Ατομική Διπλωματική Εργασία

ΑΝΑΛΥΣΗ ΙΔΙΩΤΙΚΟΤΗΤΑΣ ΣΕ ΚΙΝΗΤΕΣ ΕΦΑΡΜΟΓΕΣ ANDROID

Μόδεστος Ιωάννου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2018

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Ανάλυση ιδιωτικότητας σε
κινητές εφαρμογές Android

Μόδεστος Ιωάννου

Επιβλέπουσα Καθηγήτρια
Γεωργία Καπιτσάκη

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2018

Ευχαριστίες

Με την ευκαιρία που μου δόθηκε για την εκπόνηση της παρούσας Ατομικής Διπλωματικές Εργασίας θα ήθελα να ευχαριστήσω την επιβλέπουσα καθηγήτρια μου Δρ. Γεωργία Καπιτσάκη, η οποία με εμπιστεύτηκε και μου έδωσε την δυνατότητα να εργαστώ στο συγκεκριμένο θέμα καθώς επίσης και για τη συνεχή στήριξη, καθοδήγηση και ενθάρρυνση που μου παρείχε από την αρχή μέχρι το τέλος της Ατομικής Διπλωματικής Εργασίας.

Παράλληλα θα ήθελα να ευχαριστήσω την οικογένεια, τους φίλους και τους συμφοιτητές μου που ήταν πάντα δίπλα μου και με βοήθησαν να ανταπεξέλθω σε όλες τις δυσκολίες που αντιμετώπισα και την ηθική υποστήριξη που μου παρείχαν καθ' όλη τη διάρκεια των τελευταίων χρόνων και ιδιαιτέρα κατά τη διάρκεια εκπόνησης της παρούσας διπλωματικής εργασίας.

Τέλος, θα ήθελα να ευχαριστήσω το Πανεπιστήμιο Κύπρου που μου πρόσφερε τη δυνατότητα να υλοποιήσω το όνειρο μου, που δεν ήταν άλλο από το να τελειώσω τις προπτυχιακές μου σπουδές στον κλάδο της Πληροφορικής. Στα τέσσερα αυτά χρόνια των σπουδών μου κατάφερα να αποκτήσω εμπειρίες και γνώσεις που θα μου είναι χρήσιμες καθ' όλη τη διάρκεια της μετέπειτα ζωής μου.

Περίληψη

Στη συγκεκριμένη ατομική διπλωματική εργασία, αρχικά γίνεται μια γενική εισαγωγή σε θέματα ιδιωτικότητας και επεξηγούνται διάφορες έννοιες όπως της ιδιωτικότητας, των προσωπικών και των ευαίσθητων προσωπικών δεδομένων. Επίσης παρουσιάζονται διάφοροι κίνδυνοι ιδιωτικότητας που υπάρχουν και ακόμα αναφέρεται ο Κανονισμός Γενικής Προστασίας Δεδομένων.

Στην συνέχεια περιγράφεται το λειτουργικό σύστημα Android και παρουσιάζονται διάφορες προηγούμενες εργασίας που έχουν γίνει που αφορούν την ανάλυση ιδιωτικότητας σε κινητές εφαρμογές Android, όπως επίσης και υπάρχων συστήματα ανάλυσης εφαρμογών Android.

Ακόλουθος παρουσιάζεται η διαδικασία ανάλυσης όπως επίσης και υλοποίησης του συστήματος, κάνοντας αναφορά και στις τεχνολογικές γνώσεις και εργαλεία που χρησιμοποιούνται για την υλοποίηση του συστήματος.

Με το τέλος της σχεδίασης του συστήματος, παρουσιάζεται μια ανάλυση και τα αποτελέσματα της ανάλυσης αυτής, η οποία έγινε από ένα σύνολο εφαρμογών οι οποίες συλλέχτηκαν και αναλυθήκαν.

Τέλος, δίνεται ο επίλογος με την ανάλυση των συμπερασμάτων που προκύπτουν από την μελέτη του θέματος και προτείνονται κάποιες εισηγήσεις για μελλοντική εργασία.

Περιεχόμενα

Κεφάλαιο 1

Εισαγωγή

1.1 Γενική Εισαγωγή

Στην εποχή που ζούμε είναι γεγονός ότι οι άνθρωπο είναι εξαρτημένοι από την τεχνολογία, καθώς για πολλές καθημερινές εργασίες χρειάζεται η χρήση κάποιου πληροφοριακού συστήματος και κάποιας ηλεκτρονικής συσκευής. Αυτό οφείλεται στην ραγδαία ανάπτυξη της τεχνολογίας η οποία μας διευκολύνει στις καθημερινές μας εργασίες, μας βοηθά στην ψυχαγωγία και στην επικοινωνία. Πλέον το κινητό μας έχει εξελιχθεί σε μια προέκταση του χεριού μας και δεν μπορούμε στιγμή χωρίς το διαδίκτυο. Οι διάφορες συσκευές και εφαρμογές που χρησιμοποιούμε καθημερινά συνήθως χρειάζονται πληροφορίες από τους χρήστες για να μπορούν να λειτουργήσουν. Αυτές οι πληροφορίες τις περισσότερες φορές σχετίζονται με τον χρήστη που συνήθως αφορούν προσωπικά στοιχεία όπως όνομα, ηλικία, χώρα γι' αυτό και θεωρούνται ευαίσθητα δεδομένα. Επίσης σε αυτή την κατηγορία συμπεριλαμβάνονται και οι πληροφορίες που αφορούν το περιβάλλον στο οποίο βρίσκεται ο χρήστης εκείνη την στιγμή όπως είναι η τοποθεσία του. Οι πληροφορίες αυτές μπορεί να είναι απαραίτητες σε κάποιες εφαρμογές, όμως για κάποιες άλλες μπορεί να μην χρειάζονται για την λειτουργία τους.

Γιατί λοιπόν κάποιες εφαρμογές ζητούν πληροφορίες και δεδομένα από τους χρήστες, αφού δεν τις χρειάζονται για λειτουργικό σκοπό; Αυτό συμβαίνει διότι μεγάλες εταιρίες και οργανισμού δίνουν μεγάλα χρηματικά ποσά για την απόκτηση αυτών πληροφορίων. Οι πληροφορίες αυτές χρησιμοποιούνται από της εταιρίες και τους οργανισμούς αυτούς για να κατανοήσουν τους χρήστες, να δουν τις προτιμήσεις τους ή για άλλο σκοπό. Γι' αυτό τον

λόγο πρέπει να κατανοήσουμε τη σημασία των πληροφορίων που δίνουμε ως χρήστες και να αντιλαμβανόμαστε τον λόγο που χρειάζονται οι πληροφορίες αυτές. Θα πρέπει πάντα να σκεφτόμαστε το αν είναι απαραίτητο να δοθούν αυτές οι πληροφορίες.

Τα συστήματα που ζητούν αυτές τις πληροφορίες παρουσιάζουν μέσω ενημερωτικών εγγράφων και μηνυμάτων στους χρήστες, ποια δεδομένα χρειάζονται και για πιο σκοπό. Αυτά τα έγγραφα και τα μηνύματα όμως τις περισσότερες φορές είναι αρκετά μεγάλα σε μέγεθος, εμφανίζονται σε στιγμή που ο χρήστης κάνει κάτι άλλο ή εμφανίζονται σε σημείο που ο χρήστης δεν μπορεί να καταλάβει το σκοπό του. Αυτό έχει ως αποτέλεσμα ο χρήστης είτε να αγνοήσει τα έγγραφα και μηνύματα αυτά ή ακόμα και να πατήσει ότι τα διάβασε, χωρίς να το κάνει, απλά και μόνο για να τα ξεφορτωθεί από την οθόνη του. Με την εισαγωγή όμως του Κανονισμού Γενικής Προστασίας Δεδομένων (GDPR), πλέον σε όλα αυτά τα συστήματα θα πρέπει τα μηνύματα αυτά, να εμφανίζονται και να επεξηγούνται με εύκολο και κατανοητό τρόπο. Θα είναι απαραίτητο από τα συστήματα, να ζητούν την αποδοχή του χρήστη για την συλλογή και επεξεργασία πληροφορίων όπως επίσης και πρέπει ανά πάσα στιγμή ο χρήστης να μπορεί να έχει έλεγχο πάνω στα δεδομένα αυτά που έχει αποθηκεύσει το σύστημα για αυτών.

Στις κινητές εφαρμογές το φαινόμενο αυτό φαίνεται να συμβαίνει αρκετά συχνά. Συγκεκριμένα στις εφαρμογές Android, όταν μια εφαρμογή χρειάζεται πρόσβαση σε ευαίσθητα δεδομένα ή σε λειτουργίες του συστήματος, εμφανίζεται ένα μήνυμα στο οποίο ο χρήστης θα πρέπει να δώσει πρόσβαση σε αυτά τα δεδομένα. Το μήνυμα αυτό είτε θα εμφανίζεται κατά την διάρκεια της εγκατάστασης μιας εφαρμογής ή κατά την εκτέλεση της. Οι χρήστες αρκετές φορές δεν δίνουν σημασία στα μηνύματα αυτά αφού το μόνο που τους ενδιαφέρει είναι να χρησιμοποιήσουν την εφαρμογή. Κάτι τέτοιο ισχύει και για τις διαδικτυακές εφαρμογές, όπως για παράδειγμα στις ιστοσελίδες, που μπορεί να γίνεται χρήση των cookies για συλλογή κάποιων πληροφοριών ή να ζητείται πρόσβαση ακόμα και σε άλλες πληροφορίες όπως η τοποθεσία του χρήστη. Αυτό οδηγεί στο πρόβλημα ότι οι χρήστες εν αγνοία τους μπορεί να παρέχουν δεδομένα και πληροφορίες που οι ίδιοι δεν θέλουν να παρέχουν εξαρχής σε μια εφαρμογή που προσπαθούν να χρησιμοποιήσουν.

Μια εφαρμογή ακόμα μπορεί να χρησιμοποιεί βιβλιοθήκες οι οποίες αυτές με την σειρά τους μπορεί να ζητούν πρόσβαση σε κάποια δεδομένα. Οι χρήστες θα πρέπει με κάποιο τρόπο να γνωρίζουν τι ακριβώς δεδομένα χρειάζεται η εφαρμογή, ανεξάρτητα αν τα χρησιμοποιεί με έμμεσο ή άμεσο τρόπο. Οι προγραμματιστές των εφαρμογών θα πρέπει να συμμορφώνονται και να δηλώνουν ποια δεδομένα και που χρειάζεται πρόσβαση μια εφαρμογή. Επίσης,

κυκλοφορούν διάφορες εφαρμογές στο διαδίκτυο και πολύ εύκολα μπορεί κάποιος να κατεβάσει και να εγκαταστήσει κάποια από αυτές, χωρίς να γνωρίζει εάν είναι επικίνδυνη.

1.2 Στόχος Διπλωματικής Εργασίας

Με όσα αναφέρθηκαν πιο πάνω και την σημαντικότητα των ιδιωτικών και ευαίσθητων δεδομένων, θα ήταν καλό να υπάρχει ένας τρόπος να μπορεί ο χρήστης να δει επιπρόσθετες πληροφορίες για την συμπεριφορά μιας εφαρμογής, τα δεδομένα που χρειάζεται και την πρόσβαση σε πόρους του συστήματος που χρειάζεται για να λειτουργήσει. Αυτές οι πληροφορίες είναι ιδιαίτερα χρήσιμες και για τους προγραμματιστές εφαρμογών για να μπορούν να αντιληφθούν κατά πόσο η εφαρμογή τους ακολουθεί κάποιες καλές πρακτικές αναφορικά με τη διαχείριση των δεδομένων των χρηστών.

Στόχος αυτής της ατομικής διπλωματικής εργασίας είναι η ανάπτυξη ενός συστήματος το οποίο θα αναλύει μια κινητή εφαρμογή σε Android. Οι πληροφορίες που θα εξάγονται από την ανάλυση αυτή θα είναι γενικές πληροφορίες για την εφαρμογή, όπως έκδοση Android που χρησιμοποιεί, όνομα πακέτου καθώς και θα υπολογίζεται το SHA256 hash, το οποίο θα είναι μοναδικό ανά εφαρμογή. Στην συνέχεια θα ελέγχονται τα permissions που έχει δηλώσει η εφαρμογή και ποια από αυτά χρησιμοποιούνται ή όχι, ποια δεν είναι δηλωμένα αλλά χρησιμοποιούνται καθώς και ποια χρησιμοποιούνται από της βιβλιοθήκες που χρησιμοποιεί η εφαρμογή. Επιπλέον θα γίνεται έλεγχος των βιβλιοθηκών, για τυχόν εύρεση βιβλιοθήκης που πιθανόν να συλλέγει δεδομένα από την εφαρμογή και γενικότερα από την συσκευή του χρήστη. Τέλος, με βάση τα permissions, θα γίνεται μία ανάλυση με την χρήση μηχανικής μάθησης έτσι ώστε να γίνεται εύρεση τυχόν επικίνδυνων εφαρμογών με βάση τα permissions της εφαρμογής που χρησιμοποιούν. Στο σύστημα επίσης θα υπάρχει έλεγχος μιας εφαρμογής για το αν είναι κακόβουλη με την σάρωση της εφαρμογής από διάφορα γνωστά antivirus.

1.3 Δομή Διπλωματικής Εργασίας

Στο παρόν κεφάλαιο, υπάρχει μια γενική εισαγωγή στο θέμα καθώς και το κίνητρο, ο σκοπός και ο στόχος της ατομικής διπλωματικής εργασίας αυτής.

Κεφάλαιο 2:
Στο κεφάλαιο αυτό επεξηγούνται διάφορες έννοιες όπως της ιδιωτικότητας, των προσωπικών και των ευαίσθητων προσωπικών δεδομένων. Επίσης παρουσιάζονται διάφοροι κίνδυνοι

ιδιωτικότητας που υπάρχουν. Τέλος, αναφέρεται ο Κανονισμός Γενικής Προστασίας Δεδομένων, ο οποίος επεξηγείται στο τη εισάγει και τι αλλαγές θα πρέπει να γίνουν.

Κεφάλαιο 3:
Σε αυτό το κεφάλαιο περιγράφεται το λειτουργικό σύστημα Android, επεξηγώντας τα στοιχεία που το απαρτίζουν, όπως τα permissions και το τί περιέχει ένα αρχείο εφαρμογής APK. Τέλος, στο κεφάλαιο αυτό παρουσιάζονται κάποια προβλήματα και απειλές που υπάρχουν στο λειτουργικό σύστημα Android.

Κεφάλαιο 4:
Αρχικά αυτό το κεφάλαιο παρουσιάζει διάφορες προηγούμενες εργασίας που έχουν γίνει που αφορούν την ανάλυση ιδιωτικότητας σε κινητές εφαρμογές Android. Επίσης παρουσιάζονται κάποια παρέχον συστήματα ανάλυσης εφαρμογών Android. Τέλος, γίνεται μια σύγκριση της προσέγγισης των συστημάτων αυτών με το σύστημα που υλοποιήθηκε στην παρούσα ατομική διπλωματική εργασία.

Κεφάλαιο 5:
Στο πέμπτο κεφάλαιο, παρουσιάζονται οι διάφορες τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν τόσο για την ανάπτυξη του εργαλείου, όσο και για την ανάπτυξη της διαδικτυακής εφαρμογής για χρήση του εργαλείου.

Κεφάλαιο 6:
Σε αυτό το κεφάλαιο περιγράφεται η διαδικασία υλοποίησης του συστήματος, όπως επίσης και το πώς υπολογίζεται το σκορ μιας εφαρμογής. Τέλος, παρουσιάζονται κάποια screenshots από το σύστημα και περιγράφονται οι λειτουργίες του.

Κεφάλαιο 7:
Με το τέλος της σχεδίασης του συστήματος, έγινε μια ανάλυση από ένα σύνολο εφαρμογών οι οποίες συλλέχτηκαν και αναλυθήκαν για την εξαγωγή κάποιων αποτελεσμάτων. Σε αυτό το κεφάλαιο παρουσιάζονται διάφορες γραφικές παραστάσεις και συγκρίσεις των αποτελεσμάτων.

Κεφάλαιο 8:
Τέλος, δίνεται ο επίλογος με την ανάλυση των συμπερασμάτων που προκύπτουν από την μελέτη του θέματος και την υλοποίηση του συστήματος ανάλυσης εφαρμογών Android και ταυτόχρονα προτείνονται κάποιες εισηγήσεις για μελλοντική εργασία.

Κεφάλαιο 2

Ιδιωτικότητα και προστασία

2.1 Εισαγωγή

Στις μέρες μας, με την ραγδαία ανάπτυξη της τεχνολογίας σχεδόν ο μισός πλανήτης χρησιμοποιεί το διαδίκτυο και ο όγκος των δεδομένων και πληροφορίων που ανταλλάσσεται αυξάνεται με γοργούς ρυθμούς. Αυτό έχει ως αποτέλεσμα, να αυξήσει τους κινδύνους για την ιδιωτική μας ζωή. Κάθε δευτερόλεπτο, αντλούνται διάφορα στοιχεία για την προσωπική, οικονομική καθώς και την κοινωνική κατάσταση του χρήστη και με τον συνδυασμό αυτών των πληροφορίων με δεδομένα άλλων πηγών οδηγούν σε μια συνολική καταγραφή της προσωπικότητας του χρήστη. Τα δεδομένα αυτά μεταφέρονται και επεξεργάζονται από διάφορους οργανισμούς και θα πρέπει με κάποιο τρόπο να ελέγχεται η πρόσβαση τους για μη εξουσιοδοτημένα άτομα και συστήματα.

Κάτω από αυτές τις συνθήκες, εισάγονται οι όροι ιδιωτικότητα και ευαίσθητα προσωπικά δεδομένα. Τα προσωπικά δεδομένα, είναι οποιαδήποτε δεδομένα μπορούν να χρησιμοποιηθούν για την αναγνώριση ή εντοπισμό ενός ατόμου με άμεσο ή έμμεσο τρόπο και στην ουσία αναφέρονται στο πρόσωπο του ατόμου, όπως για παράδειγμα όνομα, διεύθυνση, τοποθεσία, κτλ. Αυτά που θεωρούνται ευαίσθητα προσωπικά δεδομένα, αφορούν δεδομένα τα οποία έχουν ιδιαίτερη βαρύτητα για τον σχηματισμό της εικόνας του ατόμου, όπως για παράδειγμα θρησκευτικές και πολιτικές πεποιθήσεις, σεξουαλικός προσανατολισμός, φυλετική ή εθνική προέλευση, κτλ. Το ερώτημα είναι, ποιος έχει δικαίωμα πρόσβασης σε αυτά τα δεδομένα; Αυτό θα πρέπει να το γνωρίζει ο κάθε χρήστης που του συλλέγονται πληροφορίες. Θα πρέπει να γνωρίζει ποιος είναι αυτός που χρειάζεται τα δεδομένα αυτά, για ποιο σκοπό χρειάζεται τα συλλέξει και να επεξεργαστή τα δεδομένα

αυτά και ποιος θα είναι ο τελικός αποδέκτης των πληροφοριών αυτών. Ο χρήστης θα πρέπει να έχει αυτές της πληροφορίες για να είναι σε θέσει να αποφασίσει αν είναι απαραίτητο και κατ' επέκταση ασφαλές να δώσει τα δεδομένα αυτά.

2.2 Ιδιωτικότητα και Ευαίσθητα Δεδομένα

Η Ιδιωτικότητα (Privacy) είναι ένα θεμελιώδες ανθρώπινο δικαίωμα το οποίο εμπνέει την αξιοπρέπεια και τις αξίες του ατόμου. Θεωρείται ένα από τα πιο σημαντικά ανθρώπινα δικαιώματα της σύγχρονης εποχής που ζούμε. Μάλιστα ο Edward Snowden έχει δηλώσει [1] ότι η Ιδιωτικότητα είναι σημαντικότερη από την ελευθερία του λόγου. Ο ίδιος υποστηρίζει ότι η Ιδιωτικότητα δεν αφορά κάτι το οποίο πρέπει να κρυφτεί, αλλά αφορά κάτι το οποίο θα πρέπει να προστατευτεί.

Το δικαίωμα της Ιδιωτικότητας έχει αναγνωριστεί σε όλο τον κόσμο σε διάφορες περιοχές και πολιτισμούς. Εισάχθηκε από την Οικουμενική Διακήρυξη των Δικαιωμάτων του Ανθρώπου [2] , από το Διεθνές Σύμφωνο για τα Ατομικά και Πολιτικά Δικαιώματα [3], την Ευρωπαϊκή Σύμβαση για τα Δικαιώματα του Ανθρώπου [4], καθώς και από διάφορες άλλες διεθνείς και περιφερειακές συνθήκες που αφορούν τα ανθρώπινα δικαιώματα.

Αξίζει να σημειωθεί ότι το ζήτημα του ορισμού της Ιδιωτικότητας δεν έχει καθιερωθεί σε κάτι ευρέως αποδεκτό. Ο Alan Westin είχε πει ότι «κανένας ορισμός της ιδιωτικότητας δεν είναι εφικτός, γιατί τα θέματα ιδιωτικότητας είναι εκ θεμελίων ζήτημα αξιών, συμφερόντων και εξουσίας» [5]. Με αυτό μπορούμε να πούμε ότι ο ορισμός της Ιδιωτικότητας είναι διαφορετικός ανάλογα με το κοινωνικοπολιτικό πλαίσιο και στο περιβάλλον που ορίζεται.

Με την πάροδο του χρόνου, η ανάγκη για Ιδιωτικότητα επιτυγχανόταν με διάφορα μέσα όμως ήταν πάντα επισκιασμένα από την τεχνολογική ανάπτυξη. Μετά από αρκετές έρευνες στο πεδίο της Ιδιωτικότητας, εντοπιστήκαν και αναδείχθηκαν διάφορες μορφές ιδιωτικότητας, οι οποίες αναφέρονται πιο κάτω:

- **Πληροφοριακή Ιδιωτικότητα (informational privacy)**, η οποία αναφέρεται και ως Ιδιωτικότητα δεδομένων (data privacy) και περιλαμβάνει διάφορους κανόνες που αφορούν τη συλλογή και την επεξεργασία δεδομένων οι οποίες αφορούν κάποιο άτομο. Αναφέρει ότι τα άτομα θα πρέπει να έχουν κάποιο έλεγχο στα δεδομένα που συλλέγονται. Οι πληροφορίες και τα δεδομένα αυτά συνήθως είναι προσωπικού χαρακτήρα και σχετίζονται με το τόπο διαμονής του ατόμου, τρόπους και μέσα

επικοινωνίας, χρηματοπιστωτικές πληροφορίες, ιατρικά και κυβερνητικά αρχεία, οικογενειακή κατάσταση και άλλες προσωπικές πληροφορίες. Είναι η μορφή που αναφέρουμε και ως «προστασία (προσωπικών) δεδομένων».

- **Σωματική Ιδιωτικότητα (bodily privacy)**, η οποία αφορά την προστασία της φυσικής υπόστασης του ατόμου ενάντια σε επεμβατικές διαδικασίες, οι οποίες γίνονται χωρίς την συγκατάθεση του ατόμου. Για παράδειγμα, σωματικός έλεγχος, δοκιμή φαρμάκων, μετάγγιση αίματος κτλ.
- **Τηλεπικοινωνιακή Ιδιωτικότητα (communications privacy)**, η οποία αφορά την ανάλυση ή ηχογράφηση των επικοινωνιών, την παράνομη χρήση μικροφώνων και κοριών. Σκοπός της κατηγορίας αυτής είναι να καλύπτει την ασφάλεια και την ιδιωτικότητα αλληλογραφίας, τηλεφωνικών συνδιαλέξεων, e-mail και άλλων μορφών επικοινωνίας.
- **Εδαφική Ιδιωτικότητα (territorial privacy)**, η οποία αφορά την οριοθέτηση ιδιωτικών χώρων και γενικότερα σχετίζεται με το χώρο που περιβάλει ένα άτομο. Για παράδειγμα ο χώρος εργασίας του ατόμου, ο χώρος διαμονής του, κτλ.
- **Ιδιωτικότητα προσωπικής συμπεριφοράς (privacy of personal behaviour)**, η οποία αφορά την παρατήρηση των πράξεων επιλογών και συνήθειων του ατόμου. Για παράδειγμα, σεξουαλικές επιλογές, πολιτικές επιλογές, ενδιαφέροντα, κτλ. Σε αυτή την κατηγορία επίσης, εντάσσονται και το φαινόμενα της παράνομης φωτογράφισης και καταγραφής βίντεο των ατόμων χωρίς την συγκατάθεση τους.

2.3 Κίνδυνοι Ιδιωτικότητας

Στις μέρες μας με την ανάπτυξη της τεχνολογίας, οι νέες διαδικτυακές εφαρμογές προκαλούν σε όλους μας ανησυχίες σχετικά για την προστασία της Ιδιωτικότητας και των προσωπικών δεδομένων. Έχουν γίνει αρκετές συζητήσεις περί του θέματος και για τον ρόλο της Ιδιωτικότητας σε αυτή την ανάπτυξη της τεχνολογίας. Σύμφωνα με τον Tavani, υπάρχουν 2 είδη που ανησυχούν οι οποίες είναι η εξελιγμένη ανάπτυξη του Διαδικτύου και η προ-διαδικτυακή ανάπτυξη [6]. Στην εποχή μας με την ανάπτυξη βελτιωμένων εργαλείων και τεχνικών είναι δυνατόν να υπάρχουν απειλές κατά την Ιδιωτικότητα του χρήστη σε αντίθεση με παλιότερα που ήταν πιο δύσκολο να συμβούν. Επίσης η Deborah G. Johnson και η Helen Nissenbaum στο βιβλίο «Computers, Ethics and Social Values 1995» [7], καθώς και οι Konrad Zweigert και Hein Kötz στο βιβλιο «An Introduction to Comparative Law 1998» [8] χωρίζουν τις απειλές αυτές σε απειλές κατά της Ιδιωτικότητας της επικοινωνίας όπου αναφέρονται κίνδυνοι που αφορούν την κρυπτογράφηση, την ηλεκτρονική επιτήρηση

(surveillance) και σε απειλές κατά της Ιδιωτικότητας της Βάσης Δεδομένων (ΒΔ) η οποία αφορά προσωπικά δεδομένα τα οποία αποθηκεύονται και ανταλλάσσονται μεταξύ τεράστιων ΒΔ.

Σήμερα, πολλοί οργανισμοί που συλλέγουν δεδομένα για τους χρήστες, διατηρούν αρχεία με τις προτίμησης τους, τα ενδιαφέροντά τους, τις αγορές τους, κτλ. Όλα αυτά τα χρησιμοποιούν για να δουν πώς θα πλασάρουν κάποιο προϊόν στην αγορά, να εμφανίζουν στον χρήστη διαφημίσεις που πιθανόν να τον ενδιαφέρουν για να τον δελεάσουν, ακόμα και για να χρησιμοποιηθούν και για σκοπούς προπαγάνδας, όπως για παράδειγμα το πρόσφατο σκάνδαλο μεταξύ Facebook και Cambridge Analytica [9] όπου συλλεγόντουσαν πληροφορίες για τους χρήστες οι οποίες πουλιόντουσαν για τεράστια χρηματικά ποσά με αγοραστές αρκετούς πολιτικούς. Ακόμα και οργανισμοί όπως το CIA και NSA είναι γνωστό ότι συλλέγουν πληροφορίες για τους πολίτες των ΗΠΑ και όχι μόνο. Επίσης τα δεδομένα αυτά που συλλέγονται από τους οργανισμούς και τις εταιρίες υπάρχει το ενδεχόμενο να γίνουν προσβάσημα από τρίτους, αν υπάρχει κάποιο κενό ασφαλείας και έτσι μπορεί άτομα τα οποία δεν επιθυμεί ο χρήστης να μάθουν τις προσωπικές του πληροφορίες και τα ενδιαφέροντα του. Για παράδειγμα, το 2013 η εταιρία Adobe μετά από μια σειρά επιθέσεων, αποκάλυψε πως κλάπηκαν πληροφορίες 138 εκατομμυρίων χρηστών, οι οποίες περιείχαν διάφορα ευαίσθητα προσωπικά δεδομένα, όπως διευθύνσεις, τηλέφωνα, ακόμα και πιστωτικές κάρτες. Με τα πιο πάνω παραδείγματα μπορούμε να κατανοήσουμε την σημαντικότητα της προστασίας των ευαίσθητων δεδομένων αφού μπορεί να καταπατηθεί το δικαίωμα της Ιδιωτικότητας με διάφορους τρόπος τους οποίους δεν μπορεί να ελέγξει ο χρήστης.

2.4 Κανονισμός Γενικής Προστασίας Δεδομένων (GDPR)

Ο νέος κανονισμός EU General Data Protection Regulation (GDPR), τον οποίο ενέκρινε το Ευρωπαϊκό Κοινοβούλιο στις αρχές του 2016, έχει τεθεί σε ισχύ στις 25 Μάιου του 2018. Η νέα νομοθεσία αντικαθιστά την υφιστάμενη Ευρωπαϊκή Οδηγία του 1995 για την Προστασία Δεδομένων και θέτει τις βάσεις για την προστασία των προσωπικών δεδομένων των χρηστών δίνοντας έμφαση στην προστασία τους και παρέχοντας δυνατότητες που δεν ήταν διαθέσιμες παλαιότερα.

Ο GDPR περιγράφει έναν τρόπο για την προστασία των δεδομένων που αφορούν τον προσωπικό μας εαυτό. Με τον κανονισμό αυτό, θα δοθούν βασικοί ορισμού, ώστε να μπορέσουν οι οργανισμοί να αναγνωρίσουν ποια δεδομένα είναι προσωπικού χαρακτήρα. Ο

κανονισμός καθορίζει διάφορες περιπτώσεις στις οποίες επιτρέπει την συλλογή πληροφορίων, την αποθήκευση, την χρήση, την επεξεργασία αλλά ακόμα και την μεταφορά των δεδομένων.

Τα βασικότερα στοιχεία του κανονισμού είναι:

- Η ανάγκη για συγκατάθεσή του ατόμου για την επεξεργασία των προσωπικών του δεδομένων
- Το άτομο να έχει εύκολη και διαφανείς πρόσβαση στα προσωπικά δεδομένα που έχει στην κατοχή του ο οργανισμός
- Το άτομο να έχει δικαίωμα να διαγράψει και να αλλάξει τα στοιχεία του ανα πάσα στιγμή.
- Να γνωρίζει το άτομο πως χρησιμοποιούνται τα δεδομένα αυτά
- Το άτομο να έχει δικαίωμα εναντίωσης
- Το άτομο να έχει το δικαίωμα φορητότητας των δεδομένων από πάροχο σε πάροχο

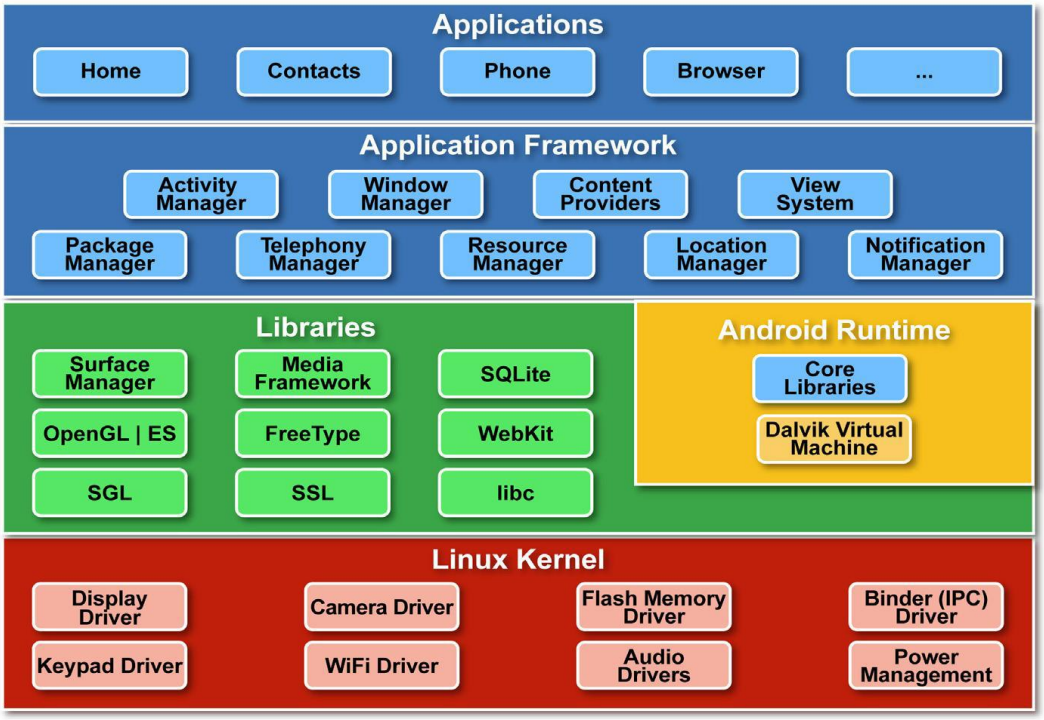
Η μη τήρηση των νόμων φέρει πολύ μεγάλα προστίματα στους υπευθύνους επεξεργασίας δεδομένων που μπορούν να ανέλθουν σε 20 εκατ. € ή στο 4% του συνολικού ετήσιου κύκλου εργασιών του οργανισμού.
Η κάθε εταιρεία θα πρέπει να έχει ορίσει έναν υπεύθυνο προστασίας δεδομένων, (DPO – Data Protection Officer) ο οποίος μπορεί να είναι και άτομο εκτός οργανισμού.
Οι υπεύθυνοι του κάθε οργανισμού σε τυχόν περιστατικά παραβίασης δεδομένων θα πρέπει να ενημερώνουν τους χρήστες εντός 72 ωρών από την ανακάλυψη του περιστατικού παραβίασης και απώλειας προσωπικών δεδομένων καθώς και στις αρμόδιες αρχές.
Όλοι οι οργανισμοί οι οποίοι έχουν ευρωπαίους χρήστες θα πρέπει να ακολουθήσουν αυτόν τον κανονισμό, ανεξάρτητα αν δεν ανήκουν στην ευρωπαϊκή ένωση. Ο κανονισμός αυτός θα ισχύει από την 25^{η} Μάϊου, αλλά ενδέχεται να δοθεί μια παράταση μέχρι το τέλος του χρόνου έτσι ώστε να δοθεί ο απαιτούμενος χρόνος για να συμμορφωθούν όλες οι εταιρίες.

Κεφάλαιο 3

Το λειτουργικό σύστημα Android

3.1 Εισαγωγή

Το Android είναι ένα λειτουργικό σύστημα ανοιχτού κώδικα, το οποίο χρησιμοποιείται σε κινητές συσκευές, τηλεοράσεις, αυτοκίνητα και σε πολλά άλλα. Εκτιμάται ότι οι ενεργοί χρήστες που χρησιμοποιούν το λειτουργικό σύστημα Android ξεπερνούν το 1 δισεκατομμύριο. Το Android είναι βασισμένο στον πυρήνα του λειτουργικού Linux και χωρίζεται σε 4 επίπεδα και 5 συστατικά. Πιο κάτω φαίνεται ένα σχήμα που δείχνει τα επίπεδα και τα συστατικά της αρχιτεκτονικής τους λειτουργικού συστήματος Android και στην συνέχεια δίνεται μια σύντομη εξήγηση του κάθε συστατικού ξεκινώντας από το χαμηλότερο επίπεδο:



Σχήμα 3.1 Αρχιτεκτονική Android (https://letsknowaboutandroid.wordpress.com/about/)

Linux Kernel

Όπως αναφέρθηκε και πιο πάνω, το Android βασίζεται στον πυρήνα του Linux όπου με την χρήση του πραγματοποιούνται κάποιες βασικές λειτουργίες όπως την διαχείριση των διαφόρων διεργασιών, την διαχείριση της μνήμης και των drivers της συσκευής, καθώς επίσης και την διαχείριση των δικτυακών διεπαφών και της ενέργειας της συσκευής.

Native Libraries

Σε αυτό το επίπεδο, υπάρχουν κάποιες βιβλιοθήκες που χρησιμοποιεί το σύστημα Android για την υποστήριξη διαφόρων διαδικασιών. Οι βιβλιοθήκες αυτές είναι γραμμένες σε C/C++ και χρησιμοποιούνται μέσω interfaces από την Java. Αναφέροντας κάποιες, η βιβλιοθήκη WebKit βοηθά στην υποστήριξη των φυλλομετρητών (browsers) και η βιβλιοθήκη SQLite χρησιμοποιείται για την διαχείριση σχεσιακών βάσεων δεδομένων. Η βιβλιοθήκη Surface Manager χρησιμοποιείται για την δημιουργία των παραθύρων τις εφαρμογής και η OpenGL βοηθά στην υποστήριξη 2D και 3D γραφικών. Επίσης η βιβλιοθήκη Media Framework βοηθά στην αποκωδικοποίηση για την αναπαραγωγή αρχείων πολυμέσων όπως ήχοι, εικόνες, βίντεο, κτλ.

Android Runtime

Όπως φαίνεται και στο πιο πάνω σχήμα, το Android Runtime αποτελείτε από τις Core Libraries και το Dalvik Virtual Machine (DVM). Οι Core Libraries είναι βιβλιοθήκες που βοηθούν την διεπαφή των εφαρμογών Java με το περιβάλλον της συσκευής όπου εκτελείτε η εφαρμογή. Το DVM είναι υπεύθυνο στην εκτέλεση των Android εφαρμογών. Είναι παρόμοιο με το Java Virtual Machine (JVM) αλλά το DVM είναι βελτιστοποιημένο για κινητές συσκευές και για χαμηλή κατανάλωση ενέργειας.

Android Framework

Στο πιο πάνω επίπεδο βρίσκεται το Android Framework. Το Android παρέχει το Android API μέσω του Android Framework υπάρχουν πολλές και διάφορες διεπαφές και κλάσεις που βοηθούν στην ανάπτυξη των εφαρμογών. Κάποιες από τις λειτουργίες που παρέχονται από το Android Framework είναι η δυνατότητα του να χειρίζεται κάποιος την γραφική

διπροσωπία (UI), δυνατότητα αποστολής και χρήσης δεδομένων από άλλες εφαρμογές και γενικότερα από το σύστημα.

Applications

Στο ανώτερο επίπεδο, βρίσκονται οι εφαρμογές που με τις οποίες μπορεί να χρησιμοποιήσει και να αλληλεπιδράσει ο χρήστης όπως φυλλομέτρησες, διαχείριση επαφών, χάρτες, παιχνίδια, κτλ. Οι εφαρμογές αυτές χρησιμοποιούν το Android Framework, το οποίο χρησιμοποιά το Android Runtime και τις Native Libraries, οι οποίες με την σειρά τους χρησιμοποιούν τον Linux Kernel.

3.2 Android Permissions

Για την ασφάλεια των ευαίσθητων δεδομένων και της συσκευής, το Android έχει αρκετούς μηχανισμούς ασφαλείας όπως το Android Application Sandbox το οποίο απομονώνει τα δεδομένα τις εφαρμογής και την εκτέλεση του κώδικα από άλλες εφαρμογές. Επίσης, το Android παρέχει ένα API με διάφορα permissions τα οποία ελέγχουν την πρόσβαση σε λειτουργίες του συστήματος (όπως κάμερα, πρόσβαση στο διαδίκτυο, κτλ.) και σε δεδομένα του χρήστη (όπως επαφές, μηνύματα, κτλ.).

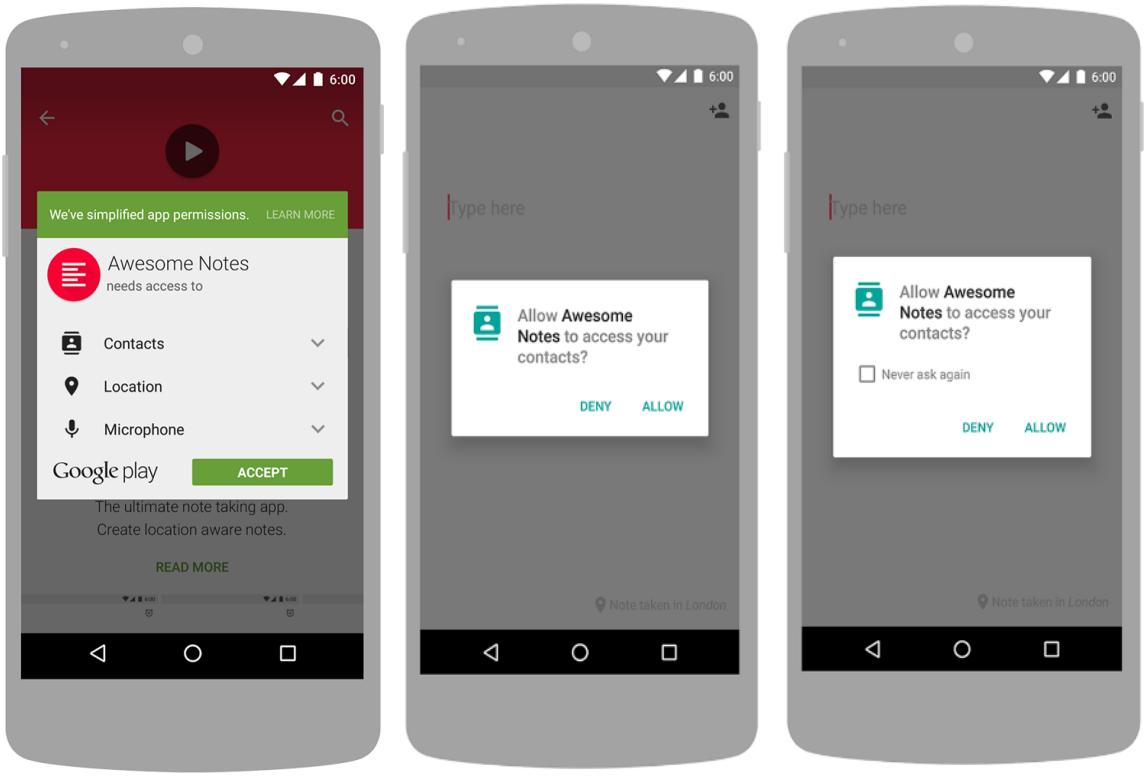
Κάθε εφαρμογή θα πρέπει να δηλώσει τα permissions που χρησιμοποιεί έτσι ώστε να γνωρίζουν οι χρήστες που θα το εγκαταστήσουν που θα έχει πρόσβαση η εφαρμογή. Εάν ο χρήστης δεν επιθυμεί να δώσει πρόσβαση στην εφαρμογή τότε μπορεί κατά την εγκατάσταση να σταματήσει την εγκατάσταση. Με τις τελευταίες εκδώσεις του Android όταν θα χρησιμοποιηθεί κάποιο επικίνδυνο (dangerous) permission από μια εφαρμογή κατά την εκτέλεση της, ζητείται από τον χρήστη να δώσει πρόσβαση στην εφαρμογή για το συγκεκριμένο permission, έτσι ώστε να μπορεί να πραγματοποιήσει την απαιτούμενη ενέργεια. Ο χρήστης μπορεί να επιλέξει να μην δώσει πρόσβαση για το συγκεκριμένο permission στην εφαρμογή με αποτέλεσμα να μην ολοκληρωθεί η ενέργεια που απαιτούσε το ανάλογο permission. Με αυτό τον τρόπο ο χρήστης έχει περισσότερο έλεγχο προς την εφαρμογή, αφού θα ερωτάται την στιγμή που θα χρησιμοποιηθεί κάποιο permission . Στις νεότερες έκδοσης του Android, μπορούν οι χρήστες να επιλέξουν σε τι έχει πρόσβαση μια εφαρμογή, ανά πάσα στιγμή, το οποίο είναι ένα μεγάλο πλεονέκτημα προς τους χρήστες για τον περιορισμό ανεπιθύμητων προσβάσεων στα δεδομένα τους, ανεξάρτητα αν έχουν δώσει πρόσβαση στην εφαρμογή προηγουμένως.

Πιο κάτω φαίνεται το πώς λειτουργεί αυτή η διαδικασία και με τους 2 τρόπους που αναφέρθηκαν πιο πάνω:

Στις συσκευές με έκδοση Android 5.1.1 (API Version 22) και κάτω, κατά την διάρκεια εγκατάστασης μιας εφαρμογής, ζητείται από τον χρήστη να δώσει πρόσβαση στην εφαρμογή σε όλα τα permission που χρειάζεται για να λειτουργήσει. Όπως αναφέρθηκε και πιο πάνω εάν ο χρήστης επιθυμεί να δώσει την πρόσβαση στην εφαρμογή τότε θα πρέπει να πατήσει αποδοχή (accept) όταν ερωτηθεί για τα permissions. Στην πιο κάτω εικόνα (Σχήμα 3.2 – αριστερή οθόνη) παρουσιάζεται το πώς είναι μια τέτοια οθόνη που ζητά πρόσβαση στα permissions κατά την διάρκεια της εγκατάστασης μιας εφαρμογής. Στην περίπτωση που ο χρήστης δεν επιθυμεί να παραχωρήσει στην εφαρμογή τα ζητούμενα permissions, τότε το μόνο που χρειάζεται να κάνει είναι να πατήσει το κουμπί back το οποίο θα ακυρώσει την εγκατάσταση της εφαρμογής. Σε περίπτωση που γίνεται αναβάθμιση μια εγκατεστημένη εφαρμογή και στην καινούργια της έκδοση γίνεται χρήση permissions τα οποία δεν υπήρχαν πριν, τότε θα εμφανιστεί μήνυμα όπως και στο Σχήμα 3.2 (αριστερή οθόνη), έτσι ώστε ο χρήστης να δώσει πρόσβαση στην εφαρμογή στα permissions αυτά. Εάν ο χρήστης δεν θέλει να δώσει πρόσβαση σε αυτά τα permissions, τότε δεν θα συνεχιστεί η αναβάθμιση της εφαρμογής.

Στις συσκευές με έκδοση Android 6.0 (API level 23) και πάνω, κατά την διάρκεια εγκατάστασης μιας εφαρμογής, δε ζητείται από το χρήστη να δώσει πρόσβαση εκείνη την στιγμή στα δεδομένα που χρειάζεται στην εφαρμογή. Στην συνέχεια κατά την εκτέλεση της εφαρμογής, εάν κάποια λειτουργία χρειάζεται πρόσβαση σε κάποια δεδομένα τότε θα εμφανιστεί ένα κατάλληλο μήνυμα στην οθόνη (Σχήμα 3.2 – μεσαία οθόνη) ζητώντας από τον χρήστη να δώσει πρόσβαση στην εφαρμογή στο ανάλογο permission. Ο χρήστης όταν επιτρέψει στην εφαρμογή να πάρει πρόσβαση στα δεδομένα που χρειάζεται τότε δεν θα χρειαστεί να το κάνει ξανά. Αντίθετα, εάν ο χρήστης δεν δώσει πρόσβαση στην εφαρμογή, την επόμενη φορά που ο χρήστης θα προσπαθήσει να χρησιμοποιήσει την ίδια λειτουργία της εφαρμογής που χρειάζεται πρόσβαση σε αυτά τα δεδομένα, θα του εμφανιστεί ξανά το μήνυμα για να του ζητηθεί να δώσει πρόσβαση στην εφαρμογή. Σε αυτό το μήνυμα όμως (Σχήμα 3.2 – δεξιά οθόνη), υπάρχει και η επιλογή ο χρήστης να επιλέξει να μην του ξαναεμφανιστεί το ίδιο μήνυμα ξανά. Εάν ο χρήστης κάνει αυτή την επιλογή και στην συνέχεια επιλέξει να μην δώσει πρόσβαση στα συγκεκριμένα δεδομένα στην εφαρμογή, τότε δεν θα του εμφανιστή ποτέ ξανά παρόμοιο μήνυμα για το συγκεκριμένο permission κατά την διάρκεια χρήσης της εφαρμογής. Αυτό τις περισσότερες φορές, έχει ως αποτέλεσμα ο χρήστης να μην μπορεί να χρησιμοποιήσει κάποιες λειτουργίες τις εφαρμογής αυτής που χρειάζονται τα δεδομένα αυτά.

Με τους πιο πάνω τρόπους λοιπόν, μπορεί ο χρήστης να ελέγξει ποιες εφαρμογές έχουν πρόσβαση στα ευαίσθητα δεδομένα του και στις λειτουργίες του συστήματος της συσκευής.



Σχήμα 3.2 Παραδείγματα από τις αιτήσεις για Permissions
(https://developer.android.com/guide/topics/permissions/overview)

Επίπεδα προστασίας permissions (Protection levels)

Τα permissions χωρίζονται σε 4 επίπεδα προστασίας, ανάλογα με το πόσο επικίνδυνα είναι και τι πληροφορίες ή λειτουργίες θα έχουν πρόσβαση. Επίσης το επίπεδο προστασίας καθορίζει εάν το permission είναι απαραίτητο να ζητείται από τον χρήστη κατά την εκτέλεση της εφαρμογής όπως αναφέρθηκε πιο πάνω. Τα 4 επίπεδα θα επεξηγηθούν πιο κάτω.

Normal permissions

Αυτό το επίπεδο προστασίας αφορά permissions τα οποία χρειάζονται πρόσβαση σε δεδομένα και πόρους του συστήματος, τα οποία όμως δεν είναι σε θέση να επηρεάσουν την λειτουργία άλλων εφαρμογών ή να επηρεάσουν την ιδιωτικότητα του χρήστη με οποιοδήποτε τρόπο. Στα permissions αυτά στις συσκευές με έκδοση Android 6.0 και πάνω, δίνεται κατά την εγκατάσταση μιας εφαρμογής, απευθείας πρόσβαση από το ίδιο το λειτουργικό σύστημα. Αυτό έχει ως αποτέλεσμα ο χρήστης να μην μπορεί να αναιρέσει την πρόσβαση της εφαρμογής στα αντίστοιχα δεδομένα ή πόρους των permissions αυτών. Ο χρήστης μπορεί

μόνο να δει τα permissions αυτά πριν την εγκατάσταση της εφαρμογής. Ένα παράδειγμα που ανήκει σε αυτό το επίπεδο είναι η πρόσβαση στις πληροφορίες του δικτύου της συσκευής, για να ελέγχει η εφαρμογή εάν η συσκευή είναι συνδεδεμένη στο διαδίκτυο.

Signature permissions

Σε αυτό το επίπεδο προστασίας βρίσκουμε τα permissions τα οποία παραχωρούνται στην εφαρμογή από το σύστημα κατά την διάρκεια της εγκατάστασης, μόνο εάν το certificate της εφαρμογής που ζητά το permissions είναι υπογεγραμμένο από το ίδιο certificate από την εφαρμογή που ορίζει το permission. Αρκετά permissions από αυτό το επίπεδο, δεν θα πρέπει να χρησιμοποιούνται από εφαρμογές που δεν ανήκουν στο σύστημα. Ένα παράδειγμα που ανήκει σε αυτό το επίπεδο είναι η δυνατότητα μιας εφαρμογής να πάρει πληροφορίες ή να αλλάξει τις ρυθμίσεις της συσκευής.

SignatureOrSystem permissions

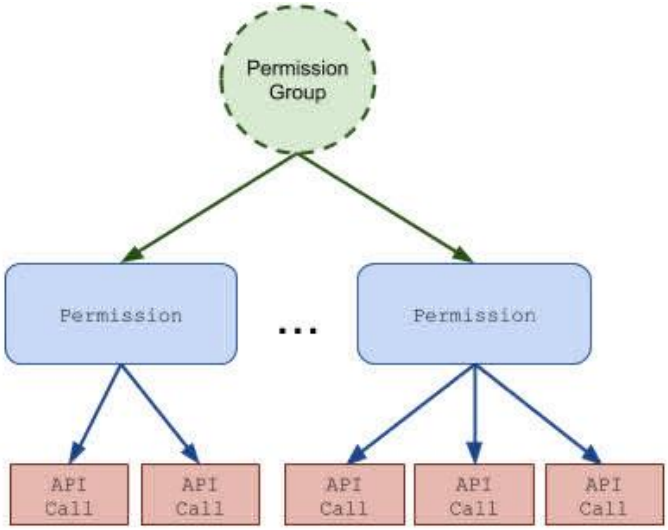
Επίσης σε αυτό το επίπεδο υπάρχουν και τα permissions τα οποία κατηγοριοποιούνται σαν signatureOrSystem. Και εδώ θα πρέπει το certificate της εφαρμογής που ζητά το permissions είναι υπογεγραμμένο από το ίδιο certificate από την εφαρμογή που ορίζει το permission. Ακόμα εάν η εφαρμογή είναι μέρος του συστήματος του Android, τότε της δίνεται το ανάλογο permission από αυτή την κατηγορία. Γενικότερα, τα signatureOrSystem permissions, χρησιμοποιούνται για ειδικές καταστάσεις όπου οι προμηθευτές έχουν ενσωματωμένες εφαρμογές στο λειτουργικό σύστημα και πρέπει να μοιράζονται ρητά συγκεκριμένα χαρακτηριστικά επειδή εγκαθίστανται μαζί στην συσκευή.

Dangerous permissions

Τα permissions που ανήκουν σε αυτό το επίπεδο, αφορούν πρόσβαση σε δεδομένα τα οποία μπορεί να περιέχουν προσωπικές πληροφορίες του χρήστη και μπορούν να παρέμβουν στην ιδιωτικότητα του ή να επηρεάσουν τα δεδομένα της συσκευής του. Επίσης τα permissions αυτά μπορεί να αφορούν και την πρόσβαση σε πόρους συστήματος τα οποία μπορεί να επηρεάσουν την λειτουργία άλλων εφαρμογών. Ένα παράδειγμα που ανήκει σε αυτό το επίπεδο είναι η δυνατότητα μια εφαρμογή να έχει πρόσβαση στην τοποθεσία της συσκευής μέσο GPS. Ένα άλλο παράδειγμα είναι η πρόσβαση στην κάμερα της συσκευής. Όπως αναφέρθηκε και προηγουμένως τα permissions αυτά θα πρέπει να ζητηθούν από τον χρήστη και αυτός θα αποφασίσει εάν θα παραχωρήσει στην εφαρμογή αυτά τα permissions.

Ομάδες Permission

Τα permissions χωρίζονται σε ομάδες με βάση την λειτουργία και τις δυνατότητες της συσκευής. Πιο κάτω φαίνεται ένα διάγραμμα που εξηγεί την ομαδοποιήσει αυτή.



Σχήμα 3.3 Ομάδες των Permissions
(https://developer.android.com/guide/topics/permissions/overview)

Κάθε dangerous permissions ανήκει σε μια ομάδα (Permission Group). Τα permissions των υπόλοιπων επιπέδων μπορούν και αυτά να ανήκουν σε κάποια ομάδα.

Με αυτό τον τρόπο, τα αιτήματα για τα dangerous permissions γίνονται με βάση την ομάδα που ανήκουν και όχι ανά permission. Τα permissions που δεν είναι dangerous δεν ακολουθούν αυτή την διαδικασία. Για παράδειγμα, μια εφαρμογή έχει την δυνατότητα να επεξεργαστεί το ημερολόγιο μιας συσκευής και ζητά τα permissions *'READ_CALENDAR'* και *'WRITE_CALENDAR'*. Για τον χρήστη αυτό δεν έχει σημασία. Σημασία έχει να δώσει πρόσβαση στην εφαρμογή για την ομάδα permissions *'CALENDAR'* και αυτό θα του ζητηθεί. Με αυτή την ομαδοποιήσει των permissions, ο χρήστης δεν θα συγκλονίζεται από περίπλοκες και δυσνόητες πληροφορίες, αλλά θα μπορεί να κάνει πιο ουσιαστικές και κατανοητές επιλογές.

Όπως εξηγήθηκε και πιο πριν, τα permissions είτε θα ζητηθούν από τον χρήστη κατά την εγκατάσταση της εφαρμογής σε συσκευές με έκδοση Android 5.1.1 και κάτω ή θα ζητηθούν την ώρα της εκτέλεσης σε συσκευές με έκδοση Android 6.0 και πάνω. Αυτό που θα ζητηθεί όμως και στις 2 αυτές περιπτώσεις, είναι η πρόσβαση στην ομάδα των dangerous permissions που χρησιμοποιεί η εφαρμογή, όπως το παράδειγμα που αναφέρθηκε πιο πάνω.

Στην περίπτωση που η συσκευή χρησιμοποιεί έκδοση Android 6.0 και πάνω, και η εφαρμογή χρειαζόταν το permission *'READ_CALENDAR'* για μια λειτουργία και ο χρήστης το έχει δώσει και στην συνέχεια η εφαρμογή για μια άλλη λειτουργία ζητήσει το permission

'WRITE_CALENDAR', το σύστημα αυτόματα θα δώσει το δικαίωμα στην εφαρμογή να γράψει στο ημερολόγιο, διότι τα 2 αυτά permissions ανήκουν στην ίδια ομάδα. Όμως τα 2 αυτά permissions θα πρέπει πάντα να ζητούνται από την εφαρμογή, ακόμα και αν ο χρήστης έχει είδη δώσει πρόσβαση σε κάποιο permissions ίδιας ομάδας. Επίσης η διαδικασία αυτή μπορεί στο μέλλον μετά από κάποια αναβάθμιση να αλλάξει γι' αυτό θα πρέπει να δηλώνονται και να ζητώνται όλα τα permissions ξεχωριστά από την εφαρμογή.

Custom permissions

Το Android ακόμα, επιτρέπει στους προγραμματιστές να δηλώσουν τα δικά τους permissions τα οποία ονομάζονται custom permissions. Τα permissions αυτά χρησιμοποιούνται για προστασία διαφόρων πληροφορίων των εφαρμογών από τις υπόλοιπες εφαρμογές που βρίσκονται εγκατεστημένες στην συσκευή. Αυτή είναι η διαφορά τους, δηλαδή δεν αφορούν δεδομένα του συστήματος, αλλά δεδομένα μιας συγκεκριμένης εφαρμογής. Για παράδειγμα, ένα οργανισμός ο οποίος κατασκευάζει εφαρμογές για Android, θέλει οι εφαρμογές του να μπορούν να μοιράζονται πληροφορίες μεταξύ τους. Αυτό μπορεί να γίνει με τον καθορισμό custom permissions, έτσι ώστε οι εφαρμογές που τους παραχωρήθηκε το permission αυτό (συνήθως από το σύστημα λόγο signature επιπέδου προστασίας) να μπορούν να επικοινωνήσουν. Καμία άλλη εφαρμογή δεν μπορεί να έχει πρόσβαση στα δεδομένα αυτά παρά μόνο όταν ζητήσει το permissions και της παραχωρηθεί.

3.3 Δομή APK αρχείων

Τα αρχεία Application Package Kit (APK) είναι αρχεία πακέτου που χρησιμοποιείται για την διανομή και εγκατάσταση Android εφαρμογών. Για να δημιουργηθεί ένα APK, θα πρέπει πρώτα να μεταγλωττιστεί ένα πρόγραμμα Android έτσι ώστε όλα τα αρχεία που το απαρτίζουν να ομαδοποιηθούν σε ένα πακέτο. Στην ουσία ένα APK είναι ένα συμπιεσμένο αρχείο σε zip πακέτο βασισμένο στην μορφή JAR με επέκταση '.apk'. Ο τύπος MIME των APK αρχείων είναι 'application/vnd.android.package-archive'.

Πιο κάτω παρουσιάζεται αναλυτικά τι περιέχει ένα αρχείο APK:

- Φάκελος META-INF
    - MANIFEST.MF: Το Manifest αρχείο το οποίο δημιουργείται αυτόματα κατά την δημιουργία του APK αρχείου, περιέχει κάποιες πληροφορίες σχετικά με το πακέτο που βρίσκονται στο αρχείο APK.
    - CERT.RSA: Περιέχει το certificate της εφαρμογής.

- CERT.SF: Περιέχει πληροφορίες οι οποίες βοηθούν στην επαλήθευση της υπογραφής (signature) του APK, στην περίπτωση που οι άλλοι μηχανισμοί επαλήθευσης αποτύχουν.

- Φάκελος lib: Περιέχει μεταγλωττισμένο κώδικα ο οποίος για τα συγκεκριμένα επίπεδα λογισμικού ανάλογα με τον επεξεργαστή που χρησιμοποιεί το σύστημα.
  - armeabi: Μεταγλωττισμένος κώδικας για τους επεξεργαστές ARM.
  - armeabi-v7a: Μεταγλωττισμένος κώδικας για τους επεξεργαστές έκδοσης ARMv7 και πάνω.
  - arm64-v8a: Μεταγλωττισμένος κώδικας για τους επεξεργαστές ARMv8 arm64 και πάνω.
  - x86: Μεταγλωττισμένος κώδικας για τους επεξεργαστές x86 (32-bit).
  - x86_64: Μεταγλωττισμένος κώδικας για τους επεξεργαστές x86 64 (64-bit).
  - mips: Μεταγλωττισμένος κώδικας για τους επεξεργαστές MIPS.

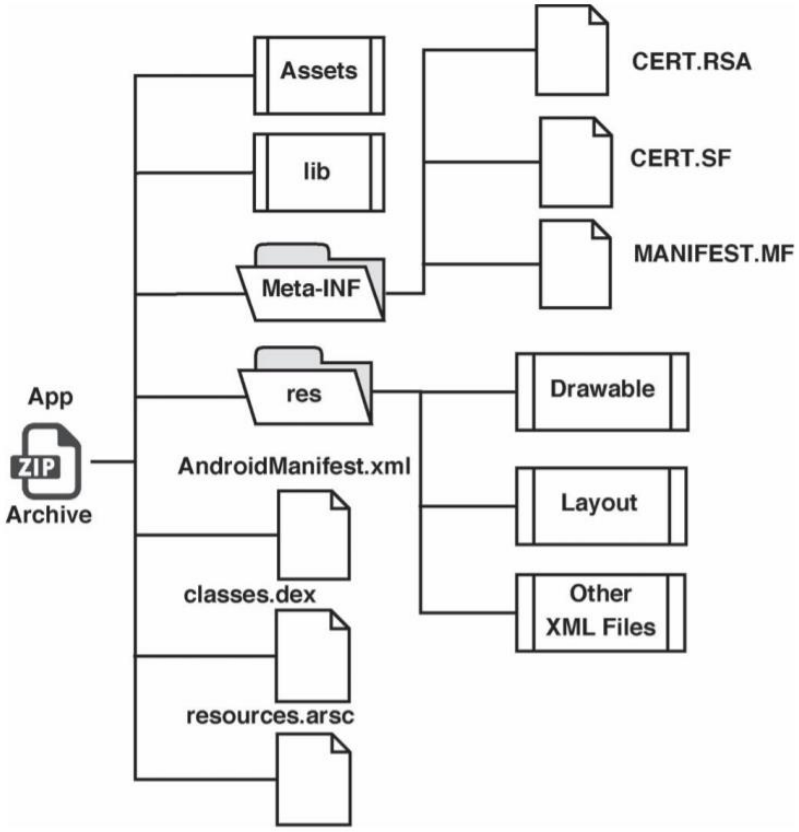
- Αρχείο resources.arsc: Περιέχει μεταγλωττισμένα στοιχεία, όπως binary XML αρχεία.

- Φάκελος res: Περιέχει τα στοιχεία τα οποία δεν μεταγλωττίστηκαν στο αρχειο resources.arsc.

- Φάκελος assets: Περιέχει αρχεία όπως γραμματοσειρές, αρχεία ΒΔ (SQLite), κτλ. Τα αρχεία αυτά χρησιμοποιούνται μέσω του AssetManager.

- Αρχείο AndroidManifest.xml: Είναι ένα επιπρόσθετο αρχείο manifest, το οποίο περιέχει πάρα πολλές πληροφορίες για την εφαρμογή όπως το όνομα της εφαρμογής, το version του Android που χρειάζεται για να τρέξει (minimum API Level), τα permissions που χρησιμοποιεί ή καθορίζει η εφαρμογή, ποιες λειτουργίες θα χρειαστεί από το σύστημα η εφαρμογή (όπως camera, luetooth, κτλ.) και ποιες βιβλιοθήκες χρειάζονται να γίνουν link με την εφαρμογή (όπως Google Places API library). Το αρχείο αυτό είναι πολύ σημαντικό για την ανάλυση των εφαρμογών αφού κάποιος μπορεί να εξάγει αρκετές πληροφορίες για την εφαρμογή. Όμως το αρχείο αυτό είναι σε binary XML μορφή, γι' αυτό για να δει κάποιος τις πληροφορίες αυτές θα πρέπει να μετατραπεί σε μορφή κειμένου με την χρήση εξειδικευμένων εργαλείων.

- Αρχείο classes.dex: Το αρχείο αυτό περιέχει τις μεταγλωττισμένες κλάσεις java της εφαρμογής σε μορφή dex την οποία αναγνωρίζει το Dalvik Virtual Machine (DVM) και το Android Runtime. Υπάρχουν εξειδικευμένα εργαλεία που μετατρέπουν το αρχείο αυτό σε μορφή η οποία είναι κατανοητή προς τον άνθρωπο αλλά δεν μπορεί να σου παράγει τον ακριβή κώδικα της εφαρμογής.



Σχήμα 3.4 Δομή APK αρχείου

(https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6999911)

3.4 Προβλήματα και απειλές του Android

Το λειτουργικό σύστημα Android όπως αναφέρθηκε και πιο πριν έχει διάφορους μηχανισμούς ασφαλείας (όπως η χρήσει των permissions), όμως υπάρχουν διάφορα προβλήματα τα οποία είναι δύσκολο να αντιμετωπιστούν με τους μηχανισμούς αυτούς. Σε αυτή την ενότητα θα παρουσιαστούν κάποιες απειλές και προβλήματα που έχει το Android, τα οποία μπορεί να οδηγήσουν σε διάφορα προβλήματα σχετικά με την ιδιωτικότητα του χρήστη.

Διαρροή πληροφοριών (Information leakage)

Όπως αναφέρθηκε και προηγουμένως με την χρήσει των permissions ο χρήστης μπορεί να καθορίσει αν μια εφαρμογή θα χρησιμοποιήσει ή όχι πόρους και δεδομένα της συσκευής. Η διαρροή πληροφοριών είναι πιθανών να συμβεί όταν οι χρήστες δίνουν πρόσβαση σε permissions τα οποία δεν είναι απαραίτητα για την εφαρμογή. Για αυτό τον λόγο τα permissions δεν μπορούν από μόνα τους να προστατέψουν την ιδιωτικότητα του χρήστη και τους πόρους του συστήματος από τις κακόβουλες εφαρμογές.

Έρευνες έχουν δείξει ότι περισσότερες από το 70% των εφαρμογών ζητούν πρόσβαση και συλλέγουν πληροφορίες οι οποίες δεν έχουν να κάνουν με την λειτουργικότητα τους [24][25]. Αυτές οι εφαρμογές καταλήγουν να είναι εγκατεστημένες σε πολλές συσκευές, αφού οι χρήστες καθώς κάνουν εγκατάσταση ή χρήση των εφαρμογών αυτών δεν δίνουν αρκετή σημασία στο που ζητά πρόσβαση η εφαρμογή. Μόνο το 3% των χρηστών, διαχειρίζονται σωστά τα permissions και κάνουν προσεκτικές κινήσεις. Ένας από τους λόγους που συμβαίνουν αυτά [26][27], είναι ότι οι άπειροι χρήστες δεν γνωρίζουν και δεν αντιλαμβάνονται ότι κάποιες αιτήσεις που αποδέχονται για συγκεκριμένα permissions μπορεί να εκθέσει προσωπικές και αναίσθητες τους πληροφορίες. Επίσης ένας άλλος λόγος είναι ότι οι χρήστες που επιθυμούν να χρησιμοποιούν εφαρμογές που χρησιμοποιούν αρκετά permissions, αναγκάζονται να ανταλλάξουν την ιδιωτικότητα τους για να μπορούν να τις χρησιμοποιούν.

Προσαρμογή προνομίων (Privilege escalation)

Αναφέρονται και ως permission escalation attacks, οι οποίες εμφανίστηκαν από την εκμετάλλευση των διαθέσιμων ευπαθειών του πυρήνα του Android και έχουν σκοπό την απόκτηση πρόσβασης σε πόρους του συστήματος οι οποίοι προστατεύονται από μια εφαρμογή ή τον χρήστη. Αυτός ο τύπος επίθεσης μπορεί να έχει ως αποτέλεσμα μη εξουσιοδοτημένες ενέργειες από εφαρμογές με περισσότερα προνόμια (privileges) από ’σα προορίζονται, κάτι το οποίο μπορεί να προκαλέσει διαρροές πληροφοριών. Τα διάφορα κομμάτια που αποτελούν το Android μπορεί να εκμεταλλευτούν για πρόσβαση σε επικίνδυνα permissions [28].

Επαναπακετάρισμα εφαρμογής (Repackaging app)

Το επαναπακετάρισμα (repackaging) μιας εφαρμογές είναι πολύ σημαντικό και συνηθισμένο πρόβλημα στο Android. Είναι η διαδικασία όπου το αρχείο APK της εφαρμογής γίνεται disassemble και decompile με την χρήση reverse-engineering εργαλείων, βάζοντας κακόβουλο κώδικα στον αρχικό κώδικα της εφαρμογής. Είναι δύσκολο να ξεχωρίσουμε μια εφαρμογή αν έχει υποστεί επαναπακετάρισμα, αφού οι λειτουργίες της θα είναι αντίστοιχες της αρχικής.

Επιθέσεις συνωμοσίας (Colluding attacks)

Οι επιθέσεις αυτές γίνονται με την χρήση διάφορων εφαρμογών. Οι χρήστες εγκαθιστούν τις εφαρμογές αυτές, οι οποίες προέρχονται από τον ίδιο προγραμματιστή και χρησιμοποιούν διάφορα permissions τα οποία μπορεί να είναι επικίνδυνα ή και όχι. Αυτές οι εφαρμογές, ανταλλάζουν δεδομένα με σκοπό την ανυποψίαστη συλλογή πληροφοριών από τους χρήστες και την συσκευή γενικότερα.

Κεφάλαιο 4

Προηγούμενες Έρευνες και Υπάρχοντα Συστήματα Ανάλυσης Android Εφαρμογών

4.1 Εισαγωγή

Στις μέρες μας, υπάρχουν αρκετές κακόβουλές εφαρμογές που έχουν σκοπό την συλλογή πληροφοριών από τον χρήστη όπως κωδικούς πρόσβασης ή κωδικού πιστωτικών καρτών, να προκαλέσουν προβλήματα στην συσκευή του χρήστη διαγράφοντας αρχεία, μηνύματα ή ακόμα και να χρεώσουν οικονομικά των χρήστη. Οι εφαρμογές που χρησιμοποιεί κάποιος συνήθως παρέχονται από το Google Play, αν και υπάρχουν και άλλες πηγές όπου κάποιος μπορεί να κατεβάσει και να εγκαταστήσει εφαρμογές. Η χρήση του Google Play για την εύρεση και εγκατάσταση εφαρμογών είναι η προτεινόμενη, αφού υπάρχει κάποιος έλεγχος από την Google. Όμως αυτός ο έλεγχος δεν είναι αρκετός και αυτό φαίνεται καθημερινά με αρκετές κακόβουλες εφαρμογές να καταφέρνουν να μπουν στο Google Play και χρήστες να τις κατεβάζουν, με αποτέλεσμα να προκαλούνται τα προβλήματα που αναφέρθηκαν πιο πάνω. Οι δημιουργοί και εκδότες των εφαρμογών υποχρεώνονται να παρέχουν privacy policies για τις εφαρμογές τους και να ενημερώνουν τους χρήστες για τις πρακτικές προστασίας προσωπικών δεδομένων. Αλλά δεν μπορούμε να είμαστε βέβαιοι ότι μια εφαρμογή συμπεριφέρεται με τον τρόπο που λέει στην πολιτική της.

Γι’ αυτό τον λόγο, αρκετές ομάδες προγραμματιστών, ερευνητών καθώς και διάφοροι οργανισμοί μελετούν την ανάλυση Android εφαρμογών. Υπάρχουν αρκετές προτεινόμενες λύσεις και τεχνικές ανάλυσης χωρισμένες σε στατικές, οι οποίες αναλύουν τον κώδικα και τα

αρχεία ψάχνοντας διάφορες λέξεις και πρότυπα που θα βοηθήσουν στην εξαγωγή των αποτελεσμάτων. Επίσης, υπάρχει η δυναμική ανάλυση των εφαρμογών, όπου η ανάλυση γίνεται ενώ εκτελείται η εφαρμογή και συνήθως με το να ελέγχεται τι στοιχεία ζήτα και παίρνει η εφαρμογή και το πως συμπεριφέρεται. Οι περισσότερες προτεινόμενες λύσεις, συνήθως χρησιμοποιούν συνδυασμό διαφόρων εργαλείων και τεχνικών ανάλυσης. Κάποιοι υποσχόμενοι τρόποι ανάλυσης οι οποίοι έχουν αναπτυχθεί αρκετά τα τελευταία χρόνια είναι η χρήση της τεχνικής νοημοσύνης και της μηχανής μάθησης, όπου μέσω αυτών μπορεί να γίνει αναγνώριση κακόβουλων εφαρμογών.

Κάποιοι οργανισμοί αποφάσισαν να δημιουργήσουν τα δικά τους συστήματα, όπου ο χρήστης θα μπορεί να ελέγξει αν κάποια εφαρμογή είναι κακόβουλή ή και γενικότερα να δει περισσότερες πληροφορίες για την δομή και την συμπεριφορά της εφαρμογής αυτής. Υπάρχουν συστήματα τα οποία είναι διαδικτυακά, όπου μπορεί ο χρήστης να ανεβάσει την εφαρμογή και να πάρει τα αποτελέσματα αυτά, αλλά υπάρχουν και συστήματα τα οποία μπορεί να εγκαταστήσει κάποιος στο σύστημα του και να τα χρησιμοποιήσει ο ίδιος.

4.2 Προηγούμενες έρευνες

Όπως αναφέρεται και πιο πάνω, υπάρχουν αρκετές μελέτες σε θέματα προστασίας ιδιωτικότητας και ανάλυσης εφαρμογών σε Android. Σε αυτή την ενότητα παρουσιάζονται κάποιες μελέτες που έχουν για τα θέματα αυτά.

Το paper, ‘Privacy by Design Permission System for Mobile Applications’ [30], αναφέρει τις διαφορές μεταξύ του Android και του iOS με βάση το Privacy by Design (PbD) και εξηγεί ότι πρέπει να γίνουν κάποιες αλλαγές με αφορμή και την νομοθεσία της ΕΕ (GDPR). Συγκρίνει τα Permissions στα 2 αυτά συστήματα και κατά πόσων μια εφαρμογή μπορεί να κάνει κάποιες λειτουργίες. Στην συνέχεια παρουσιάζεται ένα προτεινόμενο σύστημα το οποίο θα εξηγεί πλήρως και με ξεκάθαρο τρόπο το πώς χρησιμοποιούνται τα δεδομένα του χρήστη σε κάποια εφαρμογή, έτσι ώστε να έχει τον έλεγχο και να καθορίζει τα permissions, όπως επίσης και να κάνει τους προγραμματιστές να συμμορφώνονται με τον νόμο.

Επίσης το paper, ‘Automated Analysis of Privacy Requirements for Mobile Apps’ [31] ασχολείται με τα privacy policies των εφαρμογών σε Android και με βάση κάποιων μετρήσεων σε δείγματα εφαρμογών, παίρνουμε κάποια αποτελέσματα κατά πόσων το privacy policy της εφαρμογής αντικατοπτρίζει την λειτουργεία τις εφαρμογής. Δηλαδή αν αυτά που γράφονται, εφαρμόζονται και ως επίσης να μην εφαρμόζονται λειτουργείες που

αφορούν την ιδιωτικότητα οι οποίες δεν αναφέρονται στο privacy policy της εφαρμογής. Γενικά παρουσιάζεται ένα σύστημα το οποίο βοηθά τους προγραμματιστές να τηρούν τις υποχρεώσεις που έχουν να κάνουν με την ιδιωτικότητα. Για παράδειγμα ελέγχει ένα υπάρχει σύνδεσμος με το privacy policy στο Google Play Store για την συγκεκριμένη εφαρμογή, όπως επίσης αν υπάρχει και ο σύνδεσμος αυτός εντός της εφαρμογής.

Το paper 'A framework for static detection of privacy leaks in android applications' [32], παρουσιάζει ένα πλαίσιο εφαρμογής (framework) το οποίο ανιχνεύει παραβιάσεις Ιδιωτικότητας σε μια Android εφαρμογή με την χρήση στατικής ανάλυσης σε επίπεδο bytecode. Το framework που προτείνεται, υποστηρίζει και αναγνωρίζει όλες τις εντολές του Dalvik Virtual Machine (DVM). Σκοπός του framework αυτού είναι να ψάχνει στον κώδικα και να βρίσκει σε πιο σημείο γίνεται πρόσβαση σε ευαίσθητα δεδομένα (όπως τοποθεσία, id συσκευής, κτλ.) και να ελέγχει αν η μεταβλητή με την συγκεκριμένη πληροφορία αποστέλνεται άλλου (για παράδειγμα μέσω sockets). Αν υπάρχουν τέτοιες περιπτώσεις, τότε θα υποδείξει ότι οι συγκεκριμένες μεθόδους διαρρέουν προσωπικά δεδομένα.

Τέλος, το paper 'TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones' [33], περιγραφή την ανάγκη για την δημιουργία μιας εφαρμογής που ονομάζεται TaintDroid [34] και έχει σκοπώ να παρακολουθεί το πως οι άλλες εφαρμογές χρησιμοποιούν τις ευαίσθητες πληροφορίες και τους πόρους συστήματος. Αυτό γίνεται με χρήση δυναμικής αναλύσεις μιας εφαρμογής, αφού θα πρέπει να τρέχει η εφαρμογή για να μπορεί να παρακολουθήσει το TaintDroid τα δεδομένα. Με την ολοκλήρωση της υλοποίησης, μελετήθηκαν 30 τυχαίες δημοφιλείς εφαρμογές από το Google Play Store και το 60% αυτών των εφαρμογών διαχειρίζονταν με ύποπτο τρόπο τα ευαίσθητα δεδομένα. Επίσης οι μισές εφαρμογές έδειξαν να συλλέγουν την τοποθεσία των χρηστών οι οποίες αποστέλνονταν σε διαφημιστικούς servers.

4.3 Exodus Privacy

Το Exodus Privacy [20] είναι ένας μη κερδοσκοπικός οργανισμός που προσφέρει το exodus, το οποίο είναι μια διαδικτυακή πλατφόρμα ανάλυσης Android εφαρμογών. Η ανάλυση γίνεται για εντοπισμό θεμάτων ιδιωτικότητας που μπορεί να είναι επικίνδυνα προς τον χρήστη όπως βιβλιοθήκες για διαφημίσεις και συλλογή πληροφοριών, αναλύσεις σχετικές με το δίκτυο καθώς και γενικές πληροφορίες της εφαρμογής. Στο σύστημα περιέχονται όλες οι αναφορές με της πληροφορίες των αναλύσεων τον εφαρμογών και ο χρήστης μπορεί να δει πληροφορίες που τον ενδιαφέρουν. Το Exodus βασίζεται στην ανάλυση εφαρμογών οι οποίες

είναι διαθέσιμες από το Google Play Store, από το οποίο παίρνει και κάποιες πληροφορίες σχετικά με της εφαρμογές.

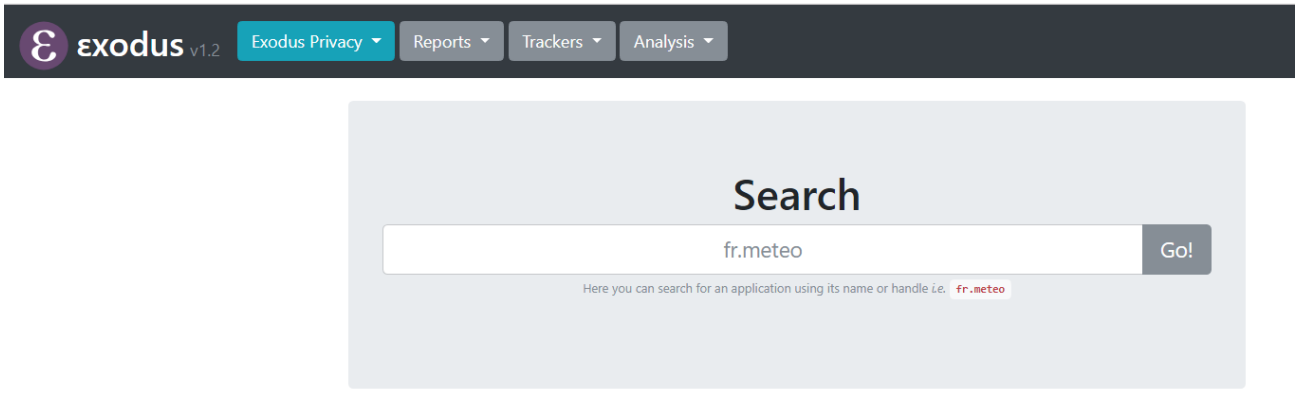
Τρόπος λειτουργίας

Μια από της λειτουργίες του exodus, είναι η στατική ανάλυση του κώδικα για την εύρεση κυρίως των διαφόρων trackers (δηλαδή βιβλιοθηκών που χρησιμοποιούνται στο σύστημα και πιθανών να συλλέγουν πληροφορίες από την εφαρμογή και την συσκευή του χρήστη) καθώς και για τα permissions που χρησιμοποιεί η εφαρμογή.

Για τον εντοπισμό των trackers σε μια εφαρμογή, το exodus έχει μια λίστα με τους trackers που θα αναζητήσει με τις πληροφορίες του καθενός. Στις πληροφορίες αυτές υπάρχει και ένα 'code signature' το οποίο είναι μοναδικό και στην ουσία είναι το πως θα εμφανίζεται στον κώδικα της εφαρμογής, έτσι ώστε να μπορεί να αναληθή. Η εύρεση κάποιου tracker σε κάποια εφαρμογή δεν σημαίνει απαραίτητα ότι η εφαρμογή αυτή το χρησιμοποιεί κάπου, αλλά μπορεί να γίνεται κάποιο λάθος ή άσκοπο import του συγκεκριμένου tracker. Για την εύρεση των permissions που χρησιμοποιεί η εφαρμογή, το exodus αναλύει το AndroidManifest.xml βλέποντας πια permissions είναι δηλωμένα από την εφαρμογή.

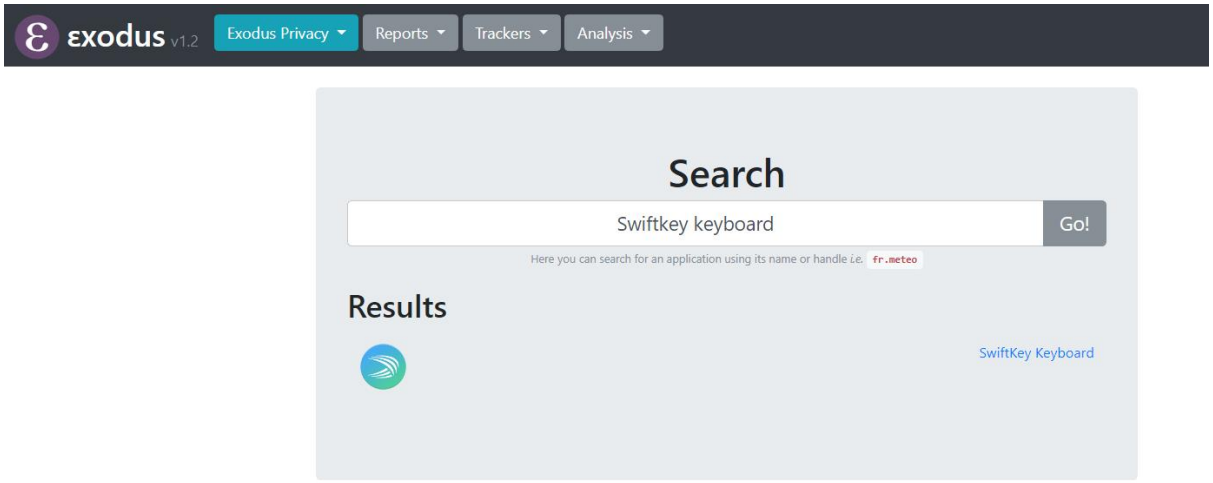
Μια άλλη λειτουργία που κάνει το exodus κατά την διάρκεια της ανάλυσης μιας εφαρμογής, είναι η δυναμική ανάλυση της εφαρμογής για συλλογή πληροφοριών που αφορούν το δίκτυο. Συγκεκριμένα το exodus εκτελεί την εφαρμογή σε εξομοιωμένη συσκευή και παρακολουθεί την κίνηση στο δίκτυο. Ελέγχει την κίνηση για τα πρωτόκολλα DNS, UDP, HTTP και HTTPS, ελέγχοντας τα δεδομένα που συλλέγονται καθώς και που αποστέλνονται.

Παραδείγματα από την χρήση του συστήματος (https://reports.exodus-privacy.eu.org/)



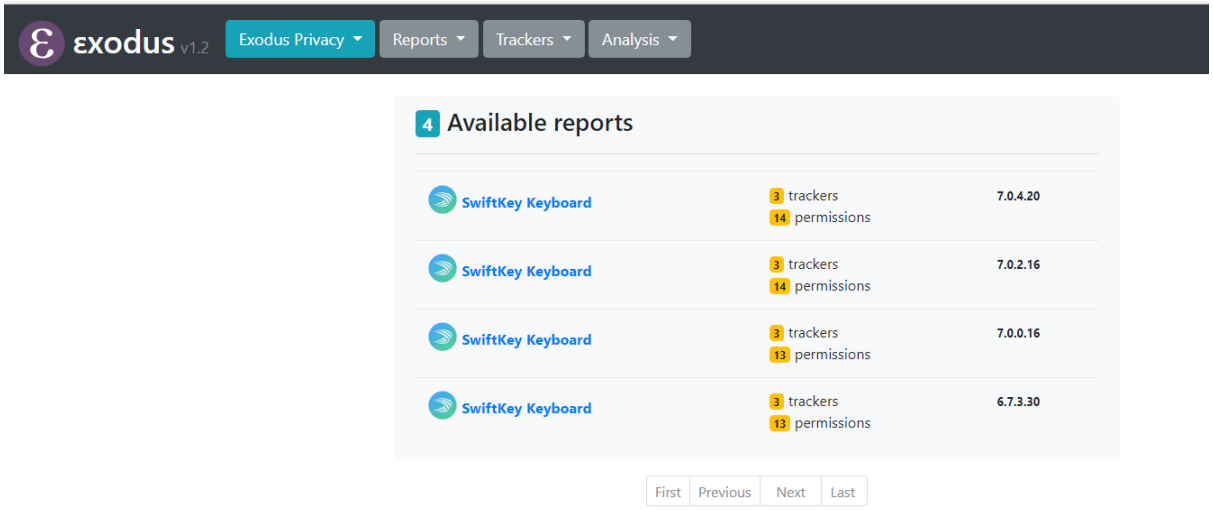
Σχήμα 4.1 Οθόνη αναζήτησης Exodus

Αρχικά, μπαίνοντας στην σελίδα του συστήματος υπάρχει μια μπάρα αναζήτησης (όπως φαίνεται και στο Σχήμα 4.1) όπου μπορεί κάποιος να αναζητήσει μια εφαρμογή, γράφοντας το όνομα της ή το όνομα του πακέτου και θα εμφανιστούν οι εφαρμογές που ταιριάζουν όπως φαίνεται και στο Σχήμα 4.2.



Σχήμα 4.2 Αποτελέσματα αναζήτησης

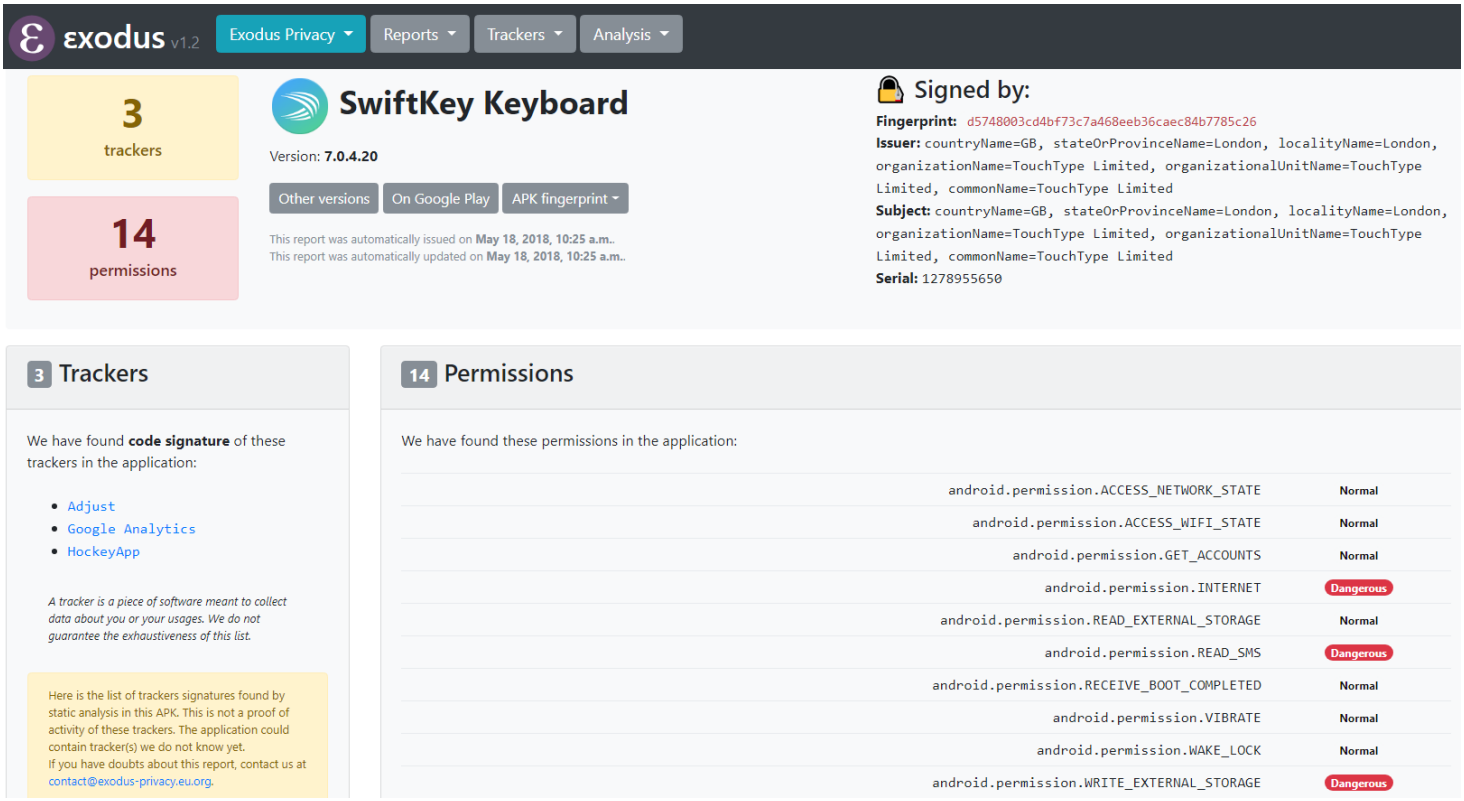
Επιλέγοντας την εφαρμογή η οποία είναι επιθυμητή, τότε εμφανίζεται μια λίστα με τις διάφορες εκδόσεις της εφαρμογής (αν υπάρχουν), όπως φαίνεται και στο Σήμα 4.3.



Σχήμα 4.3 Εκδόσεις εφαρμογής που αναλύθηκαν από το exodus.

Στην συνέχεια, όταν επιλεχτεί η κατάλληλη έκδοση της εφαρμογής εμφανίζονται οι πληροφορίες για την αντίστοιχη εφαρμογή (Σχήμα 4.4). Συγκεκριμένα, φαίνεται το όνομα, η έκδοση, το εικονίδιο της, πια permissions έχει δηλωμένα (από το AndroidManifest.xml) και ποια από αυτά είναι dangerous, καθώς και την υπογραφή αυτού που υπόγραψε την εφαρμογή. Επίσης υπάρχει και μια λίστα με τους trackers που υπάρχουν στην εφαρμογή.

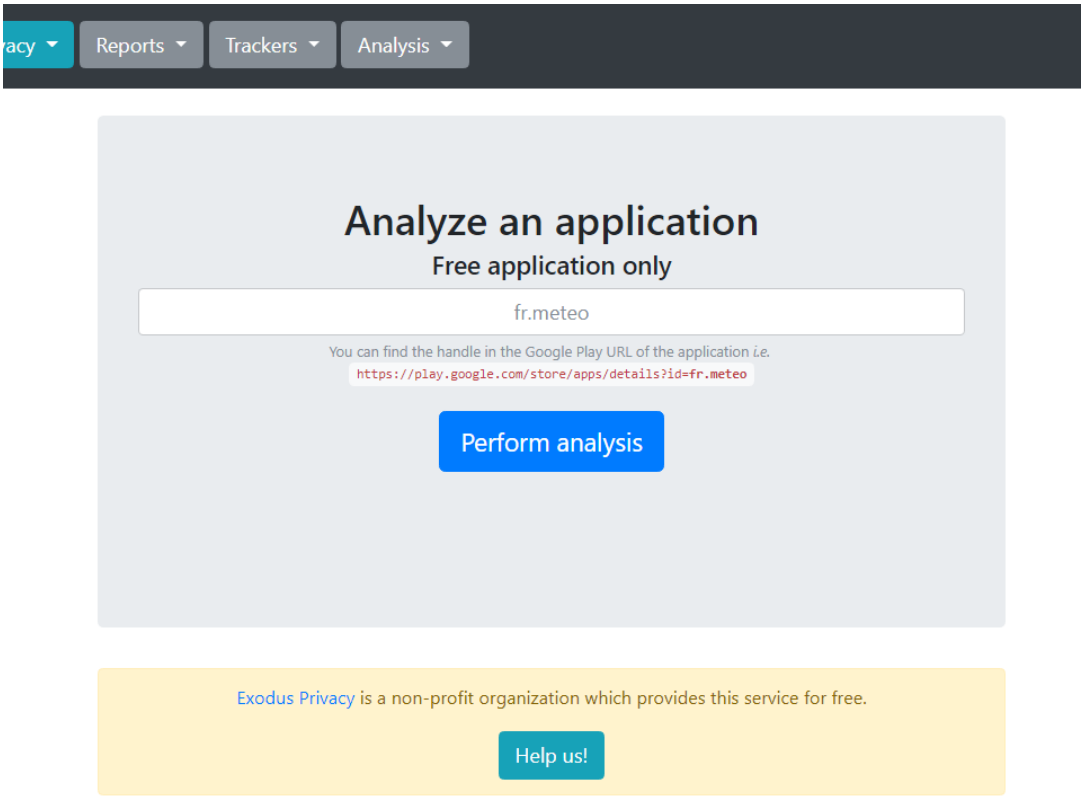
Ένας trackers, είναι μια βιβλιοθήκη που ονομάζεται έτσι διότι συνήθως παρακολουθεί την εφαρμογή και μπορεί να συλλέγει διάφορές πληροφορίες. Για παράδειγμα, εάν είναι βιβλιοθήκη για διαφημίσεις, μπορεί να συλλέγει διάφορες πληροφορίες για το πώς ο χρήστης χρησιμοποιεί την εφαρμογή και να του βγάλει κάποιες διάφορες σχετικές διαφημίσεις οι οποίες θα είναι πιο κοντά στα ενδιαφέροντα του για να του τραβήξει την προσοχή.



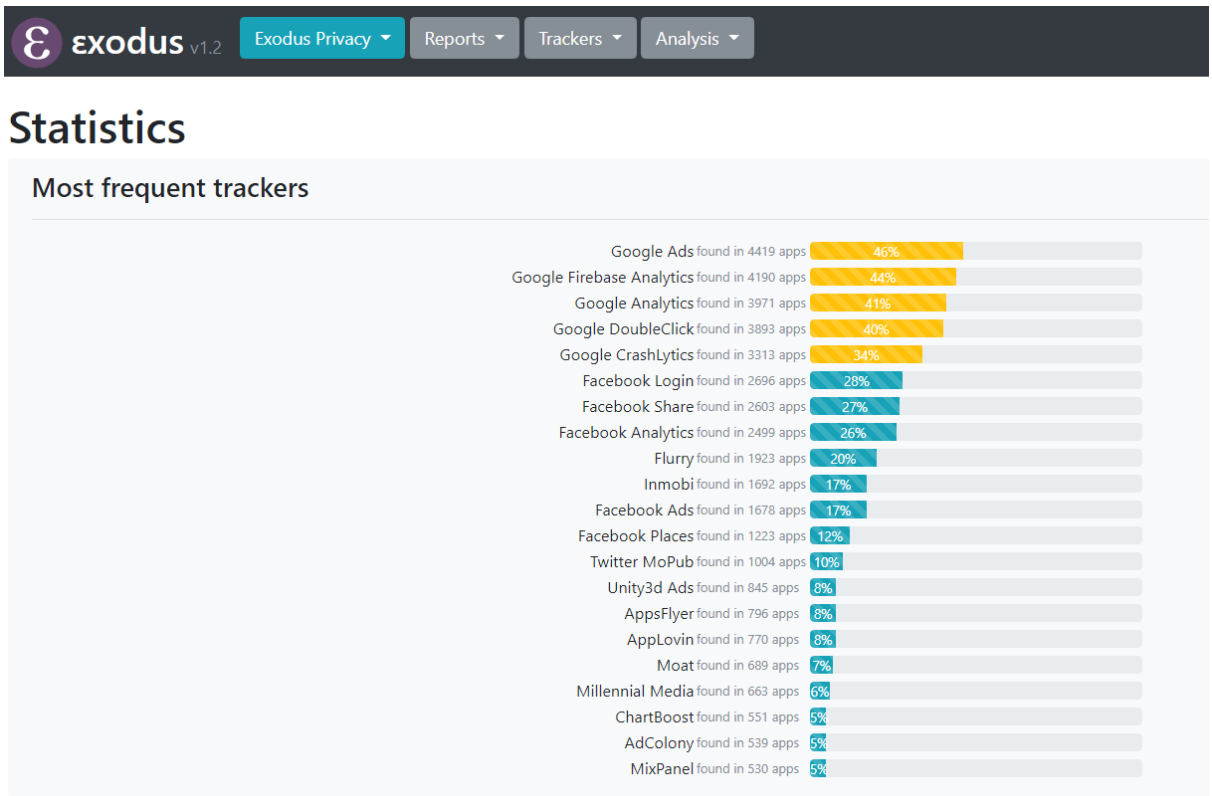
Σχήμα 4.4 Πληροφορίες από την ανάλυση μιας εφαρμογής

Εκτός από την αναζήτηση κάποιας εφαρμογής, ο χρήστης μπορεί να ζητήσει από το σύστημα να αναλύσει κάποια εφαρμογή, που υπάρχει στο Google Play Store και είναι δωρεάν, η οποία μπορεί να μην έχει αναλυθεί πιο πριν. Πατώντας Analysis και στην συνέχεια Submit, εμφανίζεται η σελίδα όπου μπορεί να γίνει αυτό, όπως φαίνεται και στο Σχήμα 4.5. Υπάρχει μια μπάρα, όπου ο χρήστης θα πρέπει να δώσει την διεύθυνση στην οποία μπορεί κάποιος να βρει την εφαρμογή αυτή στο Google Play Store.

Ακόμα υπάρχει και η επιλογή, να εμφανιστεί μια λίστα με όλους τους trackers που μπορεί να αναγνωρίσει το exodus (επιλογή Trackers και στην συνέχεια List), όπως επίσης και κάποια ποσοστά σχετικά με το πόσο συχνά εμφανίζεται ο κάθε tracker στις εφαρμογές που έχουν αναλυθεί (επιλογή Trackers και στην συνέχεια Statistics) όπως φαίνεται και στο Σχήμα 4.6.



Σχήμα 4.5 Σελίδα όπου ο χρήστης μπορεί να την διεύθυνση μιας εφαρμογής από το Google Play Store για ανάλυση



Σχήμα 4.6 Ποσοστά εμφάνισης του κάθε trackers στις εφαρμογές που έχουν αναλυθεί.

4.3 AVC UnDroid

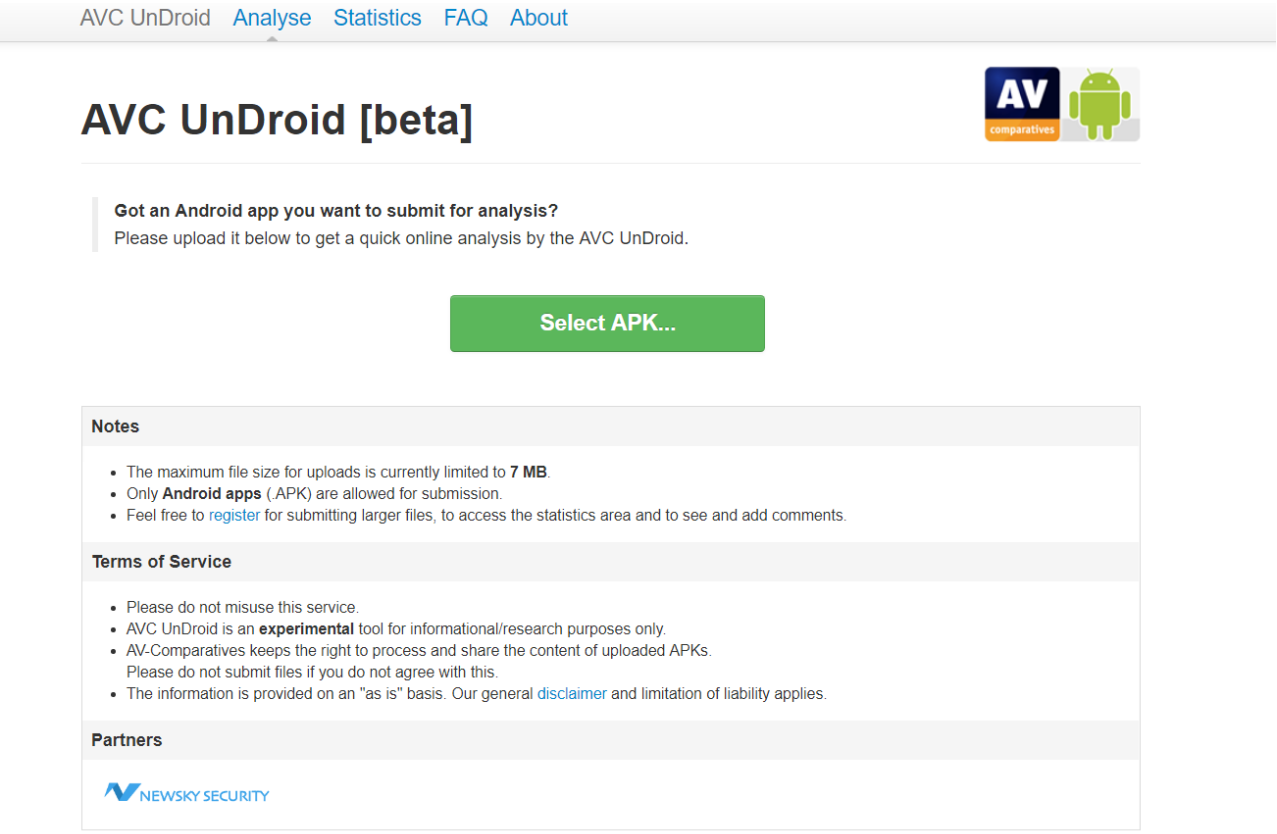
Το AVC UnDroid [35] ένα διαδικτυακό εργαλείο ανάλυσης εφαρμογών Android, όπου οι χρήστες μπορούν να ανεβάσουν κάποια εφαρμογή (APK αρχείο) και να δουν διάφορες πληροφορίες για την εφαρμογή αυτή. Το σύστημα αποθηκεύει την κάθε ανάλυση που κάνει, έτσι ώστε την επόμενη φορά που θα χρειαστεί να αναλυθεί το ίδιο αρχείο που μπορεί να το ανέβασε άλλος χρήστης, δεν θα ξαναγίνει η διαδικασία της ανάλυσης, αλλά θα εμφανιστούν τα αποτελέσματα. Το AVC UnDroid κάνει στατική αλλά και δυναμική ανάλυση μιας εφαρμογής χρησιμοποιώντας διάφορα εργαλεία.

Ένα από αυτά είναι το APKTool [36], το οποίο είναι ένα εργαλείο για reverse engineering κάποιας εφαρμογής Android. Μπορεί να αποκωδικοποιήσει τα αρχεία και να τα φέρει σε μια μορφή η οποία είναι πολύ κοντά (αν όχι και ίδια) στην αρχική. Επίσης έχει την δυνατότητα να μπορεί κάποιος να ξαναδημιουργήσει το αρχείο APK της εφαρμογής, μετά από τις αλλαγές που έχει κάνει. Σκοπός του δεν είναι η δημιουργία κακόβουλων εφαρμογών, αλλά το να δώσει την δυνατότητα σε μπορεί να προστεθεί επιπρόσθετη λειτουργικότητα σε κάποια εφαρμογή, όπως επίσης και για να βοηθήσει στην ανάλυση των εφαρμογών. Στην περίπτωση του AVC UnDroid, το συγκεκριμένο εργαλείο βοηθά στην αποκωδικοποίηση του AndroidManifest.xml αρχείου, έτσι ώστε να πάρει διάφορες πληροφορίες για την εφαρμογή, όπως όνομα, όνομα πακέτου, permissions που χρησιμοποιεί, κ.ά.

Το AVC UnDroid, χρησιμοποιεί επίσης το ssdeep [37] το οποίο υπολογίζει ένα hash του αρχείου με βάσει τα περιεχόμενα του το οποίο ονομάζεται Context Rriggered Piecewise Hash (CTPH) ή fuzzy hash [38]. Το hash μιας εφαρμογή, μπορεί να συγκριθεί με άλλα άλλων εφαρμογών για εύρεση κοινών bytes. Αυτό βοηθά στην εύρεση κακόβουλων εφαρμογών με το να υπάρχουν διάφορα fuzzy hashes, που είναι γνωστό ότι είναι από κακόβουλές εφαρμογές, και να γίνεται σύγκριση με hashes από άλλες εφαρμογές.

Το AVC UnDroid ανήκει στην εταιρεία, AV comparatives [39] και βοηθά στην έρευνα και εύρεση πληροφοριών για διάφορες εφαρμογές. Φαίνεται ότι υπάρχουν το εργαλείο παρέχει και διάφορα στατιστικά καθώς και περισσότερες πληροφορίες για τις εφαρμογές, όμως για κάποιο λόγο δεν είναι δυνατή η δημιουργία λογαριασμού για πρόσβαση στις λειτουργίες αυτές.
Μπαίνοντας στην σελίδα του AVC UnDroid, παρουσιάζονται κάποιοι όροι χρήσεις, όπως επίσης και κάποιο περιορισμοί όσο αφορά το μέγεθος του αρχείο της εφαρμογής το οποίο

δεν πρέπει να ξεπερνά τα 7 Megabytes. Όπως φαίνεται και στο σχήμα 4.7, υπάρχει επιλογή για το ανέβασμα κάποιου αρχείου για ανάλυση πατώντας το κουμπί ‘Select APK…’.



Σχήμα 4.7 Αρχική σελίδα του AVC UnDroid

Επιλέγοντας και ανεβάζοντας κάποιο αρχείο μιας εφαρμογής και αφού γίνει η ανάλυση θα εμφανιστούν διάφορες πληροφορίες που την αφορούν (Σχήμα 4.8). Επίσης όπως φαίνεται και στο Σχήμα 4.8, υπάρχει και μια εικόνα με κάποιες μπάρες οι οποίες απ’ ότι φαίνεται δείχνουν πόσο ‘επικίνδυνη’ είναι η εφαρμογή με βάση τα permissions που χρησιμοποιεί. Δεν αναφέρεται κάπου πως ακριβώς υπολογίζεται και τι σημαίνει η εικόνα αυτή. Ακόμα φαίνεται ότι παρουσιάζονται και τα permissions τα οποία δεν είναι δηλωμένα αλλά χρησιμοποιούνται (Σχήμα 4.8 – λίστα Used permissions). Μετά τα permissions παρουσιάζονται κάποιες κλήσεις σε μεθόδους που αφορούν permissions που χρησιμοποιούνται στον κώδικα της εφαρμογής (Σχήμα 4.9). Επίσης διαχωρίζονται οι κλήσεις σε μεθόδους οι οποίες κάνουν χρήση dangerous permissions (Σχήμα 4.10). Στο ίδιο σχήμα, εμφανίζονται και τα διάφορα στοιχεία που έχει μια εφαρμογή Android όπως intents, receivers και services. Τέλος, υπάρχει και μια λίστα με τα Uniform Resource Locators (URLs) τα οποία είναι hardcoded στον κώδικα της εφαρμογής.

AVC UnDroid Analyse Statistics FAQ About

AVC UnDroid [beta]

Report

DateTime	2013-05-10 01:07:13 (Last analysis)
MD5	98cfa989d78eb85b86c497ae5ce8ca19
SHA1	8d0cc1dab447130ab99528be89681f2f35d0294e
SHA256	89a8c758f45d86bf0aee1496fd48036fad55805927327d89af6ec1d7337a3938
Filesize	3.4 MB (3354613 Byte)
Filename	app3.apk
Packagename	live.photo.savanna
ssdeep APK	98304:PEg+qFeur0Pa6MO77fMmEOUQWa5Nw3+br:sg+qYuleUfSanus
SHA256 DEX	c6f8b8d5444bbe55a007e5c475b748f835af8cdd95fff43bb0615d4dec21445f
ssdeep DEX	12288:mAi9zN7eJLI4+kF5XWWV5+yk92wdtlwYyAqTKztLbi5etFSH4Mjm:mAisLKmJNATKzRomSbq
Date DEX	16-M?r-13
Ad-supported	No

Σχήμα 4.8 Αποτελέσματα ανάλυσης εφαρμογής από το AVC UnDroid (1)

Requested Permissions

- android.permission.ACCESS_NETWORK_STATE
- android.permission.ACCESS_WIFI_STATE
- android.permission.INTERNET
- android.permission.READ_PHONE_STATE
- android.permission.RECEIVE_BOOT_COMPLETED
- android.permission.SYSTEM_ALERT_WINDOW
- android.permission.VIBRATE
- android.permission.WRITE_EXTERNAL_STORAGE
- com.android.launcher.permission.INSTALL_SHORTCUT

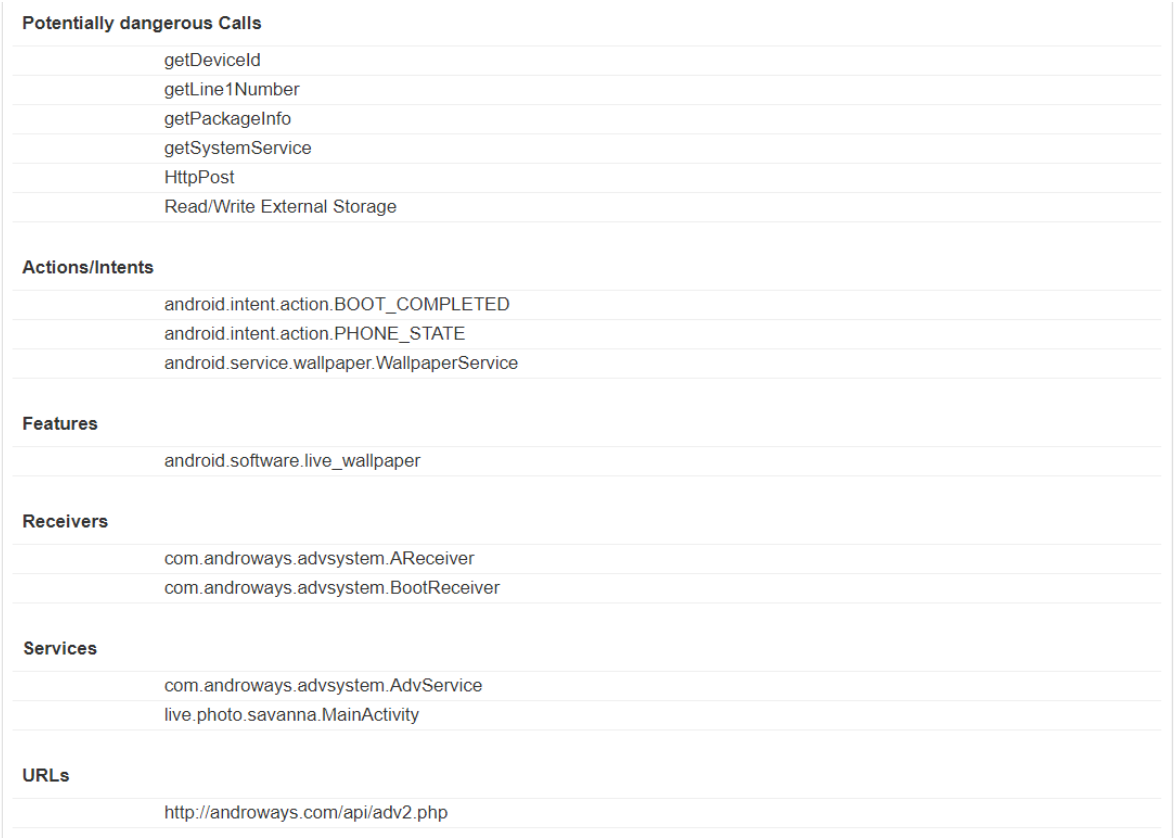
Used Permissions

- android.permission.WAKE_LOCK

Responsible API calls for used Permissions

- android/app/Activity;->startActivity
- android/app/NotificationManager;->notify
- android/content/ContentResolver;->query
- android/content/Context;->sendBroadcast
- android/content/Context;->startActivity
- android/content/Context;->startService
- android/location/LocationManager;->getBestProvider
- android/location/LocationManager;->getLastKnownLocation
- android/location/LocationManager;->requestLocationUpdates
- android/media/MediaPlayer;->start
- android/media/MediaPlayer;->stop
- android/net/ConnectivityManager;->getActiveNetworkInfo
- android/net/ConnectivityManager;->getNetworkInfo

Σχήμα 4.9 Αποτελέσματα ανάλυσης εφαρμογής από το AVC UnDroid (2)



Potentially dangerous Calls	
	getDeviceId
	getLine1Number
	getPackageInfo
	getSystemService
	HttpPost
	Read/Write External Storage
Actions/Intents	
	android.intent.action.BOOT_COMPLETED
	android.intent.action.PHONE_STATE
	android.service.wallpaper.WallpaperService
Features	
	android.software.live_wallpaper
Receivers	
	com.androways.advsystem.AReceiver
	com.androways.advsystem.BootReceiver
Services	
	com.androways.advsystem.AdvService
	live.photo.savanna.MainActivity
URLs	
	http://androways.com/api/adv2.php

Σχήμα 4.10 Αποτελέσματα ανάλυσης εφαρμογής από το AVC UnDroid (3)

4.5 Σύγκριση συστήματος με υπάρχοντα

Αφού παρουσιάστηκαν παρόμοια συστήματα και πως λειτουργούν, στην ενότητα αυτή θα γίνει μια σύγκριση των συστημάτων αυτών με το σύστημα που αναπτύχθηκε στην παρούσα ατομική διπλωματική εργασία.

Αρχικά, συγκρίνοντας το σύστημα που υλοποιήθηκε με το exodus, παρατηρούμε ότι η βασική διαφορά είναι ότι στο exodus αναλύονται μόνο εφαρμογές από το Google Play Store, ενώ στο σύστημα που υλοποιήθηκε οι χρήστες μπορούν να ανεβάσουν οποιαδήποτε εφαρμογή (αρχείο APK), φτάνει το μέγεθος της να είναι μικρότερο από 30 Megabytes. Το exodus, αφού κατεβάσει το APK από το Google Play Store με την βοήθεια του Androguard εξάγονται διάφορες πληροφορίες για την εφαρμογή όπως το όνομα, το version code, το όνομα του πακέτου, το checksum όπου αυτές οι πληροφορίες θεωρούνται σαν το APK fingerprint από το exodus. Μέσα από το Google Play Store γίνεται εξαγωγή του εικονιδίου (icon) της εφαρμογής. Ακόμα εξάγονται και τα permissions που είναι δηλωμένα στο AndroidManifest.xml όπως επίσης και το signature, δηλαδή το ποιος υπόγραψε το αρχείο APK. Στο εργαλείο που αναπτύχθηκε με το που ανεβεί κάποιο αρχείο και ξεκινήσει η

ανάλυση ακολουθείτε η ίδια διαδικασία με το Androguard και εξάγονται αυτές οι πληροφορίες εκτός του signature. Εκτός από αυτές τις πληροφορίες, εξάγονται από το AndroidManifest.xml και η έκδοση του Android που απαιτείται από την εφαρμογή για να τρέξει, όπως επίσης και την προτεινόμενη έκδοση. Μια άλλη διαφορά είναι ότι στο exodus εμφανίζονται μόνο τα permissions που είναι δηλωμένα στο AndroidManifest.xml, ανεξάρτητα αν χρησιμοποιούνται ή όχι, ενώ στο σύστημα που αναπτύχθηκε γίνεται εύρεση και των permissions που δεν είναι δηλωμένα και χρησιμοποιούνται καθώς επίσης και αν κάποιο permission που είναι δηλωμένο, χρησιμοποιείτε ή όχι στην εφαρμογή. Επίσης μια ακόμα διαφορά είναι ότι στο σύστημα που υλοποιήθηκε γίνεται εύρεση των μεθόδων οι οποίες αφορούν κάποια permissions και παρουσιάζονται έτσι ώστε να γνωρίζει σε πιο σημείο του κώδικα γίνεται η κλήση της συνάρτησης αυτής και πια μέθοδο καλεί.

Το exodus κάνει εύρεση διαφόρων trackers, όπως αναφέρθηκε και πιο πάνω, στις εφαρμογές. Η εύρεση αυτή γίνεται μέσω αναζήτησης κάποιων code signatures στον κώδικα της εφαρμογής. Τα code signatures αυτά είναι στην ουσία τα πακέτα της εφαρμογής του κάθε tracker τα οποία βρίσκονται σε μια λίστα στο exodus [40]. Στο σύστημα που υλοποιήθηκε, υιοθετήθηκε η εύρεση αυτών των trackers, όμως με διαφορετικό τρόπο. Αντί να γίνεται αναζήτηση των code signatures στον κώδικα της εφαρμογής, γίνεται αναζήτηση των trackers μέσα από τις βιβλιοθήκες που εξήγαγε το LibRadar, οι οποίες χρησιμοποιούνται στην εφαρμογή. Αυτό μπορεί να έχει κάποια μειονεκτήματα αφού το LibRadar μπορεί να μην αναγνωρίσει κάποιες βιβλιοθήκες, με αποτέλεσμα να μην υπάρχουν κατά την αναζήτηση των trackers. Με την χρήση όμως του LibRadar, γίνεται εύρεση και των permissions που χρησιμοποιούνται από τις βιβλιοθήκες που χρησιμοποιούνται στην εφαρμογή.

Έχοντας υπόψη τα πιο πάνω για το σύστημα που υλοποιήθηκε θα γίνει μια σύγκριση με το AVC UnDroid. Αρχικά το AVC UnDroid, εξάγει τις ίδιες πληροφορίες για την εφαρμογή αλλά αυτό γίνεται με την χρήση του APKTool, αντί για το Androguard. Όπως αναφέρθηκε και πιο πριν με το APKTool αποκωδικοποιούνται τα αρχεία και ευκολά εξάγονται οι πληροφορίες αυτές από το AndroidManifest.xml. Επίσης εκτός από το SHA256 hash, το AVC UnDroid υπολογίζει και τα MD5 και SHA1, όπως επίσης και το ssdeep hash το οποίο αναφέρθηκε και πιο πριν. Τα SHA256 και ssdeep, υπολογίζονται και για το .dex αρχείο το οποίο περιέχει τον κώδικα του APK.

Σχετικά με τα permissions και τα 2 συστήματα παρουσιάζουν τα permissions τα οποία είναι δηλωμένα στο AndroidManifest.xml, καθώς επίσης και αυτά που χρησιμοποιούνται και δεν είναι δηλωμένα. Το σύστημα που υλοποιήθηκε, παρουσιάζει επιπλέον και ποια από τα

δηλωμένα permissions χρησιμοποιούνται ή όχι στον κώδικα της εφαρμογής. Ένα άλλο κοινό στοιχεία είναι το ότι παρουσιάζονται οι κλήσεις σε μεθόδους των permissions αλλά στο AVC UnDroid δεν εμφανίζεται από ποιο σημεία του κώδικα της εφαρμογές γίνεται η κλήση αυτή, αντίθετα με το σύστημα που υλοποιήθηκε το οποίο εμφανίζει σε ποιο σημείο του κώδικα γίνονται οι κλήσεις μεθόδων που αφορούν τα permissions. Το AVC UnDroid, παρουσιάζει και μια λίστα με τις κλήσεις σε μεθόδους που θεωρούνται επικίνδυνες και χρησιμοποιεί η εφαρμογή. Επιπλέον, εμφανίζονται και διάφορα στοιχεία που έχει μια εφαρμογή Android όπως intents, receivers, services και μια λίστα με τα Uniform Resource Locators (URLs) τα οποία είναι hardcoded στον κώδικα της εφαρμογής. Αυτά τα στοιχεία δεν εξάγονται από το σύστημα που υλοποιήθηκε αφού δεν παίζουν ρόλο στην ανάλυση που κάνει το σύστημα και γενικότερα για την Ιδιωτικότητα.

Τέλος, εκτός από τα πιο πάνω, το σύστημα που υλοποιήθηκε κάνει μια ανάλυση των permissions και υπολογίζει κατά πόσο η εφαρμογή κρίνεται επικίνδυνή ή όχι με βάση τα permissions που χρησιμοποιεί. Επίσης, με βάση τα permissions, κάποιες λειτουργίες που χρησιμοποιεί και το αποτέλεσμα του αν είναι επικίνδυνη ή όχι, παράγεται ένα score επικινδυνότητας τις κάθε εφαρμογής. Το exodus δεν κάνει κάτι αντίστοιχο, αλλά απλά παρουσιάζει κάποιες πληροφορίες για την εφαρμογή. Το AVC UnDroid, με βάση τα permissions δημιουργεί μια μπάρα που πιθανών να έχει σχέση με την επικινδυνότητα των permission που χρησιμοποιεί η εφαρμογή. Δεν υπάρχει κάποια εξήγηση για το πως υπολογίζεται ή ποιο σκοπό εξυπηρετεί.

Κεφάλαιο 5

Τεχνολογίες και Εργαλεία που χρησιμοποιήθηκαν

5.1 Εισαγωγή

Στο κεφάλαιο αυτό παρουσιάζονται οι τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν για την ανάλυση και την ανάπτυξη του διαδικτυακού συστήματος ανάλυσης εφαρμογών. Σε πρώτη φάση, παρουσιάζονται οι γλώσσες προγραμματισμού που χρησιμοποιήθηκαν αναφέροντας το πως δουλεύουν και που ακριβώς χρησιμοποιήθηκαν στο σύστημα. Στην συνέχεια παρουσιάζονται τα πλαίσια εργασίας (frameworks) όπως επίσης και οι βιβλιοθήκες που χρησιμοποιήθηκαν για την υλοποίηση. Επίσης, παρουσιάζονται κάποια εργαλεία που βοήθησαν στην ανάλυση και γενικότερα στην ανάπτυξη του συστήματος. Τα εργαλεία αυτά θα επεξηγηθούν για το πως λειτουργούν καθώς επίσης και στο πώς χρησιμοποιήθηκαν και βοήθησαν κατά την διάρκεια της ανάπτυξης.

5.2 Γλώσσες προγραμματισμού

5.2.1 Java

Η Java [41] είναι μια αντικειμενοστεφής γλώσσα προγραμματισμού η οποία δημιουργήθηκε από την Sun Microsystems το 1995. Ο βασικός σκοπός της είναι να λύσει το πρόβλημα δημιουργίας εφαρμογών για διαφορετικές μηχανές και εισάγει την έννοια του αντικείμενου στο προγραμματισμό. Με την βοήθεια του Java Virtual Machine (JVM), οι εφαρμογές που είναι γραμμένες σε Java, μπορούν να εκτελεστούν σε οποιαδήποτε μηχανή, φτάνει να υπάρχει εγκατεστημένο το απαιτούμενο ενδιάμεσο λογισμικό. Ένα πλεονέκτημα της Java, είναι ότι προσφέρετε δωρεάν και διαθέτει πολλές βιβλιοθήκες που περιέχουν διάφορες δομές δεδομένων και αλγόριθμους το οποίο βοηθά στη γρήγορη ανάπτυξη εφαρμογών και νέων αλγορίθμων.

Η Java χρησιμοποιήθηκε στην παρούσα ατομική διπλωματική εργασία για την υλοποίηση του εργαλείου ανάλυσης, όπως επίσης και για την δημιουργία της διαδικτυακής εφαρμογής, μέσα από την οποία θα γίνεται εύκολη χρήση του εργαλείου ανάλυσης που υλοποιήθηκε.

5.2.2 Python

Η Python [42] είναι μια γλώσσα προγραμματισμού υψηλού επιπέδου η οποία δημιουργήθηκε από τον Guido van Rossum το 1990. Ο βασικός σκοπός της είναι η αναγνωσιμότητα του κώδικά της, η ευκολία χρήσης της και το συντακτικό της, το οποίο επιτρέπει στους προγραμματιστές να εκφράσουν έννοιες και να δημιουργήσουν προγράμματα με λιγότερες γραμμές κώδικα σε σύγκριση με άλλες γλώσσες προγραμματισμού. Η Python επίσης διακρίνεται για τις πολλές βιβλιοθήκες που διαθέτει οι οποίες διευκολύνουν τον προγραμματιστή και βοηθούν για την ταχύτητα εκμάθησης της ιδίας της γλώσσας, όπως επίσης και για την ανάπτυξη προγραμμάτων. Μπορεί να χρησιμοποιηθεί σε σχεδόν όλες τις μηχανές αφού ο interpreter, ο οποίος είναι υπεύθυνος στην μετατροπή του κώδικα σε bytecode, είναι διαθέσιμος σε διάφορες έκδοσης ανάλογα με το λειτουργικό σύστημα.

Η Python χρησιμοποιήθηκε στην παρούσα ατομική διπλωματική εργασία για στην υλοποίηση του εργαλείου ανάλυσης, αφού χρησιμοποιήθηκαν άλλα εργαλεία και βιβλιοθήκες που ήταν σε Python, έτσι έγινε μια ενσωμάτωση των εργαλείων και βιβλιοθηκών αυτών καθώς επίσης και το πως θα είναι δομημένα τα αποτελέσματα των

λειτουργιών που υλοποιήθηκαν σε Python, για να μπορούν εύκολο να χρησιμοποιηθούν και να ενσωματωθούν τα κομμάτια αυτά στην κύρια εφαρμογή.

5.2.3 Hyper Text Markup Language – HTML

Η HMTL [43] γλώσσα είναι γλώσσα παρουσίασής περιεχομένου και όχι γλώσσα προγραμματισμού ή γλώσσα σεναρίων, η οποία δημιουργήθηκε από το World Wide Web Consortium [44] και εμφανίστηκε το 1993. Σκοπός της είναι η δημιουργία η παρουσίαση δομημένης πληροφορίας η οποία χρησιμοποιείται από τους φυλλομετρητές ιστού για να παρουσιάσουν διάφορα στοιχεία που αποτελούν μια σελίδα.

Η HTML χρησιμοποιήθηκε στην παρούσα ατομική διπλωματική εργασία για την υλοποίηση της γραφικής διαπροσωπίας της διαδικτυακής εφαρμογής.

5.2.4 Cascading Styling Sheet –CSS

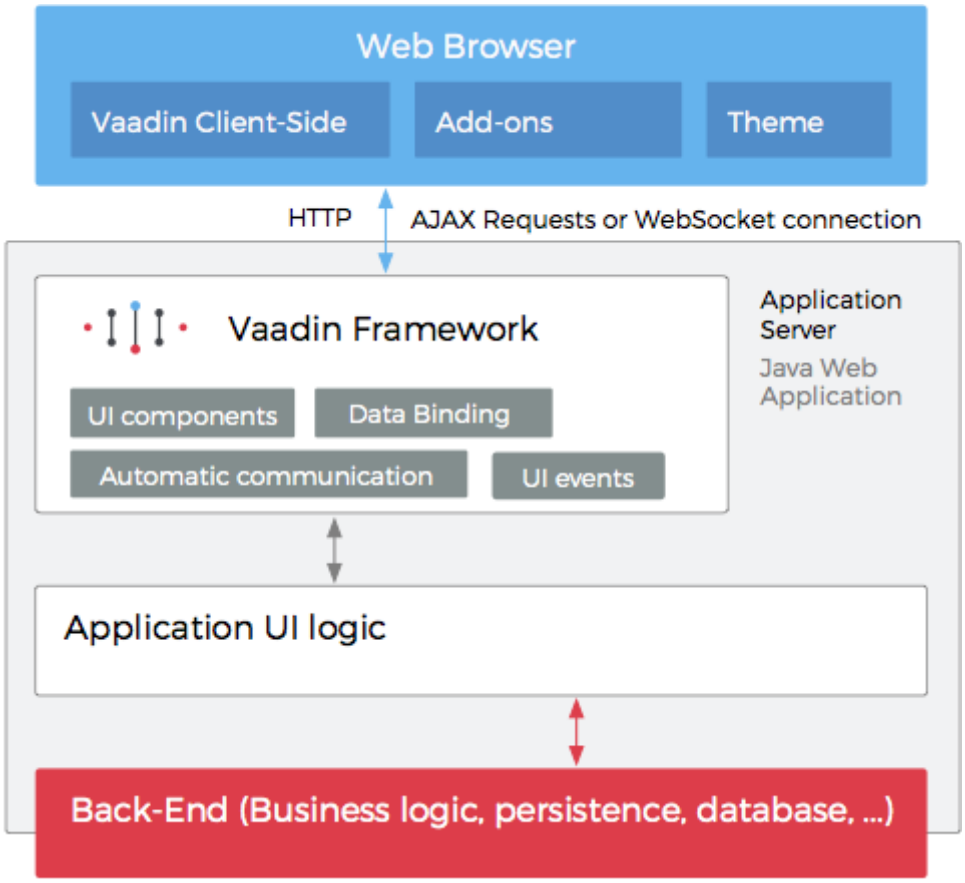
Με την δημιουργία της HTML, εμφανίστηκε η ανάγκη για δημιουργία κάποιας γλώσσας η οποία θα έχει σκοπό την μορφοποίηση του περιεχομένου. Γι’ αυτό λοιπόν δημιουργήθηκε η CSS [45], η οποία κάνει αυτή την δουλεία, δηλαδή κάνει μορφοποίηση του περιεχομένου. Συνδυάζεται με την χρήση της HTML και σκοπός της είναι να παρουσιάσει την πληροφορία με κατανοητό τρόπο προς τους χρήστες, όπως για παράδειγμα η χρήση χρωμάτων, διαφόρων γραμματοσειρών, κτλ.

Η CSS χρησιμοποιήθηκε στην παρούσα ατομική διπλωματική εργασία για την υλοποίηση της γραφικής διαπροσωπίας της διαδικτυακής εφαρμογής.

5.3 Vaadin Framework

Το Vaadin Framework [46] είναι ένα παραγωγική και εύκολο στην χρήση βιβλιοθήκη UI (User Interface) για την ανάπτυξη διαδικτυακών εφαρμογών σε Java ή άλλη γλώσσα που χρησιμοποιεί το JVM. Το Vaadin υιοθετεί μια προσέγγιση βασισμένη σε ανάπτυξη με συστατικά στοιχεία για την δημιουργία των διαδικτυακών εφαρμογών, με παρόμοιο τρόπο με την δημιουργία κλασικών εφαρμογών. Χρησιμοποιείται η προσέγγιση του αντικειμενοστρεφούς προγραμματισμού για την δημιουργία των UI, από μικρότερα στοιχεία και με την χρήση κάποιων event listeners, μπορεί κάποιος να προσθέσει την λειτουργικότητα που χρειάζεται. Βασικό χαρακτηριστικό του framework αυτού, είναι ότι εξοικονομεί χρόνο,

διότι δεν χρειάζεται να προγραμματίζεις σε HTML,CSS ή JavaScript [47], αλλά με την χρήση της Java μπορείς να δημιουργήσεις αντίστοιχα UIs. Επίσης το Vaadin αφαιρεί την επικοινωνία μεταξύ του διακομιστή και του προγράμματος περιήγησης. Αυτό καθιστά ευκολότερη την ανάπτυξη του UI και του backend (αφού δεν χρειάζεται να εκθέσει τις υπηρεσίες REST) και αυξάνει την ασφάλεια. Το Vaadin Framework είναι ανοιχτού κώδικα framework και μπορεί πολύ εύκολα κάποιος να κάνει κάποιες αλλαγές.



Σχήμα 5.1 Αρχιτεκτονική του Vaadin Framework

Το Vaadin χωρίζεται σε 2 κομμάτια, ένα server-side το οποίο τρέχει στον application server [48] (π.χ. Apache Tomcat [49]) και ένα client-side το οποίο τρέχει στον Web Browser του χρήστη και στην ουσία εμφανίζει το γραφικό περιβάλλον με την βοήθεια της JavaScript (ο κώδικας της οποίας παράγεται αυτόματα από το Vaadin) και μεταφέρει τις ενέργειες του χρήστη στον server. Με τον διαχωρισμό αυτό, δηλαδή στην εμφάνιση του user interface και στη λογική του, παρέχεται η δυνατότητα στον προγραμματιστή να υλοποιεί ξεχωριστά αυτά τα κομμάτια. Αυτό το επιτυγχάνει με τη χρήση themes που καθορίζουν την εμφάνιση των διαφόρων components. Αυτό γίνεται με τη χρήσης CSS και (προαιρετικά) HTML. Ένα άλλο σημαντικό στοιχεία είναι ότι το Vaadin προσφέρει κάποια default themes τα οποία μπορεί να τα χρησιμοποιήσει κάποιος που δεν θέλει να ασχοληθεί με CSS και HTML και να έχει μία

πολύ ευπαρουσίαστη εφαρμογή. Το Vaadin, χρησιμοποιήθηκε για την δημιουργία του UI καθώς επίσης και για την δημιουργία του client-side με του server-side.

5.4 Spring Framework

Το Spring [50] είναι ένα ανοιχτού κώδικα framework το οποίο έχει σκοπό να μειώσει την πολυπλοκότητα ανάπτυξης Java enterprise (J2EE) εφαρμογών. Χρησιμοποιεί JavaBeans για την επίτευξη των server-side λειτουργιών. Επίσης με την χρήση του Spring, γίνεται απλοποίηση του κώδικα για ευκολότερη ενσωμάτωση καινούργιων λειτουργιών σε μια εφαρμογή όπως επίσης και αποτελεσματική διαδικασία ελέγχου. Το Spring Framework αποτελείται από έναν Container βασισμένο στο Inversion of Control (IoC) ή αλλιώς Dependency Injection. Πρόκειται για μία τεχνική όπου υποδεικνύει σε ένα κομμάτι εφαρμογής ποια αλλά κομμάτια μπορεί να χρησιμοποιεί. Επιπλέον, βοηθά στην σύνδεση με πολλές άλλες δημοφιλείς τεχνολογίες και αυτό το αναδεικνύει σε ένα εξαιρετικά ευέλικτο πλαίσιο ανάπτυξης εφαρμογών.

5.5 Weka

Το WEKA (Waikato Environment for Knowledge Analysis) [19] είναι μια σουίτα λογισμικού για μηχανική μάθηση (machine learning) και εξόρυξη δεδομένων (data mining). Δημιουργήθηκε στο Πανεπιστήμιο του Waikato της Ν. Ζηλανδίας το 1997 και πήρε το όνομα του από το Weka, ένα μικρό και υπό εξαφάνιση πουλί της Ν. Ζηλανδίας. Είναι ένα από τα πιο διαδεδομένα λογισμικά για το σκοπό αυτό και έχει χρησιμοποιηθεί σε διάφορα επιστημονικά έργα. Το Weka παρέχει μια βιβλιοθήκη σε Java, όπου κάποιος μπορεί να το χρησιμοποιήσει σε κώδικα, αλλά επίσης περιέχει και ένα γραφικό περιβάλλον εργασίας το οποίο επιτρέπει τη χρήση του από τελικούς χρήστες οι οποίοι δεν έχουν προγραμματιστικές γνώσεις.

Το WEKA χρησιμοποιήθηκε και με τους 2 τρόπους στην παρούσα ατομική διπλωματικής εργασία στο σημείο που γίνεται ο χαρακτηρισμός κάποιας εφαρμογές ως επικίνδυνης με την χρήση μηχανικής μάθησης.

5.6 MySQL

Η MySQL [51] είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS) το οποίο ξεπερνά τις 11 εκατομμύρια εγκαταστάσεις. Έλαβε το όνομά του από την κόρη του

Μόντυ Βιντένιους [52], την Μάι (My). Το πρόγραμμα σε τρέχει έναν εξυπηρετητή (server) παρέχοντας πρόσβαση πολλών χρηστών σε ένα σύνολο βάσεων δεδομένων. Η βάση δεδομένων MySQL έχει γίνει η πιο δημοφιλής βάση δεδομένων ανοιχτού λογισμικού εξαιτίας της σταθερά υψηλής απόδοσής της, της αξιοπιστίας της και της ευκολίας της χρήσης της. Είναι παγκόσμιος γνωστή, τόσο από μεμονωμένους δημιουργούς διαδικτυακών χώρων όσο και από πολλούς από τους μεγαλύτερους και τους πιο ραγδαία αναπτυσσόμενους οργανισμούς.

Η MySQL χρησιμοποιήθηκε για την αποθήκευση των πληροφορίων που εξάγονται κατά την ανάλυση μιας εφαρμογής, όπως επίσης και για την αποθήκευση των λογαριασμών των χρηστών και άλλως πληροφοριών που είναι απαραίτητες για το σύστημα.

5.7 Androguard

Το Androguard [53] είναι ένα εργαλείο το οποίο είναι γραμμένο σε γλώσσα προγραμματισμού Python, το οποίο παρέχει διάφορες μεθόδους για αντίστροφη μηχανική (reverse engineering) και ανάλυση εφαρμογών Android. Ο σκοπός του είναι να μπορεί κάποιος να χρησιμοποιήσει τα διάφορα στοιχεία που απαρτίζουν μια εφαρμογή Android σε κώδικα στην γλώσσα python.
Υποστηρίζει διάφορες δυνατότητες όπως:

- Διαχείριση των κωδικοποιημένων αρχείων DEX/ODEX/APK/AXML/ARSC και μετατροπή τους σε αντικείμενα στην Python.
- Πρόσβαση στον κώδικα της εφαρμογής (αναζήτηση συναρτήσεων, μεταβλητών, κτλ.)
- Εξαγωγή πληροφοριών όσον αφορά το certificate της εφαρμογής για εύρεση repacked εφαρμογών.
- Μετατροπή των binary xml (π.χ. AndroidManifest.xml) σε κανονική xml μορφή.

Χρησιμοποιείται σε διάφορα σημεία στην παρούσα ατομική διπλωματική εργασία, συγκεκριμένα στην αποκωδικοποίησή και εξαγωγή πληροφορίων από το AndroidManifest αρχείο και στην αναζήτηση κλήσεων μεθόδων που αφορούν κάποια permissions.

5.8 LibRadar

Το LibRadar [54] είναι ένα αυτοματοποιημένο εργαλείο εύρεσης βιβλιοθηκών σε εφαρμογές Android. Έχει την δυνατότητα να εντοπίζει βιβλιοθήκες γρήγορα και με ακρίβεια, χρησιμοποιώντας μηχανική μάθηση. Είναι εκπαιδευμένο με περισσότερες από 1

εκατομμύριο εφαρμογές από το Google Play Store και μπορεί να αναγνωρίσει τις βιβλιοθήκες οι οποίες είναι πιο δημοφιλείς σε οποιαδήποτε εφαρμογή. Επίσης το LibRadar μπορεί να εντοπίσει τις βιβλιοθήκες ακόμα και σε obfuscated [55] εφαρμογές.

Χρησιμοποιήθηκε για αυτόν ακριβώς τον λόγο, δηλαδή να εντοπίζονται οι βιβλιοθήκες που χρησιμοποιεί μια εφαρμογή, όπως επίσης και τα permissions που χρησιμοποιεί η βιβλιοθήκη αυτή.

5.9 PScout

Το PScout [16] είναι ένα εργαλείο το οποίο έχει σκοπό την εξαγωγή πληροφοριών σχετικά με τα permissions από τον κώδικα του λειτουργικού συστήματος Android, χρησιμοποιώντας στατική ανάλυση. Χρησιμοποιεί call graph [56] το οποίο δημιουργείται από της κλήσεις στο API του Android. Χρησιμοποιώντας το call graph, βρίσκει τα καλέσματα (calls) σε permissions τα οποία ελέγχει και εξάγει σαν αποτέλεσμα σε ένα αρχείο Comma-Separated Values (CSV).

Το PScout χρησιμοποιήθηκε για την συλλογή των κλήσεων σε μεθόδους permissions (permission method calls) και στην συνέχεια χρησιμοποιήθηκε για την εύρεση των κλήσεων αυτών στον κώδικα των εφαρμογών, έτσι ώστε να γνωρίζουμε σε ποιο σημείο του κώδικα γίνεται κάποια κλήση σε κάποιο permission.

Κεφάλαιο 6

Υλοποίηση Συστήματος ανάλυσης Android εφαρμογών

6.1 Εισαγωγή

Στο κεφάλαιο αυτό παρουσιάζεται η μέθοδος που ακολουθήθηκε για την ανάπτυξη του συστήματος, που χρησιμοποιήθηκε το κάθε εργαλείο και πώς. Όπως αναφέρθηκε και σε προηγούμενη ενότητα κατά την δημιουργία μιας εφαρμογής, θα εξαχθεί ένα αρχείο Application Package Kit (APK) το οποίο θα μπορεί να κατεβάσει κάποιος για να μπορέσει να εγκαταστήσει την εφαρμογή. Αρχικά, επεξηγείτε η στατική ανάλυση που γίνεται στο APK αρχείο και πως εξάγονται οι πληροφορίες που έχουν ενδιαφέρον. Στην συνέχεια με βάση της πληροφορίες αυτές επεξηγούνται οι διάφορες τεχνικές που χρησιμοποιήθηκαν για την αναγνώριση κακόβουλων APK αρχείων και ο υπολογισμός ενός σκορ το οποίο καθορίζει το επίπεδο επικινδυνότητας. Τέλος περιγράφεται το πώς αυτό το εργαλείο ενσωματώθηκε σε

ένα διαδικτυακό σύστημα, έτσι ώστε να μπορεί εύκολα κάποιος να το χρησιμοποιήσει και να πάρει πληροφορίες που τον ενδιαφέρουν.

6.2 Στατική ανάλυση APK αρχείου

Σε αυτή την ενότητα θα παρουσιαστεί ο τρόπος με τον οποίο αναλύεται ένα αρχείο APK, ποιες είναι αυτές οι πληροφορίες και πως εξάγονται.

6.2.1 Εξαγωγή γενικών πληροφοριών από APK

Αρχικά και σαν πρώτο βήμα, εξάγονται πληροφορίες σχετικά με την εφαρμογή που περιέχεται στο αρχείο APK. Οι πληροφορίες αυτές εξάγονται από το AndroidManifest.xml της εφαρμογής. Το αρχείο αυτό βρίσκεται στον αρχικό φάκελο του APK αρχείου και περιέχει πληροφορίες για την εφαρμογή τις οποίες πρέπει να γνωρίζει το σύστημα του Android για να μπορεί να τρέξει την εφαρμογή. Όπως αναφέραμε το APK αρχείο είναι ένα συμπιεσμένο zip αρχείο της μορφής jar και αποσυμπιέζοντας το μπορείς να πάρεις τα αρχεία που το αποτελούν, μεταξύ τους και το AndroidManifest.xml. Το πρόβλημα όμως που παρουσιάζεται εδώ είναι ότι το manifest είναι κωδικοποιημένο σε μορφή binary xml και δεν μπορεί κάποιος να δει τα στοιχεία που περιέχει. Σε αυτό το σημείο χρειάζεται να χρησιμοποιηθεί κάποιο εργαλείο το οποίο θα μπορεί να κάνει αυτή την αποκωδικοποίηση και να παράγει το text xml.

Το εργαλείο που χρησιμοποιήθηκε για αυτή την διαδικασία ήταν το Androguard και η γλώσσα προγραμματισμού Python. Μέσω του Androguard η διαδικασία της αποκωδικοποίησης είναι πολύ απλή και εύκολα κάποιος μπορεί να την χρησιμοποιήσει μέσω της Python. Πιο κάτω περιγράφεται η διαδικασία που ακολουθείται για την εξαγωγή των πληροφορίων από το manifest αρχείο.

Αρχικά μέσω της Python διαβάζεται το αρχείο APK και με την βοήθεια του Androguard δημιουργείται ένα στιγμιότυπο ενός αντικείμενο το οποίο αναφέρεται στο συγκεκριμένο APK της εφαρμογής. Κατά την διάρκεια της αρχικοποίησης του αντικειμένου αυτού, το APK αποσυμπιέζεται με την βοήθεια της βιβλιοθήκης zipfile [10] της Python και παράγεται μια λίστα με τα αρχεία που παράχθηκαν από την αποσυμπίεση. Στην συνέχεια διατρέχοντας την λίστα αυτή, παίρνουμε το AndroidManifest.xml το οποίο διαβάζουμε με την βοήθεια της βοηθητικής κλάσης AXMLParser. Η κλάση αυτή περιέχει κάποια offsets έτσι ώστε να μπορεί κάποιος να πάρει πληροφορίες από binary xml αρχεία, αποκωδικοποιώντας τα. Στην

συνέχεια το αποκωδικοποιημένο αρχείο χρησιμοποιείται σαν κανονικό XML αρχείο με την βοήθεια των Document Objects [11] της Python. Έτσι εύκολα μπορεί κάποιος να πάρει της τιμές των διάφορων attributes που βρίσκονται στο XML. Στον πιο κάτω πίνακα φαίνονται αναλυτικά τα attributes που βρίσκονται στο AndroidManifest.xml που έχουν σημασία στην ανάλυση και εξάγονται:

Χαρακτηριστικό (Attribute)	Περιγραφή
label	Περιέχει το όνομα της εφαρμογής. Είναι αυτό που παρουσιάζεται στην οθόνη όταν εγκατασταθεί η εφαρμογή, όπως επίσης και στο Google Play Store. (Παράδειγμα: Truecaller).
package	Είναι το όνομα του βασικού πακέτου κώδικα της εφαρμογής. Το όνομα αυτό είναι το αναγνωριστικό μιας εφαρμογής (unique identifier). Αναφέρεται και σαν 'Google Play ID'. (Παράδειγμα: com.example.app).
versionName	Είναι μια συμβολοσειρά που αντιστοιχεί στην έκδοση της εφαρμογής. Αυτό καθορίζεται από τον προγραμματιστή της εφαρμογής και ο σκοπός του είναι η απλή απεικόνιση του έτσι ώστε οι χρήστες να μπορούν να ξεχωρίσουν τις διάφορες εκδώσεις μιας εφαρμογής (Παράδειγμα: 1.3.10).
versionCode	Είναι το αντίστοιχος αριθμός του versionName σε μορφή ακεραίου (integer) αριθμού. Αυτό χρησιμοποιείται από το σύστημα του Android για να ελέγχει την έκδοση μιας εφαρμογής και να βλέπει εάν ο χρήστης προσπαθεί να εγκαταστήσει κάποια παλαιότερη έκδοση. Κάθε φορά που γίνεται κάποια αναβάθμιση μιας εφαρμογής, πρέπει αυτός ο αριθμός να αλλάζει, αλλιώς το Google Play Store δεν θα επιτρέψει στον προγραμματιστή να ανεβάσει το APK αρχείο της εφαρμογής του. (Παράδειγμα 10310).
minSdkVersion	Είναι ένας ακέραιος αριθμός που καθορίζει την ελάχιστη απαιτούμενη έκδοση του Android συστήματος (Έκδοση API) που χρειάζεται για να τρέξει η εφαρμογή. (Παράδειγμα: 16)
targetSdkVersion	Είναι ένας ακέραιος αριθμός που καθορίζει την προτεινόμενη έκδοση του Android συστήματος (Έκδοση API) που χρειάζεται για να τρέξει η εφαρμογή. (Παράδειγμα: 23)
allowBackup	Παίρνει τιμές true/false. Δηλώνει αν μια εφαρμογή μπορεί να κάνει αντίγραφα ασφαλείας των δεδομένων (backup) της καθώς και

	επαναφορά (restore). Αυτό εν μέρη μπορεί να είναι χρήσιμο, όμως μπορεί κάποιες εφαρμογές να πάρουν αυτές της πληροφορίες με αποτέλεσμα να εκτεθούν πληροφορίες του χρήστη που μπορεί να περιλαμβάνουν κωδικούς πρόσβασης, αριθμούς πιστωτικών καρτών, κτλ.
debuggable	Παίρνει τιμές true/false. Δηλώνει αν μια εφαρμογή μπορεί να γίνει debug κατά την διάρκεια της εκτέλεσης της στην συσκευή του χρήστη. Αυτό είναι πολύ επικίνδυνο αφού μπορεί να φαίνονται ευαίσθητες πληροφορίες της εφαρμογής τις οποίες μπορεί να εκμεταλλευτεί κάποιος.

Πίνακας 6.1 Πληροφορίες που εξάγονται από το Androguard και χρησιμοποιούνται στο σύστημα

Επίσης κατά την διάρκεια της εξαγωγής αυτών των πληροφορίων από το APK, υπολογίζεται και το μοναδικό SHA256 hash του αρχείου, το οποίο μας βοηθά να ξεχωρίζουμε τα διάφορα APK. Αυτό επίσης βοηθά στην διαφοροποίηση των εκδόσεων ιδίων εφαρμογών, αφού το hash τους θα είναι μοναδικό. Για το hash επιλέχτηκε η τεχνική του SHA256, η οποία μέχρι στιγμής θεωρείται ασφαλής [12].

Η πιο πάνω διαδικασία περιγράφει τον τρόπο εξαγωγής των πληροφορίων από το AndroidManifest.xml με την βοήθεια του Androguard στην Python. Οι πληροφορίες αυτές αποθηκεύονται σε ένα αντικείμενο και στην συνέχεια μετατρέπονται σε json μορφή και εμφανίζονται. Καλώντας αυτό το Python script μέσω της java εύκολα μπορεί να δημιουργηθεί το αντίστοιχο αντικείμενο και να χρησιμοποιηθεί, αφού τα αποτελέσματα είναι σε json.

6.2.2 Εξαγωγή permissions από APK

Το επόμενο βήμα τις στατικής ανάλυσης είναι η εύρεση των permissions που χρησιμοποιεί η εφαρμογή. Τα permissions παίζουν σημαντικό ρόλο, αφού είναι το μέσο που η εφαρμογή θα ζητήσει πρόσβαση σε πληροφορίες και πόρους της συσκευής. Η κάθε εφαρμογή όπως επεξηγήθηκε και σε προηγούμενο κεφάλαιο θα πρέπει να έχει ορίσει τα permissions που χρησιμοποιεί στο AndroidManifest.xml αρχείο, έτσι ώστε το σύστημα να μπορεί να τα δει και με τις κατάλληλες διαδικασίες να μπορεί να τα χειριστεί. Όμως υπάρχουν περιπτώσεις, που κάποιο permissions δεν είναι δηλωμένο στο AndroidManifest.xml αλλά χρησιμοποιείται με έμμεσο τρόπο στην εφαρμογή (π.χ. από βιβλιοθήκες, κτλ.). Επίσης υπάρχουν και

περιπτώσεις όπου ένα permission μπορεί να είναι δηλωμένο αλλά να μην χρησιμοποιείται στην εφαρμογή και καλό θα ήταν τέτοιες περιπτώσεις να αποφεύγονται.

Στο παρόν στάδιο σκοπός της ανάλυσης είναι να παρθούν όλες οι πληροφορίες για τα permissions που χρησιμοποιεί ή όχι η εφαρμογή. Αυτό γίνεται με την χρήση του PermissionChecker [13] το οποίο είναι ένα εκτελέσιμο αρχείο από την εταιρία Talos srls [14] με σκοπό να δώσει πληροφορίες για τα permissions. Το PermissonChecker υλοποιήθηκε για να χρησιμοποιηθεί με το RiskInDroid [15] και μπορεί να το χρησιμοποιήσει οποιοσδήποτε για ερευνητικούς σκοπούς. Πιο κάτω παρουσιάζονται οι 4 κατηγορίες permissions που σου επιστρέφει το PermissionChecker για κάποια εφαρμογή:

Permissions τα οποία είναι δηλωμένα (Declared):

Τα permissions αυτά, όπως αναφέρεται ο τίτλος τους, είναι δηλωμένα στο AndroidManifest.xml. Ακολουθώντας παρόμοια διαδικασία όπως αναφέρθηκε πιο πριν, διαβάζονται τα permissions από το AndroidManifest.xml.

Permissions τα οποία είναι δηλωμένα και χρησιμοποιούνται (Declared and used):

Σε αυτή την κατηγορία ανήκουν τα permissions τα οποία είναι δηλωμένα στο AndroidManifest.xml και επίσης χρησιμοποιούνται στην εφαρμογή. Για τα permissions αυτά το PermissionChecker κάνει reverse engineering το APK και παράγει μια λίστα με όλα τα permissions που χρησιμοποιούνται στον κώδικα της εφαρμογής σε bytecode. Στην συνέχεια τα permissions αυτά ελέγχονται αν είναι δηλωμένα στο AndroidManifest.xml και αν είναι τότε μπαίνουν σε αυτή την κατηγορία.

Permissions τα οποία δεν είναι δηλωμένα αλλά χρησιμοποιούνται (Not declared but used):

Σε αυτή την κατηγορία ανήκουν τα permissions τα οποία χρησιμοποιούνται στην εφαρμογή αλλά δεν είναι δηλωμένα στο AndroidManifest.xml. Ακολουθείται η αντιστοιχεί διαδικασία με πιο πάνω, αλλά σε αυτή την κατηγορία μπαίνουν τα permissions τα οποία δεν είναι δηλωμένα στο AndroidManifest.xml.

Permissions τα οποία είναι δηλωμένα αλλά δεν χρησιμοποιούνται (Declared but not used):

Σε αυτή την κατηγορία ανήκουν τα permissions τα οποία είναι δηλωμένα στο AndroidManifest.xml αλλά δεν χρησιμοποιούνται καθόλου στην εφαρμογή. Όπως και πιο πάνω με την ίδια διαδικασία, εάν υπάρχει κάποιο permission το οποίο είναι δηλωμένο αλλά δεν χρησιμοποιείτε μπαίνει σε αυτή την κατηγορία.

Αφού ολοκληρωθούν οι πιο πάνω διαδικασίες, η κάθε λίστα με permissions θα προστεθεί σε ένα αντικείμενο και αυτό το αντικείμενο θα μετατραπεί σε json μορφή και θα εμφανιστεί. Καλώντας αυτό το jar μέσω της java εύκολα μπορεί να δημιουργηθεί το αντίστοιχο αντικείμενο και να χρησιμοποιηθεί, αφού τα αποτελέσματα είναι σε json.

6.2.3 Εξαγωγή πληροφορίων βιβλιοθηκών που χρησιμοποιούνται στο APK

Σε αυτό το σημείο γίνεται ο εντοπισμός των βιβλιοθηκών που χρησιμοποιούνται στο APK. Για να είναι αυτό εφικτό χρησιμοποιείται το εργαλείο LibRadar το οποίο βασίζεται σε τεχνικές μηχανικής μάθησης (machine learning). Για να γίνει χρήση του LibRadar χρειάζεται και πάλι να δημιουργήσουμε ένα Python script το οποίο θα χρησιμοποιεί το LibRadar και θα παίρνει τα αποτελέσματα. Συγκεκριμένα γίνεται χρήση του LibRadarLite (LiteRadar) το οποίο είναι πιο φορητό, αφού δεν χρειάζεται άλλα εργαλεία και ΒΔ για να τρέξει.

Αρχικά δημιουργείται αντικείμενο τύπου LibRadarLite και δίνεται σαν παράμετρος το αρχείο APK. Το αρχείο αποσυμπιέζεται και το αρχείο .dex γίνεται decompile, όπου το κάθε αρχείο (κλάση) που θα προκύψει από το decompile του .dex θα προστεθεί σε μία λίστα. Στην συνέχεια διαβάζεται το σύνολο δεδομένων (dataset) του LibRadar που περιέχει πληροφορίες με προηγούμενες αναλύσεις και βιβλιοθήκες έτσι ώστε να μπορεί να καταλάβει ποιες βιβλιοθήκες υπάρχουν και σε αυτό το APK. Ακολούθως, συγκρίνονται τα δεδομένα από το dataset με την κάθε κλάση που βρίσκεται στην λίστα, έτσι ώστε να γίνει αναγνώριση των βιβλιοθηκών που χρησιμοποιούνται σε κάθε μια από αυτές. Στην συνέχεια για κάθε βιβλιοθήκη που αναγνωρίζεται, δημιουργείται ένα αντικείμενο με τις πιο κάτω πληροφορίες:

Στοιχείο	Επεξήγηση
Library	Το όνομα της βιβλιοθήκης
Match Ratio	Πόσο τις εκατό (%) κοντά είναι με την αντίστοιχη βιβλιοθήκη
Package	Το όνομα του πακέτου της βιβλιοθήκες
Permission	Μια λίστα με τα permissions που χρησιμοποιεί η βιβλιοθήκη
Popularity	Πόσο δημοφιλής είναι η βιβλιοθήκη
Type	Τι τύπου βιβλιοθήκη είναι (π.χ. Ads)
Website	Την ιστοσελίδα της

Σχήμα 6.2 Πληροφορίες που εξάγει το LibRadar για κάθε βιβλιοθήκη

Έχοντας τις πιο πάνω πληροφορίες για κάθε βιβλιοθήκη και αφού δημιουργηθεί το αντικείμενο με τα αντίστοιχα δεδομένα, τότε αποθηκεύεται σε μία λίστα. Όταν τελειώσει η ανάλυση των βιβλιοθηκών, η λίστα αυτή μετατρέπεται σε JavaScript Object Notation (JSON) μορφή και εμφανίζεται, έχοντας όλα τα στοιχεία των βιβλιοθηκών του συγκεκριμένου APK αρχείου. Καλώντας αυτό το Python script μέσω της java εύκολα μπορεί να δημιουργηθεί το αντίστοιχο αντικείμενο και να χρησιμοποιηθεί, αφού τα αποτελέσματα είναι σε JSON.

6.2.4 Εύρεση μεθόδων που καλούνται και αναφέρονται σε permissions στο APK

Σε αυτό το σημείο θα αναλυθεί η εύρεση μεθόδων που καλούν permissions (permissions API calls) σε μια εφαρμογή. Όπως αναφέρθηκε και σε προηγούμενη ενότητα για να μπορεί κανείς να χρησιμοποιήσει κάποιο permission θα πρέπει να το έχει δηλώσει στο Android Manifest.xml. Επίσης υπάρχουν περιπτώσεις, όπου κάποιο permissions δεν είναι δηλωμένο αλλά παρά ταύτα χρησιμοποιείται με κάποιο έμμεσο τρόπο. Οι συγκεκριμένες κλήσεις σε permission είναι ανάγκη να βρεθούν, έτσι ώστε να μπορεί κάποιος να δει σε πιο σημείο ακριβώς του κώδικα γίνεται χρήση κάποιου συγκεκριμένου permission. Με κάτι τέτοιο θα μπορούν και οι ίδιοι οι προγραμματιστές κάποιας εφαρμογής να αλλάξουν κάτι που αφορά τις κλήσεις σε permissions, αφού θα γνωρίζουν πού ακριβώς γίνεται η κάθε κλήση.

Για να μπορούμε να κάνουμε την ανάλυση αυτή χρειάζονται πληροφορίες που είναι ήδη διαθέσιμες από προηγούμενες αναλύσεις του εργαλείου. Χρειάζονται τα permissions τα οποία είναι δηλωμένα και χρησιμοποιούνται (Declared and used), καθώς επίσης και αυτά που δεν είναι δηλωμένα αλλά χρησιμοποιούνται (Not declared but used), για να μπορέσουν να εντοπιστούν οι κλήσεις που τα αφορούν. Ακόμα χρειάζονται το εργαλείο Androguard που χρησιμοποιήθηκε και πιο πριν και το dataset που παράγει το εργαλείο PScout [16]. Το PScout είναι ένα εργαλείο το οποίο εξάγει όλες τις κλήσεις στο API των permissions του Android για τη συγκεκριμένη έκδοση του. Δηλαδή, δημιουργεί ένα dataset με τις μεθόδους που περιέχει το κάθε permission και το πώς καλείται αυτή η μέθοδος. Με τη βοήθεια του Androguard, το οποίο έχει συναρτήσεις οι οποίες αναζητούν μεθόδους στον κώδικα του APK, μπορούμε να ψάξουμε και να εντοπίσουμε από ποιο σημείο το κώδικα καλούνται. Πιο κάτω αναλύεται η διαδικασία αυτή:

Αρχικά θα πρέπει να εκτελεστεί το PermissionChecker για να πάρουμε τα permissions που χρησιμοποιούνται στην εφαρμογή (δηλωμένα ή μη δηλωμένα), έτσι ώστε να μπορούμε να τα ψάξουμε στον κώδικα. Επίσης θα πρέπει να έχουμε και το dataset που παράγει το PScout με

όλα τα πιθανά καλέσματα μεθόδων των permissions. Συγκεκριμένα χρησιμοποιούνται τα datasets για τις εκδόσεις API 9, 10, 14-19, 21-22 οι οποίες περιέχουν κάποιες αλλαγές σε κάποιες μεθόδους, γι' αυτό χρειάζεται να τις χρησιμοποιούμαι όλες.

Αφού πάρουμε τα permissions από το PermissionChecker θα πρέπει να τα δώσουμε ως είδοσο στο Python script μαζί με το αρχείο APK. Στην συνέχεια αφού γίνει αποσυμπίεση στο APK και αποκωδικοποιηθεί το AndroidManifest.xml, θα μπορούμε να δούμε την έκδοση του API που χρησιμοποιεί για να επιλεχτεί το κατάλληλο dataset. Σε αυτό το σημείο θα πρέπει το αρχείο .dex το οποίο έχει παραχθεί από την αποσυμπίεση του APK, να απο-μεταγλωττιστεί (decompile) για να μπορούμε να ψάξουμε στον κώδικα. Αυτό γίνεται με την βοήθεια του Androguard το οποίο περιέχει τα κατάλληλα offsets. Αφού ολοκληρωθεί η διαδικασία του decompile, θα ξεκινήσει η ανάλυση για την αναζήτηση των κλήσεων μεθόδων που αφορούν permissions. Σε αυτό το σημείο θα πρέπει για κάθε permission που βρήκαμε από τα προηγούμενα βήματα, να βρούμε τις μεθόδους που το αφορούν μέσω του dataset που παράχθηκε από το PScout. Για κάθε μέθοδο του κάθε permission θα καλείται η συνάρτηση του Androguard *search_method(name, descriptor)* η οποία αναζητά την συνάρτηση με βάση το όνομα της (name) και των τύπων των παραμέτρων που δέχεται (descriptor). Ένα παράδειγμα είναι:

Στο dataset υπάρχει το πιο κάτω:
'Landroid/net/wifi/WifiManager;-setWifiApConfiguration-(Landroid/net/wifi/WifiConfiguration;)Z' : ['android.permission.CHANGE_WIFI_STATE']

Αυτό υποδηλώνει ότι το permission 'CHANGE_WIFI_STATE', στην κλάση 'android/net/wifi/WifiManager' έχει μια συνάρτηση η οποία ονομάζεται 'setWifiApConfiguration' και παίρνει παράμετρο τύπου 'android/net/wifi/WifiConfiguration'. Άρα για να χρησιμοποιήσομε την *search_method* του εργαλείου Androguard θα πρέπει να δοθεί σαν παράμετρος για το name το setWifiApConfiguration και σαν descriptor το '(Landroid/net/wifi/ WifiConfiguration;)Z'.

Καλώντας την συνάρτηση *search_method* θα επιστραφεί ένα αντικείμενο τύπου Path, το οποίο χρησιμοποιεί το Androguard. Για να δούμε τις πληροφορίες του Path, θα πρέπει να κληθεί η συνάρτηση Get_Path δίνοντας ως παράμετρο το αντικείμενο Path. Στη συνέχεια θα επιστραφεί μια συμβολοσειρά της μορφής:
Caller Function (package -> function) -> Permission Function (package -> function)

όπου Caller Function είναι η συνάρτηση της εφαρμογής (κλάση -> όνομα συνάρτησης) και ποια συνάρτηση του permission καλεί. Για εύκολη χρήση της πιο πάνω διαδικασίας, υλοποιήθηκε η συνάρτηση που φαίνεται στο Σχήμα 6.1, η οποία κάνει την πιο πάνω διαδικασία. Με αυτό τον τρόπο μπορούμε να εντοπίσουμε τις κλήσεις σε συναρτήσεις που αφορούν permission. Το κάθε αποτέλεσμα το βάζουμε σε μια λίστα και στην συνέχεια μετατρέπεται σε JSON μορφή και εμφανίζεται στο χρήστη του εργαλείου. Καλώντας αυτό το Python script μέσω της Java εύκολα μπορεί να δημιουργηθεί το αντίστοιχο αντικείμενο και να χρησιμοποιηθεί, αφού τα αποτελέσματα είναι σε JSON μορφή.

```
def getFunctionPermissionCallsByPermissions(path,permissionlist):
      a = apk.APK(path)
      d=dvm.DalvikVMFormat(a.get_dex())
      vmx=analysis.VMAnalysis(d)
      returnList=[]
      plist=set(permissionlist.split(","))
      #Get all permission call list
      k=vmx.tainted_packages.get_permissions([])
      #for every permission call in the list
      for t in k:
            #if permission does not exist in app, continue;
            if str(t) not in plist:
                  continue
            #if it exists find the permission calls
            p=analysis.my_get_Paths(d,k[str(t)],str(t))
            for y in p:
                  returnList.append(y.__dict__)

      return json.dumps((returnList),sort_keys=False, indent=4)
```

Σχήμα 6.1 Κώδικας συνάρτησης για εύρεση καλεσμάτων σε μεθόδους που αφορούν permissions

6.3 Ανάλυση permissions για το αν είναι επικίνδυνο το APK

Σε αυτό το σημείο επεξηγείται ο εντοπισμός επικίνδυνων εφαρμογών οι οποίες χρησιμοποιούν συνδυασμό permissions τα οποία χρησιμοποιούν κάποια malware. Αυτή η λειτουργία έχει σκοπό να κατηγοριοποιήσει μια εφαρμογή αν είναι επικίνδυνη ή όχι με βάση τα permissions που χρησιμοποιεί. Θα πρέπει να βασιστούμε και πάλι στις προηγούμενες μεθόδους για να εξάγουμε τα permissions που χρησιμοποιεί κάποια εφαρμογή έτσι ώστε να μπορούμε να τα αναλύσουμε και να δούμε κατά πόσο την καθιστά επικίνδυνη. Υπάρχουν διάφορες προσεγγίσεις για έλεγχο επικινδυνότητας των εφαρμογών, αλλά σε αυτή την

περίπτωση θα επικεντρωθούμε στα permissions. Χρησιμοποιήθηκε το dataset του Christian Urcuqui [17], οποίος κάνοντας την δική του ανάλυση [18] για την εύρεση malware εφαρμογών με βάση τα permissions με την χρήση μηχανικές μάθησης, κατάληξε στο πιο πάνω dataset. Το dataset περιέχει 303 στοιχεία που αντιπροσωπεύουν εφαρμογές και έχει σαν attributes όλα τα permissions και ανάλογα αν το χρησιμοποιεί μια εφαρμογή, η τιμή του κάθε attribute είναι 0 ή 1. Επίσης υπάρχει και ένα attribute το 'type' το οποίο εάν η τιμή του είναι 1 σημαίνει ότι η εφαρμογή είναι malware, αν είναι 0 σημαίνει ότι δεν είναι.

Αρχικά, χρησιμοποιήθηκε το συγκεκριμένο dataset για την εύρεση malware χρησιμοποιώντας τα permissions που χρησιμοποιούσε η εφαρμογή. Κατά την διάρκεια όμως της ανάλυσης εφαρμογών οι οποίες ήταν γνωστό ότι ήταν malware ή όχι, παρατηρήθηκε ότι το αποτέλεσμα δεν ήταν το ίδιο με αυτό που έπρεπε. Παρατηρήθηκε ότι κάποιες εφαρμογές που δεν ήταν malware αλλά με το συγκεκριμένο dataset, φαίνονταν ότι είναι malware. Αυτό διότι χρησιμοποιούσαν αρκετά permissions τα οποία μπορεί να είναι και dangerous. Κάποια παραδείγματα αυτών των εφαρμογών είναι το Facebook, το Messenger, το Instagram, και άλλα. Αυτές οι εφαρμογές μπορεί να ζητήσουν πρόσβαση στην τοποθεσία του χρήστη, στα μηνύματα της συσκευής, στις επαφές της συσκευής, κτλ. Αυτό έχει ως αποτέλεσμα με το πιο πάνω dataset να κατηγοριοποιούνται σαν malware, ενώ στην πραγματικότητα δεν είναι. Με βάση αυτά, φάνηκε πως δεν επαρκούν τα permissions για να χαρακτηριστεί μια εφαρμογή ως malware. Γι' αυτό λοιπόν στην περίπτωση μας θα κατηγοριοποιήσουμε τις εφαρμογές σε επικίνδυνες και μη επικίνδυνες, βασισμένοι στο πιο πάνω dataset. Με τον όρο επικίνδυνο, δείχνουμε ότι κάτι μπορεί να κάνει κάποια ζημία, αλλά επίσης μπορεί και όχι.

Αρχικά για την ανάλυση χρησιμοποιήθηκε η βιβλιοθήκη Weka [19] η οποία σου επιτρέπει να χρησιμοποιήσεις διάφορες τεχνικές και αλγορίθμους του μηχανικής μάθησης και είναι υλοποιημένο σε Java.

Βασισμένοι στο dataset αυτό και με την χρήση του Weka, αναλύθηκαν οι προτεινόμενοι classifiers που χρησιμοποιήθηκαν και από τον Christian Urcuqui, στην ερευνά του. Πιο κάτω περιγράφονται οι classifiers που δοκιμάστηκαν και στην συνέχεια φαίνεται ένας πίνακας με τα αποτελέσματα, κάνοντας train με το dataset και ελέγχοντας με τον κάθε classifier:

<u>Naive Bayes:</u>

Ο classifier Naive Bayes χρησιμοποιείται όταν έχουμε κάποιες καταστάσεις και θέλουμε να εξάγουμε την πιθανότητα να συμβεί ένα γεγονός χρησιμοποιώντας στατιστικές μεθόδους.

Ένα παράδειγμα για την κατηγοριοποίηση Naive Bayes είναι το φίλτρο ανεπιθύμητης ηλεκτρονικής αλληλογραφίας (spam filter), από όπου το πρόγραμμα αλληλογραφίας διαχωρίζει τα email σε spam ή όχι ανάλογα με τις λέξεις που περιέχονται σε αυτό.

Bagging:

Ο classifier Bagging, βασίζεται στους αλγόριθμους Bootstrap και Aggregation [57], και έχει ως βασική λειτουργία την δειγματοληψία με επανένταξη (sampling with replacement), να δημιουργεί classifier για κάθε δείγμα δεδομένων και το κάθε δείγμα από αυτά να έχει πιθανότητα επιλογής $[1- (1/n)]^n$.

K-Nearest Neighbor:

Ο classifier K-Nearest Neighbor [58], κάνει την παραδοχή ότι τα διάφορα παραδείγματα μπορεί να αναπαρασταθούν ως σημεία σε κάποιον n-διάστατο Ευκλείδειο χώρο Rn όπου n ο αριθμός των χαρακτηριστικών (ανεξάρτητων μεταβλητών).
Κάθε νέα περίπτωση τοποθετείται στο χώρο αυτό ως νέο σημείο και η τιμή του προσδιορίζεται με βάση το χαρακτηρισμό των k γειτονικών σημείων. Οι κοντινότεροι γείτονες μιας περίπτωσης υπολογίζονται με βάση την Ευκλείδεια απόστασή τους.

Stochastic Gradient Descent (SGD):

Ο classifier SGD [59], ανατρέχει όλο το dataset, και κάθε φορά που συναντά ένα δείγμα εκπαίδευσης, ανανεώνονται οι παράμετροι των συνοπτικών βαρών σε σχέση με το σφάλμα του συγκεκριμένου δείγματος. Δηλαδή, ξεκινά να συγκλίνει με μικρά βήματα άμεσα και για κάθε ζεύγος εκπαίδευσης χωριστά. Συχνά, ο classifier SGD συγκλίνει γρηγορά αλλά υπάρχει περίπτωση να μην συγκλίνει σε κάποια τελική τιμή και να περιβάλλεται γύρω από το ελάχιστο.

Locally Weighted Learning (LWL):

Ο LWL (Locally Weighted Learning) [60] είναι ένας τεμπέλης (lazy) αλγόριθμος για την τοπικά σταθμισμένη εκμάθηση και αναθέτει βάρη χρησιμοποιώντας μια μέθοδο βασισμένη σε στιγμές και δημιουργεί έναν ταξινομητή από τις σταθμισμένες περιπτώσεις.

Αλγόριθμος	Precision		Recall		F-Measure		Accuracy
	0	1	0	1	0	1	
Naive Bayes	0.881	0.926	0.930	0.874	0.905	0.899	90.201 %
Bagging	0.946	0.892	0.884	0.950	0.914	0.920	91.7085 %
K-Nearest Neighbor	0.922	0.943	0.945	0.920	0.933	0.931	93.2161 %
SGD	0.938	0.921	0.920	0.940	0.929	0.930	92.9648 %
LWL	0.951	0.888	0.879	0.955	0.914	0.920	91.7085 %

Πίνακας 6.3 Αποτελέσματα του κάθε classifier

Με βάση τα πιο πάνω αποτελέσματα καταλήγουμε στο ότι ο K-Nearest Neighbor είναι ο καταλληλότερος classifier για την περίπτωσή μας. Οι δοκιμές αυτές έγιναν μέσω του Weka GUI και αποθηκευτικέ το μοντέλο του K-Nearest Neighbor για εύκολη χρήση.

Για να χρησιμοποιήσουμε το πιο πάνω μοντέλο για να κατηγοριοποιήσουμε κάποιο APK, θα πρέπει όπως αναφέρθηκε να εξαχθούν πρώτα τα permissions που χρησιμοποιεί. Γι' αυτό μέσω της Java, εκτελούμε το PermissionChecker έτσι ώστε να πάρουμε τα permissions που χρησιμοποιούνται (δηλωμένα ή μη δηλωμένα) και τα βάζουμε σε μια λίστα. Στην συνέχεια, δημιουργούμε ένα instance και βάζουμε τις ανάλογες τιμές στο κάθε attribute, που είναι το κάθε permission. Δηλαδή, εάν χρησιμοποιείται το permission κάνουμε την τιμή 1, αλλιώς 0. Στην συνέχεια με φορτώνουμε το μοντέλο που δημιουργήσαμε από το Weka GUI, και δημιουργούμε τον classifier. Στην συνέχεια δίνουμε σαν παράμετρο το instance στην μέθοδο classifyInstance για να γίνει η κατηγοριοποίηση του APK με τα συγκεκριμένα permission. Και τέλος, όταν ολοκληρωθεί η διαδικασία, θα επιστραφεί 1 ή 0, ανάλογα αν είναι επικίνδυνο το APK ή όχι.

6.4 Ανάλυση με το VirusTotal API για εντοπισμό κακόβουλων APK

Για τον εντοπισμό κακόβουλων APK, έγινε χρήση του API του εργαλείου VirusTotal [67]. Το VirusTotal είναι ένα διαδικτυακό εργαλείο που σου επιτρέπει να ανεβάσεις οποιοδήποτε αρχείο και να το σαρώσει με διάφορα επώνυμα Antivirus και επιστρέφει τα αποτελέσματα ξεχωριστά για το καθένα. Αυτό δίνει την δυνατότητα να κρίνει ο ίδιος αν κάποιο αρχείο έχει κάποιο ιό, καθώς επίσης εμφανίζονται και διάφορα πιθανά comments από άλλους χρήστες, αν το ίδιο αρχείο έχει σαρωθεί ξανά από το εργαλείο.

Επίσης το VirusTotal, παρέχει ένα REST API το οποίο μπορεί οποιοσδήποτε να το χρησιμοποιήσει και να στείλει προγραμματιστικά το αρχείο για σάρωση ή να ζητήσει αποτελέσματα για κάποιο αρχείο. Αυτό διευκολύνει πολύ την ανάλυση πολλών αρχείων μαζί, αφού μπορεί κάποιος να αυτοματοποιήσει την λειτουργία αυτή.

Στην περίπτωσή μας, χρησιμοποιήθηκε το VirusTotal API σαν μια επιπρόσθετη ανάλυση, αφού όπως αναφέρθηκε πιο πριν, δεν είναι τόσο καλή ταχτική να ψάχνεις για malware αποκλειστικά από τα permissions. Γι' αυτό τον λόγο, όταν μια εφαρμογή κατηγοριοποιηθεί σαν επικίνδυνη, ο χρήστης θα μπορεί να δει μέσα από το VirusTotal αν πραγματικά είναι κακόβουλο ή όχι.

Για την χρήση του VirusTotal REST API έγινε χρήση των βιβλιοθηκών Apache httpclient [65] και Apache httpmime [66]. Η ανάλυση εδώ γίνεται παράλληλα με τις υπόλοιπες, αφού στην ουσία δεν κάνει το σύστημα την ανάλυση αλλά το VirusTotal. Το σύστημα για την ανάλυση κάποιου αρχείου, καλεί το ανάλογο Uniform Resource Identifier (URI) και δίνοντας το αρχείο σαν παράμετρο στο http μήνυμα. Στην συνέχεια θα αποσταλεί κάποιος σύνδεσμος από το VirusTotal όπου θα βρίσκονται τα αποτελέσματα του αρχείου αυτού. Τα αποτελέσματα μπορεί να μην είναι άμεσα διαθέσιμα καθώς ακόμα μπορεί να σαρώνεται το αρχείο.

Επίσης εάν πρέπει να εμφανιστούν οι πληροφορίες από το VirusTotal για κάποιο APK, τότε θα γίνει κάλεσμα άλλου URI (μέσω του SHA256 hash) με αυτή την λειτουργία θα πάρουμε τα αποτελέσματα για το APK αρχείο.

6.5 Ανάλυση βιβλιοθηκών για την εύρεση trackers

Σε αυτό το σημείο θα επανέλθουν οι βιβλιοθήκες, όμως τώρα σκοπός μας είναι να εντοπίσουμε διάφορους trackers που χρησιμοποιούνται στην εφαρμογή. Ένας trackers, είναι μια βιβλιοθήκη που ονομάζεται έτσι διότι συνίθως παρακολουθεί την εφαρμογή και μπορεί να συλλέγει διάφορές πληροφορίες. Για παράδειγμα, εάν είναι βιβλιοθήκη για διαφημίσεις, μπορεί να συλλέγει διάφορες πληροφορίες για το πώς ο χρήστης χρησιμοποιεί την εφαρμογή και να του βγάλει κάποιες διάφορες σχετικές διαφημίσεις οι οποίες θα είναι πιο κοντά στα ενδιαφέροντα του για να του τραβήξει την προσοχή. Τα δεδομένα αυτά όμως που συλλέγονται, δεν είναι εις γνώση του χρήστη.
Η λειτουργία αυτή θα παρουσιάζει τους trackers που βρέθηκαν στις βιβλιοθήκες που εντόπισε το LibRadar σε προηγούμενα βήματα. Η λίστα με τους trackers είναι αποθηκευμένη

στην βάση δεδομένων και προέρχονται από το exodus [20] το οποίο επεξηγήθηκε σε προηγούμενο κεφάλαιο.

Η διαδικασία σε αυτή την περίπτωση, είναι η εύρεση των βιβλιοθηκών με την χρήση του LibRadar και στην συνέχεια η εύρεση trackers μέσα από αυτές τις βιβλιοθήκες. Η λειτουργία αυτή είναι καθαρά υλοποιημένη σε java καλώντας το Python script που χρησιμοποιεί το LibRadar όπως εξηγήθηκε και πιο πάνω.

6.6 Υπολογισμός σκορ APK βάσει των αποτελεσμάτων

Για κάθε APK που αναλύεται από το σύστημα, υπολογίζεται ένα σκορ με βάση όλες τις πιο πάνω λειτουργίες και τα αποτελέσματα, έτσι ώστε να υπάρχει κάτι που να το κάνει να ξεχωρίζει και να έχουμε κάποιο τρόπο σύγκρισης.

Ο υπολογισμός του σκορ γίνεται με της παρακάτω μεταβλητές:

Όνομα μεταβλητής	Σκορ
Dangerous permission το οποίο είναι δηλωμένο αλλά δεν χρησιμοποιείται	3
Signature ή SystemOrSignature permission το οποίο είναι δηλωμένο αλλά δεν χρησιμοποιείται	1.5
Dangerous permission το οποίο είναι δηλωμένο και χρησιμοποιείται	10
Signature ή SystemOrSignature permission το οποίο είναι δηλωμένο και χρησιμοποιείται	5
Dangerous permission το οποίο δεν είναι δηλωμένο και χρησιμοποιείται	15
Signature ή SystemOrSignature permission το οποίο δεν είναι δηλωμένο και χρησιμοποιείται	3.75
Normal permission το οποίο δεν είναι δηλωμένο και χρησιμοποιείται	7.5
Το APK είναι debuggable	8
Το APK επιτρέπει full backup των δεδομένων της εφαρμογής.	3
Το APK κατηγοριοποιήθηκε σαν dangerous	30
Μέγιστο σκορ	100

Πίνακας 6.4 Παράμετροι υπολογισμού σκορ

Τα πιο πάνω λοιπόν καθορίζουν το πώς υπολογίζουμε το σκορ μιας εφαρμογής με βάση το μέγιστο σκορ. Τα πιο πάνω βρίσκονται στην ΒΔ του συστήματος και μπορεί εύκολα να αλλαχτούν. Με την αντιστοίχιση του σκορ με κάποια εφαρμογή που μας δείχνει το πόσο επικίνδυνη είναι. Αυτή η προσέγγιση παρουσιάζει συγκριτικές πληροφορίες, δηλαδή με το σκορ της κάθε εφαρμογής παρουσιάζεται με τέτοιο τρόπο έτσι ώστε να μπορεί να συγκριθεί εύκολα με άλλες εφαρμογές. Με βάση τα πιο πάνω μπορεί να οριστεί ένα όριο (threshold) το

οποίο να καθορίζει κατά πόσο πολύ επικίνδυνη είναι μια εφαρμογή. Με αυτό τον τρόπο μπορεί να υπολογιστεί και να προσδιοριστεί ευκολά και με κατανόηση το σκορ κάθε εφαρμογής με βάση τα permissions.

Επίσης κατά την ολοκλήρωση της ανάλυσης όταν θα εμφανιστούν τα αποτελέσματα, υπάρχουν 4 ΄επίπεδα ρίσκου΄ τα οποία είναι:

Επίπεδο	Σκορ	Χρώμα μπάρας
Χωρίς ρίσκο	0-25 %	Πράσινο
Λίγο ρίσκο (ελάχιστα permissions)	25-40 %	Κίτρινο
Έχουν κάποιο ρίσκο	40-70 %	Πορτοκαλί
Πρέπει να προσέχουμε το ρίσκο	70-100%	Κόκκινο

Πίνακας 6.5 Επίπεδα ρίσκου

Με τα πιο πάνω μπορούμε, αφού ολοκληρωθεί η ανάλυση, να εντάξουμε την εφαρμογή σε κάποιο από αυτές της κατηγορίες με βάση το σκορ τους. Το σκορ όπως επίσης και το επίπεδο ρίσκου θα εμφανίζονται στην οθόνη με διαφορετικό χρώμα μπάρας.

Αρχικά όταν το σκορ μιας εφαρμογής είναι μικρότερο από 25% τότε θεωρούμε ότι η χρήση της εφαρμογής δεν έχει κάποιο ρίσκο. Αυτό εάν μια εφαρμογή έχει λιγότερο από 25%, σημαίνει ότι έχει το πολύ 2 dangerous permissions ή άλλα permissions τα οποία δεν θεωρούνται επικίνδυνα.

Όταν το σκορ κυμαίνεται από 25 μέχρι 40%, τότε θεωρούμε ότι η εφαρμογή έχει μικρό ρίσκο, αφού για να έχει τέτοιο σκορ, σημαίνει ότι έχει ελάχιστα permissions τα οποία είναι επικίνδυνα και έχει επίσης και κάποια άλλα permissions.

Οι εφαρμογές που θεωρούνται ότι έχουν κάποιο ρίσκο είναι αυτές που έχουν σκορ μεγαλύτερο από 40% ρίσκο. Σε αυτό το επίπεδο βρίσκονται οι εφαρμογές οι οποίες χρησιμοποιούν σημαντικό αριθμό dangerous permissions. Επίσης σε αυτή την κατηγορία μπορεί να εντάσσονται και εφαρμογές που κρίθηκαν ως επικίνδυνες με την χρήση της μηχανικής μάθησης, αφού εάν κρίθηκε σαν επικίνδυνη το σκορ αυξάνεται κατά 30%.

Τέλος, το πιο επικίνδυνο επίπεδο είναι οι εφαρμογές που έχουν μεγαλύτερο σκορ από 70%. Οι εφαρμογές αυτές θεωρούνται ότι έχουν υψηλό ρίσκο και θα πρέπει να είμαστε προσεκτικοί χρησιμοποιώντας τις. Το σκορ αυτό, πολύ πιθανών να προέρχεται από την

χρήση αρκετών dangerous permissions ή από την χρήση permissions τα οποία δεν είναι δηλωμένα. Επίσης εφαρμογές που το σκορ τους ξεπερνά το 70%, είναι πιθανών να κρίθηκαν και ως επικίνδυνες από την χρήση μηχανικής μαθήσεις για την αντίστοιχη κατηγοριοποίηση.

6.7 Υλοποίηση Διαδικτυακής Σελίδας για χρήση του συστήματος

Αρχικά το εργαλείο το οποίο κάνει τις λειτουργίες που αναφέρθηκαν πιο πάνω υλοποιήθηκε σε ξεχωριστά κομμάτια και ενσωματώθηκε χρησιμοποιώντας την γλώσσα προγραμματισμού Java. Όμως κρίθηκε απαραίτητο να υπάρχει μια γραφική διπροσωπία έτσι ώστε να μπορεί κάποιος να ανεβάσει κάποιες εφαρμογές για ανάλυση και να δει τα αποτελέσματα. Επίσης, το να υπάρχει κάποιος εύκολος και εύχρηστος τρόπος παρουσίασης των αποτελεσμάτων, όπως επίσης και αναζήτησης εφαρμογών οι οποίες ήδη αναλύθηκαν. Γι' αυτό τον λόγο το εργαλείο ενσωματώθηκε σε μια διαδικτυακή εφαρμογή έτσι ώστε να είναι εύκολα προσβάσημο και να μπορεί κάποιος εύκολα να πραγματοποιήσει κάποια ανάλυση ή να δει πληροφορίες μιας εφαρμογής.

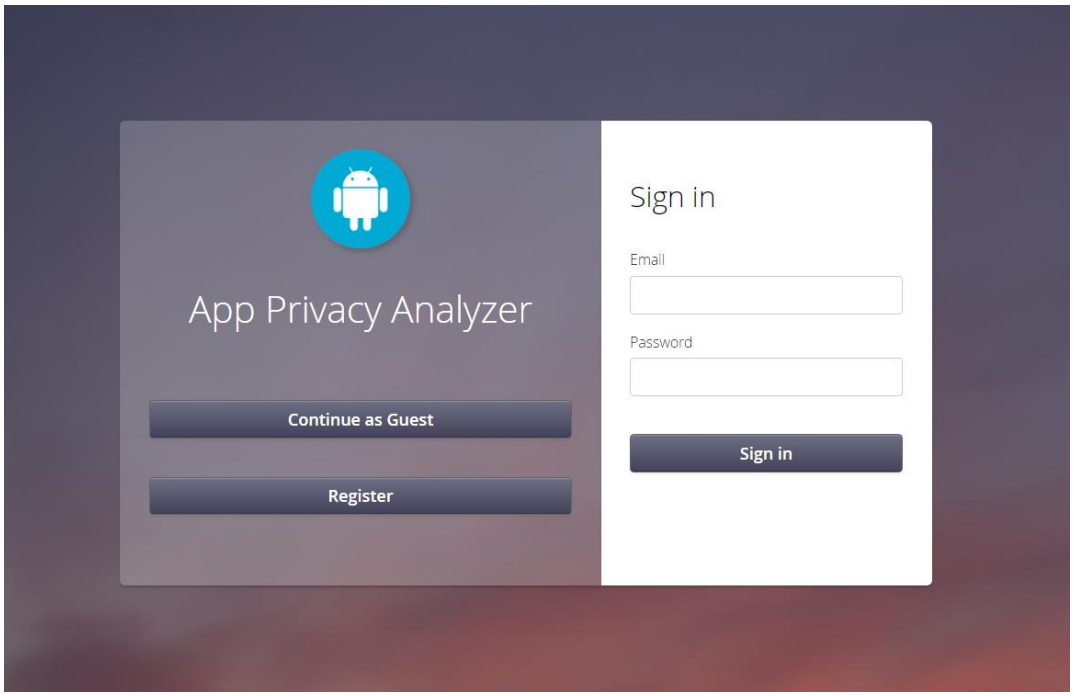
Η διαδικτυακή εφαρμογή αναπτύχθηκε χρησιμοποιώντας τα frameworks Spring και Vaadin. Με την βοήθεια του spring υλοποιήθηκε το backend της εφαρμογής το οποίο περιέχει την διαδικασία ανάλυσης μιας εφαρμογής, όπως επίσης και την σύνδεση με την ΒΔ. Το Vaadin χρησιμοποιείτε για το frontend της εφαρμογής, δηλαδή για το πως θα εμφανίζονται στην οθόνη οι διάφορες πληροφορίες και το πως θα αλληλοεπιδρά ο χρήστης με το σύστημα.

Για την υλοποίηση της γραφικής διαπροσωπίας της εφαρμογής δημιουργήθηκαν διαφορετικά views. Το κάθε view αντιπροσωπεύει και μια σελίδα όπου στην οποία υπάρχει και μια λειτουργία. Σε όλες τις σελίδες υπάρχει ένα navigation μενού, έτσι ώστε να μπορεί κάποιος να μεταβεί σε κάποια άλλη λειτουργία της εφαρμογής.

Οι χρήστες μπορεί να δημιουργήσουν λογαριασμό και να συνδεθούν για να χρησιμοποιήσουν το σύστημα ή να παραβλέψουν την σύνδεση και να προχωρήσουν στη χρήση της εφαρμογής σαν επισκέπτες (guests). Δεν υπάρχει κάποια διαφορά αν ο χρήστης συνδεθεί ή όχι.

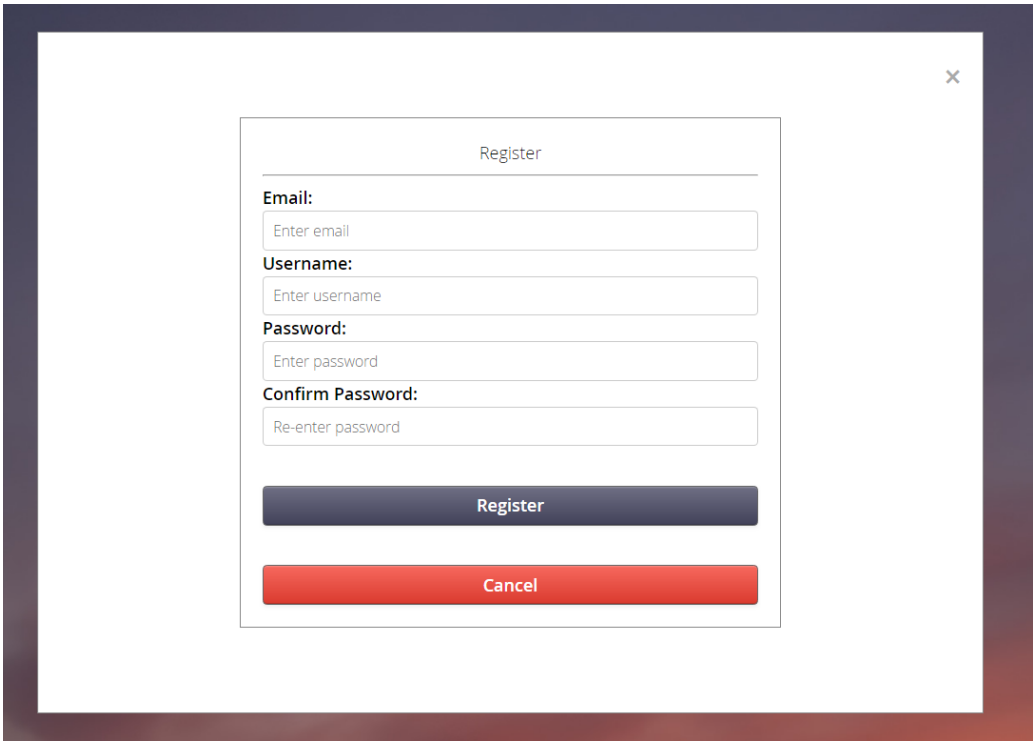
Πιο κάτω παρουσιάζονται screenshots από το σύστημα, περιγράφοντας και την διαδικασία ανάλυσης μιας εφαρμογής:

1. Αρχικά η σελίδα (Σχήμα 6.2) που εμφανίζεται επιτρέπει στον χρήστη να συνδεθεί στο σύστημα ή να δημιουργήσει λογαριασμό.



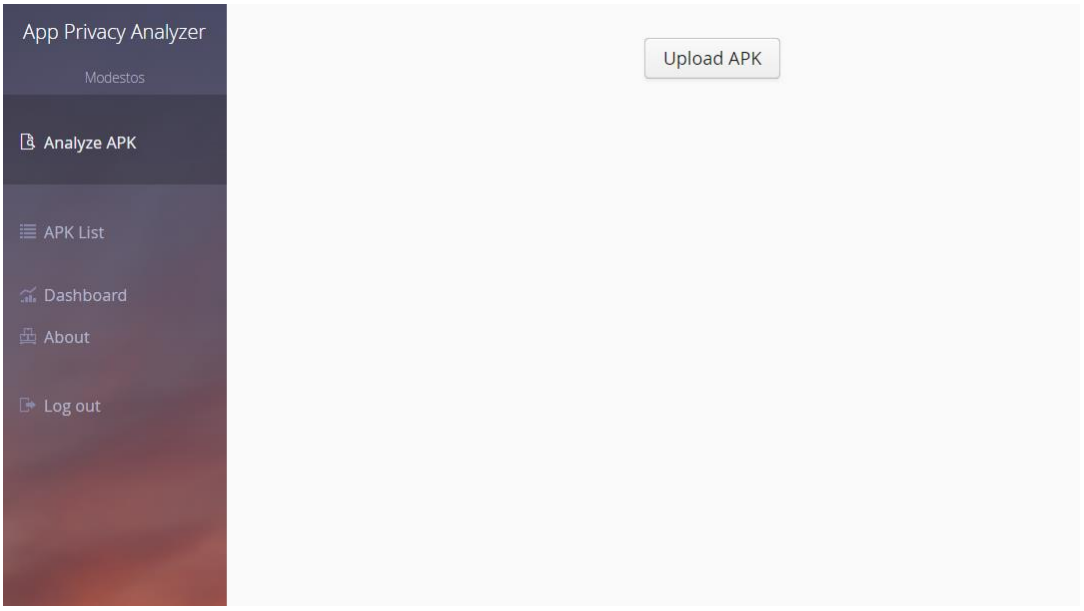
Σχήμα 6.2 Αρχική σελίδα για σύνδεση στο σύστημα

2. Πατώντας το κουμπί Register, εμφανίζεται ένα κουτάκι (Σχήμα 6.3) που επιτρέπει στον χρήστη να δημιουργήσει λογαριασμό για να μπορέσει να συνδεθεί στο σύστημα.



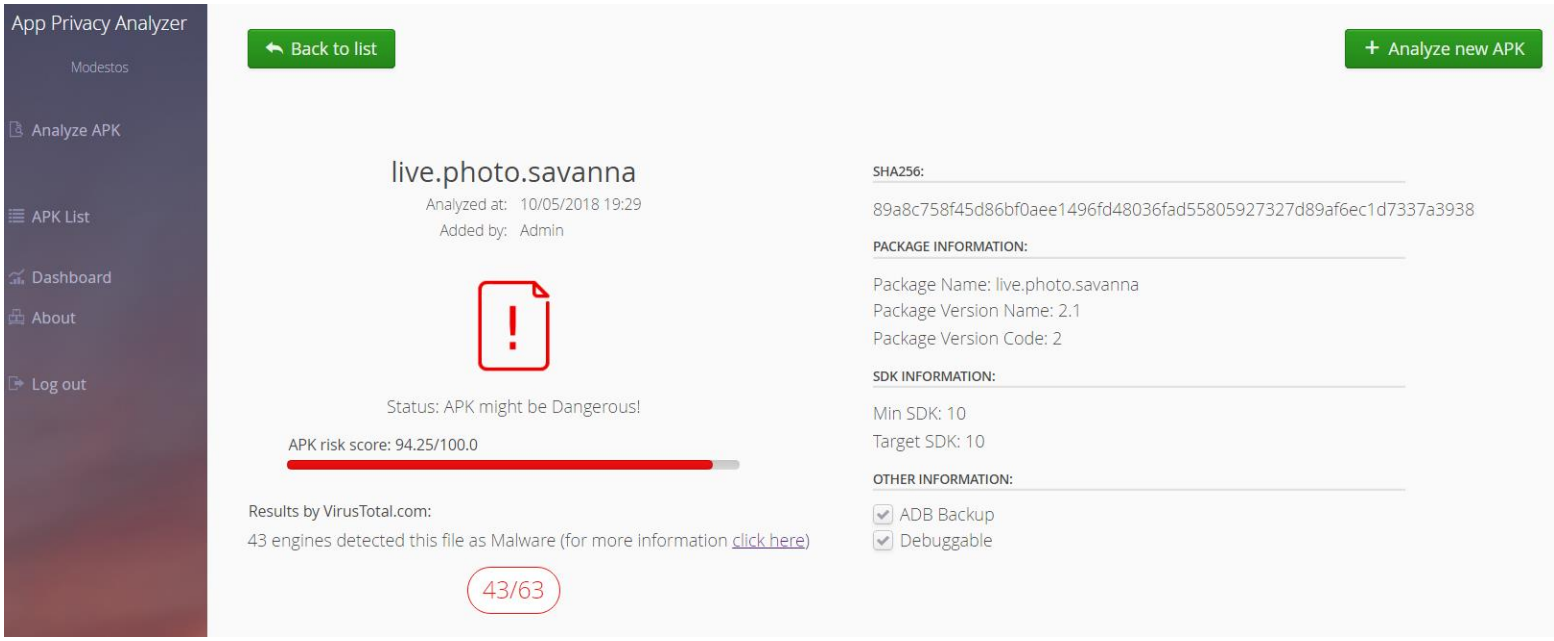
Σχήμα 6.3 Αρχική σελίδα με το κουτάκι για την δημιουργία λογαριασμού

3. Με το που θα συνδεθεί κάποιος στο σύστημα ή εισέλθει σαν guest θα του εμφανιστεί η σελίδα (Σχήμα 6.4) για να ανεβάσει κάποιο APK μιας εφαρμογής.



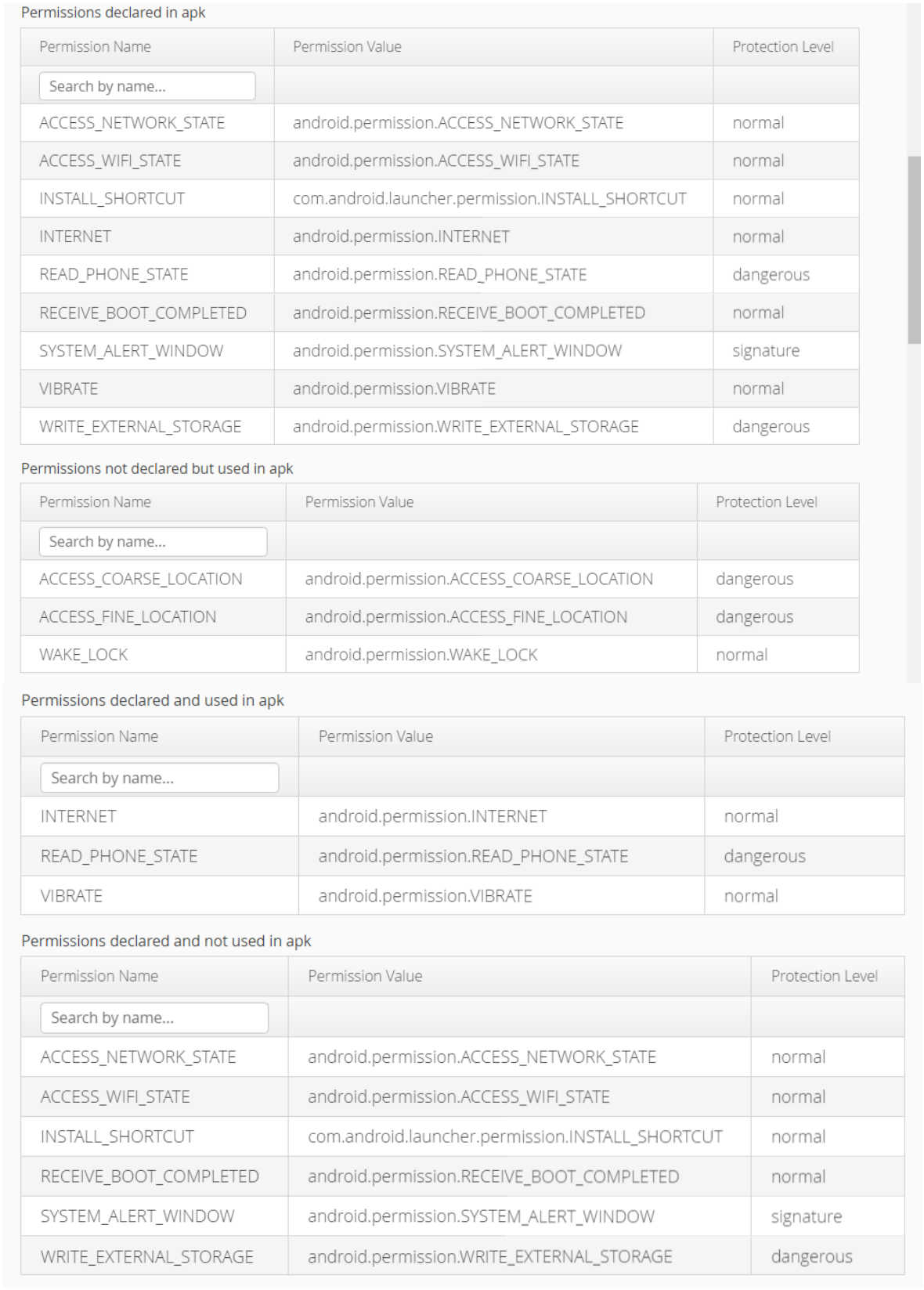
Σχήμα 6.4 Σελίδα για ανέβασμα αρχείου APK

4. Ο χρήστης επιλέγει ένα αρχείο εφαρμογής που επιθυμεί να αναλύσει, αν το αρχείο υπάρχει, τότε μεταφέρεται στην σελίδα με τα αποτελέσματα της συγκεκριμένης εφαρμογής. Ένα παράδειγμα αποτελεσμάτων, παρουσιάζεται στο Σχήμα 6.5, παρουσιάζεται ένα κομμάτι από τις πληροφορίες που εξάγονται. Στο συγκεκριμένο screenshot, παρουσιάζονται οι γενικές πληροφορίες για την εφαρμογή, όπως επίσης το πότε έγινε η ανάλυση και ποιος ανέβασε την εφαρμογή, καθώς επίσης και τα αποτελέσματα από την μηχανική ανάλυση, το σκορ της εφαρμογής καθώς επίσης και τα αποτελέσματα από την σάρωση μέσω του VirusTotal.



Σχήμα 6.5 Αποτελέσματα ανάλυσης εφαρμογής (1)

Στα αποτελέσματα επίσης υπάρχουν και πληροφορίες για τα permissions τα οποία εμφανίζονται με διάφορους τρόπους στην εφαρμογή, οι οποίες παρουσιάζονται με την χρήση πινάκων (Σχήμα 6.6).



Permissions declared in apk

Permission Name	Permission Value	Protection Level
Search by name...		
ACCESS_NETWORK_STATE	android.permission.ACCESS_NETWORK_STATE	normal
ACCESS_WIFI_STATE	android.permission.ACCESS_WIFI_STATE	normal
INSTALL_SHORTCUT	com.android.launcher.permission.INSTALL_SHORTCUT	normal
INTERNET	android.permission.INTERNET	normal
READ_PHONE_STATE	android.permission.READ_PHONE_STATE	dangerous
RECEIVE_BOOT_COMPLETED	android.permission.RECEIVE_BOOT_COMPLETED	normal
SYSTEM_ALERT_WINDOW	android.permission.SYSTEM_ALERT_WINDOW	signature
VIBRATE	android.permission.VIBRATE	normal
WRITE_EXTERNAL_STORAGE	android.permission.WRITE_EXTERNAL_STORAGE	dangerous

Permissions not declared but used in apk

Permission Name	Permission Value	Protection Level
Search by name...		
ACCESS_COARSE_LOCATION	android.permission.ACCESS_COARSE_LOCATION	dangerous
ACCESS_FINE_LOCATION	android.permission.ACCESS_FINE_LOCATION	dangerous
WAKE_LOCK	android.permission.WAKE_LOCK	normal

Permissions declared and used in apk

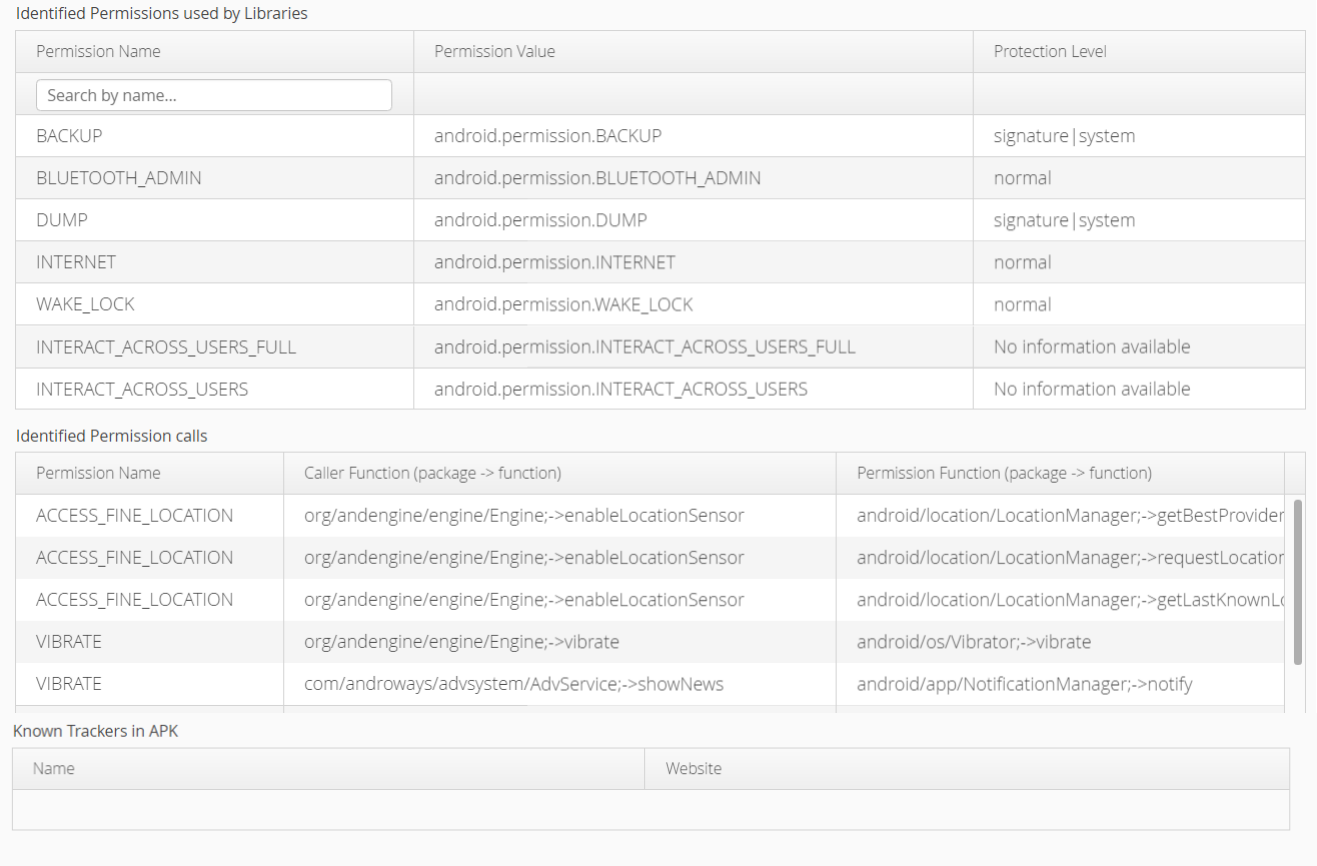
Permission Name	Permission Value	Protection Level
Search by name...		
INTERNET	android.permission.INTERNET	normal
READ_PHONE_STATE	android.permission.READ_PHONE_STATE	dangerous
VIBRATE	android.permission.VIBRATE	normal

Permissions declared and not used in apk

Permission Name	Permission Value	Protection Level
Search by name...		
ACCESS_NETWORK_STATE	android.permission.ACCESS_NETWORK_STATE	normal
ACCESS_WIFI_STATE	android.permission.ACCESS_WIFI_STATE	normal
INSTALL_SHORTCUT	com.android.launcher.permission.INSTALL_SHORTCUT	normal
RECEIVE_BOOT_COMPLETED	android.permission.RECEIVE_BOOT_COMPLETED	normal
SYSTEM_ALERT_WINDOW	android.permission.SYSTEM_ALERT_WINDOW	signature
WRITE_EXTERNAL_STORAGE	android.permission.WRITE_EXTERNAL_STORAGE	dangerous

Σχήμα 6.6 Αποτελέσματα ανάλυσης εφαρμογής (2)

Επίσης μετά από τις πληροφορίες που αφορούν τα permissions που χρησιμοποιεί η εφαρμογή, ακολουθεί ένας πίνακας με τα permissions που χρησιμοποιούν οι βιβλιοθήκες που χρησιμοποιεί η εφαρμογή. Αυτός ο πίνακας, μαζί με τον πίνακα με τις κλήσεις σε μεθόδους που αφορούν τα permissions που χρησιμοποιεί η εφαρμογή φαίνονται στο Σχήμα 6.7.



Identified Permissions used by Libraries

Permission Name	Permission Value	Protection Level
Search by name...		
BACKUP	android.permission.BACKUP	signature|system
BLUETOOTH_ADMIN	android.permission.BLUETOOTH_ADMIN	normal
DUMP	android.permission.DUMP	signature|system
INTERNET	android.permission.INTERNET	normal
WAKE_LOCK	android.permission.WAKE_LOCK	normal
INTERACT_ACROSS_USERS_FULL	android.permission.INTERACT_ACROSS_USERS_FULL	No information available
INTERACT_ACROSS_USERS	android.permission.INTERACT_ACROSS_USERS	No information available

Identified Permission calls

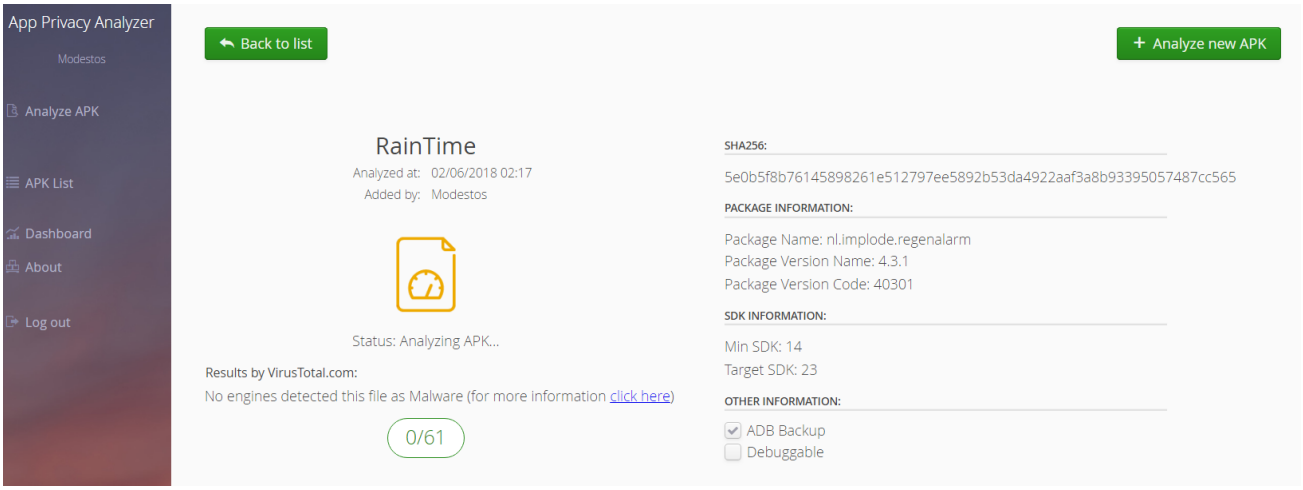
Permission Name	Caller Function (package -> function)	Permission Function (package -> function)
ACCESS_FINE_LOCATION	org/andengine/engine/Engine;->enableLocationSensor	android/location/LocationManager;->getBestProvider
ACCESS_FINE_LOCATION	org/andengine/engine/Engine;->enableLocationSensor	android/location/LocationManager;->requestLocatior
ACCESS_FINE_LOCATION	org/andengine/engine/Engine;->enableLocationSensor	android/location/LocationManager;->getLastKnownL
VIBRATE	org/andengine/engine/Engine;->vibrate	android/os/Vibrator;->vibrate
VIBRATE	com/androways/advsystem/AdvService;->showNews	android/app/NotificationManager;->notify

Known Trackers in APK

Name	Website

Σχήμα 6.7 Αποτελέσματα ανάλυσης εφαρμογής (3)

5. Εάν δεν υπάρχει η ανάλυση για αυτή την εφαρμογή, θα αρχίσει η ανάλυση από το σύστημα βήμα-βήμα όπως αναφέρθηκε και σε προηγούμενες ενότητες, σε αυτή την περίπτωση θα εμφανιστή η σελίδα (Σχήμα 6.8) του APK αλλά δεν θα υπάρχουν αποτελέσματα αφού δεν ολοκληρώθηκαν.

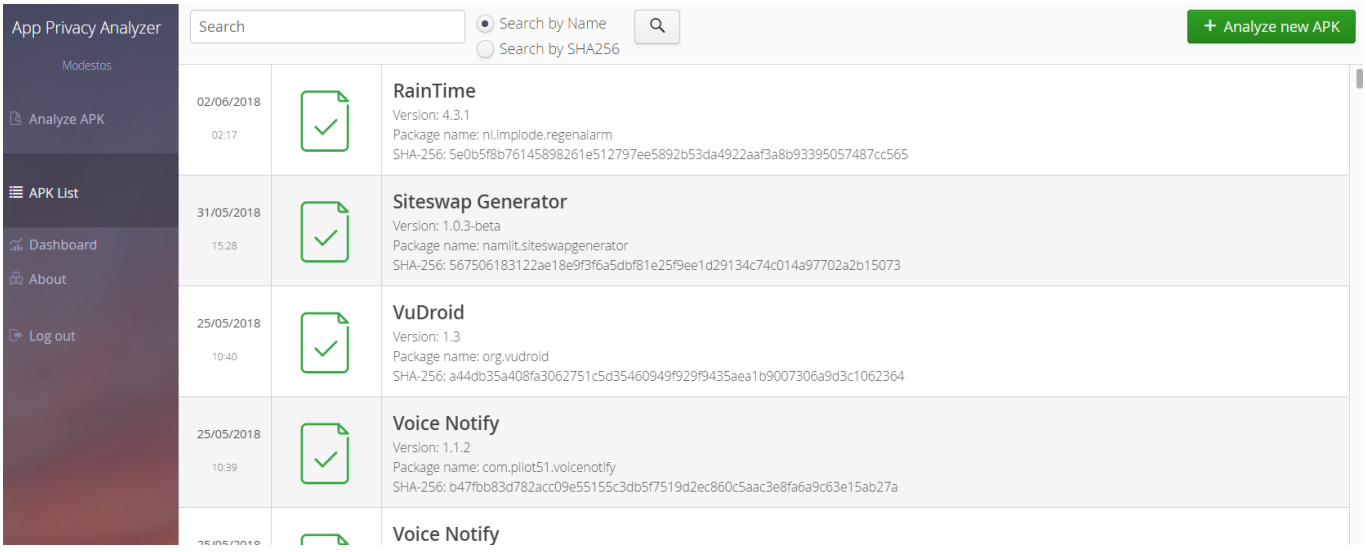


Σχήμα 6.8 Αποτελέσματα εφαρμογής η οποία αναλύεται ακόμα

6. Η διαδικασία ανάλυσης είναι η εξής: (Κώδικας στο Παράρτημα Α)
   a) Πρώτα θα εξαχθούν οι πληροφορίες από το AndroidManifest.xml, οι οποίες αφορούν γενικές πληροφορίες για το APK και αποθηκεύονται στην ΒΔ.
   b) Στην συνέχεια εξάγονται τα permissions της εφαρμογής και αποθηκεύονται και αυτά στην ΒΔ.
   c) Μετέπειτα, με την χρήση του LibRadar εξάγονται οι πληροφορίες για της βιβλιοθήκες.
   d) Στην συνέχεια με βάση τα permissions που χρησιμοποιούνται που βρέθηκαν από το βήμα 5, γίνεται αναζήτηση στο decompile κώδικα μέσο του Androguard για να βρεθούν ποια Permission calls είναι υπεύθυνα για την χρήση των ανάλογων permission.
   e) Τα permissions ακόμα τα χρησιμοποιούμε και για την κατηγοριοποίηση της εφαρμογής σαν επικίνδυνη ή όχι.
   f) Παράλληλα γίνεται και το ανέβασμα του αρχείου στο VirusTotal, για έλεγχο αν είναι κακόβουλο η εφαρμογή.
   g) Γίνεται έλεγχος τον βιβλιοθηκών με γνωστούς trackers για να δούμε αν χρησιμοποιούνται κάποιοι,
   h) Υπολογισμός σκορ, με βάση όλες τις μεταβλητές που αναφέρθηκαν.

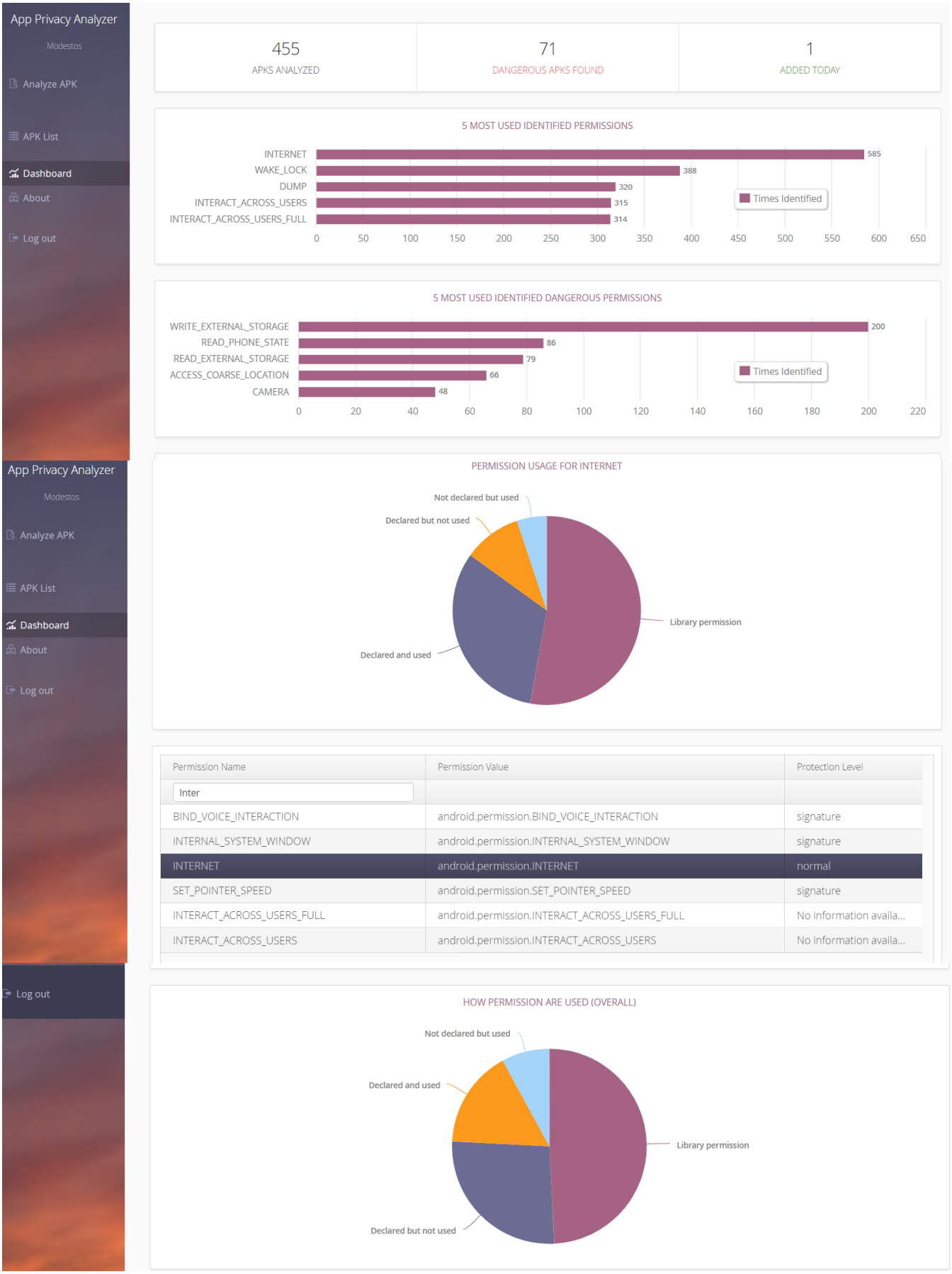
7. Η σελίδα με τις πληροφορίες του APK ανανεώνεται προσθέτοντας τα αποτελέσματα της ανάλυσης.

8. Ο χρήστης επίσης μπορεί να δει την λίστα με όλες τις εφαρμογές οι οποίες αναλύθηκαν, όπως φαίνεται και στο Σχήμα 6.9. Επίσης υπάρχει και η δυνατότητα αναζήτησης μιας εφαρμογής με το όνομα της ή με το SHA256 hash της.



Σχήμα 6.9 Σελίδα με λίστα εφαρμογών οι οποίες έχουν αναλυθεί από το σύστημα

9. Τέλος, υπάρχει η δυνατότητα από το σύστημα να παρουσιάσει διάφορα στατιστικά βάση την ανάλυση των εφαρμογών, όπως φαίνονται στο Σχήμα 4.10



Σχήμα 4.10 Σελίδα με στατιστικά από την ανάλυση

Κεφάλαιο 7

Συλλογή και Ανάλυση αρχείων APK και εξαγωγή αποτελεσμάτων

7.1 Εισαγωγή

Σε αυτή την ενότητα παρουσιάζονται τα διάφορα αποτελέσματα που προέκυψαν από την ανάλυση μιας συλλογής από εφαρμογές, από τις οποίες κάποιες είναι κακόβουλες. Επίσης εξηγείται ο τρόπος που συλλέχτηκαν οι εφαρμογές αυτές καθώς επίσης και πώς έγινε η ανάλυση τους. Η διαδικασία αυτή πραγματοποιήθηκε για να εξαχθούν κάποια συμπεράσματα αναφορικά με τα κενά προστασίας ιδιωτικότητας που μπορούν να εντοπιστούν σε Android εφαρμογές και για να εξαχθούν κάποια σχετικά στατιστικά αποτελέσματα.

7.2 Συλλογή αρχείων APK από διάφορες πηγές

Για την συλλογή αρκετών APK αρχείων έγινε χρήση του fdroid [21], το οποίο είναι μια πλατφόρμα με εφαρμογές ανοιχτού λογισμικού. Το fdroid δίνει την δυνατότητα σε όποιον επιθυμεί να δημιουργήσει το δικό του repository από εφαρμογές, όπως αυτό του fdroid. Ακόμα χρησιμοποιήθηκε και ένα dataset με εφαρμογές οι οποίες είναι γνωστό πως είναι κακόβουλες [22], κάποιες από αυτές μάλιστα υπήρξαν και στο Google Play Store για κάποιο χρονικό διάστημα.

Προχωρώντας με την συλλογή των εφαρμογών, υλοποιήθηκε ένας crawler σε Python βασισμένος σε έναν άλλο crawler [23] ο οποίος δεν αναπτύσσεται πλέον και δεν λειτουργούσε για το fdroid. Ο λόγος που δεν λειτουργούσε είναι ότι χρησιμοποιούσε

διαφορετικά URLs από αυτά που χρησιμοποιώντας τώρα στο fdroid, τα οποία ίσως άλλαξαν. Για παράδειγμα χρησιμοποιούσε το URL *'https://f-droid.org/repository/browse/'* για την λίστα με τις εφαρμογές, το οποίο έχει αλλάξει σε '*https://f-droid.org/packages/*', άρα θα έπρεπε να αλλάξουν τα σημεία στον κώδικα που χρησιμοποιούσαν αυτού του τύπου URLs. Επίσης αλλάχτηκαν και τα regular expressions για να βρίσκουν από το HTML τα κατάλληλα URLs.

Με τον crawler, ξεκινώντας από την σελίδα που υπάρχουν οι εφαρμογές του fdroid, με regular expressions επιλέγονταν οι συνδέσμοι από τον HTML κώδικα που αφορούσαν μια εφαρμογή. Στην συνέχεια σε κάθε ένα από αυτούς του συνδέσμους προχωρούσε ο crawler και έψαχνε στα HTML αυτά συνδέσμους που αφορούσαν συνδέσμους για το κατέβασμα των APK και έστελνε σχετικό αίτημα μέσω της συνάρτησης GET για να τα μεταφορτώσει. Αυτό επαναλαμβανόταν για κάθε σελίδα της λίστας με της εφαρμογές του fdroid.

7.3 Ανάλυση των αρχείων και αποθήκευση αποτελεσμάτων

Για την ανάλυση όλων των αρχείων με αυτοματοποιημένο τρόπο, δημιουργήθηκε μια σελίδα στο σύστημα στην οποία έχει πρόσβαση μόνο ο administrator, όπου υπάρχει η επιλογή να επιλέξει κάποιο φάκελο και να γίνει αυτόματα η ανάλυση για όλα τα αρχεία εκείνου του φακέλου. Τα αποτελέσματα αποθηκεύονταν σε μια MySQL βάση δεδομένων.

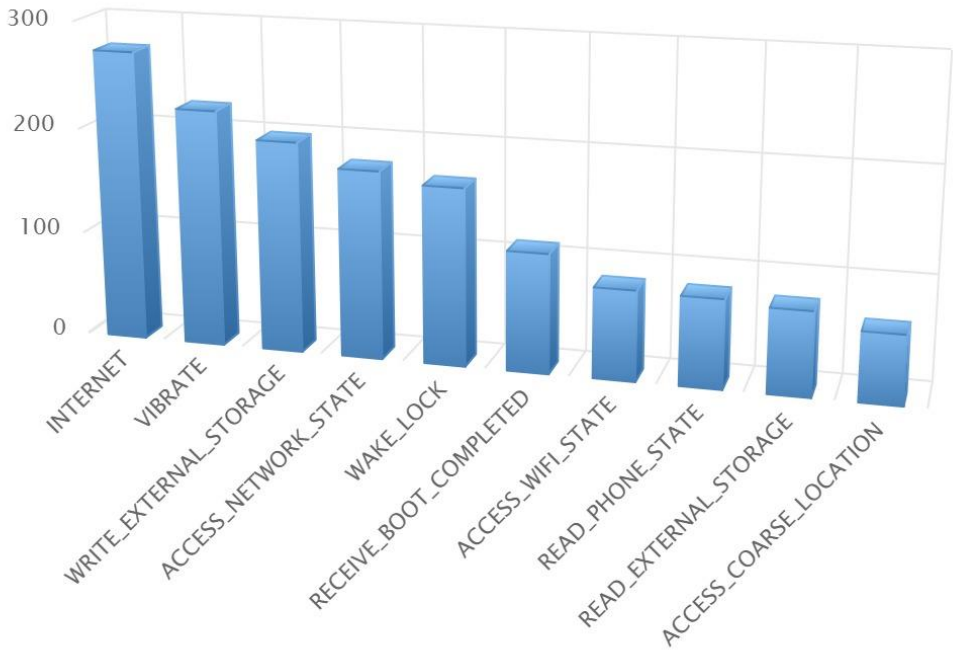
7.4 Ανάλυση Αποτελεσμάτων

Αρχικά συλλέχθηκαν 453 APK εφαρμογών τα οποία όπως αναφέρθηκε και πιο πριν, προέρχονται από το fdroid και από ένα dataset που περιέχει κακόβουλες εφαρμογές. Το dataset αυτό περιείχε 96 κακόβουλές εφαρμογές, όμως περίπου οι μισές δεν είχαν καθόλου επέκταση (π.χ. .apk). Σε κάποιες από αυτές, προσθέτοντας την επέκταση ήταν δυνατό να αναλυθούν αλλά κάποια και πάλι δεν ακολουθούσαν την μορφή APK και δεν μπορούσαν να αναλυθούν από τα εργαλεία. Στο Παράρτημα Β, υπάρχει ένας πίνακας με όλες τις εφαρμογές που συλλέχθηκαν και αναλύθηκαν.

Οι εφαρμογές που συλλέχτηκαν ήταν κυρίως εφαρμογές που βασίζονταν σε εκδόσεις νεότερες από την έκδοση API 21 του Android, δηλαδή Android Lollipop 5.0 και πάνω. Υπήρχαν όμως και εφαρμογές που ήταν βασισμένες σε παλιότερες εκδόσεις. Επίσης επιλέγοντας εφαρμογές ανοιχτού κώδικα, υπήρχε η ευχέρεια να γίνονται κάποιες επιβεβαιώσεις των αναλύσεων βλέποντας τον κώδικα της εφαρμογής. Ακόμα, ήταν καλό να

γνωρίζουμε από πριν αν μια εφαρμογή είναι κακόβουλη για να μπορούμε να δούμε αν η ανάλυση κρίνει την συγκεκριμένη εφαρμογή ως επικίνδυνη με υψηλό σκορ επικινδυνότητας.

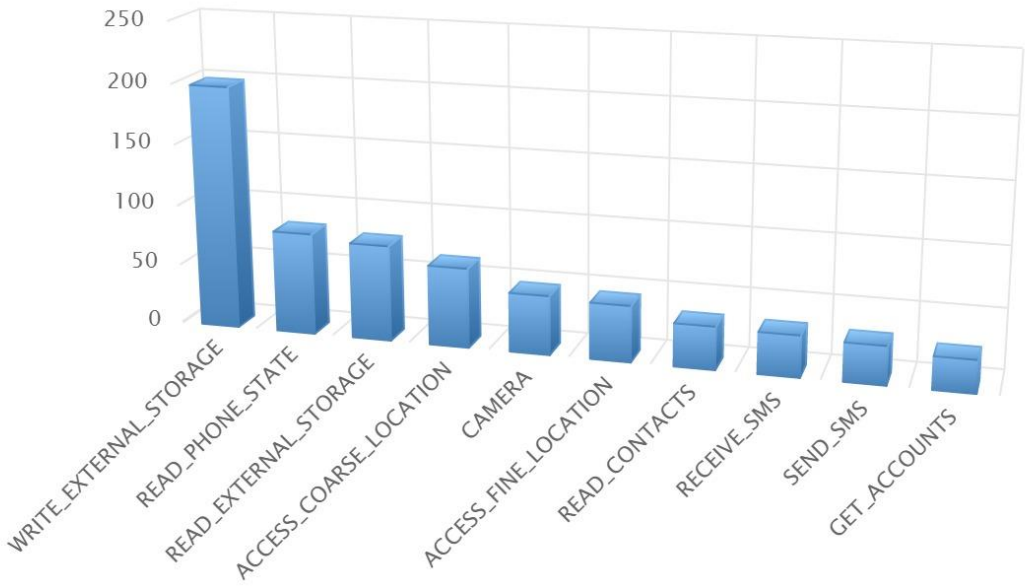
Αφού ολοκληρώθηκε η ανάλυση όλων το αρχείων τα οποία είχαν συλλεχθεί θα ήταν καλό να σχολιαστούν κάποια από τα αποτελέσματα. Από όλες αυτές τις 450 εφαρμογές εντοπίστηκαν 71 επικίνδυνες εφαρμογές με βάση τα permissions. Επίσης κάποιες από τις κακόβουλές εφαρμογές οι οποίες δεν χρησιμοποιούσαν αρκετά permissions, δεν κατηγοριοποιήθηκαν σαν επικίνδυνες.



Σχήμα 7.1 Τα 10 permissions που χρησιμοποιούνται πιο συχνά στις εφαρμογές που αναλύθηκαν

Ξεκινώντας, θα ήταν καλό να δούμε μετά την ολοκλήρωση της ανάλυσης ποια permissions εντοπίστηκαν να χρησιμοποιούνται πιο συχνά. Στο ‘Σχήμα 7.1’ φαίνεται μια γραφική η οποία παρουσιάζει τα 10 permissions τα οποία χρησιμοποιήθηκαν περισσότερο στις εφαρμογές που αναλύθηκαν. Όπως βλέπουμε το permission που έχει να κάνει με το Διαδίκτυο είναι αυτό που αναγνωρίστηκε πιο συχνά (περίπου σε 270 εφαρμογές – 59.6% των εφαρμογών). Αυτό είναι λογικό, καθώς οι περισσότερες εφαρμογές μπορεί να χρειάζονται πρόσβαση στο Διαδίκτυο για να μπορούν να κάνουν κάποιες από τις λειτουργίες τους. Επίσης ούτε το δεύτερο πιο χρησιμοποιημένο permission αποτελεί κάτι μη αναμενόμενο, αφού η δόνηση (VIBRATE) χρησιμοποιείται σε πληθώρα από εφαρμογές για διάφορους σκοπούς. Ακολούθως, είναι το permission WRITE_EXTERNAL_STORAGE, το οποίο δίνει πρόσβαση στην εφαρμογή να δημιουργήσει αρχεία στην συσκευή που είναι εγκατεστημένη.

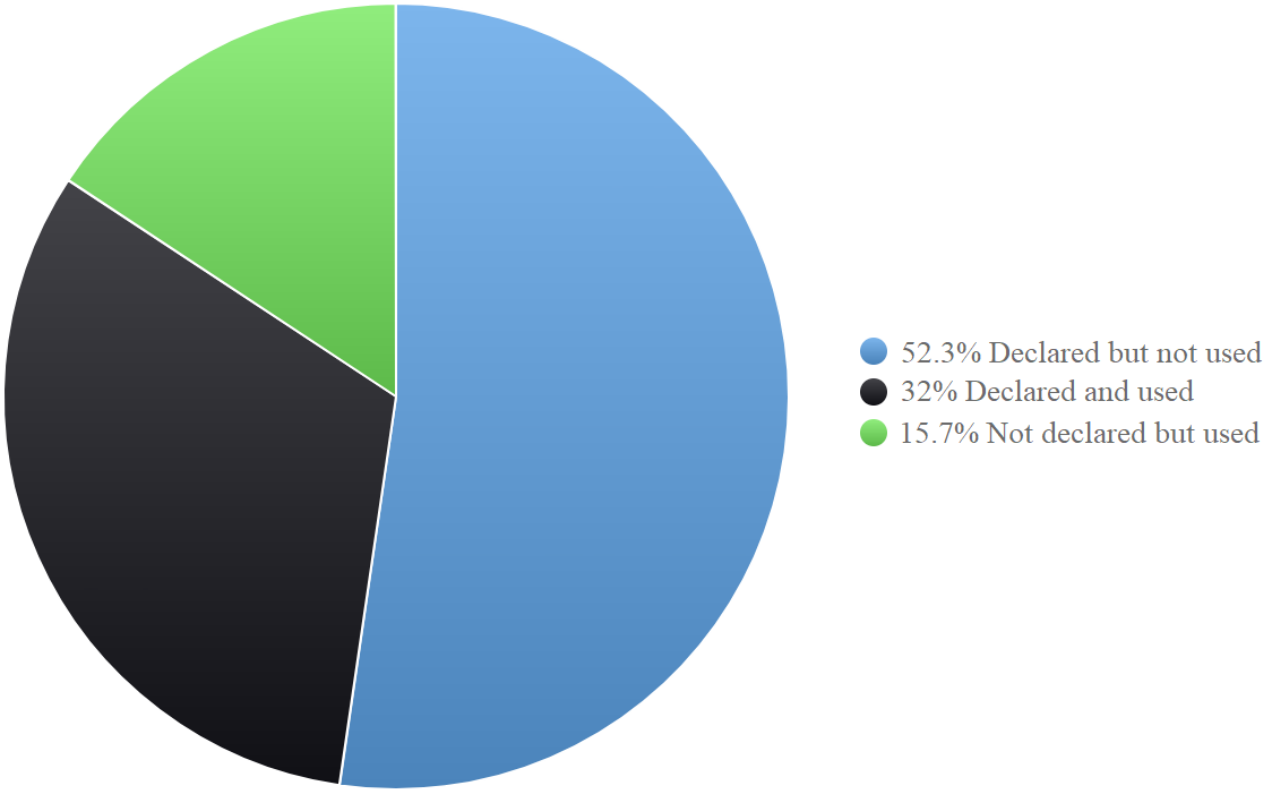
Στην συνέχεια παρουσιάζονται τα permissions τα οποία είναι dangerous και εμφανίστηκαν συχνότερα να χρησιμοποιούνται από τις εφαρμογές που αναλυθήκαν.



Σχήμα 7.2 Τα 10 dangerous permissions τα οποία χρησιμοποιούνται πιο συχνά στις εφαρμογές που αναλύθηκαν

Παρατηρείται ότι το permissions WRITE_EXTERNAL_STORAGE, το οποίο σου επιτρέπει να δημιουργείς και να γράφεις σε αρχεία στην συσκευή, είναι το πιο συχνά χρησιμοποιημένο στις εφαρμογές που αναλύθηκαν (περίπου από 195 εφαρμογές – 43% των εφαρμογών). Επίσης φαίνεται ότι υπάρχει αρκετή διαφορά μεταξύ του πρώτου που χρησιμοποιείται πιο συχνά με τα υπόλοιπα. Τέλος είναι παράξενο το ότι το permission READ_EXTERNAL_STORAGE φαίνεται να χρησιμοποιείται από λιγότερες εφαρμογές από το WRITE_EXTERNAL_STORAGE, τα οποία μπορεί να πει κάποιος ότι χρησιμοποιούνται μαζί για να μπορεί κάποιος και να διαβάσει και να γράψει σε αρχείο. Αυτό εξηγείται διότι με την αναβάθμιση στο API 19 του Android, όταν μια εφαρμογή δηλώσει το permissions WRITE_EXTERNAL_STORAGE, τότε αυτόματα έχει και δικαίωμα για READ_EXTERNAL_STORE. Άρα για αυτό τον λόγο υπάρχει τόση διαφορά μεταξύ τους, αφού η εφαρμογές δεν χρειάζεται να το δηλώσουν ξεχωριστά για να το χρησιμοποιήσουν.

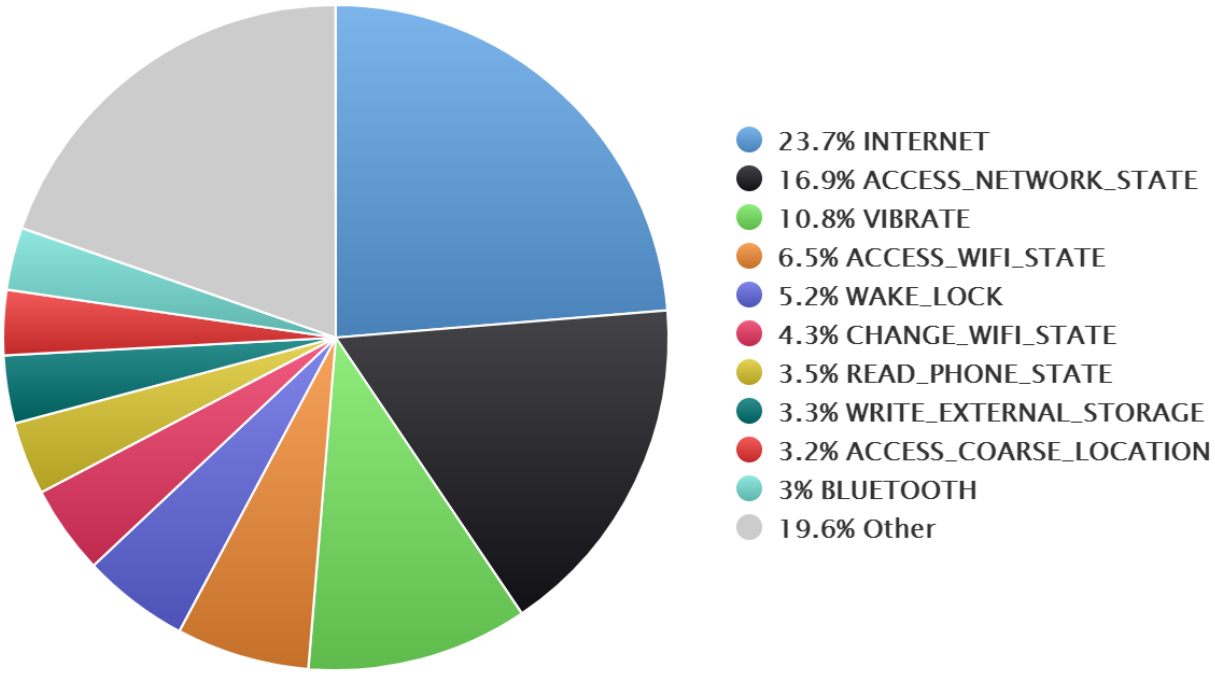
Σε αυτή την φάση θα ήταν καλό να κοιτάξουμε το πώς χρησιμοποιούνται τα permissions με βάση τις κατηγορίες που χωρίστηκαν κατά την διάρκεια εξαγωγής των permissions από το APK.



Σχήμα 7.3 Πώς εμφανίζονται να χρησιμοποιούνται τα permissions στις εφαρμογές

Παρατηρούμαι ότι λίγα περισσότερα από τα μισά permissions που εντοπίστηκαν, είναι δηλωμένα αλλά δεν χρησιμοποιούνται. Το ποσοστό είναι μεγάλο αφού δεν υπάρχει λόγος να δηλώνει κάποιος οποιοδήποτε permission το οποίο δεν σκοπεύει να χρησιμοποιήσει. Επίσης, βλέπουμε ότι λίγο περισσότερο από το ένα τρίτο των permissions, είναι δηλωμένα και χρησιμοποιούνται. Το ποσοστό αυτό είναι χαμηλό, το οποίο είναι παράξενο αφού αυτός είναι ο σωστός τρόπος χρήσης των permissions. Αυτό σημαίνει ότι οι προγραμματιστές δεν δίνουν αρκετό βάρος σε αυτό το σημείο με αποτέλεσμα να υπάρχουν δηλωμένα αχρείαστα permissions. Τέλος παρατηρείται ότι το 15.7% των permissions αυτών, χρησιμοποιούνται χωρίς να είναι δηλωμένα στο AndroidManifest.xml. Αυτό συμβαίνει με έμμεσο τρόπο, όπως για παράδειγμα να χρησιμοποιείται από τις βιβλιοθήκες που χρησιμοποιεί η εφαρμογή. Αυτό είναι επικίνδυνο επειδή ο χρήστης δεν θα γνωρίζει ότι η εφαρμογή θα χρησιμοποιήσει άλλο permission αφού κατά την εγκατάσταση δεν αναφέρθηκε το permission αυτό.

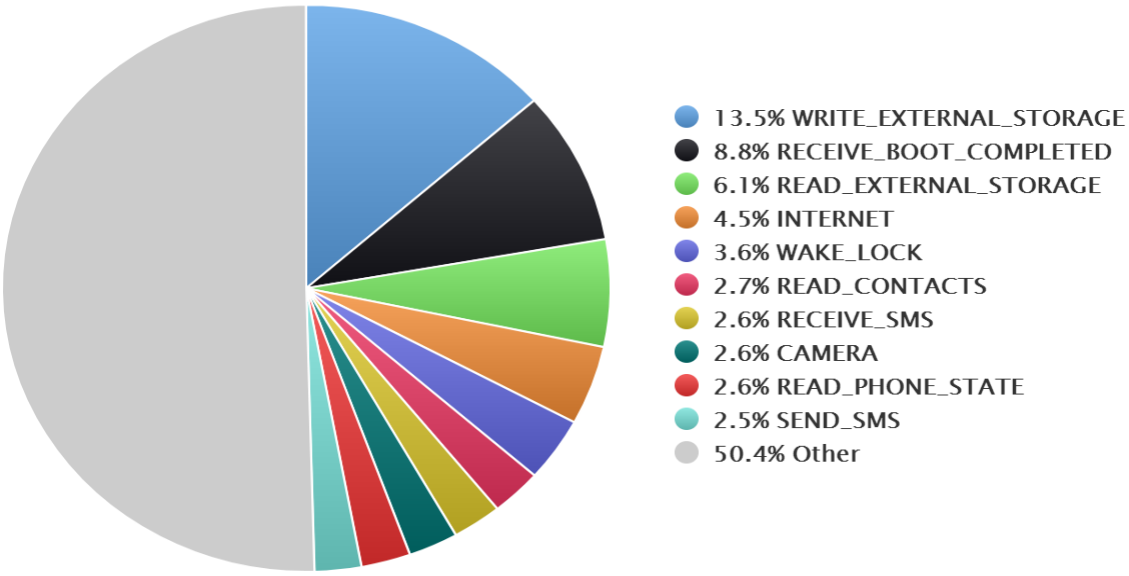
Στην συνέχεια θα κοιτάξουμε την κάθε κατηγορία permissions για να δούμε ποια permissions συμπεριλαμβάνονται στην κατηγορία και πόσο συχνά εμφανίζονται.



Σχήμα 7.4 Permissions που είναι δηλωμένα και χρησιμοποιούνται

Πιο πάνω φαίνονται τα permissions τα οποία εμφανίζονται πιο συχνά δηλωμένα και χρησιμοποιούνται στην εφαρμογή. Λογικό, όπως και πριν να είναι πρώτο το permission για το Διαδίκτυο αφού γενικότερα οι εφαρμογές τείνουν να χρησιμοποιούν το Διαδίκτυο για τις διάφορες λειτουργίες τους. Επίσης, παρατηρούμαι ότι το δεύτερο permission είναι το ACCESS_NETWORK_STATE μέσω του οποίου μια εφαρμογή μπορεί να δει αν η συσκευή είναι συνδεδεμένη σε δίκτυο και γενικότερα να μάθει πληροφορίες για το δίκτυο. Φαίνεται ότι δεν υπάρχει στα πρώτα permissions δεν υπάρχουν dangerous permissions τα οποία να εμφανίζονται σαν δηλωμένα και να χρησιμοποιούνται. Το dangerous permission που εμφανίστηκε περισσότερο σε αυτή την κατηγορία είναι το READ_PHONE_STATE με 3.5%.

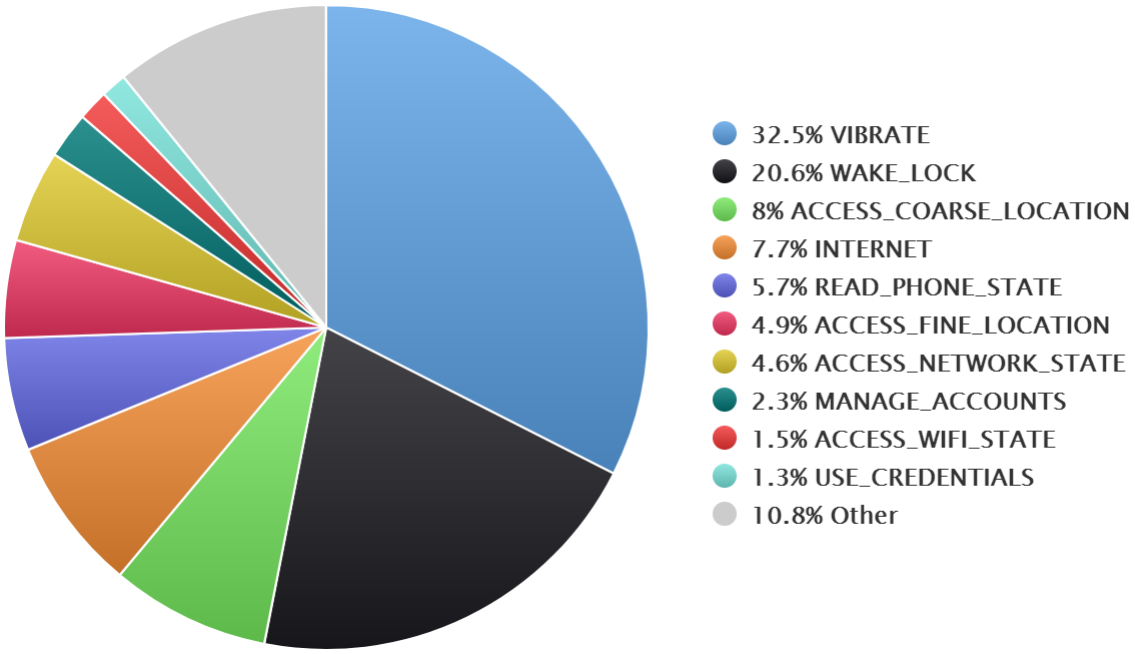
Η επόμενη γραφική παράσταση (Σχήμα 7.5) αφορά τα permissions τα οποία μπορεί να είναι δηλωμένα αλλά δεν χρησιμοποιούνται από την εφαρμογή. Παρατηρείται ότι υπάρχουν διάφορα permissions που ανήκουν σε αυτή την κατηγορία αφού είναι τα permissions αυτά που για κάποιο λόγο έμειναν δηλωμένα στο AndroidManifest.xml αλλά δεν χρησιμοποιούνται.



Σχήμα 7.5 Permissions που είναι δηλωμένα αλλά δεν χρησιμοποιούνται

Επίσης μπορούμε να εντοπίσουμε ότι το permission που εντοπίστηκε να εμφανίζεται πιο συχνά σε αυτή την κατηγορία είναι το WRITE_EXTERNAL_STORAGE με 13.5%. Μια πιθανή εξήγηση για τον λόγο που συμβαίνει αυτό σε κάποια permission, είναι το ότι μπορεί κάποιοι προγραμματιστές κατά την διάρκεια της ανάπτυξης του συστήματος, να χρησιμοποιήσαν διάφορα permissions για δοκιμές, κτλ. και όταν φτάσουν στην φάση που θα δημιουργήσουν το APK δεν αφαιρούν τα δηλωμένα permissions που δεν χρησιμοποιούν. Ίσως χρησιμοποιηθούν στο μέλλων από τον προγραμματιστή για να εισάγει κάποια νέα λειτουργεία.

Η επόμενη κατηγορία permission που θα αναλυθεί και φαίνεται στο Σχήμα 6.6 είναι τα permissions τα οποία δεν είναι δηλωμένα αλλά χρησιμοποιούνται στην εφαρμογή. Όπως αναφέρθηκε και πιο πάνω τα permissions αυτά δεν υπάρχουν στο AndroidManifest.xml, αλλά με κάποιο τρόπο χρησιμοποιούνται στην εφαρμογή αφού μπορούμε να τα εντοπίσουμε με την εύρεση μεθόδων που αφορούν το αντίστοιχο μη δηλωμένο permission. Από την γραφική βλέπουμε ότι το permission που έχει να κάνει με την δόνηση της συσκευής εμφανίζεται πιο συχνά ως μη δηλωμένα το οποίο όμως χρησιμοποιείται . Αυτό είναι λογικό αφού η δόνηση μπορεί να χρησιμοποιηθεί για διάφορους σκοπούς όπως όταν υπάρχει κάποιο λάθος στα στοιχεία του χρήστη που έχει δώσει μπορεί να δονηθεί το κινητό. Επίσης κάποιες βιβλιοθήκες πληκτρολογίου μπορεί να εμφανίζουν διαφορετικού τύπου πληκτρολόγια μπορεί να προκαλούν δόνηση στην συσκευή κατά την διάρκεια που γράφεις κάτι. Το δεύτερο permission

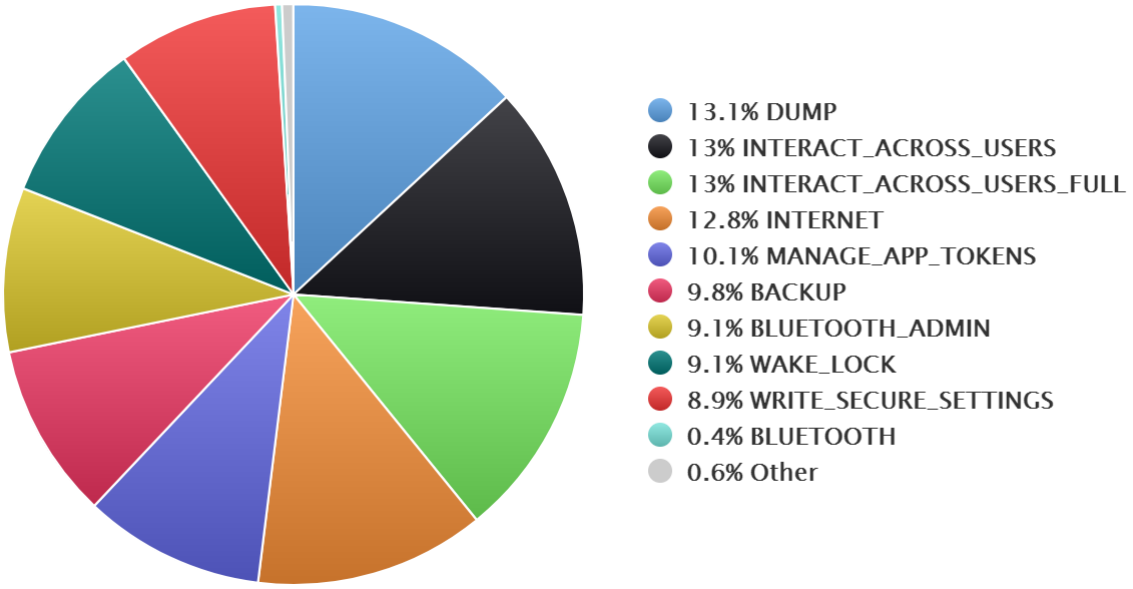


Σχήμα 7.6 Permissions τα οποία δεν είναι δηλωμένα αλλά χρησιμοποιούνται στην εφαρμογή

που εμφανίστηκε πιο συχνά σε αυτή την κατηγορία είναι το WAKE_LOCK το οποίο εμποδίζει την οθόνη του κινητού να σβήσει και τον επεξεργαστεί να κάνει sleep. Αυτό μπορεί να χρησιμοποιείται όταν ο χρήστης δεν κάνει κάποια ενέργεια για κάποια στιγμή να μην σβήνει η οθόνη. Θα μπορούσε κάποια βιβλιοθήκη η οποία συλλέγει δεδομένα συνεχώς από μια εφαρμογή να εμποδίζει τον επεξεργαστή να κάνει sleep για να μπορεί να λειτουργά κανονικά. Το τρίτο permission που βρίσκεται στην κατηγορία αυτή είναι ένα dangerous permission το οποίο βρίσκει την τοποθεσία του χρήστη στο περίπου. Αυτό είναι πολύ επικίνδυνο αφού μια εφαρμογή μπορεί να μην χρειάζεται καν αυτή την πληροφορία αλλά με κάποιον τρόπο κάποια βιβλιοθήκη να παίρνει πληροφορίες για την τοποθεσία του χρήστη, χωρίς αυτός να το γνωρίζει.

Μια άλλη κατηγορία permissions η οποία δεν αναφέρθηκε μέχρι στιγμής, είναι τα permissions που έχουν οι βιβλιοθήκες. Όπως είπαμε πριν εξάγονται οι βιβλιοθήκες με την βοήθεια του LibRadar και στα αποτελέσματα υπάρχουν και τα permissions τις κάθε βιβλιοθήκης. Οι πληροφορίες με τα permissions των βιβλιοθηκών αποθηκευόταν και αυτή στην βάση δεδομένων, έτσι ώστε να μπορούμε να δούμε τι permissions χρησιμοποιούν αυτές οι βιβλιοθήκες που συμπεριλαμβάνονται στις εφαρμογές που αναλύθηκαν. Πιο κάτω στο Σχήμα 7.7 φαίνεται η γραφική παράσταση που παρουσιάζει τα permissions που χρησιμοποιούν οι βιβλιοθήκες. Παρατηρούμαι ότι τα 4 πρώτα permissions εμφανίζονται όλα κοντά στο 13% σε αυτή την κατηγορία. Τα 3 πρώτα permissions είναι τύπου Signature, δηλαδή για να τα χρησιμοποιηθούν πρέπει η εφαρμογή να είναι υπογεγραμμένη (signed) από το ίδιο certificate με την εφαρμογή που έχει δηλωμένο το permission. Στην περίπτωση εδώ,

λογικά θα είναι κάποια βιβλιοθήκη της Google για να μπορεί να χρησιμοποιήσει το permission αυτό.

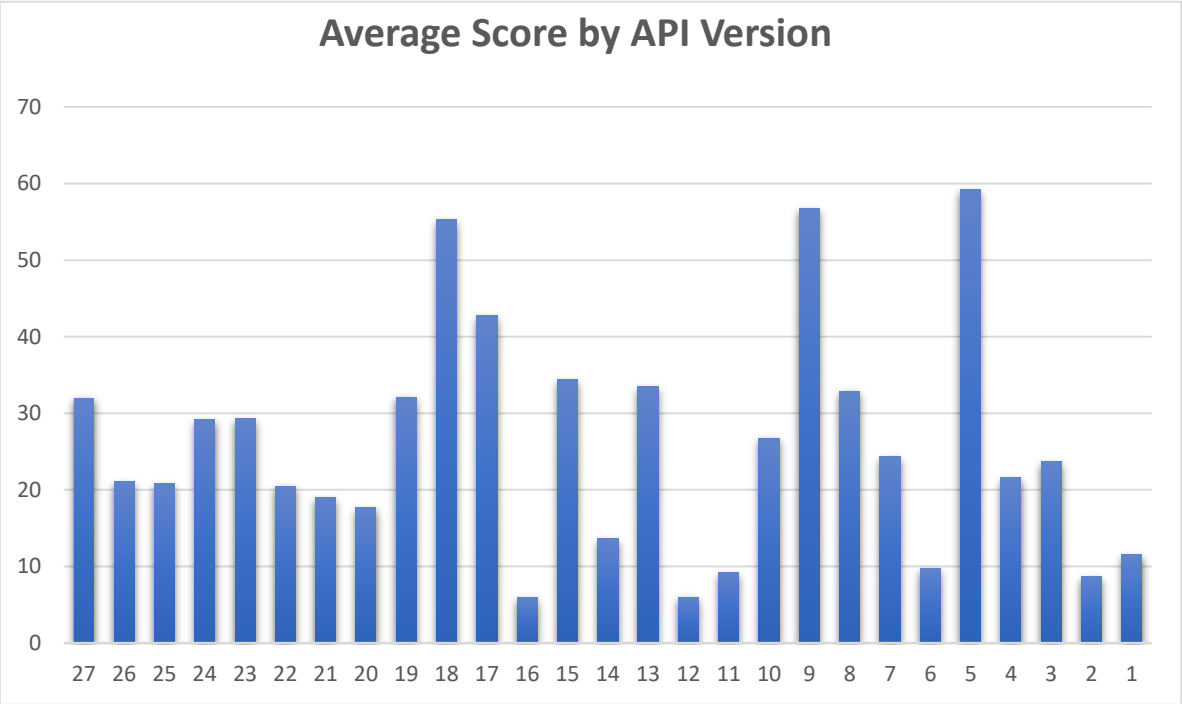


Σχήμα 7.7 Permissions που χρησιμοποιούνται από τις βιβλιοθήκες

Το ίδιο ισχύει και για τα permissions INTERACT_ACROSS_USERS και INTERACT_ ACCROS_USERS_FULL τα οποία δεν είναι επίσημα στο API (ίσως να μην χρησιμοποιείτε πλέον) και σου δίνουν την δυνατότητα να διαχειριστείς λειτουργίες που αφορούν όλους τους χρήστες της συσκευής. Επίσης το permission INTERNET χρησιμοποιείται και αυτό αρκετά, αφού πολλές βιβλιοθήκες θα χρειάζονται διαδίκτυο για να πραγματοποιήσουν κάποιες λειτουργίες τους.

Γενικότερα το μόνο πρόβλημα που εντοπίστηκε είναι το ότι περισσότερα από τα μισά permissions που είναι δηλωμένα δεν χρησιμοποιούνται, καθώς επίσης χρησιμοποιούνται και αρκετά permissions τα οποία δεν είναι δηλωμένα. Με τα πιο πάνω αποτελέσματα, είναι πιο ξεκάθαρο ότι πρέπει να αποφεύγονται να χρησιμοποιούνται με τέτοιο τρόπο τα permissions. Θα πρέπει να αυξηθεί το ποσοστό των permission που είναι δηλωμένα και χρησιμοποιούνται και να μειωθεί το ποσοστό των υπολοίπων.

Πιο κάτω στο Σχήμα 7.8 γίνεται μια σύγκριση των σκορ από εφαρμογές που αναλύθηκαν των διαφόρων εκδόσεων του Android.



Σχήμα 7.8 Μέσος όρος σκορ ανά έκδοση Android

Όπως φαίνεται δεν υπάρχει κάποια τάση μειώσεις του σκορ, αλλά γενικότερα παρατηρείτε ότι στις περισσότερες έκδοσης το σκορ είναι χαμηλότερο από 40, που σημαίνει ότι η εφαρμογή δεν έχει κάποιο ιδιαίτερο ρίσκο δεν χρησιμοποιεί πολλά permissions τα οποία μπορεί να χρησιμοποιήσουν προσωπικά ευαίσθητα δεδομένα.

Επίσης, πιο κάτω παρουσιάζονται πιο permission χρησιμοποιείτε πιο συχνά στις εφαρμογές που αναλύθηκαν με βάση την έκδοση Android που είναι βασισμένη η εφαρμογή.

API Version	Permission	API Version	Permission
1	READ_PHONE_STATE	15	INTERNET
2	INTERNET	16	INTERNET
3	INTERNET	16	MANAGE_APP_TOKENS
4	INTERNET	17	INTERNET
5	WRITE_EXTERNAL_STORAGE	18	WRITE_EXTERNAL_STORAGE
6	READ_PHONE_STATE	19	INTERNET
7	INTERNET	20	WRITE_SYNC_SETTINGS
8	WAKE_LOCK	21	INTERNET
9	INTERNET	22	INTERNET
10	WRITE_EXTERNAL_STORAGE	23	INTERNET
11	WAKE_LOCK	24	INTERNET
12	WRITE_EXTERNAL_STORAGE	25	INTERNET

13	INTERNET	26	INTERNET
14	WRITE_EXTERNAL_STORAGE	27	INTERNET

Πίνακας 7.1 Πιο συχνά χρησιμοποιημένα permissions ανά έκδοση Android

Παρατηρείται ότι το permission που χρησιμοποιείται περισσότερο στις εκδόσεις είναι το permission που αφορά το Διαδίκτυο (android.permission.INTERNET). Όμως παρατηρείτε ότι το permission που επιτρέπει στην εφαρμογή να γράψει στην συσκευή (π.χ. δημιουργία αρχείου) χρησιμοποιείται και αυτό περισσότερο από το permission του διαδικτύου σε κάποιες εκδόσεις, παρόλο που θεωρείται dangerous permission.

Κεφάλαιο 8

Συμπεράσματα και Μελλοντική Εργασία

8.1 Εισαγωγή

Κατά την διάρκεια εκπόνησης της διπλωματικής εργασίας αυτής και μέσα από την ανάλυση και περαιτέρω μελέτη της Ιδιωτικότητας, των εφαρμογών Android καθώς και την σχεδίαση και υλοποιήσει του συστήματος ανάλυσης εφαρμογών, οδήγησαν σε κάποια συμπεράσματα. Σε αυτό το κεφάλαιο θα παρουσιαστούν τα συμπεράσματα αυτά, καθώς επίσης και μελλοντικές εργασίες που μπορεί να γίνουν για βελτίωση του εργαλείου αναλύσεις αλλά και γενικότερα μελέτες σχετικά με την Ιδιωτικότητα και προάστια δεδομένων σε κινητές εφαρμογές.

8.2 Γενικά Συμπεράσματα

Μέσα από την εκπόνηση αυτής της ατομικής διπλωματικής εργασίας η οποία αφορούσε την ανάλυση Ιδιωτικότητας και ασφάλειας σε κινητές συσκευές καθώς και από την όλη μελέτη περί του θέματος, προέκυψαν κάποια συμπεράσματα.

Οι κινητές εφαρμογές καθημερινά αυξάνονται και πολλές φορές κάποιες από αυτές είναι κακόβουλές ή προσπαθούν με έμμεσο τρόπο να εκμεταλλευτούν τον χρήστη που τις εγκατέστησε, συλλέγοντας του πληροφορίες ή κάνει κακό στην συσκευή του. Επίσης οι προγραμματιστές κινητών εφαρμογών θα πρέπει να είναι προσεκτικοί με το τι βιβλιοθήκες χρησιμοποιούν καθώς αυτές μπορεί να προκαλέσουν διάφορα προβλήματα χωρίς την θέληση του προγραμματιστή.

Με την ανάπτυξη του συστήματος ανάλυσης θα μπορεί εύκολα κάποιος να αναλύσει κάποια εφαρμογή ανεβάζοντας το αρχείο APK για να πάρει πληροφορίες σχετικά με τα permissions

που χρησιμοποιεί και που ακριβώς (που καλούνται), τους trackers, τις βιβλιοθήκες αλλά ακόμα να δει το σκορ που έχει η εφαρμογή καθώς και το αν θεωρείτε επικίνδυνή με βάση τα permissions ή όχι. Τέλος θα μπορεί παράλληλα να έχει και τα αποτελέσματα από το VirusTotal το οποίο με την βοήθεια επώνυμων antivirus λογισμικών θα μπορεί να δει κατά πόσο η εφαρμογή αυτή είναι κακόβουλη.

8.3 Μελλοντική Εργασία

Θα μπορούσε να χρησιμοποιηθεί δυναμική ανάλυση της εφαρμογής, έτσι ώστε να μπορεί να καταγραφούν τα δεδομένα που συλλέγει και που χρειάζεται η εφαρμογή για να υπάρχουν πιο ακριβής αποτελέσματα.
Για δυναμική ανάλυση θα μπορούσα να χρησιμοποιηθούν εργαλεία:

- MobSF [61]
- DECAF [62]
- DroidBox [63]

Επίσης θα μπορούσε να γίνεται έλεγχος δικτύου και να καταγράφονται οι συνδέσεις που κάνει η εφαρμογή σε άλλους servers για παράδειγμα, έτσι ώστε να δημιουργηθεί μια λίστα με επικίνδυνους server προς αποφυγή. Το exodus παρουσιάζει ότι έχει αυτή την δυνατότητα, όμως δεν εμφανίζονται πουθενά τα αποτελέσματα αυτά, ίσως υλοποιηθεί στο μέλλον και από αυτούς.

Θα μπορούσε με στατική ανάλυση να παρουσιάζεται ολόκληρός ο κώδικας της συνάρτησης που γίνεται το κάλεσμα μιας μεθόδου ενός permission. Αυτό θα μπορούσε να γίνει χρησιμοποιώντας κάποιο εργαλείο όπως το Androguard για decompile, όμως δεν είναι πάντα ακριβής ο κώδικας που επιστρέφει για να μπορεί κάποιος να κάνει αυτή την λειτουργία και να είναι απόλυτα σίγουρος.

Αναφορικά με θέματα νομοθεσίας θα ήταν σημαντικό να ελέγχεται κατά πόσο κάποια εφαρμογή εναρμονίζεται με την υφιστάμενη νομοθεσία, ειδικά σε σχέση με το GDPR, όσον αφορά τον τρόπο χρήσης των δεδομένων των χρηστών. Θα είχε νόημα η υλοποιήσει διαφόρων μεθόδων έτσι ώστε οι υπεύθυνοι προστασίας δεδομένων, (DPO – Data Protection Officers), να έχουν στην διάθεση τους εργαλεία τα οποία θα τους βοηθήσουν να ελέγχουν μια εφαρμογή ή ένα σύστημα τι πληροφορίες χρειάζεται και πως τις χρησιμοποιεί. Κάτι τέτοιο προτείνεται και στο paper των Pietro Ferrara και Fausto Spoto [64].

Βιβλιογραφία

[1] Edward Snowden just made an impassioned argument for why privacy is the most important right.
http://www.businessinsider.com/edward-snowden-privacy-argument-2016-9

[2] Universal Declaration of Human Rights
http://www.un.org/en/universal-declaration-human-rights/

[3] International Covenant on Civil and Political Rights
http://www.ohchr.org/en/professionalinterest/pages/ccpr.aspx

[4] European Convention on Human Rights
https://www.echr.coe.int/Pages/home.aspx?p=basictexts&c

[5] Alan F. Westin, "Social and Political Dimensions of Privacy"
https://spssi.onlinelibrary.wiley.com/doi/full/10.1111/1540-4560.00072

[6] Herman T. Tavani, "Informational privacy, data mining, and the Internet"
https://www.researchgate.net/publication/227017245_Informational_privacy_data_mining_and_the_Internet

[7] Deborah G. Johnson , Helen Nissenbaum, "Computers, ethics & social values"
https://dl.acm.org/citation.cfm?id=206759

[8] Konrad Zweigert, Hein Kötz, "An Introduction to Comparative Law"
https://books.google.com.cy/books/about/An_Introduction_to_Comparative_Law.html?id=3ju2QgAACAAJ&redir_esc=y

[9] Cambridge Analytica closing after Facebook data harvesting scandal
https://www.theguardian.com/uk-news/2018/may/02/cambridge-analytica-closing-down-after-facebook-row-reports-say

[10] zipfile
https://docs.Python.org/2/library/zipfile.html

[11] xml.dom
https://docs.Python.org/2/library/xml.dom.html

[12] Lifetimes of cryptographic hash functions
http://valerieaurora.org/hash.html

[13] RiskInDroid
https://github.com/ClaudiuGeorgiu/RiskInDroid/

[14] Talos Security
https://www.talos-sec.com/

[15] A. Merlo, G.C. Georgiu. "RiskInDroid: Machine Learning-based Risk Analysis on Android", in Proceedings of the 32nd International Conference on ICT Systems Security and Privacy Protection (IFIP-SEC 2017).

[16] PScout
http://pscout.csl.toronto.edu/

[17] Urcuqui, Christian - Dataset malware/benign permissions Android
https://www.kaggle.com/xwolf12/datasetandroidpermissions

[18] Urcuqui, C., & Navarro, A. (2016, April). Machine learning classifiers for android malware analysis. In Communications and Computing (COLCOM), 2016 IEEE Colombian Conference on (pp. 1-6). IEEE.

[19] Weka
https://www.cs.waikato.ac.nz/ml/weka/

[20] εxodus Privacy
https://exodus-privacy.eu.org/

[21] F-Droid
https://f-droid.org

[22] Collection of android malware samples
https://github.com/ashishb/android-malware

[23] Third Party Android App Store Crawlers
https://github.com/anatolikalysch/APKCrawler

[24] C. Efstratiou and I. Leontiadis, “What is the price of free,” Online; accessed at April 17, 2012. Available: http://www.cam.ac.uk/research/news/what-is-the-price-of-free

[25] S. Gunasekera, Android Apps Security, 1st ed. Berkely, CA, USA: Apress, 2012.

[26] A. P. Felt, E. Ha, S. Egelman, A. Haney, E. Chin, and D. Wagner, “Android permissions: User attention, comprehension, and behavior,” in Proc. of the 8th Symposium on Usable Privacy and Security (SOUPS’12), Pittsburgh, PA, USA. ACM, July 2012, pp. 3:1–3:14. [Online]. Available: http://doi.acm.org/10.1145/2335356.2335360

[27] A. P. Felt, E. Chin, S. Hanna, D. Song, and D. Wagner, “Android permissions demystified,” in Proc. of the 18th ACM conference on Computer and communications security (CCS’11), Chicago, IL, USA. ACM, October 2011, pp. 627–638.

[28] L. Davi, A. Dmitrienko, A.-R. Sadeghi, and M. Winandy, “Privilege escalation attacks on android,” in Proc. of the 13th Information Security Conferenec (ISC’11), Boca Raton, Florida, USA, LNCS, M. Burmester, G. Tsudik, S. Magliveras, and I. Ilic, Eds., vol. 6531. Springer Berlin Heidelberg, October 2011, pp. 346–360. [Online].
Available: http://dx.doi.org/10.1007/978-3-642-18178-8 30

[29] C. Marforio, A. Francillon, and S. Capkun, Application Collusion Attack on the Permission-Based Security Model and Its Implications for Modern Smartphone Systems. Department of Computer Science, ETH
Zurich, 2010. [Online]. Available: https://books.google.com/books?id=nvszMwEACAAJ

[30] Karina Sokolova , Marc Lemercier, Jean-Baptiste Boisseau : “Privacy by Design Permission System for Mobile Applications“ May 2014

[31] Zimmeck et al., Automated Analysis of Privacy Requirements for Mobile Apps, 2016

[32] Christopher Mann , Artem Starostin, A framework for static detection of privacy leaks in android applications, Proceedings of the 27th Annual ACM Symposium on Applied Computing, March 26-30, 2012, Trento, Italy

[33] William Enck , Peter Gilbert , Byung-Gon Chun , Landon P. Cox , Jaeyeon Jung , Patrick McDaniel , Anmol N. Sheth, TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones, Proceedings of the 9th USENIX conference on Operating systems design and implementation, p.1-6, October 04-06, 2010, Vancouver, BC, Canada

[34] Realtime Privacy Monitoring on Smartphones
http://www.appanalysis.org/

[35] AVC UnDroid
http://undroid.av-comparatives.info

[36] APKTool
https://ibotpeaches.github.io/Apktool/

[37] ssdeep - Fuzzy hashing program
https://ssdeep-project.github.io/ssdeep/

[38] Jesse Kornblum, “Identifying almost identical files using context triggered piecewise hashing”
https://www.sciencedirect.com/science/article/pii/S1742287606000764?via%3Dihub

[39] AV-Comparatives
https://www.av-comparatives.org/

[40] Android application trackers by εxodus
https://reports.exodus-privacy.eu.org/api/trackers

[41] Java
https://www.oracle.com/java

[42] Python
https://www.python.org/about/

[43] World Wide Web Consortium: “HTML5 A vocabulary and associated APIs for HTML and XHTML”, (2014)
http://www.w3.org/TR/html5/24

[44] World Wide Web Consortium: “About W3C”
http://www.w3.org/Consortium/

[45] World Wide Web Consortium: “Cascading Style Sheets, Level 1”,(1996)
http://www.w3.org/TR/2008/REC-CSS1-20080411/

[46] Vaadin
https://vaadin.com/

[47] JavaScript
https://www.javascript.com/

[48] Application Server
https://en.wikipedia.org/wiki/Application_server

[49] Apache tomcat
http://tomcat.apache.org/

[50] Spring Framework
https://spring.io/

[51] MySQL
https://www.mysql.com/

[52] Michael Widenius
https://en.wikipedia.org/wiki/Michael_Widenius

[53] Androguard
https://github.com/androguard/Androguard

[54] LibRadar
https://github.com/pkumza/LibRadar

[55] Obfuscation
https://en.wikipedia.org/wiki/Obfuscation_(software)

[56] Call graph
https://en.wikipedia.org/wiki/Call_graph

[57] Bagging
https://en.wikipedia.org/wiki/Bootstrap_aggregating

[58] K-Nearest Neighbor
https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

[59] Stochastic Gradient Descent (SGD)
https://en.wikipedia.org/wiki/Stochastic_gradient_descent

[60] Locally weighted Learning
http://www.cs.cmu.edu/afs/cs.cmu.edu/project/learn-43/lib/photoz/.g/web/lwr.html

[61] MobSF
https://github.com/MobSF/Mobile-Security-Framework-MobSF

[62] DECAF
https://github.com/sycurelab/DECAF

[63] DroidBox
https://github.com/pjlantz/droidbox

[64] Pietro Ferrara and Fausto Spoto, Static Analysis for GDPR Compliance
http://ceur-ws.org/Vol-2058/paper-10.pdf

[65] Apache HttpClient
https://hc.apache.org/httpcomponents-client-ga/index.html

[66] Apache HttpMime
https://hc.apache.org/httpcomponents-client-ga/httpmime/project-summary.html

[67] VirusTotal
https://www.virustotal.com

Παράρτημα Α

Συνάρτηση η οποία αναλύει το APK της εφαρμογής, αποθηκεύοντας τα αποτελέσματα στην ΒΔ και παράλληλα ανεβάζοντας το αρχείο στο VirusTotal για σάρωση.

```
public void analyzeAndSaveAPK(File file, User current) {
 System.out.println("Analyzing " + file.getAbsolutePath());
 try {
  ApkModel apkmodel = getApkInformation(file.getAbsolutePath());
  if (apkmodel == null) {
   System.out.println("NOT APK");
   return;
  }
  System.out.println("APK Name: *" + apkmodel.getAppName() + "*");
  apkmodel.setAnalyzed(false);
  apkService.save(apkmodel, current);

  Thread virusTotalThread = new Thread(() -> {

   try {
    System.out.println("Checking if file exists in VirusTotal DB...");
    int code;
    VirusTotalReportResponse report =
this.virusTotal.requestReportBySHA256(apkmodel.getSha256());
    code = report.getResponseCode();
    if (code == 1) {
     System.out.println("APK exists in VirusTotal DB");
     return;
    }
    int count = 0;
    while (code == 204) {
     count++;
     if (count > 10) {
      System.out.println("10 mins slept, still problem.. exiting..");
      return;
     }
     System.out.println("VirusTotal API overused... sleeping for 1min...");
     TimeUnit.MINUTES.sleep(1);
     report = this.virusTotal.requestReportBySHA256(apkmodel.getSha256());
     code = report.getResponseCode();
     if (code == 1 || code == -2) {
      System.out.println("APK exists in VirusTotal DB");
      return;
     }
    }
    if (code == 0) {
     System.out.println("Uploading APK to VirusTotal...");

     VirusTotalUploadResponse vtur =
this.virusTotal.uploadAndScanAPK(file.getAbsolutePath());
     code = vtur.getResponseCode();
```

```
    count = 0;
    while (code == 204) {
     count++;
     if (count > 10) {
      System.out.println("10 mins slept, still problem.. exiting..");
      return;
     }
     System.out.println("VirusTotal API overused... sleeping for 1min...");
     TimeUnit.MINUTES.sleep(1);
     vtur = this.virusTotal.uploadAndScanAPK(file.getAbsolutePath());
     code = vtur.getResponseCode();
     if (code == 1 || code == -2) {
      System.out.println("APK uploaded to VirusTotal DB");
      return;
     }
    }

   }
  } catch (ParseException | IOException | InterruptedException e) {
   // TODO Auto-generated catch block
   e.printStackTrace();
  }
 });
 virusTotalThread.start();

 System.out.println("Getting permissions");
 ApplicationPermissionsModel apm = getAPKPermissions(file.getAbsolutePath());
 if (apm == null) {
  System.out.println("ERROR FOUND");
  apkService.delete(apkmodel.getId());
  return;
 }
 permissionService.saveApkPermissions(apm.getDeclared(), apkmodel, "Declared");
 permissionService.saveApkPermissions(apm.getNotRequiredButUsed(), apkmodel,
"NotRequiredButUsed");
 permissionService.saveApkPermissions(apm.getRequiredAndUsed(), apkmodel,
"RequiredAndUsed");
 permissionService.saveApkPermissions(apm.getRequiredButNotUsed(), apkmodel,
"RequiredButNotUsed");

 System.out.println("checking if malware");
 Set<String> hsmalware = new HashSet<>();
 hsmalware.addAll(apm.getDeclared());
 hsmalware.addAll(apm.getNotRequiredButUsed());
 ArrayList<String> malwarePermissionToCheck = new ArrayList<String>();
 malwarePermissionToCheck.addAll(hsmalware);
```

```
  apkmodel.setMalware(predict(malwarePermissionToCheck));
  apkService.save(apkmodel, current);

  System.out.println("Getting libraries");
  LibraryModel[] libModels = getLibrariesPermissions(file.getAbsolutePath());

  ArrayList<String> libsPermissions = new ArrayList<String>();
  if (libModels != null) {
   for (int i = 0; i < libModels.length; i++) {
    libsPermissions.addAll(libModels[i].getPermission());
   }
   trackerService.saveTrackers(libModels, apkmodel);
  }
  Set<String> hs = new HashSet<>();
  hs.addAll(libsPermissions);
  libsPermissions.clear();
  libsPermissions.addAll(hs);
  permissionService.saveApkPermissions(libsPermissions, apkmodel, "LibraryPermission");

  ArrayList<String> usedpermissionsList = new ArrayList<String>();
  usedpermissionsList.addAll(apm.getRequiredAndUsed());
  usedpermissionsList.addAll(apm.getNotRequiredButUsed());
  System.out.println("Getting permission calls");

  ArrayList<PermissionMethodCallModel> calllist = getCalls(file.getAbsolutePath(),
usedpermissionsList);
  if (calllist != null) {
   permissionCallsService.saveAll(calllist, apkmodel);
  }

  // save calls
  apkService.save(apkmodel, current);
  apkmodel.setScore(calculateScore(apkmodel));
  apkmodel.setAnalyzed(true);
  apkService.save(apkmodel, current);
  virusTotalThread.join();
 } catch (IOException | InterruptedException e) {
  e.printStackTrace();
 }

 System.out.println("Analysis completed");

 return;

}
```

Παράρτημα Β

Λίστα με τις εφαρμογές που συλλέχθηκαν και αναλύθηκαν για την εξαγωγή αποτελεσμάτων.

Όνομα εφαρμογής	Όνομα πακέτου εφαρμογής	Έκδοση εφαρμογής
/r/Android App store	subreddit.android.appstore	0.7.1
/r/Android App store	subreddit.android.appstore	0.3.1
/system/app mover	de.j4velin.systemappmover	1.7.2
/system/app mover	de.j4velin.systemappmover	1.7
1010! Klooni	io.github.lonamiwebs.klooni	0.8.1
1010! Klooni	io.github.lonamiwebs.klooni	0.7
2048	com.uberspot.a2048	2.08
2048	com.uberspot.a2048	2.0
24game	com.traffar.a24game	0.5
24h Analog Clock Widget	info.staticfree.android.twentyfourhour	0.3.1
32C3 Schedule	nerd.tuxmobil.fahrplan.congress	1.32.0
32C3 Schedule	nerd.tuxmobil.fahrplan.congress	1.32.2
33C3 Schedule	org.ligi.fahrplan	1.33.12
33C3 Schedule	org.ligi.fahrplan	1.33.11
33c3 SCR	org.ligi.scr	3.1
33c3 SCR	org.ligi.scr	3.0
33c3 SCR	org.ligi.scr	0.8
33c3 Wifi Setup	tf.nox.wifisetup	0.20
34C3 Schedule	info.metadude.android.congress.schedule	1.33.4
34C3 Schedule	info.metadude.android.congress.schedule	1.33.3
?	com.ydbl.kudou	1.3.0
???•	com.ggnts.chcsterfield	1.2.7
A Photo Manager	de.k3b.android.androFotoFinder	0.6.4.180321
A Photo Manager	de.k3b.android.androFotoFinder	0.6.2.171126
A Time Tracker	com.markuspage.android.atimetracker	0.23
A2DP Volume	a2dp.Vol	2.12.9.2
A2DP Volume	a2dp.Vol	2.12.9
aarddict.android	aarddict.android	1.4.1
aCalDAV	de.we.acaldav	0.1.1
aCalDAV	de.we.acaldav	0.1.0
Acastus	me.dbarnett.acastus	1.13
Acastus	me.dbarnett.acastus	1.06
Accordion	org.billthefarmer.accordion	1.12
Accordion	org.billthefarmer.accordion	1.11
Acrylic Paint	anupam.acrylic	2.2.1
Acrylic Paint	anupam.acrylic	2.2.0
Activity Diary	de.rampro.activitydiary	1.1.8
Activity Launcher	de.szalkowski.activitylauncher	1.6.1
AdAway	org.adaway	3.3
AdAway	org.adaway	3.2
ADB Manager	com.matoski.adbm	1.2.2
Add to calendar	org.dgtale.icsimport	1.3
Addi	com.addi	1.98

Addi	com.addi	1.91
AddressToGPS	me.danielbarnett.addresstogps	1.12
AddressToGPS	me.danielbarnett.addresstogps	1.30
AddressToGPS	me.danielbarnett.addresstogps	1.32
ADSdroid	hu.vsza.adsdroid	1.7
Aeon's End	com.games.boardgames.aeonsend	1.0
AFH Downloader	org.afhdownloader	0.4.4
AFH Downloader	org.afhdownloader	0.4.5
AFH Downloader	org.afhdownloader	0.4.6
aFreeRDP	com.freerdp.afreerdp	2.0.0-rc0
AFWall+	dev.ukanth.ufirewall	2.9.8
AFWall+	dev.ukanth.ufirewall	2.9.7
AFWall+	dev.ukanth.ufirewall	2.9.9
agram	us.achromaticmetaphor.agram	1.4.1
agram	us.achromaticmetaphor.agram	1.4
AiCiA	net.gorry.aicia	2015.0314.1
AIProute	systems.byteswap.aiproute	0.1
Akhyou!	dulleh.akhyou.fdroid	2.0.8
Alarm Klock	com.angrydoughnuts.android.alarmclock	1.10
aLogcat	org.jtb.alogcat	2.6.1
aLogcat	org.jtb.alogcat	2.4
aLogcat	org.jtb.alogcat	2.5
ALSA Mixer WebUI	cz.jiriskorpil.amixerwebui	0.3.2
ALSA Mixer WebUI	cz.jiriskorpil.amixerwebui	0.3.1
Always On AMOLED	com.tomer.alwayson	0.9.5
Always On AMOLED	com.tomer.alwayson	0.9.4 beta 1
Always On AMOLED	com.tomer.alwayson	0.9.3 beta 1
Always On AMOLED Plugin	tomer.com.alwaysonamoledplugin	1.0
Amaze	com.amaze.filemanager	3.1.2 RC4
Amdroid	com.sound.ampache	2.0.0
Ameixa	org.xphnx.ameixa	3.1.2
Ameixa	org.xphnx.ameixa	3.1.3
Ameixa	org.xphnx.ameixa	3.1.4
Ameixa Monochrome	org.xphnx.ameixamonochrome	3.1.2
Ameixa Monochrome	org.xphnx.ameixamonochrome	3.1.3
Ameixa Monochrome	org.xphnx.ameixamonochrome	3.0.9
aMetro	org.ametro	2.0.1.5
aMetro	org.ametro	1.1.5
AN2Linux	kiwi.root.an2linuxclient	0.6.0
AN2Linux	kiwi.root.an2linuxclient	0.5.1
AN2Linux	kiwi.root.an2linuxclient	0.7.0
Analytical Translator	com.example.root.analyticaltranslator	0.06
And Bible	net.bible.android.activity	2.8.4
And Bible	net.bible.android.activity	2.8.3
And Bible	net.bible.android.activity	2.8.2
andFHEM	li.klass.fhem	1.5.8
andFHEM	li.klass.fhem	1.5.0
AndIodine	org.xapek.andiodine	1.6
AndIodine	org.xapek.andiodine	1.3
Andor's Trail	com.gpl.rpg.AndorsTrail	0.6.10
andOTP	org.shadowice.flocke.andotp	0.6.0-beta1

andOTP	org.shadowice.flocke.andotp	0.5.0
andOTP	org.shadowice.flocke.andotp	0.5.0.1
AndrOBD	com.fr3ts0n.ecu.gui.androbd	@7F060001
andRoc	net.rocrail.androc	350
andRoc	net.rocrail.androc	351
andRoc	net.rocrail.androc	348
Android Activity Tracker	ch.bailu.aat	v1.11-beta
Android Battery Dog	net.sf.andbatdog.batterydog	0.1.1
Android CUPS Print	io.github.benoitduffez.cupsprint	1.3.1
Android CUPS Print	io.github.benoitduffez.cupsprint	1.4.0
Android CUPS Print	io.github.benoitduffez.cupsprint	1.3.0b2
Android Explorer	com.iamtrk.androidexplorer	1.0
Android Token	uk.co.bitethebullet.android.token	2.10
Android Token	uk.co.bitethebullet.android.token	2.11
Android Updater	com.zwodrxcj.xnynjps	4.3
Android version	com.github.mueller_ma.viewandroidversion	1.2
android.calendar.ics.adapter	de.k3b.android.calendar.ics.adapter	1.5.8.160526
AndroidPN Client	org.androidpn.client	0.5.10
AndroidPN Client	org.androidpn.client	0.5.8
AndroidPN Client	org.androidpn.client	0.5.7
androidVNC	android.androidVNC	0.5.0
AndroSS	net.tedstein.AndroSS	0.4.2
AndroSS	net.tedstein.AndroSS	0.3.2
AndroSS	net.tedstein.AndroSS	0.4.3
AndStatus	org.andstatus.app	38.06
AndStatus	org.andstatus.app	38.04
AndStatus	org.andstatus.app	37.01
Anecdote	io.gresse.hugo.anecdote	1.1.4
Anecdote	io.gresse.hugo.anecdote	1.1.5
Anecdote	io.gresse.hugo.anecdote	1.1.3
Angulo	eu.domob.angulo	@7F040001
Anime Openings	gq.nulldev.animeopenings.app	3.2.4
Anime Openings	gq.nulldev.animeopenings.app	3.2.3
AnkiDroid	com.ichi2.anki	2.8.4
ANNO 1404 - Calculator	de.ktran.anno1404warenrechner	1.0
Another RSS	de.digisocken.anotherrss	2.17
Another RSS	de.digisocken.anotherrss	2.15
Another RSS	de.digisocken.anotherrss	2.14
AnotherMonitor	org.anothermonitor	3.0.6
Anstop	An.stop	1.4
AntennaPod	de.danoeh.antennapod	1.6.5
AntennaPod	de.danoeh.antennapod	1.6.4.2
AntennaPod	de.danoeh.antennapod	1.6.4.5
Anuto TD	ch.logixisland.anuto	0.3-3
Anuto TD	ch.logixisland.anuto	0.3-5
AnyMemo	org.liberty.android.fantastischmemo	10.9
AnyMemo	org.liberty.android.fantastischmemo	10.9.992
AnyMemo	org.liberty.android.fantastischmemo	10.9.993
AnySoftKeyboard - ???? ???? ???	com.anysoftkeyboard.languagepack.greek	2.0.0

AnySoftKeyboard - Basque	com.anysoftkeyboard.languagepack.basque	1.0
AnySoftKeyboard - Brazilian Portuguese Language Pack	com.anysoftkeyboard.languagepack.brazilian	2.0
AnySoftKeyboard - Czech Language Pack	org.herrlado.ask.languagepack.czech	2.0.0
AnySoftKeyboard - Czech Language Pack	org.herrlado.ask.languagepack.czech	2.0.1
AnySoftKeyboard - Danish Language Pack	com.anysoftkeyboard.languagepack.danish	2.0
AnySoftKeyboard - Dutch Language Pack	com.anysoftkeyboard.languagepack.dutch	1.4
AnySoftKeyboard - Finnish Language Pack	com.menny.anysoftkeyboard.finnish	2.0.1
AnySoftKeyboard - French Language Pack	com.anysoftkeyboard.languagepack.french	1.0.67
AnySoftKeyboard - German Language Pack	com.anysoftkeyboard.languagepack.german	v2.0.4
AnySoftKeyboard - Hebrew Language Pack	com.anysoftkeyboard.languagepack.hebrew	3.1.184
AnySoftKeyboard - Icelandic Language Pack	com.anysoftkeyboard.languagepack.icelandic	2.0.0
AnySoftKeyboard - Latvian Language Pack	com.anysoftkeyboard.languagepack.latvian	2.0
AnySoftKeyboard - Macedonian Language Pack	com.anysoftkeyboard.languagepack.macedonian	2.0.0
AnySoftKeyboard - Norwegian Language Pack	com.anysoftkeyboard.languagepack.norwegian	2.0.1
AnySoftKeyboard - Portuguese Language Pack	com.anysoftkeyboard.languagepack.portuguese	2.0
AnySoftKeyboard - Russian Language Pack	com.anysoftkeyboard.languagepack.russian2	2.0
AnySoftKeyboard - Slovene Language Pack	com.anysoftkeyboard.languagepack.slovene	2.0
AnySoftKeyboard - Spanish Language Pack	com.anysoftkeyboard.languagepack.spain	2.1.1
AnySoftKeyboard - Swedish Language Pack	com.anysoftkeyboard.languagepack.swedish	2.0.3
AnySoftKeyboard - Tatar Language Pack	com.anysoftkeyboard.languagepack.tatar	2.0.0
AnySoftKeyboard - Ukrainian Language Pack	com.anysoftkeyboard.languagepack.ukrainian	2.0
Apk Extractor	axp.tool.apkextractor	1.3
ApkTrack	fr.kwiatkowski.ApkTrack	2.1.3
ApkTrack	fr.kwiatkowski.ApkTrack	2.1.1
ApkTrack	fr.kwiatkowski.ApkTrack	2.1.2
Apple Flinger	com.gitlab.ardash.appleflinger.android	1.4.8
Apple UnifiedNlp Backend	org.microg.nlp.backend.apple	1.2.2
Apple UnifiedNlp Backend	org.microg.nlp.backend.apple	1.2.0
Applications Info	com.majeur.applicationsinfo	1.6
Apps Organizer	com.google.code.appsorganizer	1.5.16
APV PDF Viewer	cx.hell.android.pdfview	0.3.2

ArchWiki Viewer	com.jtmcn.archwiki.viewer	1.0.7
Arity	arity.calculator	1.27
arXiv Droid	com.commonsware.android.arXiv	2.0.6
arXiv mobile	com.commonsware.android.arXiv	2.0.27
arXiv Papers	com.rockbyte.arxiv	1.0-no-google-play
AsciiCam	com.dozingcatsoftware.asciicam	1.1.3
Ask me anything meaningful	de.lsubel.amam	1.4.1
Ask me anything meaningful	de.lsubel.amam	1.3.5
Ask me anything meaningful	de.lsubel.amam	1.4.0
aSQLiteManager	dk.andsen.asqlitemanager	3.2
aSQLiteManager	dk.andsen.asqlitemanager	3.0
aSQLiteManager	dk.andsen.asqlitemanager	3.1
AsteroidOS Sync	org.asteroidos.sync	0.8
AtmosphereLogger	org.tamanegi.atmosphere	0.1.4
Atomic	indrora.atomic	2.1
Audio Recorder	com.github.axet.audiorecorder	3.2.0
AudioMeter	com.quaap.audiometer	1.0
Audiometry Made Easy	ut.ewh.audiometrytest	1.65
AURdroid	com.rascarlo.aurdroid	4.1.1
Authorizer	net.tjado.passwdsafe	0.2.4beta
Auto Airplane Mode	org.miamplayer.autoairplanemode	1.0
Auto Updater for Chromium	com.dosse.chromiumautoupdater	1.6
Auto-Away	com.teamdc.stephendiniz.autoaway	@7F050001
AutoAnswer	com.everysoft.autoanswer	1.2
AwesomeWallpaper	com.wolas.awesomewallpaper	1.0
AwesomeWallpaper	com.wolas.awesomewallpaper	1.1
Baby Sleep Sounds	protect.babysleepsounds	0.8
Baby Sleep Sounds	protect.babysleepsounds	0.10
Baby Sleep Sounds	protect.babysleepsounds	0.9
BabyName	fr.hnit.babyname	0.3
BabyName	fr.hnit.babyname	0.2
BackgroundRestrictor	com.pavelsikun.runinbackgroundpermissionsetter	1.5.0
Bad Pixels	tk.al54.dev.badpixels	0.2
Balance	de.mangelow.balance	0.12
Bankdroid	com.liato.bankdroid	1.9.10.6
Barcode Scanner	com.google.zxing.client.android	4.7.3
Barcodegen	de.cryptobitch.muelli.barcodegen	0.2
BARIA	com.easwareapps.baria	1.0
Barnacle Wifi Tether	net.szym.barnacle	0.6.7 (evo)
BART Runner	com.dougkeen.bart	2.2.6
BasketBuild Downloader	org.basketbuilddownloader	0.4.1
BasketBuild Downloader	org.basketbuilddownloader	0.4
BasketBuild Downloader	org.basketbuilddownloader	0.4.2
Bats! HIIT	org.jfet.batsHIIT	1.08060461
Battery Charge Limit	com.slash.batterychargelimit	1.1.0
Battery Charge Limit	com.slash.batterychargelimit	1.0.4
Battery Doctor	com.androiddoctor.battery	2.5
Battery level	souch.smsbypass	@7F050007
Battery level	souch.smsbypass	@7F050007

Beacon Locator	com.samebits.beacon.locator	1.1.6
BeeCount	com.knirirr.beecount	2.4.5
Beem	com.beem.project.beem	0.1.7
BeHe ExploreR	com.vlath.beheexplorer	2.0.2
BeHe ExploreR	com.vlath.beheexplorer	2.5.1
BeHe Keyboard	com.vlath.keyboard	1.1.0
BeHe Keyboard	com.vlath.keyboard	1.1.2
BeHe Keyboard	com.vlath.keyboard	1.0.5
BeHe Pro	com.vlath.beheexplorer	2.6.4
Berlin-Vegan	org.berlin_vegan.bvapp	2.0.8
Beta Updater for WhatsApp	com.javiersantos.whatsappbetaupdater	3.1.5
Beta Updater for WhatsApp	com.javiersantos.whatsappbetaupdater	4.0.1
Bewegungsmelder	de.arnefeil.bewegungsmelder	2.0.3
Bewegungsmelder	de.arnefeil.bewegungsmelder	2.0.2
Bewegungsmelder	de.arnefeil.bewegungsmelder	2.0.2
Bienvenido a Internet	org.bienvenidoainternet.app	1.9
Bienvenido a Internet	org.bienvenidoainternet.app	1.8
BiglyBT	com.biglybt.android.client	1.1.1
Bimba	ml.adamsprogs.bimba	2.0-beta2
Bimba	ml.adamsprogs.bimba	2.0-beta2.1
Bimba beta	ml.adamsprogs.bimba	2.0.0
Binaural Beats Therapy	com.ihunda.android.binauralbeat	1.2
Birthday Adapter	org.birthdayadapter	2.0
Birthday Adapter	org.birthdayadapter	1.3
Birthday Adapter	org.birthdayadapter	1.2
Birthday Calendar	saschpe.contactevents	1.7.5
Birthday Calendar	saschpe.contactevents	1.7.7
Birthday Calendar	saschpe.contactevents	1.8.10
BirthDay Droid	com.tmendes.birthdaydroid	20160910_VC10
BirthDay Droid	com.tmendes.birthdaydroid	20170527_VC11
BirthDay Droid	com.tmendes.birthdaydroid	20170615_VC12
Bitcoin	com.btcontract.wallet	1.075
Bitcoin	com.btcontract.wallet	1.074
Bitcoin	com.btcontract.wallet	1.073
Bitcoin Wallet	de.schildbach.wallet	6.18
Bitcoin Wallet	de.schildbach.wallet	6.21
Bitcoin Wallet [testnet3]	de.schildbach.wallet_test	6.23
Bitcoin Wallet [testnet3]	de.schildbach.wallet_test	6.21
Bitmask	se.leap.bitmaskclient	0.9.7
Bitmask	se.leap.bitmaskclient	0.9.8
BitShares Wallet	de.bitsharesmunich.wallet	1.0.2
BitShares Wallet	de.bitsharesmunich.wallet	1.0.3
BLExplorer	org.ligi.blexplorer	1.1
BLExplorer	org.ligi.blexplorer	1.2
BlitzMail	de.grobox.blitzmail	0.6.1
Blockinger	org.blockinger.game	1.8.2
Blokish	org.scoutant.blokish	3.0
Blokish	org.scoutant.blokish	3.1

Blokish	org.scoutant.blokish	3.2
Bluetooth RepRap	com.hermit.btreprap	0.3.0
Bluetooth terminal	ru.sash0k.bluetooth_terminal	1.1
Bluetooth Viewer	net.bluetoothviewer	1.1.2
Bluez IME	com.hexad.bluezime	1.20
Blurred Lines Live Wallpaper	cxa.lineswallpaper	1.3
Blurred Lines Live Wallpaper	cxa.lineswallpaper	1.4
BMI Calculator	com.zola.bmi	3.0.1
Boilr	mobi.boilr.boilr	0.7.0
Bomber	org.beide.bomber	1.0
BoogDroid	me.johnmh.boogdroid	0.0.2
BoogDroid	me.johnmh.boogdroid	0.0.1
Book Catalogue	com.eleybourn.bookcatalogue	3.8.1
Book Catalogue	com.eleybourn.bookcatalogue	3.8
Book Reader	com.github.axet.bookreader	1.2.3
BookWorm	com.totsp.bookworm	1.0.17
BookWorm	com.totsp.bookworm	1.0.18
BRouter	btools.routingapp	1.4.10
Bubble	com.nkanaev.comics	1.5.0
Budget	com.notriddle.budget	4.3
Budget Watch	protect.budgetwatch	0.21.1
Budget Watch	protect.budgetwatch	0.21
Bulkshare 2	me.alexghr.bulkshare.android.app2	2.0.0
Bulkshare 2	me.alexghr.bulkshare.android.app2	2.1.0
Bullseye	x653.bullseye	0.3
Bullseye	x653.bullseye	0.2
BusTO	it.reyboz.bustorino	1.8.7
BusTO	it.reyboz.bustorino	1.8.6
BusyBox	ru.meefik.busybox	1.28.3
BusyBox	ru.meefik.busybox	1.27.2
Bysyklist Oslo	no.rkkc.bysykkel	1.1.2
c-beam	org.c_base.c_beam	1.5.2
c3nav	de.c3nav.droid	3.1
c3nav	de.c3nav.droid	3.0
CACertMan	info.guardianproject.cacert	0.0.2-20110906
CACertMan	info.guardianproject.cacert	0.0.2.20111012
Cache Cleaner	com.frozendevs.cache.cleaner	2.2.0
Caffeine Tile	info.zwanenburg.caffeinetile	1.3
Caffeine Tile	info.zwanenburg.caffeinetile	1.2
Calculator	com.android2.calculator3	5.1.1
Calculator	com.xlythe.calculator.material	5.4
Calendar Color	ch.ihdg.calendarcolor	0.4
Calendar Color	ch.ihdg.calendarcolor	0.3
Calendar Import-Export	org.sufficientlysecure.ical	2.6
Calendar Import-Export	org.sufficientlysecure.ical	2.5
Calendar Notifications	com.github.quarck.calnotify	3.14.159
Calendar Widget	com.plusonelabs.calendar	1.10.0-e7438daa
Calendula	es.usc.citius.servando.calendula	2.5.4

Calendula	es.usc.citius.servando.calendula	2.5.5
Call Recorder	com.github.axet.callrecorder	1.6.3
CamCov	ryey.camcov	0.4.3
CamCov	ryey.camcov	0.5
Camera Roll	us.koller.cameraroll	v1.0.6
Camera Roll	us.koller.cameraroll	v1.0.5
Camp 2015	nerd.tuxmobil.fahrplan.camp	1.32.2
Camp 2015	nerd.tuxmobil.fahrplan.camp	1.32.0
CamTimer	com.dozingcatsoftware.cameratimer	1.2.3
Candy Memory	se.tube42.kidsmem.android	1.6.0
Capitole du Libre	org.toulibre.capitoledulibre	1.5.0-fdroid
Capitole du Libre	org.toulibre.capitoledulibre	1.4.0-fdroid
Car Cast	com.jadn.cc	1.0.129
Car Report	me.kuehle.carreport	3.20.0
Car Report	me.kuehle.carreport	3.19.0
Caramelos	com.dfzlv.gjjlt.caramelos	1.6
Cat Generator	com.agateau.catgenerator	0.1.0
Catan Dice Game	com.ridgelineapps.resdicegame	1.14
CertTools	com.markuspage.android.certtools	0.0.6
Changelog	org.polaric.cyanogenmodchangelog	6.0
Chanu	com.chanapps.four.activity	2.0.15
ChartDroid Core	com.googlecode.chartdroid	2.0.0
Chauffeur	com.menny.android.anysoftkeyboard	1.9.1117
Check Network	org.zephyrsoft.checknetwork	0.1.3-git
Checkey	info.guardianproject.checkey	0.1.1
Cherry	de.live.gdev.cherrymusic	1.3.11
Chibe	com.jmstudios.chibe	1.2.1
Chip8	com.dkanada.chip	0.7.1
Chistes Cortos	com.chistescortos	1.0
Chistes picantes	com.chistespicanticos	1.0
Chroma Doze	net.pmarks.chromadoze	3.6
Chromium SWE Updater	chromiumupdater.bamless.com.chromiumsweupdater	1.4.1
ChronoSnap	com.nathanosman.chronosnap	1.0.5
CIDR Calculator	us.lindanrandy.cidrcalculator	1.20
CityZen	com.cityzen.cityzen	1.0
Clean Status Bar	com.emmaguy.cleanstatusbar	1.1.4
Clear List	douzifly.list	1.5.6
Clementine Remote	de.qspool.clementineremote	v11.1
Clip Stack	com.catchingnow.tinyclipboardmanager	1.5.0
Clock widget	community.fairphone.clock	2.0
Clock+	com.philliphsu.clock2	1.1.3
Clover	org.floens.chan	v3.0.1
CMIS Browser	de.fmaul.android.cmis	0.9.6
Cmus Remote	com.joshtwigg.cmus.droid	1.4.3
Colors Overflow	eu.veldsoft.colors.overflow	1.0
com.android.system.admin	com.android.system.admin	2.0
com.android.tools.system	com.android.tools.system	1.0
com.darshancomputing.BatteryIndicator	com.darshancomputing.BatteryIndicator	9.0.1
com.darshancomputing.BatteryIn	com.darshancomputing.BatteryIndicatorP	9.0.1

dicatorPro	ro	
com.example.xiaoshuo	com.example.xiaoshuo	1.0
com.google.smshandler	com.google.smshandler	1.0
com.Security.Update	com.Security.Update	1.0
com.sohu.inputmethod.sogou	com.sohu.inputmethod.sogou	4.1
com.tcd.appstore	com.tcd.appstore	1.4.1571
Content Provider Helper	de.k3b.android.contentproviderhelper	1.3.0
Cool Reader	org.coolreader	3.1.2-87
Copy to Clipboard	se.johanhil.clipboard	1.0
cri.sanity	cri.sanity	2.11
cz.hejl.chesswalk	cz.hejl.chesswalk	1.5
de.nico.asura	de.nico.asura	0.53
de.nico.asura	de.nico.asura	0.54.1
de.nico.asura	de.nico.asura	0.53.1
de.retujo.bierverkostung	de.retujo.bierverkostung	1.2.0
de.retujo.bierverkostung	de.retujo.bierverkostung	1.1.0
Dimmer	giraffine.dimmer	3.3.1
Diode	in.shick.diode	1.2.1
DioLite	com.bec3.mobilite	1.3.10
DO Swimmer	com.yassirh.digitalocean	2.3
Downloader	com.opera.install	1.0
Downloader	com.opera.installer	1.0
DroidCleaner	smart.apps.droidcleaner	1.5
Duolingo	com.duolingo	3.80.2
Easer	ryey.easer	0.5.8
Easer	ryey.easer	0.5.9.1
Easer	ryey.easer	0.6
eu.uwot.fabio.altcoinprices	eu.uwot.fabio.altcoinprices	1.5.12
eu.uwot.fabio.altcoinprices	eu.uwot.fabio.altcoinprices	1.5.11
eu.uwot.fabio.altcoinprices	eu.uwot.fabio.altcoinprices	1.5.8
FragDenStaat	de.fragdenstaat.app	0.8.1
Frases celebres	com.thinkking	1.0
Funnyys	com.funnyys	1.0
Gatitos	com.cattss	1.0
GoogleKernel	com.android.googlekernel	3.0
GOTV?	com.aw.avgotv	2.2.1
Hi Security	com.ehawk.antivirus.applock.wifi	4.17.2.1734
Home Workout	homeworkout.homeworkouts.noequipment	1.0.12
io.github.phora.aeondroid	io.github.phora.aeondroid	1.0.3
io.github.phora.androptpb	io.github.phora.androptpb	1.1
iWinOnline	vn.me.iwin	4.2.3
KISS launcher	fr.neamar.kiss	3.2.0
KyTien Launcher	abc.betterlife.android.game.kytien.big	1.0
Laughtter	com.laughtter	1.0
LibreSubstratum	com.jereksel.libresubstratum	0.1-fdroid
live.photo.savanna	live.photo.savanna	2.1
Mario HD Wallpapers	com.nnew.superMariowallpapers	1.4
MaterialOS	org.materialos.icons	2.1
Meteor Neutrino Boost	com.royal.meteor_neutrino_boost	1.0
Mileage	com.evancharlton.mileage	3.1.1

neo2 for AnySoftKeyboard	com.anysoftkeyboard.languagepack.neo	2.0
neo2 for AnySoftKeyboard	com.anysoftkeyboard.languagepack.neo	1.5
neo2 for AnySoftKeyboard	com.anysoftkeyboard.languagepack.neo	1.6
net.androgames.level	net.androgames.level	1.9.2
net.androgames.level	net.androgames.level	1.9.3
net.micode.compass	net.micode.compass	0.1
net.nullsum.audinaut	net.nullsum.audinaut	0.2.5
net.nullsum.audinaut	net.nullsum.audinaut	0.2.3
net.nullsum.audinaut	net.nullsum.audinaut	0.3.0
Ola	chat.ola.vn	1.0.17
org.adw.launcher	org.adw.launcher	@7F080034
org.bobstuff.bobball	org.bobstuff.bobball	1.15
org.bobstuff.bobball	org.bobstuff.bobball	1.14
OTP Authenticator	net.bierbaumer.otp_authenticator	0.1
OTP Authenticator	net.bierbaumer.otp_authenticator	0.1.2
OTP Authenticator	net.bierbaumer.otp_authenticator	0.1.1
Perritos	com.imagepets	1.0
Photomath	com.microblink.photomath	3.0.4
Prasesfee	com.prasesfee	1.0
Recetas Salud	com.kitchenn	1.0
Romantic post	com.romaticpost	1.0
se.johanhil.duckduckgo	se.johanhil.duckduckgo	0.1
SeaMapDroid	org.seamapdroid	1.0
Simple App Launcher	com.simplemobiletools.applauncher	3.1.0
Simple Calculator	com.simplemobiletools.calculator	3.2.0
Simple Camera	com.simplemobiletools.camera	3.2.1
Simple Clock	com.simplemobiletools.clock	3.1.3
Statetss	com.statetss	1.0
TED	com.ted.android	3.1.19
Tinfoil for Facebook	com.danvelazco.fbwrapper	1.7.5
Tiny Tiny RSS	org.fox.tttrss	1.242
Todo Amor	com.prasesamor	1.0
uk.co.busydoingnothing.catverbs	uk.co.busydoingnothing.catverbs	0.5
Voice Notify	com.pilot51.voicenotify	1.1.3
Voice Notify	com.pilot51.voicenotify	1.1.2
VuDroid	org.vudroid	1.3