει τα δεδομένα του χρήστη.
Μετά την κρυπτογράφηση δημιουργείται μία ψηφιακή υπογραφή με τη χρήση μίας συνάρτησης κατακερματισμού η οποία τροφοδοτείται με τα κρυπτογραφημένα δεδομένα του χρήστη και το ιδιωτικό του κλειδί. Ύστερα η υπογραφή ενσωματώνεται στο μήνυμα. Το ζεύγος κλειδιών του χρήστη δημιουργείται την πρώτη φορά που εκείνος ανοίγει την εφαρμογή. Τέλος, το δημόσιο κλειδί του ενσωματώνεται επίσης στο μήνυμα ώστε να είναι δυνατή η πιστοποίηση των δεδομένων στον εξυπηρετητή ανωνυμίας.
Κατά την άφιξη του μηνύματος στον πρώτο εξυπηρετητή, πρώτα ελέγχεται η εγκυρότητα των δεδομένων χρησιμοποιώντας το δημόσιο κλειδί του χρήστη και τα κρυπτογραφημένα δεδομένα. Ο σκοπός του ελέγχου είναι να αποφευχθεί η αλλοίωση των δεδομένων από κάποιον κακόβουλο. Ύστερα, χρησιμοποιώντας μία συνάρτηση κατακερματισμού, ο εξυπηρετητής δημιουργεί ένα αναγνωριστικό χρήστη (User ID) βασισμένο στη λεκτική μονάδα και έναν τυχαίο αριθμό. Ο τυχαίος αριθμός βοηθάει στην αποφυγή επιθέσεων βασισμένων σε κάποιο λεξικό (dictionary attacks) ή σε πίνακες ουρανίου τόξου (rainbow table attack). Η αντιστοιχία της λεκτικής μονάδας με το αναγνωριστικό του χρήστη αποθηκεύεται σε μία βάση δεδομένων. Κατόπιν, αφαιρούνται από το μήνυμα η λεκτική μονάδα, το δημόσιο κλειδί του χρήστη και η υπογραφή των δεδομένων. Η τελευταία αντικαθίσταται από μία νέα υπογραφή η οποία δημιουργείται με βάση το ιδιωτικό κλειδί του εξυπηρετητή και η λεκτική μονάδα με το αναγνωριστικό που παρήχθη από τη συνάρτηση κατακερματισμού. Τέλος, το μήνυμα προωθείται στον ΕΑπ.
Όταν το μήνυμα παραληφθεί από το δεύτερο εξυπηρετητή, γίνεται έλεγχος της εγκυρότητας των δεδομένων, αυτή τη φορά χρησιμοποιώντας το δημόσιο κλειδί του ΕΑν. Μετά την επιτυχή πιστοποίηση, ο δεύτερος αποκρυπτογραφεί το συμμετρικό κλειδί χρησιμοποιώντας το ιδιωτικό του και τελικά αποκρυπτογραφούνται τα δεδομένα. Το συμμετρικό κλειδί, το αναγνωριστικό συνεδρίας και το αναγνωριστικό του χρήστη αποθηκεύονται σε μία βάση δεδομένων και πλέον ο εξυπηρετητής είναι σε θέση να προωθήσει τα δεδομένα στον κατάλληλο τελικό προορισμό.

Εικόνα 5.2, Διάγραμμα ροής αποστολής δεδομένων μεταξύ των εξυπηρετητών

Στην περίπτωση που ο τελικός προορισμός θελήσει να στείλει κάποια απάντηση στο συγκεκριμένο μήνυμα, ακολουθείται η αντίστροφη διαδικασία. Ο εξυπηρετητής αποκρυπτογράφησης κρυπτογραφεί το μήνυμα με το αντίστοιχο συμμετρικό κλειδί και υπογράφει τα δεδομένα χρησιμοποιώντας το ιδιωτικό του κλειδί. Ύστερα ενσωματώνει στο μήνυμα τον τελικό παραλήπτη ο οποίος είναι το αναγνωριστικό χρήστη που έχει αποθηκευτεί στη βάση. Τέλος, στέλνει το μήνυμα στον εξυπηρετητή ανωνυμίας. Ο τελευταίος, αφού λάβει το μήνυμα, ελέγχει για την εγκυρότητα των δεδομένων. Δημιουργεί για ακόμα μία φορά μία ψηφιακή υπογραφή χρησιμοποιώντας το ιδιωτικό του κλειδί, αναζητάει στη βάση δεδομένων την αντιστοιχία του αναγνωριστικού χρήστη με τη λεκτική μονάδα και τέλος προωθεί το μήνυμα στο χρήστη. Μόλις η εφαρμογή λάβει την απάντηση, ελέγχει για μία τελευταία φορά την εγκυρότητα των δεδομένων και αφού τα ελέγξει, τα αποκρυπτογραφεί και παρουσιάζει την απάντηση στο χρήστη.
Αν και τα δεδομένα είναι κρυπτογραφημένα από άκρο σε άκρο, για επιπλέον ασφάλεια, η επικοινωνία μεταξύ των εξυπηρετητών είναι επίσης κρυπτογραφημένη χρησιμοποιώντας το πρωτόκολλο TLS.

5.3 Υλοποίηση

Για την υλοποίηση της παραπάνω αρχιτεκτονικής χρησιμοποιήθηκαν δύο εικονικοί υπολογιστές (virtual machines), ένας για κάθε εξυπηρετητή. Οι εικονικοί υπολογιστές φιλοξενήθηκαν στην υποδομή Openstack του τμήματος Ηλεκτρονικών Μηχανικών του ΑΕΙ Πειραιά ΤΤ.

Ο προγραμματισμός του κάθε εξυπηρετητή έγινε με χρήση της πλατφόρμας Node.js. Η πλατφόρμα αυτή είναι βασισμένη στην πλατφόρμα εκτέλεσης JavaScript του φυλλομετρητή Chrome της Google. Χρησιμοποιείται για την εύκολη και γρήγορη δημιουργία εφαρμογών δικτύου. Βασίζεται στη γλώσσα JavaScript με βιβλιοθήκες για Node.js. Επειδή η συγκεκριμένη πλατφόρμα χρησιμοποιεί ασύγχρονες λειτουργίες, ενδείκνυται για προγραμματισμό γρήγορων εφαρμογών. Τέλος, μία εφαρμογή σε Node.js χρησιμοποιεί μόνο μία διεργασία για να εξυπηρετήσει τα εισερχόμενα αιτήματα.

Παρόλο που μία εφαρμογή Node.js μπορεί να λειτουργήσει αυτόνομα αλλά και σαν εξυπηρετητής ιστού (web server), η συνήθης τακτική είναι η χρήση ενός επιπλέον εξυπηρετητή ιστού μπροστά από τη Node.js εφαρμογή. Χρησιμοποιώντας τον επιπλέον εξυπηρετητή ιστού, προστατεύεται η Node.js εφαρμογή από επιθέσεις DoS [59]. Επιπλέον, αυξάνεται η ταχύτητα με την οποία μπορούν να προσπελαστούν στατικά αρχεία (πχ. αρχεία εικόνας ή κειμένου) και η ταχύτητα παράδοσης περιεχομένου όταν χρησιμοποιείται το πρωτόκολλο SSL [60], [61]. Τέλος, απλοποιείται η διαχείριση των δικαιωμάτων που έχει η εφαρμογή στους πόρους του εξυπηρετητή και η ανάθεση των θυρών. Ανάμεσα στον Apache και τον NGINX, τους δύο πιο συχνά χρησιμοποιούμενους εξυπηρετητές ιστού, επιλέχθηκε ο δεύτερος. Ένας βασικός λόγος επιλογής είναι ότι ο δεύτερος είναι πιο γρήγορος από τον πρώτο. Επίσης ο Apache χρησιμοποιεί άλλο νήμα (thread) για κάθε αίτημα, κάτι που είναι αντίθετο με τον τρόπο λειτουργίας της Node.js.

Για την αποθήκευση των δεδομένων χρησιμοποιήθηκε η MariaDB, μία βάση δεδομένων βασισμένη στη MySQL η οποία προσφέρει μεγαλύτερη ασφάλεια και μεγαλύτερη ταχύτητα από την τελευταία.

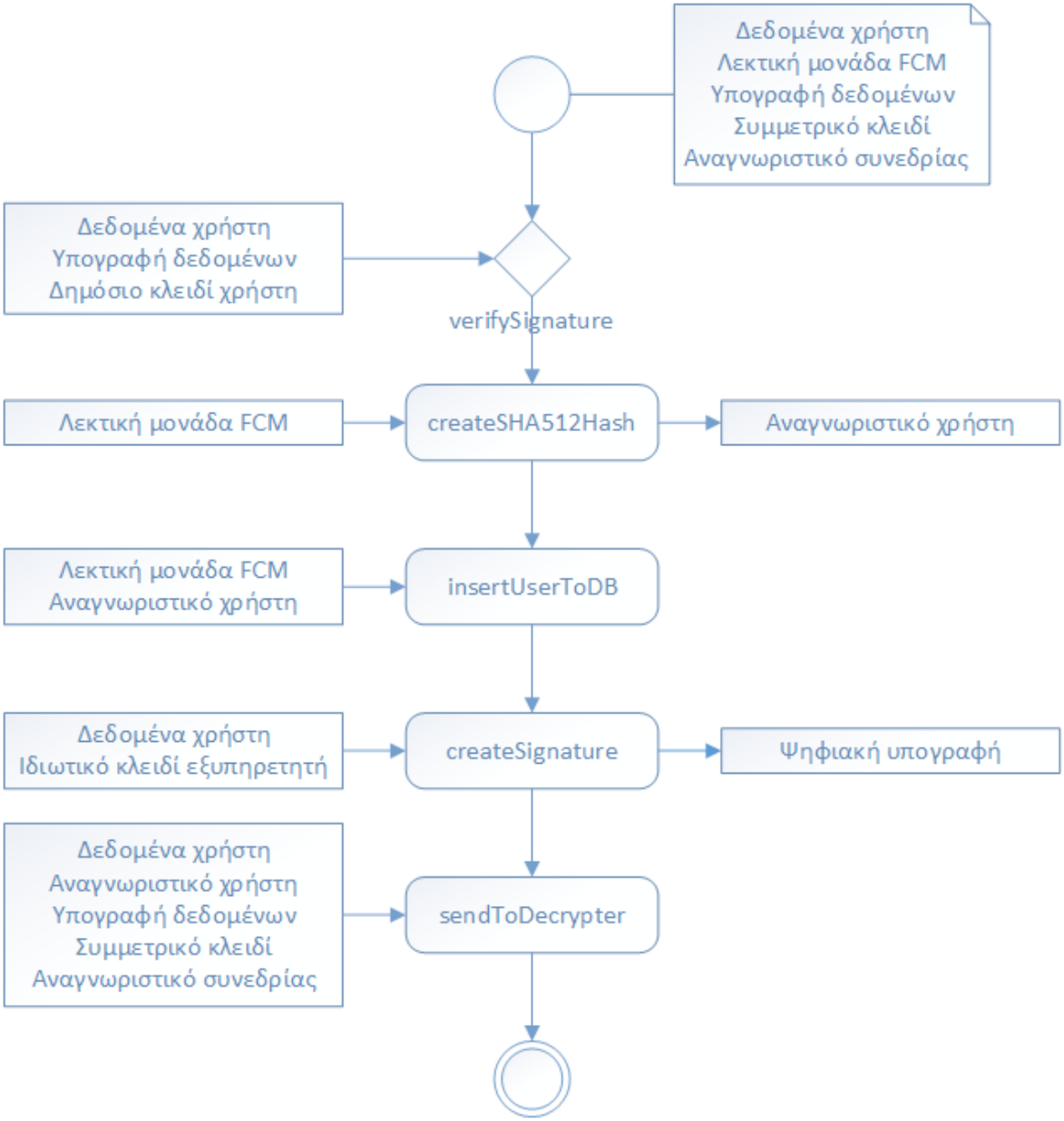
Η αποστολή απαντήσεων στο χρήστη γίνεται μέσω της υπηρεσίας Firebase Cloud Messaging (FCM) της Google. Παρόλο που υπάρχουν αρκετές υπηρεσίες οι οποίες προσφέρουν προώθηση ειδοποίησης, επιλέχθηκε η FCM λόγω του ότι τόσο οι συσκευές με Android όσο και οι συσκευές με iOS διατηρούν ήδη ανοιχτή σύνδεση με τους εξυπηρετητές της FCM οπότε δεν προστίθεται όγκος δεδομένων (εκτός από τη στιγμή της απάντησης) και δεν αυξάνεται η χρήση της μπαταρίας.

Και οι δύο εξυπηρετητές βασίζονται στην αρχιτεκτονική του REST, μίας υπηρεσίας ιστού για την επικοινωνία πελάτη-εξυπηρετητή. Στον ΕΑν, υλοποιήθηκαν τέσσερα τελικά σημεία (endpoints) για την επικοινωνία μεταξύ της εφαρμογής του χρήστη και του ΕΑν και του ΕΑν με τον ΕΑπ. Ένα τελικό σημείο χρησιμοποιείται για την αποστολή μηνύματος από την εφαρμογή του χρήστη στον εξυπηρετητή, ένα για την αποστολή κάποιας απάντησης από τον ΕΑπ στον ΕΑν και τα δύο τελευταία για να μπορεί η εφαρμογή του χρήστη να λάβει τα δημόσια κλειδιά των του συστήματος.

Τελικά σημεία Εξυπηρετητή Ανωνυμίας

Στο πρώτο τελικό σημείο, χρησιμοποιώντας τη μέθοδο POST, η εφαρμογή του χρήστη στέλνει στον εξυπηρετητή τα ακόλουθα δεδομένα σε μορφή JSON.

```
{
     "data":"Κρυπτογραφημένα δεδομένα του χρήστη",
     "signature":"Υπογραφή των δεδομένων με το ιδιωτικό κλειδί
του χρήστη",
     "token":"Λεκτική μονάδα FCM για απάντηση στο χρήστη",
     "key":"Δημόσιο κλειδί του χρήστη",
     "session_id":"Αναγνωριστικό συνεδρίας"
}
```



Εικόνα 5.3, Διάγραμμα ροής πρώτου τελικού σημείου ΕΑν

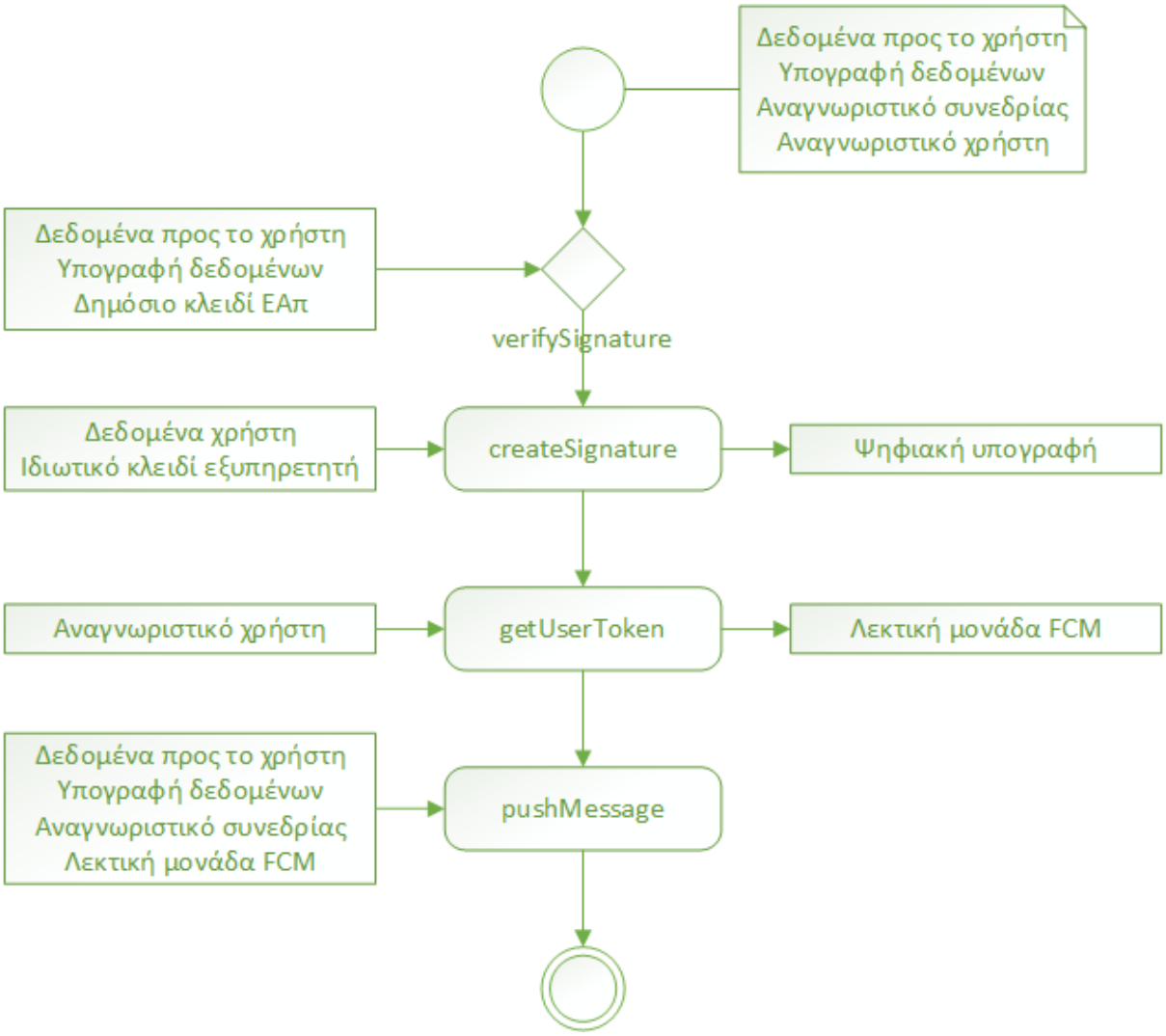
Στην **Error! Reference source not found.**5.3 παρουσιάζεται το διάγραμμα ροής του ελικού σημείου. Αριστερά εμφανίζονται οι παράμετροι εισόδου των συναρτήσεων και δεξιά η έξοδος. Όπως διαπιστώνεται, όταν ο εξυπηρετητής λάβει τα δεδομένα, τρέχει τη συνάρτηση *verifySignature* η οποία δέχεται σαν είσοδο τα δεδομένα, την υπογραφή του χρήστη και το δημόσιο κλειδί του και επιστρέφει αν είναι έγκυρα τα δεδομένα ή όχι. Στην περίπτωση που δεν είναι έγκυρα, επιστρέφει μήνυμα λάθους στο αίτημα. Αν είναι έγκυρα, καλείται η συνάρτηση *createSHA512Hash* όπου δέχεται σαν είσοδο τη λεκτική μονάδα FCM και επιστρέφει το αποτέλεσμα το οποίο είναι και το αναγνωριστικό του χρήστη. Ύστερα, με τη συνάρτηση *insertUserToDB*, γίνεται εγγραφή των δύο τελευταίων στη βάση δεδομένων. Κατόπιν, καλώντας τη συνάρτηση *createSignature*, δημιουργείται μία καινούργια υπογραφή βάση των δεδομένων του χρήστη και του ιδιωτικού κλειδιού του εξυπηρετητή. Μετά τη δημιουργία της υπογραφής, αφαιρούνται από το JSON η λεκτική μονάδα FCM και το κλειδί του χρήστη, αντικαθίσταται η υπογραφή με τη νέα και προστίθεται μία νέα τιμή που περιλαμβάνει το αναγνωριστικό χρήστη. Τέλος, καλείται η μέθοδος *sendToDecrypter* όπου το νέο JSON αποστέλλεται στον ΕΑπ.

Το δεύτερο τελικό σημείο χρησιμοποιείται για την αποστολή κάποιας απάντησης στον χρήστη η οποία έχει προέλθει από τον ΕΑπ. Και σε αυτή την περίπτωση χρησιμοποιείται η μέθοδος POST στην οποία περιλαμβάνεται το ακόλουθο JSON.

```
{
     "recepient":"Αναγνωριστικό χρήστη",
     "data":"Δεδομένα προς το χρήστη",
     "signature":"Υπογραφή των δεδομένων με το ιδιωτικό κλειδί
του εξυπηρετητή αποκρυπτογράφησης",
     "session_id":"Αναγνωριστικό συνεδρίας"
}
```

Ο κώδικας που τρέχει όταν φτάσει κάποιο αίτημα περιλαμβάνει την εκτέλεση της εντολής *verifySignature* για έλεγχο της εγκυρότητας των δεδομένων. Σε θετική απάντηση, καλείται η *createSignature* για τη δημιουργία νέας υπογραφής με το κλειδί του εξυπηρετητή και η *getUserToken* η οποία δεχόμενη σαν είσοδο το αναγνωριστικό του χρήστη, επιστρέφει τη λεκτική μονάδα FCM. Ύστερα γίνεται αντικατάσταση της υπογραφής με τη νέα και καλείται η *pushMessage* η οποία στέλνει το νέο JSON στο χρήστη.
Τα δύο τελευταία τελικά σημεία απαντάνε σε αιτήματα GET και επιστρέφουν το πρώτο το δημόσιο κλειδί του εξυπηρετητή και το δεύτερο το δημόσιο κλειδί του ΕΑπ. Η ανάκτηση του δημοσίου κλειδιού του δεύτερου εξυπηρετητή μέσω του πρώτου γίνεται ώστε να μην υπάρχει κάποια επικοινωνία μεταξύ του χρήστη και του ΕΑπ ώστε να ενισχυθεί η ανωνυμία.

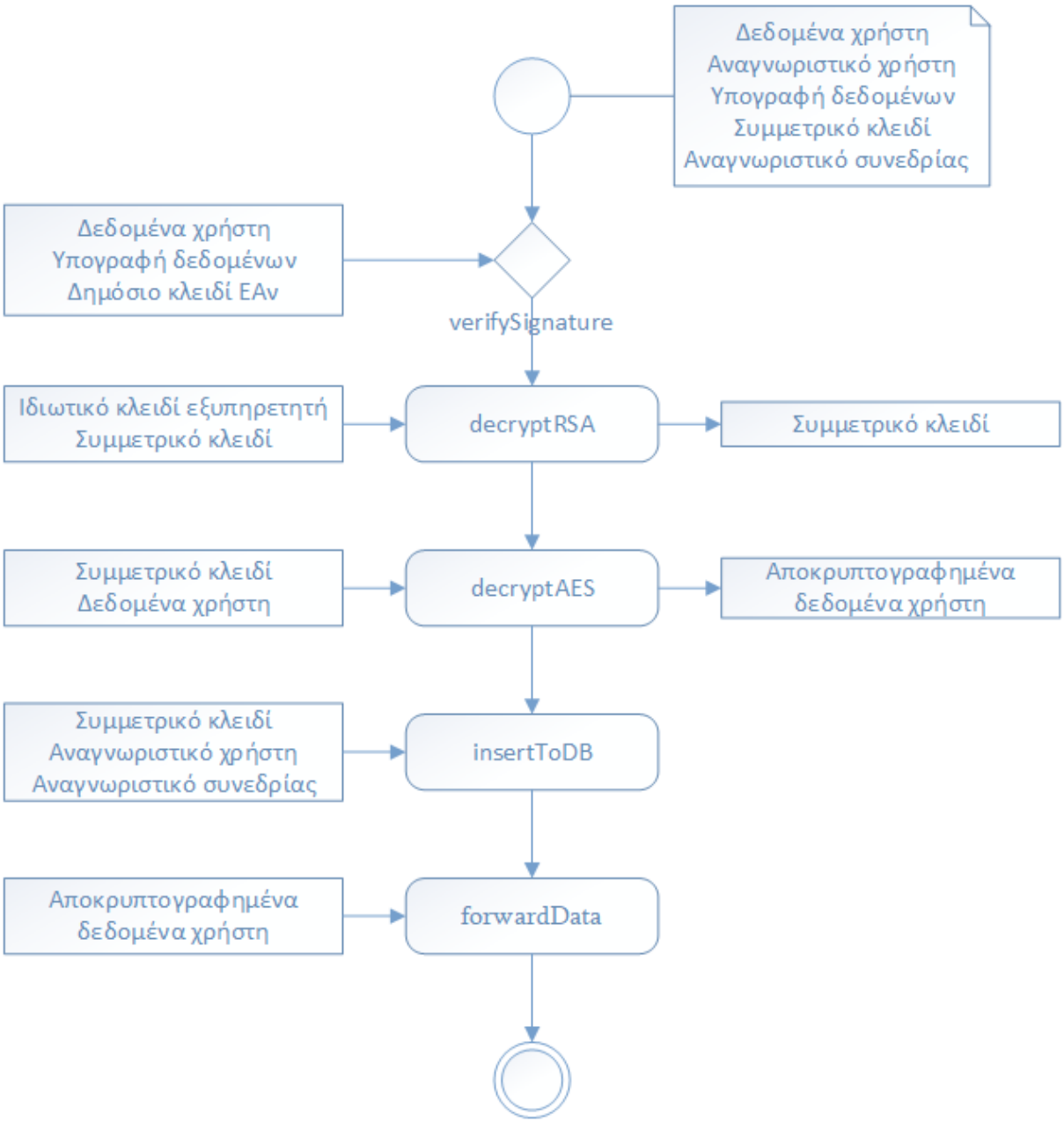
Τελικά σημεία Εξυπηρετητή Αποκρυπτογράφησης



Εικόνα 5.4, Διάγραμμα ροής δεύτερου τελικού σημείου ΕΑν

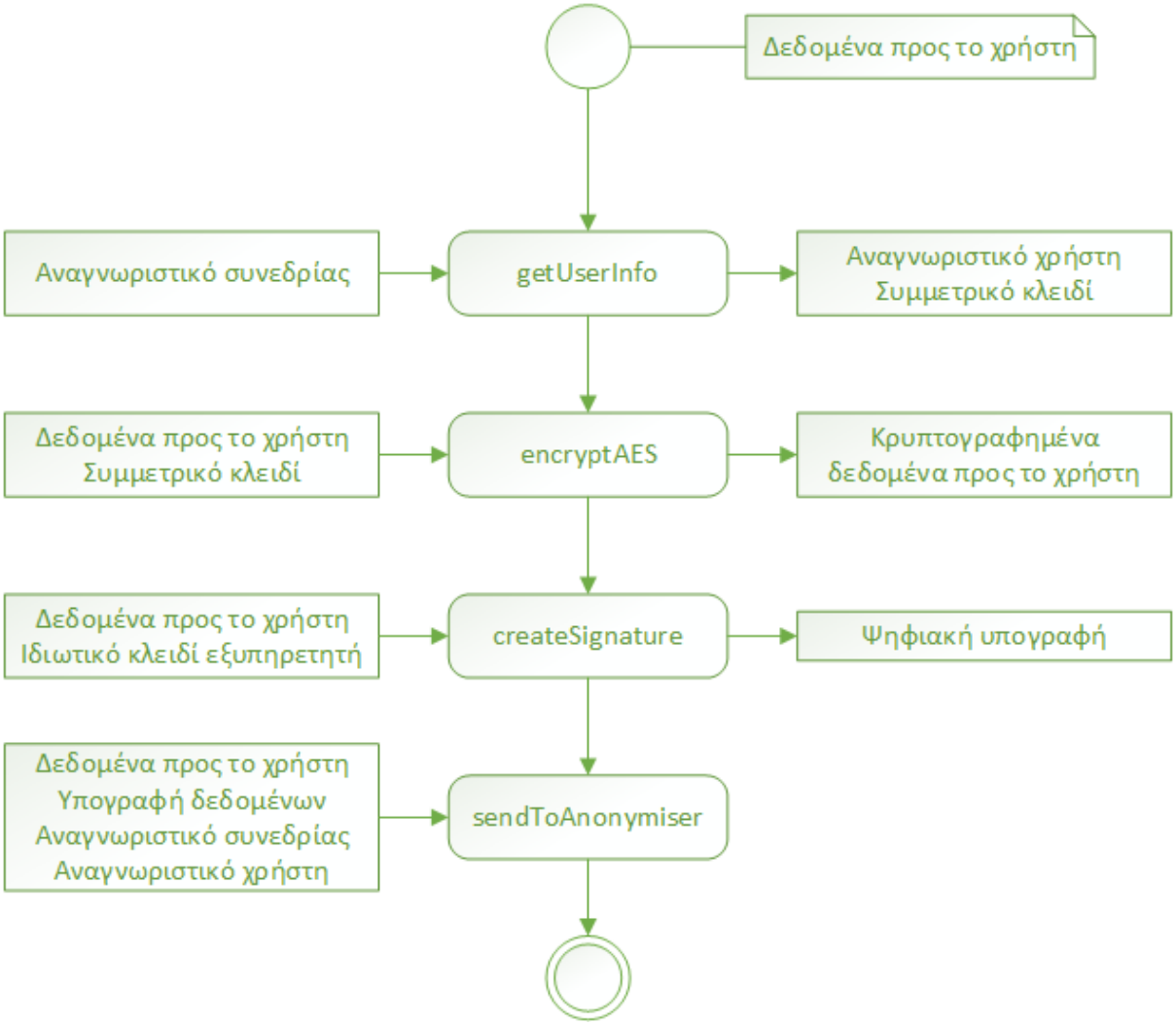
Στον ΕΑπ υλοποιήθηκαν τρία τελικά σημεία εκ των οποίων τα δύο χρησιμοποιούνται στην επικοινωνία μεταξύ των εξυπηρετητών και το τρίτο μεταξύ του εξυπηρετητή αποκρυπτογράφησης και του τελικού προορισμού.
Στο πρώτο τελικό σημείο, το οποίο εξυπηρετεί αιτήματα με τη μέθοδο POST, λαμβάνονται το παρακάτω JSON από τον ΕΑν.

```
{
     "data":"Δεδομένα του χρήστη",
     "signature":"Υπογραφή των δεδομένων με το ιδιωτικό κλειδί
του ΕΑν",
     "id":"αναγνωριστικό χρήστη",
     "secret_key":"Συμμετρικό κλειδί κρυπτογράφησης",
     "session_id":"Αναγνωριστικό συνεδρίας"
}
```



Εικόνα 5.5, Διάγραμμα ροής πρώτου τελικού σημείου ΕΑπ

Όπως και στον ΕΑν, αρχικά γίνεται έλεγχος της εγκυρότητας των δεδομένων με τη συνάρτηση *verifySignature*. Αν τα δεδομένα είναι έγκυρα, καλείται η *decryptRSA* η οποία επιστρέφει το συμμετρικό κλειδί αποκρυπτογραφημένο με τη βοήθεια του ιδιωτικού κλειδιού του ΕΑπ. Έπειτα, όπως παρουσιάζεται στην Εικόνα 5.5 με τη συνάρτηση *decryptAES* και τη χρήση του συμμετρικού κλειδιού, γίνεται αποκρυπτογράφηση των δεδομένων. Με την κλήση της συνάρτησης *insertToDB* γίνεται εγγραφή στη βάση δεδομένων το αναγνωριστικό χρήστη και το συμμετρικό κλειδί ώστε να μπορέσουν να χρησιμοποιηθούν σε περίπτωση απάντησης. Τέλος, η συνάρτηση *forwardData* προωθεί τα δεδομένα στον τελικό προορισμό.



Εικόνα 5.6, Διάγραμμα ροής δεύτερου τελικού σημείου ΕΑπ

Το δεύτερο τελικό σημείο του ΕΑπ υλοποιήθηκε με σκοπό την αποστολή απάντησης από τον τελικό προορισμό προς το χρήστη. Εξυπηρετεί επίσης αιτήματα με τη μέθοδο POST και δέχεται ένα JSON το οποίο περιλαμβάνει σαν προορισμό το αναγνωριστικό συνεδρίας. Χρησιμοποιώντας το τελευταίο, καλείται η *getUserInfo* και επιστρέφει το συμμετρικό κλειδί που έχει χρησιμοποιηθεί για την κρυπτογράφηση των δεδομένων και το αναγνωριστικό χρήστη με το οποίο θα μπορέσει ο ΕΑν να ανακτήσει τη λεκτική μονάδα FCM ώστε να προωθήσει την απάντηση στο χρήστη. Αφού επιστραφούν αυτές οι τιμές ετοιμάζεται το JSON που θα αποσταλεί στον ΕΑν. Πρώτα ενσωματώνεται το αναγνωριστικό συνεδρίας στο μήνυμα. Ύστερα με τη συνάρτηση *encryptAES* κρυπτογραφούνται τα δεδομένα με το συμμετρικό κλειδί που αντιστοιχεί σε αυτή τη συνεδρία. Μετά, δημιουργείται η ψηφιακή υπογραφή με το ιδιωτικό κλειδί του ΕΑπ. Τελικά, με την ενσωμάτωση των κρυπτογραφημένων δεδομένων και της ψηφιακής υπογραφής, με τη συνάρτηση *sendToAnonymiser* το μήνυμα αποστέλλεται στον ΕΑν ώστε να επεξεργαστεί και να αποσταλεί στο χρήστη.

Το τρίτο και τελευταίο τελικό σημείο του εξυπηρετητή, απαντάει σε αιτήματα GET και υλοποιήθηκε ώστε να μπορεί ο ΕΑν να πάρει το κλειδί του ΕΑπ.

5.4 Χρήση σε περιβάλλον κινητού και φορετού υπολογισμού

Η αρχιτεκτονική που παρουσιάστηκε και υλοποιήθηκε θα μπορούσε να χρησιμοποιηθεί σε ποικίλες εφαρμογές όπου είναι απαραίτητο να διασφαλιστεί η ιδιωτικότητα του χρήστη. Ενδεικτικά αναφέρονται περιπτώσεις ανώνυμης αποστολής μηνυμάτων, ανώνυμης αποστολής αποτελεσμάτων πληθοπορισμού και ανώνυμης αποστολής καταγγελιών.

Έγινε ανάπτυξη μίας εφαρμογής για κινητές συσκευές με λειτουργικό σύστημα Android η οποία με τη βοήθεια ενός κουμπιού έχει τη δυνατότητα να στείλει ανώνυμα μηνύματα ηλεκτρονικού ταχυδρομείου. Ο κώδικας της εφαρμογής βρίσκεται στο Παράρτημα 1.

Το κουμπί έχεις τις διαστάσεις ενός κέρματος και μπορεί ο χρήστης να το έχει στην τσέπη του ή στο τιμόνι του αυτοκινήτου ώστε να είναι εύκολα προσβάσιμο. Το κουμπί προσφέρεται από την εταιρία Shortcut Labs με κύριο σκοπό τον εύκολο χειρισμό του κινητού τηλεφώνου αλλά η εταιρία προσφέρει API ώστε οι προγραμματιστές να αναπτύξουν εφαρμογές για διαφορετικές χρήσεις. Η επικοινωνία του κουμπιού με το κινητό τηλέφωνο γίνεται μέσω Bluetooth χαμηλής κατανάλωσης (Bluetooth Low Energy – BLE) και διαθέτει τρεις λειτουργίες: απλό πάτημα, διπλό πάτημα και παρατεταμένο πάτημα.

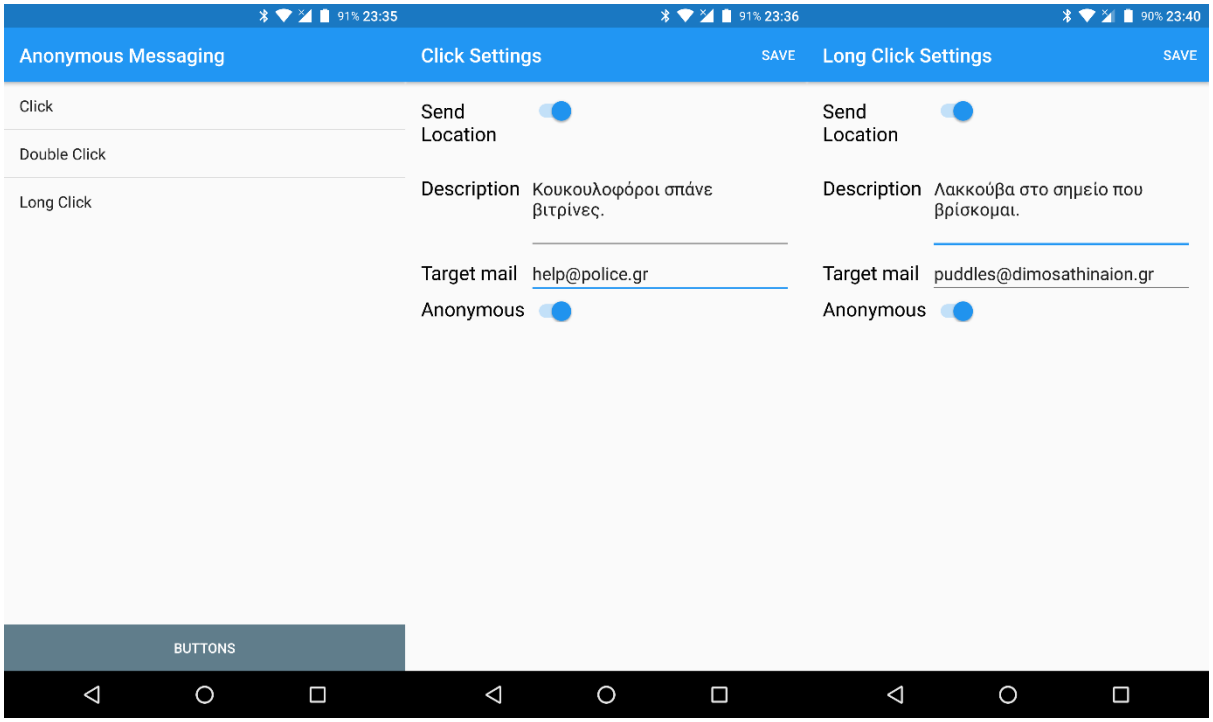
Για τελικός προορισμός υλοποιήθηκε ένας εξυπηρετητής ο οποίος λαμβάνει κάποιο μήνυμα με τη μέθοδο POST και αναλαμβάνει να το προωθήσει στη διεύθυνση ηλεκτρονικού ταχυδρομείου που περιλαμβάνεται στο μήνυμα. Στο Παράρτημα 2 παρουσιάζεται ο κώδικας του εξυπηρετητή.

Ο χρήστης έχει τη δυνατότητα μέσα από την εφαρμογή να καθορίσει για κάθε τύπο πατήματος του κουμπιού το περιεχόμενο του μηνύματος, την ηλεκτρονική διεύθυνση που θα σταλεί και την τοποθεσία του. Τέλος, έχει τη δυνατότητα να γράψει τη δική του ηλεκτρονική διεύθυνση στην περίπτωση που επιλέξει το μήνυμα να μη σταλεί ανώνυμα. Αυτές οι πληροφορίες αποθηκεύονται ώστε να είναι δυνατή η ανάκτησή τους μετά το πάτημα ενός κουμπιού.

Στην εφαρμογή τρέχει μία υπηρεσία στο παρασκήνιο η οποία λαμβάνει τα πατήματα του κουμπιού. Η υπηρεσία ξεκινάει να περιμένει κάποιο πάτημα μόλις ο χρήστης ενεργοποιήσει τη συσκευή. Μόλις αντιληφθεί κάποιο πάτημα, καθορίζει τον τύπο του και συνεχίζει την προετοιμασία του ανάλογου μηνύματος.

Γνωρίζοντας τον τύπο του πατήματος, φορτώνονται οι παράμετροι που έχει καθορίσει ο χρήστης. Αν ο χρήστης έχει επιλέξει να στείλει και την τοποθεσία του, η εφαρμογή λαμβάνει την τοποθεσία είτε από το GPS είτε από το δίκτυο και φτιάχνει το JSON που θα σταλεί. Τέλος, ανάλογα την επιλογή του χρήστη για ανώνυμη ή επώνυμη αποστολή, το JSON αποστέλλεται στον κατάλληλο εξυπηρετητή. Στην πρώτη περίπτωση

αποστέλλεται στον πρώτο εξυπηρετητή της αρχιτεκτονικής που παρουσιάστηκε σε προηγούμενη ενότητα ο οποίος με τη σειρά του το στέλνει στο δεύτερο και τελικά στον εξυπηρετητή αλληλογραφίας αλλιώς αποστέλλεται απευθείας στον εξυπηρετητή αλληλογραφίας.
Μόλις γίνει η αποστολή, ο χρήστης ειδοποιείται στην κινητή του συσκευή σχετικά με το αν ήταν επιτυχής ή όχι η αποστολή. Αν ο χρήστης διαθέτει και κάποιο έξυπνο ρολόι, ειδοποιείται και σε αυτό.



Εικόνα 5.7, Εφαρμογή Android η οποία στέλνει ανώνυμα δεδομένα χρησιμοποιώντας την αρχιτεκτονική που παρουσιάστηκε

Το μήνυμα που λαμβάνεται στο ηλεκτρονικό ταχυδρομείο του χρήστη έχει την ακόλουθη μορφή.

```
A user with hidden id,
has sent on 2017-10-22T21:28:38.56200GMT+02:00,
from an unknown location,
the message:
"Κουκουλοφόροι σπάνε βιτρίνες".
```

Στην περίπτωση που υπάρχει τοποθεσία, εμφανίζεται και αυτή στο μήνυμα. Αντίστοιχα εμφανίζεται το email του αποστολέα αν έχει επιλέξει να μη σταλεί ανώνυμα το μήνυμα.

5.5 Θέματα χρήσης και συμφωνία με ρυθμιστικό πλαίσιο

Η προτεινόμενη αρχιτεκτονική για μια πύλη επικοινωνίας μπορεί να είναι πολύ χρήσιμη τόσο στην ενίσχυση της ιδιωτικότητας του χρήστη όσο και στην υποστήριξη ανώνυμων καταγγελιών συμβάντων όπως ύποπτης συμπεριφοράς, εγκληματικής δραστηριότητας, διαφθοράς και κατάχρησης εξουσίας.

Σε τέτοιες περιπτώσεις, έχει αποδειχθεί ότι για την παροχή ανώνυμων καταγγελιών από τους πολίτες είναι κρίσιμη η εμπιστοσύνη του μηχανισμό καταγγελιών και κατ' επέκταση η χρήση του. Ένα τρανταχτό παράδειγμα είναι τα προγράμματα Crime Stoppers. Τα προγράμματα αυτά επιτρέπουν την παροχή ανώνυμων πληροφοριών σχετικά με κάποια εγκληματική δραστηριότητα. Οι πολίτες που έχουν δει ή ακούσει κάποια πληροφορία για κάποιο έγκλημα πολλές φορές διστάζουν να δώσουν αυτές τις πληροφορίες στις απευθείας στις αρχές. Χρησιμοποιώντας αυτό το πρόγραμμα, οι πολίτες είναι σε θέση να δώσουν τις πληροφορίες ανώνυμα και τελικά οι αρχές να τις λάβουν εκ μέρους των ανθρώπων υπεύθυνων για τη λειτουργία του προγράμματος. Σύμφωνα με στατιστικές, το 2014-2015, κατά μέσο όρο 14 άνθρωποι συλλαμβάνονται καθημερινά λόγω της χρήσης αυτού του προγράμματος [62]. Επιπροσθέτως, οπουδήποτε το νομικό σύστημα παρέχει ένα πλαίσιο μέσα στο οποίο η ανωνυμία μπορεί να διασφαλιστεί (ενάντια σε αιτήσεις για την αποκάλυψη των αρχείων επικοινωνίας), η χρήση της προτεινόμενης αρχιτεκτονικής μπορεί να προσφέρει τη βάση για την εμπιστοσύνη των πολιτών στην καταγγελία σε εκτελεστικούς φορείς του νόμου.

Το 2006 η Ευρωπαϊκή Ένωση εξέδωσε μία Οδηγία που αφορούσε την διατήρηση των τηλεπικοινωνιακών δεδομένων των πολιτών. Σύμφωνα με αυτή την οδηγία, τα κράτη-μέλη έπρεπε να διατηρούν τα δεδομένα από 6 έως 24 μήνες και η αστυνομία θα μπορούσε να ζητήσει πρόσβαση σε αυτά μετά από εντολή δικαστηρίου, [63]. Η Οδηγία τελικά θεωρήθηκε άκυρη το 2014 από το Δικαστήριο της Ευρωπαϊκής Ένωσης (ΔΕΕ) διότι παραβίαζε το δικαίωμα της ιδιωτικότητας σύμφωνα με το Χάρτη Θεμελιωδών Δικαιωμάτων της Ευρωπαϊκής Ένωσης, [64]. Παρόλα αυτά σε κάποιες χώρες υπάρχει ακόμα νομοθεσία η οποία επιβάλλει την διατήρηση αρχείων καταγραφής όπως στη Σουηδία, [65]. Για αυτούς τους λόγους, στην αρχιτεκτονική που παρουσιάστηκε υπάρχει η δυνατότητα διατήρησης αρχείων καταγραφής ώστε να είναι δυνατός ο εντοπισμός του χρήστη ύστερα από νόμιμες διαδικασίες.

Παρόλα αυτά το δικαστήριο του Καναδά θεωρεί σημαντική την προστασία της ανωνυμίας των πολιτών που κάνουν χρήση του προγράμματος Crime Stoppers και με εντολή του απαγορεύει την αποκάλυψη τυχόν στοιχείων που μπορούν να αποκαλύψουν την ταυτότητα του χρήστη, [66]. Με βάση αυτό, στην παραπάνω αρχιτεκτονική προστέθηκε και το χαρακτηριστικό της διαγραφής των πληροφοριών που μπορούν να αποκαλύψουν την ταυτότητα του χρήστη αυτών, ώστε να είναι δυνατή η διατήρηση της ανωνυμίας του σε κάθε περίπτωση.

ΚΕΦΑΛΑΙΟ 6:
Συμπεράσματα

Στο σύγχρονο κόσμο αφενός καθημερινά δημιουργείται και ανταλλάσσεται τεράστιος όγκος ψηφιακής πληροφορίας χρησιμοποιώντας φορητές συσκευές και αφετέρου διάφοροι «τρίτοι» προσπαθούν να υποκλέψουν αυτή την πληροφορία για διάφορους σκοπούς. Η διεκδίκηση του δικαιώματος της προστασίας της ιδιωτικότητας του καθενός είναι πλέον επιτακτική. Στην παρούσα εργασία έγινε μία προσπάθεια συλλογής και παρουσίασης των σύγχρονων τρόπων με τους οποίους μπορεί κανείς να προστατεύσει τα προσωπικά του δεδομένα. Η συγκεκριμένη εργασία θα μπορούσε να αποτελέσει οδηγό για τη δημιουργία μίας ιστοσελίδας η οποία θα είχε σαν σκοπό την ενημέρωση των χρηστών κινητών συσκευών σχετικά με τα ευαίσθητα δεδομένα καθώς και την φιλοξενία σύντομων και εύκολων οδηγιών όπου θα μπορούσαν οι χρήστες να ακολουθήσουν για να προστατεύσουν την ιδιωτικότητά τους.

Όπως διαπιστώθηκε η κρυπτογραφία είναι μείζονος σημασίας στη μεταφορά και αποθήκευση πληροφοριών χωρίς αυτή να υποπέσει σε χέρια άλλων για τους οποίους δεν προορίζονται. Για την καλύτερη προστασία των δεδομένων θα μπορούσαν να αναπτυχθούν συστήματα τα οποία κάνουν χρήση μοντέρνων τεχνικών κρυπτογράφησης όπως η ομομορφική κρυπτογραφία και η κρυπτογραφία διατήρησης μορφής (Format Preserving Encryption – FPE). Το πρώτο κρυπτοσύστημα επιτρέπει την πραγματοποίηση υπολογισμών στο κρυπτοκείμενο και το αποτέλεσμα είναι ίδιο με το αποτέλεσμα που προκύπτει από τους υπολογισμούς στο απλό κείμενο. Το δεύτερο αναφέρεται στην κρυπτογράφηση δεδομένων με τέτοιο τρόπο ώστε το κρυπτοκείμενο να έχει την ίδια μορφή με το απλό κείμενο. Κρυπτογραφώντας για παράδειγμα τον αριθμό ενός τηλεφώνου, το κρυπτοκείμενο θα είχε πάλι τη μορφή ενός αριθμού τηλεφώνου. Ένα πεδίο εφαρμογής του πρώτου θα μπορούσε να είναι η πρόληψη καρδιακών παθήσεων. Ένα άτομο φοράει μία συσκευή η οποία ανά τακτά χρονικά διαστήματα μετράει τους παλμούς της καρδιάς του και τους στέλνει στο κινητό τηλέφωνο μαζί με δεδομένα από το γυροσκόπιο και τον αισθητήρα επιτάχυνσης. Το κινητό τηλέφωνο γνωρίζοντας κάποια φυσικά χαρακτηριστικά του ατόμου (βάρος, ύψος κλπ) υπολογίζει τη δραστηριότητά του με τα δεδομένα των αισθητήρων κίνησης και τέλος υπολογίζει αν είναι φυσιολογικοί οι παλμοί και ειδοποιεί ανάλογα το άτομο. Χρησιμοποιώντας ομομορφική κρυπτογράφηση, τα δεδομένα θα μπορούσαν να αποσταλούν από τη φορετή συσκευή κρυπτογραφημένα και ο αλγόριθμος θα μπορούσε να κάνει τους υπολογισμούς χωρίς να τα αποκρυπτογραφήσει. Με αυτόν τον τρόπο δεν υπάρχει κίνδυνος τα ευαίσθητα αυτά δεδομένα να προσπελαστούν από κανέναν άλλο.

Στα έξυπνα κινητά τηλέφωνα και ρολόγια υπάρχει η δυνατότητα πραγματοποίησης ανέπαφων τραπεζικών συναλλαγών με στοιχεία κάρτας που έχει εισάγει ο χρήστης.

Κατά τη διαδικασία της συναλλαγής τα στοιχεία μεταφέρονται μέσω επικοινωνίας κοντινού πεδίου (Near-field Communication - NFC) από τη συσκευή του χρήστη στην τράπεζα μέσω τερματικής συσκευής πληρωμής. Κάνοντας χρήση της κρυπτογραφίας διατήρησης μορφής η συσκευή του χρήστη θα μπορούσε να στέλνει κρυπτογραφημένα τα δεδομένα στην τράπεζα χωρίς να τα εμποδίζει η τερματική συσκευή επειδή θα τα θεωρούσε έγκυρα δεδομένα τραπεζικής συναλλαγής.

Στο τρίτο κεφάλαιο αναλύθηκαν οι αρχιτεκτονικές για την προστασία της ιδιωτικότητας. Φυσικά υπάρχουν και αρχιτεκτονικές που βρίσκονται υπό ανάπτυξη. Μία τέτοια είναι η αλυσίδα μπλοκ (blockchain). Αυτή είναι μία λίστα με εγγραφές (ή μπλοκ) οι οποίες είναι ενωμένες μεταξύ τους και ασφαλισμένες με κρυπτογράφηση. Ουσιαστικά πρόκειται για ένα μεγάλο λογιστικό βιβλίο το οποίο περιέχει οικονομικές συναλλαγές. Αυτές οι εγγραφές δεν είναι αποθηκευμένες σε κάποιο εξυπηρετητή αλλά μοιρασμένες σε ένα κατανεμημένο υπολογιστικό σύστημα. Αυτή η ιδιότητα κάνει τις εγγραφές ανεκτικές σε αλλοιώσεις, αφού αν θελήσει κάποιος να αλλάξει μία εγγραφή πρέπει να αλλάξει την εγγραφή σε όλους τους κόμβους που συμμετέχουν, κάτι το οποίο είναι πρακτικά αδύνατο. Με την κρυπτογράφηση εγγυάται η ασφάλεια των δεδομένων που περιέχονται σε αυτές τις εγγραφές. Μία γνωστή πρώτη χρήση της αλυσίδας μπλοκ είναι το Bitcoin, ένα ψηφιακό κρυπτονόμισμα. Οι ιδιότητες αυτές κάνουν την αλυσίδα μπλοκ κατάλληλη για αποθήκευση δεδομένων [67], [68].

Στο τέταρτο κεφάλαιο, λαμβάνοντας υπόψη τις περισσότερο κοινές χρήσεις μίας κινητής συσκευής, εξετάστηκε το θέμα του τρόπου συλλογής προσωπικών δεδομένων καθώς και το περιεχόμενο των δεδομένων αυτών. Σύμφωνα με αυτά, παρουσιάστηκαν τρόποι αποφυγής της παραβίασης των προσωπικών δεδομένων για εκείνα που ανταλλάσσονται μέσω διαδικτύου, αλλά και για αυτά που υπάρχουν στην ίδια τη συσκευή αφού υπάρχει και φυσικός κίνδυνος παραβίασης. Δυστυχώς όμως υπάρχει αρκετή άγνοια στους χρήστες σχετικά με αυτές τις πληροφορίες. Κάποιοι άνθρωποι μπορεί να έχουν ακούσει ότι “τους παρακολουθούν” αλλά χωρίς να γνωρίζουν λεπτομέρειες επί του θέματος απλώς συνεχίζουν να χρησιμοποιούν τις συσκευές τους με τον ίδιο τρόπο. Θα πρέπει λοιπόν να γίνει μία προσπάθεια ενημέρωσης του κόσμου κυρίως μέσω ιστοσελίδων που έχουν μεγάλη επισκεψιμότητα. Προσπάθεια θα πρέπει να γίνει και από την κοινότητα των προγραμματιστών ανοιχτού λογισμικού αναπτύσσοντας εφαρμογές ισάξιας ή και καλύτερης λειτουργικότητας από τις υπάρχουσες με κλειστό κώδικα. Οι τελευταίες προσφέρουν πολλές φορές χαρακτηριστικά πιο εύκολα προσβάσιμα στο χρήστη από τις άλλες.

Ακολουθώντας τους τρόπους αυτούς, σίγουρα μετριάζεται η παραβίαση της ιδιωτικότητας αλλά δεδομένων αρκετών πόρων και υπολογιστικής ισχύος, η απειλή δεν παύει να υφίσταται. Όλα τα συστήματα έχουν αδυναμίες και η παραβίασή τους δεν είναι αδύνατη. Η πρώτη από τις δύο ακραίες λύσεις είναι η αποφυγή χρήσης συστημάτων διασφάλισης ιδιωτικότητας αφού τελικά θα παραβιαστούν τα δεδομένα και η δεύτερη η αποφυγή χρήσης των συσκευών. Η ρεαλιστική προσέγγιση απαιτεί να κινηθεί κανείς σε λύσεις που βρίσκονται κάπου στη μέση, μεταξύ των δύο αυτών

άκρων. Οι φορητές συσκευές σαφώς διευκολύνουν την καθημερινή ζωή, ωστόσο η ιδιωτικότητα παραμένει θεμελιώδες δικαίωμα και πρέπει να προστατεύεται.

Όσον αφορά τη χρήση της αρχιτεκτονικής που προτάθηκε και παρουσιάστηκε, [69], η ενσωμάτωσή της σε προγράμματα Crime Stoppers φαίνεται να ταιριάζει απόλυτα. Το σύστημα θα μπορούσε να διοικείται πλήρως από αυτά τα προγράμματα ή από τα σώματα ασφαλείας στην περίπτωση που υπάρχει απόλυτη εμπιστοσύνη. Φυσικά, μια υβριδική προσέγγιση θα μπορούσε να επιτευχθεί όπου ο πρώτος εξυπηρετητής θα χειρίζεται από μία ανεξάρτητη αρχή και ο δεύτερος από τα σώματα ασφαλείας.

Εν κατακλείδι, η ιδιωτικότητα είναι μία μεγάλη περιοχή έρευνας και συνδέεται άμεσα με την ασφάλεια και την κρυπτογραφία. Η πρόοδος σε αυτούς τους τομείς θα φέρει πρόοδο και στην ιδιωτικότητα. Η ανάπτυξη νέων συστημάτων ασφαλείας καθώς και η επινόηση νέων αλγορίθμων κρυπτογράφησης θα κάνει τον φορητό και φορετό κόσμο, έναν κόσμο που συνεχώς μικραίνει, λιγότερο ευάλωτο σε παραβιάσεις προσωπικών δεδομένων.

ΒΙΒΛΙΟΓΡΑΦΙΑ – ΔΙΑΔΙΚΤΥΑΚΕΣ ΠΗΓΕΣ

[1] We are social, «Digital in 2017: Global overview,» [Ηλεκτρονικό]. Available: https://wearesocial.com/special-reports/digital-in-2017-global-overview. [Πρόσβαση Οκτώβριος 2017].

[2] Statista, «Statistics & Facts on Tablets,» [Ηλεκτρονικό]. Available: https://www.statista.com/topics/841/tablets/. [Πρόσβαση Οκτώβριος 2017].

[3] B. J. Phillips, C. D. Schmidt και D. R. Kelly, «Recovering data from USB Flash memory sticks that have been damaged or electronically erased,» 2008.

[4] Αρχή Προστασίας Δεδομένων Προσωπικού Χαρακτήρα, «Δέκα ερωτήσεις - απαντήσεις για τα προσωπικά δεδομένα,» [Ηλεκτρονικό]. Available: http://www.dpa.gr/portal/page?_pageid=33,18990&_dad=portal&_schema=PORTAL. [Πρόσβαση Σεπτέμβριος 2017].

[5] Privacy International, «What is Privacy,» [Ηλεκτρονικό]. Available: https://www.privacyinternational.org/node/54. [Πρόσβαση Οκτώβριος 2017].

[6] A. Pfitzmann και M. Hansen, «A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management,» 2010.

[7] Net Applications, «Mobile/Tablet Operating System Market Share,» [Ηλεκτρονικό]. Available: https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=8&qpcustomd=1&qpsp=2017&qpnp=1&qptimeframe=Y. [Πρόσβαση Οκτώβριος 2017].

[8] Apple Inc., «iPhone 7,» Apple Inc., [Ηλεκτρονικό]. Available: https://www.apple.com/shop/buy-iphone/iphone-7/5.5-inch-display-128gb-black-att. [Πρόσβαση Οκτώβριος 2017].

[9] Quartz, «The Google Pixel has an excellent camera, but not much else,» [Ηλεκτρονικό]. Available: https://qz.com/815488/google-pixel-review/. [Πρόσβαση Οκτώβριος 2017].

[10] Polar Electro, «Polar H10 Heart Rate Sensor,» Polar. [Ηλεκτρονικό]. [Πρόσβαση Οκτώβριος 2017].

[11] Polar Electro, «Stride sensor Bluetooth® Smart,» Polar, [Ηλεκτρονικό]. Available: https://www.polar.com/en/products/accessories/stride_sensor_bluetooth_smart. [Πρόσβαση Οκτώβριος 2017].

[12] Tizen Indonesia, «Tizen kalahkan Android Wear di Q1 2017,» [Ηλεκτρονικό]. Available: http://www.tizenindonesia.org/2017/05/tizen-kalahkan-android-wear-di-q1-2017.html. [Πρόσβαση Οκτώβριος 2017].

[13] Polar Electro, «Polar A300 Fitness watch & activity tracker,» Polar. [Ηλεκτρονικό]. [Πρόσβαση Οκτώβριος 2017].

[14] Apple Inc., «Apple Watch,» [Ηλεκτρονικό]. Available: https://www.apple.com/lae/watch/. [Πρόσβαση Οκτώβριος 2017].

[15] Michael Kors, «Grayson Silver-Tone Smartwatch,» [Ηλεκτρονικό]. Available: https://www.michaelkors.com/grayson-silver-tone-smartwatch/_/R-US_MKT5025. [Πρόσβαση Οκτώβριος 2017].

[16] Practical Cryptography, «Cryptanalysis of the Caesar Cipher,» [Ηλεκτρονικό]. Available: http://practicalcryptography.com/cryptanalysis/stochastic-searching/cryptanalysis-caesar-cipher/. [Πρόσβαση Σεπτέμβριος 2017].

[17] National Institute of Standards and Technology, *Data Encryption Standard (DES),* Federal Information Processing Standards, 1999.

[18] Electronic Frontier Foundation, Cracking DES, O'Reilly Media, 1998.

[19] W. Stallings, Cryptography and Network Security, 5th επιμ., Pearson Education Inc., 2011.

[20] National Institute of Standards and Technology, *Advanced Encryption Standard,* Federal Information

Processing Standards, 2001.

[21] R. Rivest, A. Shamir και L. Adleman, «A Method for Obtaining Digital Signatures and Public-Key Cryptosystems,» *Communications fo the ACM,* τόμ. 21, αρ. 2, pp. 120-126, 1978.

[22] R. L. Rivest, A. Shamir και L. M. Adleman, «Cryptographic communications system and method». ΗΠΑ Ευρεσιτεχνία 4,405,829, 20 Σεπτεμβρίου 1983.

[23] *An Introduction to Cryptography,* Network Associates, Inc., 2000.

[24] A. Greenberg, «Hacker Lexicon: What is End-to-End Encryption?,» [Ηλεκτρονικό]. Available: https://www.wired.com/2014/11/hacker-lexicon-end-to-end-encryption/. [Πρόσβαση Οκτώβριος 2017].

[25] M. K. Reiter και A. D. Rubin, «Crowds: anonymity fro Web transactions,» *ACM Transactions on Information and System Security (TISSEC),* τόμ. 1, αρ. 1, pp. 66-92, Νοέμβριος 1998.

[26] Onion Routing, «Onion Routing Publications,» [Ηλεκτρονικό]. Available: https://www.onion-router.net/Publications.html. [Πρόσβαση Οκτώβριος 2017].

[27] R. Dingledine, P. Mathewson και P. Syverson, «Tor: the second-generation onion router,» σε *Proceedings of the 13th conference on USENIX Security Symposium*, San Diego, CA, 2004.

[28] Tor Project, «How Tor works,» [Ηλεκτρονικό]. Available: https://www.torproject.org/about/overview.html.en. [Πρόσβαση Οκτώβριος 2017].

[29] S. Parekh, «Prospects for Remailers - Where is Anonymity Heading on the Internet?,» 5 Αυγούστου 1996. [Ηλεκτρονικό]. Available: http://www.firstmonday.dk/ojs/index.php/fm/article/view/476/397. [Πρόσβαση Ιούλιος 2017].

[30] IETF, «Mixmaster Protocol Version 2,» [Ηλεκτρονικό]. Available: https://tools.ietf.org/html/draft-sassaman-mixmaster-03. [Πρόσβαση Ιούλιος 2017].

[31] G. Danezis, R. Dingledine και N. Mathewson, «Mixminion: Design of a Type III Anonymous Remailer Protocol,» σε *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, 2003.

[32] Anderson Software, LLC, «TipSoft SMS Security and Anonymity Overview,» [Ηλεκτρονικό]. Available: http://www.smscrimetips.com/sms_security_summary.pdf. [Πρόσβαση Οκτώβριος 2017].

[33] I. Ghiciuc, «State of the mobile economy in 2014,» [Ηλεκτρονικό]. Available: https://www.thinslices.com/smartphone-statistics-tablet-usage-patterns/. [Πρόσβαση Οκτώβριος 2017].

[34] Sunday Express, «Top uses of our smartphones - and calling doesn't even make the list,» Μάρτιος 2017. [Ηλεκτρονικό]. Available: https://www.express.co.uk/life-style/science-technology/778572/Smartphone-phone-common-reason-use-call/amp. [Πρόσβαση Οκτώβριος 2017].

[35] Online Email Verify, «List of most popular email domains (by number of live emails),» 4 Ιουνίου 2016. [Ηλεκτρονικό]. Available: https://email-verify.my-addr.com/list-of-most-popular-email-domains.php. [Πρόσβαση Οκτώβριος 2017].

[36] Google Inc., «Google Privacy Policy,» 2 Οκτωβρίου 2017. [Ηλεκτρονικό]. Available: https://www.google.com/policies/privacy/. [Πρόσβαση Οκτώβριος 2017].

[37] Microsoft, «Microsoft Privacy Statement,» Οκτώβριος 2017. [Ηλεκτρονικό]. Available: https://privacy.microsoft.com/en-us/privacystatement. [Πρόσβαση Οκτώβριος 2017].

[38] Yahoo! EMEA Limited, «Yahoo Privacy Policy,» 16 Αυγούστου 2017. [Ηλεκτρονικό]. Available: https://policies.yahoo.com/ie/el/yahoo/privacy/index.htm. [Πρόσβαση Οκτώβριος 2017].

[39] ProtonMail, «ProtonMail Privacy Policy,» [Ηλεκτρονικό]. Available: https://protonmail.com/privacy-policy. [Πρόσβαση Οκτώβριος 2017].

[40] Viber Media S.à r.l., «Viber Encryption Overview,» [Ηλεκτρονικό]. Available: https://www.viber.com/security-overview. [Πρόσβαση Οκτώβριος 2017].

[41] Amnesty International, «6 really practical ways to protect your privacy online,» 21 Οκτωβρίου 2016. [Ηλεκτρονικό]. Available: https://www.amnesty.org/en/latest/campaigns/2016/10/really-practical-ways-to-protect-your-privacy-online/. [Πρόσβαση Οκτώβριος 2017].

[42] Open Whisper Systems, «Signal Privacy Policy,» [Ηλεκτρονικό]. Available: https://signal.org/signal/privacy/. [Πρόσβαση Οκτώβριος 2017].

[43] N. Perlroth και K. Benner, «Subpoenas and Gag Orders Show Government Overreach, Tech Companies Argue,» 4 Οκτωβρίου 2016. [Ηλεκτρονικό]. Available: https://www.nytimes.com/2016/10/05/technology/subpoenas-and-gag-orders-show-government-overreach-tech-companies-argue.html. [Πρόσβαση Οκτώβριος 2017].

[44] Net Applications, «Mobile/Tablet Top Browser Share Trend,» [Ηλεκτρονικό]. Available: https://www.netmarketshare.com/browser-market-share.aspx?qprid=1&qpcustomb=1#. [Πρόσβαση Οκτώβριος 2017].

[45] Apple Inc., «Πολιτική απορρήτου της Apple,» 19 Σεπτεμβρίου 2017. [Ηλεκτρονικό]. Available: https://www.apple.com/legal/privacy/gr/. [Πρόσβαση Οκτώβριος 2017].

[46] R. Linus, «What every Browser knows about you,» [Ηλεκτρονικό]. Available: http://webkay.robinlinus.com/. [Πρόσβαση Οκτώβριος 2017].

[47] ASUSTeK Computer Inc., «ASUS Nexus 7,» [Ηλεκτρονικό]. Available: https://www.asus.com/gr/Tablets/Nexus_7/.

[48] S. Perez, «App Submissions On Google Play Now Reviewed By Staff, Will Include Age-Based Ratings,» 17 Μαρτίου 2015. [Ηλεκτρονικό]. Available: https://techcrunch.com/2015/03/17/app-submissions-on-google-play-now-reviewed-by-staff-will-include-age-based-ratings/. [Πρόσβαση Οκτώβριος 2017].

[49] Android Developers, «Requesting Permissions,» [Ηλεκτρονικό]. Available: https://developer.android.com/guide/topics/permissions/requesting.html. [Πρόσβαση Οκτώβριος 2017].

[50] R. Kitchin, «Getting smarter about smart cities: Improving data privacy and data,» Dublin, 2016.

[51] Android Developers, «Manifest.permission,» [Ηλεκτρονικό]. Available: https://developer.android.com/reference/android/Manifest.permission.html. [Πρόσβαση Οκτώβριος 2017].

[52] Apple Inc., «iOS Security,» Μάρτιος 2017. [Ηλεκτρονικό]. Available: https://www.apple.com/business/docs/iOS_Security_Guide.pdf. [Πρόσβαση Οκτώβριος 2017].

[53] Apple Inc., «App Store Review Guidelines,» [Ηλεκτρονικό]. Available: https://developer.apple.com/app-store/review/guidelines/. [Πρόσβαση Οκτώβριος 2017].

[54] J. Zang, K. Dummit, J. Graves, P. Lisker και L. Sweeney, «Who Knows What About Me? A Survey of Behind the Scenes Personal Data Sharing to Third Parties by Mobile Apps,» 30 Οκτωβρίου 2015. [Ηλεκτρονικό]. Available: https://techscience.org/a/2015103001/. [Πρόσβαση Οκτώβριος 2017].

[55] R. Triggs, «34% of you aren't even using a lockscreen password,» 21 Ιανουαρίου 2016. [Ηλεκτρονικό]. Available: https://www.androidauthority.com/psa-use-a-lockscreen-password-668689/. [Πρόσβαση Οκτώβριος 2017].

[56] S. Blum, «What's more secure: Pattern lock or a PIN?,» [Ηλεκτρονικό]. Available: https://www.androidpit.com/What-s-More-Secure-Pattern-Lock-or-a-PIN. [Πρόσβαση Οκτώβριος 2017].

[57] Polar Electro, «H6, H7, H10 and OH1 Heart rate sensors,» Polar, [Ηλεκτρονικό]. Available: https://developer.polar.com/wiki/H6,_H7,_H10_and_OH1_Heart_rate_sensors. [Πρόσβαση Οκτώβριος 2017].

[58] Garmin LTD, «Garmin Developer Documentation,» [Ηλεκτρονικό]. Available: https://developer.garmin.com/. [Πρόσβαση Οκτώβριος 2017].

[59] Node.js Project, «DoS Vulnerability (fixed in Node v0.8.26 and v0.10.21),» [Ηλεκτρονικό]. Available: https://nodejs.org/en/blog/vulnerability/http-server-pipeline-flood-dos/. [Πρόσβαση Οκτώβριος 2017].

[60] K. Schroeder, «Performance of Apache 2.4 with the event MPM compared to Nginx,» [Ηλεκτρονικό]. Available: http://www.eschrade.com/page/performance-of-apache-2-4-with-the-event-mpm-compared-to-nginx/. [Πρόσβαση Οκτώβριος 2017].

[61] IWF1, «Apache Vs Nginx Vs Node.js And What It Means About The Performance Of WordPress Vs Ghost,» [Ηλεκτρονικό]. Available: https://iwf1.com/apache-vs-nginx-vs-node-js-and-what-it-means-about-the-performance-of-wordpress-vs-ghost. [Πρόσβαση Οκτώβριος 2017].

[62] Crimestoppers, «Why contact Crimestoppers?,» [Ηλεκτρονικό]. Available: https://crimestoppers-uk.org/give-information/why-contact-crimestoppers/. [Πρόσβαση Οκτώβριος 2017].

[63] The European Parliament, «DIRECTIVE 2006/24/EC,» 15 Μαρτίου 2006. [Ηλεκτρονικό]. Available: http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2006:105:0054:0063:EN:PDF. [Πρόσβαση Οκτώβριος 2017].

[64] Ευρωπαϊκή Ένωση, «Χάρτης Θεμελιωδών Δικαιωμάτων της Ευρωπαϊκής Ένωσης,» 18 Δεκεμβρίου 2000. [Ηλεκτρονικό]. Available: http://www.europarl.europa.eu/charter/pdf/text_en.pdf. [Πρόσβαση Οκτώβριος 2017].

[65] Scheizerische Eidgenossenschaft, «Bundesgesetz betreffend die Überwachung des Post- und Fernmeldeverkehrs,» 7 Ιουλίου 2016. [Ηλεκτρονικό]. Available: https://www.admin.ch/opc/de/federal-gazette/2016/1991.pdf. [Πρόσβαση Οκτώβριος 2017].

[66] Canadian Center for Information on Missing Adults, «Crime Stoppers and Protection for Tipsters,» Δεκέμβριος 2012. [Ηλεκτρονικό]. Available: https://missingpersonsinformation.ca/resources/crime-stoppers-and-protection-for-tipsters/. [Πρόσβαση Οκτώβριος 2017].

[67] A. Ekblaw και A. Azaria, «MedRec: Medical Data Management on the Blockchain,» 19 Σεπτεμβρίου 2016. [Ηλεκτρονικό]. Available: https://www.pubpub.org/pub/medrec. [Πρόσβαση Οκτώβριος 2017].

[68] G. Zyskind, O. Nathan και A. Pentland, «Decentralizing Privacy: Using Blockchain to Protect Personal Data,» σε *Proceedings of the 2015 IEEE Security and Privacy Workshops*, 2015.

[69] C. Chatzigeorgiou, L. Toumanidis, D. Kogias, C. Patrikakis και E. Jacksch, «A communication gateway architecture for ensuring privacy and confidentiality in incident reporting,» σε *2017 IEEE 15th International Conference on Software Engineering Research, Management and Applications (SERA)*, London, UK, 2017.

ΠΑΡΑΡΤΗΜΑ 1

Στο παράρτημα αυτό βρίσκεται ο κώδικας της εφαρμογής Android που χρησιμοποιήθηκε για την αποστολή μηνυμάτων.

Ακολουθεί η υπηρεσία η οποία λαμβάνει τα πατήματα του κουμπιού.

```
public class KeepAliveService extends Service {

    private static final String APP_ID = "XXXXXXXXXXXX";
    private static final String APP_SECRET = "XXXXXXXXXXXX";

    @Override
    public IBinder onBind(Intent intent) {

        return null;
    }

    @Override
    public void onCreate() {

        super.onCreate();
    }

    @Override
    public int onStartCommand(Intent intent, int flags, int startId) {

        FlicManager.init(this.getApplicationContext(), APP_ID, APP_SECRET);

        for (FlicButton button : FlicManager.getManager().getKnownButtons()) {
            button.connect();
            setupEventListener(button);
        }

        return super.onStartCommand(intent, flags, startId);
    }

    public static class BootUpReceiver extends BroadcastReceiver {

        @Override
        public void onReceive(Context context, Intent intent) {
            context.startService(new Intent(context, KeepAliveService.class));
        }
    }
```

```
    public static class UpdateReceiver extends BroadcastReceiver {
        @Override
        public void onReceive(Context context, Intent intent) {
            context.startService(new Intent(context, KeepAliveService.class));
        }
    }

    private void setupEventListener(FlicButton button) {

        FlicButtonListener listener = new FlicButtonAdapter() {

            @Override
            public void onButtonSingleOrDoubleClickOrHold(FlicButton button,
                                                          boolean wasQueued,
                                                          int timeDiff,
                                                          boolean isSingleClick,
                                                          boolean isDoubleClick,
                                                          boolean isHold) {

                // If button was pressed before 5 seconds or more, ignore click
                if (wasQueued && timeDiff > 5) {
                    return;
                }

                if (isSingleClick) {
                    MessageHandler messageHandler = new MessageHandler(
                            getApplicationContext(),
                            MessageHandler.Click.SINGLE);
                    messageHandler.prepareMessage();
                } else if (isDoubleClick) {
                    MessageHandler messageHandler = new MessageHandler(
                            getApplicationContext(),
                            MessageHandler.Click.DOUBLE);
                    messageHandler.prepareMessage();
                } else if (isHold) {
                    MessageHandler messageHandler = new MessageHandler(
                            getApplicationContext(),
                            MessageHandler.Click.LONG);
                    messageHandler.prepareMessage();
                }
            }
        };
        button.addEventListener(listener);
    }

}
```

Η επόμενη κλάση αποθηκεύει τα στοιχεία που εισάγει ο χρήστης στη μνήμη της συσκευής.

```
public final class AppPreferences {

    public static void saveDescription(Context context, String title,
                                       String buttonAction) {

        SharedPreferences sharedPref = context.getSharedPreferences("global",
                Context.MODE_PRIVATE);
        SharedPreferences.Editor editor = sharedPref.edit();
        editor.putString(buttonAction, title);
        editor.apply();
    }


    public static String getDescription(Context context, String buttonAction) {

        SharedPreferences globalPrefs = context.getSharedPreferences("global",
                Context.MODE_PRIVATE);
        return globalPrefs.getString(buttonAction, null);
    }


    public static void saveTargetEmail(Context context, String target,
                                       String buttonAction) {

        SharedPreferences sharedPref = context.getSharedPreferences("global",
                Context.MODE_PRIVATE);
        SharedPreferences.Editor editor = sharedPref.edit();
        editor.putString(buttonAction, target);
        editor.apply();
    }


    public static String getTargetEmail(Context context, String buttonAction) {

        SharedPreferences globalPrefs = context.getSharedPreferences("global",
                Context.MODE_PRIVATE);
        return globalPrefs.getString(buttonAction, null);
    }

    public static void saveGPS(Context context, boolean gpsOn,
                               String buttonAction) {

        SharedPreferences sharedPref = context.getSharedPreferences("global",
                Context.MODE_PRIVATE);
        SharedPreferences.Editor editor = sharedPref.edit();
        editor.putBoolean(buttonAction, gpsOn);
        editor.apply();
    }

    public static boolean getGPS(Context context, String buttonAction) {

        SharedPreferences globalPrefs = context.getSharedPreferences("global",
                Context.MODE_PRIVATE);
        return globalPrefs.getBoolean(buttonAction, true);
    }

    public static void saveAnonymous(Context context, boolean anonymous,
                                     String buttonAction) {

        SharedPreferences sharedPref = context.getSharedPreferences("global",
                Context.MODE_PRIVATE);
        SharedPreferences.Editor editor = sharedPref.edit();
```

```
        editor.putBoolean(buttonAction, anonymous);
        editor.apply();
    }

    public static boolean getAnonymous(Context context, String buttonAction) {

        SharedPreferences globalPrefs = context.getSharedPreferences("global",
                Context.MODE_PRIVATE);
        return globalPrefs.getBoolean(buttonAction, true);
    }

    public static String getMyEmail(Context context, String buttonAction) {

        SharedPreferences globalPrefs = context.getSharedPreferences("global",
                Context.MODE_PRIVATE);
        return globalPrefs.getString(buttonAction, null);
    }

    public static void setMyEmail(Context context, String mailTarget,
                                  String buttonAction) {
        SharedPreferences sharedPref = context.getSharedPreferences("global",
                Context.MODE_PRIVATE);
        SharedPreferences.Editor editor = sharedPref.edit();
        editor.putString(buttonAction, mailTarget);
        editor.apply();
    }
}
```

Στη κλάση Constants είναι αποθηκευμένες κάποιες σταθερές για τη λειτουργία της εφαρμογής.

```
public final class Constants {

    public static final String CLICK = "Click";
    public static final String DOUBLE_CLICK = "Double Click";
    public static final String LONG_CLICK = "Long Click";

    public static final String BUTTON_DOWN = "button_down";
    public static final String BUTTON_LONG = "button_long";
    public static final String BUTTON_DOUBLE = "button_double";

    public static final String BUTTON_DOWN_DESCRIPTION =
            "button_down_description";
    public static final String BUTTON_LONG_DESCRIPTION =
            "button_long_description";
    public static final String BUTTON_DOUBLE_DESCRIPTION =
            "button_double_description";
    public static final String BUTTON_DOWN_GPS = "button_down_gps";
    public static final String BUTTON_LONG_GPS = "button_long_gps";
    public static final String BUTTON_DOUBLE_GPS = "button_double_gps";
    public static final String BUTTON_DOWN_ANONYMOUS = "button_down_anonymous";
    public static final String BUTTON_LONG_ANONYMOUS = "button_long_anonymous";
    public static final String BUTTON_DOUBLE_ANONYMOUS =
            "button_double_anonymous";
    public static final String BUTTON_DOWN_EMAIL_TARGET = "button_down_target";
    public static final String BUTTON_LONG_EMAIL_TARGET = "button_long_target";
    public static final String BUTTON_DOUBLE_EMAIL_TARGET =
            "button_double_target";
    public static final String BUTTON_DOWN_MY_MAIL = "button_down_mail_target";
    public static final String BUTTON_LONG_MY_MAIL = "button_long_mail_target";
    public static final String BUTTON_DOUBLE_MY_MAIL =
            "button_double_mail_target";


    public static final String ANONYMISER_ENDPOINT =
            "https://vm15.openstack.puas.gr/message";
    public static final String DEFAULT_ENDPOINT =
            "https://consert.puas.gr/flic/";

    public static final String CCS_PUBLIC_KEY =
            "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX";
}
```

Η επόμενη κλάση φτιάχνει το αντικείμενο του μηνύματος.

```
public class Message implements Serializable {

    private double userLongitude = 0;
    private double userLatitude = 0;
    private gr.puas.quickreport.model.Location userLocationAddress;
    private String myMail;
    private String description;
    private boolean isAnonymous = false;
    private String targetMail;
    private String dateString;

    public Message(Location location,
                   MuLocation userLocationAddress,
                   String myMail,
                   String description,
                   boolean isAnonymous,
                   String targetMail) {

        if (location != null) {
            this.userLongitude = location.getLongitude();
            this.userLatitude = location.getLatitude();
        }
        Date date = new Date();
        this.userLocationAddress = userLocationAddress;
        this.dateString = getLocalDate(date);
        this.myMail = myMail;
        this.description = description;
        this.isAnonymous = isAnonymous;
        this.targetMail = targetMail;
    }


    public JSONObject getJSON() {

        JSONObject message = new JSONObject();
        try {

            // Location of the user
            JSONObject userLocation = new JSONObject();
            userLocation.put("latitude", userLatitude);
            userLocation.put("longitude", userLongitude);
            userLocation.put("address", userLocationAddress.getAddress());
            userLocation.put("city", userLocationAddress.getCity());
            userLocation.put("postal_code", userLocationAddress.getPostalCode());

            // Prepare final JSON
            message.put("description", description);
            if (!isAnonymous) {
                message.put("user", myMail);
            }
            message.put("anonymous", isAnonymous);
            message.put("location", userLocation);
            message.put("eventDateTime", dateString);
            message.put("mailto", targetMail);

        } catch (JSONException e) {
            return message;
        }

        return message;
    }

    private static String getLocalDate(Date date) {
```

```
        Calendar c = Calendar.getInstance();
        c.setTime(date);
        DateFormat format = DateFormat.getInstance();
        return format.format(date);
    }
}
```

Η κλάση MessageHandler αναλαμβάνει να ανακτήσει τις ρυθμίσεις του χρήστη, να φτιάξει το JSON με τα δεδομένα και τελικά να τα στείλει.

```
public class MessageHandler {

    private Context context;

    private boolean anonymous = false;
    private boolean gps = true;
    private String targetEmail = "";
    private String myEmail = "";
    private String description = "";

    private String anonymousPref;
    private String gpsPref;
    private String targetMail;
    private String descriptionPref;
    private String myMail;


    public enum Click {
        SINGLE,
        LONG,
        DOUBLE
    }

    public MessageHandler(Context context, Click action) {

        this.context = context;

        switch (action) {
            case SINGLE:
                anonymousPref = Constants.BUTTON_DOWN_ANONYMOUS;
                gpsPref = Constants.BUTTON_DOWN_GPS;
                targetMail = Constants.BUTTON_DOWN_EMAIL_TARGET;
                descriptionPref = Constants.BUTTON_DOWN_DESCRIPTION;
                myMail = Constants.BUTTON_DOWN_MY_MAIL;
                break;
            case LONG:
                anonymousPref = Constants.BUTTON_LONG_ANONYMOUS;
                gpsPref = Constants.BUTTON_LONG_GPS;
                targetMail = Constants.BUTTON_LONG_EMAIL_TARGET;
                descriptionPref = Constants.BUTTON_LONG_DESCRIPTION;
                myMail = Constants.BUTTON_LONG_MY_MAIL;
                break;
            case DOUBLE:
                anonymousPref = Constants.BUTTON_DOUBLE_ANONYMOUS;
                gpsPref = Constants.BUTTON_DOUBLE_GPS;
                targetMail = Constants.BUTTON_DOUBLE_EMAIL_TARGET;
                descriptionPref = Constants.BUTTON_DOUBLE_DESCRIPTION;
                myMail = Constants.BUTTON_DOUBLE_MY_MAIL;
                break;
        }
    }

    public void prepareReport() {
        try {
            gps = AppPreferences.getGPS(context, gpsPref);
            anonymous = AppPreferences.getAnonymous(context, anonymousPref);
            targetEmail = AppPreferences.getTargetEmail(context, targetMail);
            myEmail = AppPreferences.getMyEmail(context, myMail);
            description = AppPreferences.getDescription(context, descriptionPref);

            if (gps) {
```

```
                getLocation();
            } else {
                android.location.Location location =
                        new android.location.Location("");
                location.setLatitude(0.0);
                location.setLongitude(0.0);
                constructMessage(location);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    private void getLocation() {
        if (!isLocationEnabled(context)) {
            showNotification(context, "Please enable location first");
            return;
        }

LocalBroadcastManager.getInstance(context).registerReceiver(mMessageReceiver,
                new IntentFilter(LOCATION_SERVICE));
        Intent intent = new Intent(context, LocationService.class);
        context.startService(intent);
    }

    private final BroadcastReceiver mMessageReceiver = new BroadcastReceiver() {
        @Override
        public void onReceive(Context context, Intent intent) {
            // Get extra data included in the Intent
            LocalBroadcastManager.getInstance(context).unregisterReceiver(this);
            android.location.Location location =
                    intent.getParcelableExtra(LocationService.LOCATION_EXTRA);
            constructMessage(location);
        }
    };


    private void constructMessage(android.location.Location location) {

        Location address = getAddress(location, context);

        Message message = new Message(location, address, myEmail, description,
                anonymous, targetEmail);
        JSONObject messageJSON = message.getJSON();

        JSONObject finalPayload;

        try {
            if (anonymous) {
                finalPayload = Anonymous.getPayload(messageJSON.toString());
            } else {
                finalPayload = messageJSON;
            }
            new SendReport(context, anonymous).execute(finalPayload);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    private class SendMessage extends AsyncTask<JSONObject, Void, List<String>> {

        private Context context;
        private boolean anonymous;

        SendReport(Context context, boolean anonymous) {
```

```
            this.context = context;
            this.anonymous = anonymous;
        }

        @Override
        public List<String> doInBackground(JSONObject... params) {

            if (anonymous) {
                return HTTP.post(params[0], Constants.ANONYMISER_ENDPOINT);
            } else {
                return HTTP.post(params[0], Constants.DEFAULT_ENDPOINT);
            }
        }

        @Override
        public void onPostExecute(List<String> result) {
            if (result.size() > 0 && result.get(0).equals("200")) {
                showNotification(context, "Message submitted");
            } else {
                showNotification(context, "Error submitting message");
            }
        }
    }

    private static void showNotification(Context context, String message) {
        String title = context.getString(R.string.app_name);

        NotificationCompat.Builder mBuilder =
                new NotificationCompat.Builder(context)
                        .setSmallIcon(R.mipmap.ic_launcher)
                        .setContentTitle(title)
                        .setContentText(message)
                        .setAutoCancel(true);

        Uri alarmSound =
                RingtoneManager.getDefaultUri(RingtoneManager.TYPE_NOTIFICATION);
        mBuilder.setSound(alarmSound);

        NotificationManager mNotificationManager =
                (NotificationManager)
                        context.getSystemService(Context.NOTIFICATION_SERVICE);
        Random random = new Random();
        mNotificationManager.notify(random.nextInt(), mBuilder.build());
    }

    private static boolean isLocationEnabled(Context context) {
        int locationMode;
        String locationProviders;

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.KITKAT) {
            try {
                locationMode =
                        Settings.Secure.getInt(context.getContentResolver(),
                                Settings.Secure.LOCATION_MODE);
            } catch (Settings.SettingNotFoundException e) {
                e.printStackTrace();
                return false;
            }

            return locationMode != Settings.Secure.LOCATION_MODE_OFF;

        } else {
            locationProviders = Settings.Secure.getString(
                    context.getContentResolver(),
                    Settings.Secure.LOCATION_PROVIDERS_ALLOWED);
```

```
            return !TextUtils.isEmpty(locationProviders);
        }
    }

    private static MyLocation getAddress(
            android.location.Location location, Context context) {

        MyLocation stringLocation;
        Geocoder geocoder;
        List<Address> addresses;
        geocoder = new Geocoder(context, Locale.getDefault());

        try {
            // Here 1 represent max location result to returned
            if (location == null)
                return new MyLocation();
            addresses = geocoder.getFromLocation(location.getLatitude(),
                    location.getLongitude(), 1);

            if (addresses.size() > 0) {
                // If any additional address line present than only,
                // check with max available address lines by
getMaxAddressLineIndex()
                String address = addresses.get(0).getAddressLine(0);
                String city = addresses.get(0).getLocality();
                String postalCode = addresses.get(0).getPostalCode();

                stringLocation = new MyLocation(address, city, postalCode);
                return stringLocation;
            }
        } catch (IOException e) {
            stringLocation = new MyLocation();
            return stringLocation;
        }
        return new MyLocation();
    }
}
```

Η τελευταία κλάση ασχολείται με το να παρουσιάσει την οθόνη όπου μπορεί ο χρήστης να αλλάξει τις ρυθμίσεις της εφαρμογής.

```
public class ClickSettingsActivity extends AppCompatActivity {

    private EditText txtDescription;
    private Switch toggleGPS;
    private Switch toggleAnonymous;
    private EditText txtTargetEmail;
    private EditText txtMyEmail;
    private LinearLayout mailLayout;

    private String buttonDescription = "";
    private String buttonGPS = "";
    private String buttonAnonymous = "";
    private String buttonEmailTarget = "";
    private String myEmailPref = "";

    private String activityTitle = "";

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_click_settings);

        String buttonAction;

        if (getIntent().getExtras() != null) {
            buttonAction = getIntent().getStringExtra("buttonAction");
            findElements();
            setStringNames(buttonAction);
            setupView();
            activityTitle = activityTitle + " " + getString(R.string.settings);
            setTitle(activityTitle);
        } else {
            finish();
        }

        toggleAnonymous.setOnCheckedChangeListener(
                new CompoundButton.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(CompoundButton compoundButton, boolean b)
{
                if (b) {
                    mailLayout.setVisibility(View.VISIBLE);
                } else {
                    mailLayout.setVisibility(View.INVISIBLE);
                }
            }
        });
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        MenuInflater inflater = getMenuInflater();
        inflater.inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {

        // Handle item selection
```

```
        switch (item.getItemId()) {
            case R.id.save:
                saveSettings();
                finish();
                return true;
            default:
                return super.onOptionsItemSelected(item);
        }
    }

    private void findElements() {

        txtDescription = (EditText) findViewById(R.id.txtDescription);
        toggleGPS = (Switch) findViewById(R.id.toggleGPS);
        toggleAnonymous = (Switch) findViewById(R.id.toggleAnonymous);
        txtTargetEmail = (EditText) findViewById(R.id.txtTarget);
        txtMyEmail = (EditText) findViewById(R.id.txtMyEmail);
        mailLayout = (LinearLayout) findViewById(R.id.mailLayout);
    }

    private void setStringNames(String buttonAction) {

        switch (buttonAction) {
            case Constants.BUTTON_DOWN:
                buttonDescription = Constants.BUTTON_DOWN_DESCRIPTION;
                buttonGPS = Constants.BUTTON_DOWN_GPS;
                buttonAnonymous = Constants.BUTTON_DOWN_ANONYMOUS;
                buttonEmailTarget = Constants.BUTTON_DOWN_EMAIL_TARGET;
                activityTitle = Constants.CLICK;
                myEmailPref = Constants.BUTTON_DOWN_MY_MAIL;
                break;
            case Constants.BUTTON_LONG:
                buttonDescription = Constants.BUTTON_LONG_DESCRIPTION;
                buttonGPS = Constants.BUTTON_LONG_GPS;
                buttonAnonymous = Constants.BUTTON_LONG_ANONYMOUS;
                buttonEmailTarget = Constants.BUTTON_LONG_EMAIL_TARGET;
                activityTitle = Constants.LONG_CLICK;
                myEmailPref = Constants.BUTTON_LONG_MY_MAIL;
                break;
            case Constants.BUTTON_DOUBLE:
                buttonDescription = Constants.BUTTON_DOUBLE_DESCRIPTION;
                buttonGPS = Constants.BUTTON_DOUBLE_GPS;
                buttonAnonymous = Constants.BUTTON_DOUBLE_ANONYMOUS;
                buttonEmailTarget = Constants.BUTTON_DOUBLE_EMAIL_TARGET;
                activityTitle = Constants.DOUBLE_CLICK;
                myEmailPref = Constants.BUTTON_DOUBLE_MY_MAIL;
                break;
            default:
                finish();
                break;
        }
    }

    private void setupView() {

        boolean isAnonymous = AppPreferences.getAnonymous(this, buttonAnonymous);
        txtDescription.setText(AppPreferences.getDescription(this,
buttonDescription));
        toggleGPS.setChecked(AppPreferences.getGPS(this, buttonGPS));
        toggleAnonymous.setChecked(isAnonymous);
        txtTargetEmail.setText(AppPreferences.getTargetEmail(this,
buttonEmailTarget));
        if (isAnonymous) {
            mailLayout.setVisibility(View.VISIBLE);
            txtMyEmail.setText(AppPreferences.getMyEmail(this, myEmailPref));
```

```
        }
    }

    private void saveSettings() {

        String description = txtDescription.getText().toString();
        boolean gpsOn = toggleGPS.isChecked();
        boolean anonymous = toggleAnonymous.isChecked();
        String target = txtTargetEmail.getText().toString();
        String myEmail = txtMyEmail.getText().toString();

        AppPreferences.saveDescription(this, description, buttonDescription);
        AppPreferences.saveGPS(this, gpsOn, buttonGPS);
        AppPreferences.saveAnonymous(this, anonymous, buttonAnonymous);
        AppPreferences.saveTargetEmail(this, target, buttonEmailTarget);

        if (!anonymous) {
            AppPreferences.setMyEmail(this, myEmail, myEmailPref);
        }
    }
}
```

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fillViewport="true"
    android:gravity="center_horizontal">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:paddingLeft="@dimen/activity_horizontal_margin"
            android:paddingRight="@dimen/activity_horizontal_margin"
            android:paddingTop="@dimen/activity_vertical_margin">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content">

                <TextView
                    android:layout_width="110dp"
                    android:layout_height="wrap_content"
                    android:text="Send Location"
                    android:textColor="@android:color/black"
                    android:textSize="20sp" />

                <Switch
                    android:id="@+id/toggleGPS"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_gravity="end" />
            </LinearLayout>

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content">

                <TextView
                    android:layout_width="110dp"
                    android:layout_height="wrap_content"
                    android:text="Description"
                    android:textColor="@android:color/black"
                    android:textSize="20sp" />

                <EditText
                    android:id="@+id/txtDescription"
                    android:layout_width="match_parent"
                    android:layout_height="wrap_content"
                    android:lines="4"
                    android:maxLength="2000" />
            </LinearLayout>

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content">

                <TextView
                    android:layout_width="110dp"
                    android:layout_height="wrap_content"
                    android:text="Target mail"
                    android:textColor="@android:color/black"
                    android:textSize="20sp" />

                <EditText
                    android:id="@+id/txtTarget"
```

```
                android:layout_width="match_parent"
                android:layout_height="wrap_content" />
        </LinearLayout>

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <TextView
                android:layout_width="110dp"
                android:layout_height="wrap_content"
                android:text="Anonymous"
                android:textColor="@android:color/black"
                android:textSize="20sp" />

            <Switch
                android:id="@+id/toggleAnonymous"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:checked="false" />

        </LinearLayout>

        <LinearLayout
            android:id="@+id/mailLayout"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:visibility="invisible">

            <TextView
                android:layout_width="110dp"
                android:layout_height="wrap_content"
                android:text="My Email"
                android:textColor="@android:color/black"
                android:textSize="20sp" />

            <EditText
                android:id="@+id/txtMyEmail"
                android:layout_width="match_parent"
                android:layout_height="wrap_content" />
        </LinearLayout>

    </LinearLayout>

</ScrollView>
```

ΠΑΡΑΡΤΗΜΑ 2

Ο κώδικας του εξυπηρετητή χρειάζεται τη βιβλιοθήκη phpmailer για να λειτουργήσει και υποθέτει ότι ο φάκελός της βρίσκεται στον ίδιο φάκελο με τα τρία php αρχεία του εξυπηρετητή.

Στο αρχείο config.php εισάγονται τα στοιχεία της ηλεκτρονικής διεύθυνσης από όπου θα σταλούν τα μηνύματα.

```
<?php

define('KEY_ANONYMOUS', 'anonymous');
define('KEY_USER','user');
define('KEY_DESCRIPTION','description');
define('KEY_EVENT_DATETIME','eventDateTime');
define('KEY_LOCATION','location');
define('KEY_LOCATION_UNKNOWN','unknown');
define('KEY_LOCATION_LATITUDE','latitude');
define('KEY_LOCATION_LONGITUDE','longitude');
define('KEY_LOCATION_ADDRESS','address');


define('MAIL_USER','user@example.com');
define('MAIL_PASSWORD','very_secret');
define('MAIL_NAME','User First Name - Last Name');
define('MAIL_SUBJECT','Mail Subject');
define('SMTP_AUTH',true);
define('SMTP_SECURE','tls');
define('SMTP_HOST', 'smtp.gmail.com');
define('SMTP_PORT',587);
```

Το δεύτερο αρχείο αναλαμβάνει να συντάξει και να στείλει το μήνυμα ηλεκτρονικού ταχυδρομείου.

```
<?php

require __DIR__.'/phpmailer/PHPMailerAutoload.php';
require_once  __DIR__.'/config.php';

/**
 * Sends email using smtp (phpmailer)
 *
 * @param array: $to: the recipient(s)
 * @param string: $from: the sender's email
 * @param string: $from_name: the sender's name
 * @param string: $subject: the mail's subject
 * @param string: $body: the mail's body
 * @return bool: if mail is successfully sent
 */
function sendSmtpMail($to, $from, $from_name, $subject, $body) {
   $mail = new PHPMailer();
   $mail->CharSet = 'UTF-8';
   $mail->isSMTP();
   $mail->SMTPDebug = 0;
   $mail->SMTPAuth = SMTP_AUTH;
   $mail->SMTPSecure = SMTP_SECURE;
   $mail->Host = SMTP_HOST;
   $mail->Port = SMTP_PORT;
   $mail->Username = MAIL_USER;
   $mail->Password = MAIL_PASSWORD;
   $mail->setFrom($from, $from_name);
   $mail->Subject = $subject;
   $mail->Body = $body;
   foreach ($to as $resp) {
      $mail->addAddress($resp);
   }
   if(!$mail->send()) {
      return false;
   } else {
      return true;
   }
}

/**
 * Returns the address or the coordinates or null if any of the above found
 * in the input array
 *
 * @param $locationArray
 * @return null|string
 */
function getLocationString($locationArray) {
   $location = null;
   if (array_key_exists(KEY_LOCATION_ADDRESS, $locationArray)) {
      $address = (string)$locationArray[KEY_LOCATION_ADDRESS];
      if (strtolower($address) !== KEY_LOCATION_UNKNOWN) {
         $location = $locationArray[KEY_LOCATION_ADDRESS];
      }
   }
```

```
    if (is_null($location) &&
        array_key_exists(KEY_LOCATION_LATITUDE, $locationArray) &&
        array_key_exists(KEY_LOCATION_LONGITUDE,$locationArray)) {
        if (floatval($locationArray[KEY_LOCATION_LATITUDE]) !== 0 &&
            floatval($locationArray[KEY_LOCATION_LONGITUDE]) != 0) {
            $location = KEY_LOCATION_LATITUDE.': '.
                                        $locationArray[KEY_LOCATION_LATITUDE] .
                ', '.KEY_LOCATION_LONGITUDE. ': '.
                                        $locationArray[KEY_LOCATION_LONGITUDE];
        }
    }
    return $location;
}

/**
 * Returns a string containing the body of a message to be sent
 *
 * @param $data: the array that was posted
 * @return string
 */
function getMailBody($data) {
    $anonymous = null;
    $description = null;
    $user = null;
    $location = null;
    $dateTime = null;

    foreach ($data as $key => $value) {
        switch ($key) {
            case KEY_ANONYMOUS:
                if ($value) {
                    $anonymous = boolval($value) || null;
                }
                continue;
            case KEY_USER:
                if ($value) {
                    $user = (string)$value;
                    if (is_null($anonymous)) {
                        $anonymous = null;
                    }
                }
                continue;
            case KEY_DESCRIPTION:
                if ($value) {
                    $description = $value;
                }
                continue;
            case KEY_EVENT_DATETIME:
                if ($value) {
                    $dateTime = (string)$value;
                }
                continue;
            case KEY_LOCATION:
                if ($value) {
                    $postedLocation = $value;
                    if (is_object($value)){
                        $postedLocation = (array)($value);
                    }
```

```
                $location = getLocationString($postedLocation);
            } else {
                $location = null;
            }
        }
    }
    $str = $anonymous ? 'A user with hidden id' : 'The user with the email '

                        .$user. ',';
    $str .= PHP_EOL.'has sent on ';
    $str .= $dateTime ? (string)$dateTime : date('Y-m-d H:i:s');
    $str .= ','.PHP_EOL.'from ';
    $str .= $location ? 'location: ' . $location : 'an unknown location,';
    $str .= PHP_EOL. 'the message: '.PHP_EOL. "".$description . "".PHP_EOL;
    return $str;
}
```

Τέλος, το αρχείο index.php δέχεται τα αιτήματα POST με τα δεδομένα

```
<?php
require_once __DIR__.'/lib.php';

header('Content-Type: application/json');
$target = __DIR__.'/message.json';

$data = [];
$json = json_decode(file_get_contents('php://input', true));

// ensure we have a json file to read it's contents
if (!file_exists($target)) {
    file_put_contents($target,json_encode($data, JSON_UNESCAPED_UNICODE));
}

if ($json) {
    // close the connection before processing
    ob_end_clean();
    header("Connection: close");
    ignore_user_abort(); // optional
    ob_start();
    if ($json) {
        echo json_encode(['success' => true]);
    } else {
        $data = file_get_contents($target);
        echo $data;
    }
    $size = ob_get_length();
    header("Content-Length: $size");
    ob_end_flush();
    flush();
    if (PHP_SAPI === 'fpm-fcgi') {
        fastcgi_finish_request();
    }
    // Do processing here
    $data = (array) $json;
    $json_data = json_encode($data, JSON_UNESCAPED_UNICODE|JSON_PRETTY_PRINT);
    file_put_contents($target,$json_data);
}
if (array_key_exists('mailto', $data) ) {
    $mailto = [$data['mailto']];
    $toSend = getMailBody($data);
    sendSmtpMail($mailto, MAIL_USER, MAIL_NAME, MAIL_SUBJECT, $toSend);
} else {
    $data = file_get_contents($target);
    echo $data;
}
if ($json) {
    $data = (array) $json;
    $data['log_datetime'] = date('Y-m-d H:i:s');
    $json_data = json_encode($data, JSON_UNESCAPED_UNICODE);
    logData($json_data);
}
```

ΠΑΡΑΡΤΗΜΑ 3

Ελληνικός όρος	Αρχικά	Αρχικά	Αγγλικός όρος
Αλυσίδα μπλοκ			Blockchain
Άμεσο μήνυμα	ΑΜ	IM	Instant message
Αμμόλακκος			Sandbox
Αρχική Μετάθεση	ΑΜ	IP	Initial Permutation
Ασφάλεια Επιπέδου Μεταφοράς	ΑΕΜ	TLS	Transport Layer Security
Διεπαφή Ανάπτυξης Εφαρμογής	ΔΑΕ	API	Application Programming Interface
Δρομολόγηση κρεμμυδιού			Onion Routing
Εικονικό Ιδιωτικό Δίκτυο	ΕΙΔ	VPN	Virtual Private Network
Έξυπνο ρολόι			Smart watch
Έξυπνο τηλέφωνο			Smart phone
Επαναπροωθητής ηλεκτρονικής αλληλογραφίας			Remailer
Καταγραφέας δραστηριότητας			Activity tracker
Κινητός υπολογισμός			Mobile computing
Κρυπτογράφηση από Άκρο σε Άκρο	ΚαΑσΑ	E2EE	End to End Encryption
Κρυπτογραφία Διατήρησης Μορφής	ΚΔΜ	FPE	Format Preserving Encryption
Κρυπτοκείμενο			Ciphertext
Κρυπτοσύστημα ροής			Stream cipher
Κρυπτοσύστημα τμήματος			Block cipher
Πίνακας κατάστασης			State matrix
Πρότυπο Κρυπτογράφησης Δεδομένων	ΠΚΔ	DES	Data encryption standard
Προχωρημένο Πρότυπο Κρυπτογράφησης	ΠΠΚ	AES	Advanced Encryption Standard
Πρωτόκολλο Σήραγγας Ομότιμων Κόμβων	ΠΣΟΚ	PPTP	Peer-to-Peer Tunneling Protocol
Σύστημα Ψυχαγωγίας Αυτοκινήτου	ΣΨΑ	ICE	In-car entertainment
Ταμπλέτα			Tablet
Φορετή συσκευή			Wearable device
Φυλλομετρητής			Browser