
**ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ**

**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ**

**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

**ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ**


**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

# Ασφάλεια Κινητών Συσκευών


**Ιωάννης Α. Μονογιούδης**

**Επιβλέποντες:** **Γεωργιάδης Παναγιώτης,** Καθηγητής

**Κωνσταντίνος Παπαπαναγιώτου,** Διδάκτωρ



**ΑΘΗΝΑ**

**Σεπτέμβριος 2013**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

# Ασφάλεια Κινητών Συσκευών

.

**Ιωάννης Α. Μονογιούδης**

Α.Μ.: Μ1202

**ΕΠΙΒΛΕΠΟΝΤΕΣ:** **Γεωργιάδης Παναγιώτης,** Καθηγητής

**Κωνσταντίνος Παπαπαναγιώτου,** Διδάκτωρ

Σεπτέμβριος 2013

# ΠΕΡΙΛΗΨΗ

Στην παρούσα διπλωματική γίνεται παρουσίαση των χαρακτηριστικών ασφαλείας των κινητών συσκευών και αναλυτική παρουσίαση των λειτουργικών συστημάτων κινητών συσκευών iOS και Android, τα μοντέλα ασφαλείας που υποστηρίζουν, καθώς επίσης και οι σημαντικότεροι τύποι επιθέσεων που έχουν εμφανιστεί ως σήμερα.

Τέλος γίνεται έλεγχος ασφαλείας εφαρμογών Android οι οποίες είναι διαθέσιμες από διάφορους διαδικτυακούς τόπους και εξάγονται χρήσιμα συμπεράσματα σε θέματα ασφαλείας.

**ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ**: Ασφάλεια Κινητών Συσκευών

**ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ**: Λειτουργικά συστήματα συσκευών, Μοντέλο Ασφαλείας, Τύποι επιθέσεων, Μέθοδοι Προστασίας, Έλεγχοι Ασφαλείας Εφαρμογών

## ABSTRACT

In this thesis we present the security features of mobile devices and analytical presentation of operating systems for mobile devices iOS and Android, the security models that support this, as well as the major types of attacks that have occurred so far.

Finally a penetration test was made for Android applications which are free on the internet and some very usefull security conclutions are extreacted



**SUBJECT AREA:** Mobile Security

**KEYWORDS:** Mobile Devices Operating Systems, Security Models, Attack Vectors, Protection Methods, Application Penetration Test

**ΕΥΧΑΡΙΣΤΙΕΣ**

Η παρούσα διπλωματική εργασία συντάχθηκε στα πλαίσια του μεταπτυχιακού κύκλου σπουδών του τμήματος Πληροφορικής και Τηλεπικοινωνιών του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών. Για την εκπόνησή της θα ήθελα να ευχαριστήσω θερμά τον Διδάκτωρ ΕΚΠΑ Κωνσταντινο Παπαπαναγιώτου, για το χρόνο που διέθεσε, τις πολύτιμες συμβουλές, παρατηρήσεις και διορθώσεις του, συμβάλλοντας σημαντικά στην ολοκλήρωση του συγκεκριμένου πονήματος.

# ΠΕΡΙΕΧΟΜΕΝΑ

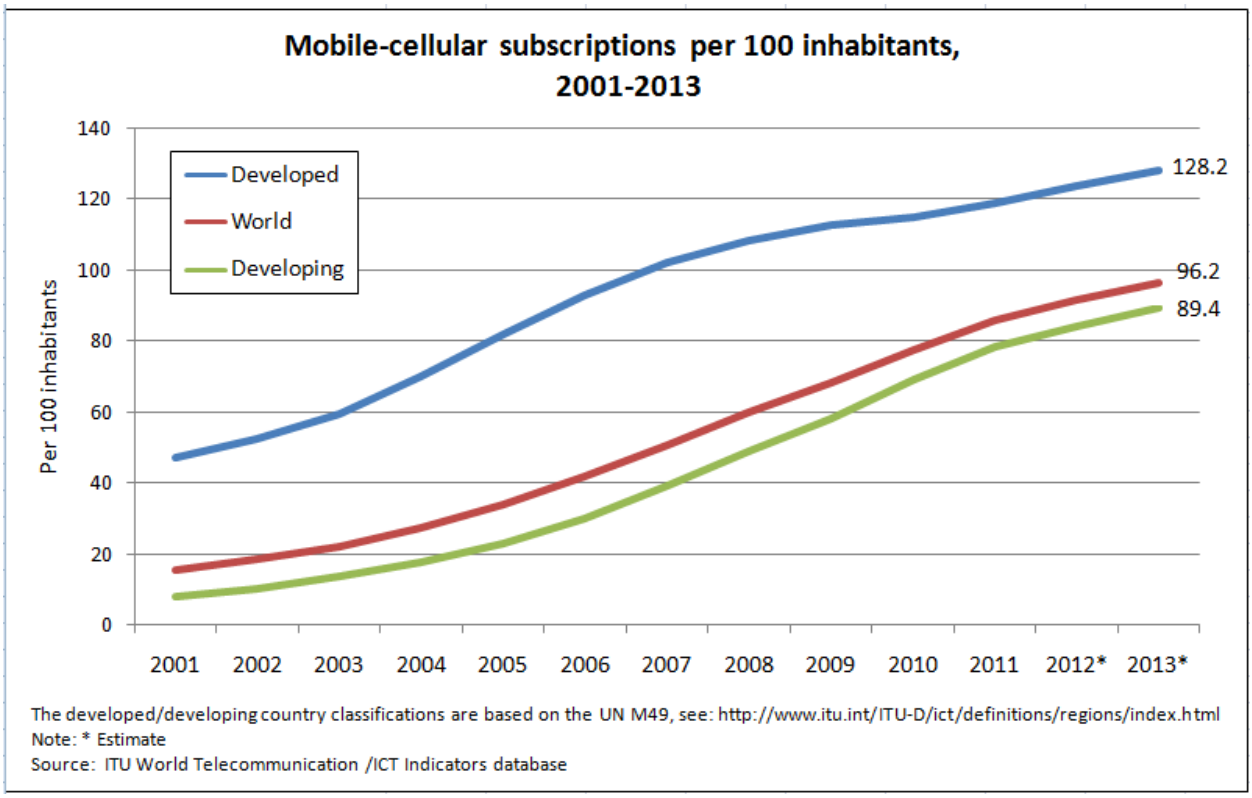
## ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

## ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

# ΚΕΦΑΛΑΙΟ 1

## 1. Εισαγωγή

Η είσοδος των smartphones και tablets στην αγορά από το 2001 και μετά, οδήγησε σε ραγδαία ανάπτυξη λειτουργικών συστημάτων για φορητές συσκευές λόγω της τεράστιας δημοφιλίας τους στο αγοραστικό κοινό.

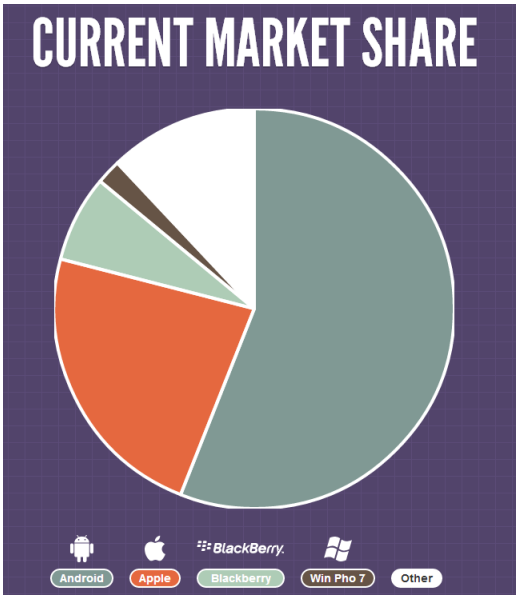



**Εικόνα 1 [36]**

Η χρήση της κινητής τηλεφωνίας παρουσιάζει αύξηση (εικ. 1) κάθε χρόνο και σε συνδυασμό με τις δυνατότητες των smartphones – tablets για περιήγηση στο διαδίκτυο είναι σχεδόν βέβαιο ότι οδηγούμαστε από την κοινωνία του σταθερού internet στην κοινωνία του ασύρματου – φορητού διαδικτύου. Μην ξεχνάμε ότι ένα smartphone της σημερινής εποχής αντιστοιχεί σε τεχνικά χαρακτηριστικά, με έναν προσωπικό υπολογιστή του 2000, αλλά έχει πολύ μικρότερο μέγεθος, είναι φορητό και με πολύ καλύτερη σχεδίαση – εργονομία.

Άμεσο αποτέλεσμα της ραγδαίας αυτής εξάπλωσης της χρήσης κινητών συσκευών, ήταν η αντίστοιχη αύξηση παραγωγής εφαρμογών που απευθύνονται στον μέσο χρήστη και είναι άμεσα διαθέσιμες μέσα από τα αντίστοιχα ηλεκτρονικά καταστήματα των

δημοφιλών πλατφόρμων (Apple store, Android market κλπ), με δημοφιλέστερες εξ αυτών την Android OS και την iOS της Apple.

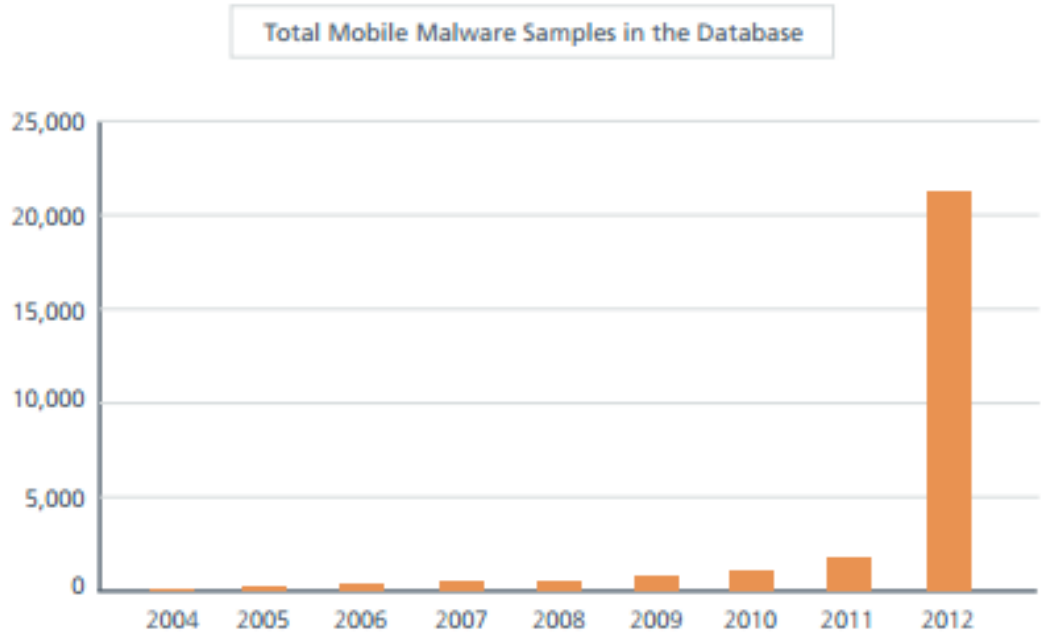



**Εικόνα 2 [36]**

Αν και οι συσκευές που τρέχουν σε Android περιβάλλον κατέχουν το μεγαλύτερο μέρος της αγοράς, οι διαθέσιμες εφαρμογές σε iOS κυριαρχούν και βρίσκονται στην πρώτη θέση των συνολικών download εφαρμογών για mobile συσκευές.


**Εικόνα 3 [36]**

Ταυτόχρονα με την άνθιση των mobile συσκευών, ως άμεσο επακόλουθο ήρθε και η αύξηση των malware, θέτωντας νέα δεδομένα στην ασφάλεια των εφαρμογών. Χαρακτηριστικές είναι οι εκθέσεις της Mcafee για το 2012 [25] [26] [27] [28], στις οποίες φαίνεται η ραγδαία εξάπλωση των malware εφαρμογών στα smartphones και tablets της αγοράς, όπου η αύξηση σε σχέση με το 2011 είναι της τάξης του 2.000%.




**Εικόνα 4 [27]**

Όπως γίνεται κατανοητό, η έννοια της ασφάλειας των κινητών συσκευών, είναι σχετικά καινούρια, και αποτελεί το νέο μεγάλο ερευνητικό πεδίο της επιστημονικής κοινότητας της πληροφορικής.

## *1.2 Ορισμοί*

**Κινητή συσκευή :** Ως 'Κινητή συσκευή' μπορούμε να ορίσουμε κάθε συσκευή η οποία περιέχει μια smartcard η οποία ελέγχεται και είναι ιδιοκτησία των παρόχων κινητής τηλεφωνίας. Πρέπει να τονιστεί ότι τα laptops τα οποία χρησιμοποιούν το ασύρματο δίκτυο κινητής τηλεφωνίας, μέσω USB sticks, για την παροχή υπηρεσιών internet, δεν θα εξεταστούν ως προς την ασφάλεια καθώς τα χαρακτηριστικά τους διαφέρουν από αυτά των smartphones και tablets. Επιπλέον, κινητά τηλέφωνα παλαιότερης τεχνολογίας τα οποία παρείχαν στοιχειώδης υπηρεσίες διαδικτύου, διαφέρουν σημαντικά ως προς το είδος των επιθέσεων και δεν πρόκειται να εξεταστούν και αυτά.

**Smartphone-tablet :** Ο διαχωρισμός των δύο αυτών συσκευών τείνει να εξαλειφθεί καθώς όλο και περισσότερα smartphones πλησιάζουν σε εμφάνιση – λειτουργικότητες τα tablets και το αντίστροφο. Γενικά μοπρούμε να ορίσουμε ως smartphone κάθε συσκευή η οποία περιέχει μια MNO smartcard με την οποία έχει την δυνατότητα σύνδεσης με ένα ασύρματο δίκτυο. Επιπλέον διαθέτει ένα open source

λειτουργικό σύστημα στο οποίο μπορούν να εγκατασταθούν εφαρμογές από τρίτες οντότητες.

**Feature Phone:** Ως feature phones καλούνται τα κινητά τηλέφωνα τα οποία διαθέτουν ‘κλειστά’ λειτουργικά συστήματα με προεγκατεστημένα προγράμματα και δεν επιτρέπουν την εγκατάσταση νέων, από τρίτες οντότητες.

## *1.3 Χαρακτηριστικά Ασφαλείας Κινητών Συσκευών*

Τα χαρακτηριστικά των κινητών συσκευών διαφέρουν από τα αντίστοιχα των σταθερών υπολογιστών διότι τα τεχνικά χαρακτηριστικά, οι λειτουργικότητες, οι τύποι των επιθέσεων, οι επιπτώσεις που αυτές προκαλούν, καθώς και τα προσωπικά δεδομένα των χρηστών που είναι αποθηκευμένα σε αυτά, διαφέρουν σημαντικά.

α. **Κόστος Επίθεσης**

Η δημιουργία κάποιου πρόσθετου κόστους στον χρήστη, αποτελεί παλαιό φαινόμενο των υπολογιστών, καθώς οι επιθέσεις τέτοιου τύπου οι οποίες απέφεραν κέρδη στους επιτιθέμενους, ανάγονται πίσω στην εποχή των dial-up συνδέσεων στο internet. Οι επιτιθέμενοι, ήταν σε θέση με κατάλληλο malware (γνωστά ως dialers) να καλούν αριθμούς με εξαιρετικά μεγάλη χρέωση, με αποτέλεσμα οι χρήστες να χρεώνονται υπέρογκα ποσά και οι επιτιθέμενοι να αποκτούν σημαντικά κέρδη.

Στους σταθερούς υπολογιστές το παραπάνω πρόβλημα έπαψε να υφίσταται με την έλευση των broadband συνδέσεων (DSL) καθώς οι υπολογιστές πλέον συνδέονται αυτόματα σε ένα δίκτυο, χωρίς την διαμεσολάβηση του σταθερού τηλεφωνικού δικτύου.

Στις κινητές συσκευές όμως, το εν λόγω πρόβλημα χρέωσης, θα υφίσταται για αρκετό καιρό ακόμη καθώς η δυνατότητα χρέωσης εξαιρετικά κοστοβόρων υπηρεσιών εν αγνοία του χρήστη είναι πάντα υπαρκτή.

**Υπηρεσίες χρεώσεως κινητών συσκευών**

Στις κινητές επικοινωνίες υπάρχουν 2 υπηρεσίες χρεώσεως. Ο 1$^{ος}$ τύπος χρεώσεων, χρησιμοποιεί την υπηρεσία ανταλλαγής μηνυμάτων για την μετάδοση πληροφοριών πιστοποίησης-αυθεντικοποίησης πχ. Online banking - online payment υπηρεσίες. Για την επίτευξη των υπηρεσιών αυτών, γενικά απαιτείται η αυθεντικοποίηση και των 2 καναλιών επικοινωνίας (χρήστη-τράπεζας και τράπεζας-χρήστη), όμως στις κινητές επικοινωνίες, η κινητή συσκευή είναι το μόνο κανάλι το οποίο θα πρέπει να αυθεντικοποιηθεί, αν ένας επιτιθέμενος έχει ήδη αποκτήσει τα στοιχεία αυθεντικοποίησης

του παρόχου της υπηρεσίας. Έτσι, ένα πιθανό malware μπορεί να κατευθύνει τα απαραίτητα μηνύματα στον επιτιθέμενο ή ακόμα να απαντά με έγκυρα μηνύματα σε αυτόν.

Ο 2ος τύπος χρεώσεων, και όχι αρκετά δημοφιλής ακόμα, χρησιμοποιεί τις κινητές συσκευές ως συσκευές πληρωμής, με την φυσική παρουσία των συσκευών να αποτελεί μέρος της διαδικασίας πληρωμής. Η φυσική παρουσία των κινητών συσκευών αυξάνει την ασφάλεια των συναλλαγών (ο επιτιθέμενος θα πρέπει να έχει στην κατοχή του την συσκευή ή ο χρήστης να βρίσκεται πλησίων των συσκευών χρεώσεως), πλήν όμως η αυξανόμενη δημοτικότητα της υπηρεσίας αναμένεται να αυξήσει σημαντικά τα περιστατικά επιθέσεων σε αυτή.

β. **Περιβάλλον Δικτύου**

Το περιβάλλον του δικτύου στο οποίο δραστηριοποιούνται οι κινητές συσκευές απαρτίζεται κυρίως από τρείς παραμέτρους :

**Strong Connection:** Με τον όρο αυτό εννοείται, η σχέση μεταξύ των παρόχων κινητής τηλεφωνίας-ασύρματου internet, με τους κατόχους των συσκευών. Ενώ στις συνδέσεις home internet οι πάροχοι δεν έχουν καμία σχέση με τους προσωπικούς υπολογιστές των χρηστών, στις κινητές επικοινωνίες οι πάροχοι είναι παράλληλα και οι κάτοχοι των SIM cards που βρίσκονται μέσα στις συσκευές. Παρόλο που οι SIM cards αποτελούν ένα αξιόπιστο στοιχείο μέσα στην συσκευή, αποτελεί ανοιχτό ερώτημα κατά πόσο μπορεί να παραβιαστεί η αξιοπιστία τους.

**Firmware Update Process:** Η διαδικασία αναβάθμισης του firmware των κινητών συσκευών, έχει αλλάξει δραματικά τα τελευταία χρόνια. Ενώ στο παρελθόν η διαδικασία αυτή γινόταν σχεδόν αποκλειστικά από τους κατασκευαστές των συσκευών τοπικά, η έλευση νέων τεχνολογιών και αρχιτεκτονικών, έμφάνισε την δυνατότητα αναβάθμισης των firmware μέσω του δικτύου. Η κρίσιμη διαδικασία της αναβάθμισης όμως, θα πρέπει να μην διακοπεί για κανένα λόγο, προκειμένου να παραμείνουν αναλλοίωτα τα ευαίσθητα δεδομένα των χρηστών. Επιπλέον, η αναβάθμιση δεν γίνεται από τους κατασκευαστές των συσκευών, αλλά από τους κατασκευαστές των λειτουργικών συστημάτων (iOS, android) με αποτέλεσμα αυτοί να έχουν τον κύριο λόγο της διαδικασίας αυτής.

**Remote Device Management:** Ένα σημαντικό χαρακτηριστικό των κινητών συσκευών είναι η δυνατότητα διαχείρισής τους από απόσταση. Την δυνατότητα αυτή χρησιμοποιούν οι διαχειριστές των δικτύων για να επιβάλλουν πολιτικές ασφαλείας

χρηστών (πχ απενεργοποίηση Bluetooth) και να αναστείλουν λειτουργίες οι οποίες πιθανών να προκαλέσουν απώλεια ευαίσθητων δεδομένων από το δίκτυο.

### γ. Περιορισμένοι πόροι των συσκευών

Γενικά οι πόροι των κινητών συσκευών είναι περιορισμένοι σε σχέση με των παραδοσιακών desktop μηχανημάτων. Αν και η σημερινή τεχνολογία των κινητών τηλεφώνων είναι παρόμοια με τους προσωπικούς υπολογιστές παλαιότερων χρόνων, εξακολουθεί να υφίσταται ένας περιορισμός ως προς την CPU και την μνήμη RAM. Το παραπάνω γεγονός περιορίζει σημαντικά τις παραμέτρους ασφαλείας των συσκευών, καθώς πολύπλοκοι αλγόριθμοι και υπολογισμοί για τον εντοπισμό επίθεσης, δεν δύναται να πραγματοποιηθούν στα smartphones.

Ένας επιπλέον πόρος ο οποίος συμβάλει ως περιοριστικός παράγοντας στις κινητές συσκευές, είναι η διάρκεια ζωής της μπαταρίας. Οποιαδήποτε εφαρμογή ασφαλείας θα πρέπει να μην καταναλώνει πολύ ενέργεια, άρα μεγάλη υπολογιστική ισχύ στην CPU, προκειμένου η κατανάλωση μπαταρίας να είναι η ελάχιστη δυνατή.

### δ. “Ακριβή” ασύρματη σύνδεση :

Ένα άλλο χαρακτηριστικό της ασφάλειας των κινητών συσκευών είναι η ακριβή επικοινωνία. Με τον όρο ακριβή εννοούμε, τόσο τους πολύπλοκους αλγορίθμους επικοινωνίας του ασύρματου δικτύου, όσο και την μεγάλη κατανάλωση μπαταρίας. Η ασύρματη επικοινωνία από μόνη της βασίζεται σε πιο πολύπλοκους αλγορίθμους και πρωτόκολλα από ότι το παραδοσιακό ενσύρματο δίκτυο, προκειμένου να επιτευχθεί η ασφάλεια των κλήσεων και της μεταφοράς δεδομένων. Αν σε αυτά προσθέσουμε και υπολογιστική ισχύ για την υλοποίηση αλγορίθμων αποτροπής επιθέσεων και αναγνώρισης εισβολέων, το κόστος σε υπολογιστική ισχύ και διάρκεια ζωής της μπαταρίας είναι πολύ μεγάλο.

### ε. Φήμη

Τέλος, ένα ιδιαίτερο χαρακτηριστικό ασφαλείας των κινητών συσκευών, αποτελεί η φήμη του πάροχου. Ενώ στο ενσύρματο δίκτυο, ο πάροχος δεν αντιμετωπίζει πρόβλημα αξιοπιστίας σε περίπωση επιτυχημένης επίθεσης σε χρήστες, στο ασύρματο δίκτυο οι πάροχοι είναι πάντα συνδεδεμένοι με την ασφάλεια των συνομιλιών και δεδομέων του δικτύου τους. Επομένως μια πιθανώς επιτυχημένη επίθεση υποκλοπής δεδομένων μπορεί να πλήξει άμεσα την φήμη του παρόχου και κατ επέκταση όλου του ασύρματου δικτύου της ευρύτερης περιοχής.

## *1.4 Απειλές – Τύποι Επιθέσεων*

Οι απειλές που προσανατολίζονται σε κινητές συσκευές, μπορούν να κατηγοριοποιηθούν σε τέσσερις κατηγορίες:

**Hardware-centric attacks :** Επιθέσεις προσανατολισμένες στο hardware των συσκευών, αν και μπορούν να παραβιάσουν σημαντικές παραμέτρους ασφαλείας (πχ την εμπιστευτικότητα των προσωπικών δεδομένων με forensic analysis), δεν μπορούν να είναι εκμεταλλεύσιμες μέσω mobile malware, καθώς ευπάθειες στο hardware μπορούν να τις εκμεταλλευτούν μονο με φυσική πρόσβαση στην συσκευή.

Ιδιαίτερη περίπτωση αποτελεί, η προσπάθεια παρεμβολής μεταξύ της επικοινωνίας SIM smartcard και της κινητής συσκευής. Η παραπάνω επικοινωνία δεν είναι κρυπτογραφημένη καθώς μια επίθεση man-in-the-middle θεωρούνταν απίθανη όταν πρωτοκατασκευάστηκε η συγκεκριμένη διεπαφή. Πλην όμως, εμφανίστηκε στην αγορά ένα προιόν, η TurboSIM. Η TurboSIM είναι ένα microchip το οποίο προσκολλάται στο πλαστικό μέρος της SIM κάρτας, και δύναται να εφαρμόσει μια πετυχημένη man-in-the-middle επίθεση και να υποκλέψει τα δεδομένα του χρήστη και της κάρτας. Και πάλι όμως, για να είναι εφικτή μια τέτοια απειλή, θα πρέπει ο επιτιθέμενος να έχει έστω και για κάποιο χρονικό διάστημα, φυσική επαφή με την κινητή συσκευή προκειμένουν να προσαρμόσει την TurboSIM στο πλαστικό κομμάτι της κάρτας.

**Device-independent attacks :** Επιθέσεις ανεξάρτητες της φύσης της συσκευής, αποσκοπούν στην κατάρρευση της εμπιστευτικότητας των προσωπικών δεδομένων του χρήστη, μέσω υποκλοπής της ασύρματης επικοινωνίας ή των δεδομένων που βρίσκονται αποθηκευμένα σε back-end συστήματα των παρόχων. Και αυτές οι επιθέσεις όμως, είναι ανεξάρτητες των χαρακτηριστικών ασφαλείας των συσκευών και των λειτουργικών συστημάτων τους. Για την ασύρματη επικοινωνία, οι κάρτες SIM των κινητών συσκευών, παρέχουν όλα τα κρυπτογραφικά κλειδιά, προκειμένου οι χρήστες να έχουν μια ασφαλή επικοινωνία με τους σταθμούς βάσης του δικτύου κινητής τηλεφωνίας. Τα GSM δίκτυα δεν χρησιμοποιούσαν αυθεντικοποίηση δικτύου, με αποτέλεσμα ο οποιοσδήποτε μπορούσε να κατασκευάσει έναν ενδιάμεσο σταθμό βάσης κινητής τηλεφωνίας, και κατόπιν να παραπλανήσει τους χρήστες (τις κινητές συσκευές) ότι αποτελεί ένα κανονικό κομμάτι του ασύρματου δικτύου. Με αυτόν τον τρόπο όλη η επικοινωνία των χρηστών, εξυπηρετούνταν μέσω του σταθμού βάσης του επιτιθέμενου, με αποτέλεσμα την αποκρυπτογράφηση και υποκλοπή των δεδομένων. Η συγκεκριμένη τρωτότητα διορθώθηκε με την έλευση των UMTS ασύρματων δικτύων.

**Software-centric attacks :** Η σημαντικότερη κατηγορία απειλών – επιθέσεων σε κινητές συσκευές, είναι αυτές που βασίζονται στις τρωτότητες και τα κενά ασφαλείας των εφαρμογών που υποστηρίζουν τα λειτουργικα συστήματα των συσκευών.

Στο πρόσφατο παρελθόν, οι τρωτότητες των mobile web browsers, οδήγησαν σε έξαρση των σχετικών επιθέσεων στις κινητές συσκευές.

Στις μέρες μας, η εμπορική επιτυχία του mobile internet, οδήγησε στην ανάπτυξη online εμπορικών καταστημάτων από τους κατασκευαστές των λειτουργικών συστημάτων (App Store, Android Market κλπ), όπου ο καθένας μπορεί να αναπτύξει και να διαθέσει προς πώληση την δική του εφαρμογή. Όπως είναι φυσικό, κανείς δεν εγγυάται την ασφάλεια των εν λόγω εφαρμογών και οι επιθέσεις που σκοπεύουν στην εξεύρεση τρωτοτήτων αυτών, έχουν αυξηθεί δραματικά τα τελευταία 2 χρόνια με σημαντικό αντίκτυπο στα δεδομένα των χρηστών.

α. **Malware**

**Συλλογή προσωπικών δεδομένων :** Μια προσφιλής τακτική των επιτιθέμενων σε κινητές συσκευές είναι η συλλογή προσωπικών δεδομένων των χρηστών και η αποστολή τους σε αυτούς. Μια τακτική η οποία μπορεί εύκολα να προσκοληθεί σε ένα παιχνίδι και να στέλνει στον επιτιθέμενο όλες τις επιθυμητές πληροφορίες των χρηστών όπως συντεταγμένες της κινητής συσκευής, πληροφορίες επικοινωνίας (αποστολή SMS, MMS ).

**Υποκλοπές :** Ένας επίσης δημοφιλής τρόπος επίθεσης και λειτουργικότητας των malware είναι η υποκλοπή και καταγραφή συνομιλιών και η αποστολή τους στους επιτιθέμενους. Χαρακτηριστική είναι η δυνατότητα ορισμένων malware με αυξημένα δικαιώματα στο λειτουργικό σύστημα να καταγράφουν συνομιλίες όχι μονο των κλήσεων, αλλά και του περιβάλλοντα χώρου με την κινητή συσκευή σε κατάσταση αναμονής.

**Οικονομικό κέρδος :** Το οικονομικό κίνητρο αποτελεί έναν κύριο άξονα δράσης των επιτιθέμενων. Μεγάλος αριθμός malware, αποσκοπεί στην δημιουργία κόστους για τον χρήστη και αντίστοιχο οικονομικό όφελος για τον επιτιθέμενο. Malware με δικαιώματα διαχείρισης – αποστολής SMS σε μία κινητή συσκευή, έχει την δυνατότητα αποστολής μηνυμάτων σε συγκεκριμένους αριθμούς χρέωσης, χωρίς την άδεια του χρήστη.

**DDOS επιθέσεις :** Οι DDOS επιθέσεις χωρίζονται σε 2 υποκατηγορίες.

1. **Επιθέσεις εναντίων οργανισμών** : Οι κινητές συσκευές έχουν ένα μεγάλο πλεονέκτημα έναντι των desktop μηχανημάτων. Είναι ενεργές σχεδόν επι 24ώρου βάσεως. Επομένως αποτελούν ένα πολύ καλό εργαλείο στα χέρια των DDOS επιτιθέμενων καθως μία μολυσμένη συσκευή μπορεί να στέλνει απεριόριστες αιτήσεις στον οργανισμό στόχο και να προκαλέσει DOS.

2. **Επιθέσεις εναντίων των ίδιων των συσκευών :** Λόγω των περιορισμένων πόρων του συστήματος, οι κινητές συσκευές είναι εύκολα ευάλωτες σε DDOS επιθέσεις. Ειδικά malware απασχολούν την CPU των συσκευών με τον διαρκή υπολογισμό απλών αλγορίθμων, οι οποίοι όμως προκαλούν δραματική μείωση της διάρκειας ζωής της μπαταρίας.

β. **Mobile Web Browser**

Οι Mobile Web Browser αποτελούν διαρκή στόχο των επιτιθέμενων. Όπως και οι κοινοί web browsers, έχουν εξελιχθεί από απλούς φυλλομετρητές, σε εφαρμογές με πρόσθετες λειτουργικότητες και δυνατότητες. Επομένως είναι και οι ίδιοι εφαρμογές με τρωτότητες, τις οποίες εκμεταλλεύονται οι επιτιθέμενοι για την υποκλοπή προσωπικών δεδομένων των χρηστών. Παραδείγματα επιτυχημένων επιθέσεων ενάντια των mobile browser η DOS επίθεση στον Internet explorer, η χρήση του iPhone browser σαν dialer κα.

**The User as Attack Vector**

Τέλος, ως σημαντικός παράγοντας ασφαλείας εξακολουθεί και παραμένει σε μεγάλο βαθμό ο ίδιος ο χρήστης των κινητών συσκευών. Η ελλιπής γνώση διαδικασιών ασφαλείας από τον μέσο χρήστη, οδηγούν σε λανθασμένες αποφάσεις οι οποίες καταρρέουν την δομη ασφαλείας των συσκευών.

## *1.5 Μοντέλα Ασφαλείας*

Η διαρκώς αυξανόμενη δράση των επιτιθέμενων σε κινητές συσκευές, εξήρε την ανάγκη για δημιουργία μοντέλων ασφαλείας προκειμένου να αντιμετωπιστούν οι επιδράσεις των επιθέσεων στα προσωπικά δεδομένα των χρηστών. Μέχρι στιγμής δύο μοντέλα ασφαλείας έχουν προταθεί και υποστηριχθεί στην κοινότητα των κινητών συσκευών.

Το πρώτο είναι το μοντέλο ασφαλείας Capability-based Application Digital Signature (CADS), το οποίο απαιτεί, ότι μόνο ψηφιακά υπογεγραμμένα πακέτα μπορούν να εγκατασταθούν σε μια συσκευή και μόνο πακέτα με την ίδια ψηφιακή υπογραφή,

μπορούν να μοιραστούν πόρους και δυνατότητες της συσκευής. Αν και εκ πρώτης όψης φαίνεται αρκετό για να εγγυηθεί την ασφάλεια των συσκευών, το CADS προστατεύει τις εφαρμογές μονο κατά την εγκατάστασή τους, υποθέτωντας ότι αυτές δεν αλλάζουν παραμέτρους-δικαιώματα κατά την εκτέλεσή τους. Πλην όμως, αν κατά την διάρκεια της εκτέλεσης ενός API αλλάξουμε τις παραμέτρους, το CADS είναι αδύνατο να εντοπίσεις τις αλλαγές και ο μολυσμένος κώδικας θα εκτελεστεί κανονικά.

Το δεύτερο προτεινόμενο μοντέλο ασφαλείας είναι το Runtime Security Policy Enforcement (RSPE), το οποίο παρέχει ένα λεπτομερή καθορισμό δικαιωμάτων πρόσβασης και υποχρεωτική εφαρμογή τους από τις εφαρμογές (πχ. Ο τηλεφωνικός κατάλογος της συσκευής μπορεί να προσπελαστεί μόνο από συγκεκριμένες εφαρμογές). Το RSPE αν και εφαρμόζει πολιτική ασφαλείας κατά την εκτέλεση των εφαρμογών, ελέγχοντας διαρκώς αν ένα API έχει δικαίωμα πρόσβασης σε πόρους του συστήματος, δεν μπορεί να εγγυηθεί ότι μια εφαρμογή εκτελείται κανονικά. Με το RSPE , η εφαρμογή τρέχει σε ένα προστατευμένο περιβάλλον, το οποίο επιτρέπει μονο σε authorized κώδικα να έχει πρόσβαση σε πόρους του συστήματος και να υλοποιήσει ενέργειες επι αυτών. Τα Domains, δικαιώματα και πολιτικές ασφαλείας των εφαρμογών περιγράφονται αναλυτικά σε ένα σύνολο από αρχεία. (Domain και Policy Stores) και οι πόροι πυρήνα προστατεύονται από το RSPE. Ο τηλεφωνικός κατάλογος που αναφέρθηκε και πριν, μπορεί να προσπελαστεί από εφαρμογές οι οποίες έχουν το κατάλληλο RSPE configuration. Ένα malware, ακόμα και με root δικαιώματα, αν δεν τρέχει σε κατάλληλο domain με τα απαραίτητα δικαιώματα, δεν μπορεί να έχει πρόσβαση στους πόρους τους οποίους έχει πρόσβαση το domain.

Γενικά τα δύο προτεινόμενα μοντέλα, αν υλοποιηθούν και τα δύο μαζί, αυξάνουν σημαντικά την ασφάλεια των κινητών συσκευών. Μία εφαρμογή, αφού πρώτα πιστοποιηθεί η ψηφιακή υπογραφή του, εγκαθίσταται στην συσκευή (λειτουργικότητα CADS) και εν συνεχεία το RSPE λαμβάνει δράση να εφαρμόσει την πολιτική ασφαλείας κατά την εκτέλεσή του. Πάραυτα όμως, δεν μπορούν να εξασφαλίσουν ότι μια εφαρμογή θα εκτελεστεί κανονικά ή δεν θα αλλάξει κάποιες παραμέτρους κατά την διάρκεια εκτέλεσης, επιτρέποντας έτσι επιθέσεις που βασίζονται στις τεχνικές code injection και code replacement [15]

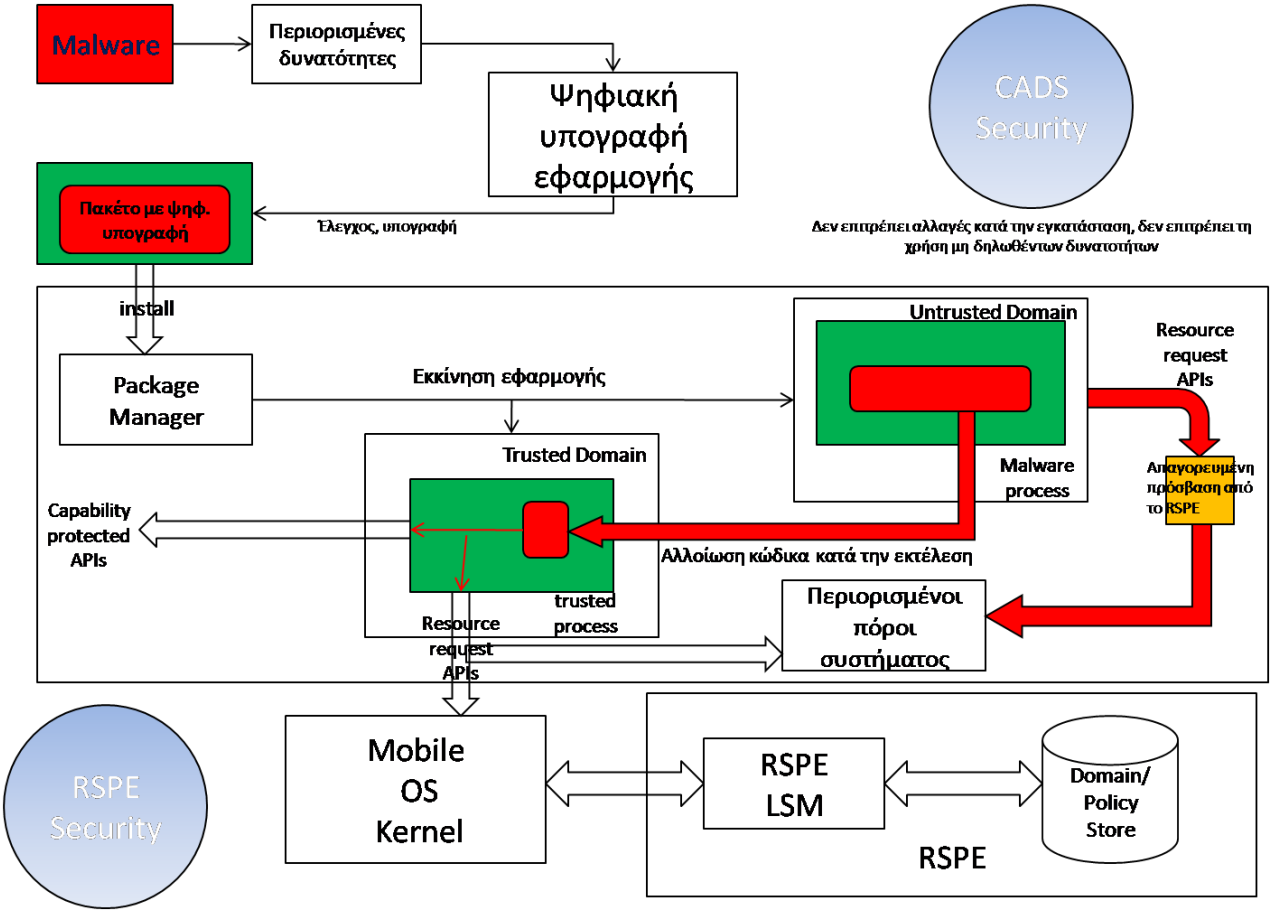
| | CADS | RSPE |
|---|---|---|
| Στάδιο | Κατά την εγκατάσταση | Κατά την εκτέλεση |
| Έλεγχος | Στατικός | Δυναμικός |
| APIs που ελέγχει | Capability-protected | General-purpose APIs & system calls |
| Στόχος | Καμία αλλαγή της εφαρμογής κατά την εγκατάσταση, δεν επιτρέπει την χρήση προστατευμένων εφαρμογών (capability-protected) | Έλεγχος πρόσβασης |
| Μειονεκτήματα | Κανένας έλεγχος στα General-purpose APIs της συσκευής | Η απειλή εντοπίζεται μονο κατά την εκτέλεση. |

**Πίνακας 1**

Στον Πίνακα 1 φαίνονται συνοπτικά οι διαφορές μεταξύ των δύο μοντέλων ασφαλείας που προαναφέρθηκαν.

## *Τρωτότητες Μοντέλων Ασφαλείας*

Όπως φαίνεται και στον Πίνακα 1 τα CADS και RSPE λειτουργούν σε διαφορετικά επίπεδα και γι αυτό το ένα μπορεί να συμπληρώσει το άλλο. Εφαρμοσμένα μαζί μπορούν να αυξήσουν σημαντικά το επίπεδο ασφάλειας των κινητών συσκευών. Κατά την εγκατάσταση το CADS ελέγχει την ψηφιακή υπογραφή της εφαρμογής και επιτρέπει στην συγκεκριμένη εφαρμογή να κάνει χρήση μόνο των δυνατοτήτων που έχουν εξ αρχής οριστεί και έχει λάβει γνώση ο χρήστης. Κατά την εκτέλεσή της, το RSPE, επιτρέπει στη εφαρμογή να κάνει χρήση μόνο των κοινόχρηστων και δηλωμένων πόρων του συστήματος, επομένως η εφαρμογή τρέχει σε ένα ασφαλές περιβάλλον. Όμως η αρχιτεκτονική αυτή δεν μπορεί να εμποδίσει την αλλαγή παραμέτρων της εφαρμογής κατά την εκτέλεσή της.




**Εικόνα 5**

Όπως φαίνεται και στην εικόνα 5, ένα malware για να εγκατασταθεί θα πρέπει να απαιτεί πρόσβαση σε περιορισμένες-συνηθισμένες δυνατότητες και να έχει ψηφιακή υπογραφή. Με την παρουσία του CADS το malware δεν μπορεί να έχει πρόσβαση σε προστατευόμενα APIs. Μετά την εγκατάσταση, το malware τρέχει σε μη αξιόπιστο περιβάλλον και όταν προσπαθήσει να ζητήσει μη επιτρεπόμενους πόρους του συστήματος, το RSPE θα το αποτρέψει. Όμως το malware, μπορεί κατά την εκτέλεση, να εκτελέσει κώδικα μιας άλλης διεργασίας που τρέχει σε ένα αξιόπιστο περιβάλλον και να καταφέρει να εξαπατήσει την διεργασία «θύμα» για να :

- Καλέσει ένα προστατευόμενο API το οποίο δεν είχε δικαίωμα το malware να καλέσει
- Καλέσει ένα API γενικού σκοπού για να έχει πρόσβαση σε πόρους συστήματος που προστατεύονται από το RSPE.

καταφέρνωντας έτσι να παρακάμψει το μοντέλο ασφαλείας (priviledge escalation attacks).

## *1.6 Τεχνικές Εντοπισμού Malware σε smartphones*

Η ανίχνευση και εντοπισμός malware στα smartphones αποτελεί ένα ιδιαίτερα δύσκολο αντικείμενο καθώς αν και οι αρχές εντοπισμού δεν διαφέρουν από τα κλασσικά malware των desktop μηχανημάτων, οι περιορισμένοι πόροι των κινητών συσκευών την καθιστούν μια δυσχερής διαδικασία.

**Signature Based Detection**

Η κλασσική προσέγγιση του προβλήματος είναι ο εντοπισμός με βάση την «υπογραφή» του malware καθώς αυτό έχει ήδη αναγνωριστεί και έχουν καταγραφεί τα χαρακτηριστικά του. Οι εφαρμογές που χρησιμοποιούν αυτήν την προσέγγιση, αφού αναγνωρίσουν την ύπαρξη ενός malware, ‘παράγουν’ την υπογραφή του και την αποθηκεύουν για περαιτέρω χρήση. Στις κινητές συσκευές όμως, αυτή η διαδικασάι είναι πιο πολύπλοκη, καθώς ο αλγόριθμος εντοπισμού του malware θα πρέπει να είναι διαρκώς σε κατάσταση ανίχνευσης, προκειμένουν να αξιολογήσει όλες τις διαδικασίες που τρέχουν στον επεξεργαστή για την ύπαρξη malware. Προφανώς αυτό από μόνο του προσθέτει ένα τεράστιο φόρτο εργασίας στους ήδη περιορισμένους πόρους των κινητών συσκευών που μπορεί να προκαλέσει δυσάρεστες παρενέργειες στον χρήστη (αργό γραφικό περιβάλλον, γρήγορη εξάντληση της μπαταρίας κλπ). Για τον λόγο αυτό έχουν προταθεί αρκετές λύσεις, που παραπέμπουν την διαδικασία ανίχνευσης της συσκευής σε περιβάλλον cloud.

Πέραν των παραπάνω, μια διαφορετική προσέγγιση έχει υιοθετήσει η Apple, καθώς για να ανέβει μια εφαρμογή στο Apple Store, ελέγχεται πρώτα από την ίδια την εταιρεία και κατόπιν εγκρίσεως, αυτή θα εγκατασταθεί στο ηλεκτρονικό κατάστημα της Apple. Όμως κανείς δεν εγγυάται ότι όλα τα malware προγραμματα εντοπίζονται πριν αυτά διατεθούν προς εμπορική χρήση. Η μέχρι τώρα εμπειρία στο χώρο έχει δείξει ότι αρκετές εφαρμογές διατέθηκαν στο κοινό με έγκριση της εταιρείας και κατόπιν αποσύρθηκαν καθώς εντοπίστηκαν ανεπιθύμητες ενέργειες από malware.

**Anomaly Detection**

Μία διαφορετική προσέγγιση, προσπαθεί να αναγνωρίσει την ύπαρξη malware τα οποία εκτελούν ‘’άγνωστες’’ ενέργειες. Έχουν προταθεί αρκετά εργαλεία που χρησιμοποιούν αυτή την τεχνική προσέγγισή όπως το SmartSiren [29] . Με το Smartsiren είναι πιθανός ο εντοπισμός malware με βάση την συμπεριφορά που έχει μια εφαρμογή, όταν επικοινωνεί μέσω SMS ή Bluetooth. Τα δεδομένα από την επικοινωνία αυτή, στέλνονται σε έναν κεντρικό proxy server ο οποίος προσπαθεί να εντοπίσει ‘’περίεργες’’ ή

‘’άγνωστες’’ συμπεριφορές. Αυτή η προσέγγιση είναι ικανή να εντοπίζει ταχέως διαδιδόμενα ‘’worms’’ καθώς και malware τα οποία δρούν αργά, στο υπόβαθρο των εφαρμογών, και στέλνουν κατά περιόδους τα ευαίσθητα δεδομένα των χρηστών στους επιτιθέμενους.

**Rootkit Detection**

Κάποια εξελιγμένα malware μπορεί να επιχειρήσουν να κρύψουν την ύπαρξή τους σε επίπεδο πυρήνα (kernel). Οι τεχνικές rootkit, για τον εντοπισμό των υπόψη malware δεν διαφέρουν από τις αντίστοιχες των παραδοσιακών ηλεκτρονικών υπολογιστών και η διαδικασία εντοπισμού παραμένει πολύ δύσκολη.

**Software-based Attestation**

Οι Jakobsson και Johansson περιγράφουν μια προσέγγιση για εντοπισμό malware με βάση το memory printing [30] των προγραμμάτων που εκτελούνται. Η ιδέα πάνω στην οποία βασίζεται χρησιμοποιεί κάποια κρυπτογραφικά σχήματα, τα οποία έχουν το χαρακτηριστικό να αργεί σημαντικά η εκτέλεσή τους, αν για κάποιο λόγο το σύστημα τους διαθέσει λιγότερη μνήμη RAM από αυτή που έχει τεθεί ως προαπαιτούμενη. Συνεπώς, τα malware τα οποία εκτελούνται στην flash memory των κινητών, λόγω της ύπαρξης των συγκεκριμένων κρυπτογραφικών σχημάτων, εκτελούνται με μεγαλύτερη χρονική καθυστέρηση, η οποία μπορεί εύκολα να παρατηρηθεί και να εντοπιστεί η ύπαρξη του malware στην συσκευή.

## *1.7 Ασφάλεια Λειτουργικών Συστημάτων*

Όπως είναι γνωστό, τα λειτουργικά συστήμτα των κινητών συσκευών, εξελίχθηκαν τόσο πολύ τα τελευταία χρόνια με αποτέλεσμα να είναι παραπλήσια σε δυνατότητες με τα παραδοσιακά λειτουργικά των desktop υπολογιστών. Οι περισσότερες εταιρείες που δραστηριοποιούνται στον χώρο, προσπαθούν να υλοποιήσουν ένα ενιαίο λειτουργικό σύστημα, το οποίο θα υποστηρίζει όλες τις λειτουργίες του σε οποιοδήποτε περιβάλλον (desktop, smartphone, tablet, laptop). Η ασφάλεια των λειτουργικών συστημάτων μπορεί μελλοντικά να αποτελέσει ενιαίο παράγοντα για όλες τις πλατφορμες και να αντιμετωπιστεί συνολικά.

Παρακάτω παρουσιάζονται μερικοί τρόποι επαύξησης της ασφάλειας των λειτουργικών συστημάτων κινητών συσκευών οι οποίες σε κάποιες περιπτώσεις έχουν υιοθετηθεί από τους κατασκευαστές.

**Περιορισμένα δικαιώματα και απομόνωση διεργασιών**

Οι περισσότερες ευάλωτες εφαρμογές, μπορεί να τρέξουν malware μέσα στα όρια των δικών τους δικαιωμάτων. Αν κάποια εφαρμογή κατέχει πολύ υψηλά δικαιώματα, τότε μια πιθανή τρωτότητα θέτει σε κίνδυνο όλο το λειτουργικό σύστημα. Αυτή η προσέγγιση έχει υιοθετηθεί από την πλατφόρμα Android καθώς κάθε εφαρμογή εκτελείται σε δικό της περιβάλλον (JVMs) και κατω από διαφορετικό χρήστη (UID). Αναλυτικότερα θα αναφερθούμε σε επόμενο κεφάλαιο.

**Δυσκολότεροι Kernels**

Στα λειτουργικά συστήματα των κλασσικών desktop μηχανημάτων, η ασφάλεια του πυρήνα έχει ανέβει σε πολύ υψηλά επίπεδα με την πάροδο του χρόνου και η εκτέλεση malware κώδικα μέσω κρίσιμων σφαλμάτων ασφαλείας, αποτελεί ιδιαίτερα δύσκολο έργο για τους επιτιθέμενους. Μια προσέγγιση για την αύξηση της ασφάλειας των κινητών συσκευών είναι να εφαρμοστούν οι ίδιες τακτικές λύσεις, όπως Address Space Layout Randomization, stack protection και non-executable writable memory σε όλες τις πλατφόρμες.

**Default Settings**

Οι προεγκατεστημένες υπηρεσίες και εφαρμογές των κινητών συσκευών θα πρεπει να έχουν οριστεί προσεκτικά και η χρήση τους να επιτρέπεται μονο εάν αυτό απαιτείται (Bluetooth υπηρεσία κλπ). Χαρακτηριστικό παράδειγμα, όπου κάποια Symbian smartphones ήταν ευάλωτα σε DoS επιθέσεις μέσω μια υπηρεσίας δικτύου η οποία ήταν διαθέσιμη με τις default ρυθμίσεις της συσκευής.

**Αναβαθμίσεις ΛΣ**

Τα λειτουργικά συστήματα δεν είναι ποτέ τέλεια και δεν είναι ποτέ 100% ασφαλή. Όσο αυτά αναπτύσσονται, τα λάθη στην υλοποίησή τους θα υπάρχουν πάντα. Όταν οι τρωτότητες ανακαλύπτονται, οι εταιρείες λειτουργικών συστημάτων ενημερώνουν τους χρήστες για την ύπαρξη διαθέσιμης αναβάθμισης προκειμένου αυτή να εγκατασταθεί και να διορθωθεί το πρόβλημα. Η παραπάνω διαδικασία στους κλασικούς ηλεκτρονικούς υπολογιστές είναι γνώριμη στους χρήστες και ελάχιστα ‘δαπανηρή’ σε υπολογιστικό κόστος. Στις κινητές συσκευές όμως η διαδικασία update είναι δύσκολη και χρονοβόρα ενώ συνήθως γίνεται από εξουσιοδοτημένο προσωπικό και όχι από τον χρήστη. Θα πρέπει λοιπόν να βρεθεί λύση προκειμένου η διαδικασία του update να είναι το ίδιο λειτουργική και εύχρηστη με αυτή των παραδοσιακών desktop μηχανημάτων.

### Διαδικασία εγκατάστασης των εφαρμογών

Η διαδικασία εγκατάστασης εφαρμογών διαφέρει σημαντικά από το παραδοσιακό περιβαλλον των ηλεκτρονικών υπολογιστών όπου η online εγκατάσταση γίνεται από το site του κατασκευαστή του προγράμματος, και του οποίου η αυθεντικότητα ελέγχεται μέσω ψηφιακών υπογραφών. Στις κινητές συσκευές, όλες οι διαθέσιμες εφαρμογές βρίσκονται στα online καταστήματα των κατασκευαστών των λειτουργικών συστημάτων . Λόγω της αξιοπιστίας που παρέχει το brandname του online καταστήματος, οι χρήστες εγκαθιστούν με μεγαλύτερη ασφάλεια, εφαρμογές από άγνωστους εκδότες οι οποίες όμως ενδέχεται να εμπεριέχουν malware κώδικα που έχει διαφύγει του ελέγχου της εταιρείας. Επιπλέον στην πλατφόρμα Android, ο χρήστης ενημερώνεται πριν εγκαταστήσει μια εφαρμογή, για τα δικαιώματα που απαιτεί αυτή για να εγκατασταθεί και αν συμφωνεί, δίνει έγκριση για την εγκατάστασή της. Συνεπώς, μια εφαρμογή είτε εγκαθίσταται με όλα τα δικαιώματα που απαιτεί, επικίνδυνα ή μη, είτε δεν εγκαθίσταται καθόλου. Δεν υπάρχει τρόπος εγκατάστασης της εφαρμογής με παράλληλη αποτροπή των κακόβουλων ενεργειών.

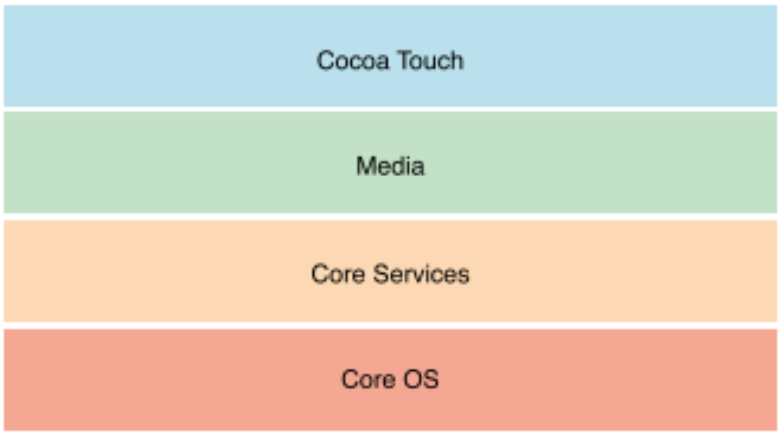
# ΚΕΦΑΛΑΙΟ 2

# ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ iOS

## *2.1 iOS - Αρχιτεκτονική*

Το λειτουργικό σύστημα iOS, όπως όλα τα λειτουργικά συστήματα, δρά ως ενδιάμεσος μεταξύ του hardware των συσκευών, και των εφαρμογών που εμφανίζονται στην οθόνη του χρήστη.

Το iOS είναι περαιτέρω διαιρεμένο στα παρακάτω λογικά επίπεδα, τα οποία παρέχουν τις απαραίτητες αφαιρετικές δομές για την λειτουργικότητα των εφαρμογών.




**Εικόνα 6 [7]**

Τα επίπεδα τα οποία είναι χαμηλότερα, περιέχουν όλες τις θεμελιώδεις υπηρεσίες και λειτουργίες που είναι απαραίτητες για την λειτουργικότητα των εφαρμογών, ενώ στα υψηλότερα επίπεδα βρίσκονται πιο εξελιγμένες τεχνολογίες και υπηρεσίες.

Αναλυτικότερα :

### *2.1.1 Επίπεδο Core OS*

Σε αυτό το επίπεδο βρίσκονται όλες οι θεμελιώδης υπηρεσίες που παρέχει το λειτουργικό σύστημα στον χρήστη. Το iOS παρέχει διεπαφές προκειμένου να είναι προσπελάσιμες αρκετά χαμηλού επιπέδου χαρακτηριστικά. Οι διεπαφές αυτές είναι γραμμένες σε γλώσσα C, είναι διαθέσιμες μέσω της βιβλιοθήκης *LibSystem* και υποστηρίζουν τις παρακάτω λειτουργικότητες.

- Threading (POSIX threads)
- Networking (BSD sockets)
- File-system access
- Standard I/O
- Bonjour and DNS services
- Locale information

- Memory allocation
- Math computations

Πλέον των παραπάνω low-level υπηρεσιών, υπάρχουν και οι δημοφιλείς λειτουργικοτητες που χρησιμοποιούν σχεδόν όλες οι εφαρμογές.

- **Accelerate Framework**

  Διεπαφές για τις υπηρεσίες DSP, Linear Algebra, and Image Processing.

- **Core Bluetooth Framework**

  Λειτουργικότητες που επιτρέπουν στις εφαρμογές να αλληλεπιδρούν με Bluetooth συσκευές.

- **External Accessories Framework**

  Διασύνδεση της συσκευής, με εξωτερικές συσκευές μέσω 30-pin connector.

- **Security Framework**

  Διαχείριση των Certificates, δημοσίων και ιδιωτικών κλειδιών, πολιτικές ασφαλείας, δημιουργία ψευδοτυχαίων κλειδιών κλπ

### *2.1.2 Επίπεδο Core Services*

Στο επίπεδο αυτό συναντάμε μια ποικιλία από βιβλιοθήκες οι οποίες είναι ασύνδετες μεταξύ τους, παρέχοντας θεμελιώδης υπηρεσίες που χρησιμοποιούν σχεδόν όλες οι εφαρμογές και αρκετά μέρη του συστήματος είναι δομημένα πάνω σε αυτές. Οι κυριότερες αυτών είναι:

• **Address Book Framework:** δίνει την δυνατότητα για πρόσβαση και διαχείριση των επαφών που είναι αποθηκευμένες στην συσκευή.

• **CF Network:** Εδώ βρίσκονται όλα τα αντικείμενα για την διεπαφή με όλα τα πρωτόκολλα δικτύου. (BSD sockets, SSL/TLS, DNS, HTTP/HTTPS, FTP

• **Core Data Framework:** παρέχει τις τεχνολογίες για την διαχείριση του μοντέλου δεδομένων των εφαρμογών (SQLite db, table views, data validation κλπ)

• **Core Foundation Framework:** βασικές δομές δεδομένων που υποστηρίζει το iOS (Date/Time, Threading κλπ).

• **Core Location :** παρέχει τις πληροφορίες θέσης και επικεφαλίδες των εφαρμογών. (GPS, wifi radios για να είναι δυνατή η εύρεση συντεταγμένων του χρήστη)

• **Core Telephony Framework:** παρέχει τις διασυνδέσεις για παροχή πληροφοριών που σχετίζονται με την κινητή τηλεφωνία (όνομα παρόχου κλπ κλπ)

• **Core Media Framework:** εδώ βρίσκονται όλοι οι low-level media τύποι δεδομένων για την διαχείριση-έλεγχο του ήχου και την εικόνας.

• **Event Kit Framework :** η διασύνδεση με γεγονότα ημερολογίου στη συσκευή. (ειδοποιήσεις με ημερομηνία κλπ)

### *2.1.3 Επίπεδο Media Layer*

Το επίπεδο media, περιέχει όλες τις τεχνολογίες ήχου, video, εικόνας για την δημιουργία των εφαρμογών. Συνοπτικά οι λειτουργικότητες που παρέχει το συγκεκριμένο επίπεδο φαίνονται στον πίνακα [2]

| | |
|---|---|
| • Core Graphics | • Media Player Framework |
| • 2D Vector and Raster Graphics | • Access to iTunes Library and Simple Playback |
| • Core Animation | •AV Foundation Frameworks |
| • View Animation | • Audio and Video Capture and Playback |
| • Core Image | • OpenAL |
| • Image and Video Manipulation, Filters | • Positional Audio |
| • OpenGL ES and GLKit | • Core Audio Framework |
| •Hardware-accelerated 3D Graphics | • Advanced Audio Playback |
| • Core Text | • Core Video Framework |
| • Text Layout and Rendering | • Advanced Video Playback |
| • Image I/O | • AirPlay |
| • Reading and Writing Images | • Stream Audio and Video to Other Devices |
| • Assets Library | |
| • Access to User’s Photos and Videos | |

Πίνακας 2

### *2.1.4 Επίπεδο Cocoa Touch*

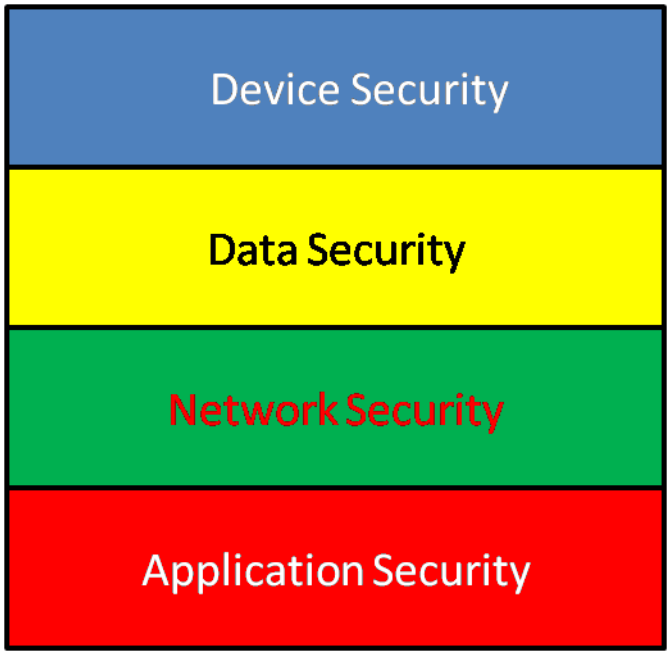
Στο ανώτερο επίπεδο, βρίσκονται όλες οι απαραίτητες λειτουργίες για την κατασκευή μιας iOS εφαρμογής. Το επίπεδο καθορίζει την απαραίτητη δομή που πρέπει να έχει η εφαρμογή, και υπσοτηρίζει όλες τις δομικές λειτουργικότητες όπως, δυνατότητα εισόδου δεδομένων από την touch screen, ειδοποιήσεις κλπ. Συνοπτικά οι σημαντικότερες υπηρεσίες που παρέχει το **cocoa touch layer** είναι οι παρακάτω:

| | |
|---|---|
| •Storyboards (iOS 5) | • Local Notifications |
| • Documents (iOS 5) | •Gesture Recognition |
| • Multitasking | •File-Sharing |
| •Printing | •Peer-peer Services |
| • Data Protection | •External Display Support |
| •Push Notifications | •SystemViewControllers |
| •Address Book | • Maps |
| • Contacts | • Message UI |
| •Event Kit UI | • E-Mail and SMS |
| • Calendar | •Twitter |
| •Game Kit | • Tweets |
| • Multiplayer Games | • UIKit |
| • Advertisements Map Kit | • Everything else |

Πίνακας 3

## 2.2 Μοντέλο Ασφαλείας

Η Apple έχει αναπτύξει ένα μοντέλο ασφαλείας το οποίο βασίζεται σε 4 επίπεδα (layers) με σκοπό να προστατέψει τους χρήστες και τα δεδομένα τους (Εικόνα 7)




**Εικόνα 7**

**Device Security**

Εδώ βρίσκονται οι τεχνικές για την παρεμπόδιση οποιουδήποτε μη εξουσιοδοτημένου ατόμου να έχει πρόσβαση στη συσκευή.

**Data Security**

Στο επίπεδο αυτό εφαρμόζονται οι τεχνικές για την προστασία των αποθηκευμένων δεδομένων του χρήστη, ακόμα και αν η κινητή συσκευή κλαπεί.

**Network Security**

Εδώ λαμβάνει χώρα η κρυπτογράφηση των δεδομένων καθώς αυτά μεταδίδονται στο ασύρματο δίκτυο.

**Application Security**

Τέλος στο application security layer βρίσκονται όλοι οι μηχανισμοί προστασίας του λειτουργικού συστήματος, και η απομόνωση των διεργασιών – εφαρμογών καθώς αυτές εκτελούνται.

### *2.2.1 Device Security*

Οι μηχανισμοί για την προστασία της συσκευής της Apple εγγυώνται ότι μια συσκευή δεν μπορεί να χρησιμοποιηθεί από μη εξουσιοδοτημένο προσωπικό/υπηρεσίες. Ο πιο γνωστός μηχανισμός ασφαλείας αυτού του επιπέδου είναι ο κωδικός PIN και ο κωδικός passcode. Η Apple επιτρέπει αυτά τα κλειδώματα ασφαλείας, είτε να είναι προαπαιτούμενα ως μέρος μιας γενικότερης πολιτικής ασφαλείας, είτε να μπορούν να εφαρμοστούν ως επιλογές των χρηστών.

### *Policy Enforcement*

Επιπλέον των παραπάνω κλειδωμάτων, το iOS περιλαμβάνει ως κομμάτι της στρατηγικής ασφαλείας, την χρήση προεπιλεγμένων προφίλ ρυθμίσεων, επιτρέποντας έτσι την κεντρική διανομή των VPN,WiFi, email και άλλων ρυθμίσεων κάτω από μια ασφαλή παραμετροποίηση. Επομένως είναι δυνατή η εφαρμογή πολιτικής ασφαλείας σε μια εταιρεία, ενεργοποιώντας/απενεργοποιώντας λειτουργικότητες των συσκευών, συγκεντρωτικά για όσες συσκευές είναι καταχωρημένες σε ένα συγκεκριμένο domain. (Εικόνα 8)

| **Feature (true/false)** | **Comment** |
|---|---|
| Install or Update Apps | Disables Appstore. Users are unable to install and update Apps. |
| Enable or disable Siri | Disables Siri |
| Allow camera | Disables camera. Users are unable to take photographs |
| Allow Explicit Content | Hides explicit music or video content purchased from the iTunes Store. |
| Allow Screenshot | Users are unable to take screenshot of the display |
| Allow Youtube | Disables Youtube app |
| Allow iTunes | Disables iTunes Music Store |
| Force iTunes Store Password | Forces user to enter iTunes password for each transaction (iOS 5) |
| Allow Safari | Disables Safari web browser. |
| Allow untrusted TLS Prompt | Automatically rejects untrusted HTTPS certificates without prompting the user (iOS5) |
| Allow Cloud Backup | Disables backing up the device to iCloud |
| Allow Cloud Document Sync | Disables document syncing to iCloud |
| Allow Photo Stream | Disables Photo Stream (iOS5) |
| Force encrypted Backup | Forcing encrypted backup with iTunes |

**Εικόνα 8 [7]**

### *Mobile Device Management*

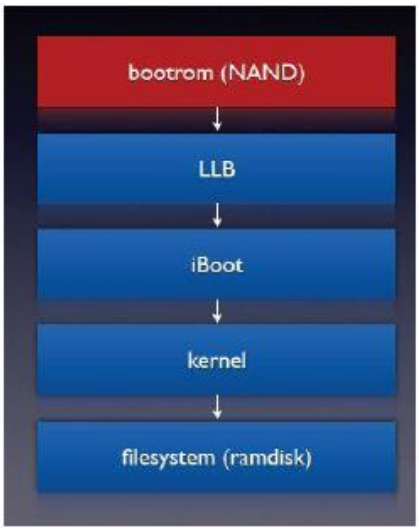
Προκειμένου να είναι δυνατή η διαχείριση μεγάλου αριθμού κινητών συσκευών Apple, το iOS υποστηρίζει την υπηρεσία MDM από όπου είναι δυνατή η κεντρική διαχείριση των συσκευών, η εγγραφή νέων, η εγκατάσταση εφαρμογών από απόσταση κλπ. Το MDM βασίζεται στο Policy Enforcement και στην υπηρεσία Apple Push Notification που θα αναλύσουμε παρακάτω.

### *Υπηρεσία Apple Push Notification*

Η συγκεντρωτική διαχείριση των Apple συσκευών γίνεται μέσω της σύνδεσης των συσκευών με έναν MDM server ο οποίος παρέχεται από την εταιρεία. Η Επικοινωνία του server με τις συσκευές, επιτυγχάνεται μέσω του μηχανισμού Apple Push Notification Service. Όταν ο server θέλει να επικοινωνήσει με τις συσκευές, στέλνει ένα μήνυμα μέσω του APNS. Οι συσκευές με την σειρά τους, μόλις λάβουν το APNS μήνυμα, επικοινωνούν με τον MDM server προκειμένου να ολοκληρώσουν την επιθυμητή από τον server εργασία (πχ το update του firmware). Συνεπώς με την χρήση αυτή της υπηρεσίας σε συνδυασμό με τον MDM server, είναι δυνατή η δημιουργία domain με τις κινητές συσκευές iOS ενός οργανισμού και η διαχείρισή τους από τους αρμόδιους IT.

### *Secure Boot Chain*

Η διαδικασία εκκίνησης της συσκευής αποτελείται από 4 βήματα τα οποία σχεδιάστηκαν για να εξασφαλίσουν την ακεραιότητα της συσκευής.(εικόνα [9)]. Κάθε βήμα κατά την έναρξη της συσκευής, περιέχει στοιχεία τα οποία είναι κρυπτο-υπογεγραμμένα από την Apple, και η διαδικασία του boot συνεχίζει μόνο όταν πιστοποιηθεί η chain of truth της συσκευής. Όταν μια iOS συσκευή εκκινέι, αμέσως εκτελείται κώδικας από την read only μνήμη, Boot ROM, ο οποίος θεωρείται αξιόπιστος. Ο κώδικας αυτός περιλαμβάνει το **Apple Root CA public key**, το οποίο χρησιμοποιείται για να πιστοποιηθεί ότι ο Low-Level Bootloader (LLB) είναι υπογεγραμμένος από την Apple.




**Εικόνα 9 [6]**

Όταν τελειώσει το LLB με τις εργασίες του, αρχίζει το δεύτερο βήμα του iBoot, το οποίο με την σειρά του πιστοποιεί την αυθεντικότητα του κώδικα που τρέχει και παραδίδει την σκυτάλη στον πυρήνα του iOS. Με αυτή την τεχνική, η Apple εξασφαλίζει ότι το iOS μπορεί να τρέξει μονο σε πιστοποιημένες συσκευές της εταιρείας. Σε περίπτωση που κάποιο στάδιο κατά την εκκίνηση αποτύχει στην διαδικασία πιστοποίησης, τότε η συσκευή εμφανίζει στην οθόνη, το μήνυμα “Connect to iTunes” το οποίο αποτελεί την διαδικασία της αποκατάστασης της κινητής συσκευής (recovery mode). Αν η συσκευή δεν μπορέσει καν να διαβάσει από την Boot ROM μνήμη το LLB, τότε αυτή εισέρχεται σε κατάσταση αναβάθμισης του firmware (DeviceFirmwareUpdate / DFU mode).

### *2.2.2 Data Security*

Η ασφάλεια των δεδομένων αποτελεί τον βασικό πυλώνα ασφαλείας των εφαρμογών. Η Apple έχει αναπτύξει έναν αριθμό από data security προσεγγίσεις για την προστασία των ευαίσθητων δεδομένων των συσκευών, ακόμα και αν μια συσκευή κλαπεί από τον κάτοχό της. Οι κυριότεροι μηχανισμοί που έχουν αναπτυχθεί στο iOS είναι : Η απομακρυσμένη διαγραφή δεδομένων (remote wipe function), η κρυπτογραφία, και διάφοροι μηχανισμοί προστασίας των δεδομένων.

#### *Remote wipe function*

Ο χρήστης μπορεί να ενεργοποιήσει αυτήν την υπηρεσία αμέσως μόλις καταλάβει την απώλεια της συσκευής του ή μπορεί να ενεργοποιηθεί από μόνη της, μετά από επιλογή του χρήστη, όταν εισαχθεί λάθος κωδικός passcode πάρα πολλές φορές. Με την ενεργοποίηση της υπηρεσίας διαγράφεται/αντικαθίσταται

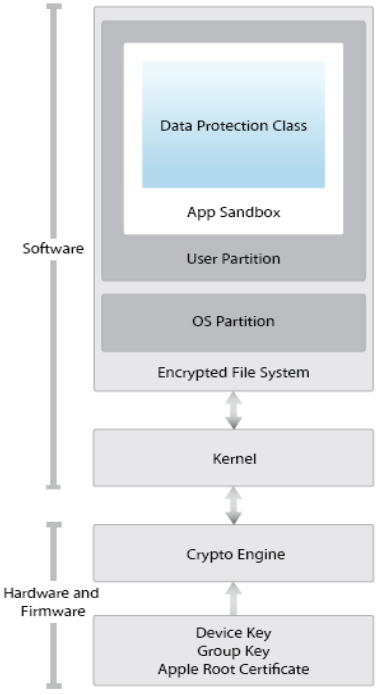
αυτόματα το master key της συσκευής (UID) και τα αποθηκευμένα δεδομένα δεν είναι δυνατόν να προσπελάστούν από τον νέο κάτοχο της συσκευής.

### *Cryptography*

Κάθε iOS συσκευή έχει μια αποκλειστική κρυπτομηχανή AES-256 η οποία παρεμβάλεται της μνήμης της συσκευής και της μνήμης του λειτουργικού συστήματος. Παράλληλα με την κρυπτομηχανή-επιταχυντή AES έχει υλοποιηθεί και ο αλγόριθμος SHA-1στο hardware προκειμένου να μειωθεί σημαντικά ο αντίκτυπος-επιβράδυνση της κρυπτογράφησης δεδομένων στον χρήστη.

Όλα τα δεδομένα στις κινητές συσκευές με λειτουργικό σύστημα iOS είναι κρυπτογραφημένα. Ακόμη και η διαδικασία backup μέσω του iTunes, χρησιμοποιεί κωδικό της συσκευής προκειμένου τα δεδομένα του backup να αποθηκευτούν κρυπτογραφημένα.

Κάθε συσκευή έχει μοναδικό κωδικό ID (UID) και ένα κωδικό ομάδος device group ID (GID). Κάθε ένα από αυτά αποτελεί ένα AES-256 κλείδί, και εισέρχονται στο hardware της συσκευής κατά την κατασκευή της. Κανένα πρόγραμμα ή εξωτερική συσκευή δεν είναι σε θέση να διαβάσει αυτά τα κλειδιά. Το UID είναι μοναδικό για κάθε συσκευή Apple ενώ το GID αποτελέι κοινό χαρακτηριστικό γνώρισμα μιας ομάδας επεξεργαστών/συσκευών. Τα συγκεκριμένα κλειδιά χρησιμοποιούνται σε όλα τα στάδια κρυπτογράφησης των δεδομένων της συσκευής.




**Εικόνα 10 [6]**

### *File Data Protection*

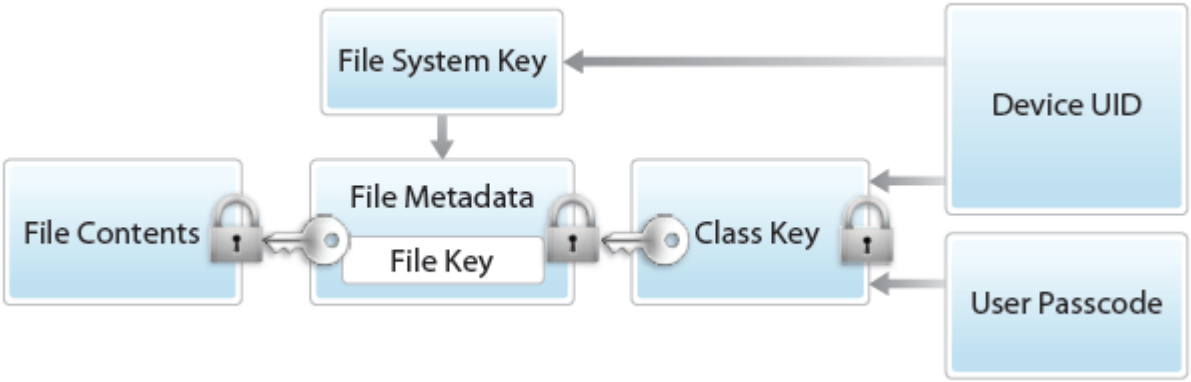
Οι συγκεκριμένοι μηχανισμοί ασφαλείας αποτελούν τον βασικότερο στόχο των επιτιθέμενων στις κινητές συσκευές με iOS λειτουργικό σύστημα. Η Apple χρησιμοποιεί όπως είδαμε, έναν κρυπτογραφικό επιταχυντή, υλοποιημένο σε hardware, προκειμένου να κρυπτογραφούνται τα δεδομένα των εφαρμογών. Σε συνδυασμό με τον κωδικό που δίνει ο ίδιος ο χρήστης (passcode), το iOS εγγυάται ότι τα δεδομένα των εφαρμογών παραμένουν ασφαλή και μπορούν να διαβαστούν παρά μονο μετά την εισαγωγή του κωδικού ασφαλειας από τον χρήστη.

Κάθε φορά που δημιουργείται ένα αρχείο στη συσκευή, το Data Protection δημιουργεί ένα νέο 256-AES key (the “per-file” key) το οποίο δίνει στην κρυπτομηχανή προκειμένου να το χρησιμοποιήσει για την κρυπτογράφηση των δεδομένων σε AES-CBC mode (το IV του CBC mode προέρχεται από το LFSR υπολογισμένο από το block offset του αρχείου και είναι κρυπτογραφημένο με τον SHA-1 χρησιμοποιώντας το “per-file” κλειδί του αρχείου).

Το “per-file” κλειδί του αρχείου, ανάλογα με τα δικαιώματα πρόσβασης που έχει το ίδο το αρχείο, κρυπτογραφείται εκ νέου με κάποια από τις κλάσεις ασφαλείας του συστήματος και αποθηκέυεται στα metadata του αρχείου.

Όταν το αρχείο ανοίγει, τα metadata αποκρυπτογραφούνται με το κλειδί του iOS, αποκαλύπτοντας το κρυπτογραφημένο “per-file” key και μια σημείωση που δείχνει σε ποια κλάση κλειδιών ανήκει. Κατόπιν αποκρυπτογραφείται με το κλειδί της κλάσης και το κρυπτογραφημένο αρχείο οδηγείται στην κρυπτομηχανή της συσκευής για αποκρυπτογράφηση. Τέλος τα metadata όλων των αρχείων της συσκευής, κρυπτογραφούνται με ένα τυχαίο κλειδί (file-system key) το οποίο δημιουργείται όταν γίνεται η εγκατάσταση του λειτουργικού συστήματος στη συσκευή ή όταν ο χρήστης ενεργοποιήσει την λειτουργία *remote wipe* που αναφέραμε παραπάνω.

Το file-system κλειδί, αποθηκεύεται στη συσκευή και γι αυτό δεν χρησιμοποιείται για την διατήρηση της εμπιστευτικότητας των δεδομένων. Με την διαγραφή του κλειδιού αυτού όλα τα δεδομένα της συσκευής είναι απροσπέλαστα.




**Εικόνα 11 [6]**

Παρόλο που ολόκληρο το filesystem είναι κρυπτογραφημένο, μονο ορισμένα αρχεία έχουν τον μηχανισμό προστασίας δεδομένων. Στις νέες συσκευές τον συγκεκριμένο μηχασνισμό έχουν μόνο τα email και τα προσαρτημμένα αυτών. Οι εφαρμογές που αναπτύσσονται από third party οντότητες, θα πρέπει να συμπεριλάβουν την προστασία δεδομένων στον κώδικά τους εάν επιθυμούν την ενεργοποίηση αυτής της υπηρεσίας.

### *Passcode*

Με την ενεργοποίηση της επιλογής του Passcode από τον χρήστη αυτόματα ενεργοποιείται και η λειτουργία του Data Protection. Το iOS επιτρέπει την χρήση 4ψήφιων αριθμητικών κλειδιών και strings οποιουδήποτε μήκους.

Επιπλέον το Passcode συμμετέχει μαζί με το UID της συσκευής, στην εντροπεία για την δημιουργία των κλειδιών κρυπτογράφησης των κλάσεων κλειδιών, τα οποία δεν αποθηκεύονται στην συσκευή. Ένας επιτιθέμενος ο οποίος έχει στην κατοχή του την συσκευή δεν μπορεί να έχει πρόσβαση στα δεδομένα του χρήστη, καθώς οι κωδικοί των κλάσεων δεν βρίσκονται τοπικά στην συσκευή.

### *Classes*

Όταν δημιουργείται ένα αρχείο, όπως αναφέραμε πρωτύτερα, ανατίθεται σε μια κλάση ασφαλείας από την εφαρμογή που δημιουργεί το αρχείο. Κάθε κλάση καθορίζει την πολιτική ασφαλείας του αρχείου και πότε είναι αυτό διαθέσιμο για ανάγνωση-εγγραφή-διαγραφή. Οι κύριες κλάσεις του iOS είναι:

*Complete Protection :* Το class key προστατεύεται από το Passcode και το UID της συσκευής. Όταν η συσκευή είναι κλειδωμένη το κλειδί δεν είναι

διαθέσιμο και τα δεδομένα που ανήκουν σε αυτήν την κλάση δεν είναι προσπελάσιμα. (από κατασκευής τα mail με τα συνημμένα τους ανήκουν στην complete protection class)

*Protected Unless Open :* Κάποια αρχεία μπορεί να χρειαστεί να είναι διαθέσιμα για εγγραφή ακόμα και όταν είναι κλειδωμένη η συσκευή με το Passcode (πχ ένα mail attachment το οποίο κατεβαίνει στην συσκευή ενώ αυτή είναι κλειδωμένη). Η λειτουργικότητα αυτή επιτυγχάνεται με ECDH κρυπτογραφία. Μαζί με το “per-file” κλειδί, το Data Protection δημιουργεί και ένα ζεύγος ιδιωτικού-δημοσίου κλειδιού. Κατόπιν δημιουργείται το “κοινό μυστικό” χρησιμοποιώντας το ιδιωτικό κλειδί του αρχείου και το *Protected Unless Open* δημόσιο κλειδί που δημιουργήθηκε *(*Το διιωτικό κλειδί προστατεύεται με το Passcode και το UID). Το “per-file” κλειδί περνάει από τον SHA-1 με κλειδί το κοινό μυστικό που δημιουργήθηκε, και το hash που δημιουργείται αποθηκεύεται στα metadata του αρχείου μαζί με το δημόσιο κλειδί. Το ιδιωτικό κλειδί αφαιρείται από την μνήμη και μόλις κλείσει το αρχείο, διαγράφεται και το “per-file” κλειδί. Για να αποκτήσει κάοπιος πρόσβαση στο αρχείο, το κοινό μυστικό επαναδημιουργείται χρησιμοποιώντας το ιδιωτικό κλειδί της κλάσης και το δημόσιο κλειδί του αρχείου.

*Protected Until First User Authentication* : Αυτή η κλάση είναι παρόμοια με την κλάση Complete protection με την διαφορά ότι το κλειδί δεν αφαιρείται από την μνήμη (άρα είναι διαθέσιμο κάθε στιγμή μετα την πρώτη σωστή εισαγωγή του Passcode) όταν η συσκευή κλειδώσει.

*No Protection :* Η συγκεκριμένη κλάση προστατέυεται μόνο από το UID. Αυτή είναι η κλάσση που ανατίθεται σε όλα τα αρχεία, εκτός και αν επιλέξει ο χρήστης να κάνει χρήση του Data Protection.

### *Keychain Data Protection*

Αρκετές εφαρμογές διαχειρίζονται κλειδιά και ευαίσθητα δεδομένα, τα οποία θα πρέπει να κρυπτογραφούνται. Η λειτουργία Keychain του iOS προσφέρει αυτή την λειτουργικότητα για τις εφαρμογές, χρησιμοποιώντας έναν 128-bit AES αλγόριθμο. Υλοποιείται με την SQLite βάση δεδομένων, η οποία είναι αποθηκευμένη στο file system της συσκευής κατω από την No Protection κλάση ασφαλείας. Υπάρχει μονο μια βάση δεδομένων keychain και ο security deamon του συστήματος καθορίζει ποια keychain είναι προσβάσιμα από ποιες διεργασίες ή εφαρμογές. Μια “αλυσίδα” του keychain μπορεί να διαμοιραστεί σε περισσότερες των μια εφαρμογών, αρκεί οι εφαρμογές αυτές να είναι προιόν από τον ίδιο developer.

Τα δεδομένα του Keychain χρησιμοποιούν παρόμοιες κλάσεις με αυτές του Data Protection με παρόμοιες συμπεριφορές αλλά έχουν διαφορετικά κλειδιά.

| Availability | File Data Protection | Keychain Data Protection |
|---|---|---|
| When unlocked | NSFileProtectionComplete | kSecAttrAccessibleWhenUnlocked |
| While locked | NSFileProtectionCompleteUnlessOpen | N/A |
| After first unlock | NSFileProtectionCompleteUntilFirstUserAuthentication | kSecAttrAccessibleAfterFirstUnlock |
| Always | NSFileProtectionNone | kSecAttrAccessibleAlways |

**Εικόνα 12 [6]**

### *System Keybag*

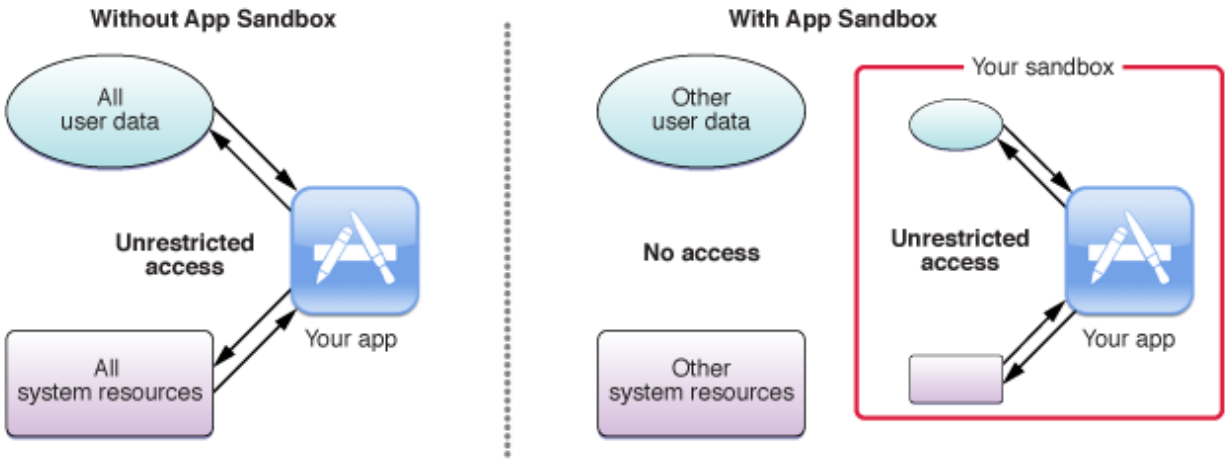
Το System Keybag είναι η τοποθεσία όπου αποθηκεύονται τα κρυπτογραφημένα κλειδιά των κλάσεων του File Data Protection, αλλά και του Keychain, κατά την λειτουργία του συστήματος.

### *2.2.3 Ασφάλεια δικτύου*

Η ασφάλεια του δικτύου είναι ένα θέμα που δεν απασχολεί μόνο τις κινητές επικοινωνίες. Η apple έχει υιοθετήσει αρκετές τεχνικές ασφαλείας με το κλασσικό internet όπως, δίκτυα VPN, χρήση πρωτοκόλλου SSL/TLS για κρυπτογράφηση επιπέδου μεταφοράς, και WEP/WPA/WPA2 για κρυπτογράφηση του ασύρματου δικτύου και αυθεντικοποίηση. Επιπλέον υποστηρίζεται το S/MIME πρωτόκολλο για την ασφαλή ανάγνωση – αποστολή email μηνυμάτων.

### *2.2.4 Ασφάλεια εφαρμογών*

Στο επίπεδο εφαρμογών, όλες οι διαθέσιμες εφαρμογές για τα iphone-ipad βρίσκονται στο App store, από όπου ο κάθε χρήστης μπορεί να τις εγκαταστήσει στην συσκευή του. Κάθε μια εφαρμογή τρέχει μέσα σε ένα “sandbox”. Με τον όρο sandboxing αναφερόμαστε σε ένα περιβάλον στο οποίο τρέχουν οι εφαρμογές, η μία ανεξάρτητα από την άλλη, και όπου ο κώδικας της εφαρμογής θεωρείται “αναξιόπιστος” (και για αυτό τον λόγο απομονώνεται κατά την εκτέλεση από τις υπόλοιπες διεργασίες).




**Εικόνα 13 [6]**

### *Sandboxing*

Με το sandbox, κάθε εφαρμογή έχει περιορισμένη μνήμη την οποία μπορεί να χρησιμοποιήσει, καθώς και περιορισμένο αριθμό CPU κύκλων. Επιπλέον κάθε εφαρμογή έχει πρόσβαση μόνο στον φάκελο στον οποίο έχει εγκατασταθεί και πουθενά αλλού. Το λειτουργικό σύστημα επιτρέπει στις εφαρμογές να επικοινωνούν με πόρους όπως η κάμερα και το GPS, πλήν όμως αρκετά στοιχεία των συσκευών είναι απροσπέλαστα από αυτές. Καμία εφαρμογή δεν μπορεί να έχει πρόσβαση στα δεδομένα άλλων εφαρμογών και δεν μπορούν να έχουν πρόσβαση στο file system της συσκευής.

### *Application Containers*

Σε κάθε εφαρμογή η οποία έχει αναπτυχθεί από τρίτες οντότητες, ανατίθεται συγκεκριμένος υποφάκελλος αρχείων μέσα στον οποίο η εφαρμογή έχει πλήρη δικαιώματα ανάνγωσης- εγγραφής των αρχείων που περιέχονται (Application Home Directory) . Η περιοχή στην οποία αποθηκέυονται όλα τα αρχεία των εφαρμογών βρίσκονται στην διαδρομή **/var/mobile/Applications/*UUID***. Ο μοναδικός κωδικός **UUID** της εφαρμογής δίνεται δυναμικά και τυχαία σε αυτήν κατά την εγκατάστασή της. Αν η εφαρμογή απεγκατασταθεί από την συσκευή και εν συνεχεία επανεγκατασταθεί, ο φάκελος αρχείων διαγράφεται εντελώς και ένας νέος κωδικός **UUID** ανατίθεται στην εφαρμογή.

### *Sandbox Profiles*

Όλες οι εφαρμογές στο iOS εκτελούνται κάτω από τον ίδιο Unix User (mobile) και συνεπώς το μοντέλο ασφαλείας που παρέχει το Unix για την απομόνωση των εφαρμογών κατά την εκτέλεση δεν είναι διαθέσιμο. Έτσι το iOS χρησιμοποιεί το Sandbox για την εφαρμογή της πολιτικής ασφαλείας του με την χρήση του MAC Framework. Το MAC framework παρέχει μια πιο λεπτομερή πολιτική ασφαλείας για την πρόσβαση σε πόρους και αντικείμενα του συστήματος.

Όταν μια εφαρμογή εκκινεί, το sandbox προφίλ (εάν έχει) καθορίζεται από τα προκαθορισμένα δικαιώματα που του δίνει το συγκεκριμένο προφίλ. Το iOS παρέχει 35 προκαθορισμένα Sandbox profiles (για εφαρμογές γραφικών, κλπ).

Μέσα στο Sandbox οι εφαρμογές μπορούν να εκτελέσουν συγκεκριμένες προκαθορισμένες εργασίες. Κάθε εφαρμογή έχει πρόσβαση μόνο στις λειτουργίες που της καθορίζει το Sandbox προφίλ της.

### *Container Sandbox Profile*

Το Container Sandbox Profile είναι το προφίλ που ανατίθεται σε όλες τις εφαρμογές που αναπτύσσονται από εταιρείες πλήν της Apple. Το συγκεκριμένο προφίλ έχει σημαντικούς περιορισμούς στην πρόσβαση των αρχείων. Μία εφαρμογή μπορεί να έχει πρόσβαση μονο στα αρχεία του καταλόγου της, σε μερικά απαραίτητα αρχεία συστήματος, και στον κατάλογο επαφών του χρήστη. Επίσης μπορεί μια εφαρμογή να διαβάσει τα αρχεία πολυμέσων της συσκευής, αλλά όχι να τα διαγράψει ή να γράψει στον κατάλογο αυτό και επίσης μπορεί να διαβάσει και να διαγράψει από τον κατάλογο είσερχομένων email αλλά όχι να γράψει σε αυτόν .

### *Application signing*

Η apple για την μεγαλύτερη προστασία των χρηστών, επιτρέπει την εγκατάσταση εφαρμογών, οι οποίες είναι εγκεκριμένες από την ίδια την εταιρεία. Μια εφαρμογή αν δεν έχει certificate από την apple, δεν μπορεί να εγκατασταθεί σε κινητή συσκευή της εταιρείας. Επιπλέον, κατά την διάρκεια της εκτέλεσης της εφαρμογής, το λειτουργικό σύστημα ελέγχει την ακεραιότητα αυτής, προκειμένου να διασφαλιστεί ότι δεν υπάρχει *code injection/replacement.*

### *Mandatory code Signing*

Προκειμένου να διασφαλίσει η apple την αυθεντικότητα του κώδικα που εκτελείται στις συσκευές, απαιτεί όλες οι εφαρμογές να είναι υπογεγραμμένες από ένα αξιόπιστο και γνωστό πιστοποιητικό. Τα σημαντικότερα δομικά στοιχεία που ελέγχει το code signing είναι:

- *Developer Certificates*

Οποιοσδήποτε επιθυμεί να αναπτύξει μια εφαρμογή για το iOS, μπορεί εύκολα να το κάνει με το εργαλείο Xcode που παρέχει ελεύθερα η Apple. Για να μπορέσει όμως η εφαρμογή να εγκατασταθεί σε μια συσκευή Apple, θα πρέπει ο developer να αποκτήσει ένα έγκυρο πιστοποιητικό (certificate) κατόπιν αιτήσεως από την εταιρεία.

- *Signed Applications*

Οι developers με την απόκτηση έγκυρου πιστοποιητικού ασφαλείας από την Apple, μπορούν να υπογράψουν την εφαρμογή τους και να την διαθέσουν στο κοινό για εμπορική χρήση. Υπενθυμίζεται ότι οι εφαρμογές της Apple θεωρούνται έμπιστες και ασφαλείς.

- *Entitlements*

Οι υπογεγραμμένες εφαρμογές, μπορούν επίσης να περιέχουν και ένα XML plist αρχείο όπου καθορίζονται τα δικαιώματα που απαιτεί η εφαρμογή να έχει στις λειτουργικότητες της συσκευής. Αυτό το αρχείο περιέχει ένα σύνολο κλειδιών που αντιπροσωπεύουν τα δικαιώματα που απαιτεί η εφαρμογή. Από την στιγμή που η Apple πριν την διάθεση μιας εφαρμογής στο App Store, ελέγχει τον κώδικα για θέματα ασφαλείας, είναι σε θέση να γνωρίζει ποια εφαρμογή χρησιμοποιεί ποια δικαιώματα, και σε περίπτωσει που εντοπίσει malware να απαγορέψει την διάθεσή της στους χρήστες.

### *Code Signing Enforcement*

Σκοπός του Code Signing Enforcement είναι η απαγόρευση εκτέλεσης νέου μη εγκεκριμένου, κώδικα κατά την διάρκεια της εκτέλεσης μιας εφαρμογής. Εμποδίζει τις εφαρμογές από το να φορτώνουν νέες μη υπογεγραμμένες βιβλιοθήκες, να κατεβάζουν νέο κώδικα και να χρησιμοποιούν κώδικα ο οποίος αλλάζει δυναμικά κατά την εκτέλεση. Παρέχει επίσης ένα επιπλέον επίπεδο ασφαλείας κατά των μεμακρυσμένων επιθέσεων οι οποίες προσπαθούν να βάλουν νέο εκτελέσιμο κώδικα

στις εφαρμογές (buffer overflow κλπ). Με το CSE ο επιτιθέμενος θα πρέπει να μπορέσει να εισβάλει στο σύστημα, εκτελώντας κώδικα ο οποίος είναι ήδη στην μνήμη και είναι εγκεκριμμένος από την Apple, δυσχαιρένοντας σημαντικά το έργο του.

Το CSE είναι υλοποιημένο μέσα στον πυρήνα του iOS και συγκεκριμένα μέσα στην virtual memory κάνωντας χρήση του “dirty” bit για την υλοποίηση της Copy-on-Write τεχνικής και για να ελέγχει τις σελίδες που εισέρχονται στην virtual memory. Κάθε νέα σελίδα που φορτώνεται στην μνήμη σημειώνεται ως “dirty” και θα πρέπει να πιστοποιηθεί η αυθεντικότητά της. Σε περίπτωση που μια σελίδα βρεθεί να είναι άκυρη, χωρίς πιστοποιητικό ασφαλείας, τότε ολόκληρη η διαδικασία θεωρείται άκυρη και δεν εκτελείται.

#### *Keychain Data Protection*

Κομάτι της ασφάλειας εφαρμογών απότελεί και το keychain, του οποίου η βασική λειτουργία είναι η αποθήκευση και επαναφορά κρυπτογραφημένων κωδικών, χαρακτηριστικών δικτύου και λοιπών άλλων πληροφοριών όπως είδαμε σε προηγούμενη παράγραφο. Το *security framework* που παρουσιάστηκε παραπάνω, διαθέτει όλες τις κατάλληλες δομές για την υλοποίηση των λειτουργιών εγγραφής-ανάγνωσης από το keychain και για την κρυπτογράφηση των δεδομένων. Οι βασικές κρυπτογραφικές δομές που υποστηρίζει το iOS είναι οι αλγόριθμοι AES, 3DES και RC4. Επιπλέον όπως ήδη προαναφέρθηκε, υπάρχει διαθέσιμος κρυπτογραφικός επιταχυντής AES καθώς και SHA1 hasing υλοποιημένα στο hardware των συσκευών.

### *2.2.5 Address Space Layout Randomization (ASLR)*

Το ASLR αποτελεί ένα σημαντικό χαρακτηριστικό ασφαλείας καθώς δυσκολεύει σε μεγάλο βαθμό το exploitation τρωτοτήτων που βασίζονται σε σφάλματα μνήμης. Όταν υλοποιείται σωστά, ο επιτιθέμενος θα πρέπει να βρεί περισσότερες από μια τρωτότητες στην μνήμη για μια επιτυχημένη επίθεση. Με το ASLR οι διευθύνσεις στην μνήμη όπου εκτελείται ο κώδικας των εφαρμογών διαφέρει από εκτέλεση σε εκτέλεση (randomized). Η υπόψη τεχνική υλοποιήθηκε μετά το iOS 4.3 και παρέχει 2 επιλογές υλοποίησης, αναλόγως αν η εφαρμογή έγινε compiled με υποστήριξη **της υπηρεσίας Position Independent Executables** ή όχι. Αν δεν χρησιμοποιήθηκε το PIE, τότε η χρήση του ASLR είναι περιορισμένη. Συγκεκριμένα, ο κύριος εκτελέσιμος κώδικας και οι περιοχές εγγραφής των δεδομένων θα φορτωθούν σε προκαθορισμένες περιοχές στην μνήμη. Επίσης η στοίβα των threads (***main thread stack)*** πάντα θα

ξεκινά στην ίδια διευθυνση στη μνήμη. Στην εικόνα [14] φαίνονται οι διαφορές μεταξύ των δύο διαφορετικών υλοποιήσεων του ASLR και οι δομές στις οποίες εφαρμόζεται η τυχαιότητα στην επιλογή των διευθύνσεων μνήμης κατά την εκτέλεση των εφαρμογών.

**Memory Region Randomization by Deployment Target Version**

| PIE | Executable | Data | Heap | Stack | Libraries | Linker |
|---|---|---|---|---|---|---|
| No | Fixed | Fixed | Randomized per execution | Fixed | Randomized per device boot | Fixed |
| Yes | Randomized per execution | Randomized per execution | Randomized per execution (more entropy) | Randomized per execution | Randomized per device boot | Randomized per execution |

**Εικόνα 14 [10]**

Αν και το ASLR παρέχει ένα επιπλέον επίπεδο ασφαλείας, οι περισσότερες εμπορικές εφαρμογές οι οποίες έχουν αναπτυχθεί από τρίτες οντότητες, δεν υποστηρίζουν την full ASLR λειτουργικότητα. Χαρακτηριστικός ο παρακάτω πίνακας με τις 10 πιο δημοφιλής εφαρμογές του App Store για το 2011, καμία εκ των οποίων δεν υποστηρίζει το PIE.εικόνα [15]

| Application | Version | Post Date | PIE |
|---|---|---|---|
| Songify | 1.0.1 | June 29, 2011 | No |
| Happy Theme Park | 1.0 | June 29, 2011 | No |
| Cave Bowling | 1.10 | June 21, 2011 | No |
| Movie-Quiz Lite | 1.3.2 | May 31, 2011 | No |
| Spotify | 0.4.14 | July 6, 2011 | No |
| Make-Up Girls | 1.0 | July 5, 2011 | No |
| Racing Penguin, Flying Free | 1.2 | July 6, 2011 | No |
| ICEE Maker | 1.01 | June 28, 2011 | No |
| Cracked Screen | 1.0 | June 24, 2011 | No |
| Facebook | 3.4.3 | June 29, 2011 | No |

**Εικόνα 15 [10]**

## *2.3 Τομείς επίθεσης*

Η ραγδαία άνοδος της χρήσης κινητών συσκευών τα τελευταία χρόνια οδήγησε και στην κατακόρυφη αύξηση των επιθέσεων σε αυτές. Οι κινητές συσκευές με iOS αποτελούν ένα πολύ μεγάλο ποσοστό της αγοράς παγκοσμίως και οι εφαρμογές που είναι διαθέσιμες για αυτές κατέχουν την $1^{η}$ θέση στην σχετική λίστα. Οι επιθέσεις εναντίων των iOS συσκευών, όπως ήταν αναμενόμενο αυξάνονται κάθε χρόνο και μπορούν να χωριστούν σε 2 βασικές κατηγορίες. Επιθέσεις εναντίων της συσκευής και επιθέσεις με στόχο τις εφαρμογές που εγκαθίστανται σε αυτές.

### *2.3.1 Επιθέσεις κατά των iOS εφαρμογών*

Οι εφαρμογές που εγκαθίστανται στις iOS κινητές συσκευές όπως είδαμε, είναι διαθέσιμες μέσω του App Store και έχουν πιστοποιηθεί για την ασφάλειά τους από την Apple. Η εταιρεία το 2011 ανακοίνωσε ότι πάνω από 10 δις εφαρμογές έχουν γίνει download από το ηλεκτρονικό κατάστημα της εταιρείας, ενώ το μεγαλύτερο μέρος αυτών ήταν από κατασκευαστές και εταιρείες που δεν ανήκουν στην apple. Οι εφαρμογές αν και έχουν ελεγχθεί από την Apple πριν την είσοδό τους στο App Store, εν τουτοις αρκετές εξ αυτών παρουσιάζουν τρωτότητες τις οποίες οι επιτιθέμενοι εκμεταλλέυονται.

Πέραν των παραπάνω, αρκετοί κάτοχοι iOS συσκευών, προκειμένου να είναι σε θέση να εγκαθιστούν εφαρμογές οι οποίες δεν έχουν λάβει πιστοποιητικό από την Apple προχωρούν στην τακτική του Jaibreaking της συσκευής,

#### *2.3.1.1 Κατηγορίες iOS εφαρμογών*

Οι εφαρμογές που αναπτύσσονται για τις iOS συσκευές κατέχουν την 1$^{η}$ θέση στην αγορά των κινητών συσκευών και μπορούν να κατηγοριοποιηθούν ως εξής :

**α. Browser based**

Διαδικτυακές εφαρμογές οι οποίες διαθέτουν browser based διεπαφή για την αλληλεπίδραση του χρήστη με αυτή. Οι εφαρμογές χρησιμοποιούν τον default browser του iOS, Safari και έχουν τα κλασσικά χαρακτηριστικά των κλασσικών browser based εφαρμογών : αναπτύσσονται με server side τεχνολογία php, .NET και χρησιμοποιούν Javascript , HTML και CSS για την παρουσίαση του γραφικού περιβάλλοντος χρήστη.

**β. Τοπικές εφαρμογές**

Εφαρμογές που τρέχουν τοπικά στην συσκευή. Συνήθως είναι γραμμένες σε γλώσσα Objective C (αποτελεί ένα υπερσύνολο των προγραμματιστικών γλωσσών C και C++). Η Apple διαθέτει στους developers το iOS SDK προκειμένου να αναπτύξουν τις εφαρμογές τους στις κινητές συσκευές.

**γ. Υβριδικές εφαρμογές**

Αποτελούνται από συνδυασμό στοιχείων των δύο άλλων κατηγοριών, δηλαδή τοπικές εφαρμογές που χρησιμοποιούν λειτουργικότητες browser .

### *2.3.1.2 Jailbreaking*

Με τον όρο Jailbreak εννοούμε ότι μια συσκευή παρακάμπτει τους περιορισμούς ασφαλείας που θέτει το λειτουργικό σύστημα iOS με την εγκατάσταση του. Προγραμματιστικά Jailbreak είναι η μετατροπή του λειτουργικού συστήματος προκειμένου να δέχεται κώδικα ο οποίος δεν έχει λάβει πιστοποιητικό από την Apple. Μια Jailbreaked συσκευή είναι ικανή να εκτελέσει κώδικα ο οποίος δεν είναι εγκεκριμένος από την apple, να αλλάξει τα εικονίδια των εγκατεστημένων εφαρμογών, να έχει πρόσβαση σε όλα τα αρχεία της συσκευής (και όχι μονο στο φάκελλο της εφαρμογής) και να μετατραπεί η ίδια η συσκευή σε hotspot. Πέρα όμως από τα πλεονεκτήματα που αποκτά κάποιος, μετά το Jailbreak της συσκευής, δημιουργούνται σημαντικά κενά ασφαλείας από την εγκατάσταση εφαρμογών που δεν έχουν πιστοποιηθεί από την Apple.

#### *Τύποι Jailbreak*

**Tethered Jailbreak** : Η μετατροπή του λειτουργικού συστήματος είναι προσωρινή και μετά από reboot η συσκευή επανέρχεται στις εργοστασιακές του ρυθμίσεις.

**Untethered Jailbreak** : Η αλλαγή του λειτουργικού συστήματος είναι μόνιμη ακόμα και μετά το reboot της συσκευής.

#### *Jaibreaking Tools*

Υπάρχουν πολλά εργαλεία για το “σπάσιμο” των apple συσκευών. Τα redsn0w, absinthe, Ac1dSn0w, Blackra1n, GreenPois0n, Blackra1n είναι μερικά από τα διαθέσιμα tools με το RedSn0w να είναι το πιο δημοφιλές όλων, καθώς πέραν του jailbreak, διαθέτει στους χρήστες και άλλες λειτουργικότητες.

#### *RedSn0w*

Όπως αναφέρθηκε το RedSn0w αποτελεί το πιο δημοφιλές εργαλείο για jailbreaking apple συσκευών και προσφέρει πολλές λειτουργικότητες οι οποίες είναι διαθέσιμες μέσω του DFU mode της συσκευής. Οι κυριότερες λειτουργικότητες που προσφέρει είναι :

**Jailbreak** : μόνιμο Jailbreaking της συσκευής

**Justboot** : προσωρινό Jailbreaking της συσκευής

**Pwned DFU** : Επιτρέπει στους χρήστες να επαναφέρουν την συσκευή μέσω του iTunes. Το Device Firmware Update mode (DFU), είναι η κατάσταση κατά την οποία η συσκευή βρίσκεται σε κατάσταση σύνδεσης με το iTunes χωρίς το iTunes να στέλνει αυτόματα την αναβάθμιση του firmware στην συσκευή.

**Recovery Fix :** Λειτουργικότητα για παράκαμψη τυχόν λαθών κατά την παραπάνω επαναφορά της συσκευής.

**SHSH blobs :** Δυνατότητα εγκατάστασης λειτουργικού iOS παλαιότερης εκδοσης από αυτή που έχει εγκατεστημένη η συσκευή.

#### *2.3.1.3 Επιθέσεις κατά των αποθηκευμένων δεδομένων*

Σκοπός των επιτιθέμενων είναι να αποκτήσουν πρόσβαση στα προσωπικά δεδομένα των χρηστών μέσω αδυναμιών που παρουσιάζουν οι εφαρμογές που έχουν εγκατασταθεί στις κινητές συσκευές.

Όπως έχει αναφερθεί όλες οι εφαρμογές τρέχουν κάτω από έναν χρήστη (user:mobile), μέσα στο δικό τους sandbox και δεν έχουν πρόσβαση σε δεδομένα άλλων εφαρμογών. Όταν εγκαθίσταται μια εφαρμογή, δημιουργούνται από το λειτουργικό σύστημα, οι παρακάτω υποφακέλοι, μέσα στον φάκελο εγκαταστασης της εφαρμογής.

| Υποφάκελος | Περιγραφή |
|---|---|
| Appname.app | Περιέχει τον κώδικα της εφαρμογής και δεδομένα |
| Documents | Δεδομένα τα οποία δύναται να είναι κοινόχρηστα με την επιφάνεια εργασίας μέσω του iTunes |
| Library | Support files της εφαρμογής |
| Library/preferences/ | Ειδικές προτεραιότητες της εφαρμογής |

| | |
|---|---|
| Library/caches/ | Δεδομένα τα οποία είναι απαιτούμενα κατά την εκκίνηση της εφαρμογής αλλά δεν απαιτείται να γίνονται αντίγραφα ασφαλείας αυτών |
| Tmp | Αρχεία τα οποία δεν είναι απαιτούμενη η παρουσία τους κατά την επιτυχή εκκίνηση των εφαρμογών. |

Πίνακας 4

Στη συνέχεια θα δούμε αναλυτικά τα είδη των αρχείων που δημιουργούν και αποθηκεύουν τοπικά στις συσκευές οι iOS εφαρμογές, καθώς και τα κενά ασφαλείας που τυχόν δημιουργούνται.

### *Plist Files*

Τα Property list files είναι αρχεία που δημιουργούν οι εφαρμογές και εκεί αποθηκεύονται τα properties των χρηστών, τα configuration settings και τα ζεύγη κλειδιών σε δυαδική ή XML μορφή. Οι εφαρμογές κατασκευάζουν τα Plist files με η χωρίς file extention αλλά μπορούν εύκολα να αναγνωριστούν καθώς όλα περιέχουν την κεφαλίδα αρχείου –**bplist**. Επιπλέον, τα αρχεία είναι εύκολο να διαβαστούν και να τροποποιηθούν με την χρήση κατάλληλων εργαλείων (πχ **plutil**) καθώς δεν προστατεύονται από το Data Protection του iOS.

Έχωντας υπόψιν τα παραπάνω, η αποθηκευση κρίσιμων πληροφοριών στα Plist files αποτελεί σημαντική πηγή κινδύνου διαροής ευαίσθητων δεδομένων. Παρολα αυτά, αρκετές εφαρμογές, εξακολουθούν και αποθηκεύουν στα Plist files ευαίσθητα δεδομένα τα οποία μπορούν εύκολα να υποκλαπούν (usernames, passwords, tokes, emails κλπ). Επιπλέον, τα Plist Files αποθηκεύονται χωρίς κρυπτογράφηση κατά την διαδικασία backup από το iTunes.

Μια πρακτική που χρησιμοποιούν αρκετές εφαρμογές είναι η διαγραφή των Plist files κατά την αποσύνδεση του χρήστη. Πλην όμως και αυτή η τεχνική δεν εγγυάται την ασφάλεια των δεδομένων καθώς όλες οι αλλαγές στο file system αποθηκεύονται στο HFS Journal file **(-l.journal**), συνεπώς εκεί καταγράφονται και τα διεγραμμένα αρχεία Plist. Με κατάλληλη τεχνική είναι δυνατή η επεξεργασία του HFS journal αρχείου και η εξαγωγή των διεγραμμέων αρχείων από αυτό με αποτέλεσμα την απώλεια των ευαίσθητων δεδομένων που υπάρχουν σε αυτά.

### *SQLite Files*

Η Apple για την λειτουργία των εφαρμογών κάνει χρήση της SQLite βάσης δεδομένων. Η SQLite είναι μια μικρή, εύχρηστη, ελαφριά και γρήγορη βάση δεδομένων κατάλληλη για κινητές συσκευές. Οι εφαρμογές χρησιμοποιούν την βάση δεδομένων για να αποθηκεύουν εκεί τα usernames, passwords κλπ, προκειμένου να είναι διαθέσιμα κατά την εκτέλεση τους, δημιουργώντας η κάθε εφαρμογή το δικό της SQLite αρχείο.

Όμως υπάρχουν κάποια στοιχεία τα οποία καθιστούν την χρήση της επιρρεπή ως προς την ασφάλεια των δεδομένων. Τα δεδομένα που αποθηκεύονται στην βάση από τις εφαρμογές είναι χωρίς κρυπτογράφηση, και επιπλέον, αν και η ίδια η βάση είναι κρυπτογραφημένη στο file system, το backup αρχείο της βάσης στο iTunes αποθηκεύεται χωρίς κρυπτογράφηση και αυτό. Αρκετές εφαρμογές για επιπλέον προστασία χρησιμοποιούν την τεχνική να διαγράφουν τα SQLite files κατά την αποσύνδεση του χρήστη, πλήν όμως όπως αναφέρθηκε και παραπάνω με την επεξεργασία του HFS Journal αρχείου είναι δυνατή η επαναφορά τους.

Πέραν των παραπάνω, μερικές εφαρμογές χρησιμοποιούν την τεχνική να διαγράφουν εγγραφές από το SQLite αρχείο τους στο πλαίσιο αυξημένης ασφάλειας των δεδομένων. Όμως η SQLite βάση, όταν σβήνεται μια εγγραφή, την σημαδεύει ως διεγραμμένη αλλά δεν την διαγράφει από την μνήμη του συστήματος. Οι εγγραφές που έχουν οριστεί ως διεγραμμένες μπορούν εύκολα να προσπελαστούν, καθώς ίχνη τους υπάρχουν στο WAL (write ahead log) αρχείο του συστήματος.

## *Keychain Data Protection*

Η λειτουργικότητα Keychain Data Protection υλοποιείται όπως προαναφέρθηκε με μια SQLite βάση δεδομένων όπου εκεί αποθηκεύονται κρυπτογραφικά κλειδιά και passwords των εφαρμογών και των δεδομένων αυτών. Η βάση δεδομένων του Keychain αποθηκεύεται σε συγκεκριμένη τοποθεσία στην συσκευή : **/var/Keychains/keychain-2.db**. Το συγκεκριμένο αρχείο είναι προσπελάσιμο από όλες τις εφαρμογές, αλλά κάθε εφαρμογή μπορεί να έχει πρόσβαση μόνο στις δικές της εγγραφές (τα δικά της keychain items) ανάλογα με το keychain access group που ανήκουν.

Σε μια Jailbreaked συσκευή, οι περιορισμοί πρόσβασης μπορούν να παρακαμφθούν εύκολα, με την σχεδίαση και εγκατάσταση στη συσκευή, μιας εφαμογής που είναι μέλος σε όλα τα keychain access groups. Τότε η συγκεκριμένη

εφαμογή θα έχει πρόσβαση σε όλες τις εγγραφές της Keychain βασης δεδομένων και άρα σε όλα τα κλειδιά και τα δεδομένα αυτών.

### *Error Log Files*

Τα log file των εφαρμογών αποθηκεύονται στην τοποθεσία : /private/var/log/syslog. Με την εφαρμογή Console App (διαθέσιμη στο App Store) είναι δυνατή η εμφάνιση των log αρχείων. Τα log files εκτελούνται έξω από το Sandbox των εφαρμογών επομένως είναι προσβάσιμα από όλες τις εφαρμογές. Εφαρμογές δύναται να αποθηκεύουν ευαίσθητα δεδομένα στα log files με αποτέλεσμα να θέτουν αυτά εκτός παραμέτρων ασφαλείας.

### *Screenshots*

Το Home Button των συσκευών Apple, όταν πατηθεί ενώ μια εφαρμογή τρέχει σε fullscreen mode, μικραίνει την εφαρμογή με ένα κινούμενο εφέ. Για την δημιουργία του εφέ, το iOS λαμβάνει ένα στιγμιαίο screenshot της εφαρμογής την στιγμή που πιέζεται το home button και το αποθηκέυει προσωρινά στην τοποθεσία : **directory/Library/Caches/Snapshots** (και το οποίο είναι προσβάσιμο από όλες τις εφαρμογές).

Σε περίπτωση που το κουμπί πατηθεί την στιγμή που ο χρήστης της συσκευής έχει πληκτρολογήσει ευαίσθητα δεδομένα (πχ κατά την πληκτρολογηση των κωδικών email ), αυτά θα είναι διαθέσιμα σε έναν επιτιθέμενο μέσω του screenshot.

### *Keyboard cache*

Οι Apple συσκευές προκειμένου να πετύχουν την αυτόματη συμπλήρωση των λέξεων καθώς ο χρήστης πληκτρολογεί, καταγράφει οτιδήποτε πληκτρολογείται στο **en_GB-dynamic-text.dat/el_GR-dynamic-text.dat** αρχείο το οποίο αποθηκεύεται στο : **Library/Keyboard**. Το αρχείο μπορεί εύκολα να διαβαστεί με την χρήση ενός HEX editor. Το iOS έχει προβλέψει πιθανή διαρροή πληροφοριών από αυτή την διαδικασία και δεν αποθηκεύει πεδία password καθώς και strings που αποτελούνται μονο από αριθμούς (pin/credit cards). Πλην όμως τα δεδομένα που πληκτρολογούνται σε διάφορα text πεδία, καταγράφονται κανονικά, επομένως Usernames και Security Question answers βρίσκονται σε αυτό το αρχείο, το οποίο και αυτό, είναι προσβάσιμο από όλες τις εφαρμογές.

### *Cookies*

Το iOS δημιουργεί το αρχείο Cookies.binarycookies προκειμένου να αποθηκεύει τα cookies των εφαρμογών. Οι περισσότερες εφαρμογές αποθηκεύουν τα session cookies τοπικά και επιπλέον, λόγω του ότι επιθυμούν να μην βάζουν τον χρήστη να μπαίνει στην διαδικασία εισαγωγής κωδικών κάθε φορά που εξέρχεται της εφαρμογής, δημιουργούν μόνιμα cookies. Τα cookies αυτά, αν υποκλαπούν είναι δυνατή η σύνδεση του επιτιθέμενου στην εφαρμογή με το λογαριασμό του χρήστη. Το Cookies.binarycookies είναι προσπελάσιμο από όλες τις εφαρμογές με το BinaryCookieReader.py που είναι ένα διαθέσιμο εργαλείο για την ανάγνωση του binary αρχείου των cookies.

### *Binary/Runtime Analysis*

Όλες οι εφαρμογές οι οποίες βρίσκονται στο Apple store, είναι κρυπτογραφημένες σε binary code όταν εγκαθίστανται στην συσκευή. Αντιθέτως οι εφαρμογές που δημιουργούνται από τρίτες οντότητες εκτός Apple Store, αποθηκεύονται μη κρυπτογραφημένες. Ο Loader του λειτουργικού συστήματος αποκρυπτογραφεί τον binary code όταν μια εφαρμογή φορτώνεται στην μνήμη της συσκευής. Με τα κατάλληλα εργαλεία debugger (πχ Craculous – Installous) είναι δυνατή η αντιγραφή της εφαρμογής από την μνήμη σε ένα αρχείο.

Ένα επιπλέον εργαλείο που είναι διαθέσιμο στους επιτιθέμενους είναι το Cycript το οποίο μπορεί να επέμβει στον κώδικα των εφαρμογών κατά την διάρκεια της εκτέλεσης (runtime) και να τροποποιήσει τα δεδομένα αυτών με αποτέλεσμα ο επιτιθέμενος να αποκτάει πρόσβαση σε ευαίσθητα δεδομένα του χρήστη.

#### *2.3.1.4 Επιθέσεις κατά των πρωτοκόλλων επικοινωνίας*

Οι εφαρμογές όπως έχουμε αναφέρει δεν μοιράζονται δεδομένα μεταξύ τους καθώς η κάθε μια τρέχει μέσα στο δικό της sandbox. Υπάρχουν όμως εφαρμογές οι οποίες επιθυμούν να μοιραστούν δεδομένα με άλλες εφαρμογές και γι αυτό δημιουργούν έναν δικό τους protocol handler. Η Apple χρησιμοποιεί τον όρο URLSchemes για να αναφέρεται στα protocol handlers. Στο URLScheme Reference document της εταιρείας, αναγράφονται όλοι οι protocol handlers που είναι εγεγραμμένοι μέσα στο λειτουργικό σύστημα iOS. Αν η υλοποίηση των URLSchemes δεν γίνει με προσοχή, είναι δυνατόν μια εφαρμογή να καλέσει ένα protocol handler χωρίς την

έγκριση του χρήστη (πχ Skype vulnerability [26]) με αποτέλεσμα την διαρροή ευαίσθητων πληροφοριών.

Τα URLSchemes που χρησιμοποιεί κάθε εφαρμογή καταγράφονται στο **Info.plist** file της το οποίο αποθηκεύεται στην διαδρομή : **/Music/iTunes/Mobile Applications.** Η καταληξη που χρησιμοποιείται είναι **appname.ipa** αλλά αμα μετονομαστεί σε **appname.zip** μπορεί να διαβαστεί εύκολα και ο επιτιθέμενος να οικειοποιηθεί τα URLSchemes των εφαρμογών.

#### *2.3.1.5 Επιθέση User Interface Spoofing*

Όλοι οι δημοφιλείς web browser δεν επιτρέπουν σε άλλα site να τροποποιούν (ή ακόμα και να κρύβουν),το URL που εμφανίζεται στο address bar, προκειμένου να αποτρέψουν τις UI Spoofing επιθέσεις. Λόγω της περιορισμένης οθόνης των apple κινητών συσκευών, είναι πιθανό ένα malware site να εμφανιστεί κατά την περιήγησή μας στο διαδίκτυο και να κρύψει εντελώς την address bar του safari. Αποτέλεσμα αυτού είναι οι χρήστες να νομίζουν ότι βρίσκονται σε ένα έγκυρο και εμπιστο site (πχ τράπεζας) αλλά αντι αυτού να είναι σε site επιτιθέμενου ο οποίος κατορθώνει να υποκλέψει τα προσωπικά δεδομένα των χρηστών με την εισαγωγή τους στο malwsare site.

#### *2.3.1.6 Επίθεση Man In The Middle*

Αρκετές iOS εφαρμογές χρησιμοποιούν το πρωτόκολλο HTTP για την επικοινωνία με server side εφαρμογές. Για την προστασία των δεδομένων από υποκλοπές, οι εφαρμογές χρησιμοποιούν το πρωτόκολλο SSL προκειμένου αυτή η επικοινωνία να είναι κρυπτογραφημένη και ασφαλής.

Σε περίπτωση που μια εφαρμογή προσπαθήσει να εγκαταστήσει μια SSL σύνδεση θα ζητηθεί από τον server το πιστοποιητικό του οργανισμού στον οποίο θέλει ο χρήστης να συνδεθεί. Έαν υπάρχει Man In The Middle επίθεση, το iOS (μέσω της NSError κλάσης) θα εμφανίσει μήνυμα αδυναμίας λήψης έγκυρου πιστοποιητικού και θα τερματίσει την σύνδεση. Αυτή είναι η επιθυμητή ενέργεια από το λειτουργικό σύστημα, όμως αρκετοί developers που δοκιμάζουν τις εφαρμογές κατά την ανάπτυξή τους, λόγω έλλειψης κατάλληλου πιστοποιητικού, παρακάμπτουν αυτήν την λειτουργία για τις δοκιμές και κατόπιν χωρίς να τροποποιήσουν τον κώδικα δίνουν την εφαρμογή στο εμπόριο, με αποτέλεσμα να είναι ευάλωτες στις Man In The Middle επιθέσεις.

#### *2.3.1.7 Επιθέσεις XSS*

Οι XSS επιθέσεις είναι παρόμειες με τις γνωστές cross site scripting επιθέσεις, που έχουν ως κύριο σκοπό την υποκλοπή του session του χρήστη. Πρόσφατο παράδειγμα XSS attack ανακαλύφθηκε στην εφαρμογή Skype [26] όπου οι εισερχόμενες κλήσεις εμφανίζονται σε ένα popup παράθυρο. Το web based παράθυρο χρησιμοποιεί ένα HTML πρότυπο (template) της συσκευής και φορτώνεται στην τοπική μνήμη αυτής. Η XSS επίθεση οδηγούσε στην υποκλοπή ολόκληρου του καταλόγου επαφών της συσκευής.

### *2.3.2 Επιθέσεις κατά των χαρακτηριστικών ασφαλείας των iOS συσκευών*

Πέρα από τις επιθέσεις που επικεντρώνονται στις εφαρμογές των Apple κινητών συσκευών και τις τρωτότητές τους, υπάρχει αριθμός επιθέσεων οι οποίες στοχέυουν σε αδυναμίες του ίδιου του λειτουργικού συστήματος με απώτερο σκοπό την υποκλοπή των ευαίσθητων δεδομένων των χρηστών.

#### *2.3.2.1 Παρακάμπτωντας το passcode*

Όπως έχει αναφερθεί, η εισαγωγή passcode κωδικού από τον χρήστη εενργοποιεί αυτόματα το Data Protection της συσκευής και ο κωδικός αποτελεί μέρος της δημιουργίας των κλειδιών κρυπτογράφησης των αρχείων στην μνήμη της συσκευής. Στα λειτουργικά συστήματα iOS προγενέστερα του iOS4, μια τρωτότητα στην bootrom (όταν η συσκευή βρίσκεται σε DFU mode) δίνει τη δυνατότητα για εκκίνηση της συσκευής χωρίς το passcode του χρήστη. Ο επιτιθέμενος , που έχει στην κατοχή του την κινητή συσκευή, αποκτά πρόσβαση στο file system, όμως απαιτείται ο passcode κωδικός προκειμένου να προσπελάσει τα αρχεία. Ο passcode δεν αποθηκεύεται πουθενά στην κινητή συσκευή επομένως η μόνη λύση είναι μια brute force επίθεση στον κωδικό. Ο default κωδικός passcode στα iOS είναι ένας 4ψήφιος αριθμός ο οποίος σπάει σε σχετικά μικρό χρονικό διάστημα. Όσο ο κωδικός γίνεται πιο περίπλοκος (μεγαλύτερο μεγεθος και εισαγωγή χαρακτήρων σε αυτόν) τόσο δυσκολεύει η επίθεση brute force. (εικόνα 16)

| Passcode Complexity | Brute force time |
|---|---|
| 4 digits | 18 minutes |
| 4 alphanumeric | 51 hours |
| 5 alphanumeric | 8 years |
| 8 alphanumeric | 13,000 years |

**Εικόνα 16 [2]**

Αν η Brute force επίθεση γίνει σε kernel level μέσω ενός custom ramdisk, τότε η επιπρόσθετη ασφάλεια που έχει θέσει το iOS για καθυστέρηση εισαγωγής νέου passcode ανάλογα με τις αποτυχημένες προσπάθειες (από λίγα λεπτά μέχρι αρκετές ημέρες), παρακάμπτεται και το Keybag ξεκλειδώνει εκθέτωντας στον επιτιθέμενο όλα τα διαθέσιμα κλειδιά.

### *2.3.2.2 Παρακάμπτωντας το ASLR*

Το ASLR μπορεί εύκολα να παρακαμφθεί και η συσκευή να είναι ευάλωτη σε επιθέσεις, με κατάλληλο configuration ενός και μόνο launchdeamon της συσκευής. Η συγκεκριμένη τρωτότητα είναι γνωστή ως **corona exploit.** Το exploit άλλαζε το configuration file του launchd αρχείου, ως εξής :

| Από | **Σε** |
|---|---|
| `<key>DisableAslr</key>`<br>`<false/>` | `<key>DisableAslr</key>`<br>`<true/>` |

Πίνακας 5

Αυτή η πολύ απλή αλλαγή του κώδικα οδηγούσε σε πλήρη απενεργοποίηση του ASLR, με την κινητή συσκευή να γίνεται ευάλωτη σε runtime analysis attacks. Η τρωτότητα ωστόσο διορθώθηκε στο λειτουργικό σύστημα iOS 5.1

### *2.3.2.3 Παρακάμπτωντας το Code Signing*

Τον Νοέμβριο του 2011 δημοσιεύτηκε μια τρωτότητα του iPhone που βασιζόταν σε αδυναμία της Javascript engine “Nitro” και η οποία πρωτοπαρουσιάστηκε στο iOS 4.3. Η μηχανή “Nitro” όταν τρέχει στον Safari browser αποκτά ειδικά δικαιώματα που της επιτρέπουν να εκτελεί Just in Time (JIT) compile του JS κώδικα, και μετά να τον εκτελεί. Για παράδειγμα μια απλή εφαρμογή αφού εγκατασταθεί στην κινητή συσκευή μπορεί να δημιουργήσει ένα reverse shell της συσκευής και να έχει πρόσβαση σε ολόκληρο το file system. [36]

#### *2.3.2.4 Παρακάμπτωντας την κρυπτογράφηση του hardware*

Ένας επιπλέον τρόπος για την κατάρρευση του Data Encryption χωρίς ο επιτιθέμενος να γνωρίζει τον passcode κωδικό, απαιτεί την φυσική πρόσβαση του επιτιθέμενου στη συσκευή. Εφαρμόζωντας μια Boot-ROM επίθεση και εγκαθιστώντας έναν SSH deamon στην συσκευή, επειδή το κλειδί κρυπτογράφησης File System Key είναι αποθηκευμένο στην συσκευή, και η συσκευή μπορεί να αποκρυπτογραφεί τα δεδομένα στο flash drive, όλα τα αρχεία είναι προσπελάσιμα μέσω του SSH login και μπορούν να αντιγραφούν από την συσκευή (η remote wipe υπηρεσία αποτρέπει αυτήν την επίθεση όπως αναλύσαμε παραπάνω)

#### *2.3.2.5 Παρακάμπτωντας το Keychain*

Πέρα της hardware κρυπτογράφησης για την προστασία των δεδομένων, υφίσταται και η Keychain κρυπτογράφηση η οποία δημιουργεί κλειδιά για κάθε μια εφαρμογή ξεχωριστά, με σκοπό την κρυπτογράφηση των δεδομένων τους. Για να μπορέσει μια εφαρμογή να προσπελάσει τα δεδομένα μιας άλλης εφαρμογής θα πρέπει να είναι καταχωρημένες και οι δύο στο ίδιο keychain access group.

Η πληροφορία αυτή (σε ποιο access group ανήκει μια εφαρμογή) βρίσκεται στην βάση δεδομένων που δημιουργεί το Keychain στην SQLite και μάλιστα είναι αποθηκευμένη χωρίς κρυπτογράφηση. Επομένως ο επιτιθέμενος μπορεί εύκολα να διαβάσει την εγγραφή για την εφαρμογή που τον ενδιαφέρει και να εγκαταστήσει μια εφαρμογή στην κινητή συσκευή με το ίδιο access group. Σε κανονικές συνθήκες η Apple δεν θα επέτρεπε την εγκατάσταση της εφαρμογής του επιτιθέμενου στην συσκευή, καθώς δεν θα είχε λάβει πιστοποιητικό από την εταιρεία ( η Apple απορρίπτει εφαρμογές που χρησιμοποιούν αυτήν την τακτική) . Πλήν όμως στις Jailbreaked συσκευές, η αυθεντικοποίηση των developers μέσω ψηφιακών υπογραφών είναι απενεργοποιημένη. Επομένως το μόνο που έχει να κάνει ο επιτιθέμενος είναι να υπογράψει ψηφιακά μόνος

του την εφαρμογή, να την εγκαταστήσει στην συσκευή και να αποκτήσει πρόσβαση στα αρχεία άλλων εφαρμογών.

## *2.4 Εκτίμηση Ασφαλείας*

Το μοντέλο ασφαλείας του iOS είναι καλοσχεδιασμένο, σε διάφορα επίπεδα ασφαλείας και αντιμετωπίζει ικανοποιητικά πολλών ειδών επιθέσεις. Παρέχει αυξημένη ασφάλεια ενάντια στα κλασσικά malware, αρχικά λόγω του sandboxing όπου μια μολυσμένη εφαρμογή δεν μπορεί να επηρρεάσει άλλες εφαρμογές, και κατά δεύτερο λόγο, μέσω της πιστοποίησης των developer που αναπτύσουν εφαρμογές για το App store.

Πλήν όμως, με διάφορες τεχνικές οι επιτιθέμενοι αποκτούν πρόσβαση στα δεδομένα των χρηστών. Τα πεδία οπου επικεντρώνονται οι επιθέσεις όπως είδαμε παραπάνω είναι το Data Storage της συσκευής καθώς και το Network security, προσπαθώντας να εκμεταλλεύτούν κυρίως κακές υλοποιήσεις των εφαρμογών στις λειτουργικότητες που προσφέρει μια iOS συσκευή.

Επιπλέον εάν ο επιτιθέμενος αποκτήσει φυσική πρόσβαση στην κινητή συσκευή (κλοπή ή απώλεια) τότε έχει αρκετές πιθανότητες στο να καταφέρει να προσπελάσει τις πληροφορίες της κινητής συσκευής, ειδικά σε περίπτωση που ο Passcode κωδικός ασφαλείας παραμένει στις default ρυθμίσεις του 4ψήφιου αριθμού.

Τέλος σημαντικός παράγοντας ασφαλείας, παραμένει ο ίδιος ο χρήστης. Η ελλειπής γνώση θεμάτων ασφαλείας, οδηγεί αριθμό χρηστών σε επικύνδινες αποφάσεις (μη χρήση passcode, jailbreaking) οι οποίες απλοποιούν σε μεγάλο βαθμό το έργο των επιτιθέμενων.

## *2.5 Συμβουλές ανάπτυξης ασφαλών εφαρμογών iOS*

Η ασφάλεια των εφαρμογών θα πρέπει να αποτελεί κύριο μέλημα των developers και των εταιρειών που αναπτύσουν λογισμικά για τις κινητές συσκευές. Το iOS προσφέρει πολλές λειτουργικότητες μέσω του μοντέλου ασφαλείας που έχει υλοποιηθεί σε αυτό και η ορθή τους χρήση εξασφαλίζει την ασφάλεια των ευαίσθητων δεδομένων των χρηστών.

Ο παρακάτω πίνακας παραθέτει τεχνικές και συμβουλές (με βάση τις επιθέσεις που αναφέρθηκαν παραπάνω) που θα πρέπει να έχουν οι developers υπόψη κατά την

ανάπτυξη των εφαρμογών τους, προκειμένου να επιτύχουν το μέγιστο βαθμό ασφαλείας των δεδομένων που αποθηκεύονται στις συσκευές.

| | |
|---|---|
| Plist Files | Μην αποθηκεύετε ευαίσθητα δεδομένα στα Plist files. Αν απαιτείται κάτι τέτοιο για την λειτουργικότητα της εφαρμογής, να γίνει χρήση κρυπτογράφησης. |
| | Χρησιμοποιήστε το Data Protection για την προστασία των plist files |
| | Δημιουργείστε τον υποφάκελλο Library/Caches για τα plist files. Το iTunes δεν εκτελεί backup τους caches φακέλους των εφαρμογών. |
| | Πριν την διαγραφή των plist files (εάν αυτό επιλεγεί ως πρόσθετο μέτρο ασφάλειας) αντικαταστείστε τα data των αρχείων με μη έγκυρες τιμές. |
| SQLite Files | Μην αποθηκεύετε ευαίσθητα δεδομένα στην βάση δεδομένων χωρίς κρυπτογράφηση |
| | Χρησιμοποιήστε το Data Protection για την προστασία των SQLite files |
| | Πριν την διαγραφή των SQLite records ή των SQLite files (εάν αυτό επιλεγεί ως πρόσθετο μέτρο ασφάλειας) αντικαταστείστε τα data με μη έγκυρες τιμές. |
| | Χρησιμοποιείστε την εντολή VACUUM της SQL για την διαγραφή των δεδομένων από την βάση. |
| Keychain | Μην αποθηκευετε ευαίσθητα δεδομένα χωρίς κρυπτογράφηση |
| | Χρησιμοποιήστε το Data Protection API κατά την αποθήκευση δεδομένων στο Keychain |
| | Μην αποθηκεύετε κρυπτογραφικά κλειδιά στην βάση. |
| Syslog | Μην αποθηκεύετε ευαίσθητα δεδομένα στα log files των εφαρμογών. |
| Screenshot | Τροποποιήστε τα δεδομένα ή αλλάξτε την οθόνη πριν εκτελεστεί η “applicationDidEnterBackground()” λειτουργία, η οποία εκτελείται |

| | |
|---|---|
| | λίγο πριν το iOS λάβει screenshot από την εφαρμογή. |
| Keyboard cache | Απενεργοποιείστε την λειτουργία autocomplete των πεδίων που χειρίζονται ευαίσθητα δεδομένα.<br>mytextField.secureTextEntry = YES |
| | Απενεργοποιείστε την αυτόματη διόρθωση από τα πεδία που χειρίζονται ευαίσθητα δεδομένα<br>mytextField.autocorrectionType = UITextAutocorrectionTypeNo; |
| Cookies | Για εφαρμογές που χειρίζονται ευαίσθητα δεδομένα, μην χρησιμοποιείτε μονιμα cookies. |
| Runtime analysis | Μην αποθηκεύετε κρυπτογραφικά κλειδιά στην μνήμη |
| Man In The Middle | Μην υλοποιείτε εφαρμογές που χρησιμοποιούν το API “setAllowsAnyHTTPSCertificate” για να δέχονται μη πιστοποιημένες συνδέσεις. |
| Protocol Handling Attacks | Ελέξτε την υλοποίηση των *URLSchemes* . |
| UIWebView | Ελέξτε τις υλοποιήσεις των Web views των εφαρμογών. Το URL πρέπει να είναι πάντοτε ορατό. |
| Copy-paste | Απενεργοποιήστει την λειτουργικότητα copy-paste για πεδία που χειρίζονται κρίσιμες πληροφορίες. |

Πίνακας 6

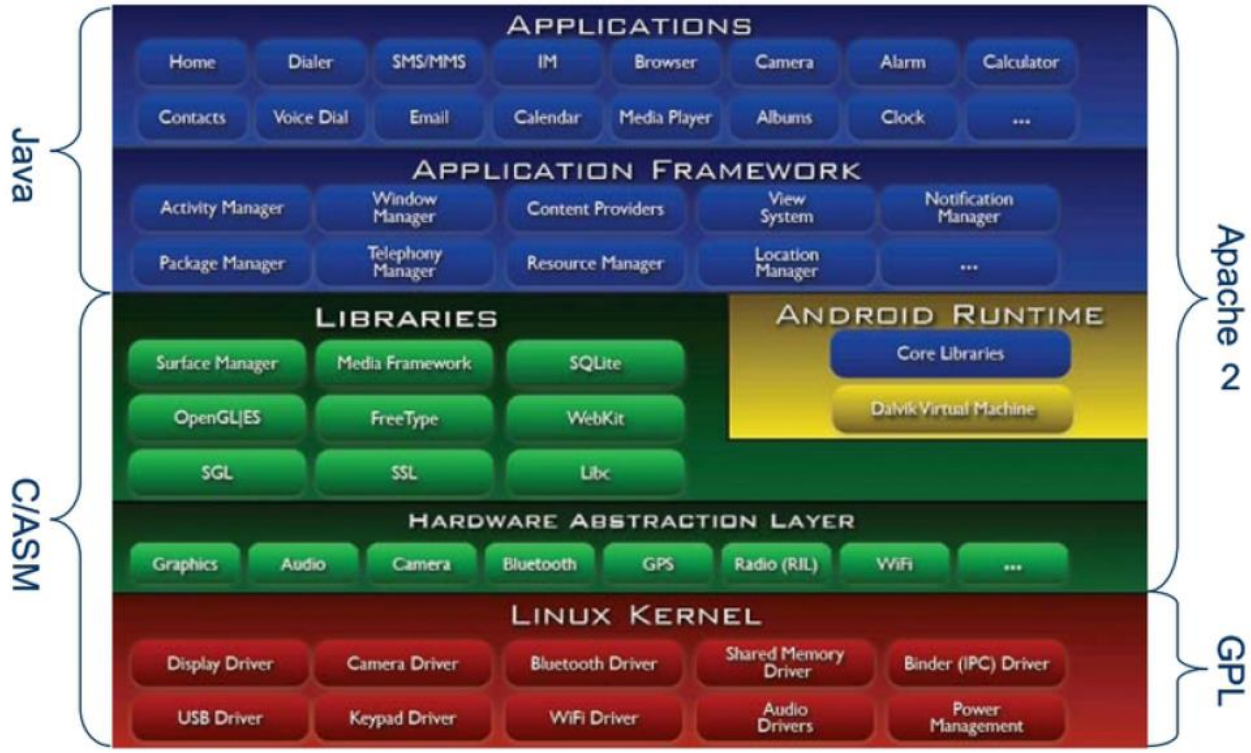
# ΚΕΦΑΛΑΙΟ 3

# ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ Android

## *3.1 ANDROID - Αρχιτεκτονική*

Το Android απότελεί το δημοφιλέστερο λειτουργικό σύστημα για κινητές συσκευές της αγοράς. Κατέχει περιπου το 70% της αγοράς στα smartphones και το 42% σε tablets.

Η αρχιτεκτονική του βασίζεται στον Linux Kernel και διαιρείται στα παρακάτω λογικά επίπεδα.




**Εικόνα 17 [37]**

### *3.1.1 Επίπεδο Linux Kernel*

Το πιο χαμηλό επίπεδο του Android είναι ο πυρήνας του Linux . Αποτελεί την καρδιά του συστήματος και ποτέ δεν αλληλεπιδρά απευθείας με τον χρήστη καθώς παρεμβάλονται τα υπόλοιπα επίπεδα. Οι λειτουργικότητες που προσφέρει το επίπεδο αυτό είναι

Αφαιρετικές δομές του hardware

Προγράμματα διαχείρισης της μνήμης

Ρυθμίσεις ασφαλείας

Προγράμματα διαχείρισης ενέργειας

Hardware drivers

Ρυθμίσεις δικτύου

Υποστηρικτικές δομές για Shared Libraries

Ο πυρήνας του Android αναπτύσσεται ταχέως και συνεχώς εμφανίζονται νέες εκδόσεις αναβαθμίζοντας διαρκώς την λειτουργικότητα των παρεχόμενων υπηρεσιών.

| Android Version | Linux Kernel Version |
|---|---|
| 1.0 | 2.6.25 |
| 1.5 (Cupcake) | 2.6.27 |
| 1.6 (Donut) | 2.6.29 |
| 2.2 (Froyo) | 2.6.32 |
| 2.3 (Gingerbread) | 2.6.35 |
| 3.0 (Honeycomb) | 2.6.36 |
| 4.0.x (Ice Cream Sandwich) | 3.0.1 |
| 4.1./4.2 (Jelly Bean) | 3.0.31 |

**Εικόνα 18 [37]**

Ο πηρύνας αυτή την στιγμή χρησιμοποιεί έναν Linux 3.0 kernel, τροποποιημένο κατάλληλα για την διαχείριση των ειδικών αναγκών ενέργειας, μνήμης και του περιβάλλοντος εκτέλεσης των εφαρμογών σε μια κινητή συσκευή.

### *3.1.2 Επίπεδο Native Libraries*

Το αμέσως επόμενο επίπεδο περιλαβμάνει όλες τις τοπικές βιβλιοθήκες του Android. Κάθε μία βιβλιοθήκη περιέχει ένα σύνολο οδηγιών για την διαχείριση των διαφόρων τύπων δεδομένων που είναι διαθέσιμοι στο λειτουργικό σύστημα. Οι σημαντικότερες βιβλιοθήκες του Android είναι οι παρακάτω:

Surface manager: Για την δημιουργία παραθύρων επικοινωνίας στην οθόνη των συσκευών

SGL : Βιβλιοθήκες για 2D γραφικά

OPEN GL/EG : Βιβλιοθήκες για 3D γραφικά

Media Framework: Υποστηρίζει playback και εγγραφή για ήχο, video και εικόνα διαφόρων format

Free Type: Διαμόρφωση γραμματοσειρών

WebKit: Browser engine

Libc : Βιβλιοθήκες συστήματος της C

SQLite : Η βάση δεδομένων της συσκευής

OpenSSL : Για την υλοποίηση των SSL συνδέσεων

Στο ίδιο επίπεδο συναντάμε και το **Android runtime** επίπεδο το οποίο περιλαμβάνει ένα σύνολο από Java βιβλιοθήκες καθώς και την **Dalvik Virtual** μηχανή μέσα στην οποία εκτελούνται οι εφαρμογές.

**Dalvik Virtual Machine :** Η Dalvik VM είναι ένα open source πρόγραμμα, στο οποίο βασίζεται όλη η λειτουργία του λειτουργικού συστήματος Android.


**Εικόνα 19 [37]**

Έιναι ένα Virtual Machine με κατάλληλες ρυθμίσεις για να τρέχει σε περιβάλλοντα με μικρές απαιτήσεις μνήμης, όπως αυτά των κινητών συσκευών. Είναι σχεδιασμένο για να επιτρέπει πολλαπλά VM instances να τρέχουν ταυτόχρονα. Το λειτουργικό σύστημα πάνω στο οποίο τρέχει ένα Dalvik VM είναι υπεύθυνο για την απομόνωση των διεργασιών, την διαχείριση της μνήμης και την υποστήριξη του threading.

### *3.1.3 Επίπεδο Application Framework*


**Εικόνα 20 [37]**

Το Application Framework επίπεδο προσφέρει όλες τις βασικές λειτουργικότητες που χρειάζονται οι εφαρμογές κατά την εκτέλεσή τους οι βασικότερες εκ των οποίων είναι οι εξής:

Activity Manager : Διαχειρίζεται τον κύκλο ζωής των εφαρμογών.

Content Providers : Διαχειρίζεται τα δεδομένα που διαμοιράζονται οι εφαρμογές.

Telephony Manager : Διαχειρίζεται όλες τις τηλεφωνικές συνδιαλέξεις. Η χρήση του manager γίνεται για να μπορέσει μια εφαρμογή να έχει πρόσβαση στις τηλεφωνιές κλήσεις.

Location Manager : Διαχείριση των συντεταγμένων τοποθεσίας της συσκευής με την χρήση του GPS ή με την χρήση των κεραιών κινητής τηλεφωνίας.

Resource Manager : Διαχείριση των διαφόρων πόρων που χρησιμοποιούν οι εφαρμογές.

### *3.1.4 Επίπεδο Application*

Το Application επίπεδο αποτελεί την κορυφή της αρχιτεκτονικής του Android. Ο μέσος χρήστης των Android συσκευών, επικοινωνεί κυρίως με αυτό το επίπεδο κάνωντας χρήση των υπηρεσιών που προσφέρουν τα υποκείμενα επίπεδα (τηλεφωνικές συνδιαλέξεις, σερφάρισμα στο διαδίκτυο κλπ)


**Εικόνα 21 [37]**

Αρκετές εφαρμογές είναι προεγκατεστημένες σε κάθε συσκευή, όπως η αποστολή/λήψη SMS μηνυμάτων, η διαχείριση τηλεφωνικών κλήσεων, ο web Browser και η διαχείριση του καταλόγου επαφών.

## *3.2 Μοντέλο Ασφαλείας*

Στην καρδιά του Android βρίσκεται ο Linux kernel επομένως αρκετά χαρακτηριστικά του Android μοντέλου ασφαλείας βασίζονται στο αντίστοιχο μοντέλο του Linux.

### *3.2.1 Linux*

Βασικό στοιχείο του μοντέλου ασφαλείας του Linux αποτελούν οι χρήστες και τα γκρούπ των χρηστών. Όταν δημιουργείται ένας χρήστης στο Linux OS αυτόματα

αντιστοιχίζεται σε αυτόν, ένα user ID (UID). Ο μοναδικός αυτός αριθμός χρησιμεύει για να διαχωρίζονται οι χρήστες μεταξύ τους. Αντίστοιχα κάθε χρήστης μπορεί να είναι μέλος ενός ή πολλών γκρούπ (user groups), σε κάθε ένα από το οποίο αντιστοιχεί ένας παρόμοιος χαρακτηριστικός αριθμός group ID (GID).

Τα δικαιώματα πρόσβασης του Linux ανατίθενται σε κάθε αρχείο ξεχωριστά. Κάθε αρχείο (σχεδόν τα πάντα στο Linux αντιστοιχούν σε ένα αρχείο) έχει έναν “ιδιοκτήτη”, ο οποίος αντιστοιχεί σε ένα UID που αντικατοπτρίζει τον χρήστη που έχει τα πρωταρχικά δικαιώματα επί του αρχείου και μπορεί να τα τροποποιήσει όπως αυτός επιθυμεί. Επιπλέον σε κάθε αρχείο αντιστοιχίζεται και από ένα GID που καθορίζει ποιο user group κατέχει ένα σύνολο δικαιωμάτων επί του αρχείου.

Αναλυτικότερα, κάθε αρχείο του Linux έχει τρείς κατηγορίες δικαιωμάτων :

Owner : Καθορίζει τα δικαιώματα επι του αρχείου που έχει ο “ιδιοκτήτης” του αρχείου

Group : Καθορίζει το σύνολο των δικαιωμάτων που έχει το συγκεκριμένο user group επι του αρχείου.

World : Καθορίζει τα δικαιώματα που έχουν όλοι οι άλλοι χρήστες που δεν βρίσκονται στις παραπάνω 2 κατηγορίες.

Κάθε ένα από τα παραπάνω δικαιώματα μπορεί να περιλαμβάνει τις λειτουργίκότητες READ – WRITE – EXECUTE που αντιστοιχούν στα δικαιώματα ΑΝΑΓΝΩΣΗΣ-ΕΓΓΡΑΦΗΣ-ΕΚΤΕΛΕΣΗΣ επί του αρχείου.

Το μοντέλο ασφαλείας του Linux βασίζεται στην ιδέα, ότι αν ένα δικαίωμα πρόσβασης δεν έχει ανατεθεί σε μια κατηγορία, τότε αυτή η κατηγορία δεν έχει μπορεί να αποκτήσει το συγκεκριμένο δικαίωμα.

### *3.2.2 Android*

Το λειτουργικό σύστημα Android, όπως είναι αναμενόμενο, βασίζεται στο μοντέλο ασφαλείας του Linux. Η κεντρική ιδέα παραμένει το User ID και το Group ID, όπου κάθε χρήστης μπορεί να είναι μέλος σε ένα η περισσότερα User Groups.

Όταν μια εφαρμογή εγκαθίσταται σε μια συσκευή Android, το ΛΣ δημιουργεί ένα νέο χρήστη (διαφορετικό από τον χρήστη που είναι συνδεδεμένος εκείνη τη στιγμή στη συσκευή) και του ανατίθεται ένα μοναδικό UID. Όλοι οι πόροι που δημιουργούνται από την εφαρμογή (αρχεία, βάση δεδομένων κλπ) τους ανατίθεται ο

συγκεκριμένος UID αριθμός , με πλήρη δικαιώματα πρόσβασης. Για κάθε άλλο UID, πλην αυτού που δημιουργήθηκε κατά την εγκατάσταση, τα αρχεία της εφαρμογής δεν είναι προσπελάσιμα.

Η βασική αρχή λειτουργίας του Android (και του Linux) είναι ο διαχωρισμός των δικαιωμάτων πρόσβασης ανάλογα με το UID που ανατίθεται σε κάθε πόρο του συστήματος. Εφαρμογές που τρέχουν κάτω από διαφορετικό UID, δεν μπορούν να προσπελάσουν η μια τα αρχεία της άλλης. Εφαρμογές με το ίδιο UID έχουν την δυνατότητα διαμοιρασμού δεδομένων και αλληλεπίδρασης μεταξύ τους.

### *3.2.3 Χαρακτηριστικά Ασφαλείας Android*

**Sandboxing:** Το Sandboxing απομονώνει τις εφαρμογές κατά την εκτέλεσή τους. Κάθε εφαρμογή στο Android εκτελείται μέσα στο δικό της instance του Dalvik VM. Επίσης μια εφαρμογή μπορεί να περιλαμβάνει και κώδικα του λειτουργικού συστήματος (για λειτουργικότητες που προσφέρει το ΛΣ στις εφαρμογές) ο οποίος εκτελείται απευθείας στον επεξεργαστή της συσκευής.

**Application Signing :** Όλες οι εφαρμογές που εγκαθίσταται σε μια συσκευή Android πρεπει να διαθέτουν ψηφιακή υπογραφή του developer. Το Android επιτρέπει την χρήση ψηφιακών υπογραφών που είναι πιστοποιημένες και ελεγμένες από τις αρχές πιστοποίησης (CAs) αλλά και την χρήση πιστοποιητικών που είναι υπογεγραμμένα από τους ίδιους τους developers. Εφαρμογές που είναι υπογεγραμμένες από τον ίδιο developer μπορούν να κάνουν χρήση κοινών δεδομένων και διεργασιών.

**Component Encapsulation :** Κάθε εφαρμογή αποτελείται από ένα ή πολλά components τα οποία μπορεί να οριστούν ως public ή private μέσω του AndroidManifest.xml αρχείου. Τα private components είναι διαθέσιμα από άλλα components της ίδιας εφαρμογής, ενώ τα public μπορούν να προσπελαστούν και να αλληλεπιδράσουν με components άλλων εφαρμογών.

**Permission Mechanism :** Η διαχείριση των δικαιωμάτων αποτελεί βασικό στοιχείο της ασφάλειας των Android συσκευών. Δικαιώματα αποδίδονται σε components των εφαρμογών προκειμένου να καθοριστεί το επίπεδο πρόσβασης που απαιτείται από άλλες εφαρμογές για να προσπελάσουν τα δεδομένα της εφαρμογής.

**ASLR:** Το Android όπως και το iOS προσφέρει την υπηρεσία ASLR (με PIE) προκειμένου να αυξήσει το επίπεδο ασφαλείας των εφαρμογών κατά την εκτέλεσή τους.

Προκειμένου να γίνει κατανοητό το μοντέλο ασφαλείας του Android και το πως εφαρμόζονται οι πολιτικές ασφαλείας στις εφαρμογές, θα αναλύσουμε τον τρόπο που οι εφαρμογές είναι χτισμένες και πως αυτές επικοινωνούν μεταξύ τους εξασφαλίζοντας την ασφάλεια των δεδομένων τους.

### *3.2.4 Δομή – Ασφάλεια των εφαρμογών*

Όπως προαναφέρθηκε, σε κάθε εφαρμογή, κατά την εγκατάστασή της, ανατίθεται ένα μοναδικό UID. Επίσης είναι δυνατόν να καθοριστεί μοναδικό UID το οποίο θα διαμοιράζεται σε περισσότερες από μία εφαρμογές. Ένας developer ο οποίος επιθυμεί οι εφαρμογές του να επικοινωνούν μεταξύ τους και να διαμοιράζονται δεδομένα, καθορίζει μοναδικό UID στο AndroidManifest.xml αρχείο των εφαρμογών του. Με τον τρόπο αυτό δηλώνει στο λειτουργικό σύστημα ότι αν στο μέλλον ο χρήστης εγκαταστήσει μια νέα εφαρμογή από τον ίδιο developer, θα εγκατασταθεί με το ίδιο UID και θα μπορεί να διαμοιράζεται δεδομένα. Το tag που χρησιμοποιείται για την λειτουργικότητα είναι :

*android:sharedUserID=”myname.sharedUID”*

Όλες οι εφαρμογές που έχουν το ίδιο *sharedUserID attribute* θα πρέπει να είναι ψηφιακά υπογεγραμμένες από την ίδια οντότητα (developer ή εταιρεία) αλλιώς η εγκατάσταση της εφαρμογής θα καταρρεύσει.

### *Components*

Οι Android εφαρμογές αποτελούνται από ένα η περισσότερα *components* . Τα διαθέσιμα components που υποστηρίζει το λειτουργικό σύστημα είναι :

*Activities* : Τα activities αντιστοιχούν σε μια οθόνη της εφαρμογής που βλέπει ο χρήστης κατά την εκτέλεσή της και αντιστοιχεί στο Επίπεδο Παρουσίασης της εφαρμογής.

*Services* : Το component service είναι σχεδιασμένο για την λειτουργία διεργασιών στο παρασκήνιο των εφαρμογών και μπορεί να τρέχει ενώ η εφαρμογή δεν είναι ορατή στον χρήστη.

*Content Provider :* Τα Content Provider προσφέρουν την λειτουργικότητα διαμοιρασμού δεδομένων μεταξύ των εφαρμογών. Αποτελούν τα public interfaces των βάσεων δεδομένων, επιτρέπωντας σε άλλες εφαρμογές να έχουν πρόσβαση σε αυτές.

*Broadcast Receiver* : Τα broadcast receivers components λειτουργούν ως παραλήπτες των μηνυμάτων του λειτουργικου συστήματος που λέγονται Intents.

*Intents* : Τα Intents δεν είναι components, αλλά είναι αιτήσεις που στέλνουν οι εφαρμογές/components προκειμένου να εκτελέσουν μια συγκεκριμένη ενέργεια. Αποτελούν το βασικό σύστημα επικοινωνίας μεταξύ των εφαρμογών και η σωστή διαχείρισή τους αποτελεί σημαντικό κομμάτι της ασφάλειας των συσκευών Android.

### *Public-Private Components*

Όπως αναφέρθηκε παραπάνω τα components μπορεί να είναι δημόσια ή ιδιωτικά των εφαρμογών. Τα δημόσια components μπορούν να αλληλεπιδράσουν με άλλα components αλλων εφαρμογών, ενώ τα ιδιωτικά είναι προσπελάσιμα μόνο από τα components της ίδιας της εφαρμογής.

Τα components ενεργοποιούν το public/private προφίλ τους μέσα από το Comfiguration της εφαρμογής στο AndroidManifest.xml αρχείο με τρείς τρόπους:

α. Δηλώνοντας το compoment ως exported = true

β. Θέτωντας ένα Intent-filter στη δήλωση του component, αυτόματα το component τροποποιεί την ιδιότητα <exported> ως true.

### *Εσωτερική Επικοινωνία μεταξύ των components*

Όπως αναφέραμε τα components των εφαρμογών χρησιμοποιούν τις αιτήσεις Intents προκειμένου να επικοινωνήσουν μεταξύ τους. Μια εφαρμογή δημιουργεί ένα intent προκειμένου να στείλει ένα μήνυμα στις άλλες εφαρμογές ή components για να τους ειδοποιήσει ότι επιθυμεί να εκτελεστεί μια ενεργεια (activity / service) που αυτές προσφέρουν.

Αναλυτικότερα όταν μια εφαρμογή δημιουργήσει μια αίτηση (intents) είτε την στέλνει απευθείας σε ένα component που προσφέρει μια υπηρεσία (explicit intent), είτε την στέλνει προς όλα τα components που μπορούν να ανταποκριθούν στην αίτηση αυτή (implicit intent). Για να καθορίσουν τα components σε ποια μήνυματα Intent

μπορούν να ανταποκριθούν, το Android υποστηρίζει την λειτουργία των **INTENT-FILTERS.**

Με την δημιουργία ενός **intent-filter** στο configuration ενός component, πέρα του ότι το component τίθεται αυτομάτως public, δηλώνονται και ποια intents μηνύματα είναι πρόθυμο να δεχτεί (και συνεπώς ποιες λειτουργικότητες προσφέρει). Με τον τρόπο αυτό, δεν κοινοποιούνται προς τις άλλες εφαρμογές τα intents, απλά εδώ γίνεται η δήλωσή τους. Ένα άλλο component που θέλει να εκτελέσει την λειτουργικότητα αυτή, στέλνει το συγκεκριμένο Intent στο component (ο developer γνωρίζει εκ των προτέρω ότι το συγκεκριμένο component δέχεται intents του τύπου Χ) και αιτείται την πρόσβαση σε αυτό.

Για να ειδοποιήσει μια εφαρμογή ότι έχει μια συγκεκριμένη δυνατότητα, και όχι απλά να την δηλώσει μέσω intent-filters, δημιουργει ένα intent και το κοινοποιεί (**broadcastIntent**) προς όλες τις εφαρμογές της συσκευής. Οι εφαρμογές που επιθυμούν να εκτελέσουν την προσφερόμενη ενέργεια θα στείλουν με την σειρά τους ένα intent στο συγκεκριμένο component.

Τα **broadcast receivers**, είναι τα components που παρατηρούν την παρουσία αυτών των “δημοσιεύσεων” των intents στο λειτουργικό σύστημα, και επιλέγουν αν θα εκτελέσουν την ενέργεια που έχει ζητηθεί. Για κάθε αίτηση intent μπορεί να ανταποκριθεί και να εκτελέσει την αιτούμενη ενέργεια παραπάνω από ένα **Broadcast Receiver** component. Υπάρχουν δύο τρόποι για να περιοριστούν οι αποδέκτες των Broadcast Receivers:

A. Στο configuration του broadcast intent (στο AndroidManifest.xml αρχείο της εφαρμογής) δηλώνεται ποιο permission πρέπει να έχει ένα Broadcast Receiver για να μπορέσει να ανταποκριθεί στην αίτηση.

B. Μέσω των Intent-filters των Broadcast Receivers, όπου μπορεί να καθοριστεί ποια Intents που έχουν γίνει broadcast, είναι πρόθυμο να δεχτεί φιλτράροντας έτσι τις εισερχόμενες αιτήσεις από όλες τις εφαρμογές που τρέχουν στην συσκευή.

### *Permissions*

Όπως είδαμε το Android έχει μια πολύ΄διαφορετική προσεγγισή για την εκτέλεση των εφαρμογών, από ότι υφίσταται στα κλασσικά desktop μηχανήματα όπου όλες οι εφαρμογές εκτελούνται κάτω από τον ίδιο χρήστη και με τα ίδια δικαιώματα

πρόσβασης στο filesystem. Το Android επιτρέπει κάθε εφαρμογή να τρέχει κάτω από διαφορετικό χρήστη (UID) και με τα δικαιώματα πρόσβασης που κατέχει ο συγκεκριμένος χρήστης. Μια εφαρμογή δεν μπορεί να προσπελάσει τα δεδομένα μιας άλλης, αν δεν είναι και οι δύο εφαρμογές ψηφιακά υπογεγραμμένες από την ίδια ονταότητα. Με αυτόν τον τρόπο επιτυγχάνεται ο διαχωρισμός και απομόνωση των εφαρμογών κατά την εκτέλεσή τους.

Πέραν όμως των παραπάνω, το Android παρέχει και ένα επιπλέον επίπεδο ασφάλειας προκειμένου να γίνεται έλεγχος των διαθέσιμων λειτουργιών και υπηρεσιών του συστήματος που αιτείται μια εφαρμογή να έχει πρόσβαση κατά την εγκατάσταση της.

Οι υπηρεσίες στις οποίες αιτείται πρόσβαση μια εφαρμογή αναγράφονται αναλυτικά στο AndroidManifest.xml αρχείο της. Πριν την εγκατάσταση, εμφανίζεται στο χρήστη το κείμενο με τις αιτούμενες άδειες πρόσβασης της εφαρμογής, και αν ο χρήστης συμφωνήσει τότε προχωράει η εγκατάσταση στην συσκευή. Αν ο χρήστης διαφωνήσει με τα δικαιώματα πρόσβασης που αιτείται η εφαρμογή, την απορρίπτει και δεν γίνεται η εγκατάστασή της.

Ένα δικαίωμα (permission) κατοχυρώνεται σε μια εφαρμογή και απαιτείται από τα APIs προκειμένου αυτά να εκτελεστούν. Για παράδειγμα αναφέρουμε το permission INTERNET. Όλα τα APIs που χρησιμοποιούν το διαδίκτυο απαιτούν από τις εφαρμογές που τα καλούν να έχουν κατοχυρώσει το δικαίωμα αυτό προκειμένου να τους επιτρέψουν την πρόσβαση και εκτέλεσή τους. Υπάρχουν 2 είδη permissions στο λειτουργικό σύστημα Android ανάλογα με την οντότητα που τα δημιουργεί.

**System Permissions** : αποτελούν τα δικαιώματα πρόσβασης για τα APIs που διαθέτει το λειτουργικό σύστημα στις εφαρμογές για να ολοκληρώσουν τις επιθυμητές λειτουργίες τους.

**Application Permissions** : Οι εφαρμογές έχουν την δυνατότητα να δημιουργούν δικά τους δικαιώματα πρόσβασης, προκειμένου δικές τους λειτουργικότητες να είναι διαθέσιμες από άλλες εφαρμογές. Η εφαρμογή για να είναι ασφαλής θα πρέπει να ελέγχει αν η καλούσα εφαρμογή έχει το κατάλληλο δικαίωμα πρόσβασης για το συγκεκριμένο API προκειμένου να επιτρέψει την εκτέλεσή του.

Επομένως, μια εφαρμογή αν είναι μόνο καταναλωτής υπηρεσιών του συστήματος ή αλλων εφαρμογών, το μόνο που έχει να κάνει είναι να δηλώσει τα δικαιώματα πρόσβασης που αιτείται στο AndroidManifest.xml αρχείο. Αν όμως επιθυμεί

την αλληλεπίδραση και τον διαμοιρασμό δεδομένων με άλλες εφαρμογές μέσω δημοσίων μεθόδων και Content Providers θα πρέπει να ορίσει τα δικαιώματα που απαιτεί από τις άλλες εφαρμογές για να την προσπελάσουν. Τα δικαιώματα που ορίζουν οι εφαρμογές για τα δημόσια component ανήκουν σε 4 κατηγορίες που έχει ορίσει το λειτουργικό σύστημα προκειμένου αυτά να προβάλλονται ανάλογα τις επικυνδινότητας τους στους χρήστες πριν αυτοί αποφασίσουν για την εγκατάσταση της εφαρμογής. Οι κατηγορίες που έχει ορίσει το Android είναι:

***Normal*** : Τα permissions που ανήκουν σε αυτή την κατηγορία δεν μπορούν να προκαλέσουν επικύνδινες για τα δεδομένα του χρήστη ενέργειες (πχ η αλλαγή του background της συσκευής) .

***Dangerous*** : Τα δικαιώματα αυτής της κατηγορίας όταν αποδωθούν σε μια εφαρμογή, αυτή θα μπορεί να προσπελάσει ευαίσθητα δεδομένα του χρήστη (κατάλογος επαφών, δημιουργία συνδέσεων κλπ) και ο χρήστης είναι υποχρεωμένος να εγκρίνει αυτά τα δικαιώματα στην εφαρμογή, προτού προχωρήσει στην εγκατάστασή της.

***Signature*** : Σε αυτή την κατηγορία ανήκουν τα δικαιώματα που αποδίδονται αυτόματα σε μια εφαρμογή, όταν αυτή είναι ψηφιακά υπογεγραμμένη από την ίδια οντότητα με την εφαρμογή που παρέχει το συγκεκριμένο permission. Με αυτόν τον τρόπο οι developers εξασφαλίζουν ότι οι εφαρμογές τους όταν εγκατασταθούν στην ίδια συσκευή θα μπορούν να επικοινωνούν μεταξύ τους και να ανταλλάσουν δεδομένα.

***SignatureOrSystem*** : Τα συγκεκριμένα permissions ακολουθούν τον ίδιο κανόνα με τα Signature, με την διαφορά ότι αποδίδονται αυτόματα και στα APIs του λειτουργικού συστήματος προκειμένου οι εφαρμογές να είναι σε θέση να αλληλεπιδρούν με αυτά.

Με τον ορισμό των παραπάνω δικαιωμάτων οι developers έχουν τον πλήρη έλεγχο για το ποια API πρόκειται να έχουν πρόσβαση στα δεδομένα των εφαρμογών που αναπτύσουν και οι χρήστες έχουν ολοκληρωμένη εικόνα ως προς τους κυνδίνους ασφαλείας προτού εγκαταστήσουν μια εφαρμογή στην κινητή συσκευή τους καθώς ενημερώνονται για τα δικαιώματα που αιτείται κάθε εφαρμογή μέσω του AndroidManifest.xml αρχείου.

### *Ασφάλεια των Activities*

Όπως αναφέρθηκε παραπάνω τα Activities αποτελούν το Presentation layer των εφαρμογών. Η αρχική οθόνη μιας εφαρμογής, είναι πάντοτε public και όλες οι εφαρμογές μπορούν να την εκκινήσουν. Πλην όμως, τα ενδιάμεσα επίπεδα της εφαρμογής δεν θα πρέπει να είναι δημόσια, καθώς μπορεί να χειρίζονται ευαίσθητα δεδομένα χρηστών (πχ το παράθυρο για την εισαγωγή του username και password). Γι αυτό οι εφαρμογές έχουν την δυνατότητα να απαιτήσουν συγκεκριμένα permissions που θα πρέπει να έχει η καλλούσα εφαρμογή προκειμένου να εκτελέσει το Activity.

Δύο ειδών permissions έχουν τα activities του Android, **START και STOP Activity.** Η εφαρμογή που επιθυμεί να εκκινήσει ένα Activity μια άλλης εφαρμογής, δημιουργεί ένα Intent προς αυτό. Αν το Activity δεν προστατεύεται από κάποιο permission, η καλούσα εφαρμογή θα εκκινήσει κανονικά το Activity. Αν προστατεύεται (η εφαρμογή που ανήκει το Activity έχει ορίσει το ανάλογο δικαίωμα) τότε το Android θα ελέγξει αν η καλούσα εφαρμογή έχει κατοχυρώσει το δικαίωμα πρόσβασης (μέσω του AndroidManifest.xml αρχείου) και θα επιτρέψει την εκτέλεση του Activity.

### *Ασφάλεια των Services*

Τα services αποτελούν τις διεργασίες που εκτελούνται στο παρασκήνιο των εφαρμογών γι αυτό θα πρέπει να γίνεται αυστηρός έλεγχος ποιες εφαρμογές θα έχουν δικαιώματα αλληλεπίδρασης με τα services αλλων εφαρμογών, καθώς αυτή γίνεται χωρίς να το γνωρίζει ο χρήστης. Μια εφαρμογή μπορεί να δημιουργήσει ένα service ή να χρησιμοποιήσει ένα service μιας άλλης εφαρμογής.

Για να έχει τον έλεγχο πρόσβασης στα services που δημιουργεί μια εφαρμογή, δηλώνει στο *AndroidManifest.xml* αρχείο, την δημιουργία του service καθώς και τα απαιτούμενα δικαιώματα που πρέπει να έχει μια άλλη εφαρμογή, προκειμένου να το χρησιμοποιήσει.

Τα σύνηθη δικαιώματα για τα sercices είναι **START-STOP-BIND**. Μια εφαρμογή που θα θέλει να χρησιμοποιήσει το public service μιας αλλης εφαρμογής, δημιουργεί ένα **intent** προκειμένου να αιτηθέι την χρήση της υπηρεσίας. Αν διαθέτει τα απαιτούμενα δικαιώματα-permissions (θα τα έχει αιτηθεί κατά την εγκατάσταση μέσω του AndroidManifest.xml αρχείου) τότε η εφαρμογή θα μπορέσει να εκκινήσει την υπηρεσία, διαφορετικά ένα security exception θα προειδοποιήσει τον χρήστη ότι η συγκεκριμένη ενέργεια απέτυχε λόγω ελλειπών δικαιωμάτων.

## *Ασφάλεια των Content Providers*

Όπως αναφέρθηκε και παραπάνω τα Content Providers είναι τα components που αντιστοιχούν στην βάση δεδομένων που δημιουργεί μια εφαρμογή για την αποθήκευση δεδομένων και καθορίζουν τον τρόπου που οι πληροφορίες αυτές θα είναι διαθέσιμες σε άλλες εφαρμογές. Τα content providers ορίζουν δύο είδη δικαιωμάτων : **READING – WRITING**. Τα δικαιώματα καθορίζονται επακριβώς στο AndroidManifest.xml αρχείο της εφαρμογής. Όταν μια εφαρμογή Α προσπαθήσει να συνδεθεί σε ένα content provider μιας άλλης εφαρμογής Β, το λειτουργικό σύστημα θα ελέγξει αν η εφαρμογή Α διαθέτει το δικαίωμα ανάγνωσης ή εγγραφής της εφαρμογής Β. Αν ναι, η σύνδεση υλοποιείται και η εφαρμογή Α μπορεί να προσπελάσει τα δεδομένα της βάσης δεδομένων της εφαρμογής Β. Αν όχι, η σύνδεση τερματίζεται και ένα μήνυμα λάθους εμφανίζεται στον χρήστη.

### URI permissions

Αν και το content provider security model δουλεύει ιδανικά καθορίζοντας επακριβώς ποιες εφαρμογές θα έχουν πρόσβαση στη βάση δεδομένων άλλων εφαρμογών, υπάρχουν περιπτώσεις όπου η συγκεκριμένη λειτουργικότητα εγκυμονεί κυνδίνους ασφαλείας. Υπάρχουν εφαρμογές οι οποίες επιθυμούν να έχουν πρόσβαση σε ορισμένα στοιχεία της βάσης δεδομένων και όχι σε όλη την βάση, πχ μια εφαρμογή που εμφανίζει τα συνημμένα αρχεία ενός email. Χρησιμοποιώντας το content provider security model, η εφαρμογή για να μπορέσει να προσπελάσει τα συνημμένα των email θα της αποδοθεί το permission READING με αποτέλεσμα να μπορεί να διαβάζει όλα τα δεδομένα της βάσης και όχι μονο τα συνημμένα που είναι το ζητούμενο.

Για τον λόγο αυτό το Android χρησιμοποιεί τα URI permissions. Θέτωντας σε ένα content provider component την μεταβλητή

*Android:grandUriPermissions = “true*” ,

αυτομάτως η προσπέλασή της βάσης δεδομένων είναι δυνατή με την χρήση του URI . Αν κάποια εφαρμογή θέλει να έχει πρόσβαση στο συγκεκριμένο URI, θα πρέπει να αιτηθεί την απόδοση του ανάλογου permission μέσα απο το AdnroidManifest.xml file.

### *Ασφάλεια των Broadcast Receivers*

Όπως είδαμε τα Broadcast Receivers είναι υπεύθυνα για την επικοινωνία των components. Όταν γίνεται δημοσίευση ενός Intent από μία εφαρμογή, τα BR την λαμβάνουν και αν επιθυμούν εκτελούν την επιθυμητή ενέργεια.

Όταν γίνεται Broadcast ένα Intent, όπως αναφέραμε, είναι δυνατόν να φιλτραριστεί ο αριθμός των BR που λαμβάνουν την ειδοποίηση και να καθοριστεί επακριβώς ποια BR και με ποια δικαιώματα μπορούν να ανταποκριθούν στην αίτηση.

Όμως θα πρέπει και τα Broadcast Receivers Components να ελέγχουν από ποιους μπορούν να δεχτούν δημοσιεύσεις. Αυτό επιτυγχάνεται με ανάλογο τρόπο μέσω του Manifest αρχείου των εφαρμογών, θέτωντας στο configuration των BR δικαιώματα πρόσβασης που πρέπει να έχουν κατοχυρώσει οι αποστολείς των broadcast intents. Με αυτό τον τρόπο μια εφαρμογή δεν λαμβάνει ανεξέλεγκτα μηνύματα από άλλες εφαρμογές, παρα μόνο από αυτές που έχουν τα απαραίτητα δικαιώματα.

### *Android Filesystem Isolation*

Κάθε εφαρμογή τρέχει στο δικό του Dalvik VM κάτω από το δικό του UID. Μόνο οι εφαρμογές που τρέχουν κάτω από το ίδιο UID μπορεί να έχουν πρόσβαση σε κοινά δεδομένα. Με την εγκατάσταση της εφαρμογής το λειτουργικό σύστημα δημιουργεί στο filesystem τον φάκελο **/data/data/app_package_name** όπου εκεί αποθηκεύονται όλα τα αρχεία της. Στο συγκεκριμένο φάκελο δημιουργείται ο υποφάκελος **/files**, και του ανατίθεται ως **Owner** το **UID** του χρήστη που έχει υπογράψει ψηφιακά την εφαρμογή, προκειμένου η εφαρμογή να έχει πλήρη δικαιώματα επι των δεδομένων που αποθηκεύονται σε αυτόν. Κανένα άλλο δικαίωμα δεν καθορίζεται στον συγκεκριμένο φάκελο επομένως η εφαρμογή απομονώνεται από τις υπόλοιπες που υπάρχουνστη συσκευή καθώς καμία άλλη δεν έχει δικαιώματα πρόσβασης στον φάκελο files.

Οι εξωτερικές μνήμες των κινητών συσκευών (SD cards) δεν υποστηρίζουν το μοντέλο δικαιωμάτων του Linux, επομένως τα δεδομένα που αποθηκεύουν οι εφαρμογές ή ο χρήστης σε αυτές είναι προσπελάσιμα από όλες τις εφαρμογές που υπάρχουν στην συσκευή.

Υπάρχουν τριά είδη αρχείων που δημιουργούν οι Android εφαρμογές.

**Common Files :** Όταν μια εφαρμογή δημιουργεί ένα αρχείο, το Android του αναθέτει τα προκαθορισμένα δικαιώματα πρόσβασης, πλήρη πρόσβαση για τον owner (UID) και τίποτα άλλο. Όμως λόγω σχεδιαστικών αναγκών ίσως είναι επιθυμητή η πρόσβαση στα δεδομένα των αρχείων μιας εφαρμογής από άλλες. Για να υλοποιηθεί η συγκεκριμένη λειτουργικότητα, το Linux διαθέτει πρόσθετες ετικέτες δικαιωμάτων που τροποιούν τα δικαιώματα πρόσβασης στα αρχεία των εφαρμογών όπως παρακάτω:

*MODE_PRIVATE* : Η προκαθορισμένη τιμή για πλήρη έλεγχο από τον UID της εφαρμογής.

*MODE_WORLD_WRITABLE* : Επιτρέπει σε όλες τις εφαρμογές της συσκευής να γράφουν δεδομένα στο αρχείο.

*MODE_WORLD_READABLE* : Επιτρέπει σε όλες τις εφαρμογές της συσκευής να διαβάζουν τα δεδομένα του αρχείου.

**Shared Preferences file:** είναι το βασικό αρχείο στο οποίο η εφαρμογή αποθηκεύει ζεύγη από ονόματα/τιμές για εύκολη και γρήγορη πρόσβαση. Αποθηκεύεται στον υποφάκελλο **/data/data/app_name/shared_prefs** και χρησιμοποιεί τα ίδια δικαιώματα πρόσβασης και ετικέτες με τα κοινά αρχεία. Προσπελαύνονται από τα Share Preference objects των εφαρμογών ή μπορούν να διαβαστούν σαν XML αρχεία του συστήματος.

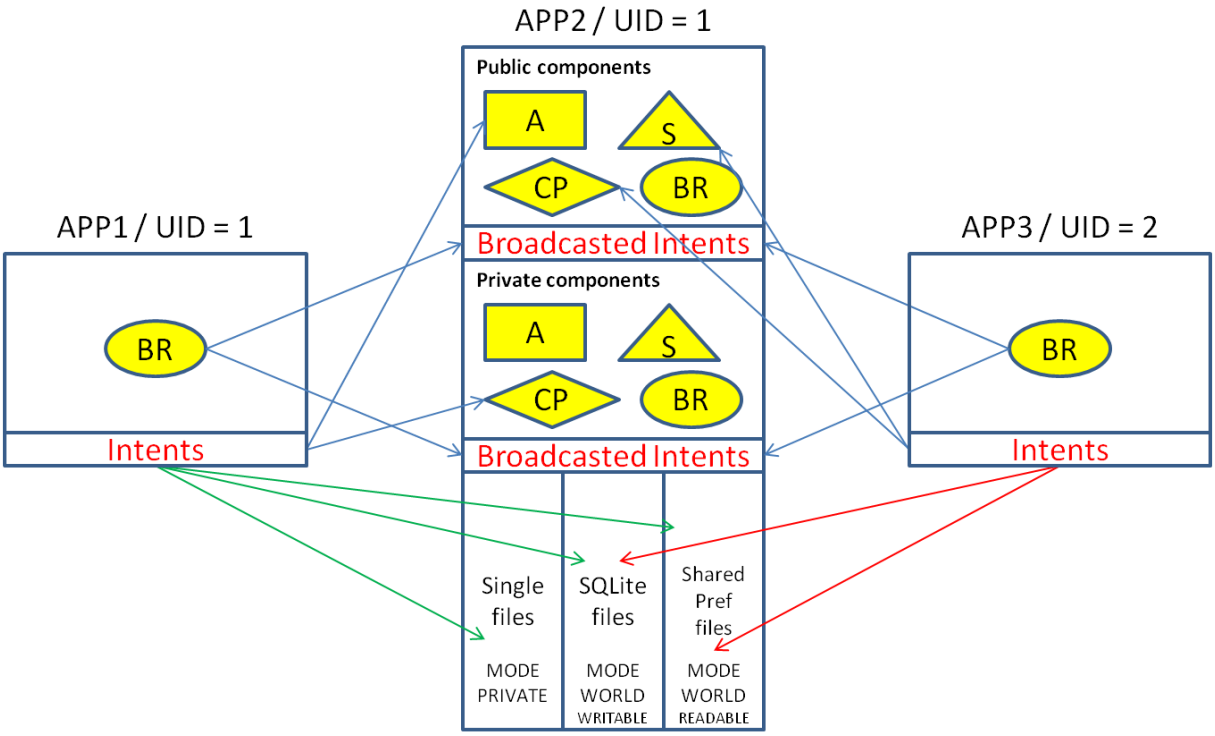
**SQLite file:** Αποτελεί την βάση δεδομένων που χρησιμοπόιούν οι εφαρμογές προκειμένου να αποθηκεύουν δομές δεδομένων πιο περίπλοκες από τα ονόματα/τιμές ζευγάρια των Shared Preferences αρχείων. Αποθηκεύονται στον υποφάκελλο **data/data/app_name/databases** και χρησιμοποιεί και αυτό τα ίδια δικαιώματα πρόσβασης και ετικέτες με τα κοινά αρχεία.

Συνεπώς η πρόσβαση σε όλα τα αρχεία που δημιουργεί μια εφαρμογή γίνεται μέσω του UID και των δικαιωμάτων που δίνει σε αυτά η ίδια η εφαρμογή καθώς όλα αποθηκεύονται κάτω από τον ίδιο φάκελο */files*. Η προσέγγιση αυτή διαφέρει σημαντικά από την ασφάλεια των δεδομένων στις iOS συσκευές όπου τα αρχεία των εφαρμογών βρίσκονται σε διαφορετικές τοποθεσίες μέσα στο filesystem της συσκευής.

## *Σύνοψη του Μοντέλου Ασφαλείας*

Το μοντέλο ασφαλείας του Android βασίζεται στην διαχείριση δικαιωμάτων πρόσβασης (permissions) μεταξύ των εφαρμογών και των components που τις αποτελούν. Για μικρές εφαρμογές όπου τα διαθέσιμα Activities , Services και Content

Providers είναι περιορισμένα, η διαχείριση των δικαιωμάτων πρόσβασης είναι απλή. Όμως όταν οι εφαρμογές αποτελούνται από πολλές δραστηριότητες και components που αλληλεπιδρούν μεταξύ τους, αλλά και με components άλλων εφαρμογών η παραπάνω διαδικασία διαχείρισης των δικαιωμάτων αποτελεί πολύπλοκη διαδικασία καθώς τυχόν παράβλεψη κατά τον σχεδιασμό μπορεί να οδηγήσει σε διαρροή ευαίσθητων δεδομένων του χρήστη.




**Εικόνα 22**

Στην παραπάνω εικόνα βλέπουμε συνοπτικά το μοντέλο ασφαλείας του Android λειτουργικού συστήματος και πως μπορεί μια εφαρμογή να προσπελάσει τα δεδομένα μια άλλης. Οι App1, App2 έχουν τον ίδιο ιδιοκτήτη (owner:UID=1) επομένως η App1 μπορεί να προσπελάσει όλα τα αρχεία που δημιουργεί η App2 ανεξαρτήτως επιπέδου πρόσβασης που έχει οριστεί. Αντιθέτως η App3 μπορεί να προσπελάσει μονο τα SQLite και Shared Pref αρχεία τα οποία η App1 έχει ορίσει ως MODE WORLD.

Επιπλέον η App1 εφαρμογή, μπορεί με intents να ζητήσει από την App2 λειτουργικότητες που παρέχουν τόσο τα public όσο και τα Private components, ενώ παράλληλα μπορεί να ανταποκριθεί και σε κάθε Broadcasted Intent.

Η App3 που τρέχει κάτω από διαφορετικό UID, έχει περιορισμένη πρόσβαση στα αρχεία της App2. Τα MODE PRIVATE αρχεία είναι μη προσπελάσιμα από την App3 καθώς επίσης δεν είναι δυνατή η αποστολή intents προς τα private components της

App2. Πλην όμως, αν διαθέτει τα κατάλληλα permissions μπορεί να αιτηθεί και να αποκτήσει δικαιώματα πρόσβασης στα public components της App2.

Ο συνδυασμός των δικαιωμάτων πρόσβασης και του Sandboxing εγγυάται την παροχή αυξημένου επιπέδου ασφαλείας για τα ευαισθητα δεδομένα των χρηστών αλλά απαιτείται προσεκτική διαχείριση τόσο από τους developers που αναπτύσουν τις εφαρμογές (για την απόδοση των κατάλληλων δικαιωμάτων) όσο και από τους χρήστες που έχουν το δικαίωμα επιλογής αν θα εγκαταστήσουν μια εφαρμογή στην συσκευή τους ή όχι (απορρίπτοντας ή εγκρίνοντας τα permissions που αιτείται κάθε εφαρμογή).

## *3.3 Τομείς επίθεσης*

Το λειτουργικό σύστημα Android κατέχει την 1$^{η}$ θέση παγκοσμίως στις κινητές συσκευές. Όμως, η εμπορικότητα και η δημοφιλία του, αύξησαν κατακόρυφα και τις επιθέσεις εναντίων των συσκευών και των εφαρμογών που τρέχουν σε αυτές. Στο Android Market υπάρχουν εφαρμογές οι οποίες είναι ψηφιακά υπογεγραμμένες από έμπιστες οντότητες. Πλήν όμως οι Android συσκευές επιτρέπουν την εγκατάσταση εφαρμογών που είναι ψηφιακά υπογεγραμμένες από τους ίδιους τους developer και αρκετές εξ αυτών βρίσκονται σε διαδικτυακές τοποθεσίες εκτός Android Market. Κανείς λοιπον, δεν μπορεί να εμποδίσει έναν χρήστη να εγκαταστήσει εφαρμογές που δεν προέρχονται από έμπιστες οντότητες και πιθανόν να περιέχουν malware κώδικα.

Οι επιτιθέμενοι στις Android συσκευές έχουν τους ίδιους σκοπούς με τις επιθέσεις στο iOS λειτουργικό σύστημα. Την υποκλοπή ευαίσθητων δεδομένων από τους χρήστες, την παραγωγή κέρδους για τους ίδιους και την κατάρρευση της συσκευής.

Μπορούμε να κατηγοριοποιήσουμε τις επιθέσεις κατά των Android συσκευών σύμφωνα με την κατηγοριοποιηση των iOS επιθέσεων:

- Επιθέσεις εναντίων των εφαρμογών που οδηγούν σε απόκτηση δικαιωμάτων πρόσβασης σε ευαίσθητα δεδομένα (privilege escalation attacks)
- Επιθέσεις εναντίων χαρακτηριστικών της συσκευής.

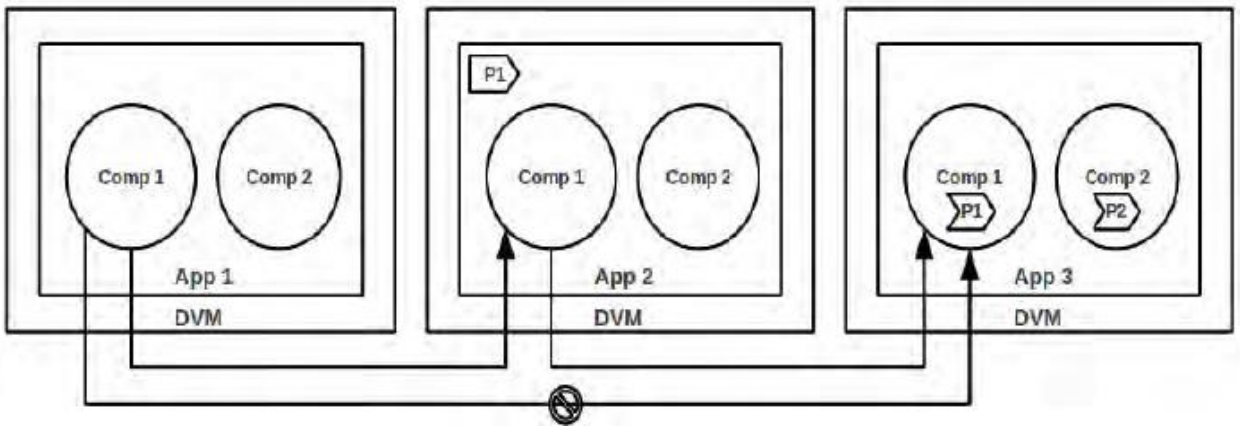
### *3.3.1 Επιθέσεις στις Εφαρμογές Android*

Το Android προσφέρει το μοντέλο ασφαλείας βασισμένο στο αντίστοιχο μοντέλο του Linux. Αν και ο μηχανισμός διαχείρισης των δικαιωμάτων πρόσβασης στις εφαρμογές σε συνδυασμό με το Sandboxing προσφέρουν ένα ικανοποιητικό τείχος

ασφαλείας, η πλειοψηφία των επιθέσεων εναντίων των Android συσκευών επικεντρώνεται στις αδυναμίες που παρουσιάζουν οι εφαρμογές που έχουν αναπτυχθεί για αυτές. Το Android Market παρέχει το μεγαλύτερο ποστοστό των διαθέσιμων εφαρμογών, οι οποίες είναι ελεγμένες ως προς την ασφάλειά τους. Όμως η ύπαρξη διαδικτυακών τόπων που προσφέρουν Android εφαρμογές (με το κίνητρο της δωρεάν εγκατάστασης δημοφιλών εμπορικών εφαρμογών) σε συνδυασμό με την δυνατότητα εγκατάστασης εφαρμογών με ψηφιακή υπογραφή χωρίς πιστοποίηση από τις CAs, έχει αυξήσει σημαντικά τις πιθανότητες εγκατάστασης malware λογισμικού στις Android συσκευές.

#### *3.3.1.1 Priviledge Escallation Επίθεση*

Όπως είδαμε παραπάνω το μοντέλο ασφαλείας του Android βασιζεται στην διαχείριση δικαιωμάτων πρόσβασης μεταξύ των εφαρμογών, δομημένο πάνω στο αντίστοιχο μοντέλο του Linux. Με τις priviledge escalation attacks οι επιτιθέμενοι, αφού εγκαταστήσουν μια φαινομενικά ασφαλή εφαρμογή στην κινητή συσκευή του χρήστη (από τοποθεσία εκτός Android Market συνηθως), προσπαθούν να εκμεταλλευτούν τρωτότητες και αδυναμίες στον σχεδιασμό των ήδη εγκατεστημένων εφαρμογών προκειμένου να αποκτήσουν δικαιώματα πρόσβασης στα ευαίσθητα δεδομένα του χρήστη.




**Εικόνα 23 [19]**

Στην παραπάνω εικόνα παρατηρούμε την απλούστερη μορφή μιας Priviledge Escalation επίθεσης και θα αναλύσουμε πως αυτή λειτουργεί.

Η Εφαρμογή App1 αποτελείται από 2 components και δεν έχει κατοχυρώσει κανένα δικαίωμα πρόσβασης. Η App2, πάλι με 2 components έχει κατοχυρώσει το permission P1 . Τέλος η εφαρμογή App3 δεν έχει κατοχυρώσει κανένα δικαίωμα αλλά

έχει θέσει ως περιορισμό το permission P1 για όποιο component θελήσει να προσπελάσει τα component comp1 και comp2. Το App1/Comp1 μπορεί μέσω ενός Intent1 μηνύματος να αιτηθεί την πρόσβαση στο App2/Comp1. Η App2 δεν έχει θέσει κάποιο Intent-filter και επομένως αποδέχεται το αίτημα της App1 καθώς η πρόσβαση στο App2/Comp1 δεν απαιτεί κανένα permission από τον καλούντα. Το App2/Comp1 component με την σειρά του αιτείται μέσω του Intent2 την πρόσβαση στο App3/Comp1 το οποίο προστατεύεται από το permission P1. Η αίτηση γίνεται αποδεκτή καθώς η App2 έχει κατοχυρώσει το P1 δικαίωμα κατά την εγκατάστασή της. Αν τώρα το Intent2 αίτημα αφορούσε στην ανάγνωση δεδομένων από ένα content provider component, η App1 αν και δεν έχει δικαίωμα πρόσβασης στο App3/Comp1 μέσω της App2 μπορεί να διαβάσει δεδομένα από την App3.

#### *3.3.1.2 Επιθέσεις κατά των Intent*

Οι περισσότερες Intent Based Attacks βασίζονται πάνω στην επικοινωνία των components και στα προγραμματιστικά λάθη κατά τον σχεδιασμό, που επιτρέπουν την απόκτηση δικαιωμάτων πρόσβασης από εφαρμογές που δεν είναι εξουσιοδοτημένες.

Οι συγκεκριμένες επιθέσεις προσπαθούν να εκμεταλλευτούν αδυναμίες στις υλοποιήσεις των intents τα οποία είναι υπεύθυνα για την επικοινωνία των components και μπορούν να εκτελέσουν ένα μεγάλο αριθμό από διαφορετικών ειδών επιθέσεις.

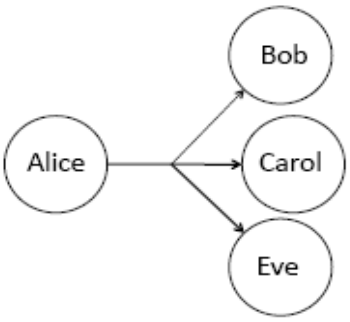
##### *Μη εξουσιοδοτημένη πρόσβαση σε Intent*

Όταν μια εφαρμογή στείλει ένα Implicit Intent, δεν υπάρχει καμία εγγύηση ότι το συγκεκριμένο intent θα ληφθεί από το κατάλληλο component. Μια εφαρμογή μπορεί να παρεμβληθεί στην επικοινωνία, θέτωντας ένα Intent-filter με όλες τις κατηγορίες, δεδομένα και ενέργειες που μπορεί βρίσκονται στο intent. Τότε η επιτιθέμενη εφαρμογή αποκτά πρόσβαση σε όλα τα στοιχεία που είναι όμοια με τα στοιχεία του Intent εκτός από εκείνα όπου έχουν τεθεί περιορισμοί πρόσβασης και ο επιτιθέμενος δεν έχει τα κατάλληλα permissions. Σε αυτή την κατηγορία ανήκουν οι παρακάτω στοχευμένες επιθέσεις.

**A. Broadcast Theft**

Τα Broadcast Intents είναι ευάλωτα στην παθητική υποκλοπή και στις denial of service επιθέσεις. Η επιτιθέμενη εφαρμογή μπορεί να

παρατηρεί τα περιεχόμενα των broadcast intents χωρίς να παρεμβαίνει στην επικοινωνία, θέτωντας ένα Intent-filter με όλες τις πιθανές ενέργειες, δεδομένα και κατηγορίες που μπορεί να χρησιμοποιήσουν οι υπόλοιπες εφαρμογές.



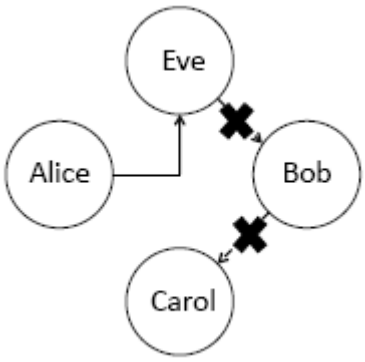

**Εικόνα 24 [18]**

Στην Εικόνα 21 βλέπουμε την υποκλοπή των broadcast intent αφού πέρα από τον Bob και την Carol που είναι οι αναμενόμενοι παραλήπτες, το λαμβάνει και η Eve.

### B. Denial of service for ordered broadcasts.

Τα ordered broadcasts είναι intents τα οποία ακολουθούν σειριακή προσπέλαση των αποδεκτών. Ο 1ος αποδέκτης λαμβάνει το broadcast και μολις τελειώσει την ενέργεια του εκπέμπει με την σειρά του νέο broadcast προκειμένου το επόμενο component να αρχίσει την λειτουργία του. Το τελευταίο component που συμμετέχει στην διαδικασία στέλνει το αποτέλεσμα στην αρχική διεργασία.

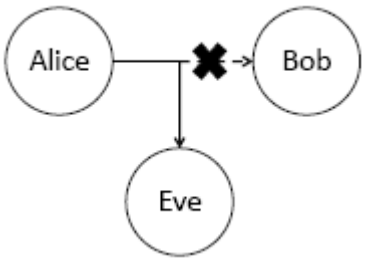
Στο προηγούμενο παράδειγμα η Eve σαν ενεργός επιτιθέμενος μπορεί να εκκινήσει μια Denial Of Service επίθεση. Η Eve παρατηρεί τα Broadcast Intents της Alice και καταναλώνει το service χωρίς να προωθήσει το Intent στον επόμενο αποδέκτη (που εδώ είναι ο Bob).




**Εικόνα 25 [18]**

### Γ. Activity Hijacking

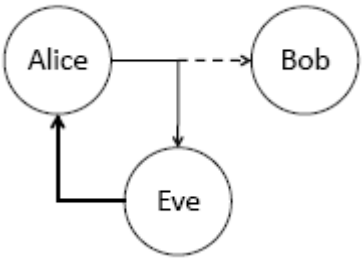
Σε αυτήν την επίθεση το επιτιθέμενο Activity εκκινεί στην θέση του κανονικού που αναμένει το broadcast intent.




**Εικόνα 26 [18]**

Στην πιο απλή περίπτωση ο επιτιθέμενος θα διαβάσει τα δεδομένα της δραστηριότητας και θα την παραδώσει στην δραστηριότητα που αναμένει.

Σε μια πιο ευφυή επίθεση η Eve διαβάζει τα δεδομένα, εκτελεί την ζητούμενη ενέργεια και επιστρέφει στην Alice λανθασμένο αποτέλεσμα. (εικόνα 24)




**Εικόνα 27 [18]**

### Δ. Service Hijacking

Παρόμοια με το Activity Hijacking μια εφαρμογή μπορεί να εκτελέσει ένα service από μια malware εφαρμογή και όχι το service που επιθυμεί. Η επίθεση γινεται στο παρασκήνιο του λειτουργικού συστήματος και ο χρήστης δεν μπορεί να την αντιληφθεί καθώς δεν υπάρχει interface με τον χρήστη για την εκκίνηση των υπηρεσιών. Ο επιτιθέμενος είναι σε θέση να στείλει malware κώδικα στην εξαπατημένη εφαρμογή, λάθος δεδομένα ή απλά μπορεί να επιστρέψει το σωστό αποτέλεσμα χωρίς να εκτελέσει καμία ενέργεια.

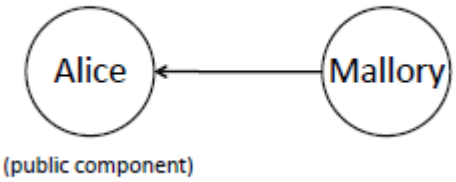
### Ε. Intents URI

Όπως αναφέρθηκε σε προηγούμενη παράγραφο τα Intents μπορούν να περιέχουν URI για την πρόσβασή τους σε συγκεκριμένα πεδία της βάσης δεδομένων, μέσω των Content Providers. Υπάρχει περίπτωση μια εφαρμογή Α να στείλει ένα intent στην εφαρμογή Β προκειμένου η εφαρμογή Β να εκτελέσει κάποια

ενέργεια στα δεδομένα της Α μέσω URI πρόσβασης. Αν η εφαρμογή Β δεν είχε εξ αρχής κατοχυρωμένα δικαιώματα για την URI πρόσβαση, η εφαρμογή Α μπορεί να στείλει μέσω των Intent flags που κατοχυρώνουν το URI permission που απαιτείται. Ο επιτιθέμενος έχει την δυνατότητα όπως είδαμε να υποκλέψει το Intent, να κατοχυρώσει τα απαιτούμενα URI δικαιώματα και να αποκτήσει πρόσβαση στα δεδομένα.

## *Intent Spoofing*

Στην intent spoofing attack, ο επιτιθέμενος δεν ενεργεί παθητικά. Για την υλοποίηση της επίθεσης στέλνεται ένα Intent σε ένα exported/public component, το οποίο όμως δεν περιμένει intents από την malicious εφαρμογή.Αν η εφαρμογή που δέχεται το intent είναι σχεδιασμένη να εκτελεί κάποια ενέργεια με την παραλαβή ενός συγκεκριμένου intent, τότε η ενέργεια αυτή θα ξεκινήσει από μη εξου σιοδοτημένη εφαρμογή και όχι από την εφαρμογή που αναμενόταν (εικόνα 28).




**Εικόνα 28 [18]**

Οι κυριότερες επιθέσεις που βασίζονται σε αυτή την τεχνική είναι:

### A. Malicious Broadcast Injection

Οι Receivers συνήθως χρησιμοποιούνται για να στέλνουν δεδομένα ή να εκκινούν μια διαδικασία όταν λάβουν ένα συγκεκριμένο broadcast intent.

Αν ένα Broadcast Receiver δέχεται broadcasted intents χωρίς περιορισμούς πρόσβασης (από όλους τους πιθανούς αποστολείς), τότε ένα malicious intent μπορεί να εκκινήσει λανθασμένα μια διαδικασία ή να στείλει μη έγκυρα δεδομένα στην εφαρμογή.

### B. Malicious Activity Launch

Τα Public Activities που έχουν γίνει exported από την εφαρμογή μπορούν να ενεργοποιηθούν με τα Intents. Μία malicious εφαρμογή μπορεί να στείλει Intents σε Public Activities προκειμένου να τα ενεργοποιήσει χωρίς την

έγκριση του χρήστη. Αμεσο αποτέλεσμα αυτού, είναι η εφαρμογή που δέχεται το malicious intent να ενεργοποιήσει το Activity και να :

(1) Τροποποιήσει τα δεδομένα ή την κατάσταση της εφαρμογής με μη έγκυρες τιμές.

(2) Να στείλει ευαίσθητα δεδομένα στην malicious εφαρμογή ως αποτέλεσμα του Activity.

### Γ. Malicious Service Launch

Αν ένα δημόσιο service δεν είναι προστατευμένο με ισχυρά permissions, τότε οποιαδήποτε εφαρμογή μπορεί να εκκινήσει ή να δεσμεύσει την υπηρεσία με κύνδινο την διαρροή πληροφοριών ή την εκτέλεση ενεργειών για τις οποίες οι επιτιθέμενες εφαρμογές δεν έχουν δικαίωμα πρόσβασης. Η Malicious Service Launch επίθεση είναι παρόμοια με την Malicious Activity αλλά οι επιπτώσεις που επιφέρει η εκκίνηση του service είναι πολύ μεγαλύτερη από αυτή του Activity. Τα services συνήθως δέχονται σαν είσοδο με τα Intent που τα ενεργοποιούν, δεδομένα από τις εφαρμογές προκειμένου να εκτελέσουν μια ή έναν αριθμό από ενέργειες, οι οποίες μερικές φορές είναι κλήσεις συστήματος. Από την στιγμή που είναι εφικτό για μια malicious εφαρμογή να στείλει τα δικά της δεδομένα για την εκκίνηση της υπηρεσίας τότε θα εκτελεστούν κλήσεις του συστήματος με malicious κώδικα, ότι ακριβώς επιθυμεί ο επιτιθέμενος.

### Δ. Malicious Content Provider Communication

Η σοβαρότερη και πιο επικύνδινη επίθεση είναι αυτή προς τα Content Providers Components. Όπως αναφέραμε τα Content Providers αποτελούν την διεπαφή της εφαρμογής με την βάση δεδομένων που τηρούν. Εκεί οι εφαρμογές αποθηκευουν ευαίσθητα προσωπικά δεδομένα των χρηστών και επομένως τυχόν παραβίαση των δικαιωμάτων σε αυτά μπορεί να οδηγήσει σε σημαντική διαρροή πληροφοριών. Αν ένα content provider είναι δημόσια ορατό και δεν έχει ορίσει δικαιώματα πρόσβασης σε αυτό (READ-WRITE) τότε είναι πιθανό μια malicious εφαρμογή να στείλει ένα Intent σε αυτό και να αιτηθεί την προσπέλαση των δεδομένων του content provider. Από την στιγμή που δεν γίνεται έλεγχος για το ποιος στέλνει τα Intents η διεργασία “θύμα” θα αποδεχτεί το μήνυμα και θα επιτρέψει την διαρροή των πληροφοριών.

#### *3.3.1.3 Λοιπές επιθέσεις*

Οι κινητές συσκευές Android είναι ευάλωτες και στις γνωστές επιθέσεις SQLinjection, XSS, auth κλπ οι οποίες όμως δεν εκμεταλλέυονται χαρακτηριστικά του λειτουργικού συστήματος (οι επιθέσεις γίνονται ακριβώς με τον ίδιο τρόπο με τα κλασσικά μηχανήματα desktop) και γι αυτό δεν αποτελούν αντικείμενο της παρούσας μελέτης.

#### *Επιθέσεις στα Log Files*

Το λειτουργικό σύστημα Android προσφέρει κεντρική υπηρεσία διαχείρισης και αποθήκευσης των log files μέσω του αντίστοιχου API. Αρκετές εφαρμογές όπως και στο iOS αποθηκεύουν ευαίσθητα δεδομένα στα log files, όπως passwords αναγνωριστικά δικτύου, τοποθεσία χρήστη μέσω GPS κλπ.

Όλες οι εφαρμογές μπορούν να ζητήσουν το READ_LOGS system permission του Android και να έχουν πρόσβαση στο LOG file αποσπώντας έτσι τα ευαίσθητα δεδομένα των άλλων εφαρμογών .

#### *Μη ασφαλή file system permissions*

Όπως είδαμε οι εφαρμογές αποθηκεύουν τα αρχεία τους κάτω από τον κατάλογο data/data/app_name. Σύμφωνα με το permission system του Adnroid μονο ο κάτοχος της εφαρμογής έχει πρόσβαση σε αυτόν τον κατάλογο. Όμως αρκετές εφαρμογές προχωρούν σε τροποποίηση των δικαιωμάτων των αρχείων που αποθηκεύουν στην μνήμη της συσκευής, θέτωντάς τα MODE_WORLD_READABLE/ WRITABLE. Με αυτή την ενέργεια τα αποθηκευμένα αρχεία μπορούν να προσπελαστούν από όποιαδήποτε εφαρμογή μέσα στη συσκευή.

#### *Επιθέσεις στο Web View των εφαρμογών*

Η WebView class είναι μια υποκλάση της View κλάσης και χρησιμοποιείται για την προβολή ιστοσελίδων. Οι εφαρμογές χρησιμοποιούν την WebView class για να προβάλουν ιστοσελίδες μέσα σε αυτές, έχωντας και την δυνατότητα να αλληλεπιδράσουν με τους web servers. Μέσα από αυτήν την δυνατότητα των εφαρμογών οι επιτιθέμενοι μπορούν να εκκινήσουν 2 ξεχωριστά είδη επιθέσεων

**Attacks from malicious web pages :** Υπάρχουν εφαρμογές ο οποίες προβάλουν εμφωλευμένες δημοφιλείς ιστοσελίδες, παρέχωντας

επιπλέον λειτουργικότητες για τον συγκεκριμένο ιστότοπο. πχ: εφαρμογή που προσφέρει λειτουργικότητα του facebook για την εύρεση φίλων που βρίσκονται κοντά στον χρήστη (μέσω GPS συντεταγμένων των κινητών συσκευών). Ο επιτιθέμενος μπορεί να ξεγελάσει την χρήστη, να φορτώσει μια μολυσμένη ιστοσελίδα (μέσω ενός παραπλανητικού email) στην θέση του facebook και μετά να εξαπολύσει επίθεση κατά του WebView[22] της εφαρμογής που φιλοξενεί την ιστοσελίδα του facebook, με σκοπό την υποκλοπή ευαίσθητων πληροφοριών.

**Attacks from malicious apps :** Από την στιγμή που η πλατφόρμα Android επιτρέπει την εγκατάσταση εφαρμογών από ιστότοπους εκτός Android Market, η παρουσία malicious εφαρμογών στις κινητές συσκευές Android είναι συχνό φαινόμενο. Μερικές μολυσμένες εφαρμογές είναι στοχευμένες για να επιτίθενται σε συγκεκριμένες ιστοσελίδες (πχ facebook) μέσω της WebView κλάσης. Ο χρήστης μολις εγκαταστήσει την malicious εφαρμογή, μέσω του WebView θα επισκεφθεί τον ιστότοπο “στόχο” . Εκεί ο επτιθέμενος μέ Javascript Injection τεχνική και Event Sniffing μπορεί να υποκλέψει τα προσωπικά δεδομένα του χρήστη για την εφαρμογή «στόχο» [22]

### *Επίθεση Man In the Middle*

Όπως είδαμε παραπάνω, το λειτουργικό σύστημα υποστηρίζει την δημιουργία κρυπτογραφημένων – ασφαλών συνδέσεων μέσω του πρωτοκόλλου SSL. Υπάρχουν όμως εφαρμογές οι οποίες δεν χρησιμοποιούν το συγκεκριμένο πρωτόκολλο και δέχονται συνδέσεις από μη πιστοποιημένους ιστοτόπους κάνωντάς τες παράλληλα ευάλωτες σε Man In The Middle επιθέσεις.

## *3.3.2 Επιθέσεις κατά των χαρακτηριστικών ασφαλείας των συσκευών Android*

Οι συγκεκριμένες επιθέσεις στοχέυουν σε τρωτότητες και χαρακτηριστικά του λειτουργικού συστήματος προκειμένου ο επιτιθέμενος να αποκτήσει πρόσβαση σε κρίσιμες πληροφορίες του χρήστη.

### *3.3.2.1 Μη ασφαλής αποθήκευση δεδομένων*

Οι εφαρμογές προκειμένου να αποκτήσουν ένα επίπεδο λειτουργικότητας ελκυστικό προς τους χρήστες, αποθηκεύουν ευαίσθητα δεδομένα στην

μνήμη των συσκευών. Το Android δεν παρέχει προκαθορισμένη κρυπτογράφηση των δεδομένων όταν αυτά αποθηκεύονται στην μνήμη. Η ασφάλεια πηγάζει από το permission system του Android όπου πρόσβαση σε ένα αρχείο έχει μονο ο ιδιοκτήτης αυτού (η εφαρμογή που το δημιουργησε). Αν και το λειτουργικό σύστημα παρέχει API για την κρυπτογράφηση των δεδομένων, οι developers είναι αυτοί που αποφασίζουν αν οι εφαρμογές θα κρυπτογραφούν τα δεδομένα πριν τα αποθηκεύσουν στην συσκευή. Αριθμός εφαρμογών δεν χρησιμοποιούν κρυπτογραφία κατά την αποθήκευση των δεδομένων στη συσκευή καθώς θεωρούν ότι το permission system σε συνδυασμό με το sandboxing είναι αρκετό για την προστασία των δεδομένων.Πλην όμως τυχόν απώλεια ή κλοπή της συσκευής θα έχει ως αντικτυπο την εύκολη πρόσβαση στα δεδομένα των χρηστών από τον επιτιθέμενο.

Επίσης αρκετές εφαρμογές επιλέγουν να αποθηκεύουν δεδομένα στις SD κάρτες μνήμης των συσκευών. Οι εξωτερικές κάρτες μνήμης όμως δεν υποστηρίζουν το permission system του Android με αποτέλεσμα τα δεδομένα που αποθηκεύονται εκεί να είναι χωρίς περιορισμούς ασφαλείας. Μια κακόβουλη εφαρμογή η οποία είναι εγκατεστημένη μπορεί να ανιχνέυσει την εξωτερική μνήμη και να υποκλέψει ευαίσθητα προσωπικά δεδομένα.

#### *3.3.2.2 Rooted Device*

Μια Rooted Device αντιστοιχει στην Jailbreaked συσκευή του iOS. Ο χρήστης απενεργοποιεί το permission system του Android και τρέχει όλες τις εφαρμογές κάτω από τον ίδιο χρήστη:device ο οποίος έχει root πρόσβαση σε όλα τα αρχεία και υπηρεσίες της συσκευής. Αυτή η ενέργεια εγκυμονεί σοβαρότατους κυνδίνους ασφαλείας καθώς οποιοδήποτε malware εγκατασταθεί στη συσκευή, θα έχει πλήρη πρόσβαση σε όλα τα δεδομένα όλων των εγκατεστημένων εφαρμογών.

#### *3.3.2.3 Αυθεντικοποίηση εφαρμογών*

Πρόσφατα δόθηκε στην δημοσιότητα σημαντική τρωτότητα του λειτουργικού συστήματος η οποία επέτρεπε στον επιτιθέμενο να τροποποιήσει τον κώδικα μιας εφαρμογής χωρίς να αλλάξει την ψηφιακή υπογραφή του developer. Η τρωτότητα εντοπίστηκε στον τρόπο υλοποίησης των ψηφιακών υπογραφών και πιστοποιητικών με αποτέλεσμα όλες σχεδόν οι συσκευές Android να είναι ευάλωτες σε αυτή [38]. Με την συγκεκριμένη τρωτότητα οι επιτιθέμενοι είναι σε θέση να τροποποιούν τον κώδικα γνωστών εφαρμογών, τοποθετώντας mallware εντολές στο εσωτερικό τους,

χωρίς να αλλάξει η ψηφιακή υπογραφή. Η συσκευή “στόχος” μπορεί να δεχθεί μνμα αναβάθμισης μιας εφαρμογής από τον επιτιθέμενο, και από την στιγμή που η ψηφιακή υπογραφή παραμένει η ίδια, κανένα μήνυμα ασφαλείας δεν θα εμφανιστεί στον χρήστη προτού αυτός εγκαταστήσει την “αναβαθμισμένη” έκδοσή της. Παρακάμπτωντας το σημαντικότερο κομμάτι της ασφάλεια των κινητών συσκευών, που είναι ο ίδιος ο χρήστης, ο επιτιθέμενος κατορθώνει τελικά να εκτελεί malware κώδικα κάτω από την ψηφιακή υπογραφή γνωστών εφαρμογών.

## *3.4 Εκτίμηση Ασφαλείας*

Το λειτουργικο σύστημα Android προσφέρει ένα πολύ δυνατο μοντέλο ασφαλείας , βασισμένο στο permission system του Linux. Αν και είναι ικανοποιητικό στην απόδοσή του, η διαχείριση των δικαιωμάτων πρόσβασης μέσα στις ίδιες τις εφαρμογές αποτελεί πολύπλοκη διεργασία, με αποτέλεσμα να υφίστανται υλοποιήσεις με σημαντικά κενά ασφαλείας οδηγώντας σε πολλές περιπτώσεις σε διαρροή πληροφοριών. Οι developers Android εφαρμογών θα πρέπει να είναι προσεκτικοί κατά την ανάπτυξη του λογισμικού και να αναθέτουν τα κατάλληλα δικαιώματα πρόσβασης σε όλα τα public components. Εάν η αποθήκευση ευαίσθητων δεδομένων είναι απαραίτητη για την ομαλή λειτουργία της εφαρμογής μεγάλη προσοχή πρέπει να δωθεί στο που και πως αυτά τα δεδομένα θα αποθηκευτούν.

Το βασικότερο όμως στοιχείο της ασφάλειας των κινητών συσκευών παραμένει ο ίδιος ο χρήστης. Το Androiod παρέχει την λειτουργικότητα, ο χρήστης να γνωρίζει εκ των προτέρω τι δικαιώματα πρόσβασης αιτείται κάθε εφαρμογή, μέσω του AndroidManifest.xml αρχείου. Η εγκατάσταση εφαρμογών από τοποθεσίες εκτός Android market εγκυμονεί σημαντικούς κυνδίνους ασφαλείας δεδομένων αλλά το λειτουργικό σύστημα δεν απαγορεύει μια τέτοια ενέργεια. Επομένως είναι στην κρίση του χρήστη ποιες και με ποια δικαιώματα εφαρμογές θα εγκατασταθούν στην συσκευή του.

## *3.5 Συμβουλές ανάπτυξης ασφαλών εφαρμογών Android*

Η ασφάλεια των εφαρμογών αποτελεί την κορωνίδα της ασφάλειας των κινητών συσκευών καθώς εκεί επικεντρώνονται οι επιτιθέμενοι, προσπαθώντας να εκμεταλλευτούν τρωτότητες του λογισμικού από λάθη κατά την σχεδίασή τους.

Ο παρακάτω πίνακας παραθέτει ορισμένες τεχνικές για ασφαλή ανάπτυξη Android εφαρμογών ο οποίος είναι επικεντρωμένος στις επιθέσεις που παρουσιάστηκαν στην παράγραφο 3.3.

| | |
|---|---|
| Activities | Καθορίστε επακριβώς ποιος έχει το δικαίωμα έναρξης των δραστηριοτήτων. |
| | Για την αρχική σελίδα της εφαρμογής δεν χρειάζονται permissions. |
| | Για τις υπόλοιπες δραστηριότητες ορίστε δικαιώματα πρόσβασης που πρέπει οι άλλες εφαρμογές να διαθέτουν. |
| Servicies | Καθορίστε επακριβώς ποιος εχει τα δικαίωματα START/STOP/BIND των υπηρεσιών που προσφέρει η εφαρμογή |
| | Εάν τα services εκτελούν ενεργειες σε ευαίσθητα δεδομένα του χρήστη, μην τα δημοσιεύετε προς όλες τις εφαρμογές. |
| | Τοποθετείστε δικαιώματα πρόσβασης σε όλα τα services των εφαρμογών |
| Content Providers | Καθορίστε επακριβώς ποιος έχει τα διακιώματα READ/WRITE επί της βάσης δεδομένων της εφαρμογής |
| | Σε περιπτωση που τμήμα των δεδομένων πρέπει είναι προσβάσιμο από τρίτες εφαρμογές χρησιμοποιήστε τα URI permissions για περιορισμένη πρόσβαση στα δεδομένα. |
| | Μην επιτρέπετε αυξημένα δικαιώματα πρόσβασης για εφαρμογές/components που δεν τα έχουν ανάγκη. |
| Broadcasts | Δημιουργείστε Broadcast Receivers που λαμβάνουν intents από συγκεκριμένους αποστολείς με τα απαραίτητα δικαιώματα. |
| Intents | Δημιουργείστε intents προς συγκεκριμένα components και θέστε permissions που πρέπει να έχει ο αποδέκτης |
| | Σε περίπτωση που χρειαστεί να δημοσιεύσετε (broadcast) ένα intent, θέστε περιορισμούς για το ποια Broadcast Receivers μπορούν να το λάβουν και με ποια permissions. |
| | Θέστε περιορισμούς στα components ως προς ποια intent δύναται |

| | |
|---|---|
| | να εξυπηρετήσουν (προστασία από priviledge escalation attacks). |
| Log Files | Μην αποθηκεύετε ευαίσθητα δεδομένα στο log file . Όλες οι εφαρμογές έχουν προσβαση σε αυτό. |
| SD Ram | Μην αποθηκεύεται ευαίσθητα δεδομένα στην SD μνήμη. Αν χρειαστεί να αποθηκεύσετε δεδομένα χρησιμοποιείστε το κατάλληλο μοντέλο κρυπτογράφησης για τα δεδομένα σας. |
| Data Storage | Μην αποθηκεύεται ευαίσθητα δεδομένα σε διάσπαρτα σημεία στη μνήμη. Αν χρειάζονται για την λειτουργικότητα της εφαρμογής αποθηκεύστε τα δεδομένα στην SQLite βάση δεδομένων. |
| | Μην τροποποιείτε τα δικαιώματα πρόσβασης των αρχείων αν δεν είναι απαραίτητο για την εφαρμογή. |
| Cryptography | Μην αποθηκεύετε τα δεδομένα χωρίς την χρήση κάποιας κρυπτογραφικής μεθόδου. |
| ManInTheMiddle | Χρησιμοποιείστε το SSL πρωτόκολλο για όλες τις επικοινωνίες της εφαρμογής. Μην επιτρέπετε συνδέσεις χωρίς αυθεντικοποίηση. |

Πίνακας 7

# ΚΕΦΑΛΑΙΟ 4

# ΕΛΕΓΧΟΣ ΤΡΩΤΟΤΗΤΩΝ ANDROID ΕΦΑΡΜΟΓΩΝ

## *4.1 Έλεχος Τρωτοτήτων Android εφαρμογών*

Η ραγδαία εξάπλωση των mallware εφαρμογών για κινητές συσκευές ανέδειξε την ανάγκη για την διενέργεια ελέγχων ασφαλείας στα smartphones και tablets ανάλογους με τους ελέγχους που εκτελούνται σε δίκτυα και υπηρεσίες με παραδοσιακούς ηλεκτρονικούς υπολογιστές.

### *Penetration Tests Features*

Τα penetration test αποτελούν ελέγχους ασφαλείας εφαρμογών για να διαπιστωθούν τυχόν κενά ασφαλείας που έχουν διαφύγει κατά την σχεδίαση – ανάπτυξη των εφαρμογών. Εκτελείται συνήθως λίγο πριν οι εφαρμογές βγούν στην αγορά και σκοπός τους είναι η ανακάλυψη τρωτοτήτων και διόρθωση αυτών.

Μία Pen Test ανάλυση μπορεί να διαιρεθεί σε 4 επιμέρους κατηγορίες:

*Planning*: Καθορίζονται οι σκοποί του Test

*Discovery*: Διαπιστώνονται οι διαθέσιμοι πόροι και χαρακτηριστικά του συστήματος (IP addresses, system information, databases κλπ)

*Attacks*: Με βάση την ανάλυση των χαρακτηριστικών του 2$^{ου}$ σταδίου, αναγνωρίζονται τα χαρακτηριστικά του συστήματος και οι εφαρμογές οι οποίες είναι ευαλωτες.

*Reporting*: Με το τέλος του pen test εκδίδεται μια αναφορά με τους πιθανούς κυνδίνους ασφαλείας του συστήματος που παρατηρήθηκαν κατά την ανάλυση η οποία παραδίδεται στους developer για χρήση-αξιολόγηση και διόρθωση εάν αυτό απαιτείται.

## *4.3 Android Pen Testing*

Οι εφαρμογές αποτελούν τον κύριο στόχο των επιτιθέμενων στις Android συσκευές. Η δυνατότητα εγκατάστασης εφαρμογών από ιστοτόπους εκτός Android Market και ψηφιακά υπογεγραμμένες από τους ίδιους τους developer ανεβάζει σημαντικά την πιθανότητα εμφάνισης mallware λογισμικού σε αυτές. Οι mallware εφαρμογές κατά κύριο λόγο προσπαθούν να εκμεταλλευτούν αδυναμίες στην σχεδίαση εμπορικών λογισμικών, προκειμένου να αποκτήσουν πρόσβαση σε ευαίσθητα δεδομένα. Τα περισσότερα Pen Test εργαλεία για Android είναι σχεδιασμένα να

εκτελούν ελέγχους κυρίως στις εγκατεστημένες εφαρμογές προκειμένου να ανακαλύψουν τυχόν κενά ασφαλείας που οι επιτιθέμενοι μπορεί να εκμεταλλευτούν προς όφελός τους.

Κατά την εκτέλεση ενός Pen Test στις Android εφαρμογές δύο είναι τα κύρια στοιχεία που επικεντρώνεται ο έλεγχος ασφαλείας.

Attack Surface : κάθε pen test επικεντρώνεται στις λειτουργικότητες που παρέχει μια εφαρμογή. Οι λειτουργικότητες αυτές καθορίζουν ποια components είναι κρίσιμα, και χειρίζονται ευαίσθητα δεδομένα προκειμένου να ελεγχθούν σε θέματα ασφαλείας.

Επικοινωνία με άλλα Comopnents : Μία εφαρμογή δύναται να επικοινωνεί με components άλλων εφαρμογών, χρησιμοποιώντας τους μηχανισμούς επικοινωνίας του Android (interprocess communication) .

Ο έλεγχος ασφαλείας των Android εφαρμογών χωρίζεται σε δύο βασικά σκέλη:

1. Android specific Security
2. General Application Security

### *4.3.1 Android specific Security*

Οι εφαρμογές Android χρησιμοποιούν χαρακτηριστικές λειτουργικότητες του λειτουργικού συστήματος οι οποίες πρέπει να ελεγχθούν ως προς την ασφάλεια που προσφέρουν στα δεδομένα του χρήστη. Τα κύρια σημεία ελέγχου είναι:

1. **Permissions/application requests** : Τις αιτήσεις για κατοχύρωση δικαιωμάτων πρόσβασης που κάθε εφαρμογή αιτείται.
2. **Exposed Functionalities** : Τις λειτουργικότητες που κάθε εφαρμογή προσφέρει στις άλλες.

Κάθε Android εφαρμογή αποτελείται από ένα αρχείο/πακέτο **app_name.apk** το οποίο όταν γίνει download σε μια συσκευή, ο intstaller αναλαμβάνει την αποκωδικοποίησή του και την εγκατάσταση της εφαρμογής. Μέσα στο **app_name.apk περιέχεται το AndroidManifest.xml αρχείο** το οποίο όπως έχουμε αναφέρει περιέχει όλες τις δηλώσεις ασφαλείας των components της εφαρμογής.

Με το **Apktool** είναι δυνατή η εξαγωγή του **AndroidManifest.xml** αρχείου προκειμένου να γίνει ο απαραίτητος έλεγχος των δικαιωμάτων που πρέπει να επικεντρωθεί στα εξής :

- Ευαίσθητα δεδομένα δεν μεταφέρονται μέσω της ICP επικοινωνίας των components
- Τα Intent-filters δεν χρησιμοποιούνται για θέματα ασφαλείας. Αν και μπορούν να περιορίσουν τους παραλήπτες των εσωτερικών μηνυμάτων πρέπει να συνοδεύονται από permissions.
- Τα Broadcasts Receivers δεν πρεπει να χρησιμοποιούνται όταν πρόκειται να μεταφέρουν ευαίσθητα δεδομένα.
- Τα permissions που ζητάει η εφαρμογή δεν πρέπει να είναι περισσότερα από αυτά που είναι αναγκαία για την υλοποίηση όλων των λειτουργικοτήτων της.

3. **Communications** : Όλες οι επικοινωνίες προς εξωτερικούς servers θα πρέπει να είναι κρυπτογραφημένη (χρήση SSL).

**HTTP Communications**

Για να μπορέσουμε να ανιχνέυσουμε την επικοινωνία των εφαρμογών, θα πρέπει να εγκατασταθεί ένας proxy server . Το **Burp Suite** είναι ένα κατάλληλο εργαλείο για την εγκατάσταση του proxy server στον υπολογιστή. Μετά την εγκατάσταση, δημιουργούμε τον proxy στην android συσκευή και ανοίγουμε έναν browser. Με την σύνδεση του browser σε μια ιστοσελίδα το burp suite μας εμφανίζει όλη την διαδικτυακή κίνηση της συσκευής.

4. **Data** : Ιδιαίτερη προσοχή πρέπει να δοθεί στον χειρισμό των δεδομένων από τις εφαρμογές. Τα Log files και Shared Preferences files δεν πρεπει να περιέχουν ευαίσθητα δεδομένα. Επίσης δεδομένα που μεταδίδονται για τις ανάκγες της εφαρμογής μέσω του δικτύου, δεν πρεπει να περιέχουν εμπιστευτικές πληροφορίες του χρήστη.

5. **Proper Use Of Cryptography** : Έλεγχος πρεπει να πραγματοποιηθεί και στις κρυπτογραφικές μεθόδους της εφαρμογής

- Γίνεται ο κατάλληλος έλεγχος των πιστοποιητικών ασφαλείας;
- Πως η εφαρμογή επικυρώνει ένα πιστοποιητικό ασφαλείας;

6. **Passing Information to Browsers**: Εάν η εφαρμογή χρησιμοποιεί κάποιον browser, πρεπει να γίνεται έλεγχος στο πως γίνεται η μεταφορά των δεδομένων από την εφαρμογή στον browser.

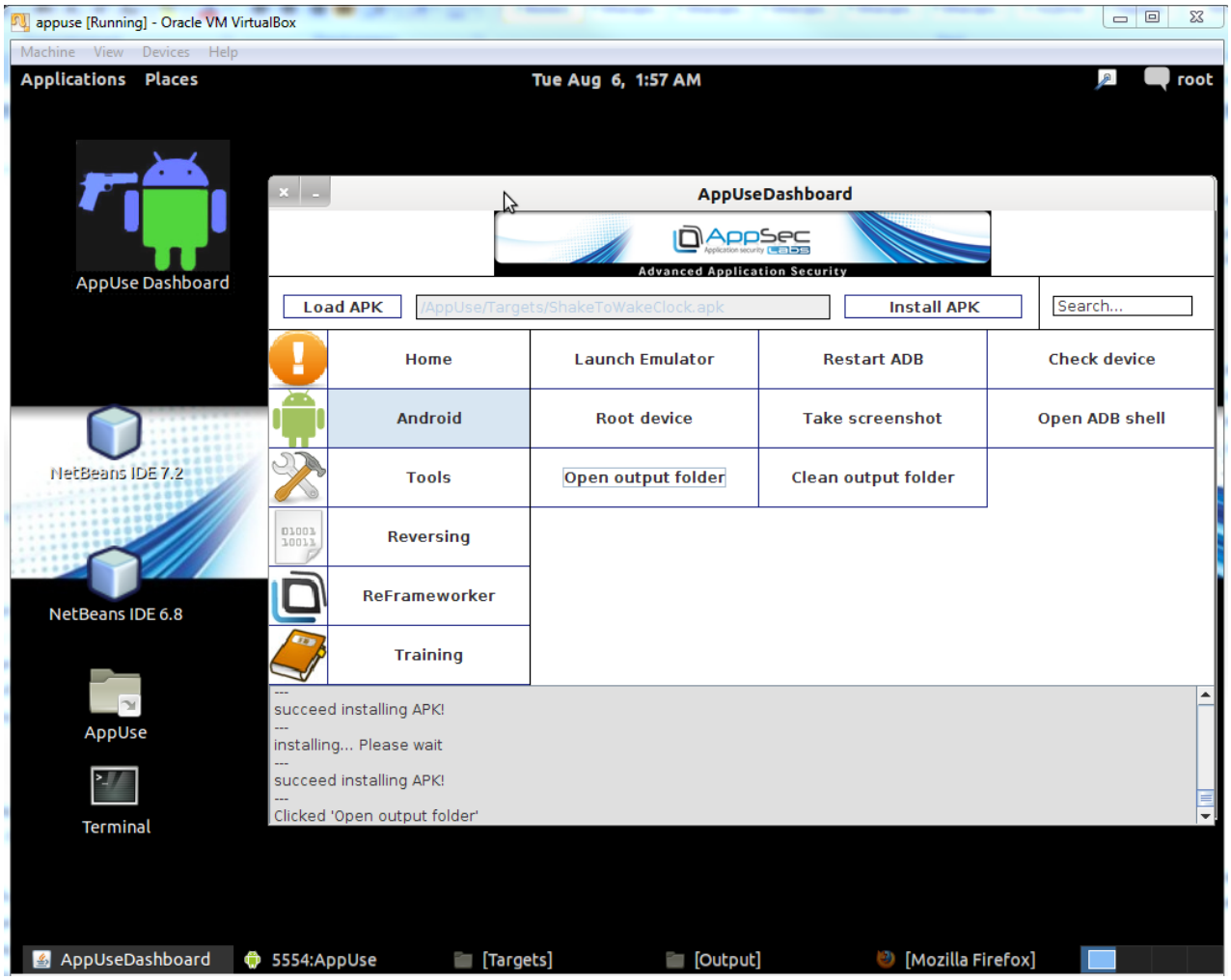
### *4.3.2 General Application Security*

Επειδή οι περισσότερες εφαρμογές που τρέχουν σε περιβάλλον Android είναι γραμμένες σε γλώσσα Java, θα πρέπει να γίνει ο απαραίτητος έλεγχος του κώδικα για την εύρεση τυχόν κενών ασφαλείας. Οι έλεγχοι που πραγματοποιούνται δεν διαφέρουν από τους αντίστοιχους ελέγχους στις κλασσικές εφαρμογές γι αυτό συνοπτικά θα αναφέρουμε τους κύριους τομείς έρευνας.

1. Authentication
2. Access Controls
3. Logs
4. Cryptography
5. Data Leakage
6. Data Validation
7. Error Reporting
8. Session Management
9. URL Parameters
10. Predictable Resources (tokens)

## 4.4 Case Study

Με την χρήση του AppUse tool (εικόνα 29) το οποίο είναι διαθέσιμο από την AppSec εκτελέσαμε Pen Test σε 10 εφαρμογές οι οποίες ήταν διαθέσιμες από ιστότοπο εκτός Android Market προκειμένου να ελεγχθούν σε θέματα ασφαλείας ως προς τα Android Specific Security Features, με σκοπό την εξαγωγή στατιστικών στοιχείων.




**Εικόνα 29**

Το Penetration Test πραγματοποιήθηκε στον προεγκατεστημένο emulator της εφαρμογής και όχι σε πραγματική συσκευή. (εικόνα 30)


**Εικόνα 30**

### *4.4.1 Permissions/application requests*

Με την χρήση του Apktool εγινάν extract τα AndroidManifest.xml αρχεία των εφαρμογών οπου εκεί αναγράφονται τα αιτούμενα permissions των εφαρμογών.

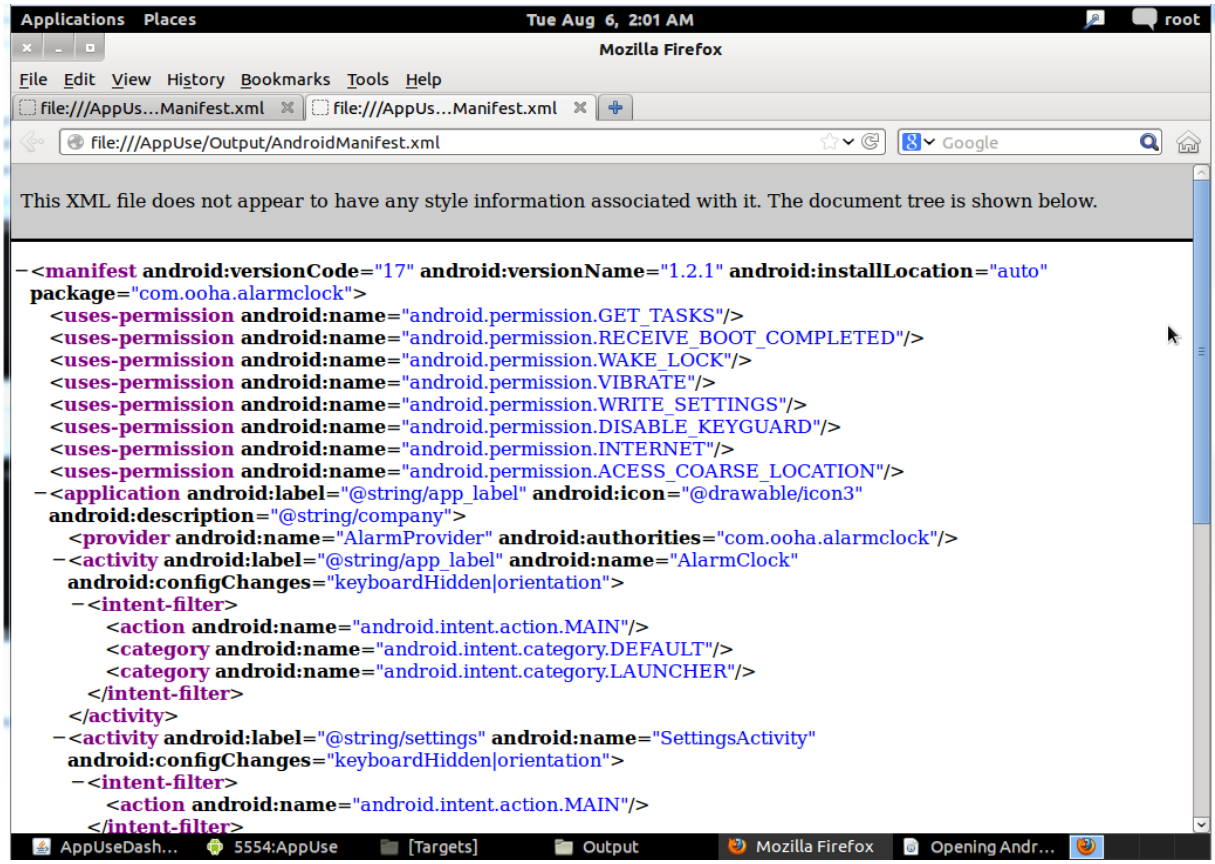
Αναλυτικότερα η κάθε εφαρμογή αιτείται τα παρακάτω δικαιώματα πρόσβασης σε υπηρεσίες του λειτουργικού συστήματος όπου αναγράφεται και ο Βαθμός επικινδυνότητας του κάθε αιτούμενου permission με βάση τον παρακάτω πίνακα [1].

| **Βαθμός Κινδύνου** | **Περιγραφή Κινδύνου** |
|---|---|
| 4 | Κρίσιμος |
| 3 | Μεγάλος |
| 2 | Μέτριος |
| 1 | Χωρίς κίνδυνο διαρροής πληροφοριών |

Πίνακας 8

### App1

Η App1 είναι μια εφαρμογή η οποία προσφέρει στον χρήστη ένα ρολόι με δυνατότητα εισαγωγής αφυπνήσεων. Τα δικαιώματα που αιτείται είναι:




Εικόνα 31

| Permission | Σκοπός | Απαιτείται | Βαθμός κινδύνου |
|---|---|---|---|
| GET_TASKS | Επιτρέπει στην εφαρμογή να λαμβάνει ειδοποιήσεις για τις εργασίες που εκτελούνται στην συσκευή | Δεν είναι απαραίτητο για μια εφαρμογή ξυπνητηριού. | 2 |
| RECEIVE_BOOT_COMPLETED | Επιτρέπει σε μια εφαρμογή να λαμβάνει το ACTION_BOOT_COMPLETED το οποίο εκπέμπεται από το σύστημα αμέσως μολις τελειώσει το boot της συσκευής | Πιθανώς προκειμένου να ενεργοποιείται αυτόματα η υπηρεσία ξυπνητηριού | 2 |
| WAKE_LOCK | Επιτρέπει την χρήση του PowerManager για να μείνει ο | Πιθανώς. | 1 |

| | | | |
|---|---|---|---|
| | επεξεργαστής σε κατάσταση λειτουργίας και όχι σε standby mode | | |
| VIBRATE | Επιτρέπει την πρόσβαση στον vibrator (δόνηση) της συσκευής | Απαραίτητο | 1 |
| WRITE_SETTINGS | Επιτρέπει την πρόσβαση (WRITE και READ) των settings του συστήματος | Είναι αναγκαίο για να μπορέσει να έχει πρόσβαση στα settings του Alarm, DATE, TIME κλπ αλλά αποκτά ταυτόχρονα πρόσβαση σε επικύνδινα settings όπως INSTALL_NON_MARKET_APPS, LOCATION_PROVIDERS_ALLOWED κλπ. | 2 |
| DISABLE_KEYGUARD | Επιτρέπει στην εφαρμογή να απενεργοποιεί το κλείδωμα της οθόνης (μετά από ένα διάστημα αδράνειας) | Πιθανώς | 2 |
| INTERNET | Επιτρέπει στην εφαρμογή να ανοίγει internet sockets. | Δεν είναι απαραίτητο | 3 |
| ACCESS_COARSE_LOCATION | Επιτρέπει σε μία εφαρμογή να έχει πρόσβαση στην τοποθεσία της συσκευής (μέσω wifi ή cell-id) | Δεν είναι απαραίτητο η εφαρμογή να λαμβάνει την τοποθεσία του χρήστη. | 2 |

Πίνακας 9

**App2**

Η App2 είναι μια εφαρμογή η οποία σύμφωνα με τον κατασκευαστή, προσφέρει ένα 3D template για την προβολή των επαφών, με δυνατότητα πραγματοποίησης κλήσεων και αποστολής μηνυμάτων μέσω της ίδιας της εφαρμογής. Τα permissions που αιτείται είναι:

| **Permission** | **Σκοπός** | **Απαιτείται** | **Βαθμός κινδύνου** |
|---|---|---|---|
| READ_CONTACTS | Επιτρέπει στην εφαρμογή να διαβάζει τις επαφές από την συσκευή | Είναι απαραίτητο (αλλά και επικύνδινο) | 3 |

| | | | |
|---|---|---|---|
| READ_SMS | Επιτρέπει στην εφαρμογή να διαβάζει τα SMS | Είναι απαραίτητο (αλλά και επικύνδινο) | 3 |
| WRITE_SMS | Επιτρέπει στην εφαρμογή να δημιουργεί SMS | Απαραίτητο. | 2 |
| CALL_PHONE | Επιτρέπει στην εφαρμογή να εκτελεί κλήσεις χωρίς την παρεμβολή του dialer user interface της συσκευής προκειμένου ο χρήστης δώσει την έγκρισή του για την πραγματοποίηση της κλήσης | Απαραίτητο | 4 |
| WRITE_EXTERNAL_STORAGE | Επιτρέπει στην εφαρμογή να αποθηκεύει δεδομένα σε εξωτερικές μνήμες | Δεν είναι απαραίτητο | 3 |

Πίνακας 10

**App3**

Η συγκεκριμένη εφαρμογή προσφέρει την GPS υπηρεσία της συσκευής και σε άλλες συσκευές (μέσω διασύνδεσης bluetooth)

| **Permission** | **Σκοπός** | **Απαιτείται** | **Βαθμός κινδύνου** |
|---|---|---|---|
| WRITE_EXTERNAL_STORAGE | Επιτρέπει σε μια εφαρμογή να γράφει σε εξωτερικές μνήμες SDRAM | Πιθανώς προκειμένου να αποθηκεύει τις συντεταγμένες στην εξωτερική μνήμη. | 3 |
| READ_LOGS | Επιτρέπει στην εφαρμογή να διαβάζει τα log αρχεία του συστήματος. | Δεν είναι απαραίτητο!!!!! | 3 |
| ACCESS_FINE_LOCATION | Επιτρέπει στην εφαρμογή να έχει πρόσβαση στην ακριβή τοποθεσία της συσκευής μέσω του GPS, των πύργων τηλεφωνικής τηλεφωνίας ή του wifi | Απάραίτητο | 2 |
| INTERNET | Επιτρέπει στην εφαρμογή να ανοίγει internet sockets. | Απαραίτητο | 3 |
| ACCESS_COARSE_LOCATION | Επιτρέπει σε μία εφαρμογή να έχει πρόσβαση στην τοποθεσία της συσκευής (μέσω wifi ή cell-towers) | Απαραίτητο για την εφαρμογή | 2 |

| | | | |
|---|---|---|---|
| BLUETOOTH_ADMIN | Επιτρέπει στην εφαρμογή να ανακαλύπτει και να επικυρώνει συσκευές Bluetooth | Απαραίτητο για την εφαρμογή | 2 |
| BLUETOOTH | Επιτρέπει στην εφαρμογή να συνδέεται σε επικυρωμένες συσκευές Bluetooth | Απαραίτητο για την εφαρμογή | 2 |

Πίνακας 11

**App4**

Η εφαρμογή App4 είναι μα εφαρμογή η οποία σύμφωνα με τον κατασκευαστή παρέχει : δωρεάν τηλεφωνικές κλήσεις στον Καναδά, δωρεάν τηλεφωνικές κλήσεις σε όλες τις συσκευές με εγκατεστημένη την συγκεκριμένη εφαρμογή παγκοσμίως, δωρεάν μηνύματα SMS, μετατροπή του tablet σε τηλεφωνική συσκευή, συγχρονισμό του τηλεφωνικού καταλόγου της συσκευής με τον κατάλογο φίλων στο Facebook. Η παρούσα εφαρμογή αιτείται τα παρακάτω permissions.

| **Permission** | **Σκοπός** | **Απαιτείται** | **Βαθμός κινδύνου** |
|---|---|---|---|
| MAPS_RECEIVE | Λαμβάνει χάρτες από τον server της εταιρείας | Απαιτείται από την εφαρμογή και το συγκεκριμένο permission είναι κατασκευασμένο από τον developer | 1 |
| C2D_MESSAGE | Επιτρέπει στην εφαρμογή να λαμβάνει μηνύματα από τον application server της εταιρείας | Απαιτείται για την λειτουργικότητα της εφαρμογής | 4 |
| INTERNET | Επιτρέπει στην εφαρμογή να δημιουργεί internet sockets | Απαιτείται για την επικοινωνία με τους servers της εταιρειας | 3 |
| READ_CONTACTS | Επιτρέπει στην εφαρμογή να διαβάζει τον τηλεφωνικό κατάλογο της συσκευής | Απαιτείται | 3 |
| RECORD_AUDIO | Επιτρέπει στην εφαρμογή να καταγράφει φωνή | Δεν απαιτείται !!!! | 3 |
| VIBRATE | Επιτρέπει στην εφαρμογή να έχει πρόσβαση στην δόνηση | Απαιτείται | 1 |
| MODIFY_AUDIO_SETTINGS | Επτρέπει στην εφαρμογή να τροποποιεί τις ρθμίσεις ήχου | Απαιτείται | 1 |

| | | | |
|---|---|---|---|
| | της συσκευής | | |
| WAKE_LOCK | Επιτρέπει στην εφαρμογή μς κρατάει ενεργό τον επεξεργαστή ή να εμποδίζει την οθόνη από το να σβήνει | Απαιτείται | 1 |
| DISABLE_KEYGUARD | Επιτρέπει στην εφαρμογή να απενεργοποιεί το κλείδωμα της οθόνης (μετά από ένα διάστημα αδράνειας) | Πιθανώς | 2 |
| ACCESS_WIFI_STATE | Επιτρέπει στην εφαρμογή να έχει πρόσβαση στις πληροφορίες για τα wifi δίκτυα | Απαιτείται καθώς η εφαρμογή στηρίζεται στην επικοινωνία μέσω internet | 1 |
| ACCESS_NETWORK_STATE | Επιτρέπει στην εφαρμογή να έχει πρόσβαση στις πληροφορίες για τα δίκτυα | Απαιτείται καθώς η εφαρμογή στηρίζεται στην επικοινωνία μέσω internet | 1 |
| WRITE_CONTACTS | Επιτρέπει στην εφαρμογή να γράφει στον κατάλογο επαφών | Απαιτείται | 3 |
| READ_PHONE_STATE | Επιτρέπει στην εφαρμογή να διαβάζει την κατάσταση του τηλεφώνου | Πιθανώς | 3 |
| RECEIVE_BOOT_COMPLETED | Επιτρέπει σε μια εφαρμογή να λαμβάνει το ACTION_BOOT_COMPLETED το οποίο εκπέμπεται από το σύστημα αμέσως μολις τελειώσει το boot της συσκευής | Πιθανώς προκειμένου να ενεργοποιείται αυτόματα η εφαρμογή | 2 |
| CALL_PHONE | Επιτρέπει στην εφαρμογή να εκτελεί κλήσεις χωρίς την παρεμβολή του dialer user interface της συσκευής προκειμένου ο χρήστης δώσει την έγκρισή του για την πραγματοποίηση της κλήσης | Απαραίτητο | 4 |
| BLUETOOTH_ADMIN | Επιτρέπει στην εφαρμογή να ανακαλύπτει και να | Δεν είναι απαραίτητο!!!! | 2 |

| | | | |
|---|---|---|---|
| | επικυρώνει συσκευές Bluetooth | | |
| BLUETOOTH | Επιτρέπει στην εφαρμογή να συνδέεται σε επικυρωμένες συσκευές Bluetooth | Δεν είναι απαραίτητο!!!! | 2 |
| WRITE_SETTINGS | Επιτρέπει στην εφαρμογή να διαβάζει ή να γράφει τις ρυθμίσεις του συστήματος | Δεν είναι απαραίτητο!!!! | 2 |
| CHANGE_WIFI_STATE | Επτρέπει στην εφαρμογή να τροποποιήσει την κατάσταση wifi σύνδεσης της συσκευής | Πιθανώς | 1 |
| BROADCAST_STICKY | Επιτρέπει στην εφαρμογή να εκπέμπει μόνιμα intents | Απαραίτητο | 2 |
| WRITE_EXTERNAL_MEMORY | Επιτρέπει στην εφαρμογή να γράφει σε εξωτερικές μνήμες | Πιθανώς | 3 |
| C2DM_RECEIVE | Η εφαρμογή χρησιμοποιεί την Android Cloud to Device Messaging (C2DM) υπηρεσία προκειμένου να μπορεί να αποστείλει μημτα από τους servers της εταιρείας προς τις συσκευές. | Πιθανώς | 4 |

Πίνακας 12

**App5**

Η συγκεκριμένη εφαρμογή προσφέρει ένα νέο interface για την αποστολή – λήψη μηνυμάτων SMS/MMS, δημιουργία CHAT, προγραμματισμένη αποστολή SMS, δημιουργία backup των SMS και δυνατότητα αποστολής του σε email λογαριασμό, εμφανιση φωτογραφιών facebook στην λίστα επαφών κα:

| **Permission** | **Σκοπός** | **Απαιτείται** | **Βαθμός κινδύνου** |
|---|---|---|---|
| RECEIVE_BOOT_COMPLETED | Επιτρέπει σε μια εφαρμογή να λαμβάνει το ACTION_BOOT_COMPLETED το οποίο εκπέμπεται από το σύστημα αμέσως μολις τελειώσει το boot της | Πιθανώς προκειμένου να ενεργοποιείται αυτόματα η εφαρμογή | 2 |

| | | | |
|---|---|---|---|
| | συσκευής | | |
| CALL_PHONE | Επιτρέπει στην εφαρμογή να εκτελεί κλήσεις χωρίς την παρεμβολή του dialer user interface της συσκευής προκειμένου ο χρήστης δώσει την έγκρισή του για την πραγματοποίηση της κλήσης | Απαραίτητο | 4 |
| READ_CONTACTS | Επιτρέπει στην εφαρμογή να διαβάζει τον τηλεφωνικό κατάλογο της συσκευής | Απαιτείται | 3 |
| WRITE_CONTACTS | Επιτρέπει στην εφαρμογή να γράφει στον κατάλογο επαφών | Απαιτείται | 3 |
| RECEIVE_SMS | Επιτρέπει στην εφαρμογή να λαμβανει εισερχόμενα SMS | Απαιτείται | 3 |
| RECEIVE _MMS | Επιτρέπει στην εφαρμογή να λαμβανει εισερχόμενα MMS | Απαιτείται | 3 |
| SEND_SMS | Επιτρέπει στην εφαρμογή να στέλνει SMS μηνύματα | Απαιτείται | 3 |
| INTERNET | Επιτρέπει στην εφαρμογή να δημιουργεί internet sockets | Απαιτείται για τον συγχρονισμό της εφαρμογής με την εφαρμογή facebook | 3 |
| READ_SMS | Επιτρέπει στην εφαρμογή να διαβάζει SMS μηνύματα | Απαιτείται | 3 |
| READ_MMS | Επιτρέπει στην εφαρμογή να διαβάζει MMS μηνύματα | Απαιτείται | 3 |
| ACCESS_NETWORK_STATE | Επιτρέπει στην εφαρμογή να έχει πρόσβαση στις πληροφορίες για τα δίκτυα | Απαιτείται καθώς η εφαρμογή στηρίζεται στην επικοινωνία μέσω internet | 1 |

| | | | |
|---|---|---|---|
| CHANGE_NETWORK_STATE | Επιτρέπει στην εφαρμογή να τροποποιεί την κατάσταση συνδεσιμότητας σε δίκτυο της συσκευής | Πιθανως | 1 |
| READ_PHONE_STATE | Επιτρέπει στην εφαρμογή να διαβάζει την κατάσταση του τηλεφώνου | Πιθανώς | 3 |
| WAKE_LOCK | Επιτρέπει στην εφαρμογή μς κρατάει ενεργό τον επεξεργαστή ή να εμποδίζει την οθόνη από το να σβήνει | Απαιτείται | 1 |
| DISABLE_KEYGUARD | Επιτρέπει στην εφαρμογή να απενεργοποιεί το κλείδωμα της οθόνης (μετά από ένα διάστημα αδράνειας) | Πιθανώς | 2 |
| WRITE_EXTERNAL_STORAGE | Επιτρέπει στην εφαρμογή να γράφει σε εξωτερικές μνήμες | Πιθανώς | 3 |
| READ_LOGS | Επιτρέπει μια εφαρμογή να έχει πρόσβαση στα log αρχεία του συστήματος | Απαιτείται για την λειτουργικότητα backup | 3 |
| PERSISTENT_ACTIVITY | Επιτρέπει στην εφαρμογή να δημιουργεί activities που είναι μόνιμα | Δεν απαιτείται (η συγκεκριμένη λειτουργικότητα στο μέλλον θα αφαιρεθεί από το android λειτουργικό) | 3 |
| GET_ACCOUNTS | Επιτρέπει την πρόσβαση στην λίστα των λογαριασμών των εφαρμογών | Πιθανώς | 3 |
| RECORD_AUDIO | Επιτρέπει την καταγραφή συνομιλιών | Δεν απαιτείται!!!! | 3 |
| MODIFY_AUDIO_SETTINGS | Επιτρέπει την τροποποίηση των ρυθμίσεων ήχου. | Απαιτείται για την λειτουργία των κλήσεων. | 1 |

Πίνακας 13

**App6**

Η συγκεκριμένη εφαρμογή δημιουργεί ένα background image με την διαθέσιμη ενέργεια της μπαταρίας της συσκευής. Τα permissions που απαιτεί είναι:

| Permission | Σκοπός | Απαιτείται | Βαθμός κινδύνου |
|---|---|---|---|
| BIND_WALLPAPER | Επιτρέπει σε ένα service τύπου wallpaperservice της εφαρμογής να δεσμέυσει το wallpaperservice της συσκευής για χρήση | Απαραίτητο | 1 |
| BATTERY_STATS | Επιτρέπει την συλλογή στατιστικών στοιχείων της μπαταρίας | Απαραίτητο | 1 |
| ACTION_POWER_ CONNECTED | Broadcast action: Εξωτερική πηγή ενέργειας έχει συνδεθεί στην συσκευή. | Απαιτείται | 1 |
| ACTION_POWER_ DISCONNECTED | Broadcast action: Εξωτερική πηγή ενέργειας έχει αποσυνδεθεί από τη συσκευή. | Απαιτείται | 1 |

Πίνακας 14

**App7**

Η εφαρμογή App7 είναι ένα παχνίδι bowling. Τα permissions που αιτείται είναι :

| Permission | Σκοπός | Απαιτείται | Βαθμός κινδύνου |
|---|---|---|---|
| INTERNET | Επιτρέπει στην εφαρμογή να δημιουργεί INTERNT sockets | Δεν είναι απαραίτητο | 3 |
| WAKE_LOCK | Επιτρέπει στην εφαρμογή μς κρατάει ενεργό τον επεξεργαστή ή να εμποδίζει την οθόνη από το να σβήνει | Απαιτείται | 1 |

Πίνακας 15

**App8**

Η εφαρμογή προσθέτει στα εισερχόμενα και εξερχόμενα μηνυματα SMS την ακριβή ώρα αποστολής τους. Τα δικαιώματα που χρησιμοποιεί είναι:

| Permission | Σκοπός | Απαιτείται | Βαθμός κινδύνου |
|---|---|---|---|
| RECEIVE_SMS | Επιτρέπει στην εφαρμογή να έχει πρόσβαση στα χαρακτηριστικά των εισερχομένων SMS | Απαραίτητο προκειμένου να διαβάσει την ακριβή ημερομηνία αποστολής των εισρεχομένων SMS | 3 |
| READ/WRITE SMS | Επιτρέπε στην εφαρμογή να τροποιεί τα SMS | Απαραίτητο προκειμένου να τοποθετηθεί η ώρα αποστολής στα εξερχόμενα μνμτα | 3 |
| WRITE_EXTERNAL_STORAGE | Επιτρέπει στην εφαρμογή να γράφει σε εξωτερκές πηγές. | Απαραίτητο για την καταγραφή των log files | 3 |

Πίνακας 16

**App9**

Η εφαρμογή 9 χρησιμοποιεί την ψηφιακή πυξίδα της συσκευής για να τοποθετήσει γεωγραφικά σε σχέση με την συσκευή, τα ασύρματα δίκτυα της περιοχής. Τα permissions που χρησιμοποιεί είναι:

| Permission | Σκοπός | Απαιτείται | Βαθμός κινδύνου |
|---|---|---|---|
| ACCESS_WIFI_STATE | Επιτρέπει στην εφαρμογή να έχει πρόσβαση στις πληροφορίες για τα wifi δίκτυα | Απαραίτητο | 1 |
| CHANGE_WIFI_STATE | Επιτρέπε στην εφαρμογή να τροποιεί την κατάσταση συνδεσιμότητας σε wifi δίκτυα | Απαραίτητο | 2 |
| ACCESS_NETWORK_STATE | Επιτρέπει στην εφαρμογή να έχει πρόσβαση στις πληροφορίες για τα δίκτυα | Απαραίτητο | 1 |
| INTERNET | Επιτρέπει στην εφαρμογή να δημιουργεί internet sockets | Δεν είναι απαραίτητο !!! | 3 |

Πίνακας 17

**App10**

Η εφαρμογή παρέχει ένα σύνολο από εργαλεία χρήσιμα στον χρήστη όπως: φακός, μετατροπέας μονάδων, χάρακας, πυξίδα, χρονόμετρο, αντίστροφη μέτρηση, κομπιουτεράκι, μεγενθυντικό φακό και καθρέφτη. Τα permissions που αιτείται είναι:

| Permission | Σκοπός | Απαιτείται | Βαθμός κινδύνου |
|---|---|---|---|
| MODIFY_AUDIO_SETTINGS | Επιτρέπει την τροποποίηση των ρυθμίσεων του ήχου | Δεν είναι απαραίτητο | 1 |
| WAKE_LOCK | Επιτρέπει στην εφαρμογή να κρατάει ενεργό τον επεξεργαστή ή να εμποδίζει την οθόνη από το να σβήνει | Απαιτείται | 1 |
| VIBRATE | Επιτρέπει στην εφαρμογή να έχει πρόσβαση στην δόνηση | Πιθανώς | 1 |
| FLASHLIGHT | Επιτρέπει την πρόσβαση στο flash της συσκευής (εάν υπάρχει) | Απαραίτητο για το εργαλείο φακος | 1 |
| CAMERA | Επιτρέπει την πρόσβαση στην κάμερα | Απαραιτητο για ορισμένες συσκευές στις οποίες το flash ενεργοποιείται μόνο εάν η κάμερα είναι σε λειτουργία | 4 |

Πίνακας 18

### *4.4.2 Λειτουργικότητες των Εφαρμογών*

**App1**

Όσα components έχουν Intent-filters, όπως αναφέραμε, γίνονται αυτόματα exported. Η συγκεκριμένη εφαρμογή παρατηρούμε ότι προσφέρει τις παρακάτω λειτουργικότητες :

| IPC component | Intent Filter | Ανάλυση |
|---|---|---|
| ACTIVITY<br>Alarm clock | Android.intent.action.MAIN | Η κύρια ενέργεια όταν εκκινεί η εφαρμογή. Απαιτείται |
| ACTIVITY<br>Settings activity | Android.intent.action.MAIN | Αποτελέι το activity με το οποίο ο χρήστης τοποθετεί τις επιθυμητές ώρες αφύπνισης. Το activity είναι προσπελάσιμο με το ίδιο intent – filter με το main activity!!!!<br>Απαιτείται |
| CONTENT PROVIDER<br>AlarmProvider | - | Αποτελεί την βάση δεδομένων όπου αποθηκέυονται οι αφυπνίσεις. Δεν είναι προσβάσιμη από άλλες εφαρμογές καθώς είναι ιδιωτικό component. Απαιτείται |

Πίνακας 19

**App2**

Η εφαρμογή που χειρίζεται τις επαφές του χρήστη έχει τα παρακάτω δημόσια components:

| IPC component | Intent Filter | Ανάλυση |
|---|---|---|
| ACTIVITY<br>Download activity | Android.intent.action.MAIN | Έιναι ένα activity το οποίο κατεβάζει αυτόματα ενημερώσεις για νέες εκδόσεις της εφαρμογής. Εκκινεί ταυτόχρονα με την MAIN activity της εφαρμογής. Δεν απαιτείται |

| | | |
|---|---|---|
| ACTIVITY<br>Main activity | Android.intent.action.MAIN | Η κύρια ενέργεια όταν εκκινεί η εφαρμογή. Απαιτείται |
| ACTIVITY<br>PickContactActivity | Android.intent.action.MAIN | Είναι η ενέργεια για να επιλεγεί κάποια επαφή από τον κατάλογος (για εκτέλεση κλήσης ή αποστολή μηνύματος).<br>Απαιτείται. |
| ACTIVITY<br>ContactInfoActivity | Android.intent.action.MAIN | Προσφέρει τις πληροφορίες που σχετίζονται με την επαφή που επιλέχτηκε.<br>Απαιτείται |
| ACTIVITY<br>ChangePhotoActivity | Android.intent.action.MAIN | Δυνατότητα αλλαγής φωτογραφίας επαφής.<br>Απαιτείται |
| ACTIVITY<br>SettingsActivity | Android.intent.action.MAIN | Δυνατότητα τροποποίησης ρυθμίσεων εφαρμογής.<br>Απαιτείται |
| ACTIVITY<br>RemoveContactDialog | Android.intent.action.MAIN | Παράθυρο διαλόγου για την διαγραφή μιας επαφής.<br>Απαιτείται. |
| SERVICE<br>Main Service | Com.gtp.nextlauncher.widget.contact.MainService | Διαθέσιμη υπηρεσία. Μπορεί να απαιτείται αλλά δεν είναι αναγκαίο να υπάρχει υπηρεσία για την διαχείριση των επαφών. |
| CONTENT PROVIDER | - | Αποτελεί την βάση |

| | | |
|---|---|---|
| Data Provider | | δεδομένων όπου η εφαρμογή αποθηκεύει τα δεδομένα. Δεν είναι δημόσιο component και επιπλέον έχουν ενεργοποιηθεί τα URI permissions.<br>Απαιτείται. |

Πίνακας 20

**App3**

Η εφαρμογή που προσφέρει GPS σε άλλες συσκευές μέσω Bluetooth:

| IPC component | Intent Filter | Ανάλυση |
|---|---|---|
| ACTIVITY<br>App_name | Android.intent.action.MAIN | Η κύρια ενέργεια όταν εκκινεί η εφαρμογή. Απαιτείται |
| ACTIVITY<br>DeviceListActivity | - | Αυτή η ενέργεια επιτρέπει να εμφανιστεί μια λίστα με τις διαθέσιμες bluetooth συσκευές. Απαιτείται και είναι ιδιωτική ενέργεια. |
| ACTIVITY<br>UnlockActivity | - | Αυτή η ενέργεια δίνει την δυνατότητα να ξεκλειδώσει κάποιο άλλο ACTIVITY το οποίο έχει περιορισμό πρόσβασης. Δεν είναι απαραίτητο αλλά και αυτό το component είναι private |
| ACTIVITY<br>Restart Process | - | Επανεκκίνηση της όλης διαδικασίας. Πιθανώς να είναι απαραίτητο για την εφαρμογή και είναι ιδιωτικό component. |

Πίνακας 21

**App4**

Η εφαρμογή που προσφέρει δωρεάν κλήσεις στον Καναδά:

| IPC component | Intent Filter | Ανάλυση |
|---|---|---|
| ACTIVITY<br>Main Activity | MainActivity.DIALPAD<br>MainActivity.HISTORY<br>MainActivity.VOICEMAIL<br>MainActivity.MESSAGES<br>MainActivity.CONTACTS<br>MainActivity.SEARCH | Η κύρια ενέργεια της εφαρμογής που προσφέρει τις λειτουργικότητες που φαίνονται δίπλα. Είναι απαραίτητη για την εφαρμογή |
| ACTIVITY<br>CallActivity | CallActivity.IN_CALL<br>CallActivity.INCOMING_CALL | Ενέργεια για την πραγματοποίηση κλήσεων. Απαραίτητο για την εφαρμογή |
| SERVICE<br>Phone Service | - | Παρέχει μια υπηρεσία για την πραγματοποίηση κλήσεων. Δεν είναι απαραίτητο. |
| SERVICE<br>CampaingnTrackingService | - | Υπηρεσία της google για την παρακολούθηση των προτιμήσεων του χρήστη και την αποστολή τους στην εφαρμογή, για προσαρμογή διαφημιστικής καμπάνιας. Δεν είναι απαραίτητο. |
| BROADCAST RECEIVER<br>CampaingnTrackingReceiver | Com.android.vending.INSTALL_REFERRER | Δέχεται intents για την υλοποίηση της παραπάνω υπηρεσίας της Google |
| RECEIVER | Android.intent.action.BOOT | Δέχεται το intent όταν |

| | | |
|---|---|---|
| BootCompleted | _COMPLETED | τελειώσει το boot της συσκευής. Πιθανώς να είναι απαραίτητο προκειμένου να εκκινήσουν τα services με το πέρας του boot process |
| RECEIVER<br>PhonePushReceiver | Com.google.android.c2dm.<br>intent.RECEIVE<br><br>Android.permission=…..<br>c2dm.SEND | Δέχεται intents από τον server της εταιρείας χρησιμοποιώντας το c2dm framework. Είναι προστατευμένο component κάνωντας χρήση permissions αλλά δεν είναι απαραίτητο. |

Πίνακας 22

**App5**

Η εφαρμογή που δημιουργεί ένα νέο interface για την αποστολή – λήψη μηνυμάτων SMS/MMS:

| IPC component | Intent Filter | Ανάλυση |
|---|---|---|
| ACTIVITY<br>PrivateBoxContactActivity | - | Εμφανίζει τις επαφές σε ένα πλάισιο. Απαραίτητη για την εφαρμογή |
| ACTIVITY<br>PrivateBoxPreferences | - | Ενέργεια για την τροποποίηση του παραπάνω πλαισίου. Απαραίτητη για την εφαρμογή |
| ACTIVITY<br>MAIN | Android.intent.action.<br>MAIN | Η κύρια ενέργεια όταν εκκινεί η εφαρμογή. Απαιτείται |
| RECEIVER<br>BootCompleted | Android.intent.action.<br>BOOT _COMPLETED | Δέχεται το intent όταν τελειώσει το boot της συσκευής. Πιθανώς να είναι απαραίτητο προκειμένου να εκκινήσουν τα |

| | | |
|---|---|---|
| | | services με το πέρας του boot process |
| CONTENT PROVIDER<br>GoSmsProvider | Exported=”false” | Αποτελεί την βάση δεδομένων όπου αποθηκεύονται τα μηνύματα SMS του χρήστη. Ιδιωτικό component και είναι απαραίτητο για την λειτουργία της εφαρμογής |
| CONTENT_PROVIDER<br>GoMmsProvider | Exported=”false” | Αποτελεί την βάση δεδομένων όπου αποθηκεύονται τα μηνύματα MMS του χρήστη. Ιδιωτικό component και είναι απαραίτητο για την λειτουργία της εφαρμογής |

Πίνακας 23

**App6**

Η εφαρμογή που δημιουργεί ένα background image με την διαθέσιμη μπαταρία της συσκευής:

| IPC component | Intent Filter | Ανάλυση |
|---|---|---|
| ACTIVITY<br>MyBatteryWallpaper | Exported=”true” | Αποτελεί την κύρια ενέργεια της εφαρμογής. Είναι απαραίτητη. |
| SERVICE<br>MyBatteryWallpaperService | Android.service.wallpaper.<br>WallpaperService<br>Permission:=BIND_WALLPAPER | Η εφαρμογή παρέχει μια υπηρεσία για την αλλαγή του wallpaper της συσκευής. Είναι ιδιωτική υπηρεσία και προστατεύεται από permissions. Είναι απαραίτητη για την εφαρμογή. |

Πίνακας 24

**App7**

Η εφαρμογή 7 είναι ένα παιχνίδι bowling με τα εξής components:

| IPC component | Intent Filter | Ανάλυση |
|---|---|---|
| ACTIVITY UtilityplayerProxyactivity | Android.intent.action.MAIN | Αποτελεί την κύρια ενέργεια της εφαρμογής και απαιτείται |
| ACTIVITY Unityplayeractivity | - | Είναι ενέργεια που αποικονίζει τα στοιχεία του χρήστη στο παιχνίδι. Απαιτείται |
| ACTIVITY UnityPlayerNativeActivity | - | Είναι η ενέργεια που κρατάει στοιχεία για τα παιχνίδια που έχει πραγματοποιήσει ο χρήστης. Απαιτείται |

Πίνακας 25

**App8**

Τα components της εφαρμογής που εμφανίζει την ακριβή ώρα αποστολής των μηνυμάτων SMS/MMS είναι:

| IPC component | Intent Filter | Ανάλυση |
|---|---|---|
| ACTIVITY TimeFixActivity | Android.intent.action.MAIN | Αποτελεί την κύρια ενέργεια της εφαρμογής και απαιτείται |
| ACTIVITY SMSTimeFixPrefs | - | Προβάλει τις ρυθμίσεις της εφαρμογής. Απαιτείται και είναι ιδιωτικό component |
| BROADCAST RECEIVER SMS.Receiver | SMS_RECEIVED | Η εφαρμογή είναι διαθέσιμη να δεχτεί broadcasted intents του τύπου SMS_RECEIVED τα οποία εκπέμπονται από το λειτουργικό σύστημα όταν |

| | | |
|---|---|---|
| | | γίνεται εισερχόμενο ένα SMS. Απαιτείται από την εφαρμογή. |
| SERVICE<br>SMSTimeFixService | - | Η εφαρμογή δημιουργεί μια υπηρεσία που τρέχει στο παρασκήνιο προκειμένου συνέχεια να είναι σε θέση να λαμβάνει τα SMS. Απαιτείται. |

Πίνακας 26

**App9**

Η εφαρμογή που χρησιμοποιεί την ψηφιακή πυξίδα για να δημιουργήσει έωα χάρτη με τα διαθέσιμα ασύρματα δίκτυα έχει τα παρακάτω δημόσια components:

| IPC component | Intent Filter | Ανάλυση |
|---|---|---|
| ACTIVITY<br>Wifiradar.main | Android.intent.action.MAIN | Αποτελεί την κύρια ενέργεια της εφαρμογής όταν εκκινεί και είναι απαραίτητη για την εφαρμογή |
| ACTIVITY<br>Wifiradar.mylist | - | Παρουσιάζει μια λίστα με τα διαθέσιμα wifi δίκτυα . Απαραίτητη για την εφαρμογή |
| ACTIVITY<br>Com.google.ads.AdActivity | - | Το συγκεκριμένο activity προσφέρει την δυνατότητα στον developer να εμφανίζει διαφημιστικές καμπάνιες της Google εντός της εφαρμογής! Δεν είναι απαραίτητο από την εφαρμογή, αλλά είναι ο λόγος για τον οποίο η συγκεκριμένη εφαρμογή |

| | | |
|---|---|---|
| | | αιτείται το permission INTERNET ενώ δεν είναι απαιτούμενο για τις λρειτουργικότητες της εφαρμογής. |

Πίνακας 27

**App10**

Τέλος η εφαρμογή που δημιουργεί ένα χρήσιμο πολύ εργαλείο από ψηφιακά gadgets έχει τα εξής components:

| IPC component | Intent Filter | Ανάλυση |
|---|---|---|
| ACTIVITY<br>knife.main | Android.intent.action.MAIN | Αποτελεί την κύρια ενέργεια της εφαρμογής όταν εκκινεί και είναι απαραίτητη για την εφαρμογή |
| ACTIVITY<br>unitConverter activity | Android.intent.action.MAIN | Με το activity αυτό εκκινεί το εργαλείο για την μετατροπή μονάδων μέτρησης. Απαραίτητο για την εφαρμογή και είναι διαθέσιμο σε όλα τα components μέσω του MAIN intent!!!! |
| ACTIVITY<br>flashlight activity | - | Με το activity αυτό εκκινεί το εργαλείο για τον φακό. Απαραίτητο και private. |
| ACTIVITY<br>Ruler activity | - | Με το activity αυτό εκκινεί το εργαλείο για τον χάρακα. Απαραίτητο και private. |
| ACTIVITY<br>timer activity | Android.intent.action.MAIN | Με το activity αυτό εκκινεί το εργαλείο για τον χάρακα. Απαραίτητο για την εφαρμογή και είναι |

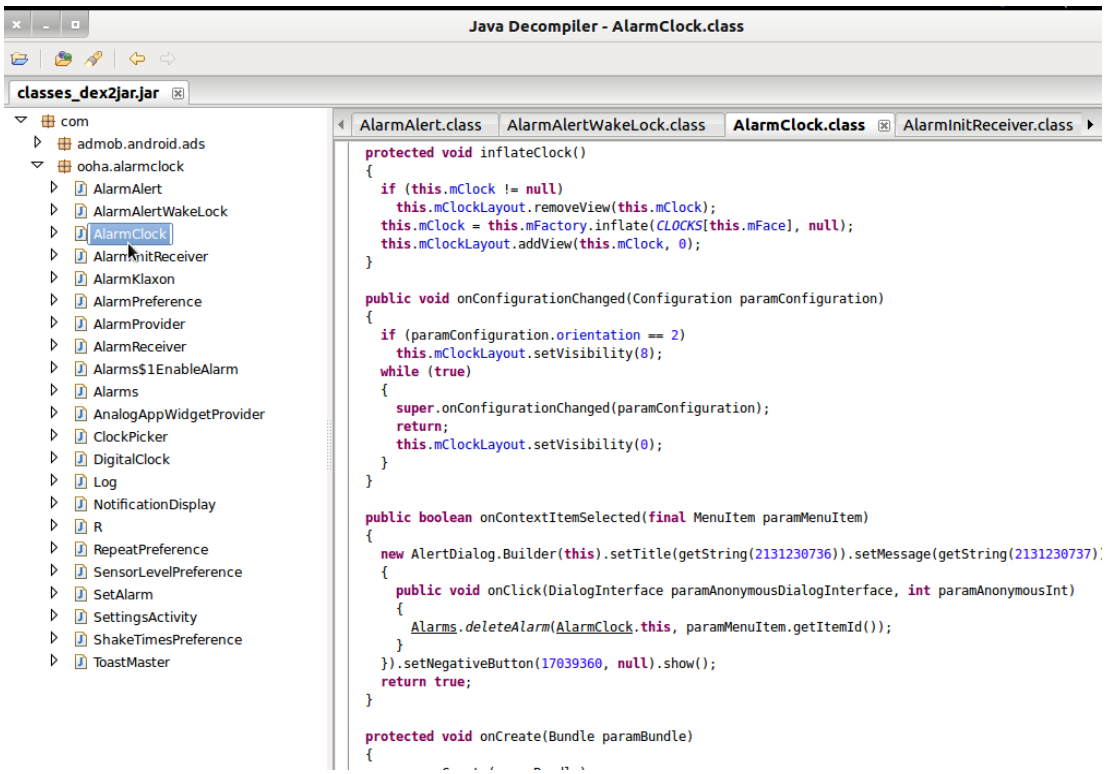
| | | |
|---|---|---|
| | | διαθέσιμο σε όλα τα components μέσω του MAIN intent!!!! |
| ACTIVITY<br>compass activity | - | Με το activity αυτό εκκινεί το εργαλείο για την πυξίδα. Απαραίτητο και private. |
| ACTIVITY<br>chrono activity | - | Με το activity αυτό εκκινεί το εργαλείο για το χρονόμετρο. Απαραίτητο και private. |
| ACTIVITY<br>Magn Glass activity | - | Με το activity αυτό εκκινεί το εργαλείο για τον μεγεθυντικό φακό. Απαραίτητο και private. |
| ACTIVITY<br>Level activity | - | Με το activity αυτό εκκινεί το εργαλείο για την αντίστροφη μέτρηση. Απαραίτητο και private. |
| ACTIVITY<br>About activity | - | Με το activity αυτό εκκινεί ένα πλαίσιο με οδηγίες χρήσεως. Απαραίτητο και private. |
| ACTIVITY<br>calculator activity | Android.intent.action.MAIN | Με το activity αυτό εκκινεί το εργαλείο για το κομπιουτεράκι. Απαραίτητο για την εφαρμογή και είναι διαθέσιμο σε όλα τα components μέσω του MAIN intent!!!! |
| ACTIVITY<br>Open Dialog activity | - | Με το activity αυτό εκκινεί ένα πλαίσιο με ερώτηση από την εφαρμογή προς |

| | | |
|---|---|---|
| | | τον χρήστη εάν επιθυμεί να συνεχίσει προκειμένου να ενεργοποιηθούν κάποια tools. Απαραίτητο και private καθώς το πλαίσιο διαλόγου εκκινεί για τα activities που ενεργοποιούν επικύνδινα components. |
| SERVICE<br>LocalService | Com.digital_and_dreams.<br>android.common.<br>LOCAL_SERVICE | Η εφαρμογή δημιουργεί μια τοπική υπηρεσία. Πιθανώς να είναι απαραίτητο. |
| BROADCAST-RECEIVER<br>Timer_expired | Swissarmyknife.intent.<br>action.TIMER_EXPIRED | Η εφαρμογή μπορεί να παραλάβει broadcasted intents του τύπου TIMER _EXRIRED τα οποία δημιουργεί η ίδια η εταιρεία. Πιθανώς να είναι απαραίτητο. |

Πίνακας 28

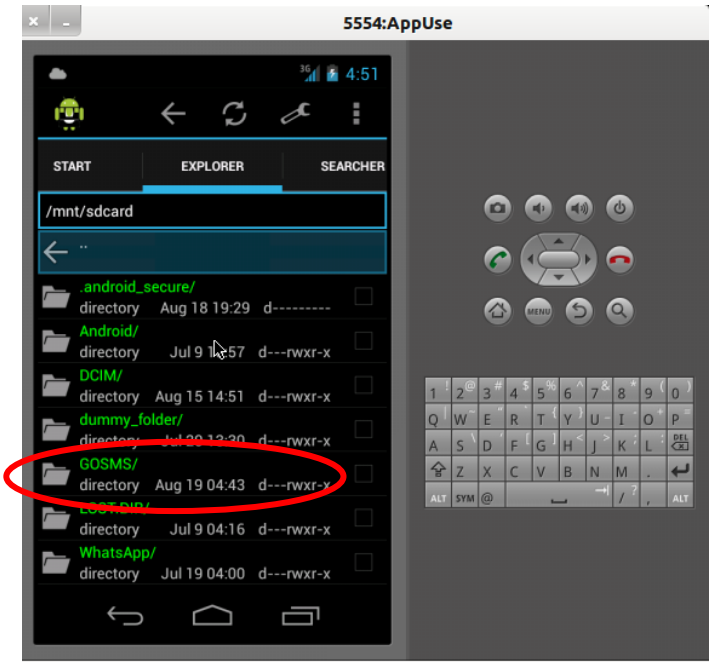
### *4.4.3 Communication- Data – Cryptography*

Με χρήση ένος Java Decompiler είναι δυνατή η εμφάνιση των java κλάσσεων των εφαρμογών προκειμένου να γίνει έλεγχος της επικοινωνίας και τα δεδομενα που ανταλλάσουν μεταξύ τους τα components. (εικόνα 31)



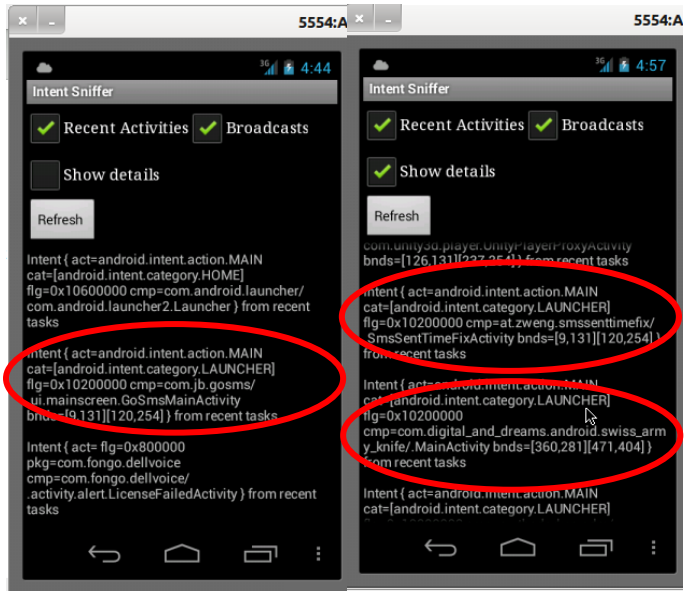

**Εικόνα 32**

Από τον έλεγχο που πραγματοποιήθηκε καμίας εφαρμογή δεν χρησιμοποιεί τα Shared Preferences files, ενώ μόλις μια κάνει χρήση της εξωτερική μνήμης για αποθήκευση των δεδομένων και η οποία κάνει χρήση του SHA-256 προκειμένου να εξασφαλίσει την εμπιστευτικότητα των δεδομένων (εικόνα 32). Επιπλέον όλες οι εφαρμογές κάνουν χρήση του πρωτοκόλλου SSL για την αξιόπιστη επικοινωνία με εξωτερικούς server.




**Εικόνα 33**

Επιπλέον με ένα Intent sniffer εργαλείο εκκινήθηκαν όλες οι εφαρμογές προκειμένου να διαπιστωθεί η επικοινωνία μέσω intents και τυχόν broadcasted intents που οι εφαρμογές εκπέμπουν. (εικόνα 33)




**Εικόνα 34**

Ο έλεγχος έδειξε φυσιολογική δραστηριότητα των intents (με την έναρξη των εφαρμογών εκκινεί μόνο η MAIN activity) ενώ δεν υπήρχε κανένα broadcasted intent.

### *4.4.5 Αποτελέσματα*

Οι εφαρμογές που εξετάστηκαν είναι διαθέσιμες μέσω των ανοιχτών ιστοτόπων που προσφέρουν δωρεάν υπηρεσίες για εγκατάσταση εφαρμογών Android, παρακάμπτωντας το Android market. Αν και όλες οι εφαρμογές φαινομενικά φαίνονται ακίνδυνες για τα δεδομένα του χρήστη η ελλειπής υλοποίησή τους τις κάνει ευάλωτες σε συγκεκριμένες επιθέσεις. Χαρακτηριστικό του ελέγχου που πραγματοποιήθηκε, ότι καμία εφαρμογή δεν χρησιμοποιεί ειδικά permissions (παράγραφος 3.2.4) για την πρόσβαση στα components καθιστώντας τες ευάλωτες σε intent-based επιθέσεις (παράγραφος 3.3.1.2). Ο παρακάτω συγκεντρωτικός πίνακας παρουσιάζει τα στοιχεία ασφαλείας των Android εφαρμογών που ελέγχθηκαν:

| | **Permissions που δεν απαιτούνται / συνολικά** | **Activities που δεν απαιτούνται / συνολικά** | **Providers που δεν απαιτούνται / συνολικά** | **Receivers που δεν απαιτούνται / συνολικά** | **Services που δεν απαιτούνται / συνολικά** | **Public Components προστατευμένα με permissions** | **Πιθανές Τρωτότητες** |
|---|---|---|---|---|---|---|---|
| **App1** | 3/8 | 0/1 | 0/1 | - | - | 0/1 | PE, IB * |
| **App2** | 1/5 | 1/6 | 0/1 | - | 1/1 | 0/7 | PE, IB |
| **App3** | 2/7 | 0/3 | - | - | - | 0/1 | PE, IB |
| **App4** | 4/22 | 0/7 | - | 2/3 | 2/2 | 1/11 | PE, IB |
| **App5** | 1/20 | 0/2 | 0/1 | - | - | 0/2 | PE, IB |
| **App6** | 0/4 | - | - | - | 0/1 | 1/1 | - |
| **App7** | 1 /2 | 0/2 | - | - | - | 0/1 | PE, IB |
| **App8** | 0/3 | 0/1 | - | 0/1 | 0/1 | 0/2 | PE, IB |
| **App9** | 0/4 | 1/ 2 | - | - | - | 0/1 | PE, IB |
| **App10** | 1/5 | 0/11 | - | 0/1 | 0/1 | 0/6 | PE, IB |
| **Σύνολο** | **7/10** εφαρμογές αιτούνται permissions που δεν απαιτούνται | **3/10** εφαρμογές δημιουργούν activities που δεν απαιτούνται | **Καμία** εφαρμογή δεν δημιουργεί CP που δεν απαιτούνται | **1/3** εφαρμογές δημιουργούν BR που δεν απαιτούνται | **2/5** εφαρμογές δημιουργούν υπηρεσίες που δεν απαιτούνται | **9/10** εφαρμογές δεν χρησιμοποι-ούν permissions για τον περιορισμό πρόσβασης στα δημόσια components τους | |

**(*) PE:** Priviledge escalation attacks **IB :** Intent Based Attacks Πίνακας 29

# *ΕΠΙΛΟΓΟΣ*

Η χρήση κινητών συσκευών αποτελεί κύρια δραστηριότητα της σημερινής κοινωνίας. Η ραγδαία αύξηση της δημοφιλίας τους έχει αποτελέσει το κίνητρο στους developers για την ανάπτυξη μεγάλου αριθμού εφαρμογών.

Αν και στην αρχή οι κινητές συσκευές θεωρούνταν αρκετά ασφαλείς (περιορισμένες δυνατότητες) η εμφάνιση των smartphones και των tablets με μεγάλη επεξεργαστική ισχύ και δυνατότητες, οδήγησε στην παράλληλη ανάπτυξη malware εφαρμογών.

Τα iOS και Android λειτουργικά συστήματα κατέχουν την συντριπτική πλειοψηφία της αγοράς κινητών συσκευών. Και οι δύο πλατφόρμες παρέχουν ικανοποιητικά μοντέλα ασφαλείας για την προστασία των δεδομένων τους, τα οποία οι developers πρέπει να λαμβάνουν υπόψη για την δημιουργία ασφαλών εφαρμογών καθώς η ελειπής σχεδίαση εφαρμογών και η τυχόν παράβλεψη των παραμέτρων ασφαλείας, μπορεί να οδηγήσει σε διαρροή κρίσιμων πληροφοριών του χρήστη.

Η σημαντικότερη παράμετρος ασφαλείας όμως, και στα δύο λειτουργικά συστήματα, παραμένει ο ίδιος ο χρήστης. Ο χρήστης είναι αυτός που αποφασίζει το είδος των εφαρμογών που εγκαθίστά στην συσκευή του και θα πρέπει να είναι σε θέση να λάβει τις σωστές αποφάσεις επι του θέματος.

# ΒΙΒΛΙΟΓΡΑΦΙΚΕΣ ΑΝΑΦΟΡΕΣ


[1] Abhishek Dubey, Anmol Misra, *Android Security Attacks and Defences*, CRC Press, 2013

[2] Jonathan Zdziarski, *Hacking and Securing iOS Applications*, O Reilly, 2011

[3] Erez Metula, *Android Mobile Application Hacking*, OWASP IL, 2012

[4] Jeff Six, *Application Security For The Android Platform* , O Reilly ,2012

[5] Rob Johnson , *Android security model ,* 2011

[6] Apple, *Apple iOS Security*, 2012

[7] Apple*, Apple iOS Tech Overview*, 2012

[8] MDSec, *Evaluating iOS Applications*, 2012

[9] Stefan Esser, *iOS5 An Exploitation Nightmare*, 2011

[10] Dino A. Dai Zovi *Apple iOS 4 Security Evaluation*, 2010

[11] NSA, *Security Configuration Recommendations for iOS5 devices,* 2012

[12] Android Manifest Permission class Specifications
http://developer.android.com/reference/android/Manifest.permission.html

[13] Android Settings System class Specifications
http://developer.android.com/reference/android/provider/Settings.System.html

[14] Adrienne Porter Felt, Erika Chin, Steve Hanna, Dawn Song, David Wagner, *Android Permissions Demystified* In Proceedings of the 18th ACM conference on Computer and communications security , 2011

[15] Michael Becher, Felix C. Freiling, Johannes Hoffmann, Thorsten Holz, Sebastian Uellenbeck, Christopher Wolf, *Revealing the Nuts and Bolts of the Security of Mobile Devices* In 2011 IEEE Symposium on Security and Privacy, 2011

[16] Lei Liu, Xinwen Zhang, Guanhua Yan, Songqing Chen, *Exploitation and Threat Analysis of Open Mobile Devices* In Proceedings of the 5th ACM/IEEE Symposium on Architectures for Networking and Communications Systems, 2009

[17] Michael Backes, Sebastian Gerling, Philipp von Styp-Rekowsky *A Local Cross-Site Scripting Attack against Android Phones*

[18] Erika Chin Adrienne Porter Felt Kate Greenwood David Wagner, *Analyzing Inter-Application Communication in Android*, in Proceedings of the International Conference on Mobile Systems, Applications, and Services, 2011

[19] Rejo Mathew (2012 )*Study of Privilege Escalation Attack on Android and its Countermeasures*, in International Journal of Engineering Science & Technology, 2012

[20] Kyoung Soo Han, Yeoreum Lee, Biao Jiang, Eul Gyu Im, *How to Violate Android's Permission System without Violating It*, in The Third International conference on Digital Information Processing and Comunications, 2013

[21] Lucas Davi, Alexandra Dmitrienko, Ahmad-Reza Sadeghi, Marcel Winandy , *Privilege Escalation Attacks on Android*, in 13th international conference on Information security, 2010

[22] Tongbo Luo, Hao Hao, Wenliang Du, Yifei Wang, and Heng Yin , *Attacks on WebView in the Android System*, in 27th ACSAC, 2011

[23] Cedric Halbronn, Jean Sigwald, *iPhone security model & vulnerabilities* In HITB SecConf, 2010

[24] Oliver Karow, Symantec, *Apple iOS Security in the Enterprise,* 2010

[25] Dionysus Blazakis, *The Apple Sandbox* In BlackHat EU 2011

[26] Nitesh Dhanjani, *New Age Application Attacks Against Apple's iOS* In BlackHat EU, 2011

[27] McAffe Threats Report First Quarter 2012

http://www.mcafee.com/us/resources/reports/rp-quarterly-threat-q1-2012.pdf

[28] McAffe Threats Report Second Quarter 2012

http://www.mcafee.com/us/resources/reports/rp-quarterly-threat-q2-2012.pdf

[29] McAffe Threats Report Third Quarter 2012

http://www.mcafee.com/us/resources/reports/rp-quarterly-threat-q3-2012.pdf

[30] McAffe Threats Report Fourth Quarter 2012

http://www.mcafee.com/us/resources/reports/rp-quarterly-threat-q4-2012.pdf

[31] J. Cheng, S. Wong, H. Yang, and S. Lu., Smartsiren: *Virus detection and alert for smartphones* in International Conference on Mobile Systems, Applications and Services (MobiSys), 2007

[32] M. Jakobsson and K. Johansson, *Retroactive Detection of Malware With Applications to Mobile Platforms*, in HotSec 10, Washington, 2010

[33] A. P. Fuchs et al., *SCanDroid: Automated Security Certification of Android Applications*, 2010

[34] W. Enck et al., *TaintDroid: An Information-flow Tracking System for Realtime Privacy Monitoring on Smartphones*, in USENIX Symposium on Operating Systems Design and Implementation (OSDI), 2010

[35] ITU Statistics http://www.itu.int/en/ITU-D/Statistics/Pages/default.aspx

[36] Leon Romanovsky, *Android Architecture For Beginners,* 2013

[37] Uncovering Android Master Key That Makes 99% of Devices Vulnerable http://bluebox.com/corporate-blog/bluebox-uncovers-android-master-key/, 2013