ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ – Π.Μ.Σ. ΑΣΦΑΛΕΙΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Βιομετρική Αυθεντικοποίηση σε Android

Διπλωματική Εργασία

Αγγελική Ζαπαλίδη
ΜΤΕ 1714
Πειραιάς, 21 Φεβρουαρίου 2020

Επιβλέπων Καθηγητής: Χ. Ξενάκης

Επιτελική Σύνοψη/Abstract

Τη σύγχρονη εποχή, οι τεχνολογίες εξελίσσονται με ταχύτατους ρυθμούς. Συχνά ασφάλεια και ευχρηστία δημιουργούν δίλημμα στους προγραμματιστές ως προς το πού να δώσουν βαρύτητα. Η ενσωμάτωση βιομετρικών τεχνολογιών στις προσωπικές συσκευές των χρηστών, με χαρακτηριστικό παράδειγμα τους αισθητήρες δακτυλικών αποτυπωμάτων σε έξυπνες κινητές συσκευές Android ή άλλου OS, φαίνεται να φέρνει την ισορροπία στις δύο αυτές έννοιες και να κάνει τόσο την εμπειρία του χρήστη όσο και την ασφάλεια των εφαρμογών προτεραιότητες που συμβαδίζουν. Ωστόσο, όπως και με κάθε νέα τεχνολογία ειδικά στον τομέα της ψηφιακής ασφάλειας και της προστασίας δεδομένων, ενέχουν κίνδυνοι. Συχνά υποτιμώνται τα ρίσκα που ενέχουν τέτοιες εφαρμογές ακόμα και κατά την ανάπτυξη τους, ή υπάρχει ελλειπής ενημέρωση των προγραμματιστών ως προς τις κατευθυντήριες γραμμές ασφαλείας ή των βέλτιστων πρακτικών. Υπάρχοντα πρότυπα όπως αυτό του FIDO παρέχουν έτοιμα εργαλεία, βιβλιοθήκες και API για να ενσωματώνονται από τους προγραμματιστές στον κώδικα της εφαρμογής τους. Παρόλα αυτά, συχνά η ενσωμάτωση αυτή γίνεται εσφαλμένα, μερικώς ή και καθόλου. Άλλοι προγραμματιστές μπορεί να επιλέγουν να κατασκευάσουν εκ νέου όλο τους το σύστημα, μαζί και της βιομετρικής αυθεντικοποίησης, αλλά δε λαμβάνουν πάντα υπόψη τις αρχές των προτύπων αυτών. Εν τέλει, μένει να διαπιστωθεί εάν εφαρμογές που διαχειρίζονται ευαίσθητα δεδομένα, όπως πχ. προσωπικά και τραπεζικά στοιχεία χρηστών, εφαρμόζουν τις βιομετρικές τεχνολογίες αυθεντικοποίησης με αποτελεσματικότητα στην ασφάλεια, χωρίς να είναι φαινομενική ή να θυσιάζεται χάριν ευχρηστίας. Η παρούσα διπλωματική εργασία μελετά τις τεχνολογίες αυτές και δοκιμάζει στην πράξη τα παραπάνω για να εξαχθούν συμπεράσματα σχετικά με την ασφάλεια που παρέχουν.

Λέξεις – Κλειδιά/Keywords

Βιομετρικές τεχνολογίες, βιομετρική αυθεντικοποίηση, τραπεζικές εφαρμογές, android OS, FIDO, δακτυλικό αποτύπωμα

Ευχαριστίες/Acknowledgements

Με την περάτωση της παρούσας διπλωματικής εργασίας θα ήθελα να εκφράσω τις ευχαριστίες μου στους καθηγητές του Προγράμματος Μεταπτυχιακών Σπουδών Ασφάλεια Ψηφιακών Συστημάτων, κους Χρήστο Ξενάκη και Χριστόφορο Νταντογιάν για την πολύτιμη βοήθεια και καθοδήγηση καθόλη τη διάρκεια της παρούσας μελέτης. Επίσης, θα ήθελα να ευχαριστήσω τους συναδέλφους Θωμά Μούρτο, Νικόλαο Καταχανά, Ιωάννα Ζαπαλίδη και Άννα Αγγελογιάννη για την υποστήριξη και την παροχή υλικού σε διάφορα στάδια εκπόνησης της εργασίας, καθώς και το εργαστήριο του Π.Μ.Σ. για το δανεισμό εξοπλισμού για την εκπόνηση των σχετικών πειραμάτων.

Πίνακας περιεχομένων

Κατάλογος Εικόνων

Κατάλογος Πινάκων

Κατάλογος Εξισώσεων

Κεφάλαιο 1ο – Εισαγωγικά

1.1 Εισαγωγή

Η παρούσα διπλωματική εργασία αποτελεί τη μελέτη και έρευνα επί των βιομετρικών τεχνολογιών σε τραπεζικές εφαρμογές σε Android smartphones, στα πλαίσια της Μεταπτυχιακής Διατριβής της φοιτήτριας του Τμήματος Μεταπτυχιακών Σπουδών «Ασφάλεια Ψηφιακών Συστημάτων», Ζαπαλίδη Αγγελικής, υπό την επίβλεψη του Καθηγητή κου Χρήστου Ξενάκη. Η εργασία αυτή εκπονήθηκε σε δύο τμήματα: 1) Την ανασκόπηση του βιβλιογραφικού υλικού και τη θεωρητική μελέτη. 2) Την εκτέλεση πειραμάτων και ανάλυση της λειτουργικότητας και αποδοτικότητας τραπεζικών εφαρμογών στον τομέα της βιομετρικής αυθεντικοποίησης.

1.2 Περιγραφή του υπό μελέτη προβλήματος

Οι έξυπνες κινητές συσκευές (smartphones) έχουν κατακλύσει τη σύγχρονη αγορά, και πλέον καθίστανται εξαιρετικά χρήσιμες για πολλές καθημερινές λειτουργίες των απλών χρηστών. Μία από τις πλέον αναπτυσσόμενες λειτουργίες που εκτελούνται με ευκολία και ταχύτητα από τις προσωπικές κινητές συσκευές των χρηστών είναι οι πάσης φύσεως ηλεκτρονικές συναλλαγές. Τη σύγχρονη εποχή, οι χρήστες μπορούν να πραγματοποιήσουν πληρωμές ανά πάσα στιγμή και σε οποιοδήποτε μέρος, απλώς με τη συσκευή που διαθέτουν πάνω τους. Η άνεση και η ευχρηστία των συναλλαγών αυτών ωστόσο δεν πρέπει να θέτει σε δεύτερη μοίρα την ασφάλεια τους.

Η διαρκής αύξηση της δημοτικότητας των smartphones και η χρήση τους για τέτοιου είδους διαδικασίες (και όχι μόνο) έχει ως φυσικό συνεπακόλουθο την ταυτόχρονη εξέλιξη των τεχνολογιών που διαθέτουν, ώστε να γίνεται η εμπειρία του χρήστη καλύτερη, ταχύτερη, αποδοτικότερη και ασφαλέστερη. Χαρακτηριστικό παράδειγμα τέτοιων τεχνολογιών είναι η βιομετρική. Πολλά μοντέλα συσκευών φέρουν σαρωτές βιομετρικών χαρακτηριστικών και πολλούς αισθητήρες, με χαρακτηριστικό παράδειγμα το σαρωτή δακτυλικών αποτυπωμάτων. Τα βιομετρικά χαρακτηριστικά χρησιμοποιούνται για τον προσδιορισμό της ταυτότητας ενός χρήστη και την εξουσιοδότηση πρόσβασης του σε κάποια λειτουργία ή αρχείο. Για το λόγο αυτό χρησιμοποιούνται για ξεκλείδωμα συσκευών. Επιπλέον, πολλές εφαρμογές αξιοποιούν τις τεχνολογίες αυτές για την ασφαλή αυθεντικοποίηση των χρηστών, μεταξύ αυτών και πολλές τραπεζικές εφαρμογές.

Σκοπός της εργασίας αυτής είναι η μελέτη των βιομετρικών τεχνολογιών σε έξυπνες συσκευές Android και στη συνέχεια η μελέτη και αξιολόγηση της λειτουργίας και του τρόπου αυθεντικοποίησης που χρησιμοποιούν, σε σχέση με αναγνωρισμένα πρότυπα και το FIDO.

1.3 Βασικοί Ορισμοί

- **Βιομετρικές Τεχνολογίες:** Βιομετρική είναι η επιστήμη εντοπισμού και αναγνώρισης ανθρώπινων χαρακτηριστικών, μέσω μέτρησης και ανάλυσης βιολογικών δεδομένων με χρήση διαφόρων ηλεκτρονικών τεχνολογιών.

- **Smartphone (έξυπνη κινητή συσκευή):** Κινητό τηλέφωνο βασισμένο σε λειτουργικό σύστημα κινητής τηλεφωνίας με περισσότερο προηγμένη υπολογιστική ισχύ και συνδεσιμότητα σε σχέση με ένα συμβατικό κινητό τηλέφωνο.
- **Αυθεντικοποίηση:** Αυθεντικοποίηση χρήστη (user authentication) ονομάζεται η διαδικασία που αποσκοπεί στην επιβεβαίωση της ταυτότητας ενός χρήστη.
- **Ιδιωτικότητα:** Ιδιωτικότητα είναι το απαράγραπτο δικαίωμα κάθε ατόμου να ασκεί έλεγχο στον τρόπο με τον οποίο οι προσωπικές του πληροφορίες τηρούνται, υποβάλλονται σε επεξεργασία ή διανέμονται και χρησιμοποιούνται από οποιαδήποτε άλλη οντότητα.
- **Ακεραιότητα:** Ακεραιότητα (integrity) είναι η ιδιότητα της ασφάλειας πληροφορίας σύμφωνα με την οποία, είναι δυνατή η τροποποίησή της μόνο σε εξουσιοδοτημένες οντότητες.
- **Εμπιστευτικότητα:** Εμπιστευτικότητα (confidentiality) είναι η ιδιότητα της ασφάλειας πληροφορίας σύμφωνα με την οποία, είναι δυνατή η ανάγνωσή της μόνο σε εξουσιοδοτημένες οντότητες.
- **Διαθεσιμότητα**: Διαθεσιμότητα (availability) είναι η ιδιότητα της ασφάλειας πληροφορίας σύμφωνα με την οποία, είναι διαθέσιμη για χρήση μόνο σε εξουσιοδοτημένες οντότητες. [(4)]
- **FIDO:** Η FIDO ("Fast IDentity Online") Alliance είναι μια βιομηχανική κοινοπραξία ιδρυθείσα το 2013 και ασχολείται με τη διαλειτουργικότητα μεταξύ συσκευών με ισχυρή αυθεντικοποίηση και τα προβλήματα που αντιμετωπίζουν οι χρήστες στη δημιουργία και αποστήθιση usernames και passwords. [(5)]

1.4 Δομή της Εργασίας

Στο 1ο κεφάλαιο της εργασίας, γίνεται μια εισαγωγή στο γενικότερο θέμα το οποίο μελετήθηκε κατά την εκπόνηση της εργασίας αυτής, και διασαφηνίζονται συνοπτικά ορισμένες έννοιες που θα αναλυθούν περισσότερο σε επόμενες ενότητες. Επιπλέον, περιγράφονται τα περιεχόμενα, τα παραδοτέα αρχεία και η δομή της εργασίας.

Στο 2ο κεφάλαιο γίνεται η ανασκόπηση της βιβλιογραφίας που μελετήθηκε και η μελέτη της θεωρίας στην οποία βασίστηκε η εργασία αυτή. Παρουσιάζεται η γνώση που αποκομίστηκε από τη μελέτη αυτή, και αναλύονται με περισσότερη σαφήνεια οι διάφορες έννοιες και ορισμοί.

Στο 3ο κεφάλαιο περιγράφονται τα διάφορα πειράματα που εκτελέστηκαν επί των υπό μελέτη εφαρμογών, τραπεζικών και μη, για την αξιολόγηση της λειτουργίας, του σχεδιασμού και της υλοποίησης των βιομετρικών τεχνολογιών στα πλαίσια της χρήσης τους.

Στη συνέχεια, στο 4ο κεφάλαιο παρατίθενται τα συμπεράσματα που προέκυψαν από τη μελέτη αυτή, οι ιδέες για περαιτέρω βελτίωση των βιομετρικών τεχνολογιών στα πλαίσια των εφαρμογών κινητών συσκευών με λειτουργικό σύστημα Android, καθώς και οι κίνδυνοι κακοχρησίας των τεχνολογιών αυτών στα περιβάλλοντα αυτά.

Τέλος, ακολουθεί η παράθεση των βιβλιογραφικών πηγών που αξιοποιήθηκαν.

Κεφάλαιο 2ο – Θεωρητική Μελέτη

2.1 Επισκόπηση του χώρου μελέτης

Οι **βιομετρικές τεχνολογίες** είναι η επιστήμη της ανίχνευσης και της αναγνώρισης των ανθρωπίνων χαρακτηριστικών, με τη μέτρηση και την ανάλυση βιολογικών δεδομένων μέσω της χρήσης διάφορων ηλεκτρονικών τεχνολογιών.

Figure 1 - Οι βιομετρικές τεχνολογίες δεν αποτελούν επιστημονική φαντασία

Πηγή: https://epicecommercetools.com/

Η χρήση των τεχνολογιών αυτών προσφέρει έναν ταχύτατο και αποτελεσματικό τρόπο ταυτοποίησης ενός χρήστη, ενώ παρέχουν προστασία της ταυτότητάς του, της ιδιωτικότητας και της προσωπικής του ασφάλειας, μειώνοντας τον κίνδυνο υποκλοπής ταυτότητας (identity theft).

Μέχρι πρόσφατα, ο όρος «βιομετρικές τεχνολογίες» έφερνε στο μυαλό του κοινού εικόνες επιστημονικής φαντασίας, με φουτουριστικές τεχνολογίες και εντυπωσιακές καινοτομίες. Ωστόσο, πολλά από αυτά τα sci-fi gadgets όχι μόνο υπάρχουν και βρίσκουν εφαρμογή στην εποχή μας, αλλά έχουν πλέον παρεισφρήσει σε μεγάλο βαθμό στην καθημερινή μας ζωή και τα χρησιμοποιούμε συνεχώς.

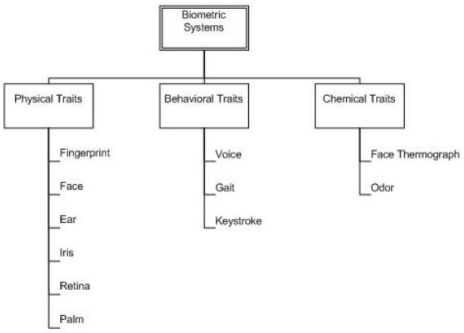


Figure 2 - Είδη Βιομετρικών Χαρακτηριστικών [9]

Η εφαρμογή τους και η πρόσβαση των end users σε τέτοιες τεχνολογίες ξεκίνησε σε κτίρια με μεγάλη ανάγκη ασφαλείας, όπου χρησιμοποιούνταν βιομετρικά **δακτυλικών αποτυπωμάτων** για να βεβαιώνεται πως μόνο εγκεκριμένο προσωπικό έχει πρόσβαση σε ασφαλείς περιοχές. Σύντομα άρχισαν να απαντώνται σε αεροδρόμια τεχνολογίες **αναγνώρισης της ίριδας του ματιού**, όπου οι επιβάτες αυθεντικοποιούνταν σε δευτερόλεπτα, αντί να περιμένουν σε ουρά για τον έλεγχο του διαβατηρίου τους. Εκπαιδευτικοί φορείς υιοθέτησαν επίσης τις βιομετρικές τεχνολογίες, για να αυτοματοποιήσουν αποτελεσματικά πολλές λειτουργίες τους. Έτσι, με χρήση βιομετρικών, εξουσιοδοτημένο προσωπικό, μαθητές και φοιτητές δηλώνουν παρουσία σε μάθημα, δανείζονται από τη βιβλιοθήκη ή ψωνίζουν από το κυλικείο.

Τα βιομετρικά συστήματα χρησιμοποιούν μια πληθώρα διαφορετικών χαρακτηριστικών που αποκομίζουν από ένα άτομο, για να αποδείξουν την ταυτότητά του. Τέτοια χαρακτηριστικά (traits, modalities) μπορεί να είναι **σωματικά** ή **συμπεριφορικά**. Παρέχουν ευκολία, ασφάλεια και βοηθούν στην αποτροπή της πλαστοπροσωπίας.

- **Σωματικά:** δακτυλικό αποτύπωμα, μορφολογία προσώπου, ίρις ματιού, φλέβες, DNA
- **Συμπεριφορικά:** υπογραφή, ήχος φωνής, βηματισμός, ταχύτητα πληκτρολόγησης, κίνηση ποντικιού

Ο συχνότερος τρόπος αυθεντικοποίησης στις μέρες μας είναι η χρήση καρτών (πχ. τραπέζης ή μέλους) και κωδικών (συνθηματικών ή τετραψήφιων PIN). Ωστόσο, οι πρώτες μπορεί να χαθούν, να διαμοιραστούν ή να κλαπούν, ενώ οι δεύτεροι να ξεχαστούν ή να διαδοθούν μεταξύ ατόμων. Τη λύση στα παραπάνω προβλήματα έρχονται να δώσουν οι βιομετρικές τεχνολογίες αυθεντικοποίησης.

Υπάρχουν συστήματα βιομετρικών που αξιοποιούν μόνο ένα από τα παραπάνω χαρακτηριστικά και συστήματα που αξιοποιούν περισσότερα (multimodal) αυξάνοντας την ακρίβεια του αποτελέσματος. Η αποτελεσματικότητα καθενός εξαρτάται από τις ανάγκες του οργανισμού που το εφαρμόζει. Συνεπώς, δεν υπάρχει ένα μόνο σύστημα βιομετρικών που να θεωρείται ιδανικό.

Οι σημερινές μέθοδοι ταυτοποίησης είναι αναποτελεσματικές, ιδιαιτέρως με την παγκόσμια αγορά στα χέρια μας. Ως τελικοί χρήστες, πρέπει να έχουμε εμπιστοσύνη σε ένα σύστημα αναγνώρισης που να παρέχει ασφάλεια και ιδιωτικότητα στις προσωπικές μας πληροφορίες και η βιομετρική αυθεντικοποίηση είναι η καταλληλότερη λύση για να επιτευχθεί αυτό.

2.2 Ιστορική Αναδρομή

Ο όρος «βιομετρική» έχει τις ρίζες του στις Ελληνικές λέξεις «βίος» δηλαδή ζωή, και «μετρική». Ως έννοια υφίσταται εδώ και χιλιετίες, ωστόσο η τεχνολογική συσχέτιση βιομετρικής και ηλεκτρονικής ταυτοποίησης πρωτοεμφανίστηκε το 1981, σε άρθρο των New York Times (Pollack, 1981). Μέχρι τότε, υπήρχαν πολλές μη υπολογιστικές εφαρμογές της επιστήμης αυτής σε μεθόδους ταυτοποίησης. Την παλαιότερη μορφή βιομετρικής ταυτοποίησης αποτελεί φυσικά η αναγνώριση προσώπου και η εξ όψεως ταυτοποίηση ενός ατόμου από κάποιο άλλο άτομο! Ένας πιο απτός τρόπος ταυτοποίησης εντοπίζεται και σε προϊστορικές σπηλαιογραφίες, όπου οι άνθρωποι «υπέγραφαν» αποτυπώνοντας τις παλάμες τους, ενώ κατά την αρχαιότητα συναντάται στους Βαβυλώνιους, οι οποίοι χρησιμοποιούσαν τα δακτυλικά τους αποτυπώματα σε πινακίδες από πηλό ως μέθοδο ταυτοποίησης και υπογραφής σε εμπορικές συναλλαγές.[7] Παρόμοια ευρήματα συναντώνται στην Αρχαία Ελλάδα, στη Ρώμη, στην Αρχαία Κίνα και στην Αίγυπτο.


Figure 3 - J.E. Purkinje

Η πρώτη σύγχρονη μελέτη βιομετρικής έγινε το 19ο αιώνα, όταν ο καθηγητής ανατομίας Johannes Evangelista Purkinje του Πανεπιστημίου του Breslau πρότεινε το 1823 ένα σύστημα ταξινόμησης δακτυλικών αποτυπωμάτων, βάσει της θεωρίας του ότι τα αποτυπώματα αυτά παραμένουν σταθερά και ίδια σε κάθε άτομο από τη γέννηση έως το θάνατό του. Η έρευνα αυτή είναι ήσσονος σημασίας, καθώς αποτελεί την πρώτη μελέτη χρήσης δακτυλικών αποτυπωμάτων ως μεθόδου ταυτοποίησης ενός ατόμου.

Figure 4 - Αποτύπωμα εργαζομένου στην Ινδία, 19ος αι.

Ωστόσο, το πρώτο σύστημα που έθεσε σε εφαρμογή τη θεωρία του Purkinje και χρησιμοποίησε καταγραφές αποτυπωμάτων χεριών αναπτύχθηκε στην Ινδία το 1858, από τον Sir William Herschel, στα οπισθόφυλλα των συμβολαίων των εργαζομένων. Έτσι τους ταυτοποιούσε, ώστε να μη μπορεί κανείς με πλαστοπροσωπία να πληρωθεί ως εργαζόμενος. Στη συνέχεια, ο Herschel θεώρησε επαρκή μορφή ταυτοποίησης τα αποτυπώματα δύο δακτύλων του δεξιού χεριού: του δείκτη και του μέσου.

Το 1870, η μέθοδος αναγνώρισης που ονομάζεται **ανθρωπομετρία**, η οποία καταγράφει μετρήσεις του σώματος, χρησιμοποιήθηκε για τον εντοπισμό κατά συρροήν εγκληματιών. Ωστόσο, μελέτες διαπίστωσαν ότι οι άνθρωποι συχνά μοιράζονταν τις ίδιες φυσικές μετρήσεις, έτσι η χρήση αυτής της τεχνικής μειώθηκε. Στα τέλη του 19ου αιώνα, ο Sir Francis Galton εντόπισε ένα σύστημα απόλυτης ταξινόμησης, χαρτογραφώντας και τα δέκα δάχτυλα, δείχνοντας ότι η πιθανότητα να είναι ταυτόσημα δύο δακτυλικά αποτυπώματα ήταν 1 στα 64 δισεκατομμύρια. Ο Galton εντόπισε τα χαρακτηριστικά των δακτυλικών αποτυπωμάτων και τον τρόπο διάκρισής τους, αναλύοντας τα μοναδικά μοτίβα που σχηματίζονται από το δέρμα στις άκρες των δακτύλων μας. Πρόκειται για την ίδια τεχνική που χρησιμοποιείται σήμερα από ειδικούς εγκληματολογίας.

Figure 5 - Η εξέλιξη της συλλογής αποτυπωμάτων

Την ίδια περίοδο, η αστυνομία της Ινδίας σε συνεργασία με τον Sir Edward Richard Henry, επίτροπο της βρετανικής αστυνομίας, θέτει σε εφαρμογή τη χρήση δακτυλικών αποτυπωμάτων ως μέσο αναγνώρισης εγκληματιών. Το 1901, ο Sir Henry ίδρυσε τα πρώτα αρχεία των βρετανικών δακτυλικών αποτυπωμάτων και ανέπτυξε επίσης το σύστημα ταξινόμησης Henry, το οποίο χρησιμοποιήθηκε για την ταξινόμηση και κατηγοριοποίηση δακτυλικών αποτυπωμάτων. Αξίζει να σημειωθεί πως, μέχρι τότε, στην εγκληματολογική αναγνώριση χρησιμοποιούνταν ως μέσο αναγνώρισης και ταυτοποίησης το αυτί, που αποτελεί επίσης διακριτικό χαρακτηριστικό όχι όμως με τον ίδιο βαθμό λεπτομέρειας.[15] Το 1903, το Κρατικό Σύστημα Φυλακών της Νέας Υόρκης άρχισε

να χρησιμοποιεί δακτυλικά αποτυπώματα στην παρακολούθηση των κρατουμένων. Ένα χρόνο αργότερα ακολούθησε το ομοσπονδιακό σωφρονιστικό σώμα Leavenworth στο Κάνσας. Κατά τα επόμενα 25 χρόνια, όλο και περισσότεροι φορείς επιβολής του νόμου υιοθέτησαν τη χρήση δακτυλικών αποτυπωμάτων ως μέσο αναγνώρισης εγκληματιών.

Ορισμένες από τις πρώτες εργασίες για την αναγνώριση των προσώπων από μηχανές εντοπίζονται στη δεκαετία του 1960, όπου ο Woody Bledsoe ανέπτυξε την «αναγνώριση προσώπου», που σήμερα αναφέρεται ως τεχνητή νοημοσύνη. Η εποχή της ηλεκτρονικής βιομετρικής ξεκινά το 1974, όταν το Πανεπιστήμιο της Georgia ξεκίνησε το βιομετρικό σύστημα γεωμετρίας χεριών στους κοιτώνες και άρχισε επίσης να εφαρμόζεται η αναγνώριση υπογραφών στις ΗΠΑ και στο Ηνωμένο Βασίλειο. Από τότε, η βιομετρική ταυτοποίηση έχει κάνει ένα τεράστιο άλμα προς το μέλλον, έχοντας καταστεί εγγενής για όλες τις τεχνολογίες ασφάλειας, από κυβερνητικούς και στρατιωτικούς φορείς έως ιδιωτικούς οργανισμούς, στην υγειονομική περίθαλψη και στον τομέα των τηλεπικοινωνιών. Αποτελεί μια από τις καλύτερες λύσεις αυθεντικοποίησης για την αύξηση της προστασίας χρηστών στο κινητό περιβάλλον και υπάρχουν πολλά παραδείγματα ενσωμάτωσης της τεχνολογίας στην ανάπτυξη εφαρμογών κινητής ψηφοφορίας, τραπεζικών και ηλεκτρονικών συναλλαγών. Με τη ραγδαία αύξηση στην παγκόσμια ανταλλαγή πληροφοριών, η ανάγκη καλύτερης προστασίας της ταυτότητάς μας και των συστημάτων ασφαλείας που μειώνουν τις κλοπές ταυτότητας αποτελούν πρωτεύουσα ανάγκη. [1][2][6]

Figure 6 - Σύγχρονη Βιομετρική

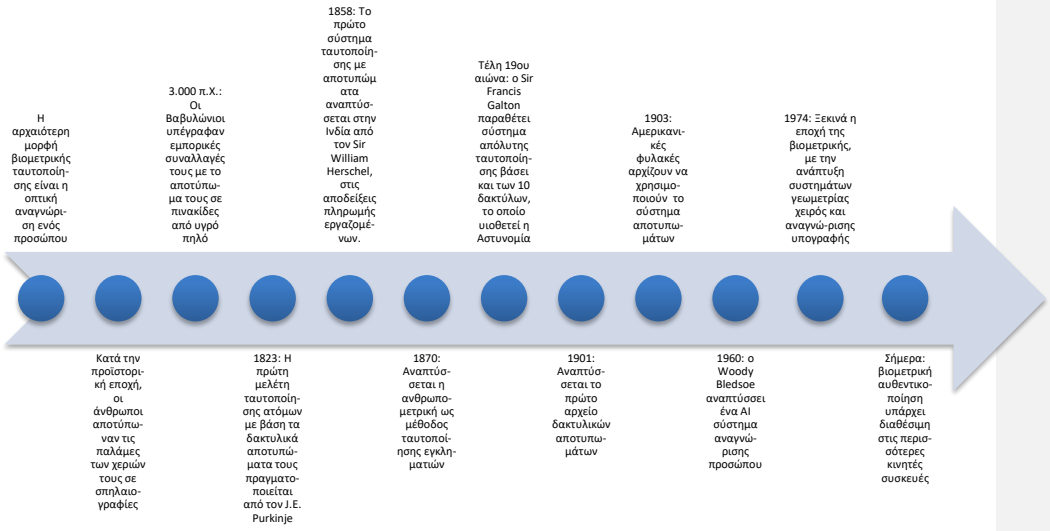


Figure 7 - Ιστορική Αναδρομή

2.3 Ταυτοποίηση και Αυθεντικοποίηση

Όπως στα περισσότερα πληροφοριακά συστήματα ασφαλείας, έτσι και στα βιομετρικά καθ' όλο τον κύκλο ζωής τους λαμβάνονται υπόψη τα πλεονεκτήματα της χρήσης τους, οι κίνδυνοι, το κόστος υλοποίησης και η προστασία των στοιχειωδών αρχών ασφαλείας (εμπιστευτικότητα, ακεραιότητα, διαθεσιμότητα – CIA). Η βαρύτητα που δίνεται σε καθέναν από τους παραπάνω παράγοντες, εξαρτάται σε μεγάλο βαθμό από τον τύπο του συστήματος στο οποίο αναφερόμαστε. Η βασική διάκριση συστημάτων βιομετρικών είναι μεταξύ συστημάτων ταυτοποίησης και συστημάτων αυθεντικοποίησης. Κατά Nanavati et. al. (2002) [1], ο πιο εύκολος τρόπος διαχωρισμού των δύο αυτών ειδών συστημάτων γίνεται με τις εξής ερωτήσεις:

Αυθεντικοποίηση (authentication/verification): «Είμαι αυτός που ισχυρίζομαι ότι είμαι;»

Τα συστήματα αυθεντικοποίησης έχουν ως προϋπόθεση ότι το άτομο έχει αρχικά ισχυριστεί ότι έχει μια συγκεκριμένη ταυτότητα την οποία καλείται να επαληθεύσει, πχ. με χρήση έξυπνων καρτών, επίδειξη ταυτότητας ή πληκτρολόγηση ενός username. Στην περίπτωση της βιομετρικής αυθεντικοποίησης, το σύστημα συγκρίνει το δείγμα βιομετρικού υλικού του χρήστη με τα δείγματα αποθηκευμένα στη βάση δεδομένων του, για να επιτύχει μια ταύτιση (one-to-one match). Πρόκειται για μια διαδικασία που δεν απαιτεί μεγάλη υπολογιστική ισχύ και συνήθως η ταύτιση επιτυγχάνεται σε λιγότερο από ένα δευτερόλεπτο. Ωστόσο τα συστήματα αυτά δεν έχουν τη δυνατότητα αναγνώρισης διπλοτύπων σε μια βάση δεδομένων, γεγονός που ενέχει κινδύνους ασφαλείας, ειδικά αν δεν έχει γίνει απολύτως σωστή εφαρμογή και χειρισμός του συστήματος.

Figure 8 - Verification [11]

Ταυτοποίηση (identification): «Ποιος είμαι;»

Στα συστήματα αυτά δεν απαιτείται από το χρήστη να ισχυριστεί ότι έχει μια ταυτότητα. Όταν το άτομο παρέχει το βιομετρικό υλικό του στον αισθητήρα του συστήματος, αυτό το συγκρίνει με εκατομμύρια άλλα δείγματα αποθηκευμένα στη βάση του, για να βρει άλλο δείγμα να ταιριάζει (one-to-many matching). Συνεπώς, τα συστήματα αυτά απαιτούν τεράστια υπολογιστική ισχύ και χωρητικότητα βάσης δεδομένων και η απόδοσή τους σε χρόνο και ακρίβεια εξαρτάται από το πλήθος χρηστών των οποίων τα δείγματα αποθηκεύει. [2] [3]

Συνοψίζοντας, η επαλήθευση της ταυτότητας του χρήστη, και άρα η απάντηση στην ερώτηση «είμαι αυτός που ισχυρίζομαι πως είμαι;» δίνεται από τη σύγκριση και ταύτιση του βιομετρικού δείγματος του χρήστη με αυτό ενός ατόμου του οποίου τα στοιχεία βρίσκονται στο σύστημα. Από την άλλη, τα

συστήματα ταυτοποίησης απαντούν στην ερώτηση «ποιος είμαι;» συγκρίνοντας το βιομετρικό χαρακτηριστικό του χρήστη με δείγματα στη βάση που φιλοξενεί το σύστημα.

Figure 9 - 1:1 - Verification

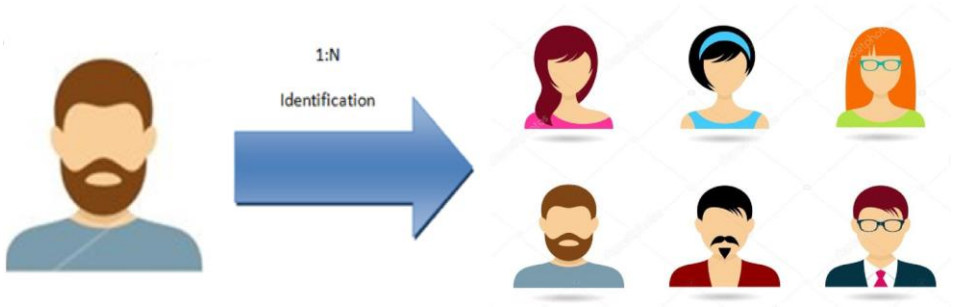


Figure 10 - 1:N – Identification

Η ταυτοποίηση και η επαλήθευση ταυτότητας μπορούν να πραγματοποιηθούν με πολλούς μηχανισμούς. Αυτοί οι μηχανισμοί εμπίπτουν σε τρεις κατηγορίες: παράγοντες γνώσης, παράγοντες κατοχής και βιομετρικούς παράγοντες. Οι παράγοντες **γνώσης** ζητούν κάτι που γνωρίζει ο χρήστης (όπως ένα PIN ή έναν κωδικό πρόσβασης), οι παράγοντες **κατοχής** ζητούν κάτι που κατέχει (όπως key generator ή κλειδί ασφαλείας) και οι **βιομετρικοί** παράγοντες ζητούν κάτι που είναι (όπως δακτυλικό αποτύπωμα, ίριδα ή πρόσωπο).

2.4 Είδη Βιομετρικών Χαρακτηριστικών

Όπως αναφέραμε, τα βιομετρικά χαρακτηριστικά είναι βιολογικά χαρακτηριστικά του χρήστη και διακρίνονται σε φυσιολογικά, συμπεριφορικά και βιοχημικά.

Τα **φυσιολογικά** βιομετρικά αφορούν στην απευθείας μέτρηση χαρακτηριστικών του ανθρωπίνου σώματος. Τέτοια χαρακτηριστικά είναι:

- Πρόσωπο: Η αναγνώριση προσώπου αποτελεί την αρχαιότερη μορφή ταυτοποίησης, έχει πλέον αυτοματοποιηθεί με τον εντοπισμό προσώπων σε οπτικό υλικό και την ανάλυση της

γεωμετρίας του προσώπου βάσει 80 σημείων για τη δημιουργία ενός μοναδικού face-print. Ωστόσο, διαφορετικές οπτικές γωνίες και κακός φωτισμός δυσκολεύουν τη λειτουργία αυτή.

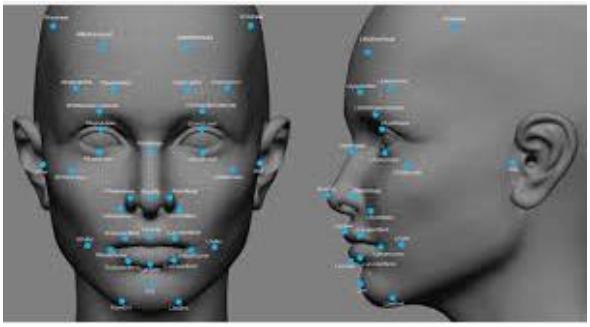


Figure 11 - Ψηφιακή Αναγνώριση Προσώπου[11]

- Θερμογραφική Προσώπου: Με υπέρυθρες, εντοπίζει τη θερμότητα και τις φλέβες κάτω από το δέρμα. Επιβεβαιώνει επίσης ότι το άτομο είναι ζωντανό.
- Εγκεφαλικά Κύματα: Εξελισσόμενη τεχνολογία βάσει της θεωρίας ότι τα σήματα του εγκεφάλου κάθε ατόμου παράγουν ένα μοναδικό σύνολο χαρακτηριστικών του.
- Δακτυλικό Αποτύπωμα: Χρησιμοποιείται για αιώνες και βασίζεται στα μοτίβα που παράγει το δέρμα στα δάχτυλα, για λόγους τριβής (friction ridge). Οικονομικό και αξιόπιστο, καθώς τα αποτυπώματα δεν αλλάζουν κατά τη διάρκεια της ζωής του ατόμου.
- Φλέβες δακτύλου: Οι φλέβες παράγουν ένα μοτίβο μοναδικό για κάθε άτομο.
- Υπέρυθρη εικόνα δακτύλου: Αναλύει το μοναδικό για τον καθένα θερμικό αποτύπωμα.
- Ίριδα ματιού: Η υφή της ίριδας περιέχει διακριτά μοτίβα που χρησιμοποιούνται για ταυτοποίηση. Η υφή αυτή δε μπορεί να αλλοιωθεί χειρουργικά και είναι μοναδική για κάθε άτομο, ακόμα και μεταξύ ομοζυγωτικών διδύμων είναι διαφορετική.[19]
- Αμφιβληστροειδής Αδένας: Και πάλι, τα μοτίβα των αιμοφόρων αγγείων στον αμφιβληστροειδή είναι πολύπλοκα και μοναδικά για κάθε άτομο. Ο αμφιβληστροειδής επί του παρόντος είναι αδύνατο να ξεγελάσει την αναγνώριση αυτή. [19]
- Εξωτερικό αυτί: Για αιώνες αποτελούσε διακριτικό χαρακτηριστικό, καθώς τα μοτίβα του εξωτερικού ωτός είναι μοναδικά για κάθε άτομο. Σήμερα, δε χρησιμοποιείται ευρέως.
- Σώμα: Η ανθρωπομετρία αποτελεί φυσιολογικό χαρακτηριστικό, προφανώς όμως οι διαστάσεις κάθε σώματος δεν είναι μοναδικές στο σύνολο του ανθρώπινου πληθυσμού.
- Φλέβες: Όπως στα χέρια, έτσι και σε όλο το σώμα παράγουν μοναδικά μοτίβα.
- Γεωμετρία και κινήσεις χειρός: Χρησιμοποιείται στο 50% των συστημάτων ελέγχου προσπέλασης, βασίζεται στο μήκος, το πλάτος και το άνοιγμα των δακτύλων στο χώρο, ενώ και οι κινήσεις άρχισαν πρόσφατα να αναγνωρίζονται ως χαρακτηριστικά κάθε ατόμου με βάση την ανατομία των χεριών.

- DNA: Το γενετικό υλικό, που αποτελεί ίσως την ακριβέστερη μορφή ταυτοποίησης, αφού μένει αναλλοίωτο καθ' όλη τη ζωή του ατόμου, ακόμα και μετά θάνατον. Πιο ακριβή και χρονοβόρα μορφή αναγνώρισης, που επίσης δε μπορεί να διακρίνει μεταξύ ομοζυγωτικών διδύμων. Έχει ποσοστό αναγνώρισης περίπου ίσο ή και λίγο κατώτερο από αυτό της αναγνώρισης με δακτυλικό αποτύπωμα!

Τα **συμπεριφορικά** βιομετρικά αφορούν στην έμμεση μέτρηση ανθρωπίνων χαρακτηριστικών, καθώς βασίζονται σε μετρήσεις και δεδομένα που προέρχονται από κάποια δράση και τη σύσταση του ανθρώπινου σώματος. Τέτοια χαρακτηριστικά αποτελούν:

- Φωνή: Στην πραγματικότητα αποτελεί συνδυασμό φυσιολογικών και συμπεριφορικών χαρακτηριστικών, καθώς εξαρτάται από τη φυσιολογία των φωνητικών χορδών, τη ρινική και στοματική κοιλότητα κ.ά. τα οποία παράγουν μοναδικό ήχο κατά την ομιλία του ατόμου και μένουν αναλλοίωτα, αλλά και από συμπεριφορικούς παράγοντες όπως η ηλικία, η υγεία, η ψυχική κατάσταση κλπ. Θεωρείται ηθικά αποδεκτή μέθοδος αλλά είναι ευάλωτη σε απάτες.
- Βάδισμα: Ο τρόπος που περπατά ένα άτομο. Πολύπλοκο βιολογικό χαρακτηριστικό. Δεν είναι μοναδικό χαρακτηριστικό, και αλλάζει με την πάροδο του χρόνου, αλλά αρκεί για ταυτοποίηση. Χρησιμοποιείται με βάση βίντεο και αισθητήρες κινητών συσκευών.
- Πληκτρολόγηση (keystroke): Βασίζεται στον ιδιαίτερο τρόπο που πληκτρολογεί κάθε άτομο. Ανεπαρκής μέθοδος αυθεντικοποίησης. Βασίζεται στο χρόνο μεταξύ πίεσης πλήκτρων και την ταχύτητα πληκτρολόγησης.
- Υπογραφή: Είναι γνωστό χαρακτηριστικό κάθε ατόμου ο τρόπος με τον οποίο υπογράφει. Βασίζεται στο μοτίβο της υπογραφής και, με νέες τεχνολογίες, την ταχύτητα και πίεση του χεριού. Είναι επιρρεπής σε απάτες ακόμα και με τις νέες αυτές τεχνολογίες.

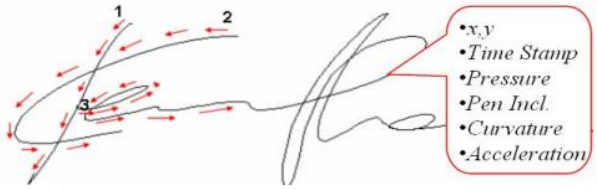


Figure 12 - Αναγνώριση Υπογραφής[11]

Τέλος, τα **βιοχημικά** χαρακτηριστικά για την αναγνώριση ενός ατόμου αφορούν στη μελέτη της βιοχημείας του ανθρώπινου σώματος, όπως:

- Οσμή σώματος: Αναπτυσσόμενη τεχνολογία, άρωμα από μέρη του σώματος όπως τα χέρια συλλέγεται και μετατρέπεται μέσω αλγορίθμου σε προφίλ δεδομένων για μελλοντική αναγνώριση.

- Αλατότητα σώματος: Επίσης αναπτυσσόμενη, μέσω μικρών ηλεκτρικών παλμών μέσα στο σώμα προσδιορίζονται τα επίπεδα άλατος και δημιουργούνται αντίστοιχα προφίλ.

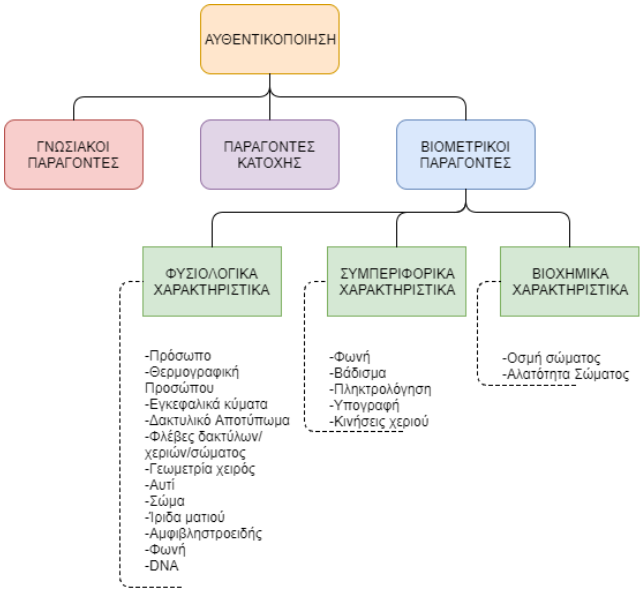


Figure 13 - Είδη και Κατηγορίες Βιομετρικών Χαρακτηριστικών

2.5 Βιομετρικά στη σύγχρονη αγορά

Οι βιομετρικές εφαρμογές μπορούν να ταξινομηθούν σε τρεις κύριες κατηγορίες:

- **Εμπορικές εφαρμογές**: πχ. έλεγχος φυσικής και λογικής πρόσβασης, και η ηλεκτρονική τραπεζική (e-banking).
- **Κυβερνητικές εφαρμογές:** πχ. τα εθνικά δελτία ταυτότητας, οι πληρωμές κοινωνικής ασφάλισης και η ασφάλεια των συνόρων.
- **Ιατροδικαστικές εφαρμογές:** πχ. στην επιβολή του νόμου, η αναγνώριση του εγκλήματος και η αναγνώριση του θύματος.

Στην παρούσα εργασία θα επικεντρωθούμε στην πρώτη κατηγορία εφαρμογών, και ειδικότερα στη μελέτη *βιομετρικών σε τραπεζικές εφαρμογές*. Οι εμπορικές εφαρμογές με βιομετρικά όχι μόνο εξασφαλίζουν και βελτιώνουν τις υπάρχουσες χρηματοπιστωτικές υπηρεσίες, αλλά επιτρέπουν επίσης την παροχή ασφαλών απομακρυσμένων υπηρεσιών που δεν θα ήταν διαφορετικά εφικτές χωρίς τη χρήση βιομετρικών τεχνολογιών.

Ασφαλώς, η χρήση των βιομετρικών δεν περιορίζεται μόνο σε αυτόν τον τομέα.

Τα **συστήματα ελέγχου πρόσβασης** χρησιμοποιούν βιομετρικά στοιχεία για τον προσδιορισμό ή την επαλήθευση της ταυτότητας των ατόμων που εισέρχονται ή εξέρχονται από μια περιοχή, συνήθως ένα δωμάτιο ή ένα κτίριο. Πρόκειται για βιομετρικά που χρησιμοποιούνται για να συμπληρώσουν ή να αντικαταστήσουν τους απαρχαιωμένους μηχανισμούς όπως κλειδιά, μάρκες, κάρτες και πάσο. Αυτά τα συστήματα συνήθως υιοθετούνται για να επιτρέπουν την πρόσβαση σε επιλεγμένα δωμάτια σε εγκαταστάσεις, κτίρια ή περιβάλλον γραφείου και χρησιμοποιούνται σπάνια για τον έλεγχο της πρόσβασης σε όλες τις πόρτες. Ο **έλεγχος πρόσβασης** είναι επίσης κρίσιμος για τα χρηματοπιστωτικά ιδρύματα, ειδικά εκείνα που διαθέτουν θυρίδες ασφαλείας και άλλες περιοχές που χρειάζονται ασφάλεια. Το κύριο όφελος για τον πελάτη είναι η τραπεζική ασφάλεια. Ωστόσο, έχει το πρόσθετο πλεονέκτημα ότι επιτρέπει στους πελάτες να έχουν πρόσβαση στις θυρίδες ασφαλείας τους χωρίς να χρειάζεται να περιμένουν βοήθεια από μέλος του προσωπικού της τράπεζας.

Figure 14 - Μέθοδοι ελέγχου πρόσβασης

Πηγή: www.protectingschools.com

Οι βιομετρικές τεχνολογίες μπορούν επίσης να χρησιμοποιηθούν για τον εντοπισμό ή την επαλήθευση της ταυτότητας των ατόμων που πραγματοποιούν συναλλαγές σε ATM. Ο βιομετρικός έλεγχος ταυτότητας για τον έλεγχο πρόσβασης είναι μια εξαιρετικά επιτακτική λύση, όπως τα κλειδιά και τα συνθηματικά μπορούν εύκολα να μοιραστούν χωρίς να μπορούν να οδηγήσουν ξανά στον πραγματικό χρήστη, ενώ τα βιομετρικά στοιχεία δεν μπορούν να μοιραστούν ή να κλαπούν. Επίσης, όταν χρησιμοποιείται ένα βιομετρικό σύστημα για τη δημιουργία ενός διαγράμματος ελέγχου, είναι δύσκολο να αμφισβητηθεί. Η **λογική πρόσβαση** είναι ένας άλλος τομέας ανάπτυξης για τη διαχείριση των βιομετρικών τεχνολογιών των εργαζομένων. Σε μεγάλο βαθμό, χρησιμοποιούν συσκευές σάρωσης δακτυλικών αποτυπωμάτων για πρόσβαση σε σταθμούς εργασίας και δίκτυα. Οι εφαρμογές αυτές στοχεύουν γενικά στην αύξηση της ασφάλειας του δικτύου και της παραγωγικότητας των εργαζομένων. Επιπλέον, οι τεχνολογίες αυτές χρησιμοποιούνται για τον έλεγχο και τη βεβαίωση παρουσίας και ώρας προσέλευσης ατόμων, μια λειτουργία που δεν αποσκοπεί τόσο στον έλεγχο προσπέλασης αλλά στον εντοπισμό και την αποφυγή κάθε απόπειρας απάτης ή πλαστοπροσωπίας. Επιτυχή και ορθώς εγκατεστημένα συστήματα φυσικής πρόσβασης έχουν εξοικονομήσει εκατοντάδες χιλιάδες δολάρια σε εταιρείες, ακριβώς επειδή αποτρέπουν τέτοιες απόπειρες.

Όσον αφορά το **ηλεκτρονικό εμπόριο** (e-commerce) και τις **διαδικτυακές συναλλαγές** (internet banking) αποτελούν κλάδους όπου η εισαγωγή βιομετρικών τεχνολογιών για ταυτοποίηση και αυθεντικοποίηση σε απομακρυσμένες συναλλαγές αγαθών και υπηρεσιών θεωρείται επιθυμητή, με την εφαρμογή των τεχνολογιών αυτών να δρα συμπληρωματικά ή να αντικαθιστά πλήρως άλλες αντίστοιχες τεχνολογίες όπως η χρήση συνθηματικών (passwords), τετραψήφιων κωδικών (PINs) και

ερωτήσεων αυθεντικοποίησης. Η διαδικτυακή διαχείριση λογαριασμών με χρήση βιομετρικής ταυτοποίησης αναμένεται να κατακλύσει την παγκόσμια αγορά και να γίνει ευρέως χρησιμοποιούμενη από καταναλωτές σε πολλά είδη εφαρμογών, ειδικά με την εισαγωγή υλικού που υποστηρίζει τις εφαρμογές αυτές σε νέες κινητές συσκευές. Συνεπώς, οι υπάρχουσες πλέον υποδομές για ένα τέτοιο σύστημα το καθιστούν άμεσα διαθέσιμο στον καταναλωτή, και οικονομικότερο στην υλοποίηση για τον πάροχο αφού πλέον δεν απαιτείται να παρέχει ο ίδιος τις επιπλέον τεχνολογίες στο καταναλωτικό κοινό του.

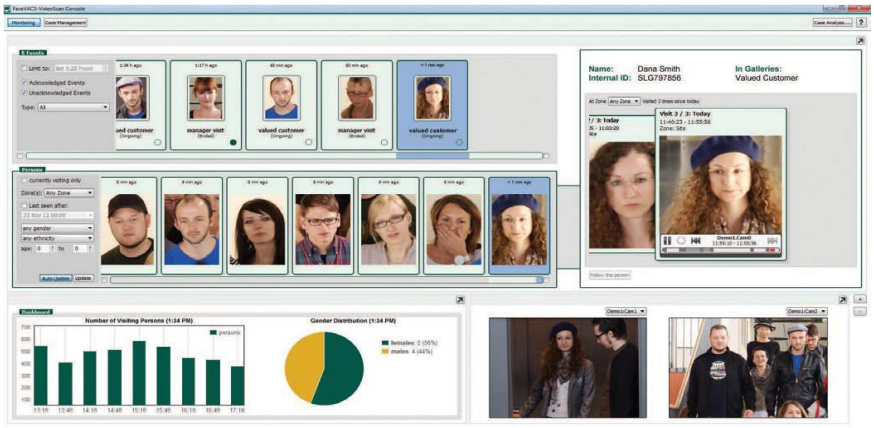


Figure 15 - Κρατική ταυτοποίηση – αναγνώριση προσώπου

Στον ***κυβερνητικό τομέα***, οι βιομετρικές τεχνολογίες χρησιμοποιούνται για ταυτοποίηση των πολιτών. Πιο συγκεκριμένα, εφαρμόζονται στην ταυτοποίηση και επαλήθευση ταυτότητας ατόμων που αλληλεπιδρούν με κυβερνητικές υπηρεσίες προς έκδοση κάποιων κυβερνητικών εγγράφων, ή για σκοπούς ψήφου, για μεταναστευτικούς λόγους ή έλεγχο μητρώων. Μελλοντικά, οι τεχνολογίες αυτές θα αντικαταστήσουν άλλες μεθόδους αυθεντικοποίησης όπως παροχή εγγράφων ή παραστατικών, καθώς και υπογραφών. Επιπλέον, θα παρέχουν λειτουργικότητα και προστασία απέναντι σε διπλότυπες εγγραφές προς εκμετάλλευση παροχών και γενικότερα ενάντια σε απάτες, πλαστοπροσωπείες και απόπειρες εκμετάλλευσης κυβερνητικών υπηρεσιών. Άλλες εφαρμογές των βιομετρικών συναντώνται ήδη στην προστασία συνόρων των Η.Π.Α. σε συνεργασία με τις κυβερνήσεις του Καναδά και του Μεξικού, με σκοπό τον περιορισμό της λαθρομετανάστευσης και του λαθρεμπορίου, ενώ στην Αυστραλία χρησιμοποιούνται βιομετρικά φωνητικής αναγνώρισης σε συστήματα φορολόγησης. Διεθνή αεροδρόμια υιοθετούν τις τεχνολογίες αυθεντικοποίησης παγκοσμίως, στη Νέα Ζηλανδία χρησιμοποιείται αναγνώριση προσώπου, σε ορισμένα αεροδρόμια της Αμερικής χρησιμοποιούνται σαρωτές δακτυλικών αποτυπωμάτων, και στο Ντουμπάι χρησιμοποιούνται συνδυαστικά η αναγνώριση προσώπου και η σάρωση αμφιβληστροειδούς.

Οι βιομετρικές πληροφορίες κάθε ατόμου αποθηκεύονται και προστατεύονται με βάση πρότυπα του Διεθνούς Οργανισμού Πολιτικής Αεροπορίας (International Civil Aviation Organisation - ICAO), υπό την επίβλεψη του ΟΗΕ. Προβλέπεται επίσης η δημιουργία ηλεκτρονικών δελτίων ταυτότητας και ηλεκτρονικών διαβατηρίων (e-passports) τα οποία θα περιέχουν chips με όλες τις βιομετρικές πληροφορίες του κατόχου του δελτίου, για την εφαρμογή όλων των παραπάνω υπηρεσιών και την εξάλειψη της κλοπής ταυτότητας (identity theft) σε κυβερνητικό και ομοσπονδιακό επίπεδο. Επιπροσθέτως, οι βιομετρικές τεχνολογίες βρίσκουν εφαρμογή και σε συστήματα παρακολούθησης (surveillance) καθιστώντας τη χειροκίνητη διαχείρισή τους μη απαραίτητη. Εν κατακλείδι, οι βιομετρικές τεχνολογίες μπορούν να χρησιμοποιηθούν για την πρόληψη της απάτης και της κατάχρησης κυβερνητικών παροχών και εξασφαλίζουν γρήγορη και εύκολη πρόσβαση στις κρατικές υπηρεσίες.

Η ***εγκληματολογική βιομετρία*** αξιοποιεί τις βιομετρικές τεχνολογίες για την επαλήθευση και αναγνώριση της ταυτότητας ενός υπόπτου στις διαδικασίες επιβολής του νόμου. Πρόκειται για τον πρώτο τομέα με εκτεταμένη χρήση των τεχνολογιών αυτών με χρήση μη αυτοματοποιημένων εφαρμογών, για δεκαετίες. Την τελευταία 25ετία χρησιμοποιούνται καθολικά κρατικές και διεθνείς βάσεις δεδομένων για αποθήκευση δακτυλικών αποτυπωμάτων και την πιο αυτοματοποιημένη επεξεργασία τους στην εγκληματολογία. Το AFIS, το αυτοματοποιημένο σύστημα αναγνώρισης δακτυλικών αποτυπωμάτων, έχει χρησιμοποιηθεί για τον εντοπισμό εγκληματιών εδώ και πολλά χρόνια από διεθνείς αστυνομικούς οργανισμούς. Το δακτυλικό αποτύπωμα είναι μια παγκοσμίως αναγνωρισμένη μέθοδος αναγνώρισης, που χρησιμοποιείται κατά κόρον από το FBI το οποίο διαθέτει τεράστια βάση δεδομένων αποτυπωμάτων. Πέρα από το AFIS, η σάρωση προσώπου χρησιμοποιείται στην εγκληματική ταυτοποίηση, αλλά με χαμηλότερο βαθμό ακρίβειας. Ωστόσο, σημειώνει ικανοποιητική πρόοδο σε ό, τι αφορά την παρακολούθηση πλήθους χρησιμοποιώντας υλικό από κλειστά κυκλώματα ασφαλείας (CCTV footage) και σάρωση εικόνων προσώπου σε ιστότοπους όπως το Facebook, για να βοηθήσει στον εντοπισμό ατόμων.[15] Με την πάροδο του χρόνου, είναι πιθανό ότι η σάρωση της ίριδας θα υιοθετηθεί στην εγκληματική ταυτοποίηση καθώς παρέχει ταχύτατη αναγνώριση, όμως καθώς εξακολουθεί να είναι μια σχετικά νέα και εξελισσόμενη βιομετρική, η έλλειψη βάσεων δεδομένων του συγκεκριμένου χαρακτηριστικού περιορίζει τη χρήση αυτής της τεχνολογίας. Εφαρμογή βρίσκει και η αναγνώριση φωνής, που χρησιμοποιείται για την

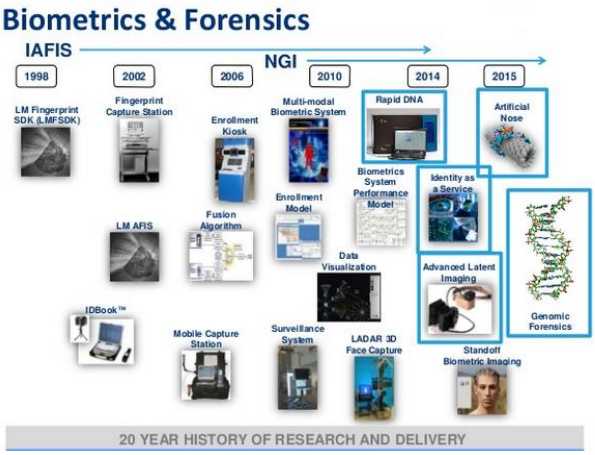


Figure 16 - 20 χρόνια εξέλιξης των Εγκληματολογικών Βιομετρικών Τεχνολογιών[19]

αναγνώριση ατόμων από καταγεγραμμένες συνομιλίες και την επαλήθευση ταυτότητας ατόμων που βρίσκονται υπό κατ' οίκον περιορισμό. Τέλος, το DNA επιτρέπει στους εγκληματολόγους να ταυτοποιήσουν άτομα από μικροσκοπικές ποσότητες βιολογικών υλικών. Ως βιομετρικό χαρακτηριστικό, το DNA έχει τρία βασικά οφέλη:

- Είναι μοναδικό για κάθε άτομο, και παραμένει αναλλοίωτο καθ' όλη τη ζωή του
- Είναι αξιόπιστο για τη δημιουργία μοναδικού προφίλ για κάθε άτομο
- Μπορεί να συλλεχθεί, να πολλαπλασιαστεί και να αναλυθεί ακόμα και από μικροσκοπικά δείγματα

Συχνά, η βιομετρική ταυτοποίηση με DNA βρίσκει εφαρμογή στην αναγνώριση θυμάτων καταστροφών (Disaster Victim Identification – DVI) με βάση συγκρίσεις του γενετικού υλικού των λειψάνων και των συγγενών των αγνοούμενων.[13] Πρόκειται για την πιο αποτελεσματική μέθοδο ταυτοποίησης και επαλήθευσης ταυτότητας, καθώς η πιθανότητα ύπαρξης ίδιου DNA σε δύο διαφορετικά άτομα είναι 1 προς 64 δισεκατομμύρια. Ωστόσο, ακόμα και αυτή η μέθοδος δεν είναι 100% ασφαλής, καθώς λόγω της φύσης του DNA και των νόμων της Μενδελικής κληρονομικότητας, δεν μπορεί να διακρίνει μεταξύ ομοζυγωτικών διδύμων, συνεπώς είναι επιθυμητό να χρησιμοποιείται σε συνδυασμό με άλλες βιομετρικές τεχνολογίες για να επιφέρει απολύτως ορθό αποτέλεσμα στην αναγνώριση ενός ατόμου. [12][18]

2.5.1 Βιομετρικά για προσωπική χρήση – Smartphone Biometrics

Τη σύγχρονη εποχή, οι έξυπνες κινητές συσκευές (smartphones) έχουν γίνει αναπόσπαστο κομμάτι της καθημερινής ζωής μας. Ένας τεράστιος αριθμός χρηστών έχει διαρκώς πάνω του τουλάχιστον μία τέτοια συσκευή, την οποία και χρησιμοποιεί συνεχώς για ποικίλους σκοπούς. Ασφαλώς, η ραγδαία αύξηση της δημοτικότητας των συσκευών αυτών και της χρησιμότητάς τους έχει ως φυσική συνέπεια τη διαρκή εξέλιξή τους ως προς την τεχνολογία που διαθέτουν, και τα πολυάριθμα ετερογενή δεδομένα που επεξεργάζονται. Οι συσκευές αυτές διαθέτουν πολυάριθμες νέες λειτουργίες και ενσωματώνουν πολλούς διαφορετικούς και ισχυρούς αισθητήρες. Η τελευταία γενιά έξυπνων τηλεφώνων είναι ιδιαίτερα εξοπλισμένη με αισθητήρες όπως:

Figure 17 - Δακτυλικό αποτύπωμα σε smartphone

- αισθητήρες τοποθεσίας (GPS)
- οπτικούς αισθητήρες (κάμερες)
- αισθητήρες ήχου (μικρόφωνα)

- αισθητήρες φωτός
- αισθητήρες θερμοκρασίας
- αισθητήρες κατεύθυνσης (πυξίδες)
- αισθητήρες επιτάχυνσης (accelerometer). [16][18][19]

Η νεότερη γενιά των συσκευών αυτών εισάγει και έναν ανιχνευτή δακτυλικού αποτυπώματος στην τεχνολογία των κινητών τηλεφώνων, ενώ χρησιμοποιεί και τους υπάρχοντες αισθητήρες για να εισάγει τη βιομετρική τεχνολογία στις προσωπικές συσκευές των χρηστών.

Κατά βάση, τα βιομετρικά στις κινητές συσκευές χρησιμοποιούνται για το ξεκλείδωμά τους και την πρόσβαση στα αρχεία, τις εφαρμογές και τα προσωπικά δεδομένα που περιέχουν. Τα πιο συνηθισμένα βιομετρικά χαρακτηριστικά που χρησιμοποιούνται είναι το δακτυλικό αποτύπωμα, η αναγνώριση προσώπου, η αναγνώριση φωνής και η σάρωση της ίριδας, ενώ χρησιμοποιούνται πολλοί αισθητήρες για ανίχνευση συμπεριφορικών χαρακτηριστικών.

Το 2013 η Apple έθεσε σε κυκλοφορία το iPhone 5S, το πρώτο smartphone που διαθέτει **σαρωτή δακτυλικών αποτυπωμάτων**. Η συσκευή επιτρέπει στους χρήστες να ξεκλειδώνουν το τηλέφωνο με το δακτυλικό τους αποτύπωμα και στη συνέχεια τους έδωσε τη δυνατότητα να πιστοποιούν με αυτό τον τρόπο την ταυτότητά τους πριν πραγματοποιήσουν μια ηλεκτρονική πληρωμή με την υπηρεσία Apple Pay. Έκτοτε, πολλοί κατασκευαστές smartphones εισήγαγαν στις δικές τους συσκευές σαρωτές δακτυλικών αποτυπωμάτων.

Figure 18 - Touch ID της Apple

Πηγή: https://en.wikipedia.org/wiki/Touch_ID

Το Samsung Galaxy S8 ήταν μία από τις πρώτες συσκευές που υλοποίησε την **αναγνώριση προσώπου**, αλλά ήδη από το 2011 υπήρχε ως χαρακτηριστικό στη συσκευή Galaxy Nexus της Google σε συνεργασία με τη Samsung. Με το S8 το Face Unlock έγινε πιο διαδεδομένο, αλλά και πάλι το μεγάλο εμπορικό άλμα πραγματοποιήθηκε από το iPhone X της Apple με το Face ID, το οποίο εισήγαγε αμέσως στην υπηρεσία Apple Pay και έκανε πολύ πιο εύχρηστη την αυθεντικοποίηση με αναγνώριση προσώπου. Αν και χρησιμοποιείται στην υπηρεσία πληρωμών της Apple, το λογισμικό σε πολλά Android smartphones θεωρείται ότι δεν είναι αρκετά ασφαλές για χρήση με εφαρμογές τραπεζικής ή κινητής τηλεφωνίας. Παρόλα αυτά, η Android παρέχει πολλές βιομετρικές επιλογές ασφαλείας και επιτρέπει τη συνδυαστική και ταυτόχρονη χρήση περισσότερων από μίας για διάφορες λειτουργίες, πχ. αναγνώριση προσώπου για ξεκλείδωμα και δακτυλικό αποτύπωμα για επαλήθευση πληρωμών.

Η **σάρωση της ίριδας** είναι αρκετά διαδεδομένη στις συσκευές αλλά όχι και στους χρήστες, καθώς δεν είναι ευρέως γνωστό χαρακτηριστικό. Συναντάται σε συσκευές όπως το Fujitsu NX F-04G και το Microsoft Lumia 950, τα οποία κυκλοφόρησαν το 2015. Η Samsung παρουσίασε επίσης το

χαρακτηριστικό γνώρισμα στο Galaxy Note 7 το 2016 και αργότερα στο Galaxy S8 και Note 8 το 2017. Οι χρήστες απλά ευθυγραμμίζουν τα μάτια τους με το πλαίσιο σάρωσης στα τηλέφωνά τους. Όπως η αναγνώριση προσώπου σε πολλά Android smartphones, η σάρωση ίριδας θεωρείται επίσης ότι δεν είναι αρκετά ασφαλής για τον έλεγχο των τραπεζικών συναλλαγών και των εφαρμογών που σχετίζονται με πληρωμές. Οι χρήστες πρέπει επίσης να έχουν έναν κωδικό πρόσβασης και να δημιουργούν αντίγραφο ασφαλείας σε περίπτωση αποτυχίας κάποιας ή όλων των επιλογών βιομετρικών στοιχείων.

Παρουσιάζει ενδιαφέρον το ότι η **αναγνώριση φωνής** ως βιομετρική αυθεντικοποίηση δεν είναι ευρέως διαθέσιμη, αν και μεγάλες εταιρείες αρχίζουν να προωθούν σχετικό λογισμικό και τα προϊόντα τεχνολογίας είναι διαθέσιμα στο κοινό. Το smartphone LG V30 περιλαμβάνει μια λειτουργία ξεκλειδώματος φωνής που ονομάζεται Voice Print, η οποία επιτρέπει στους χρήστες να ορίσουν μια φράση που το τηλέφωνο μπορεί να αναγνωρίσει και να ξεκλειδώσει με εντολή. Η αναγνώριση φωνής ενσωματώνεται επίσης στο λειτουργικό σύστημα Android μέσω των ρυθμίσεων Smart Lock. [22]

Οι Pisani et. al. προτείνουν ένα σύστημα που χρησιμοποιεί **αισθητήρες επιτάχυνσης** στο τηλέφωνο (επιταχυνσιόμετρα) για τον εντοπισμό και την ταυτοποίηση χρηστών κινητών τηλεφώνων. Αυτή η μορφή ταυτοποίησης βάσει συμπεριφορικής βιομετρίας είναι δυνατή, επειδή οι συνολικές κινήσεις ενός ατόμου μπορούν να αποτελούν μια μοναδική υπογραφή και αυτό αντικατοπτρίζεται στα δεδομένα επιταχυνσιόμετρου που παράγουν. Καθημερινές δραστηριότητες όπως περπάτημα, τρέξιμο και ανέβασμα σκάλας, ενώνονται ως δεδομένα και εφαρμόζονται σε αλγόριθμους τυποποιημένης ταξινόμησης για τη δημιουργία προτύπων. Αυτά τα πρότυπα είτε διακρίνουν την ταυτότητα του ατόμου από το σύνολο των χρηστών (ταυτοποίηση) χρήστη ή αναγνωρίζουν αν ο χρήστης είναι ένας συγκεκριμένος άνθρωπος (έλεγχος ταυτότητας). Το μοντέλο αυτό επιτρέπει την ταυτοποίηση και την αυθεντικοποίηση χωρίς οι χρήστες να κάνουν επιπλέον ενέργειες - το μόνο που χρειάζεται να κάνουν είναι να φέρουν τα κινητά τους τηλέφωνα. Υπάρχουν πολλές χρήσεις για αυτήν τη μελέτη, πχ. για την παροχή ασφάλειας συσκευών επιτρέποντας τη χρήση μόνο για συγκεκριμένους χρήστες και προσφέροντας ένα επιπλέον επίπεδο επαλήθευσης ταυτότητας.[21]

2.5.2 Βιομετρικά και Android

Όπως αναφέραμε και σε προηγούμενη ενότητα, για την ασφάλεια των χρηστών πολλές συσκευές και εφαρμογές χρησιμοποιούν μηχανισμούς αυθεντικοποίησης και επαλήθευσης ταυτότητας. Οι μηχανισμοί αυτοί διακρίνονται σε τρεις κατηγορίες: μηχανισμούς που βασίζονται σε παράγοντες γνώσης (κάτι που γνωρίζει ο χρήστης, πχ PIN ή συνθηματικό ασφαλείας), μηχανισμούς που βασίζονται σε παράγοντες κατοχής (κάτι που κατέχει ο χρήστης όπως token ή κλειδί) και μηχανισμούς που βασίζονται σε βιομετρικούς παράγοντες (κάτι που είναι ο χρήστης, όπως το δακτυλικό αποτύπωμα, η ίρις και το πρόσωπο).

Figure 19 - Android: Είδη αυθεντικοποίησης

Οι βιομετρικοί μηχανισμοί αυθεντικοποίησης γίνονται πολύ δημοφιλείς στο καταναλωτικό κοινό, προφανώς λόγω της ταχύτητας και ευχρηστίας τους, αλλά και της ασφάλειας που φαίνονται να παρέχουν. Είναι πιο γρήγορες από την πληκτρολόγηση συνθηματικού και δεν απαιτούν την απομνημόνευση κωδικού, ενώ αποτρέπουν τον πιο γνωστό κίνδυνο γνωσιακής αυθεντικοποίησης, το λεγόμενο shoulder surfing, δηλαδή την υποκλοπή στοιχείων αυθεντικοποίησης σε συσκευή απλώς κοιτώντας το χρήστη να πληκτρολογεί. Η βελτίωση των τεχνολογιών στο λειτουργικό Android προφανώς αποσκοπεί στην αύξηση της ασφάλειας των χρηστών αλλά και στη διευκόλυνση των developers να ενσωματώσουν βιομετρική αυθεντικοποίηση στις εφαρμογές τους, με την παροχή μιας κοινής πλατφόρμας.

Η αποδοτικότητα και η ποιότητα της απόδοσης των βιομετρικών μετράται με βάση δύο μετρικές έννοιες του machine learning: το False Accept Rate (FAR) και το False Reject Rate (FRR). Το FAR μετρά το ποσοστό συχνότητας αποδοχής μη έγκυρων χρηστών, ενώ το FRR μετρά το ποσοστό συχνότητας απόρριψης έγκυρων χρηστών. Τις έννοιες αυτές θα τις μελετήσουμε πιο αναλυτικά στην επόμενη ενότητα. Εν προκειμένω, το FAR μετρά πόσο συχνά ένας άλλος χρήστης αναγνωρίζεται λανθασμένα ως ο νόμιμος κάτοχος της συσκευής, ενώ το FRR μετρά πόσο συχνά ένας νόμιμος κάτοχος της συσκευής πρέπει να ξαναδοκιμάσει τον έλεγχο ταυτότητας. Προφανώς, το πρώτο αποτελεί ζήτημα ασφάλειας, ενώ το δεύτερο είναι πρόβλημα χρηστικότητας. Οι μέθοδοι αυτές μετρούν αποτελεσματικά τη στατιστική αποδοτικότητα των βιομετρικών, αλλά δεν παρέχουν επαρκή μέτρηση απόδοσης απέναντι σε επιθέσεις. Για το λόγο αυτό, η Android εισήγαγε νέες μετρικές μετά το Android 8.1 για τον υπολογισμό απόδοσης των βιομετρικών σε περίπτωση επίθεσης: το Spoof Accept Rate (SAR) δηλαδή το ποσοστό αποδοχής απάτης, και το Imposter Accept Rate (IAR) δηλαδή ποσοστό αποδοχής πλαστοπροσωπίας. Αυτές οι μετρικές υπολογίζουν πόσο εύκολα ένας εισβολέας μπορεί να παρακάμψει ένα βιομετρικό σύστημα ελέγχου ταυτότητας. Το spoofing αναφέρεται στη χρήση μιας γνωστής «καλής» καταγραφής (π.χ. αναπαραγωγή μιας φωνητικής εγγραφής ή χρήση εικόνας προσώπου ή δακτυλικών αποτυπωμάτων), ενώ η πλαστοπροσωπία αφορά σε μια

επιτυχημένη μίμηση της βιομετρίας άλλου χρήστη (π.χ. προσπάθεια να ακούγεται ή να μοιάζει με το χρήστη).

Ο τρόπος υλοποίησης και λειτουργίας θα μελετηθεί πιο αναλυτικά στην επόμενη υποενότητα.[12]

2.5.3 Βιομετρικά και e-Banking

Η πρόθεση από περισσότερους κατασκευαστές smartphone για ενσωμάτωση βιομετρικών δυνατοτήτων στις συσκευές τους παράλληλα με τα διαλειτουργικά πρότυπα πιστοποίησης, όπως το FIDO, θα συμβάλει στην ταχεία και ευρεία υιοθέτηση τεχνολογιών ελέγχου ταυτότητας κινητών βιομετρικών στοιχείων. Οι βιομετρικές τεχνολογίες σε κινητές συσκευές θα διαδοθούν λόγω των αυξημένων πωλήσεων smartphones και θα χρησιμοποιηθούν κυρίως για την προστασία του κινητού εμπορίου και των τραπεζικών συναλλαγών.

Figure 20 - Προστασία στο e-Banking

Όταν ένας χρήστης πραγματοποιεί μια συναλλαγή EFTPOS [1], χρησιμοποιεί ταυτόχρονα αυθεντικοποίηση με παράγοντα κατοχής (την τραπεζική κάρτα που διαθέτει) και παράγοντα γνώσης (τον αριθμό PIN του) για να ταυτοποιήσει τον εαυτό του. Αυτή η προσέγγιση ενέχει κινδύνους εκμετάλλευσης, σε περίπτωση κλοπής ή απώλειας της κάρτας, ή σε περίπτωση που ξεχαστεί ή διαμοιραστεί το PIN, με αποτέλεσμα να χρησιμοποιούνται οι παράγοντες αυτοί από μη εγκεκριμένα άτομα. Επιπλέον, στη γνωσιακή αυθεντικοποίηση τα διαπιστευτήρια μπορεί να είναι δύσκολο να απομνημονευθούν ή να είναι εύκολο να τα μαντέψει ο επιτιθέμενος. Τα συνθηματικά (passwords) μπορεί να ξεχαστούν ή να τα μαντέψει ο επιτιθέμενος μέσω social engineering, ενώ τα PIN δεν είναι μοναδικά – τα 20 συνηθέστερα PIN αντιστοιχούν στο 25% των χρησιμοποιούμενων τετραψήφιων κωδικών πρόσβασης. Συνεπώς το σύστημα δε μπορεί να διακρίνει τον έγκυρο χρήστη από τον απατεώνα. Εν αντιθέσει με τις παραπάνω μεθόδους, η βιομετρική αυθεντικοποίηση δεν ενέχει αυτούς τους κινδύνους. Σε καμία περίπτωση όμως δεν αποτελεί πανάκεια. Οποιοσδήποτε συνδυασμός των μεθόδων αυτών ωστόσο μπορεί να παρέχει εργαλεία ισχυρής αυθεντικοποίησης και να επιλύσει προβλήματα false positive και false negative περιπτώσεων, αλλά και να αποτρέψει την υποκλοπή ταυτότητας (identity theft) και άλλες περιπτώσεις απάτης. Προφανώς, η μείωση χρήσης τέτοιων κωδικών και η αξιοποίηση μοναδικών φυσικών χαρακτηριστικών ή βιομετρικών στοιχείων για την ταυτοποίηση είναι πιο επωφελής.

[1] **Electronic funds transfer at point of sale** (**EFTPOS**) – σύστημα ηλεκτρονικών συναλλαγών βάσει καρτών (πιστωτικών και χρεωστικών) σε τερματικά πληρωμών[23]

Οι ειδικοί είχαν προβλέψει τη ραγδαία αύξηση των συναλλαγών μέσω κινητής και υπολογίζεται πως ο αριθμός τους θα τριπλασιαστεί έως το 2020. Η βιομετρία αναμένεται να επιταχύνει τις συναλλαγές

Figure 21 - Identity theft και e-Banking

αυτές, να προσφέρει υψηλότερα επίπεδα ασφαλείας και να διευκολύνει την εμπειρία χρήστη των πελατών. Προφανώς, αυτό συνεπάγεται και την εξέλιξη των τεχνολογιών αυτών, ενώ επιπλέον, αναμένεται πως θα ενσωματωθούν περισσότερες βιομετρικές τεχνολογίες στις έξυπνες φορητές συσκευές επόμενης γενιάς. Με βάση ένα κεντρικό σύστημα βιομετρικής αναγνώρισης, οι κινητές συσκευές βιομετρικής αναγνώρισης επεκτείνουν τη λειτουργικότητα και τις δυνατότητες ενός στατικού συστήματος αναγνώρισης επιτρέποντας στους χρήστες τη λήψη δακτυλικών αποτυπωμάτων και εικόνων προσώπου ή για τη σύγκριση προτύπων ή εικόνων με μια βάση δεδομένων, είτε αποθηκευμένη τοπικά στη συσκευή είτε απομακρυσμένα σε συγκεντρωτικά βιομετρικά συστήματα αντιστοίχισης. Στη δεύτερη περίπτωση η κινητή συσκευή και το τμήμα αναγνώρισης επικοινωνεί με την κεντρική βάση μέσω κοινών ασύρματων τεχνολογιών (3G, Wi-Fi, Bluetooth) και εάν κατά τη σύγκριση βρεθεί ταυτοποίηση, οι πληροφορίες του συγκεκριμένου ατόμου μεταφέρονται πίσω στην κινητή συσκευή. Η ανάπτυξη θα ακολουθηθεί γρήγορα από άλλες καινοτόμες βιομετρικές τεχνολογίες που αναπτύσσονται ως μέρος είτε των λύσεων FIDO Aware, είτε ενσωματωμένες σε πλατφόρμες αυθεντικοποίησης πολλαπλών παραγόντων. Το πλήθος και η ποικιλότητα αισθητήρων input και οι μέθοδοι βιομετρικής αυθεντικοποίησης που επιτρέπουν, μπορούν να αντικαταστήσουν πλήρως τα συστήματα πρόσβασης με χρήση κωδικών (passwords) ειδικά με δεδομένο ότι είναι ιδιαίτερα εύχρηστα και οι χρήστες τα αξιοποιούν ήδη για απλές λειτουργίες, όπως το ξεκλείδωμα της συσκευής τους. Συνδυασμοί βιομετρικών χαρακτηριστικών θα μπορούν να ενσωματωθούν σε σύστημα ελέγχου ταυτότητας δύο (ή περισσότερων) παραγόντων, στα πλαίσια συναλλαγών, επαγγελματικών ταυτοποιήσεων και ελέγχου ή διαχείρισης πρόσβασης. Σύντομα, η τεχνολογία αυτή θα υιοθετηθεί σε μεγάλο βαθμό από χρηματοπιστωτικές υπηρεσίες και τραπεζικά ιδρύματα, για την αυθεντικοποίηση εργαζομένων και χρηστών, τη διευκόλυνση και επιτάχυνση της εξυπηρέτησης πελατών και την πρόληψη και αποτροπή κλοπής ταυτότητας σε συναλλαγές.

2.6 Λειτουργία

2.6.1 Βιομετρικά Συστήματα

Τα βιομετρικά συστήματα αποτελούνται από πέντε στοιχεία:

- Έναν αισθητήρα για τη συλλογή και ψηφιοποίηση του βιομετρικού χαρακτηριστικού
- Ένα τμήμα αξιολόγησης ποιότητας του δείγματος και εξαγωγής χαρακτηριστικών

- Ένα τμήμα σύγκρισης, που συγκρίνει και, όπου γίνεται, ταυτίζει το δείγμα με άλλα υπάρχοντα στη βάση δεδομένων
- Η Βάση Δεδομένων, όπου αποθηκεύονται τα δείγματα
- Το τμήμα απόφασης, που με βάση τα αποτελέσματα της σύγκρισης επιστρέφει ένα αποτέλεσμα

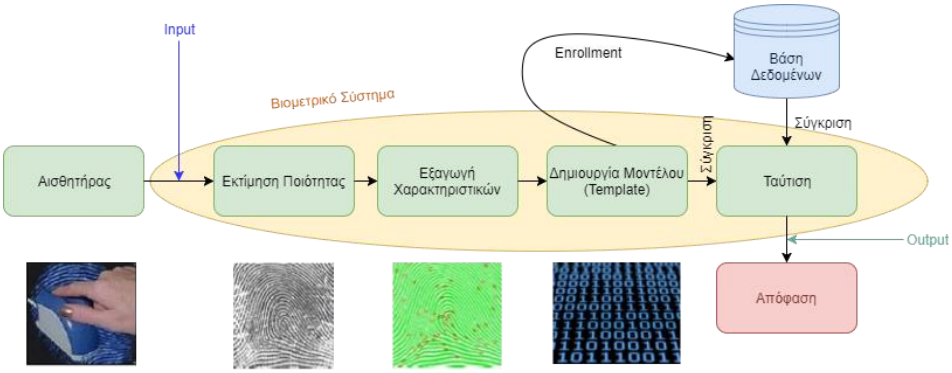


Figure 22 - Βιομετρικό Σύστημα

Το κυρίως σύστημα περιλαμβάνει και εσωτερικά υποσυστήματα, και τις εξής βασικές λειτουργίες:

- Εγγραφή
- Ταυτοποίηση ή Επαλήθευση ταυτότητας
- Συλλογή δεδομένων
- Επεξεργασία δεδομένων (εξαγωγή χαρακτηριστικών)
- Αποθήκευση δεδομένων σε Βάση
- Σύγκριση για αυθεντικοποίηση

Κατά την εγγραφή του χρήστη, το τμήμα του αισθητήρα καταγράφει τα βιομετρικά δεδομένα του ατόμου (raw data) αλλά παρέχει επίσης επιπλέον βοηθητικές λειτουργίες για την επεξεργασία, την αποθήκευση και τη σύγκριση των δεδομένων αυτών. Εσφαλμένη εγκατάσταση ή κακή λειτουργία των αισθητήρων προκαλεί υψηλό FAR, το ποσοστό που μετρά την αποτυχία συλλογής και εξαγωγής κατάλληλων βιομετρικών δεδομένων.

False Acceptance Rate (FAR): Μετρά την πιθανότητα ένα βιομετρικό σύστημα να ταυτοποιήσει εσφαλμένα έναν μη εξουσιοδοτημένο χρήστη ως έγκυρο. Προφανώς, χαμηλό FAR σημαίνει υψηλό επίπεδο ασφάλειας.

$$FAR = \frac{\text{Πλήθος Εσφαλμένων Αποδοχών}}{\text{Πλήθος Προσπαθειών Ταυτοποίησης}}$$

Equation 1 - FAR

Η σωστή υλοποίηση του βιομετρικού **αισθητήρα** απαιτεί τον ορισμό μιας υλικής διεπαφής, που συνήθως βασίζεται σε βιομηχανικά πρότυπα. Αυτά ορίζουν τόσο τη διεπαφή όσο και τα πρότυπα επικοινωνίας μεταξύ συσκευών. Αυτό είναι απαραίτητο καθώς τα βιομετρικά δεδομένα, ειδικά τα raw data, μπορεί να αποθηκεύονται ως μεγάλου μεγέθους αρχεία και συνεπώς απαιτείται μια διεπαφή με ικανοποιητική ταχύτητα συνδεσιμότητας.

Στη συνέχεια, τα ικανοποιητικά δεδομένα αναλύονται μέσα στο τμήμα αξιολόγησης. Η διαδικασία αυτή βελτιώνει την ποιότητα των αρχείων (πχ. εικόνων) και στη συνέχεια διαπιστώνει εάν είναι κατάλληλα για περαιτέρω επεξεργασία ή απαιτείται εκ νέου παροχή βιομετρικού υλικού.

Το τμήμα της Βάσης Δεδομένων είναι ο χώρος αποθήκευσης του συνόλου χαρακτηριστικών που συλλέχθηκαν και αποθηκεύτηκαν ως template κατά την εγγραφή, και στον ίδιο χώρο αποθηκεύονται τα προσωπικά στοιχεία και δεδομένα του χρήστη (ονοματεπώνυμο, ID, PIN, διεύθυνση κλπ.). Συνήθως, συγκεντρώνονται πολλαπλά δείγματα από κάθε άτομο, για να υπολογίζονται τυχόν αλλαγές και μεταβλητές.

Το σύνολο των εξαγομένων χαρακτηριστικών συγκρίνεται με τα αποθηκευμένα στη Βάση δεδομένα μέσω ενός αλγορίθμου που παράγει ένα ποσοστό ταύτισης. Μέσα από τον αλγόριθμο λήψης απόφασης, προκύπτει εάν υπάρχει ταύτιση ή όχι. Αφού ληφθεί η απόφαση, το σύστημα είτε παραχωρεί πρόσβαση στο χρήστη ή την αρνείται.

Εκτός από το δείκτη FAR, η απόδοση ενός βιομετρικού συστήματος μετράται και με άλλους τρόπους. Συμπληρωματικά με το FAR, χρησιμοποιείται ο δείκτης **FRR (False Rejection Rate)** που αποτελεί το ποσοστό ή τη συχνότητα απόρριψης και άρνησης πρόσβασης έγκυρων χρηστών από το σύστημα.

$$FRR = \frac{\text{Πλήθος Εσφαλμένων Αρνήσεων}}{\text{Πλήθος Προσπαθειών Ταυτοποίησης}}$$

Equation 2 – FRR

Η μέτρηση της απόδοσης είναι πολύ σημαντική, αφού προσδιορίζουν την επιτυχία ορθής και ακριβούς ταύτισης του ατόμου με το αντίστοιχο template τους από το σύστημα, και άρα το πόσο ισχυρή ασφάλεια παρέχει. [24][25]

2.6.2 Ισχυρά και μη Βιομετρικά Συστήματα κατά Android

Η Google χρησιμοποιεί την αναλογία SAR/IAR για την κατηγοριοποίηση των μηχανισμών αυθεντικοποίησης σε ισχυρούς και μη. Όπως αναφέραμε συνοπτικά σε προηγούμενη υποενότητα, οι δύο νέες αυτές μετρικές εισήχθησαν από την έκδοση Android 8.1 και εξής, για να βοηθήσουν τους κατασκευαστές υλικού στην ακριβή εκτίμηση ασφαλείας των συστημάτων τους.

Imposter Accept Rate (IAR): Η πιθανότητα το σύστημα να αποδεχτεί ως έγκυρο input που προσπαθεί να μιμηθεί γνωστό έγκυρο δείγμα (πλαστοπροσωπία).

Spoof Accept Rate (SAR): Η πιθανότητα το σύστημα να αποδεχτεί ως έγκυρο δείγμα ένα καταγεγραμμένο γνωστό έγκυρο δείγμα (υποκλοπή).

Οι μετρήσεις IAR δεν είναι χρηστικές για κάθε είδος βιομετρικών (modalities), ωστόσο οι μετρήσεις SAR είναι.

Οι μηχανισμοί βιομετρικής αυθεντικοποίησης με αναλογία SAR/IAR ≤ 7% θεωρούνται ισχυροί, ενώ για SAR/IAR ≥ 7% θεωρούνται αδύναμοι. Το ποσοστό αυτό επιλέγεται συγκεκριμένα με βάση τις περισσότερες υλοποιήσεις συστημάτων για δακτυλικά αποτυπώματα. Με την εξέλιξη των τεχνολογιών και τη βελτίωση της απόδοσης τους, το ποσοστό αυτό ενδέχεται να αλλάξει. Ο σκοπός αυτών των μετρήσεων είναι να μην περιορίζουν κατασκευαστές και χρήστες με αδύναμα βιομετρικά, αλλά να προσθέτουν επιπλέον βήματα αυθεντικοποίησης ώστε να μειωθεί ο κίνδυνος μη εξουσιοδοτημένης πρόσβασης.

2.6.3 BiometricPrompt API

Μετά το Android P, η Google διέθεσε στους προγραμματιστές ένα BiometricPrompt API για να ενσωματώσουν βιομετρική αυθεντικοποίηση στις εφαρμογές τους χωρίς απαραίτητα να γνωρίζουν τη λειτουργία τους. Η συνοδευτική βιβλιοθήκη επιτρέπει τη χρήση του API σε πολλά είδη συσκευών. Με την προσθήκη αυτή βεβαιώνεται ένα επαρκές επίπεδο ασφαλείας της εφαρμογής, ενώ διευκολύνει τους developers αφού αναγκάζει την πλατφόρμα να επιλέξει το καταλληλότερο βιομετρικό για αυθεντικοποίηση και δε χρειάζεται να κάνουν την επιλογή αυτή οι ίδιοι.

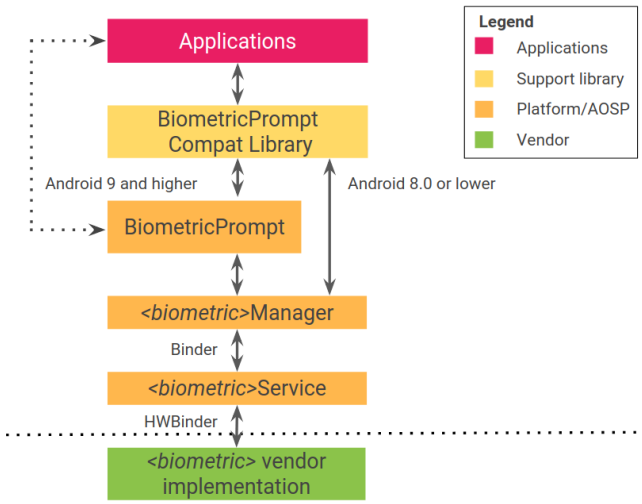


Figure 23 - Android: Αρχιτεκτονική Βιομετρικών [12]

```
val prompt = BiometricPrompt.Builder(this)
    .setTitle("Verify purchase")
    .setSubtitle("...")
    .setDescription("...")
    .setNegativeButton("Cancel", executor, listener)
    .build()

prompt.authenticate(
    cryptoObject, cancellationSignal, executor, callback);
```

Figure 24 - Κώδικας BiometricPrompt [12]

Η Android κάνει βήματα για την ενίσχυση της διαδικασίας αυθεντικοποίησης, με οδηγίες ασφαλούς σχεδιασμού, εύχρηστα και κοινά API και παρότρυνση των developers να χρησιμοποιούν ορθά τις τεχνολογίες αυτές. Με τον τρόπο αυτό, επιτυγχάνεται πολύ μεγαλύτερο επίπεδο ασφαλείας στη διαδικασία της αυθεντικοποίησης και την προστασία της ψηφιακής ταυτότητας του χρήστη.[12]

2.7 FIDO

Figure 25 - FIDO [27]

Η FIDO Alliance είναι μια βιομηχανική κοινοπραξία η οποία ασχολείται με τη διαλειτουργικότητα μεταξύ συσκευών και την ισχυρή αυθεντικοποίηση χρηστών σε αυτές. Με βάση τα ανοικτά και δωρεάν πρότυπα που δημιουργεί, η FIDO Authentication επιτρέπει την είσοδο χρήστη (login) με μεθόδους πιο γρήγορες και πιο ασφαλείς από τις παραδοσιακές μεθόδους με χρήση μόνο password. Τα πρωτόκολλα της χρησιμοποιούν κρυπτογράφηση δημοσίου κλειδιού (public key cryptography) και σχεδιάζονται έτσι ώστε να παρέχουν στο χρήστη ιδιωτικότητα «από κάτω προς τα πάνω». Έτσι, δεν παρέχουν πληροφορίες για το χρήστη σε άλλες δικτυακές υπηρεσίες, αποτρέπουν τον εντοπισμό του μέσω υπηρεσιών και δεδομένα όπως τα βιομετρικά του χρήστη αποθηκεύονται μόνο τοπικά, στη συσκευή του.

Figure 26 - Η ασφάλεια που παρέχει το FIDO [26]

Κατά την εγγραφή του χρήστη σε μια online υπηρεσία, η συσκευή του χρήστη (client device) δημιουργεί ένα νέο ζεύγος κλειδιών, ένα ιδιωτικό και ένα δημόσιο, το οποίο και εγγράφεται στην υπηρεσία. Η αυθεντικοποίηση του χρήστη πραγματοποιείται με την απόδειξη κατοχής του ιδιωτικού κλειδιού στην υπηρεσία μέσω κάποιου challenge. Τα ιδιωτικά κλειδιά του client μπορούν να

χρησιμοποιηθούν μόνο αν ο χρήστης τα ξεκλειδώσει τοπικά, στη συσκευή του, με χρήση PIN, βιομετρικού (φωνή, δακτυλικό αποτύπωμα κλπ.) ή συσκευής second factor. [26]

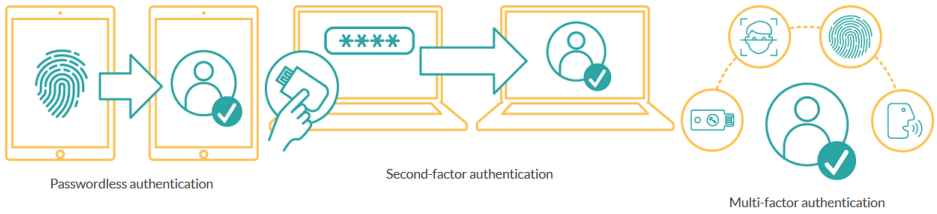


Figure 27 - Αυθεντικοποίηση FIDO [26]

Καθώς η δημοτικότητα των βιομετρικών αυξάνεται και τείνει να αντικαταστήσει πλήρως τα PINs, αυξάνεται και η ανησυχία της αποδοτικότητας τους, καθώς δεν έχει επαληθευτεί εφόσον πρόκειται για νέα ακόμα τεχνολογία. Για το λόγο αυτό η FIDO έχει θέσει σε εφαρμογή και πρόγραμμα πιστοποίησης βιομετρικών μέσων, για τη βεβαίωση της συμφωνίας τους με τα διεθνή πρότυπα. Η μελέτη και επαλήθευση της αποδοτικότητας λειτουργίας τους και το Presentation Attack Detection (PAD) γίνεται με πειράματα που εκτελούνται σε ανεξάρτητα εργαστήρια, για να βεβαιώσουν ότι η εφαρμογή είναι κατάλληλη για εμπορική χρήση. [27]

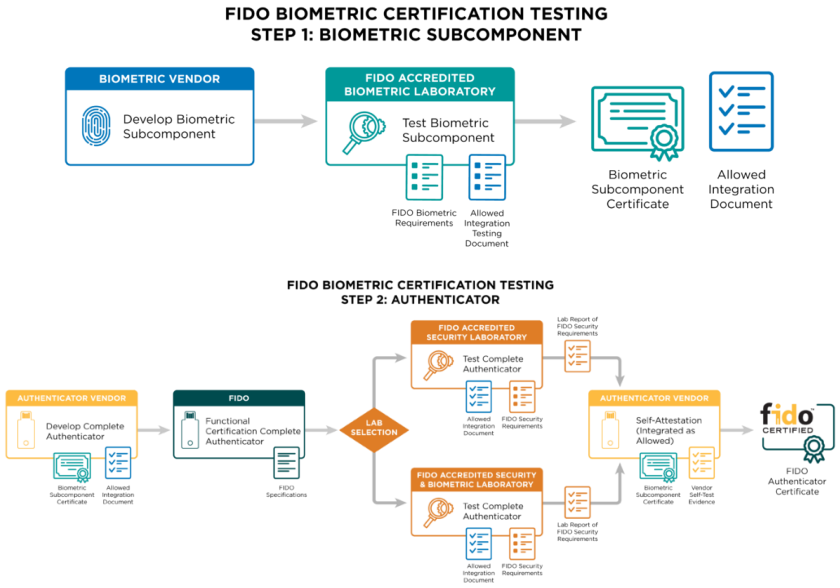


Figure 28 - Πιστοποίηση κατά FIDO [27]

Όπως αναφέρθηκε και πιο πριν, τα πρωτόκολλα FIDO χρησιμοποιούν κρυπτογράφηση δημοσίου κλειδιού για ισχυρή αυθεντικοποίηση, κατά την εγγραφή και είσοδο χρήστη σε οποιαδήποτε πλατφόρμα και υπηρεσία στο κινητό δίκτυο. Έχουμε ήδη μιλήσει για την **εγγραφή**, κατά την οποία η συσκευή του χρήστη (client device) δημιουργεί ένα νέο ζεύγος κλειδιών, ένα ιδιωτικό και ένα δημόσιο. Ο χρήστης επιλέγει ένα FIDO authenticator για την online υπηρεσία στην οποία εγγράφεται, τον ξεκλειδώνει με επιλεγμένη μέθοδο αυθεντικοποίησης (δακτυλικό αποτύπωμα, PIN, second factor συσκευή κ.ο.κ.) και έτσι δημιουργείται το ζεύγος κλειδιών για την τοπική συσκευή, την υπηρεσία και το λογαριασμό χρήστη. Στη συνέχεια το δημόσιο κλειδί αποστέλλεται στην υπηρεσία και συσχετίζεται στο σύστημα με το λογαριασμό χρήστη, ενώ το ιδιωτικό κλειδί και οποιεσδήποτε τεχνικές πληροφορίες (πχ. βιομετρικές μετρήσεις) αποθηκεύονται τοπικά και δε φεύγουν ποτέ από τη συσκευή.

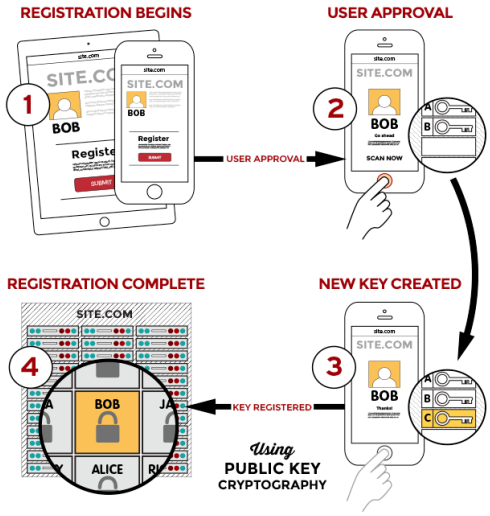


Figure 30 - Εγγραφή FIDO [25]

Όσον αφορά την **είσοδο** του χρήστη, η υπηρεσία χρησιμοποιεί challenges από μια προεγγεγραμμένη συσκευή του χρήστη, που να συμφωνεί με την πολιτική της. Ο χρήστης χρησιμοποιεί το FIDO authenticator με την ίδια μέθοδο που χρησιμοποίησε στην εγγραφή του, και η συσκευή του χρήστη χρησιμοποιεί το σωστό κλειδί για να υπογράψει το challenge και να το αποστείλει στην υπηρεσία, όπου επαληθεύεται με το αποθηκευμένο δημόσιο κλειδί και επιτρέπει την είσοδο στο χρήστη. [25]

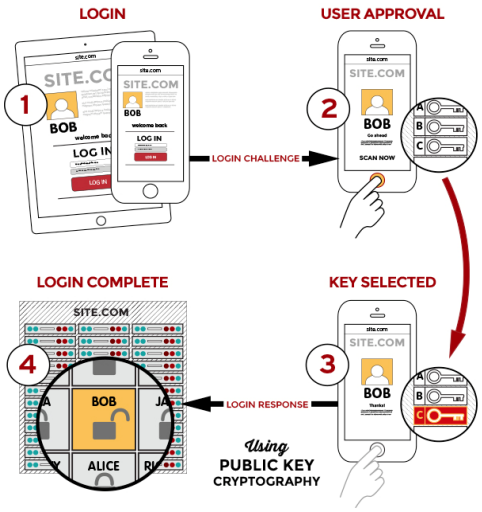


Figure 29 - Είσοδος FIDO [26]

2.8 Κίνδυνοι

Όπως ισχύει για κάθε τεχνολογία ασφαλείας, έτσι και τα βιομετρικά δεν είναι πανάκεια. Εσφαλμένος σχεδιασμός ενός συστήματος βιομετρικής αυθεντικοποίησης ή ανεπαρκής υλοποίηση του ενέχει κινδύνους ασφαλείας από πολλαπλές επιθέσεις ή αστοχίες από το χρήστη, ή και από κενά ασφαλείας στο σύστημα. Πχ.:

- Man in the middle (MITM)
- Biometric Storage (Digital lockers)
- BYOD (Bring your own device)
- IoT (Internet of things)
- Malware [30]

Όπως και με κάθε είδους διαπιστευτήρια χρήστη (credentials) έτσι και με τα βιομετρικά επιβάλλεται να λαμβάνονται προφυλάξεις κατά την αποθήκευση και μεταφορά τους, προς αποφυγή τέτοιων κινδύνων.

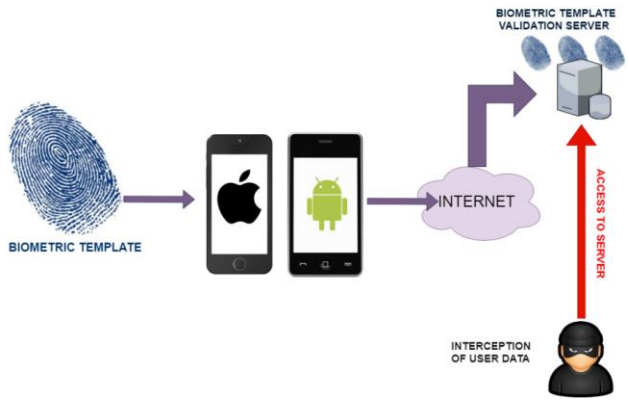


Figure 31 - MitM attack - Biometric Storage

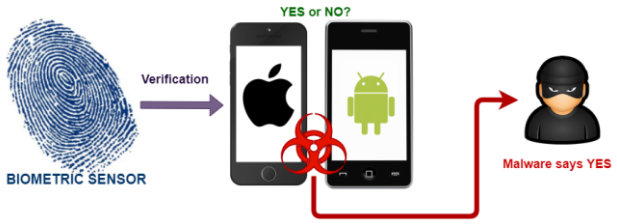


Figure 32 – Malware – Client-Side Verification Bypass

Οι γενικές αρχές για την ασφαλή αυθεντικοποίηση με βιομετρικά είναι στην πραγματικότητα καθολικές για την αποθήκευση διαπιστευτηρίων και την απομακρυσμένη αυθεντικοποίηση.

Πρέπει:

- ✓ Τα δεδομένα να αποθηκεύονται κρυπτογραφημένα
- ✓ Να εφαρμόζεται device tracking και behavioral analysis
- ✓ Να απαιτείται ασφάλεια τριών παραγόντων (3 factor)

Δεν πρέπει:

- × Να αποθηκεύονται βιομετρικά δεδομένα σε κεντρικό repository
- × Να γίνεται client side verification
- × Να γίνεται απομακρυσμένη επαλήθευση δεδομένων template
- × Να βασιζόμαστε αποκλειστικά σε passwords και PINs.

Τόσο η FIDO όσο και ο OWASP παρέχουν πλούσιο υλικό, πληροφορίες, πρότυπα και πιλοτικά προγράμματα για την ασφαλέστερη αυθεντικοποίηση σε εφαρμογές, προς αντιμετώπιση τέτοιων προβλημάτων και απειλών. [28]

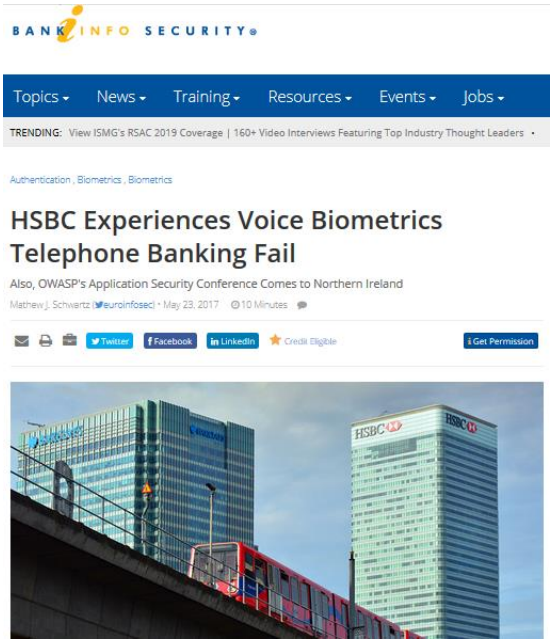


Figure 33 - HSBC Biometric Attack (2017) [29]

Κεφάλαιο 3ο – Πρακτική Εφαρμογή

3.1 Εισαγωγικά

Σε αυτή την ενότητα και με βάση τη θεωρία και τις πληροφορίες που εξετάσαμε στο προηγούμενο κεφάλαιο, θα μελετήσουμε πειραματικά ορισμένες εφαρμογές κινητών σε περιβάλλον Android, και συγκεκριμένα θα μελετήσουμε τη λειτουργία και απόδοση της βιομετρικής αυθεντικοποίησης σε ορισμένες Ελληνικές εμπορικές εφαρμογές του τραπεζικού τομέα. Επιπλέον, θα αναλύσουμε τη χρηστικότητα τους, τον πηγαίο κώδικα τους και την αποτελεσματικότητά τους.

Commented [Az1]: Safe?

Για τις ανάγκες των πειραμάτων χρησιμοποιήθηκαν:

- Φορητοί υπολογιστές με λογισμικό Windows 10, Ubuntu & Kali Linux (live mode & virtual machine)
- Η πλατφόρμα Android Studio και ο ενσωματωμένος virtual device/emulator
- Οι εφαρμογές:
  - apkDownloader
  - dex2jar
  - Java Decompiler (JD-GUI)
  - BurpSuite, OWASP ZAP, Wireshark & Charles Proxy
- Τα εργαλεία ανάλυσης κινητών εφαρμογών:
  - Qark
  - MobSF
  - Drozer
  - Broken Fingers
- Κινητό τηλέφωνο Xiaomi Redmi 5, με λογισμικό Android 7.1.2 Nougat
- Κινητό τηλέφωνο Samsung Galaxy S6
- Android Emulator Genymotion

Commented [Az2]: Select one

Οι εφαρμογές που επιλέξαμε να αναλύσουμε είναι διαθέσιμες στο Google Play Store και συνεπώς είναι **ανοιχτού κώδικα**. Στα πλαίσια της εργασίας μας, διαλέξαμε προφανώς εφαρμογές που να υλοποιούν την αυθεντικοποίηση με χρήση δακτυλικού αποτυπώματος. Για προφανείς νομικούς λόγους, τα ονόματα και οι πληροφορίες των εφαρμογών αυτών δε θα αναφερθούν. Αντί αυτού, επιλέχθηκαν οι ονομασίες «A», «E», «N», «W».

Για την εκτέλεση των πειραμάτων, κινηθήκαμε με οδηγό το γνωστό δεκάλογο ασφαλείας του OWASP, στον οποίο το πεδίο μελέτης μας υπάγεται στην μη ασφαλή αυθεντικοποίηση δηλαδή την παράγραφο M4.[32]

Επιπλέον, ακολουθήσαμε τις κατευθυντήριες οδηγίες του OWASP σχετικά με τη μελέτη και αποτίμηση ασφαλείας εφαρμογών, όπως φαίνεται στο σχήμα.

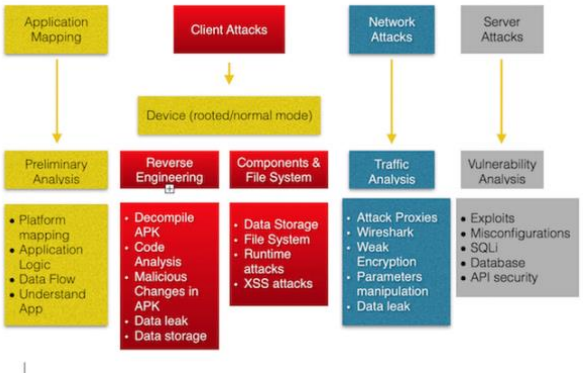


Figure 34 - Μεθοδολογία κατά OWASP

Θα επικεντρωθούμε στα πρώτα δύο σκέλη της μελέτης, δηλαδή στην πρακτική μελέτη εφαρμογής (application mapping) και στη μελέτη του κώδικα της (reverse engineering) όσον αφορά τη βιομετρική αυθεντικοποίηση που χρησιμοποιούν, και στη συνέχεια θα πραγματοποιηθούν client side network μελέτες.

3.2 Application Mapping

Προφανώς, για το πρώτο μέρος χρειάζεται η εγκατάσταση των εφαρμογών σε λειτουργικές κινητές συσκευές, και η πρακτική χρήση τους. Δε συνίσταται η χρήση emulator, μόνο για την ευχρηστία του αισθητήρα δακτυλικού αποτυπώματος. Σημειώνεται ότι καμία από τις συσκευές στο συγκεκριμένο πείραμα δεν είναι rooted σε αυτή τη φάση. Σκοπός είναι η εμπειρική κατανόηση της λειτουργίας, των διαδικασιών και της ροής δεδομένων της εφαρμογής. Οι εφαρμογές λήφθηκαν από το Google Play Store σε όλες τις κινητές συσκευές που χρησιμοποιήθηκαν, και ελέγχθηκαν για την αξιοπιστία τους αυτόματα κατά την εγκατάσταση από το λειτουργικό με βάση τις ψηφιακές υπογραφές των προγραμματιστών, σύμφωνα με το πρωτόκολλο της Google. Η διαδικασία αυτή είναι πλήρως αυτοματοποιημένη.

Στα πλαίσια αυτού του σταδίου της μελέτης, θα κατασκευάσουμε ένα πρωτογενές και απλό διάγραμμα ροής για να απεικονίσουμε τις εκτελούμενες διαδικασίες, όπως διακρίνονται από τη χρήση των εφαρμογών.

A mobile application

Sign Up – Εγγραφή

Κατά την εγγραφή του χρήστη στην εφαρμογή, του ζητούνται διαπιστευτήρια σχετικά με τον τραπεζικό του λογαριασμό και την πιστωτική του κάρτα: ημερομηνία γέννησης του cardholder, ο αριθμός της κάρτας και δύο τυχαία ψηφία του PIN. Συνεπώς, περιορίζεται η χρήση της συσκευής σε

άτομα που δε σχετίζονται με την Τράπεζα, εφόσον δε μπορούν να προχωρήσουν σε άλλες λειτουργίες της εφαρμογής.

Καθώς η Τράπεζα αναπτύσσει την πλατφόρμα e-banking της, ο χρήστης μπορεί και να συνδεθεί με τα στοιχεία e-banking της Τραπέζης και απαιτείται μόνο το e-mail του και ο κωδικός που του έχει παραχωρηθεί από την Τράπεζα. Καθώς ο χρήστης έχει ήδη πραγματοποιήσει εγγραφή στο σύστημα με τη φυσική του παρουσία σε κατάστημα της Τραπέζης παρέχοντας τα στοιχεία του και πιστοποιώντας τα με την αστυνομική του ταυτότητα, η ίδια η εφαρμογή πρέπει απλώς να ταυτοποιήσει το χρήστη στο σύστημα e-banking και να αυθεντικοποιηθεί. Αυτό επιτυγχάνεται με παροχή των κωδικών που του δόθηκαν στο κατάστημα (το προσωρινό password που ανατίθεται ισχύει μόνο για την πρώτη είσοδο του χρήστη, που καλείται να το αλλάξει αμέσως μετά) και αυθεντικοποίηση δύο παραγόντων με αποστολή one time pass στο κινητό τηλέφωνο που καταχώρησε ο χρήστης κατά την εγγραφή. Στη συνέχεια, η εφαρμογή ζητά από το χρήστη να τοποθετήσει το δακτυλικό του αποτύπωμα στον αισθητήρα, και αποθηκεύει τη βιομετρική πληροφορία για μελλοντική αυθεντικοποίηση. Δε φαίνεται στο χρήστη εάν η πληροφορία αποθηκεύεται τοπικά ή απομακρυσμένα.

Commented [AZ3]: Hide info

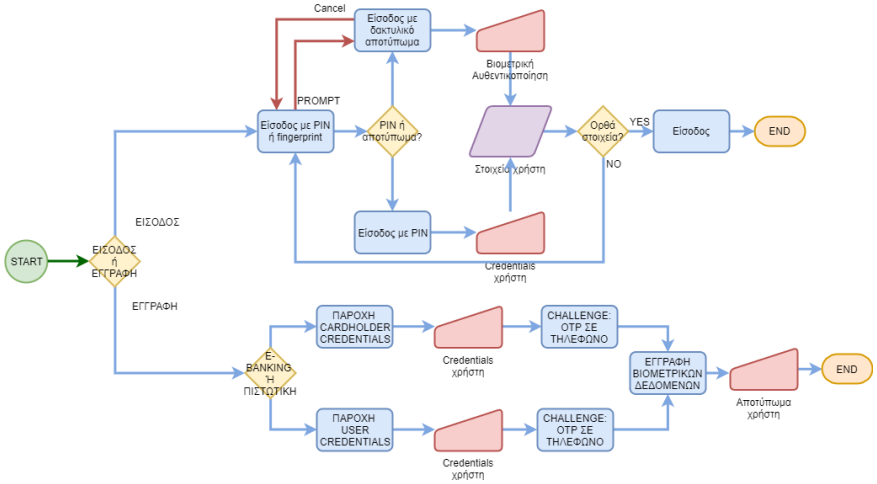


Figure 35 – A app: Preliminary mapping

Login – Είσοδος

Η είσοδος στην εφαρμογή Α γίνεται ταχύτατα και πολύ απλά, με την εφαρμογή κατά το άνοιγμα της να δίνει ως αρχική οθόνη τις επιλογές εισόδου του χρήστη (Fingerprint και κωδικούς), ενώ σχεδόν αμέσως εμφανίζεται ένα prompt που ζητά από το χρήστη να χρησιμοποιήσει το δακτυλικό του αποτύπωμα, όπως φαίνεται στην εικόνα. Η διαδικασία γίνεται γρήγορα, απλά και φιλικά προς το χρήστη, και δε φαίνονται λεπτομέρειες ως προς τις λειτουργικές διαδικασίες της, που να μπορεί

κάποιος να εκμεταλλευτεί. Πρόκειται δηλαδή για black box διαδικασία, που δε διαφαίνεται στο χρήστη η ροή των δεδομένων και οι εκτελούμενες λειτουργίες.

Commented [AZ4]: Hide info

E mobile application

Sign Up – Εγγραφή

Κατά την εγγραφή ενός χρήστη στην εφαρμογή, ζητά τα στοιχεία e-banking του χρήστη, τα οποία έχει αποκτήσει απευθείας από την τράπεζα για να δημιουργήσει λογαριασμό με στοιχεία (username, password) της επιλογής του τα οποία και καλείται να παρέχει στην εφαρμογή. Εν συνεχεία αποστέλλεται one time pass στο τηλέφωνο του (τον αριθμό του οποίου πρέπει να επιβεβαιώσει) με το οποίο πιστοποιεί την ταυτότητα του. Μετά τα παραπάνω, πραγματοποιεί την πρώτη του σύνδεση (login) στην εφαρμογή από τη συσκευή του και μετά εισάγει το δακτυλικό του αποτύπωμα για μελλοντικές συνδέσεις. Και πάλι δε διαφαίνονται λεπτομέρειες για τη διαδικασία.

Login – Είσοδος

Αντίστοιχα με την εφαρμογή Α, και σε αυτή την περίπτωση δεν υπάρχουν πολλά που να διαφαίνονται κατά την είσοδο του χρήστη. Από τη στιγμή που έχει ενεργοποιήσει την είσοδο με δακτυλικό αποτύπωμα κατά την εγγραφή του ή όποτε το επιθυμεί, η εφαρμογή κατά το άνοιγμα ζητά αυθεντικοποίηση χρήστη με 4-ψήφιο PIN αυτόματα αλλά στην περίπτωση βιομετρικής αυθεντικοποίησης και πάλι εμφανίζεται σχετικό prompt που ζητά από το χρήστη επιβεβαίωση μέσω δακτυλικού αποτυπώματος. Η όλη διαδικασία γίνεται ταχύτατα και εύκολα για το χρήστη, χωρίς να χρειάζεται να θυμάται κωδικούς και στοιχεία για να πραγματοποιήσει είσοδο στην εφαρμογή.

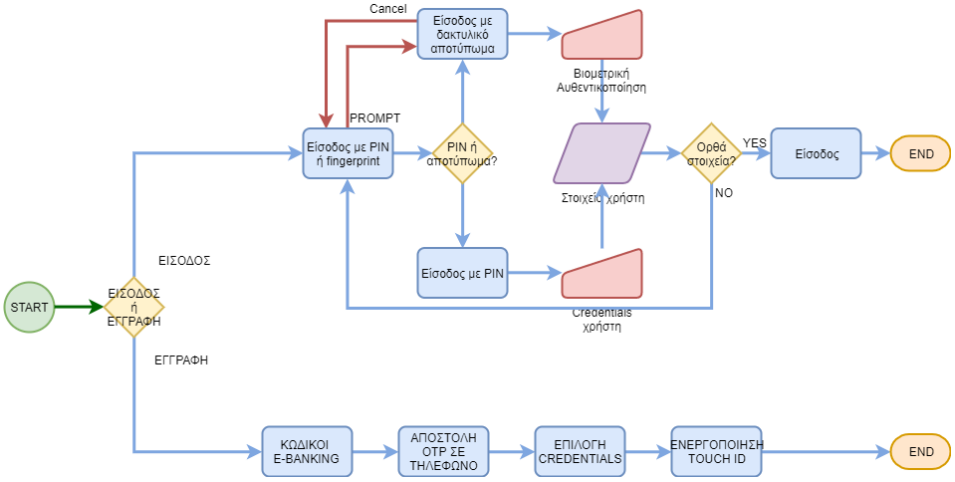


Figure 36 - E - Preliminary mapping

N mobile application

Sign Up – Εγγραφή

Κατά την εγγραφή στην εφαρμογή, ζητούνται τα διαπιστευτήρια του χρήστη (username, password) τα οποία του έχει ήδη παραχωρήσει η τράπεζα και έχει επεξεργαστεί ο ίδιος κατά το δοκούν. Μετά την σύνδεση με τα διαπιστευτήρια, ζητείται από το χρήστη να εγγραφεί ως νέος χρήστης στην εφαρμογή του κινητού του τηλεφώνου, για να έχει πρόσβαση σε όλες τις λειτουργίες της. Για να εγγραφεί η συσκευή του, πρέπει να πιστοποιήσει και πάλι την ταυτότητα του. Αυτό επιτυγχάνετια με αυθεντικοποίηση 2 παραγόντων (2-factor authentication), όπου ο χρήστης επιλέγει να αποσταλεί στο κινητό του τηλέφωνο κωδικός OTP τον οποίο και εισάγει στην εφαρμογή, και έχει πλέον πλήρη πρόσβαση.

Για να προσθέσει μεθόδους ταχείας εισόδου, ο χρήστης θα πρέπει να αναζητήσει τη σχετική επιλογή στο πλαϊνό μενού της εφαρμογής, στην επιλογή “Fast Login”. Από εκεί του δίνονται τρεις επιλογές, η αλλαγή του password και η ενεργοποίηση ταχείας εισόδου είτε με κωδικό PIN είτε με χρήση δακτυλικού αποτυπώματος. Επιλέγοντας το δεύτερο, εμφανίζεται το σχετικό prompt για να τοποθετήσει το αποτύπωμα του ο χρήστης στον αισθητήρα και μόλις ληφθεί το δείγμα, επιβεβαιώνεται η διαδικασία ως ολοκληρωμένη.

Η διαδικασία είναι απλή και δομημένη, και δεν αφήνει να φανούν λεπτομέρειες αποθήκευσης και διαχείρισης δεδομένων στο χρήστη. Η επιλογή της βιομετρικής αυθεντικοποίησης δεν προωθείται ιδιαίτερα από το ίδιο το UI.

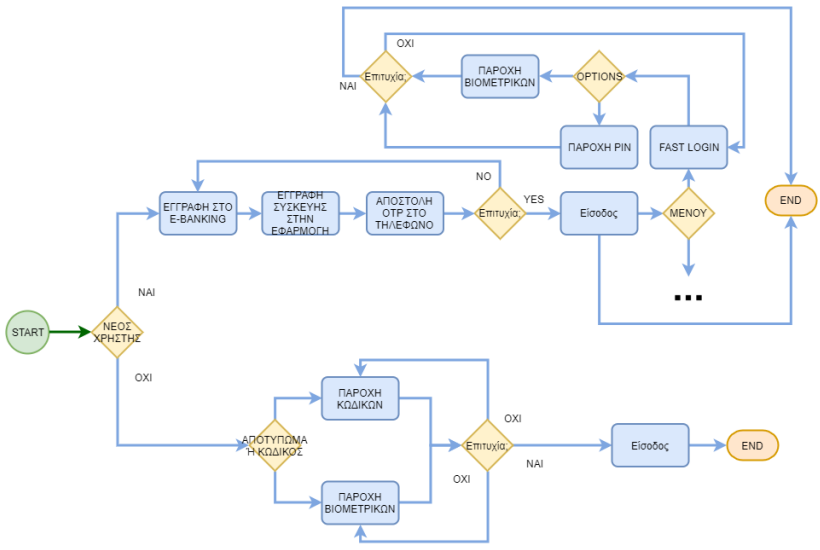


Figure 37 - N - Preliminary mapping

Login – Είσοδος

Σε αντίθεση με τα δύο προηγούμενα παραδείγματα εισόδου, η εφαρμογή Ν δεν επιλέγει αυτόματα τη βιομετρική αυθεντικοποίηση με σχετικό prompt, αλλά δίνει στο χρήστη την επιλογή της μεθόδου αυθεντικοποίησης κατά την είσοδο. Ο χρήστης επιλέγει μεταξύ αυθεντικοποίησης με δακτυλικό αποτύπωμα και αυθεντικοποίησης με μυστικό κωδικό. Στην πρώτη περίπτωση, εμφανίζεται το σχετικό prompt, δείχνοντας στο χρήστη ότι η εφαρμογή είναι έτοιμη να δεχθεί το δακτυλικό του αποτύπωμα και επιβεβαιώνοντας του την επιτυχημένη ή αποτυχημένη σύνδεση του.

W mobile application

Παρατηρήθηκε ότι, για λόγους ασφαλείας, η εφαρμογή σε γενικές γραμμές δεν επιτρέπει τη λήψη στιγμιότυπων οθόνης (screenshots) αλλά ο χρήστης μπορεί να απενεργοποιήσει αυτή τη ρύθμιση. Αντίστοιχη ρύθμιση δεν εντοπίστηκε στις άλλες εφαρμογές.

Sign Up – Εγγραφή

Η εφαρμογή παρακινεί το χρήστη να εισέλθει στο σύστημα με τα στοιχεία e-banking του και να ορίσει ένα PIN για μελλοντικές εισόδους. Δεν δίνεται κάποιο prompt αυτομάτως στο χρήστη για επιλογή βιομετρικής αυθεντικο-ποίησης, καθώς πρέπει να ενεργοποιήσει την επιλογή μέσω του μενού, οπότε και η εφαρμογή του ζητά με σχετικό prompt να εισάγει τα στοιχεία του (PIN για επιβεβαίωση και το δακτυλικό του αποτύπωμα στον αισθητήρα).

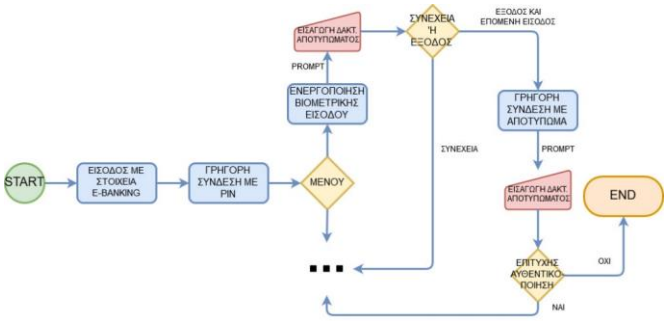


Figure 38 - W - Preliminary Mapping

Login – Είσοδος

Ο αυτόματος τρόπος εισόδου πριν την ενεργοποίηση της εισόδου με αποτύπωμα, είναι με εισαγωγή τετραψήφιου PIN. Αφού ενεργοποιήσει την επιλογή γρήγορης εισόδου ο χρήστης, κατά το άνοιγμα της εφαρμογής θα του ζητείται κάθε φορά με prompt να δώσει το δακτυλικό του αποτύπωμα για είσοδο. Βάσει νομικού πλαισίου, περιοδικά του ζητείται το επιλεγμένο PIN.

Και σε αυτήν την περίπτωση, η διαδικασία είναι ευθεία και δεν προδίδει κάποια τεχνική λεπτομέρεια για τη βιομετρική αυθεντικοποίηση που πραγματοποιείται (πχ. αν η επαλήθευση γίνεται τοπικά ή στέλνονται στοιχεία στο δίκτυο, αν υπάρχει κρυπτογράφηση και τι είδους κ.ο.κ.).

3.3 Reverse Engineering – Source Code Analysis

Η πρωτοβάθμια ανάλυση των εφαρμογών μας βοηθά να κατανοήσουμε τον κώδικα τους και να εντοπίσουμε πού πρέπει να επικεντρωθούμε στη μελέτη μας. Για το δεύτερο μέρος των πειραμάτων, οι εφαρμογές κατεβάστηκαν ως αρχεία APK μέσω **apkDownloader** [2] τα οποία και χρησιμοποιήθηκαν για τα πειράματα μας σε υπολογιστή, εκτός της εγκατάστασής τους σε κινητές συσκευές. Το εργαλείο αυτό επιτρέπει τη λήψη αρχείων apk στον υπολογιστή μας απευθείας από το δίκτυο, μέσω Google Play. Το μόνο που απαιτείται είναι να δοθεί ορθά το όνομα του πακέτου της εφαρμογής (που έχει τη μορφή com.example.appname) ή ο σύνδεσμος στο κατάστημα Google Play.

Figure 39 - apkDownloader

Στη συνέχεια το πρώτο στάδιο για την ανάλυση του κώδικα είναι η μελέτη της δομής της εφαρμογής. Αυτή η μελέτη μπορεί να πραγματοποιηθεί με πολλούς τρόπους, όπως η χρήση εφαρμογών όπως το **Dexplorer**, που εγκαθίσταται σε κινητή συσκευή ή emulator και δείχνει τη βασική δομή κάθε αρχείου apk, δηλαδή τα αρχεία `.dex` (των κλάσεων του) τα resources και φυσικά το αρχείο `AndroidManifest.xml`, απαραίτητο σε κάθε εφαρμογή και ενδεικτικό της δομής του. Στο αρχείο αυτό επίσης παρουσιάζονται και πολλές λειτουργικές πληροφορίες της εφαρμογής όπως τα permissions που ορίζονται από τον προγραμματιστή, δηλαδή οι λειτουργίες της συσκευής στις οποίες έχει πρόσβαση η εφαρμογή.

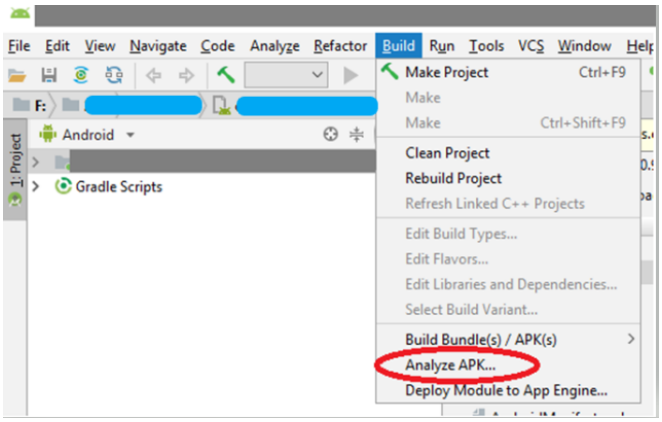


Figure 40 - Analyze APK

[2] Προσπέλαση στο: https://apps.evozi.com/apk-downloader/

Ένας ακόμα τρόπος να πραγματοποιήσουμε αυτή την ανάλυση είναι με μια νέα λειτουργία του **Android Studio**, επιλέγοντας `Build > Analyze APK`. Η πλατφόρμα μας εμφανίζει την αναλυτική δομή του APK και μας παρουσιάζει πληροφορίες για τη μνήμη, τα μεγέθη των αρχείων, τον αριθμό των java classes που περιλαμβάνονται σε κάθε αρχείο `.dex` και φυσικά μπορεί να προβληθεί το αρχείο `AndroidManifest.xml`. Επιπλέον, η πλατφόρμα πραγματοποιεί συγκρίσεις μεταξύ εφαρμογών. Αυτό το χαρακτηριστικό είναι πολύ χρήσιμο για native Android developers καθώς βοηθά στη βελτιστοποίηση διαφόρων χαρακτηριστικών της εφαρμογής κατά τη διάρκεια και μετά το πέρας της ανάπτυξης της. Στη συνέχεια παρατίθενται τα αποτελέσματα της πρωταρχικής ανάλυσης.

Commented [AZ5]: Search for FIDO

Ανάλυση Android Manifest – Fingerprint API

Η σημασία του συγκεκριμένου αρχείου είναι μεγάλη τόσο για έναν Android Developer όσο και για έναν security analyst, καθώς το συγκεκριμένο αρχείο είναι αναγκαίο για κάθε εφαρμογή καθώς δίνει τις απαραίτητες «οδηγίες» στο λειτουργικό σύστημα για να «χτίσει» την εφαρμογή κατά την εγκατάσταση της από συμπιεσμένο αρχείο `.apk` αλλά προσφέρει επίσης πολλές πληροφορίες για θέματα ασφαλείας, ειδικά για τα permissions. Third-party εφαρμογές που εκτελούνται με απομονωμένα resources (όπως private αρχεία) έχουν περιορισμούς στις δυνατότητες τους, οι οποίες ορίζονται από τα σχετικά permissions τα οποία καθορίζονται στο `AndroidManifest.xml` και από το Android 6 και εξής κατηγοριοποιούνται, ενώ ζητείται από το χρήστη παραχώρηση άδειας σε τέτοιες εφαρμογές να αξιοποιήσουν στοιχεία του κινητού και του λειτουργικού, αν το σχετικό permission χαρακτηρίζεται ως “dangerous”. Το permission “use_fingerprint” κατηγοριοποιείται ως “normal”.

Commented [AZ6]: Search for FIDO

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:versionCode="58"
    android:versionName="3.0.9"
    package="[redacted]"
    platformBuildVersionCode="27"
    platformBuildVersionName="8.1.0">

    <uses-sdk
        android:minSdkVersion="21"
        android:targetSdkVersion="27" />

    <uses-permission
        android:name="android.permission.INTERNET" />

    <uses-permission
        android:name="android.permission.USE_FINGERPRINT" />

    <uses-permission
        android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
```

Figure 41 - E AndroidManifest

Επιπλέον, το αρχείο `AndroidManifest.xml` μπορεί να μας παρέχει πολύτιμες πληροφορίες για το implementation των διαφόρων διαδικασιών, όπως η βιομετρική αυθεντικοποίηση. Αρκεί να αναζητήσουμε στο αρχείο στοχευμένες λέξεις-κλειδιά όπως “fingerprint” και να εντοπίσουμε τα σχετικά permissions. Στο συγκεκριμένο πείραμα θα αναζητήσουμε τη λέξη-κλειδί “FIDO”, για να δούμε αν κάποια εκ των εφαρμογών ενσωματώνει στη λειτουργία της το σχετικό API.

Commented [AZ7]: Search for FIDO

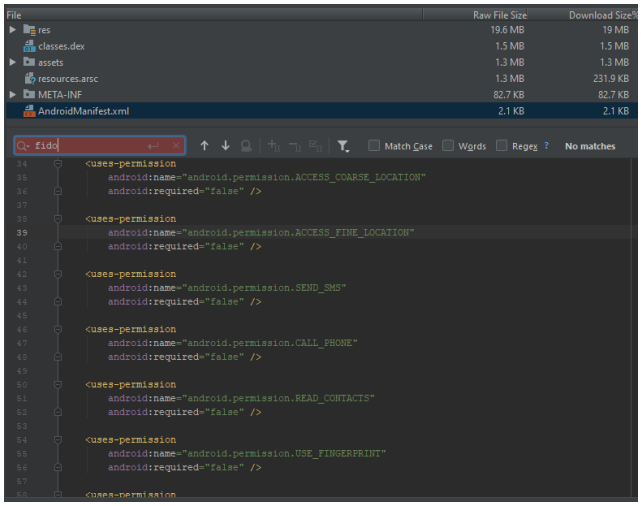


Figure 42 - FIDO στην εφαρμογή A

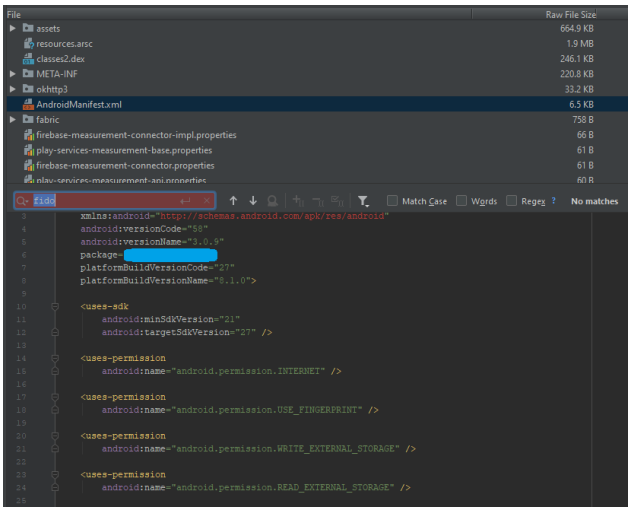


Figure 43 - FIDO στην εφαρμογή E

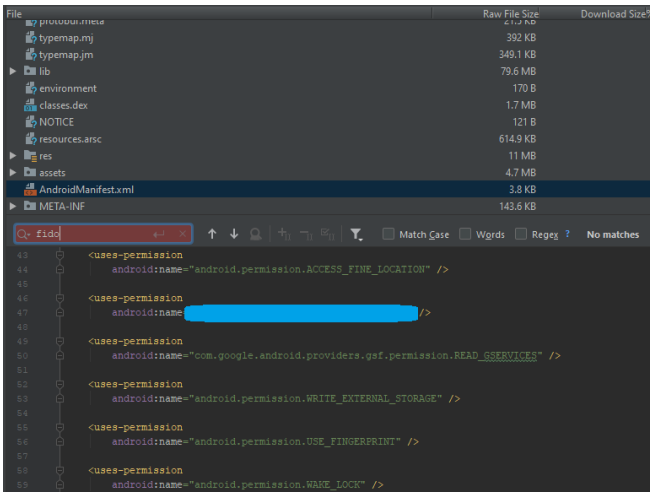


Figure 44 - FIDO στην εφαρμογή N

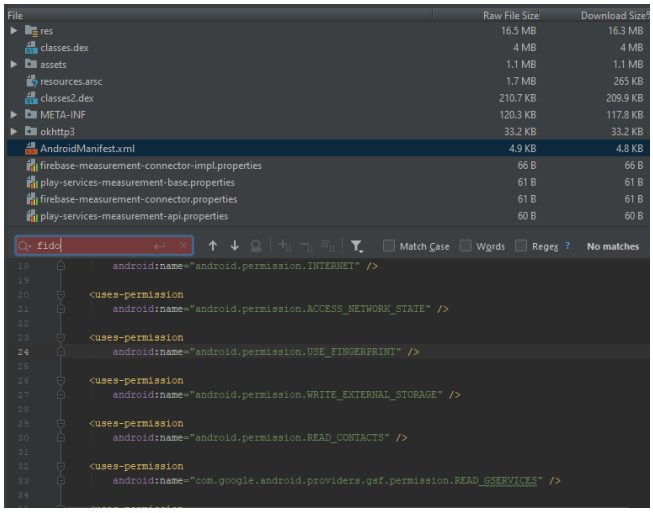


Figure 45 - FIDO στην εφαρμογή W

Αμέσως διαπιστώνουμε πως καμία από τις υπό μελέτη εφαρμογές δε συμπεριλαμβάνει και δεν εφαρμόζει το FIDO Android API. Ενδιαφέρον εμφανίζει ωστόσο η μελέτη της εφαρμογής του Android fingerprint API σε εφαρμογές, ειδικά εφόσον διαχειρίζονται ευαίσθητα προσωπικά δεδομένα, όπως αυτές που μελετώνται. Κατά Bianchi et al.[5] και μετά από συστηματική ανάλυση του συγκεκριμένου API, διαπιστώθηκε πως σε εξαιρετικά υψηλό ποσοστό οι Android Developers δεν κατανοούν ούτε

Commented [Az8]: The results are worrisome: Our tool indicates that 53.69% of theanalyzed apps do not use any cryptographic check to ensure thatthe user actually touched the fingerprint sensor. Depending on thespecific use case scenario of a given app, it is not always possible tomake use of cryptographic checks. However, a manual investigationon a subset of these apps revealed that 80% of them could have doneso, preventing multiple attacks. Furthermore, the tool indicates thatonly the 1.80% of the analyzed apps use this API in the most secureway possible, while many others, including extremely popular appssuch as Google Play Store and Square Cash, use it in weaker ways.To make things worse, we find issues and inconsistencies even in thesamples provided by the official Google documentation. We end thiswork by suggesting various improvements to the fingerprint API toprevent some of these problematic attacks.

εφαρμόζουν καλή πρακτική στην ενσωμάτωση του στις εφαρμογές τους. Συγκεκριμένα, η εν λόγω μελέτη αναλύει τη λειτουργία τόσο του μηχανισμού του αισθητήρα δακτυλικού αποτυπώματος όσο και της εφαρμογής του συστήματος ως μέρους two-factor authentication. Τα σύγχρονα smartphones εμπεριέχουν Trusted Execution Environment (TEE) τα οποία παράγουν και αποθηκεύουν κρυπτογραφικά κλειδιά. Το σύστημα αναγνώρισης δακτυλικού αποτυπώματος επικοινωνεί με το TEE απευθείας, και καθώς πρόκειται για απομονωμένο περιβάλλον εκτέλεσης βάσει υλικού («νησίδα») δεν μπορούν τα κλειδιά αυτά να κλαπούν, να διαρρεύσουν ή να χρησιμοποιηθούν ακόμα και σε περίπτωση έκθεσης, διακινδύνευσης ή και αλλοίωσης του λειτουργικού συστήματος του τηλεφώνου, πχ. Root attacker.

Με τη λειτουργία του Android Studio, “Analyze APK”, μελετήσαμε το περίγραμμα της υλοποίησης της λειτουργίας του δακτυλικού αποτυπώματος σε κάθε μία από τις εφαρμογές μας, δίνοντας βάση στις σχετικές κλάσεις Java.

E mobile app

Η εφαρμογή E περιέχει τις εξής κλάσεις (με μπλε χρώμα) που περιέχουν τις σχετικές μεθόδους (με κόκκινο χρώμα). Όπως διαπιστώσαμε, όλες αυτές οι κλάσεις δε γίνονται διαθέσιμες και φανερές μετά από reverse engineering. Αυτό συνάδει με τη θεωρία που αναλύσαμε και μελετήσαμε σχετικά με τη χρήση TEE και αισθητήρα δακτυλικού αποτυπώματος. Οι κλάσεις αυτές συνεπώς εκτελούνται εντός του TrustZone, του TEE των συσκευών Android που υλοποιείται μέσω ARM.

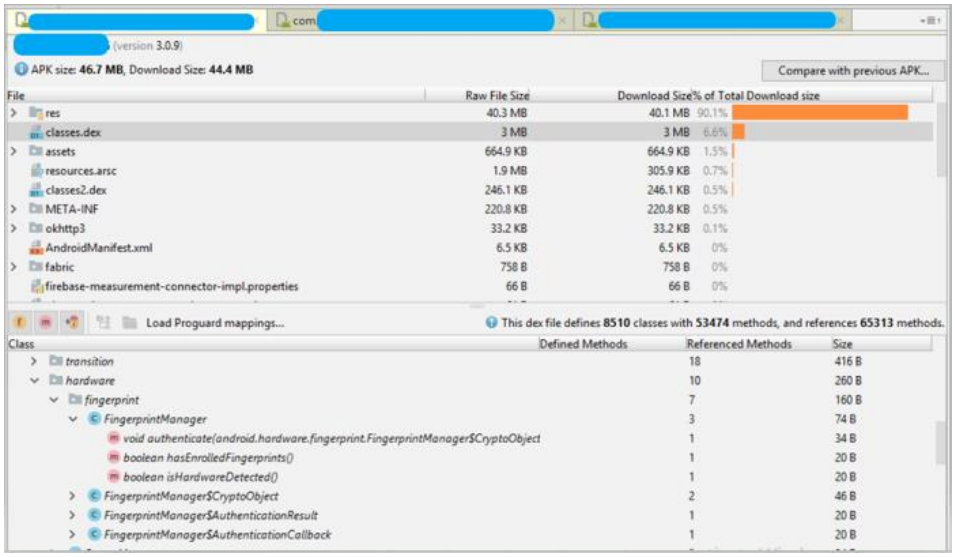


Figure 46 - E - Analyze APK (Android Studio)

A mobile app

Η εφαρμογή Α περιέχει πολλές μεθόδους για τη λειτουργία του αποτυπώματος, όπως φαίνεται από την ανάλυση. Σε σύγκριση με τις 4 κλάσεις και τις μόλις 7 μεθόδους java στην προηγούμενη εφαρμογή που αναλύσαμε, εδώ παρατηρούνται 148 μέθοδοι σε 37 κλάσεις. Είναι πιθανόν οι developers να μην έκαναν import έτοιμων μεθόδων αυθεντικοποίησης με δακτυλικό αποτύπωμα αλλά να ανέπτυξαν εξ αρχής κάποιο μοντέλο ή να προσάρμοσαν κάποιο προϋπάρχον στις ανάγκες της εφαρμογής, να προσέθεσαν πιθανά exceptions ή άλλες λειτουργίες. Παρατηρείται επίσης η ύπαρξη πολλών adaptors, handlers και διαλόγων. Οι επιμέρους ονομασίες δεν αποκρύπτουν τις λειτουργίες, αλλά και δεν αποκαλύπτουν πολλά.

Σε γενικές γραμμές, η αρχιτεκτονική της εφαρμογής αυτής και ειδικότερα της βιομετρικής αυθεντικοποίησης δείχνει να είναι η πιο πολύπλοκη αλλά και η πιο αναλυτική εκ των τεσσάρων υπό ανάλυση εφαρμογών, καθώς αποτελείται από πολλές κλάσεις και μεθόδους για μια φαινομενικά μικρή ομάδα λειτουργιών. Η αποτελεσματικότητα και η αποδοτικότητα της επιλογής αυτής, τόσο σε λειτουργικότητα όσο και σε ασφάλεια, κατά την ανάπτυξη της εφαρμογής θα φανεί σε περαιτέρω ανάλυση.

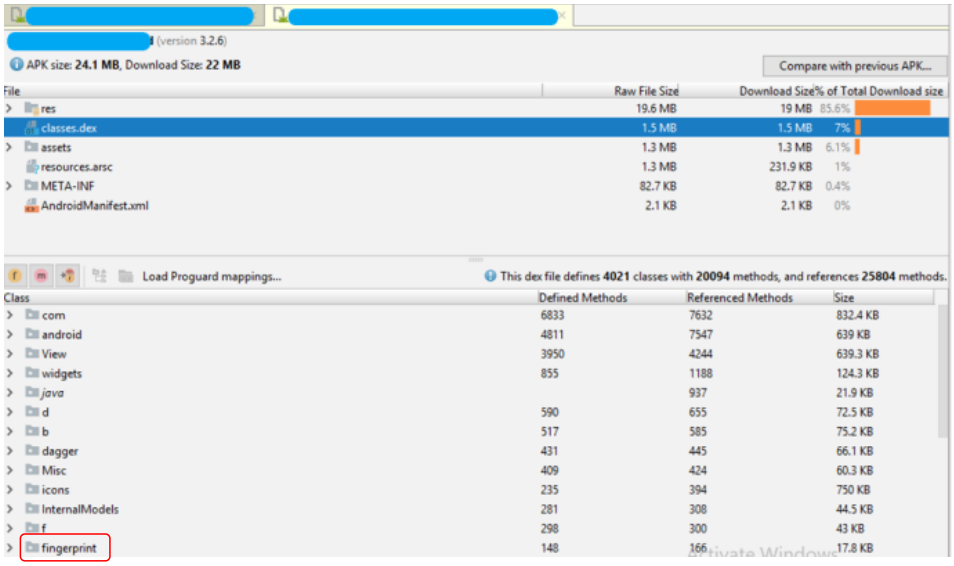


Figure 47 - A - Classes outline

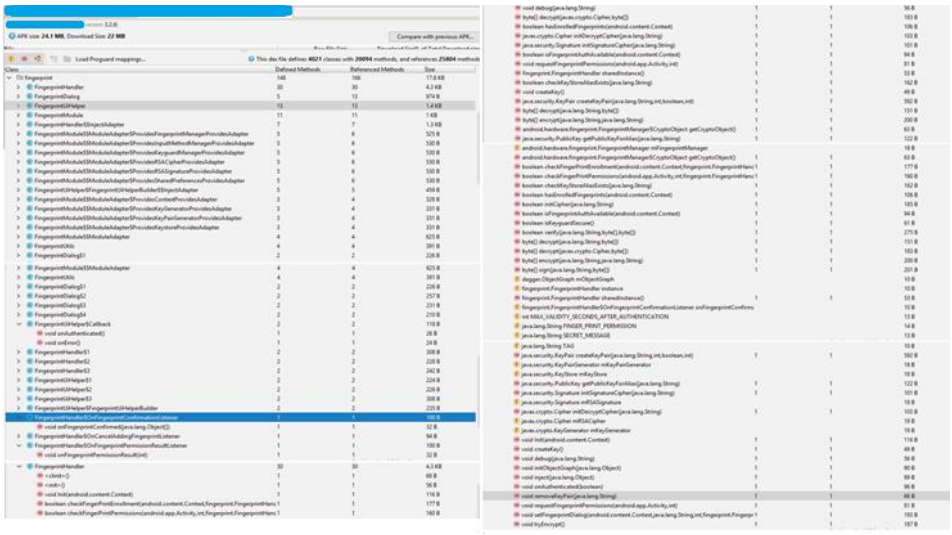


Figure 48 - A - Fingerprint authentication classes

N mobile app

Στις εφαρμογή αυτή παρατηρούμε ότι οι developers πραγματοποίησαν ένα μερικό obfuscation στους φακέλους και στις κλάσεις των μεθόδων για διάφορες λειτουργίες της εφαρμογής, όχι όμως σε ότι αφορά τη βιομετρική αυθεντικοποίηση (πιθανόν επειδή χρησιμοποιήθηκαν προϋπάρχουσες μέθοδοι). Παρατηρείται στην αρχιτεκτονική και η κρυπτογράφηση των δεδομένων ως διαδικασία, χωρίς να φαίνονται άλλες πληροφορίες. Αντίστοιχα, παρατηρείται το ίδιο στην τέταρτη εφαρμογή.

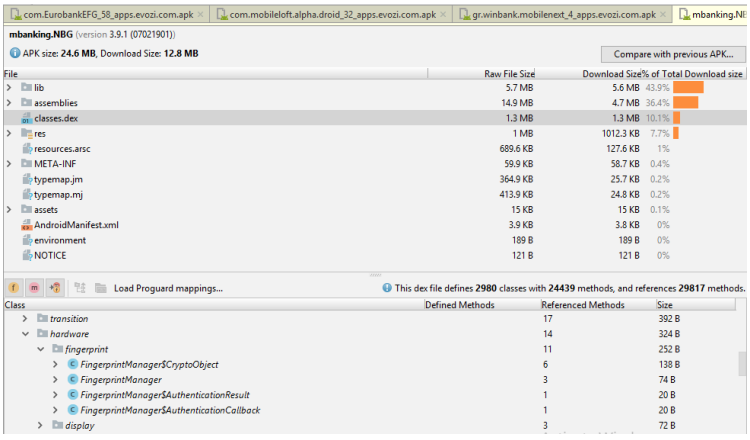


Figure 49 - N - APK Analysis

W mobile app

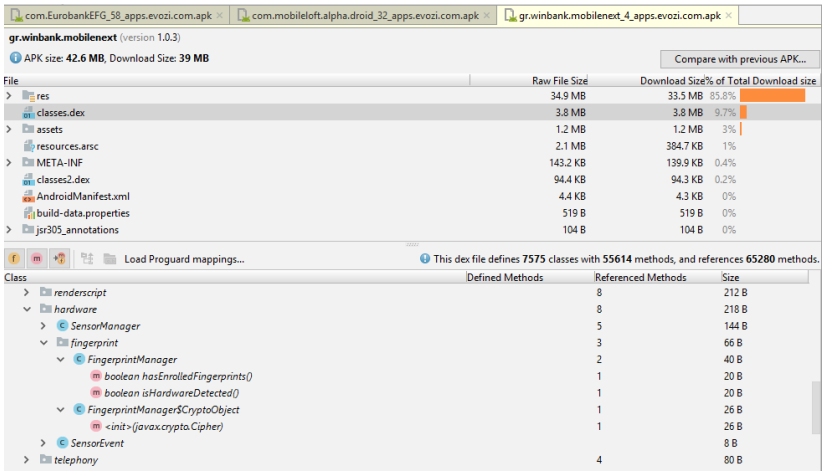


Figure 50 – W - APK analysis

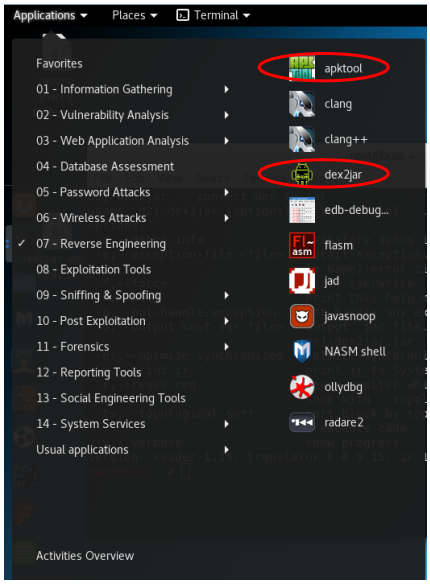


Figure 49 - dex2jar Application

Τέλος, μπορούμε να μελετήσουμε αναλυτικά τη δομή μιας εφαρμογής αφού πραγματοποιήσουμε reverse engineering στο αρχείο της. Για την εξαγωγή του κώδικα από το αρχείο `.apk` (που αποτελεί μορφή συμπίεσης), απλώς **μετονομάζουμε** το αρχείο της εφαρμογής από επέκταση `.apk` σε `.zip` και το **αποσυμπιέζουμε**. Με τον τρόπο αυτό μας εμφανίζονται όλα τα περιεχόμενα του κώδικα της εφαρμογής, δηλαδή τα αρχεία XML και Java ή Kotlin. Η αποσυμπίεση αρχείων XML πραγματοποιείται με το εργαλειο `apkTool`. Για την παρούσα εργασία θα επικεντρωθούμε στο λειτουργικό κώδικα δηλαδή στο κομμάτι της Java. Το σημαντικότερο βήμα στην εξαγωγή των αρχείων APK είναι η αποσυμπίεση του αρχείου DEX που περιέχει όλες τις κλάσεις της εφαρμογής, δηλαδή τον πηγαίο κώδικα σε Java ή Kotlin που περιέχει όλες τις λειτουργίες της. Η εξαγωγή των αρχείων γίνεται σε δύο φάσεις. Κατά την πρώτη, μεταφέρουμε το αρχείο .dex της εφαρμογής στο φάκελο του εργαλείου **dex2jar**[30] (προεγκατεστημένο στα Kali Linux, στην κατηγορία reverse engineering) και με την παρακάτω εντολή

(*figure 50*) εξάγονται οι κλάσεις του κώδικα σε μορφή JAR. Στη συνέχεια το αρχείο `classes_dex2jar.jar` που δημιουργείται το ανοίγουμε με το πρόγραμμα Java Decompiler και επιλέγουμε File > Save all sources, οπότε ο κώδικας αποθηκεύεται σε φάκελο `src`.

```
root@kali:~# d2j-dex2jar -d /root/Desktop[redacted]/classes.dex
dex2jar /root/Desktop/[redacted]/classes.dex -> classes-dex2jar.jar
root@kali:~#
```

Figure 50 - dex2jar command

Στο συγκεκριμένο κομμάτι θα εξάγουμε και θα αναλύσουμε τον κώδικα των υπό μελέτη εφαρμογών. Για τις ανάγκες τις εργασίας δε θα επικεντρωθούμε τόσο στον κώδικα XML αλλά μόνο στο προγραμματιστικό μέρος σε Java/Kotlin.

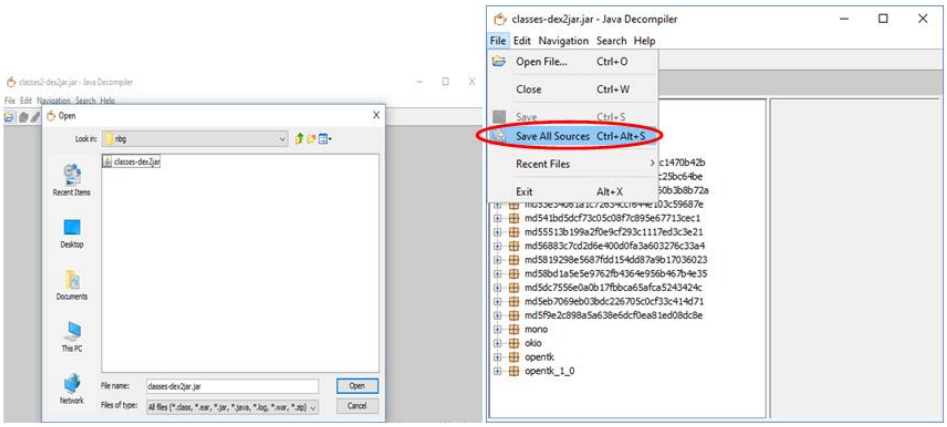


Figure 51 - Εξαγωγή πηγαίου κώδικα

Σε ένα πρώτο βήμα ανάλυσης του κώδικα, πραγματοποιήσαμε αναζήτηση των αρχείων Java για λέξεις κλειδιά όπως: *biometric, fingerprint, sensor, authentication*. Φαίνεται πως οι προγραμματιστές επέλεξαν να μην ονομάσουν τις μεθόδους του κώδικα που εκτελούν τις παραπάνω λειτουργίες με τέτοιες λέξεις κλειδιά για λόγους ασφαλείας και άμυνα σε πιθανό κακόβουλο λογισμικό (**obfuscation**). Από την ανάλυση όμως του κώδικα στο Android Studio βλέπουμε ότι υπάρχουν πολλές κλάσεις και μέθοδοι με τις συγκεκριμένες λέξεις κλειδιά, καθώς και σε ποιο φάκελο ανήκουν. Θα αναζητήσουμε αναλυτικότερα στον κώδικα που αποσυμπιέσαμε το

πώς υλοποιήθηκαν οι συγκεκριμένες λειτουργίες και θα προσπαθήσουμε να βγάλουμε συμπεράσματα για την ασφάλεια τους.

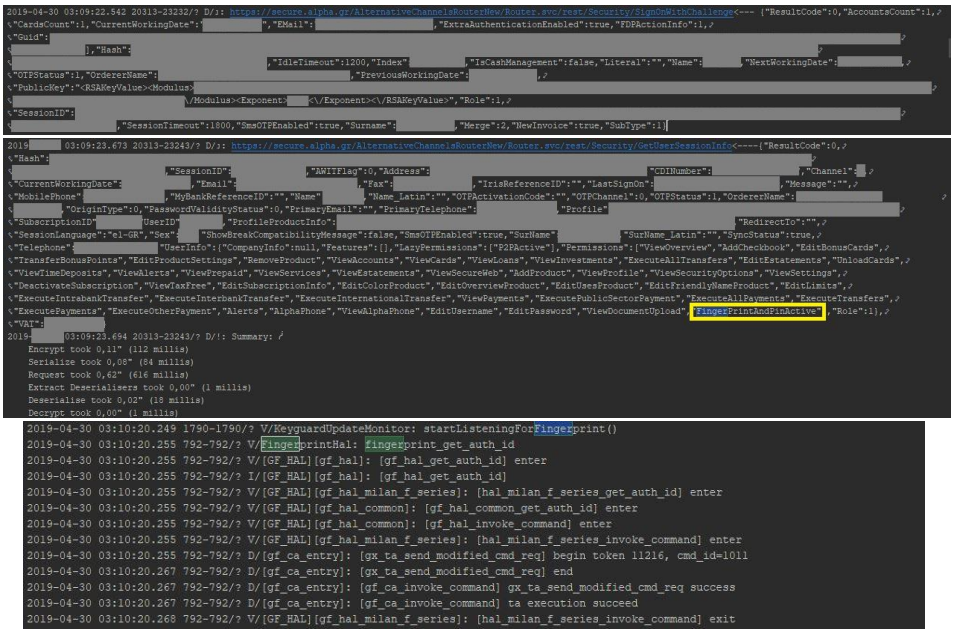
Intent Sniffing

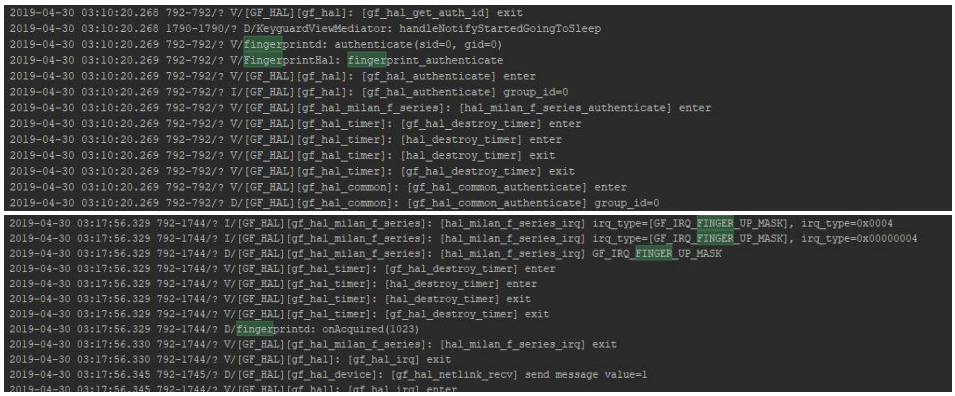
Για να κάνουμε ακόμα πιο συγκεκριμένη την αναζήτηση μας (λόγω του μεγάλου πλήθους των κλάσεων), στο επόμενο βήμα της ανάλυσης θα μελετήσουμε την εκτέλεση της εφαρμογής συνδέοντας τη συσκευή μας (ή σε άλλη περίπτωση, τον emulator) με το Android Studio και πιο συγκεκριμένα την κονσόλα **logcat**. Ο σκοπός μας είναι να πραγματοποιήσουμε **intent sniffing**.

A mobile app

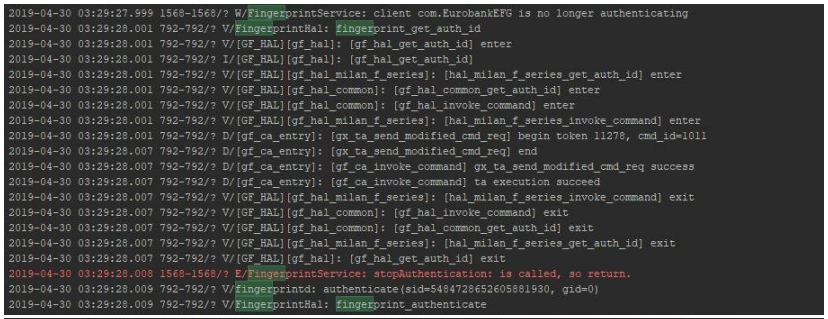
Η εφαρμογή αυτή ζητά κατά καιρούς να ξαναδώσει ο χρήστης τους κωδικούς αυθεντικοποίησης του για λόγους ασφαλείας. Παρατηρήσαμε, όταν μας ζητήθηκε αυτό, ότι στην πλατφόρμα εμφανίζονται τα σχετικά tags και πολλές πληροφορίες για το χρήστη (καλείται το `security/signOnWithChallenge`) οι οποίες φαίνεται να λαμβάνονται από κανάλι επικοινωνίας με το server.

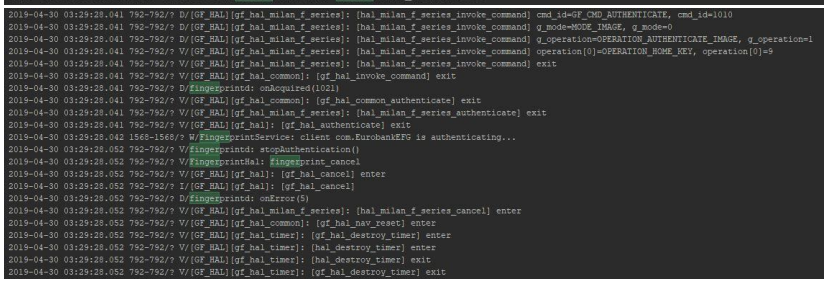
Παρατηρείται επίσης η δήλωση των permissions της εφαρμογής για τις εσωτερικές λειτουργίες της, μία εξ αυτών φαίνεται να είναι και η αυθεντικοποίηση με δακτυλικό αποτύπωμα (`FingerPrintPinAuthentication`) όπως βλέπουμε στις εικόνες.

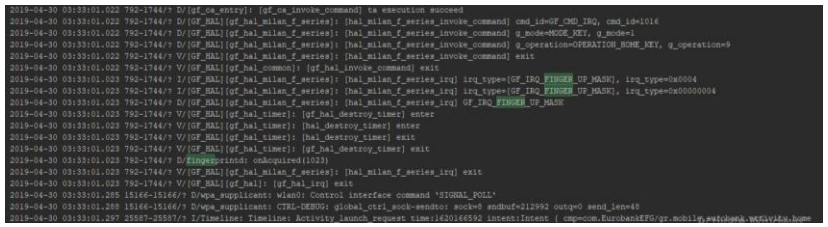



2019-04-30 03:10:20.268 792-792/? V/[GF_HAL][gf_hal]: [gf_hal_get_auth_id] exit
2019-04-30 03:10:20.268 1790-1790/? D/KeyguardViewMediator: handleNotifyStartedGoingToSleep
2019-04-30 03:10:20.269 792-792/? V/fingerprintd: authenticate(sid=0, gid=0)
2019-04-30 03:10:20.269 792-792/? V/FingerprintHal: fingerprint_authenticate
2019-04-30 03:10:20.269 792-792/? V/[GF_HAL][gf_hal]: [gf_hal_authenticate] enter
2019-04-30 03:10:20.269 792-792/? I/[GF_HAL][gf_hal]: [gf_hal_authenticate] group_id=0
2019-04-30 03:10:20.269 792-792/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_authenticate] enter
2019-04-30 03:10:20.269 792-792/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] enter
2019-04-30 03:10:20.269 792-792/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] enter
2019-04-30 03:10:20.269 792-792/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] exit
2019-04-30 03:10:20.269 792-792/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] exit
2019-04-30 03:10:20.269 792-792/? V/[GF_HAL][gf_hal_common]: [gf_hal_common_authenticate] enter
2019-04-30 03:10:20.269 792-792/? D/[GF_HAL][gf_hal_common]: [gf_hal_common_authenticate] group_id=0
2019-04-30 03:17:56.329 792-1744/? I/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] irq_type=[GF_IRQ_FINGER_UP_MASK], irq_type=0x0004
2019-04-30 03:17:56.329 792-1744/? I/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] irq_type=[GF_IRQ_FINGER_UP_MASK], irq_type=0x00000004
2019-04-30 03:17:56.329 792-1744/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] GF_IRQ_FINGER_UP_MASK
2019-04-30 03:17:56.329 792-1744/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] enter
2019-04-30 03:17:56.329 792-1744/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] enter
2019-04-30 03:17:56.329 792-1744/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] exit
2019-04-30 03:17:56.329 792-1744/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] exit
2019-04-30 03:17:56.329 792-1744/? D/fingerprintd: onAcquired(1023)
2019-04-30 03:17:56.330 792-1744/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] exit
2019-04-30 03:17:56.330 792-1744/? V/[GF_HAL][gf_hal]: [gf_hal_irq] exit
2019-04-30 03:17:56.345 792-1745/? D/[GF_HAL][gf_hal_device]: [gf_hal_netlink_recv] send message value=1

E mobile app


2019-04-30 03:29:27.999 1568-1568/? W/FingerprintService: client com.EurobankEFG is no longer authenticating
2019-04-30 03:29:28.001 792-792/? V/FingerprintHal: fingerprint_get_auth_id
2019-04-30 03:29:28.001 792-792/? V/[GF_HAL][gf_hal]: [gf_hal_get_auth_id] enter
2019-04-30 03:29:28.001 792-792/? I/[GF_HAL][gf_hal]: [gf_hal_get_auth_id]
2019-04-30 03:29:28.001 792-792/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_get_auth_id] enter
2019-04-30 03:29:28.001 792-792/? V/[GF_HAL][gf_hal_common]: [gf_hal_common_get_auth_id] enter
2019-04-30 03:29:28.001 792-792/? V/[GF_HAL][gf_hal_common]: [gf_hal_invoke_command] enter
2019-04-30 03:29:28.001 792-792/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] enter
2019-04-30 03:29:28.001 792-792/? D/[gf_ca_entry]: [gx_ta_send_modified_cmd_req] begin token 11278, cmd_id=1011
2019-04-30 03:29:28.007 792-792/? D/[gf_ca_entry]: [gx_ta_send_modified_cmd_req] end
2019-04-30 03:29:28.007 792-792/? D/[gf_ca_entry]: [gf_ca_invoke_command] gx_ta_send_modified_cmd_req success
2019-04-30 03:29:28.007 792-792/? D/[gf_ca_entry]: [gf_ca_invoke_command] ta execution succeed
2019-04-30 03:29:28.007 792-792/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] exit
2019-04-30 03:29:28.007 792-792/? V/[GF_HAL][gf_hal_common]: [gf_hal_invoke_command] exit
2019-04-30 03:29:28.007 792-792/? V/[GF_HAL][gf_hal_common]: [gf_hal_common_get_auth_id] exit
2019-04-30 03:29:28.007 792-792/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_get_auth_id] exit
2019-04-30 03:29:28.007 792-792/? V/[GF_HAL][gf_hal]: [gf_hal_get_auth_id] exit
2019-04-30 03:29:28.008 1568-1568/? E/FingerprintService: stopAuthentication: is called, so return.
2019-04-30 03:29:28.009 792-792/? V/fingerprintd: authenticate(sid=5484728652605881930, gid=0)
2019-04-30 03:29:28.009 792-792/? V/FingerprintHal: fingerprint_authenticate


2019-04-30 03:29:28.041 792-792/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] cmd_id=GF_CMD_AUTHENTICATE, cmd_id=1010
2019-04-30 03:29:28.041 792-792/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] g_mode=MODE_IMAGE, g_mode=0
2019-04-30 03:29:28.041 792-792/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] g_operation=OPERATION_AUTHENTICATE_IMAGE, g_operation=1
2019-04-30 03:29:28.041 792-792/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] operation[0]=OPERATION_HOME_KEY, operation[0]=9
2019-04-30 03:29:28.041 792-792/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] exit
2019-04-30 03:29:28.041 792-792/? V/[GF_HAL][gf_hal_common]: [gf_hal_invoke_command] exit
2019-04-30 03:29:28.041 792-792/? D/fingerprintd: onAcquired(1021)
2019-04-30 03:29:28.041 792-792/? V/[GF_HAL][gf_hal_common]: [gf_hal_common_authenticate] exit
2019-04-30 03:29:28.041 792-792/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_authenticate] exit
2019-04-30 03:29:28.041 792-792/? V/[GF_HAL][gf_hal]: [gf_hal_authenticate] exit
2019-04-30 03:29:28.042 1568-1568/? W/FingerprintService: client com.EurobankEFG is authenticating...
2019-04-30 03:29:28.052 792-792/? V/fingerprintd: stopAuthentication()
2019-04-30 03:29:28.052 792-792/? V/FingerprintHal: fingerprint_cancel
2019-04-30 03:29:28.052 792-792/? V/[GF_HAL][gf_hal]: [gf_hal_cancel] enter
2019-04-30 03:29:28.052 792-792/? I/[GF_HAL][gf_hal]: [gf_hal_cancel]
2019-04-30 03:29:28.052 792-792/? D/fingerprintd: onError(5)
2019-04-30 03:29:28.052 792-792/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_cancel] enter
2019-04-30 03:29:28.052 792-792/? V/[GF_HAL][gf_hal_common]: [gf_hal_nav_reset] enter
2019-04-30 03:29:28.052 792-792/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] enter
2019-04-30 03:29:28.052 792-792/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] enter
2019-04-30 03:29:28.052 792-792/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] exit
2019-04-30 03:29:28.052 792-792/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] exit


2019-04-30 03:33:01.022 792-1744/? D/[gf_ca_entry]: [gf_ca_invoke_command] ta execution succeed
2019-04-30 03:33:01.022 792-1744/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] cmd_id=GF_CMD_IRQ, cmd_id=1016
2019-04-30 03:33:01.022 792-1744/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] g_mode=MODE_KEY, g_mode=1
2019-04-30 03:33:01.022 792-1744/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] g_operation=OPERATION_HOME_KEY, g_operation=9
2019-04-30 03:33:01.022 792-1744/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] exit
2019-04-30 03:33:01.022 792-1744/? V/[GF_HAL][gf_hal_common]: [gf_hal_invoke_command] exit
2019-04-30 03:33:01.023 792-1744/? I/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] irq_type=[GF_IRQ_FINGER_UP_MASK], irq_type=0x0004
2019-04-30 03:33:01.023 792-1744/? I/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] irq_type=[GF_IRQ_FINGER_UP_MASK], irq_type=0x00000004
2019-04-30 03:33:01.023 792-1744/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] GF_IRQ_FINGER_UP_MASK
2019-04-30 03:33:01.023 792-1744/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] enter
2019-04-30 03:33:01.023 792-1744/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] enter
2019-04-30 03:33:01.023 792-1744/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] exit
2019-04-30 03:33:01.023 792-1744/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] exit
2019-04-30 03:33:01.023 792-1744/? D/fingerprintd: onAcquired(1023)
2019-04-30 03:33:01.023 792-1744/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] exit
2019-04-30 03:33:01.023 792-1744/? V/[GF_HAL][gf_hal]: [gf_hal_irq] exit
2019-04-30 03:33:01.285 15166-15166/? D/wpa_supplicant: wlan0: Control interface command 'SIGNAL_POLL'
2019-04-30 03:33:01.288 15166-15166/? D/wpa_supplicant: CTRL-DEBUG: global_ctrl_sock-sendto: sock=8 sndbuf=212992 outq=0 send_len=48

N mobile app

```
2019-04-30 03:22:28.869 792-1744/? V/[GF_HAL][gf_hal_common]: [gf_hal_update_stitch] enter
2019-04-30 03:22:28.869 792-1744/? V/[GF_HAL][gf_hal_common]: [gf_hal_invoke_command] enter
2019-04-30 03:22:28.869 792-1744/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] enter
2019-04-30 03:22:28.869 792-1744/? D/[gf_ca_entry]: [gx_ta_send_modified_cmd_req] begin token 11256, cmd_id=1085
2019-04-30 03:22:28.861 1744-1744/? W/fingerprintd: type=1400 audit(0.0:729336): avc: denied { search } for name="/" dev="sdcardfs" ino=1205313 scontext=u:r:fingerprintd:s0
 tcontext=u:object_r:sdcardfs:s0 tclass=dir permissive=0
2019-04-30 03:22:28.861 1744-1744/? W/fingerprintd: type=1400 audit(0.0:729337): avc: denied { search } for name="/" dev="sdcardfs" ino=1205313 scontext=u:r:fingerprintd:s0
 tcontext=u:object_r:sdcardfs:s0 tclass=dir permissive=0
2019-04-30 03:22:28.861 1744-1744/? W/fingerprintd: type=1400 audit(0.0:729338): avc: denied { read } for name="/" dev="dm-1" ino=2 scontext=u:r:fingerprintd:s0
 tcontext=u:object_r:system_data_file:s0 tclass=dir permissive=0
2019-04-30 03:22:28.878 24570-24570/? W/art: JNI RegisterNativeMethods: attempt to register 0 native methods for md53e34d61a1c72634ccf644e103c59687e.AuthenticationResult
2019-04-30 03:22:28.880 24570-24570/? W/art: JNI RegisterNativeMethods: attempt to register 0 native methods for md53e34d61a1c72634ccf644e103c59687e.CryptoObject
```

```
2019-04-30 03:22:29.015 792-1744/? I/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] irq_type=[GF_IRQ_FINGER_UP_MASK], irq_type=0x0004
2019-04-30 03:22:29.015 792-1744/? I/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] irq_type=[GF_IRQ_FINGER_UP_MASK], irq_type=0x00000004
2019-04-30 03:22:29.015 792-1744/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] GF_IRQ_FINGER_UP_MASK
2019-04-30 03:22:29.015 792-1744/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] enter
2019-04-30 03:22:29.015 792-1744/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] enter
2019-04-30 03:22:29.015 792-1744/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] exit
2019-04-30 03:22:29.015 792-1744/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] exit
2019-04-30 03:22:29.015 792-1744/? D/fingerprintd: onAcquired(1023)
2019-04-30 03:22:29.016 792-1744/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] exit
2019-04-30 03:22:29.016 792-1744/? V/[GF_HAL][gf_hal]: [gf_hal_irq] exit
```

W mobile app

```
2019-04-30 03:31:17.449 792-1744/? D/[gf_ca_entry]: [gx_ta_send_modified_cmd_req] end
2019-04-30 03:31:17.449 792-1744/? D/[gf_ca_entry]: [gf_ca_invoke_command] gx_ta_send_modified_cmd_req success
2019-04-30 03:31:17.449 792-1744/? D/[gf_ca_entry]: [gf_ca_invoke_command] ta execution succeed
2019-04-30 03:31:17.449 792-1744/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] cmd_id=GF_CMD_IRQ, cmd_id=1016
2019-04-30 03:31:17.449 792-1744/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] g_mode=MODE_KEY, g_mode=1
2019-04-30 03:31:17.449 792-1744/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] g_operation=OPERATION_HOME_KEY, g_operation=9
2019-04-30 03:31:17.449 792-1744/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_invoke_command] exit
2019-04-30 03:31:17.449 792-1744/? V/[GF_HAL][gf_hal_common]: [gf_hal_invoke_command] exit
2019-04-30 03:31:17.449 792-1744/? I/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] irq_type=[GF_IRQ_FINGER_UP_MASK], irq_type=0x0004
2019-04-30 03:31:17.449 792-1744/? I/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] irq_type=[GF_IRQ_FINGER_UP_MASK], irq_type=0x00000004
2019-04-30 03:31:17.449 792-1744/? D/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] GF_IRQ_FINGER_UP_MASK
2019-04-30 03:31:17.449 792-1744/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] enter
2019-04-30 03:31:17.449 792-1744/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] enter
2019-04-30 03:31:17.449 792-1744/? V/[GF_HAL][gf_hal_timer]: [hal_destroy_timer] exit
2019-04-30 03:31:17.449 792-1744/? V/[GF_HAL][gf_hal_timer]: [gf_hal_destroy_timer] exit
2019-04-30 03:31:17.449 792-1744/? D/fingerprintd: onAcquired(1023)
2019-04-30 03:31:17.449 792-1744/? V/[GF_HAL][gf_hal_milan_f_series]: [hal_milan_f_series_irq] exit
2019-04-30 03:31:17.449 792-1744/? V/[GF_HAL][gf_hal]: [gf_hal_irq] exit
```

Με βάση όλα τα παραπάνω, αναλύουμε πότε και από ποιο activity καλεί η εφαρμογή τη συγκεκριμένη λειτουργία που μελετούμε, και βγάζουμε πολλά άλλα συμπεράσματα για τη γενικότερη ασφάλεια της. Παρατηρήθηκε μετά από πολλές προσπάθειες ότι η εφαρμογή A Mobile App δεν επιτρέπει το reverse engineering και την εξαγωγή του κώδικα της (`dex>jar>java`) πιθανώς λόγω κάποιας άμυνας που έχουν εφαρμόσει οι προγραμματιστές. Επιπλέον, παρατηρήθηκε ότι ενώ στην ανάλυση του κώδικα με το Android Studio logcat εμφανίζονται οι κλάσεις και μέθοδοι της βιομετρικής αυθεντικοποίησης, στην ανάλυση με dexplorer και μετά το reverse engineering που πραγματοποιήσαμε δεν είναι ορατές οι συγκεκριμένες κλάσεις. Για το λόγο αυτό θα επιλέξουμε να επικεντρωθούμε στη μελέτη και ανάλυση των λειτουργιών αυτών όπως φαίνονται από την ανάλυση APK.

3.4 Static & Dynamic Analysis

Τέλος, η ανάλυση του κώδικα και της λειτουργίας μιας εφαρμογής γίνεται και μέσω της δυναμικής και της στατικής ανάλυσης. Πλέον πρόκειται για διαδικασίες που γίνονται και αυτοματοποιημένα. Η στατική ανάλυση πραγματοποιείται σε περιβάλλον μη-εκτέλεσης. Συνήθως, ένα εργαλείο στατικής ανάλυσης εκτελεί reverse engineering ώστε να επιθεωρήσει τον κώδικα του προγράμματος για όλες τις πιθανές συμπεριφορές χρόνου εκτέλεσης και θα αναζητήσει ελαττώματα στον κώδικα, backdoor αδυναμίες και δυνητικά κακόβουλο κώδικα. Η δυναμική ανάλυση υιοθετεί την αντίθετη

προσέγγιση και εκτελείται κατά τη λειτουργία μιας εφαρμογής. Μια δυναμική δοκιμή παρακολουθεί τη μνήμη του συστήματος, τη λειτουργική συμπεριφορά, το χρόνο απόκρισης και τη συνολική απόδοση του συστήματος. Και οι δύο μέθοδοι καλύπτουν διαφορετικές πτυχές ασφαλείας της εφαρμογής.

Η στατική ανάλυση, με white box visibility, είναι σίγουρα η πιο εμπεριστατωμένη προσέγγιση και μπορεί να αποδειχθεί πιο αποδοτική από πλευράς κόστους, λόγω του ότι μπορεί να ανιχνεύσει σφάλματα στις πρώτες φάσης του κύκλου ζωής της ανάπτυξης λογισμικού. Για παράδειγμα, εάν εντοπιστεί σφάλμα σε revision session ή σε office check - και τα δύο τύποι στατικής ανάλυσης – η διόρθωση μπορεί να διορθωθεί χωρίς μεγάλο κόστος. Αν το σφάλμα καταγραφεί στο σύστημα, το κόστος είναι σαφώς μεγαλύτερο. Η στατική ανάλυση μπορεί επίσης να εντοπίσει μελλοντικά σφάλματα που ίσως δεν προκύψουν σε μια δυναμική δοκιμή. Η δυναμική ανάλυση, από την άλλη πλευρά, μπορεί να εντοπίσει ένα μικρό ελάττωμα ή ευπάθεια πολύ περίπλοκο για τη στατική. Μια δυναμική δοκιμή, ωστόσο, θα βρει μόνο ελαττώματα στο τμήμα του κώδικα που όντως εκτελείται. Ενώ η στατική ανάλυση μπορεί να θεωρηθεί ανώτερη μέθοδος δοκιμών, δεν προκύπτει κατ' ανάγκη ότι θα πρέπει να επιλέγεται αυτόματα σε σχέση με τη δυναμική ανάλυση, αν πρέπει να επιλεγεί η μία. Ιδανικά, δρουν συμπληρωματικά και δίνουν μαζί τα πλέον σφαιρικά αποτελέσματα. Πιο συνοπτικά:

Static analysis:

- Non-runtime environment - χωρίς να εκτελείται το πρόγραμμα
- Ανάλυση κώδικα για όλες τις πιθανές συμπεριφορές κατά το runtime
- Εντοπίζει ελαττώματα, backdoors, κακόβουλο κώδικα
- Whitebox visibility
- Πιο ενδελεχής
- Εντοπίζει αδυναμίες έγκαιρα, ακόμα και κατά το development
- Μπορεί να εντοπίσει αδυναμίες στον κώδικα δίνοντας ακριβή τοποθεσία
- Δεν εντοπίζει ευπάθειες που εισάγονται στο περιβάλλον εκτέλεσης
- Μπορεί να διεξαχθεί από εκπαιδευμένους προγραμματιστές διασφάλισης λογισμικού που κατανοούν πλήρως τον κώδικα
- Είναι σχετικά γρήγορο εάν χρησιμοποιούνται αυτοματοποιημένα εργαλεία που επίσης παρέχουν προτάσεις αντιμέτρων, μειώνοντας τον χρόνο της έρευνας. Είναι χρονοβόρα αν πραγματοποιηθεί χειροκίνητα. Επιπλέον, τα αυτοματοποιημένα εργαλεία παράγουν false positives και false negatives.

Dynamic analysis:

- Έλεγχος ενώ το πρόγραμμα βρίσκεται σε λειτουργία
- Αναλύει πώς αλληλεπιδρά ο κώδικας με άλλα system components (SQL DBs, app servers, web services κλπ.)

- Παρακολούθηση μνήμης συστήματος, χρόνου απόκρισης, απόδοση, λειτουργία
- Εντοπίζει πιο πολύπλοκες αδυναμίες
- Επιτρέπει την ανάλυση εφαρμογών στις οποίες δεν έχετε πρόσβαση στον πραγματικό κώδικα
- Προσδιορίζει ευπάθειες που μπορεί να ήταν false negatives στη στατική ανάλυση
- Επικυρώνει τα ευρήματα στατικής ανάλυσης κώδικα
- Μπορεί να διεξαχθεί σε οποιαδήποτε εφαρμογή.

Figure 52 - MobSF & QARK

Για τις δοκιμές στατικής ανάλυσης θα χρησιμοποιήσουμε δύο εργαλεία ανοιχτού κώδικα, διαθέσιμα για το σκοπό αυτό. Για τη στατική ανάλυση, θα χρησιμοποιήσουμε το MobSF (Mobile Security Framework) ένα από τα καλύτερα και πληρέστερα εργαλεία ανάλυσης κινητών εφαρμογών το οποίο παρέχει και τη δυνατότητα δυναμικής ανάλυσης. Επιπλέον για τη στατική ανάλυση θα χρησιμοποιήσουμε και το QARK (Quick Android Review Kit), που θεωρείται από τα αποδοτικότερα εργαλεία ανάλυσης κώδικα. Συχνά, επειδή τα διάφορα εργαλεία ανάλυσης και αποτίμησης ασφαλείας κατασκευάζονται με διαφορετικούς σκοπούς, για τις διάφορες ανάγκες και εξειδικεύσεις των εταιρειών, συνίσταται να χρησιμοποιούνται περισσότερα από ένα ακόμα και για το ίδιο είδος ανάλυσης, για την πληρέστερη αξιολόγηση της ασφάλειας μιας εφαρμογής.

Το QARK είναι εργαλείο στατικής ανάλυσης, που αναπτύχθηκε αρχικά για λειτουργικό Linux, αλλά οι δημιουργοί του έχουν ενημερώσει την εφαρμογή που πλέον είναι συμβατή και με άλλα συστήματα. Στη μελέτη μας το QARK εγκαταστάθηκε και χρησιμοποιήθηκε σε περιβάλλον Kali Linux. Προσπελάστηκε η έκδοση Μαρτίου 2019, όπου τόσο οι εντολές όσο και το αποτέλεσμα της ανάλυσης έχουν απλοποιηθεί αρκετά. Απαιτείται Python (2.7.13 και εξής). Το QARK παρέχει τη δυνατότητα ανάλυσης αρχείων apk, καθώς πραγματοποιεί αυτόματα reverse engineering του πηγαίου κώδικα με τα εργαλεία apkTool και Dex2Jar, αλλά και ανάλυση πηγαίου κώδικα java με τη σχετική εντολή. [3] Τα αποτελέσματα της ανάλυσης παρατίθενται σε report που παράγει το QARK σε JSON ή HTML,

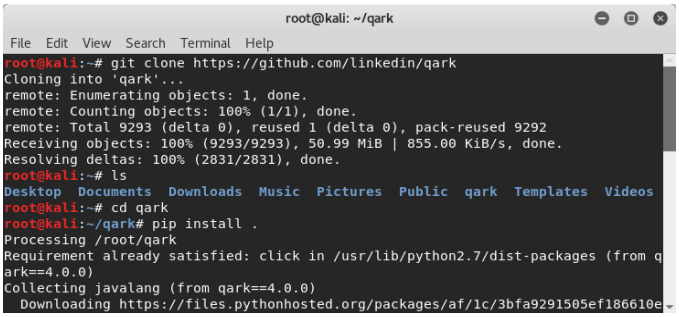


Figure 53 - Εγκατάσταση QARK

[3] Όλες οι εντολές που χρησιμοποιήθηκαν βρίσκονται στο παράρτημα Α

καθώς ο χρήστης μπορεί να αλλάξει τη μορφή εξόδου με το flag `--report`. [33]

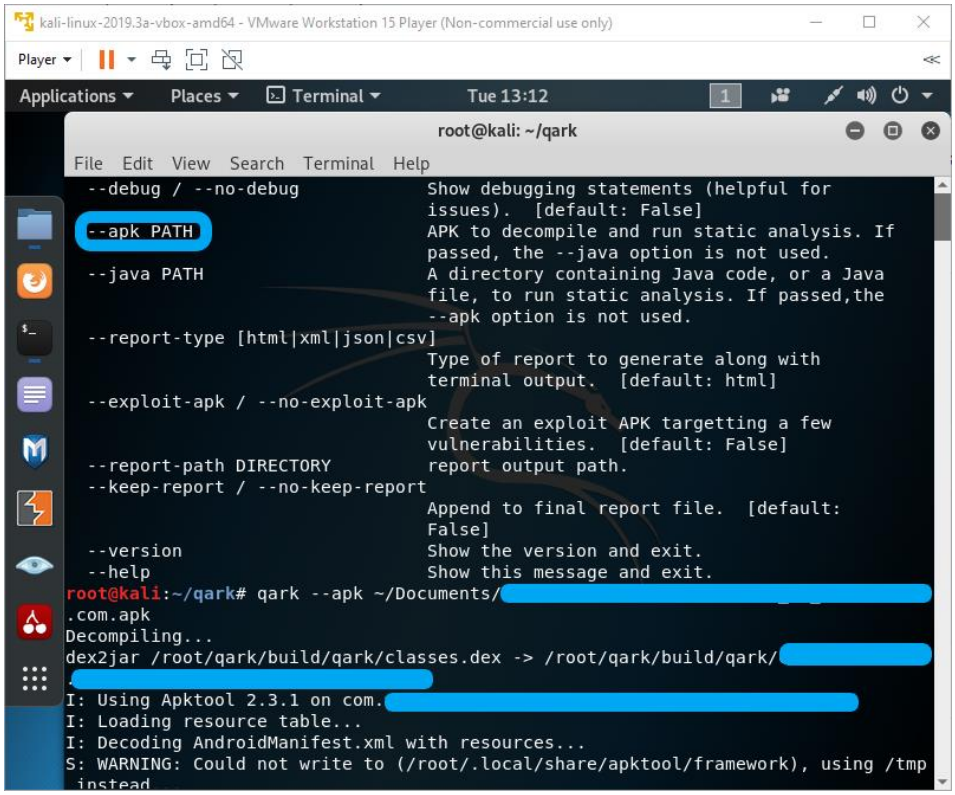


Figure 54 - QARK Static Analysis command

Commented [Az9]: Better screenshot

Στη συνέχεια παρατίθενται τα σημαντικότερα ευρήματα της ανάλυσης, όπως προκύπτουν από τα reports του QARK.

E mobile app

WARNING Exported tags

The activity **.******.********.activity.****.******.**********Activity is exported, but not protected by any permissions. Failing to protect activity tags could leave them vulnerable to attack by malicious apps. The activity tags should be reviewed for vulnerabilities, such as injection and information leakage.

File: /root/Desktop/build/qark/AndroidManifest.xml

WARNING launchMode=singleTask found

This results in AMS either resuming the earlier activity or loads it in a task with same affinity or the activity is started as a new task. This may result in Task Poisoning.
https://www.usenix.org/system/files/conference/usenixsecurity15/sec15-paper-ren-chuangang.pdf

File: /root/Desktop/build/qark/AndroidManifest.xml::

WARNING Exported tags

The service **.******.********.network.push.firebase.****************************** is exported, but not protected by any permissions. Failing to protect service tags could leave them vulnerable to attack by malicious apps. The service tags should be reviewed for vulnerabilities, such as injection and information leakage.
WARNING Exported tags

The activity **.******.********.activity.launch.LaunchActivity is exported, but not protected by any permissions. Failing to protect activity tags could leave them vulnerable to attack by malicious apps. The activity tags should be reviewed for vulnerabilities, such as injection and information leakage.

File: /root/Desktop/build/qark/AndroidManifest.xml

WARNING Backup is allowed in manifest

Backups enabled: Potential for data theft via local attacks via adb backup, if the device has USB debugging enabled (not common). More info:
http://developer.android.com/reference/android/R.attr.html#allowBackup

File: /root/Desktop/build/qark/AndroidManifest.xml

A mobile app

Δεδομένου ότι το εργαλείο πραγματοποιεί reverse engineering στο αρχείο apk που μελετά και εξάγει τον κώδικα, προέκυψε ακριβώς το ίδιο πρόβλημα κατά την ανάλυση της συγκεκριμένης εφαρμογής με το QARK που αντιμετωπίσαμε και κατά το manual reverse engineering, δηλαδή προέκυψε σφάλμα και η διαδικασία αποσυμπίεσης και ανάλυσης διακόπηκε. Πιθανόν οι προγραμματιστές να τοποθέτησαν μηχανισμούς άμυνας στην εφαρμογή για αυτόν ακριβώς το λόγο.

W & N mobile apps

Στις υπόλοιπες εφαρμογές που μελετούμε παρατηρήθηκαν μόνο ενδείξεις και παρατηρήσεις που εφιστούν την προσοχή σε συγκεκριμένα apk keys, τίποτα όμως που να δείχνει επικίνδυνο σε θέματα ασφαλείας.

Στη συνέχεια θα επαναλάβουμε τη διαδικασία της στατικής ανάλυσης με το δεύτερο εργαλείο μας, το MobSF. Το συγκεκριμένο εργαλείο ανάλυσης θεωρείται ένα από τα καλύτερα όσον αφορά τις κινητές εφαρμογές. Απαιτείται εγκατάσταση του docker, αν και υπάρχουν εναλλακτικές που παρέχουν οι κατασκευαστές. Αφού έχουμε εγκαταστήσει το docker, ανοίγουμε το σχετικό interface για να τρέξουμε το MobSF με την εντολή

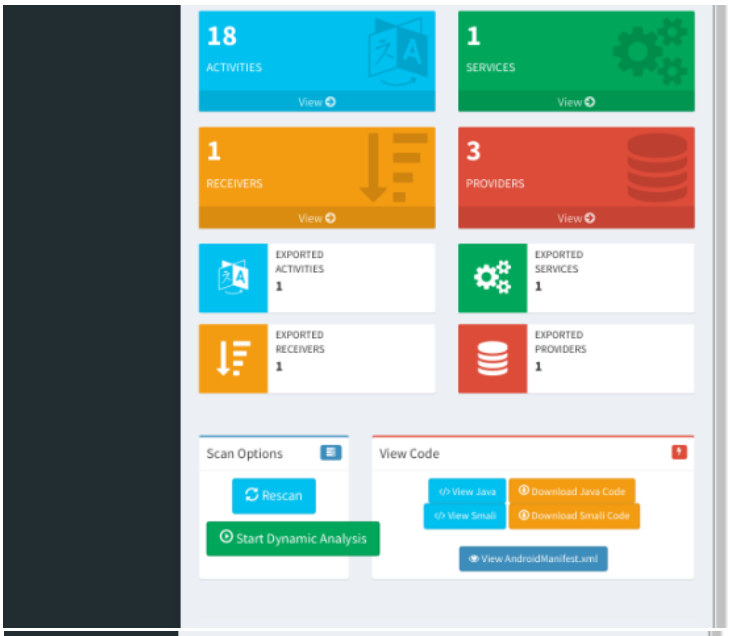
```
>docker pull opensecurity/mobile-security-frameworkmobsf
```

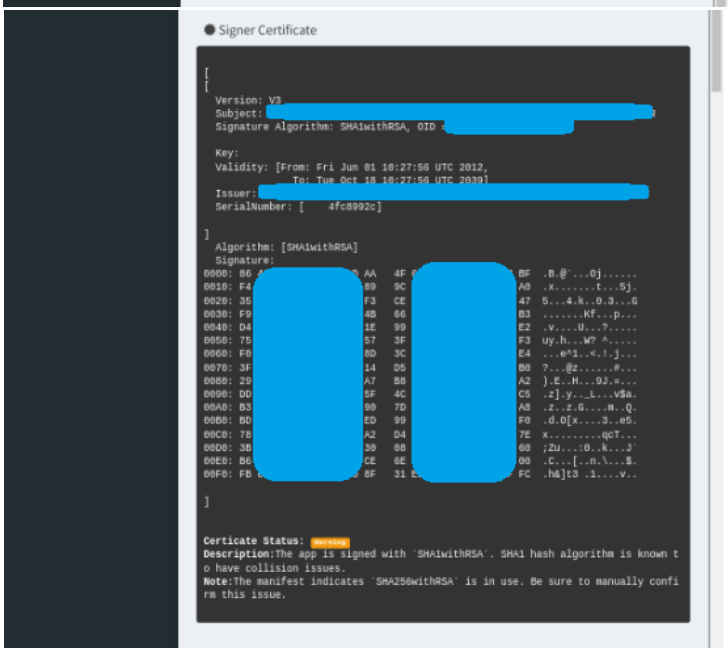
```
>docker run -it -p 8000:8000 opensecurity/mobilesecurity-framework-mobsf:latest
```

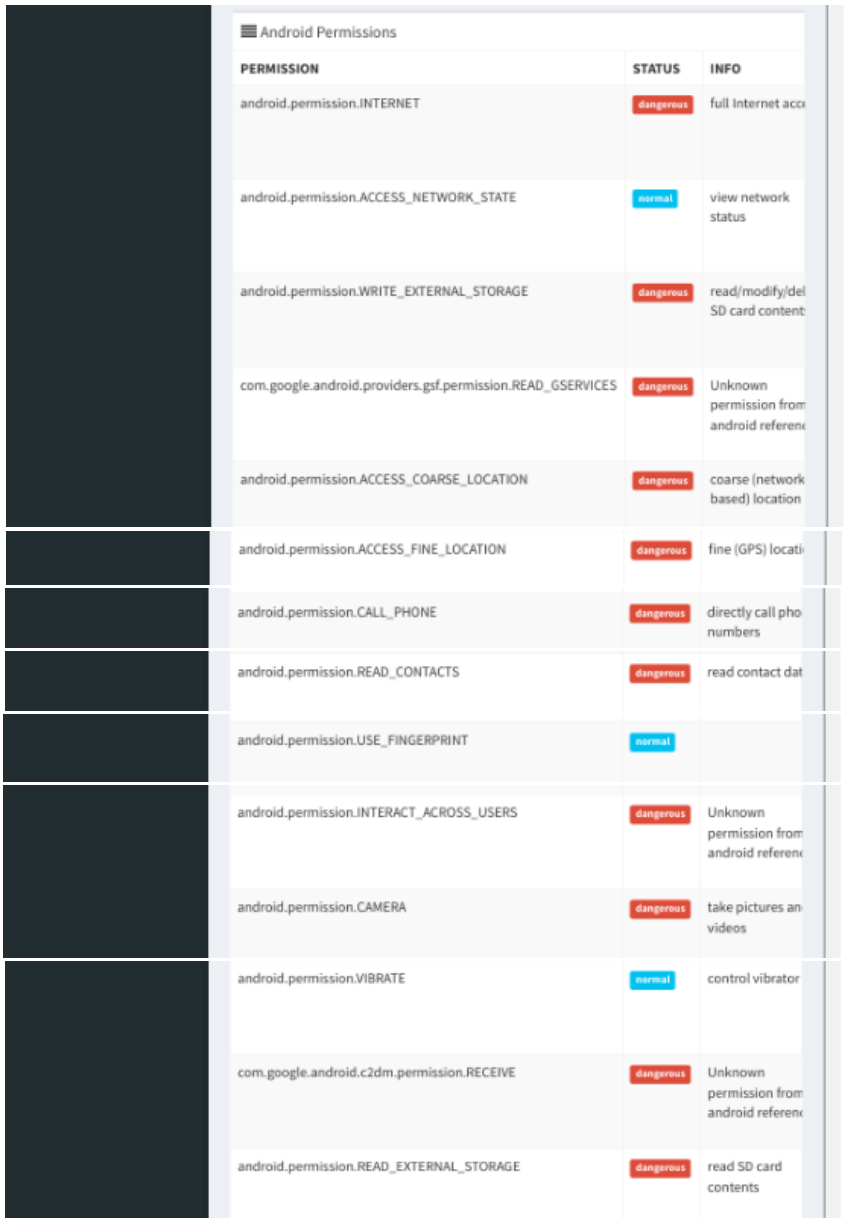
Με την εντολή αυτή ανοίγει στο browser μας το περιβάλλον χρήσης του MobSF, από όπου μπορούμε πολύ εύκολα να επιλέξουμε αρχείο APK για να πραγματοποιήσουμε την ανάλυση.

Το αποτέλεσμα της ανάλυσης είναι μια σελίδα που παρουσιάζει το αναλυτικό report της ανάλυσης. Τα reports παρατίθενται παρακάτω.

A mobile app

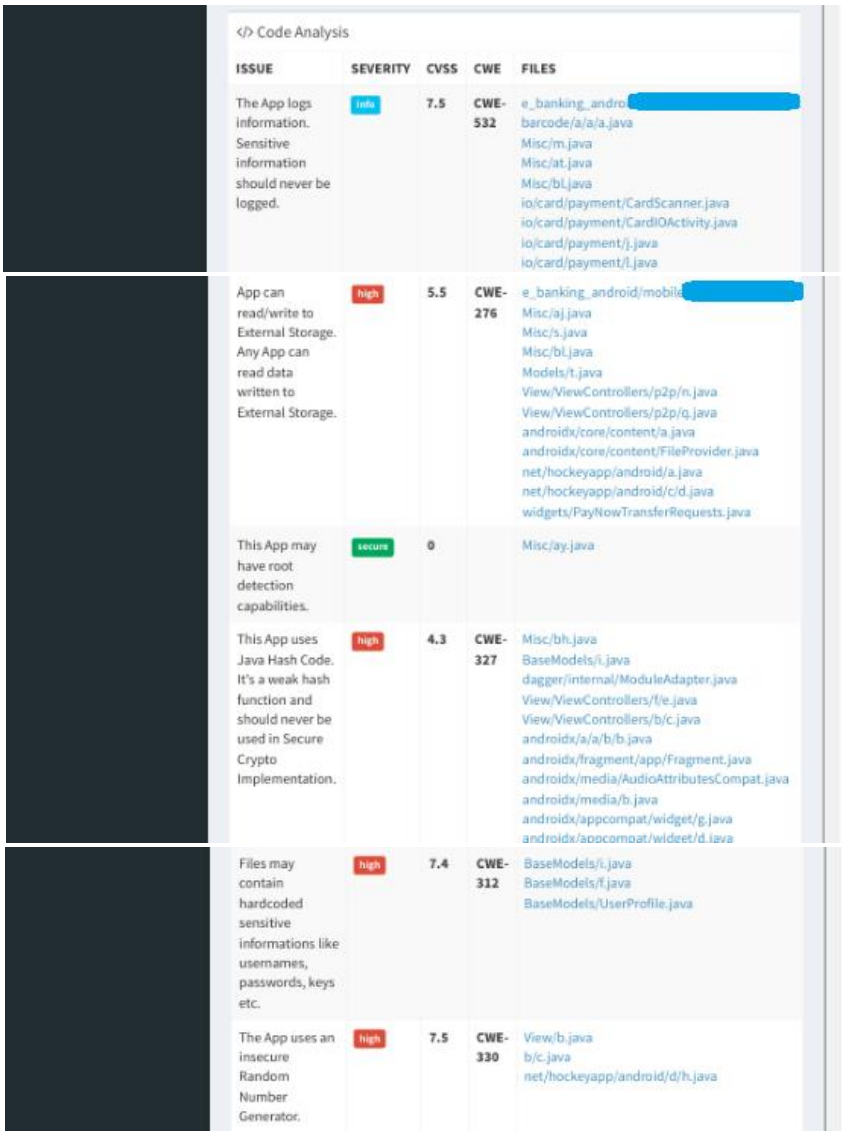

18
ACTIVITIES
View
1
SERVICES
View
1
RECEIVERS
View
3
PROVIDERS
View
EXPORTED ACTIVITIES
1
EXPORTED SERVICES
1
EXPORTED RECEIVERS
1
EXPORTED PROVIDERS
1
Scan Options
Rescan
Start Dynamic Analysis
View Code
View Java
Download Java Code
View Smali
Download Smali Code
View AndroidManifest.xml


Signer Certificate
Version: V3
Subject:
Signature Algorithm: SHA1withRSA, OID
Key:
Validity: [From: Fri Jun 01 10:27:56 UTC 2012,
To: Tue Oct 18 10:27:56 UTC 2039]
Issuer:
SerialNumber: [4fc8992c]
Algorithm: [SHA1withRSA]
Signature:
Certicate Status:
Description:The app is signed with `SHA1withRSA`. SHA1 hash algorithm is known to have collision issues.
Note:The manifest indicates `SHA256withRSA` is in use. Be sure to manually confirm this issue.



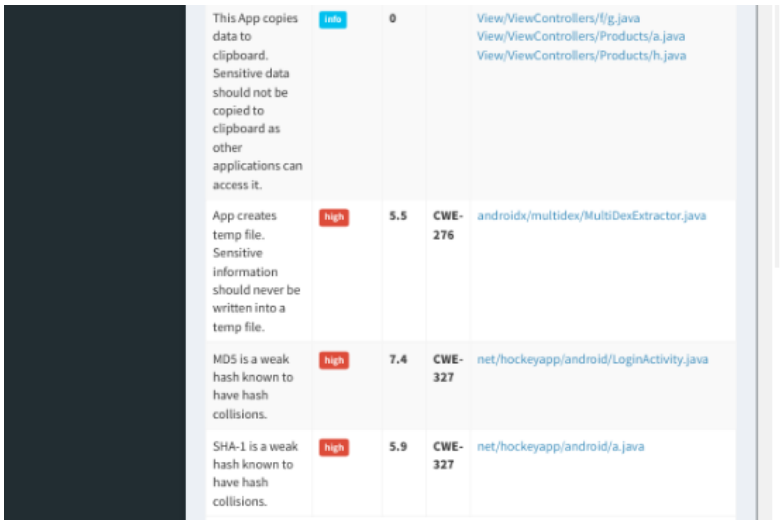
Android Permissions

PERMISSION	STATUS	INFO
android.permission.INTERNET	dangerous	full Internet acc
android.permission.ACCESS_NETWORK_STATE	normal	view network status
android.permission.WRITE_EXTERNAL_STORAGE	dangerous	read/modify/del SD card content
com.google.android.providers.gsf.permission.READ_GSERVICES	dangerous	Unknown permission from android referenc
android.permission.ACCESS_COARSE_LOCATION	dangerous	coarse (network based) location
android.permission.ACCESS_FINE_LOCATION	dangerous	fine (GPS) locati
android.permission.CALL_PHONE	dangerous	directly call pho numbers
android.permission.READ_CONTACTS	dangerous	read contact dat
android.permission.USE_FINGERPRINT	normal	
android.permission.INTERACT_ACROSS_USERS	dangerous	Unknown permission from android referenc
android.permission.CAMERA	dangerous	take pictures an videos
android.permission.VIBRATE	normal	control vibrator
com.google.android.c2dm.permission.RECEIVE	dangerous	Unknown permission from android referenc
android.permission.READ_EXTERNAL_STORAGE	dangerous	read SD card contents

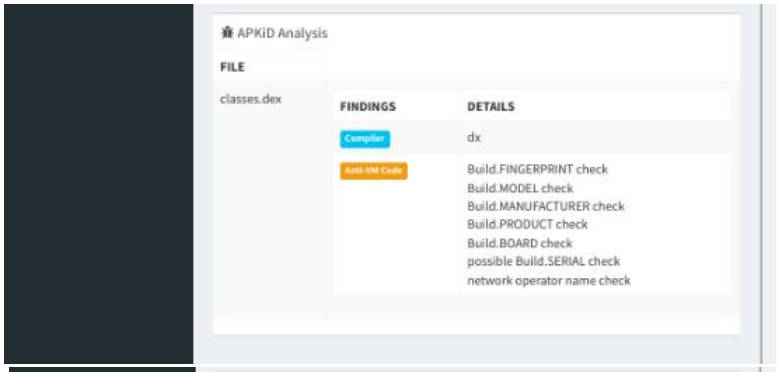


</> Code Analysis

ISSUE	SEVERITY	CVSS	CWE	FILES
The App logs information. Sensitive information should never be logged.	info	7.5	CWE-532	e_banking_andro[redacted] barcode/a/a/a.java Misc/m.java Misc/at.java Misc/bl.java io/card/payment/CardScanner.java io/card/payment/CardIOActivity.java io/card/payment/j.java io/card/payment/l.java
App can read/write to External Storage. Any App can read data written to External Storage.	high	5.5	CWE-276	e_banking_android/mobile[redacted] Misc/aj.java Misc/s.java Misc/bl.java Models/t.java View/ViewControllers/p2p/n.java View/ViewControllers/p2p/q.java androidx/core/content/a.java androidx/core/content/FileProvider.java net/hockeyapp/android/a.java net/hockeyapp/android/c/d.java widgets/PayNowTransferRequests.java
This App may have root detection capabilities.	secure	0		Misc/ay.java
This App uses Java Hash Code. It's a weak hash function and should never be used in Secure Crypto Implementation.	high	4.3	CWE-327	Misc/bh.java BaseModels/i.java dagger/internal/ModuleAdapter.java View/ViewControllers/f/e.java View/ViewControllers/b/c.java androidx/a/a/b/b.java androidx/fragment/app/Fragment.java androidx/media/AudioAttributesCompat.java androidx/media/b.java androidx/appcompat/widget/g.java androidx/appcompat/widget/d.java
Files may contain hardcoded sensitive informations like usernames, passwords, keys etc.	high	7.4	CWE-312	BaseModels/i.java BaseModels/f.java BaseModels/UserProfile.java
The App uses an insecure Random Number Generator.	high	7.5	CWE-330	View/b.java b/c.java net/hockeyapp/android/d/h.java

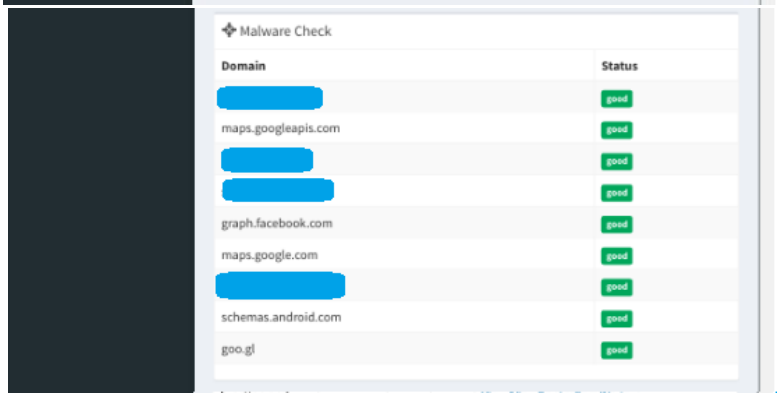


This App copies data to clipboard. Sensitive data should not be copied to clipboard as other applications can access it.	info	0		View/ViewControllers/f/g.java View/ViewControllers/Products/a.java View/ViewControllers/Products/h.java
App creates temp file. Sensitive information should never be written into a temp file.	high	5.5	CWE-276	androidx/multidex/MultiDexExtractor.java
MD5 is a weak hash known to have hash collisions.	high	7.4	CWE-327	net/hockeyapp/android/LoginActivity.java
SHA-1 is a weak hash known to have hash collisions.	high	5.9	CWE-327	net/hockeyapp/android/a.java



APKiD Analysis

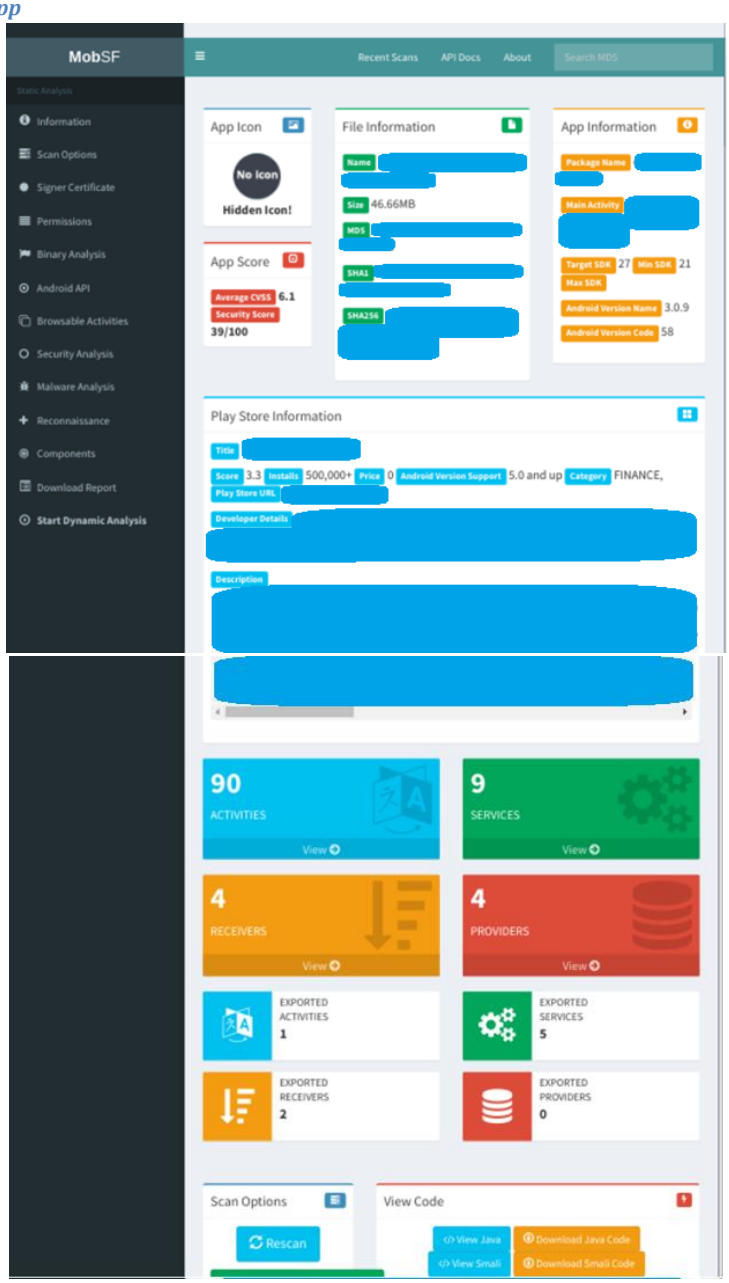
FILE	FINDINGS	DETAILS
classes.dex	Compiler	dx
	Anti-VM Code	Build.FINGERPRINT check Build.MODEL check Build.MANUFACTURER check Build.PRODUCT check Build.BOARD check possible Build.SERIAL check network operator name check

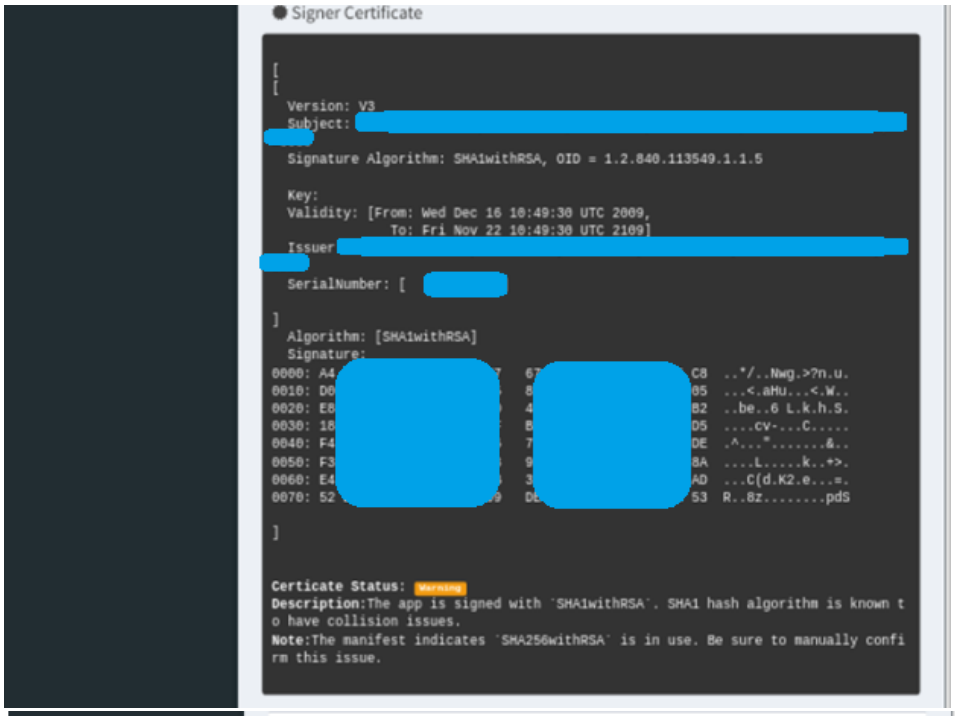


Malware Check

Domain	Status
[redacted]	good
maps.googleapis.com	good
[redacted]	good
[redacted]	good
graph.facebook.com	good
maps.google.com	good
[redacted]	good
schemas.android.com	good
goo.gl	good

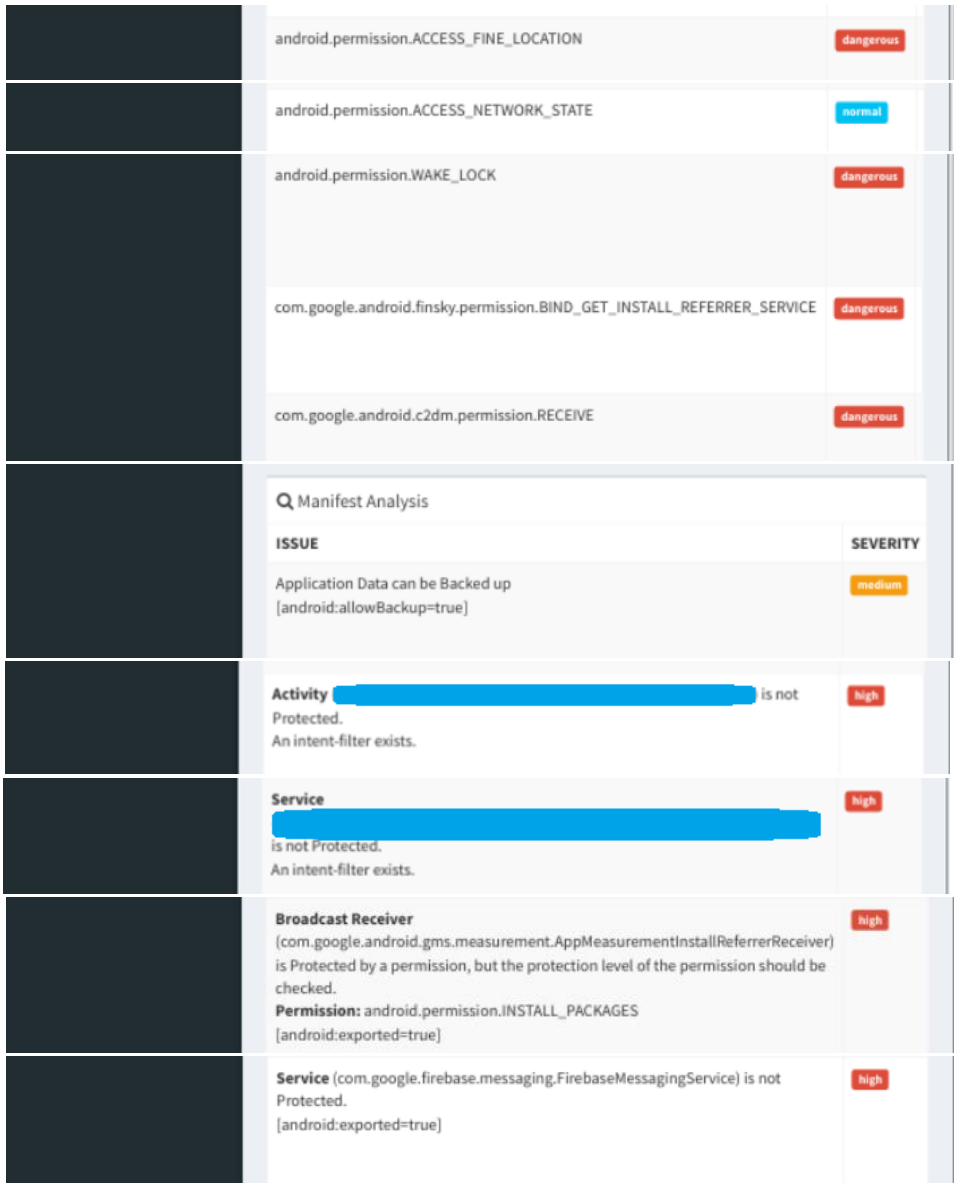
E mobile app


MobSF
Recent Scans
API Docs
About
Search MDS
Information
Scan Options
Signer Certificate
Permissions
Binary Analysis
Android API
Browsable Activities
Security Analysis
Malware Analysis
Reconnaissance
Components
Download Report
Start Dynamic Analysis
App Icon
No Icon
Hidden Icon!
App Score
Average CVSS 6.1
Security Score
39/100
File Information
Name
Size 46.66MB
MD5
SHA1
SHA256
App Information
Package Name
Main Activity
Target SDK 27
Min SDK 21
Max SDK
Android Version Name 3.0.9
Android Version Code 58
Play Store Information
Title
Score 3.3
Installs 500,000+
Price 0
Android Version Support 5.0 and up
Category FINANCE,
Play Store URL
Developer Details
Description
90
ACTIVITIES
View
9
SERVICES
View
4
RECEIVERS
View
4
PROVIDERS
View
EXPORTED ACTIVITIES 1
EXPORTED SERVICES 5
EXPORTED RECEIVERS 2
EXPORTED PROVIDERS 0
Scan Options
Rescan
View Code
View Java
Download Java Code
View Smali
Download Smali Code



Android Permissions

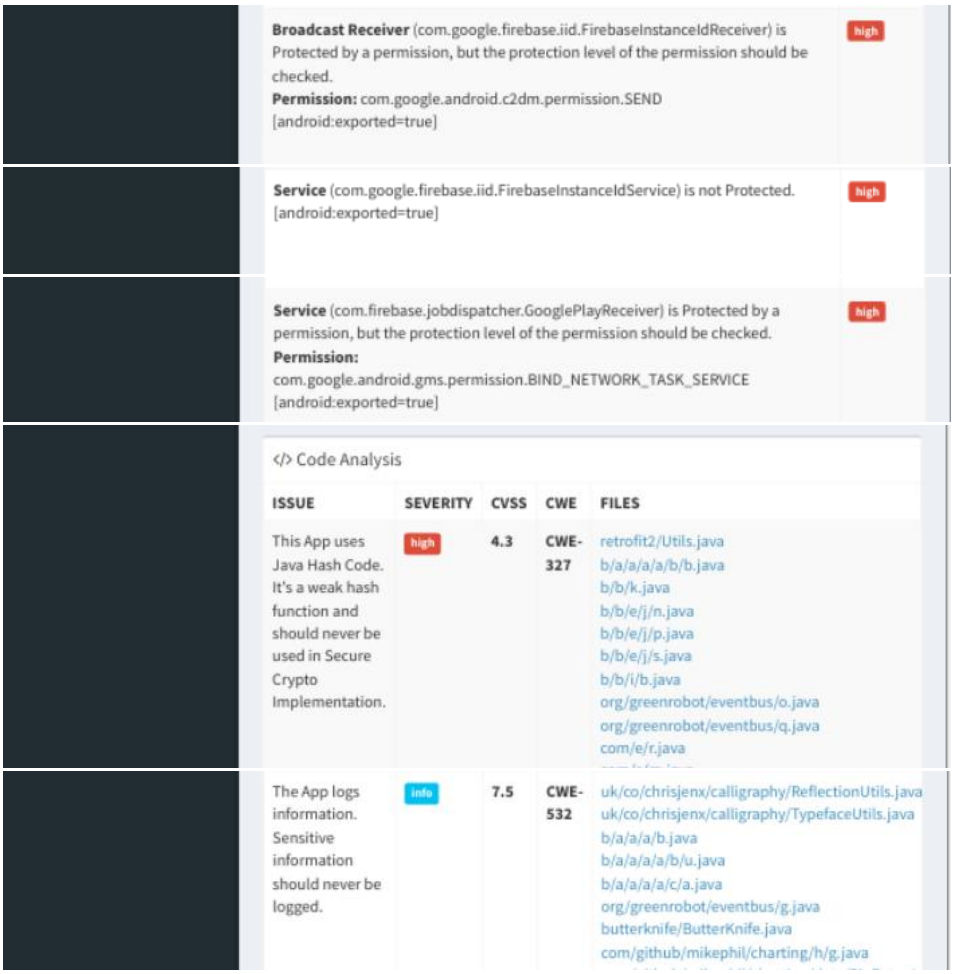
PERMISSION	STATUS
android.permission.INTERNET	dangerous
android.permission.USE_FINGERPRINT	normal
android.permission.WRITE_EXTERNAL_STORAGE	dangerous
android.permission.READ_EXTERNAL_STORAGE	dangerous
android.permission.VIBRATE	normal
android.permission.CAMERA	dangerous
android.permission.READ_CONTACTS	dangerous
android.permission.READ_PHONE_STATE	dangerous



android.permission.ACCESS_FINE_LOCATION	dangerous
android.permission.ACCESS_NETWORK_STATE	normal
android.permission.WAKE_LOCK	dangerous
com.google.android.finsky.permission.BIND_GET_INSTALL_REFERRER_SERVICE	dangerous
com.google.android.c2dm.permission.RECEIVE	dangerous

Manifest Analysis

ISSUE	SEVERITY
Application Data can be Backed up [android:allowBackup=true]	medium
Activity [redacted] is not Protected. An intent-filter exists.	high
Service [redacted] is not Protected. An intent-filter exists.	high
Broadcast Receiver (com.google.android.gms.measurement.AppMeasurementInstallReferrerReceiver) is Protected by a permission, but the protection level of the permission should be checked. **Permission:** android.permission.INSTALL_PACKAGES [android:exported=true]	high
Service (com.google.firebase.messaging.FirebaseMessagingService) is not Protected. [android:exported=true]	high



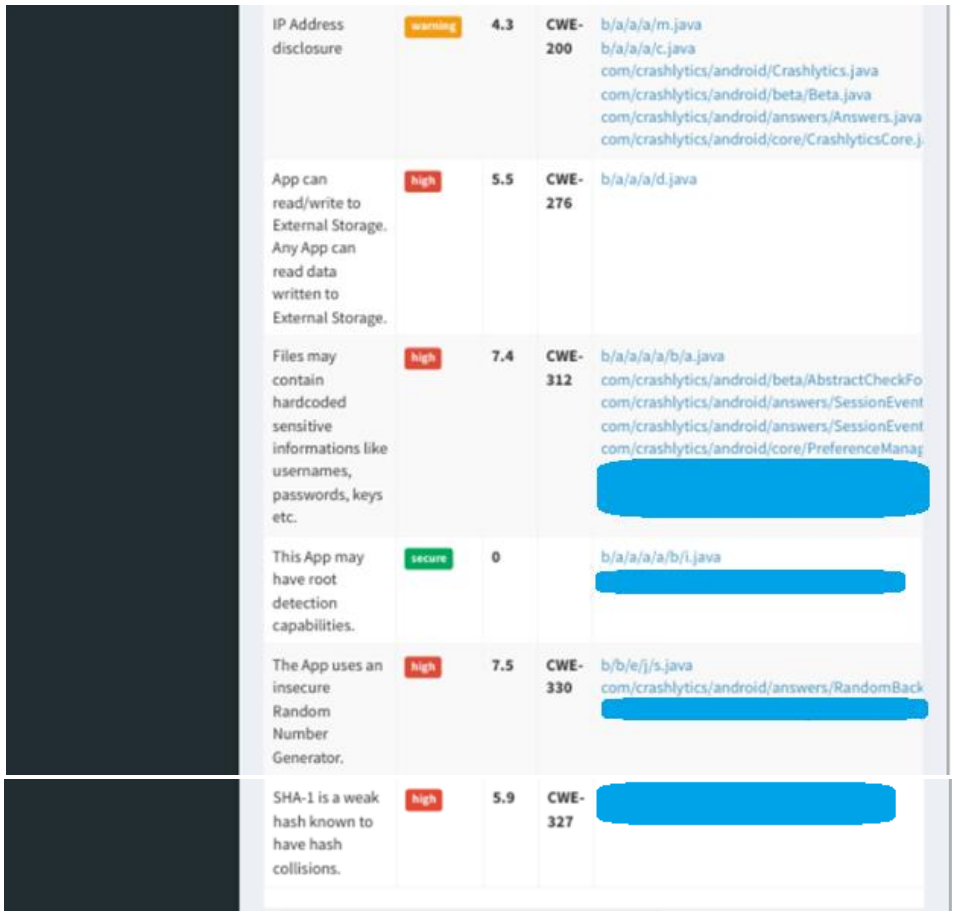
Broadcast Receiver (com.google.firebase.iid.FirebaseInstanceIdReceiver) is Protected by a permission, but the protection level of the permission should be checked.
Permission: com.google.android.c2dm.permission.SEND
[android:exported=true]
high

Service (com.google.firebase.iid.FirebaseInstanceIdService) is not Protected.
[android:exported=true]
high

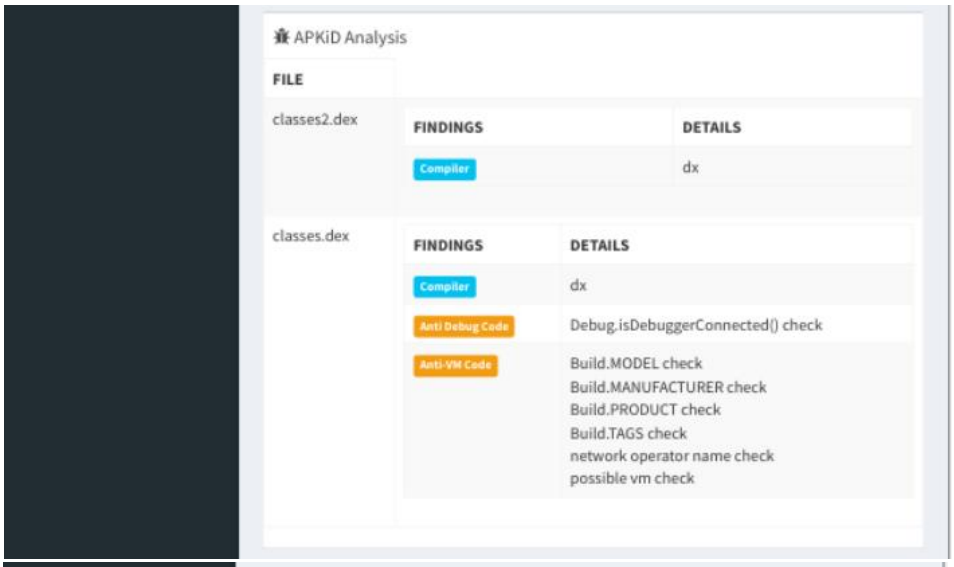
Service (com.firebase.jobdispatcher.GooglePlayReceiver) is Protected by a permission, but the protection level of the permission should be checked.
Permission:
com.google.android.gms.permission.BIND_NETWORK_TASK_SERVICE
[android:exported=true]
high

</> Code Analysis

ISSUE	SEVERITY	CVSS	CWE	FILES
This App uses Java Hash Code. It's a weak hash function and should never be used in Secure Crypto Implementation.	high	4.3	CWE-327	retrofit2/Utils.java b/a/a/a/a/b/b.java b/b/k.java b/b/e/j/n.java b/b/e/j/p.java b/b/e/j/s.java b/b/i/b.java org/greenrobot/eventbus/o.java org/greenrobot/eventbus/q.java com/e/r.java
The App logs information. Sensitive information should never be logged.	info	7.5	CWE-532	uk/co/chrisjenx/calligraphy/ReflectionUtils.java uk/co/chrisjenx/calligraphy/TypefaceUtils.java b/a/a/a/b.java b/a/a/a/a/b/u.java b/a/a/a/a/c/a.java org/greenrobot/eventbus/g.java butterknife/ButterKnife.java com/github/mikephil/charting/h/g.java

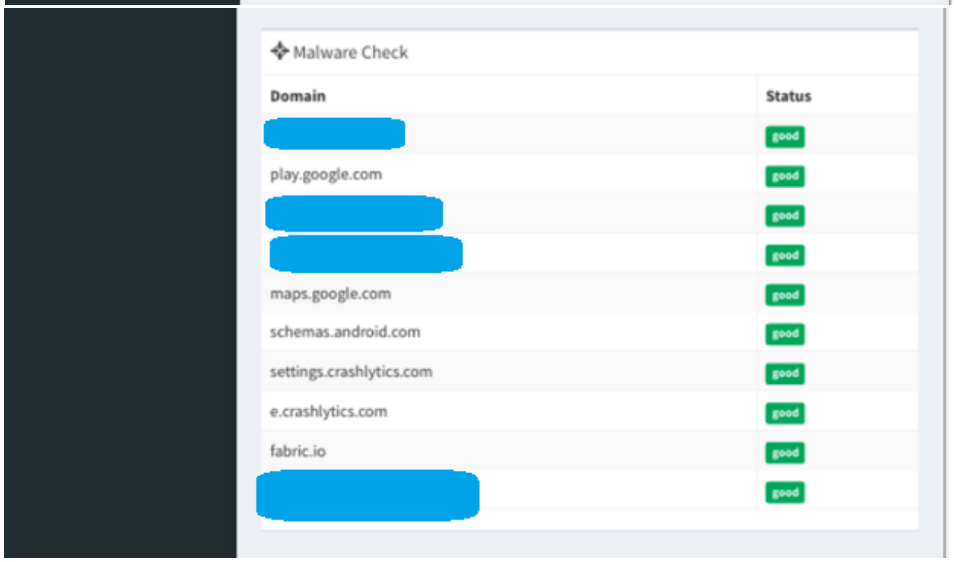


IP Address disclosure	warning	4.3	CWE-200	b/a/a/a/m.java b/a/a/a/c.java com/crashlytics/android/Crashlytics.java com/crashlytics/android/beta/Beta.java com/crashlytics/android/answers/Answers.java com/crashlytics/android/core/CrashlyticsCore.j
App can read/write to External Storage. Any App can read data written to External Storage.	high	5.5	CWE-276	b/a/a/a/d.java
Files may contain hardcoded sensitive informations like usernames, passwords, keys etc.	high	7.4	CWE-312	b/a/a/a/a/b/a.java com/crashlytics/android/beta/AbstractCheckFo com/crashlytics/android/answers/SessionEvent com/crashlytics/android/answers/SessionEvent com/crashlytics/android/core/PreferenceManag [redacted]
This App may have root detection capabilities.	secure	0		b/a/a/a/a/b/i.java [redacted]
The App uses an insecure Random Number Generator.	high	7.5	CWE-330	b/b/e/j/s.java com/crashlytics/android/answers/RandomBack [redacted]
SHA-1 is a weak hash known to have hash collisions.	high	5.9	CWE-327	[redacted]



APKiD Analysis

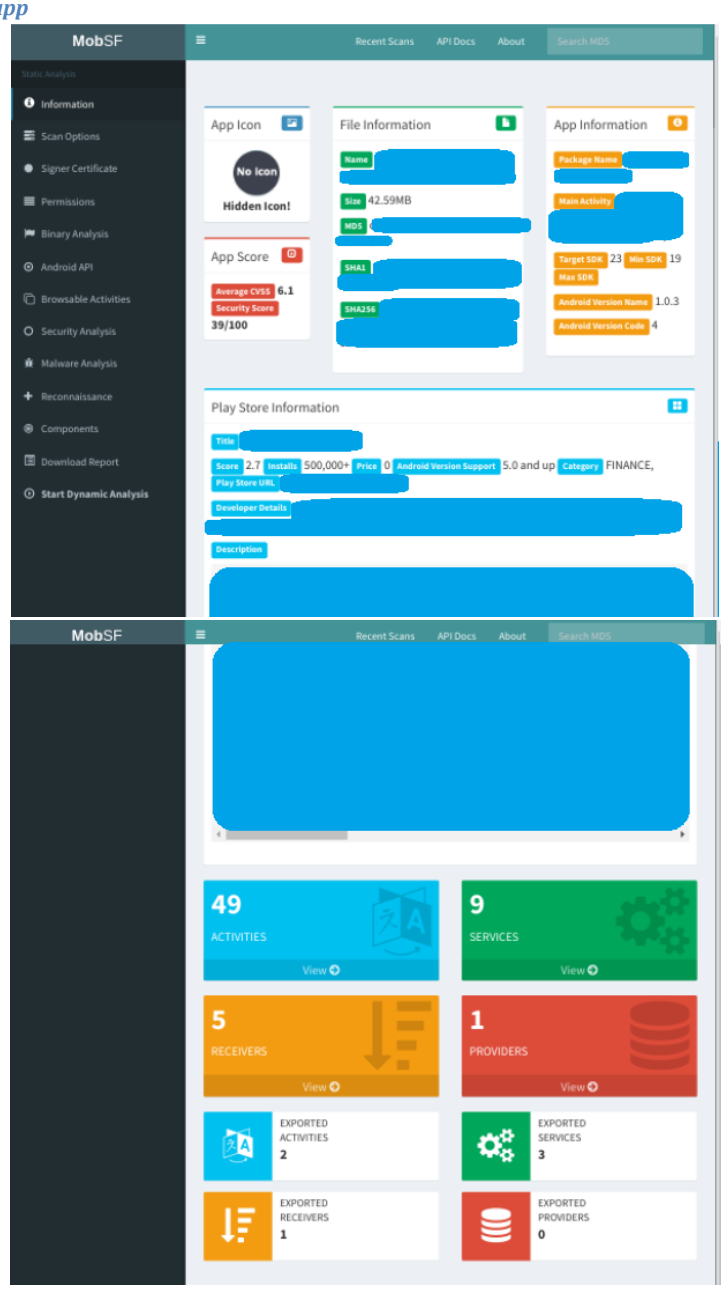
FILE	FINDINGS	DETAILS
classes2.dex	Compiler	dx
classes.dex	FINDINGS	DETAILS
	Compiler	dx
	Anti Debug Code	Debug.isDebuggerConnected() check
	Anti-VM Code	Build.MODEL check Build.MANUFACTURER check Build.PRODUCT check Build.TAGS check network operator name check possible vm check

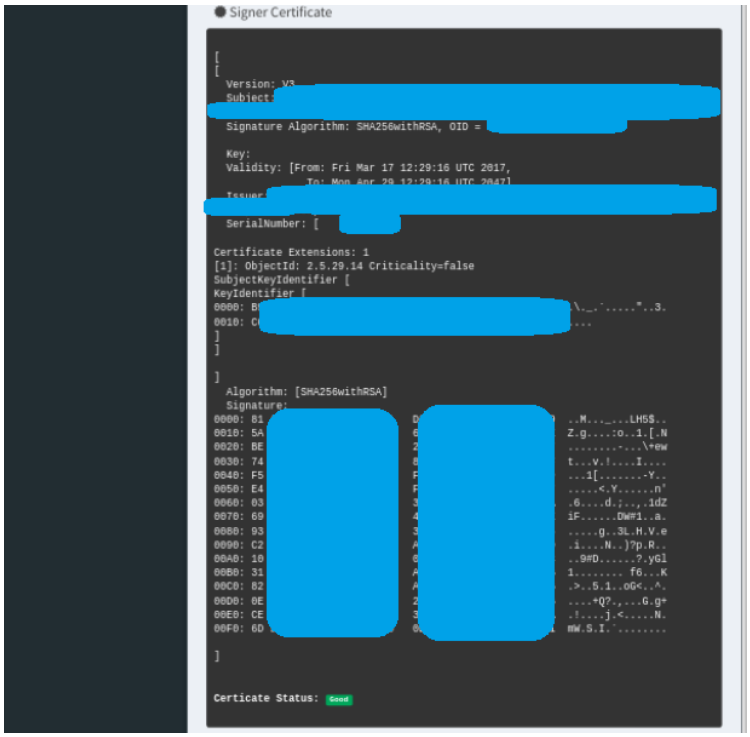


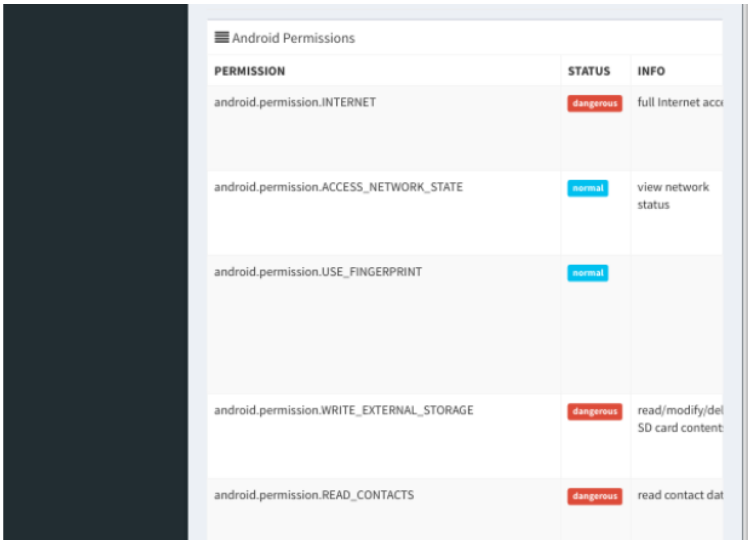
Malware Check

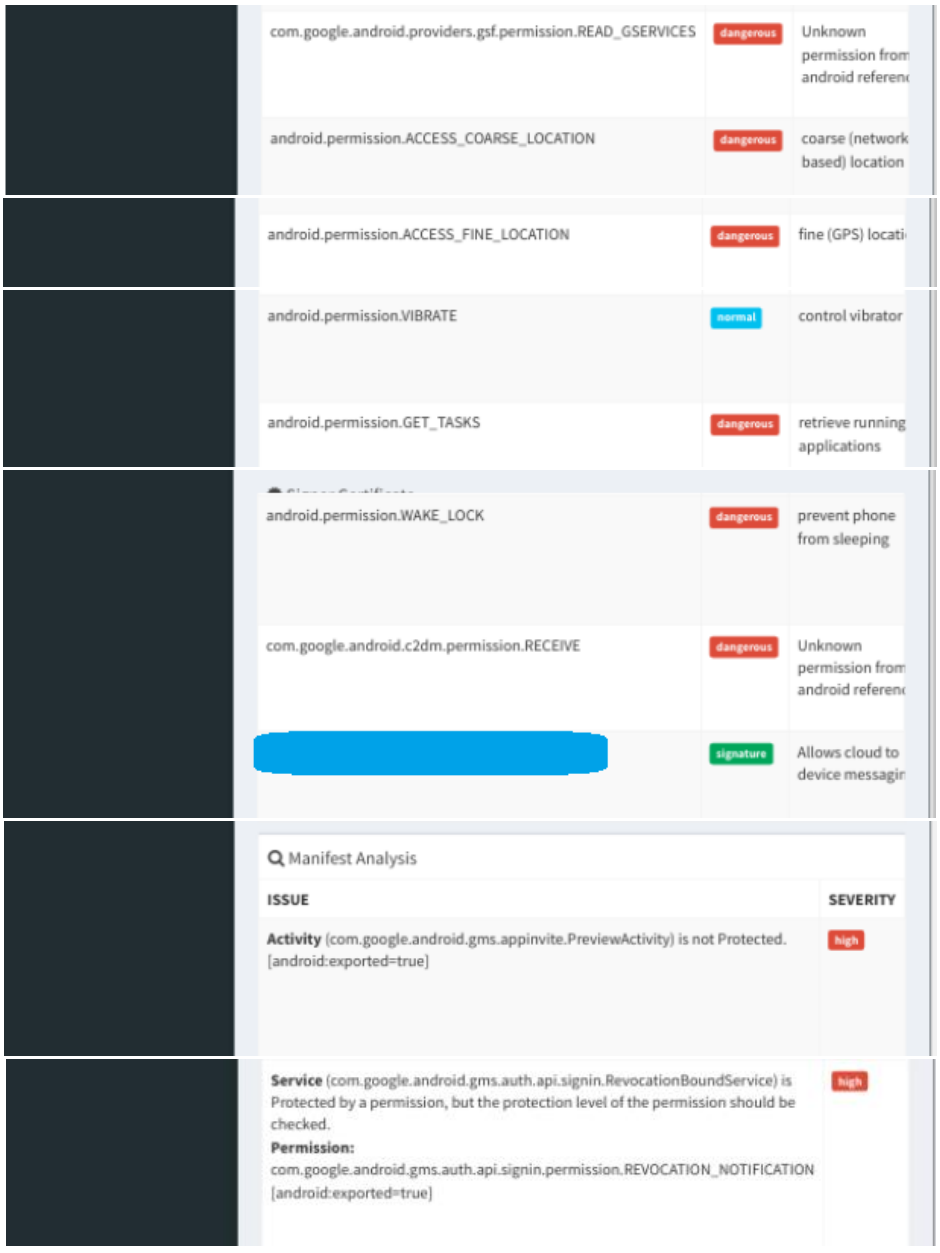
Domain	Status
[redacted]	good
play.google.com	good
[redacted]	good
[redacted]	good
maps.google.com	good
schemas.android.com	good
settings.crashlytics.com	good
e.crashlytics.com	good
fabric.io	good
[redacted]	good

W mobile app


MobSF
Recent Scans
API Docs
About
Search MD5
Information
Scan Options
Signer Certificate
Permissions
Binary Analysis
Android API
Browsable Activities
Security Analysis
Malware Analysis
Reconnaissance
Components
Download Report
Start Dynamic Analysis
App Icon
No Icon
Hidden Icon!
App Score
Average CVSS 6.1
Security Score
39/100
File Information
Name
Size 42.59MB
MD5
SHA1
SHA256
App Information
Package Name
Main Activity
Target SDK 23
Min SDK 19
Max SDK
Android Version Name 1.0.3
Android Version Code 4
Play Store Information
Title
Score 2.7
Installs 500,000+
Price 0
Android Version Support 5.0 and up
Category FINANCE,
Play Store URL
Developer Details
Description
MobSF
Recent Scans
API Docs
About
Search MD5
49
ACTIVITIES
View
9
SERVICES
View
5
RECEIVERS
View
1
PROVIDERS
View
EXPORTED ACTIVITIES
2
EXPORTED SERVICES
3
EXPORTED RECEIVERS
1
EXPORTED PROVIDERS
0


Signer Certificate
[
[
Version: V3
Subject:
Signature Algorithm: SHA256withRSA, OID =
Key:
Validity: [From: Fri Mar 17 12:29:16 UTC 2017,
To: Mon Apr 29 12:29:16 UTC 2047]
Issuer:
SerialNumber: [
Certificate Extensions: 1
[1]: ObjectId: 2.5.29.14 Criticality=false
SubjectKeyIdentifier [
KeyIdentifier [
0000: B
0010: C
]
]
]
Algorithm: [SHA256withRSA]
Signature:
0000: 81
0010: 5A
0020: BE
0030: 74
0040: F5
0050: E4
0060: 03
0070: 69
0080: 93
0090: C2
00A0: 10
00B0: 31
00C0: 82
00D0: 0E
00E0: CE
00F0: 6D
]
Certicate Status: Good


Android Permissions
PERMISSION
STATUS
INFO
android.permission.INTERNET
dangerous
full Internet acc
android.permission.ACCESS_NETWORK_STATE
normal
view network status
android.permission.USE_FINGERPRINT
normal
android.permission.WRITE_EXTERNAL_STORAGE
dangerous
read/modify/del SD card content
android.permission.READ_CONTACTS
dangerous
read contact dat

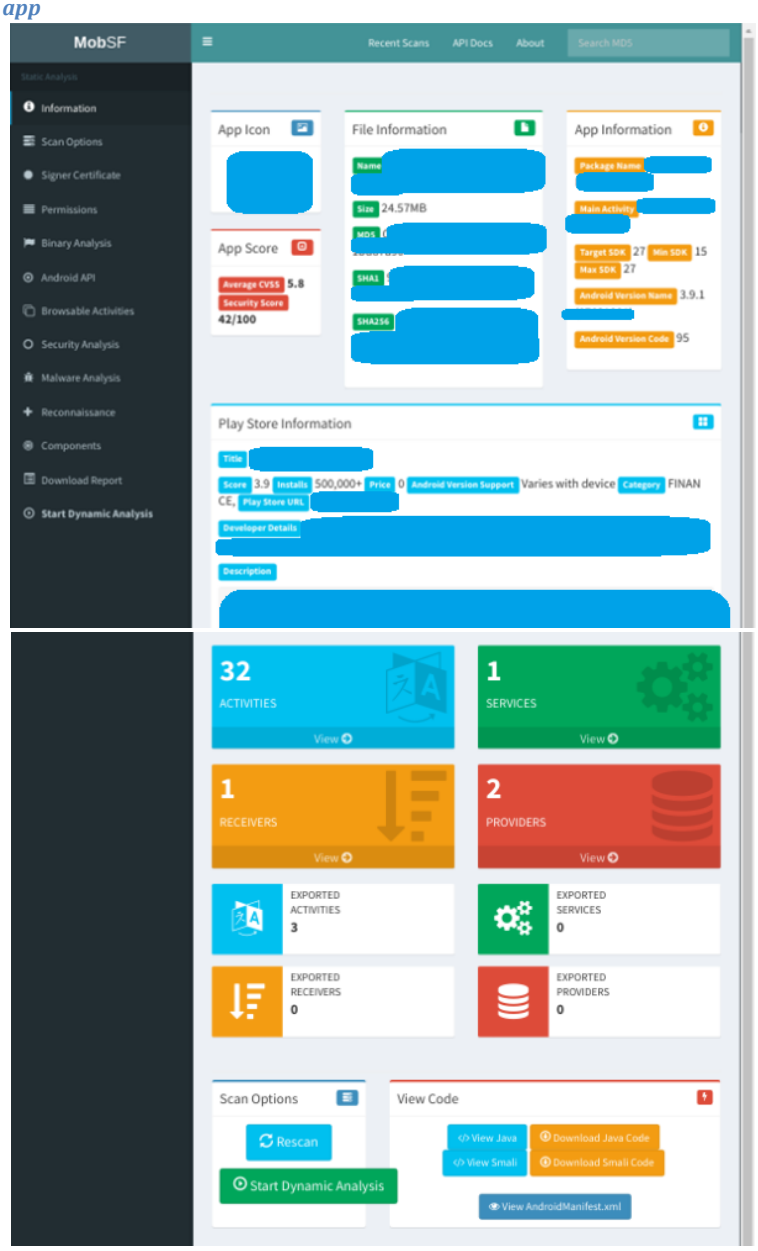


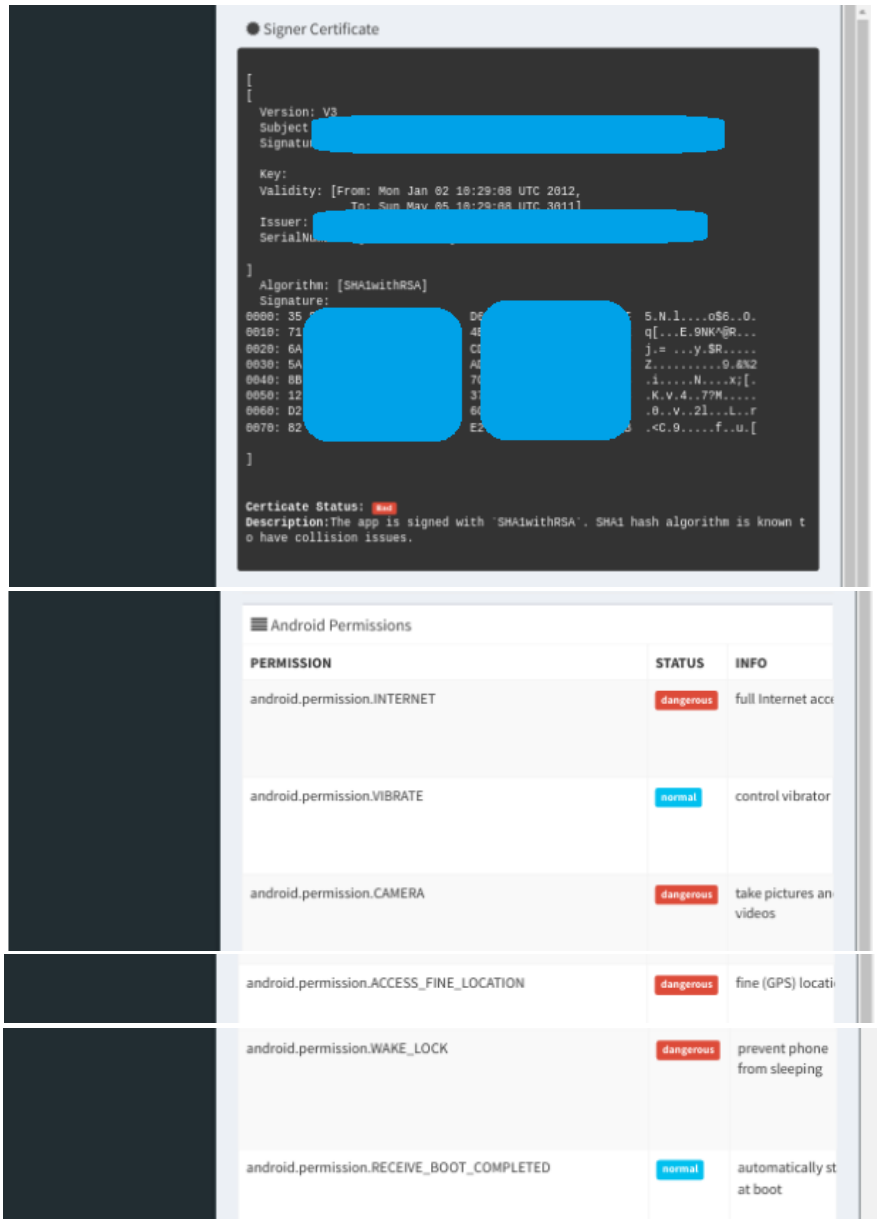
com.google.android.providers.gsf.permission.READ_GSERVICES	dangerous	Unknown permission from android referenc
android.permission.ACCESS_COARSE_LOCATION	dangerous	coarse (network based) location
android.permission.ACCESS_FINE_LOCATION	dangerous	fine (GPS) locati
android.permission.VIBRATE	normal	control vibrator
android.permission.GET_TASKS	dangerous	retrieve running applications
android.permission.WAKE_LOCK	dangerous	prevent phone from sleeping
com.google.android.c2dm.permission.RECEIVE	dangerous	Unknown permission from android referenc
[redacted]	signature	Allows cloud to device messagir

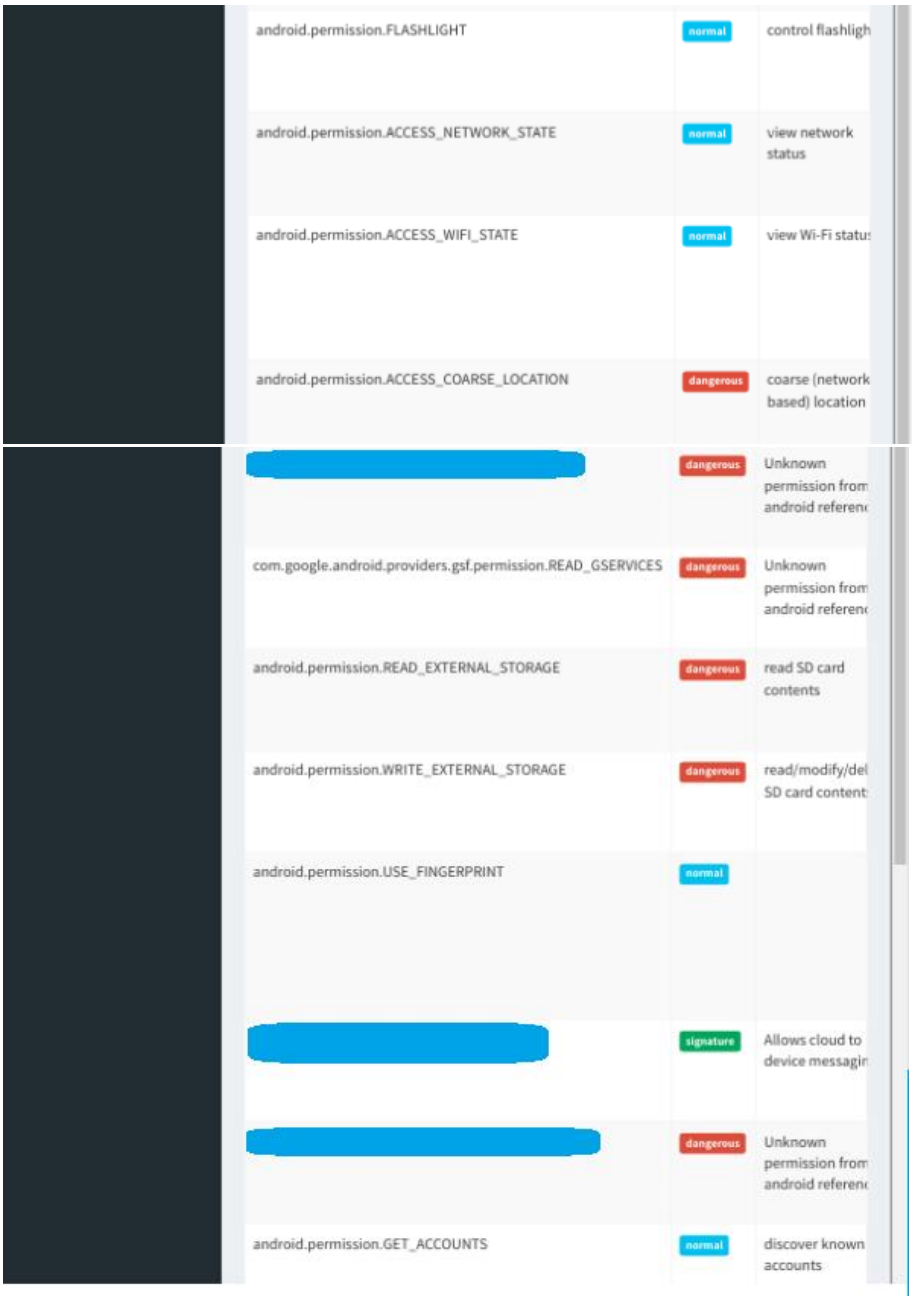
Manifest Analysis

ISSUE	SEVERITY
Activity (com.google.android.gms.appinvite.PreviewActivity) is not Protected. [android:exported=true]	high
Service (com.google.android.gms.auth.api.signin.RevocationBoundService) is Protected by a permission, but the protection level of the permission should be checked. **Permission:** com.google.android.gms.auth.api.signin.permission.REVOCATION_NOTIFICATION [android:exported=true]	high

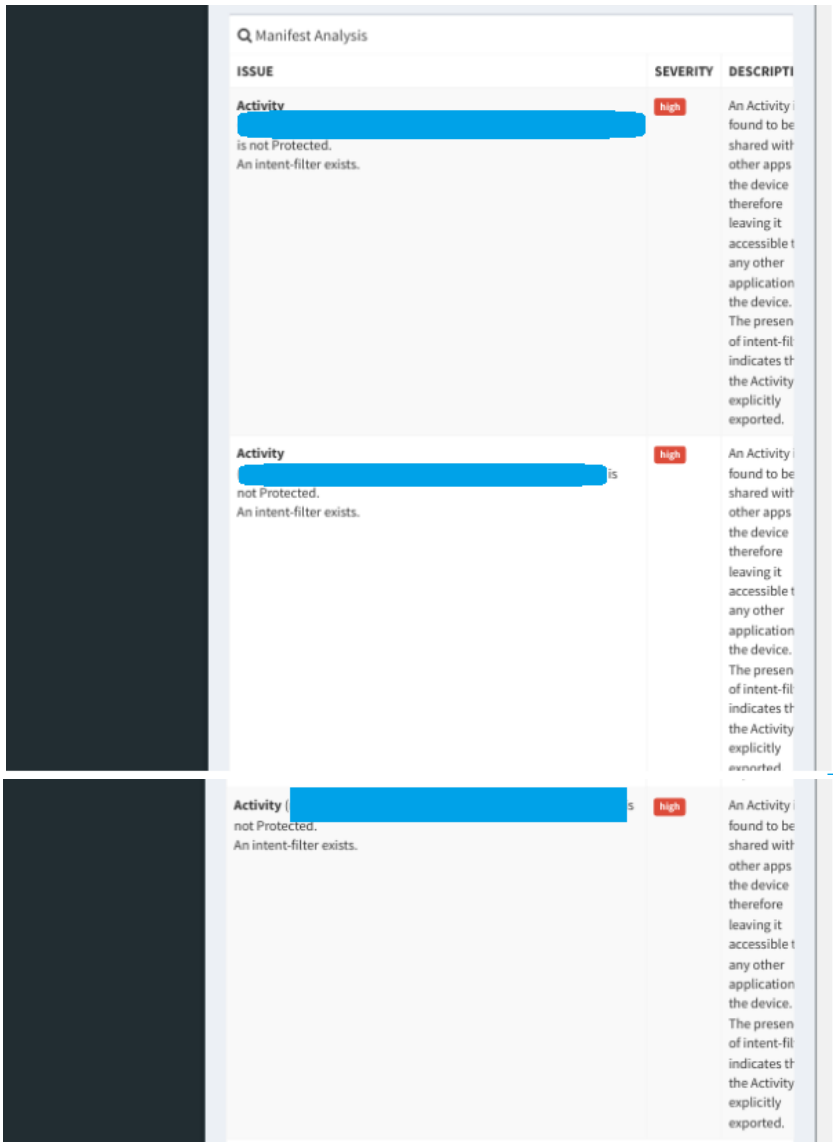
N mobile app

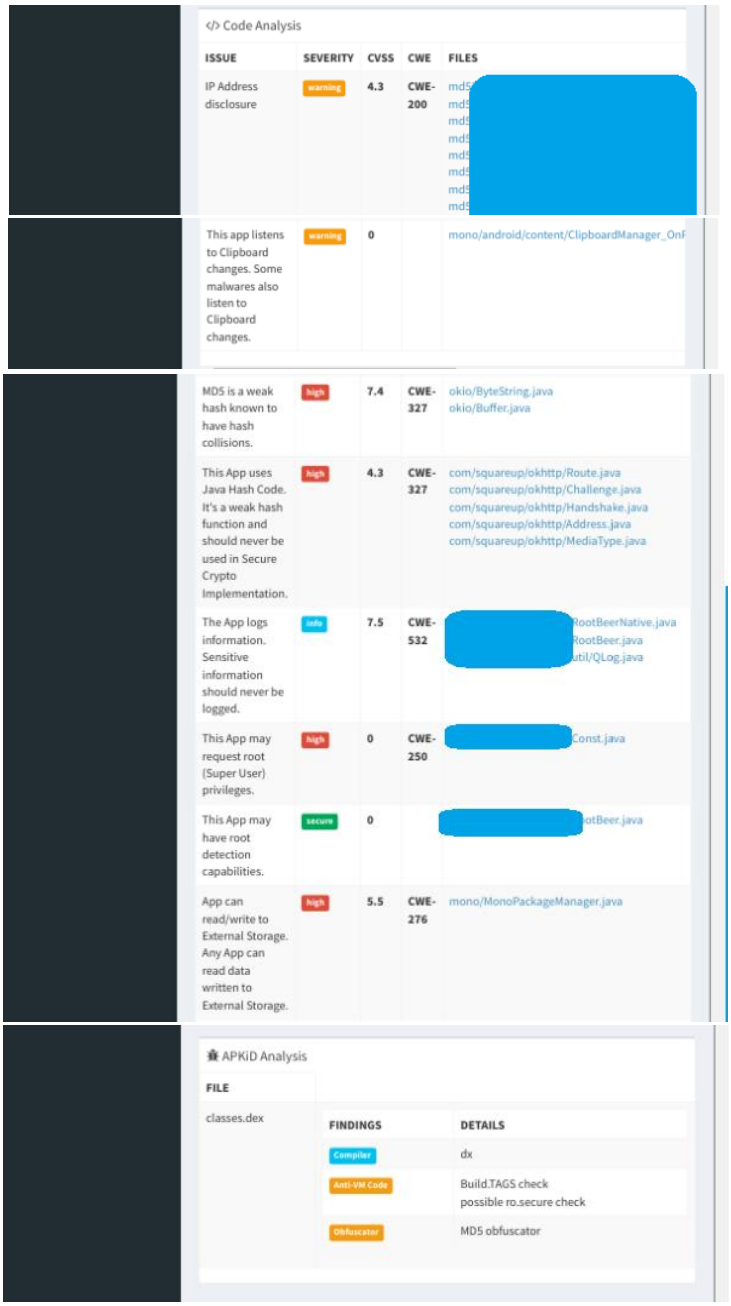

MobSF
Recent Scans
API Docs
About
Search MD5
Information
Scan Options
Signer Certificate
Permissions
Binary Analysis
Android API
Browsable Activities
Security Analysis
Malware Analysis
Reconnaissance
Components
Download Report
Start Dynamic Analysis
App Icon
File Information
App Information
App Score
Average CVSS 5.8
Security Score
42/100
Size 24.57MB
MD5
SHA1
SHA256
Package Name
Main Activity
Target SDK 27
Min SDK 15
Max SDK 27
Android Version Name 3.9.1
Android Version Code 95
Play Store Information
Title
Score 3.9
Installs 500,000+
Price 0
Android Version Support
Varies with device
Category
FINANCE
Play Store URL
Developer Details
Description
32
ACTIVITIES
View
1
SERVICES
View
1
RECEIVERS
View
2
PROVIDERS
View
EXPORTED ACTIVITIES
3
EXPORTED SERVICES
0
EXPORTED RECEIVERS
0
EXPORTED PROVIDERS
0
Scan Options
Rescan
Start Dynamic Analysis
View Code
View Java
Download Java Code
View Smali
Download Smali Code
View AndroidManifest.xml


Signer Certificate
Version: V3
Subject
Signatu
Key:
Validity: [From: Mon Jan 02 10:29:08 UTC 2012,
To: Sun May 05 10:29:08 UTC 3011]
Issuer:
SerialNu
Algorithm: [SHA1withRSA]
Signature:
0000: 35
0010: 71
0020: 6A
0030: 5A
0040: 8B
0050: 12
0060: D2
0070: 82
5.N.l....o$6..O.
q[...E.9NK^@R...
j.= ...y.$R.....
Z.........9.&%2
.i.....N....X;[.
.K.v.4..7?M.....
.0..v..2l...L..r
.<C.9.....f..u.[
Certicate Status:
Description:The app is signed with SHA1withRSA. SHA1 hash algorithm is known t
o have collision issues.
Android Permissions
PERMISSION
STATUS
INFO
android.permission.INTERNET
dangerous
full Internet acc
android.permission.VIBRATE
normal
control vibrator
android.permission.CAMERA
dangerous
take pictures an
videos
android.permission.ACCESS_FINE_LOCATION
dangerous
fine (GPS) locati
android.permission.WAKE_LOCK
dangerous
prevent phone
from sleeping
android.permission.RECEIVE_BOOT_COMPLETED
normal
automatically st
at boot



android.permission.FLASHLIGHT	normal	control flashligh
android.permission.ACCESS_NETWORK_STATE	normal	view network status
android.permission.ACCESS_WIFI_STATE	normal	view Wi-Fi statu
android.permission.ACCESS_COARSE_LOCATION	dangerous	coarse (network based) location
[redacted]	dangerous	Unknown permission from android referen
com.google.android.providers.gsf.permission.READ_GSERVICES	dangerous	Unknown permission from android referen
android.permission.READ_EXTERNAL_STORAGE	dangerous	read SD card contents
android.permission.WRITE_EXTERNAL_STORAGE	dangerous	read/modify/del SD card content
android.permission.USE_FINGERPRINT	normal	
[redacted]	signature	Allows cloud to device messagi
[redacted]	dangerous	Unknown permission from android referen
android.permission.GET_ACCOUNTS	normal	discover known accounts


Manifest Analysis
ISSUE
SEVERITY
DESCRIPTI
Activity
is not Protected.
An intent-filter exists.
high
An Activity
found to be
shared with
other apps
the device
therefore
leaving it
accessible
any other
application
the device.
The presen
of intent-fil
indicates th
the Activity
explicitly
exported.
Activity
is not Protected.
An intent-filter exists.
high
An Activity
found to be
shared with
other apps
the device
therefore
leaving it
accessible
any other
application
the device.
The presen
of intent-fil
indicates th
the Activity
explicitly
Activity
not Protected.
An intent-filter exists.
high
An Activity
found to be
shared with
other apps
the device
therefore
leaving it
accessible
any other
application
the device.
The presen
of intent-fil
indicates th
the Activity
explicitly
exported.



</> Code Analysis

ISSUE	SEVERITY	CVSS	CWE	FILES
IP Address disclosure	warning	4.3	CWE-200	[redacted]
This app listens to Clipboard changes. Some malwares also listen to Clipboard changes.	warning	0		mono/android/content/ClipboardManager_Onf
MD5 is a weak hash known to have hash collisions.	high	7.4	CWE-327	okio/ByteString.java okio/Buffer.java
This App uses Java Hash Code. It's a weak hash function and should never be used in Secure Crypto Implementation.	high	4.3	CWE-327	com/squareup/okhttp/Route.java com/squareup/okhttp/Challenge.java com/squareup/okhttp/Handshake.java com/squareup/okhttp/Address.java com/squareup/okhttp/MediaType.java
The App logs information. Sensitive information should never be logged.	info	7.5	CWE-532	[redacted]RootBeerNative.java RootBeer.java util/QLog.java
This App may request root (Super User) privileges.	high	0	CWE-250	[redacted]Const.java
This App may have root detection capabilities.	secure	0		[redacted]otBeer.java
App can read/write to External Storage. Any App can read data written to External Storage.	high	5.5	CWE-276	mono/MonoPackageManager.java

APKiD Analysis

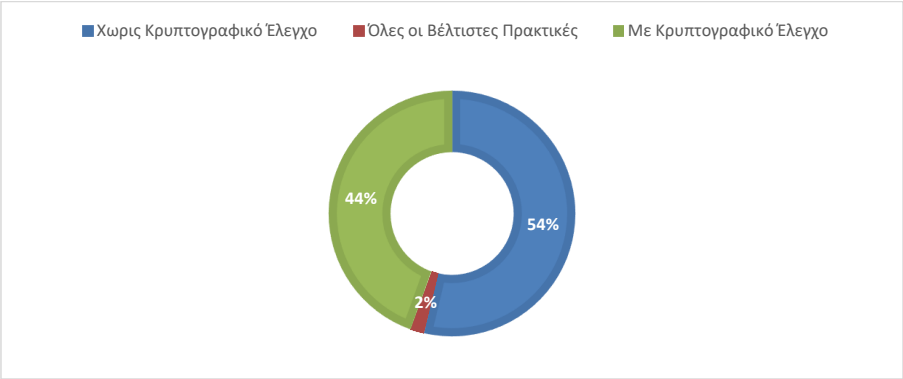
FILE	FINDINGS	DETAILS
classes.dex	Compiler	dx
	Anti-VM Code	Build.TAGS check possible ro.secure check
	Obfuscator	MD5 obfuscator

Σε ένα τελευταίο πείραμα, θα επιχειρήσουμε μια τελευταία static analysis του κώδικα των εφαρμογών χρησιμοποιώντας το εργαλείο που αναπτύχθηκε στα πλαίσια της σχετικής δημοσίευσης των Bianchi et al.[5]

Με βάση τη μελέτη της ομάδας ερευνητών, διαπιστώθηκε πως ένα τραγικά υψηλό ποσοστό Android Developers δεν υλοποιεί ορθά το fingerprint API ή την κρυπτογράφηση και αποθήκευση κωδικών του, με μεγάλες επιπτώσεις στην ασφάλεια. Συγκεκριμένα, τα αποτελέσματα της συστηματικής ανάλυσης τους σε μεγάλο πλήθος εφαρμογών έδειξαν ότι όχι μόνο δεν ακολουθούνται οι βέλτιστες πρακτικές αλλά, επιπλέον λόγω του πόσο νέα είναι αυτή η τεχνολογία στις κινητές συσκευές, επικρατεί σύγχυση ως προς το πού πρέπει να δοθεί έμφαση στην άμυνα. Πολλοί προγραμματιστές Android δε γνωρίζουν ίσως πως η λειτουργία του αισθητήρα και του API πραγματοποιείται σε επίπεδο λειτουργικού συστήματος (OS) και θα μπορούσε να χρησιμοποιηθεί ως πλεονέκτημα για τον επιτθέμενο ή τον αμυνόμενο σε ένα root attack.

Σύμφωνα με τα αποτελέσματα της ανάλυσης:

- Ποσοστό της τάξης του 53,69% δεν πραγματοποιεί κρυπτογραφικό έλεγχο για την επαλήθευση της ταυτότητας του χρήστη που άγγιξε τον αισθητήρα, τη στιγμή που μόλις ένα 20% δεν είχε αυτή τη δυνατότητα.
- Μόλις το 1,8% χρησιμοποιεί το API με τις βέλτιστες πρακτικές ασφαλείας και με το καλύτερο δυνατό αποτέλεσμα.
- Σε μετέπειτα έρευνα σε εφαρμογές εκ των οποίων το 80% αποθηκεύουν δεδομένα του χρήστη ή τον αυθεντικοποιούν απομακρυσμένα, διαπιστώθηκε ότι η αυθεντικοποίηση με δακτυλικό αποτύπωμα μπορεί να παρακαμφθεί πλήρως (fully bypassable).
- Υπάρχει ακόμα σύγχυση από τους προγραμματιστές σχετικά με τις βέλτιστες πρακτικές, ακόμα και αυτές που έχουν εκδοθεί από την Google, καθώς υπάρχουν αποκλίσεις και συγκρούσεις εντός των οδηγιών αλλά και υποτίμηση του προβλήματος από τους προγραμματιστές.



Βεβαίως, αν και ο πιο ασφαλής τρόπος να χρησιμοποιηθεί το συγκεκριμένο API είναι στην υπογραφή συναλλαγών, μόλις το 1,8% το χρησιμοποιεί κατ' αυτόν τον τρόπο. Ωστόσο, τα raw data του αισθητήρα (η εικόνα του αποτυπώματος) δε φεύγουν ποτέ από το TEE και δεν είναι προσπελάσιμα από κώδικα. Η υπογραφή αυτή πραγματοποιείται ως εξής: η υπηρεσία keymaster και το δακτυλικό αποτύπωμα εντός του TrustZone παράγουν εκ νέου ένα κρυπτογραφικό κλειδί, ενώ εξασφαλίζουν ότι ακόμα και επιτιθέμενος που έχει επιτύχει root privileges, δε μπορεί να εκμεταλλευτεί το κλειδί χωρίς το σωστό αποτύπωμα. Ο χρήστης αυθεντικοποιείται τοπικά στη συσκευή με το δακτυλικό του αποτύπωμα και έχει πλέον πρόσβαση στο κρυπτογραφικό κλειδί, το οποίο μπορεί να χρησιμοποιήσει.

Με βάση τα παραπάνω, συχνά το fingerprint API χρησιμοποιείται από εφαρμογές σε two-factor authentication. Το μοντέλο state-of-the-art υλοποίησης είναι το Universal Second Factor (U2F) που αποτελείται από δύο φάσεις: της εγγραφής, όπου παράγονται ζεύγη κλειδιών public και private, το πρώτο αποστέλλεται στον απομακρυσμένο server και το δεύτερο αποθηκεύεται με ασφάλεια στη τοπικά συσκευή, και της αυθεντικοποίησης όπου ο server στέλνει ένα challenge στη συσκευή που υπογράφει με το αποθηκευμένο private key. Όπως σε κάθε πλατφόρμα ανάπτυξης λογισμικού, έτσι και σε Android, υπάρχει το ρίσκο ότι οι προγραμματιστές δεν θα ακολουθήσουν το μοντέλο και θα υλοποιήσουν δικό τους ή θα υπάρξει κάποιο σφάλμα στην ανάπτυξη του.

Οι βασικές απειλές του API είναι παρόμοιες με άλλως μορφών αυθεντικοποίησης:

- Confused deputy

- Once for All

- Full Fingerprint Bypass

Δυστυχώς, λόγω σφαλμάτων στον πηγαίο κώδικα του εργαλείου που ανέπτυξε και χρησινοποίησε η συγκεκριμένη ομάδα, παρά τα πειράματα που επιχειρήσαμε δεν προέκυψε κάποιο αποτέλεσμα και η εφαρμογή μας επέστρεφε μόνο ERROR. Μετά από σύντομη έρευνα στο σχετικό repository του προγράμματος στο GitHub, διαπιστώθηκε πως και άλλοι χρήστες έχουν αντιμετωπίσει το ίδιο πρόβλημα. Συνεπώς μπορούμε μόνο να βασιστούμε στα συμπεράσματα της ομάδας όπως αυτά αναφέρονται στο συνοδευτικό paper[5].

Commented [Az10]: FIGURE

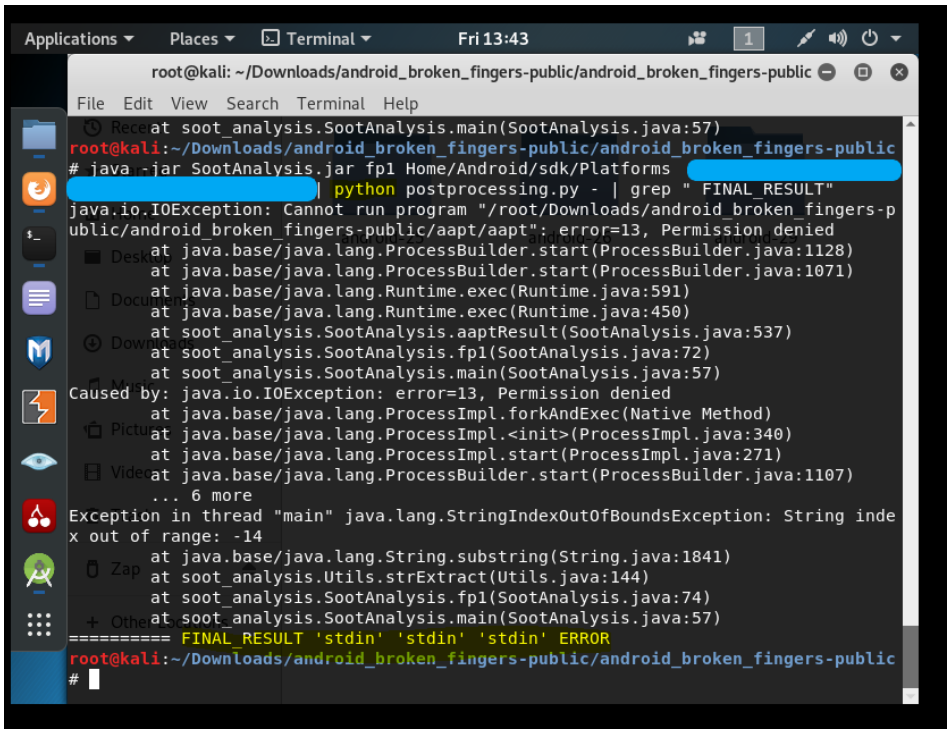


Figure 55 – Broken Fingers Static Analysis Tool

Όσον αφορά το κομμάτι του **dynamic analysis**, χρησιμοποιήσαμε την εφαρμογή drozer, και μια φυσική συσκευή (smartphone), για να εκτελέσουμε μια σειρά από πειράματα. Το εργαλείο drozer έρχεται με ένα πρόγραμμα για τον υπολογιστή και μια εφαρμογή client για τη συσκευή την οποία θα επιχειρήσουμε να «χτυπήσουμε» για να εντοπίσουμε ευπάθειες.

Figure 56 - Drozer

Εγκαταστήσαμε το drozer client app στο κινητό τηλέφωνο που χρησιμοποιούμε και συνδέσαμε τη συσκευή με τον υπολογιστή που εκτελούσε το πρόγραμμα μέσω USB. Στη συνέχεια, με τις σχετικές εντολές[4] εκτελέσαμε μια σειρά από πειράματα-επιθέσεις στις υπό μελέτη εφαρμογές. Ενδεικτικά, θα σας αναφέρουμε τη διαδικασία που ακολουθήθηκε και τις επιθέσεις που εκτελέσθηκαν αλλά για νομικούς και ηθικούς λόγους, δε θα παρουσιάσουμε τη διαδικασία με τις υπό μελέτη εφαρμογές ονομαστικά, παρά μόνο τα ευρήματα που προέκυψαν.

[4] Παρατίθενται στο Παράρτημα Β

Τα πειράματα που εκτελέσθηκαν είναι:

- Επιλογή στόχου (attack surface)
- Αποτίμηση και εκμετάλλευση ευπαθειών στα accessible app activities
- Αποτίμηση και εκμετάλλευση ευπαθειών στους content providers της εφαρμογής
- Αποτίμηση και εκμετάλλευση ευπαθειών στους broadcast receivers της εφαρμογής
- Αποτίμηση και εκμετάλλευση ευπαθειών στις υπηρεσίες της εφαρμογής
- Authentication Bypass
- SQL Injection & Data extraction

Κατ' ακρίβεια, παρατηρήθηκε πως σε καμία από τις υπό μελέτη εφαρμογές δεν επιτεύχθηκε με επιτυχία κάποια επίθεση, καθώς δεν επέστρεφαν κάποιο αποτέλεσμα. Η πλειονότητα των εφαρμογών δεν αντιδρούσαν στις πρώτες επιθέσεις της λίστας, ούτε και επέστρεφαν κάποια πληροφορία. Στις πιο πρακτικές επιθέσεις, ονοματικά στο authentication bypass, οι τρεις εκ των εφαρμογών (A, N, W) δεν πραγματοποίησαν είσοδο και έκλεισαν (abrupt termination) για να μην επιτρέψουν στον επιτιθέμενο να πραγματοποιήσει την παραβίαση. Παρότι εντοπίσαμε τη συγκεκριμένη άμυνα, δε μπορέσαμε να βγάλουμε περισσότερα συμπεράσματα όσον αφορά τη βιομετρική αυθεντικοποίηση. Στην τέταρτη εφαρμογή (E) πραγματοποιήσαμε την επίθεση και στα δύο διαθέσιμα activities του attack surface. Συγκεκριμένα η επίθεση επιχειρή να αποκτήσει πρόσβαση στις λειτουργίες της εφαρμογής χωρίς να προηγηθεί αυθεντικοποίηση. Στο πρώτο (main activity) η εφαρμογή έμεινε σε κατάσταση buffering της αρχικής οθόνης επ αόριστον. Στο δεύτερο διαθέσιμο activity, η εφαρμογή άνοιξε μια κενή, λευκή οθόνη, μην επιτρέποντας στο χρήστη να πραγματοποιήσει καμία απολύτως ενέργεια, κοινώς επετεύχθη denial of service (DoS).

Ασφαλώς, η λίστα των πειραμάτων δεν είναι εξαντλητική, και με χρήση επιπλέον εργαλείων (όπως είναι και το MobSF με τη λειτουργία του dynamic analysis) πιθανόν να μπορούσαν να εξαχθούν περισσότερα συμπεράσματα. Δεδομένων όσων διαπιστώσαμε, μπορούμε να πούμε πως οι εφαρμογές έχουν μια υποτυπώδη άμυνα στις επιθέσεις αλλά και τις κακές πρακτικές σε client side επίπεδο. Δεν μπορούμε ωστόσο να βγάλουμε επαρκή συμπεράσματα όσον αφορά τη βιομετρική αυθεντικοποίηση. Θα ήταν μελλοντικά χρήσιμη η ανάπτυξη σχετικού εργαλείου ανάλυσης και όσον αφορά το runtime περιβάλλον, για διασφάλιση της ορθής εφαρμογής και ασφαλούς λειτουργίας.

3.5 Network Analysis

Στο τελευταίο κομμάτι των πειραμάτων μας θα επιχειρήσουμε να εκτελέσουμε και ορισμένες επιθέσεις δικτύου client side, προκειμένου να διαπιστώσουμε πόσο ορθά εκτελείται η βιομετρική αυθεντικοποίηση ως προς τις αρχές του FIDO, αν δηλαδή αποστέλλονται στοιχεία όπως κρυπτογραφικά κλειδιά μέσω δικτύου ή ορθώς παραμένουν στη συσκευή.

Συγκεκριμένα, οι επιθέσεις που θα επιχειρήσουμε να εκτελέσουμε είναι:

- Network Scanning

- Proxying
- SSL Bypass

Για το πρώτο μέρος των δοκιμών χρησιμοποιήσαμε τα γνωστά προγράμματα Wireshark και BurpSuite Community Edition. Δεν παρατηρήθηκε κάποια πληροφορία πέραν των διευθύνσεων που αποστέλλονται οι πληροφορίες, δηλαδή όλα τα δεδομένα μεταφέρονται στο δίκτυο κρυπτογραφημένα.

Για τα proxies δοκιμάστηκε ξανά το BurpSuite αλλά και τα προγράμματα OWASP ZAP και Charles Proxy. Το configuration γίνεται ως εξής:

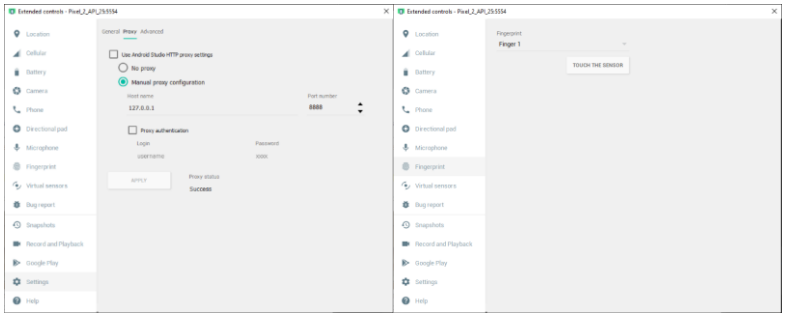


Figure 57 – Proxy Configuration

Εντοπίζουμε τις απαραίτητες πληροφορίες για τη χρήση του κάθε proxy, πχ. το port το οποίο χρησιμοποιούν και παρακολουθούν. Αν και μπορούν να αλλαχθούν από το χρήστη, τα default ports στα οποία «ακούει» κάθε proxy είναι γνωστά στο σχετικό documentation. Ο BurpSuite χρησιμοποιεί το 8888, ο Charles το 8080 κ.ο.κ.

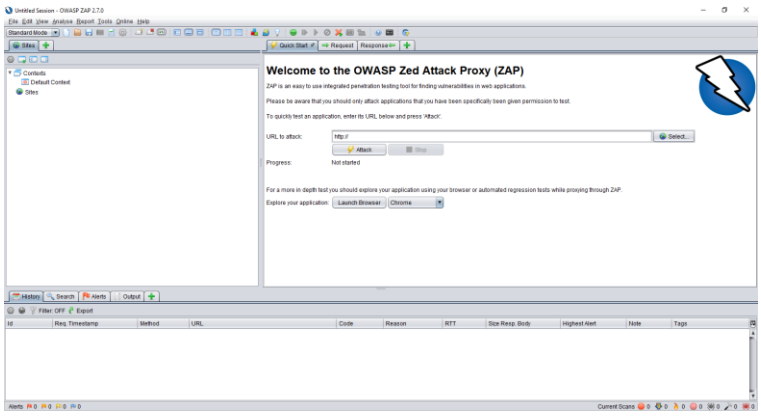


Figure 58 – OWASP ZAP Configuration

Στο client περιβάλλον που χρησιμοποιούμε (εν προκειμένω το built-in emulator του Android Studio) επιλέγουμε από τις ρυθμίσεις δικτύου τη χρήση proxy, δίνουμε την κατάλληλη διεύθυνση IP (πχ. localhost 127.0.0.1) και το σχετικό port, όπως φαίνεται στην εικόνα 57.

Αρχικά, για το virtual environment που χρησιμοποιούμε, προσπαθήσαμε να χρησιμοποιήσουμε hardware αισθητήρα δακτυλικού αποτυπώματος καθώς προφανώς δεν υπάρχει αντίστοιχη λειτουργία στους android emulators. Διαπιστώθηκε ωστόσο ότι στην ενημερωμένη έκδοση και με τα κατάλληλα μοντέλα κινητών συσκευών στο Android Virtual Device manager του Android Studio, υπάρχει ενσωματωμένη η λειτουργία «εικονικού» δακτυλικού αποτυπώματος, και μάλιστα με δυνατότητα 10 διαφορετικών αποτυπωμάτων ως input.

Στη συνέχεια φαίνεται η ρύθμιση του Charles Proxy για τη σάρωση δικτύου.

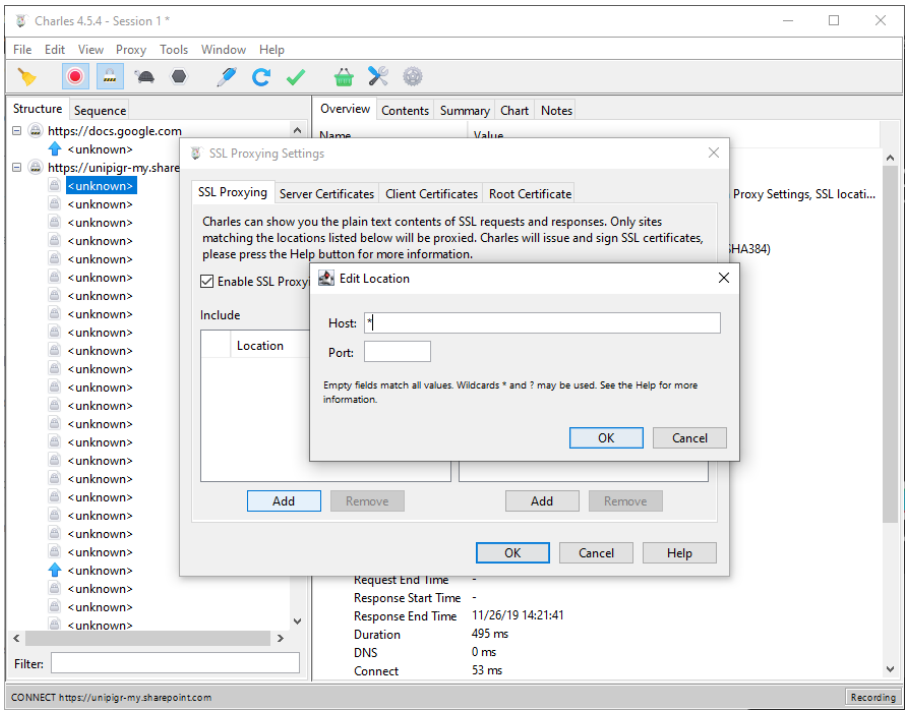


Figure 59 – Charles Proxy Configuration

Αφού γίνουν οι απαραίτητες ρυθμίσεις, εκτελούμε σάρωση.

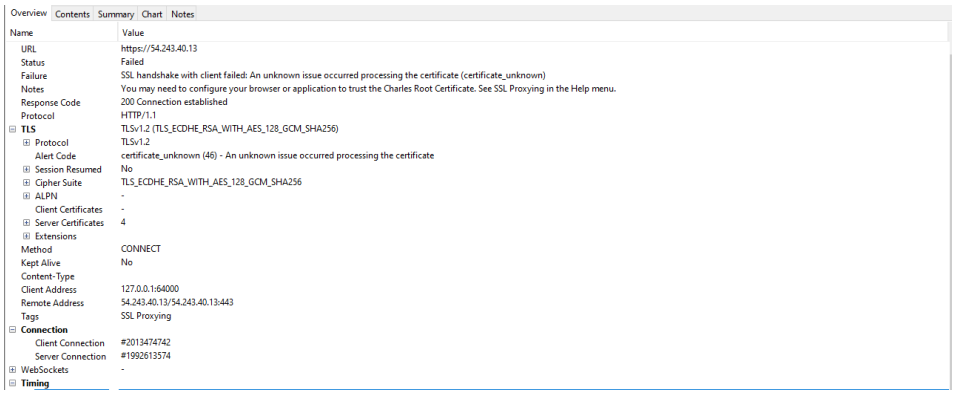


Figure 60 – Charles Proxy Scan Overview

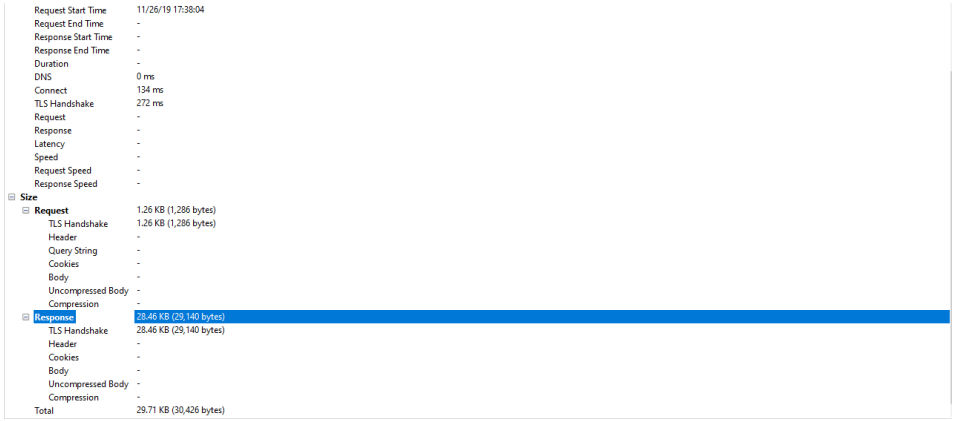


Figure 61 – Charles Proxy Scan – Network Request

Ωστόσο, το απλό proxying επίσης δεν επέφερε κάποιο αποτέλεσμα καθώς οι εφαρμογές μας εντοπίζουν την ύπαρξη proxy και δεν ανοίγουν καν αφού θεωρούν το περιβάλλον δικτύου επικίνδυνο. Επιχειρώντας SSL pinning, εγκαταστήσαμε τα κατάλληλα πιστοποιητικά του εκάστοτε proxy στα περιβάλλοντα σάρωσης.

Τα αποτελέσματα είχαν ως εξής:

- Για το Burp Suite, επεστράφη μήνυμα μη ασφαλούς δικτύου και οι εφαρμογές δε συνδέονταν στο proxy
- Για το Charles δεν υπήρχε κάποια προειδοποίηση ασφαλείας, και οι εφαρμογές άνοιγαν κανονικά. Ωστόσο, όσον αφορλα τη σύνδεση και αυθεντικοποίηση:

- Η Ε εκτελούσε buffering και συνεχώς επανεκκινούνταν για ένα διάστημα, μην επιτρέποντας την αυθεντικοποίηση, ενώ μετά την εισαγωγή των στοιχείων (όνομα χρήστη και κωδικός πρόσβασης) έμεινε σε κατάσταση buffering. Όσον αφορά τη σάρωση, επέδτρεψε περιεχόμενο “forbidden”, εννοώντας ότι δεν επιτρέπεται ούτε η σάρωση και προβολή δεδομένων αλλά ούτε και η διέλευση τους από το δίκτυο. Όταν εξαιρέθηκε το συγκεκριμένο domain από τη σάρωση, η σύνδεση πραγματοποιήθηκε κανονικά.
- Οι υπόλοιπες εφαρμογές δεν πραγματοποίησαν σύνδεση, πχ. η Α δεν άνοιξε πέρα από την αρχική οθόνη, μην επιτρέποντας στο χρήστη να εκτελέσει αυθεντικοποίηση.

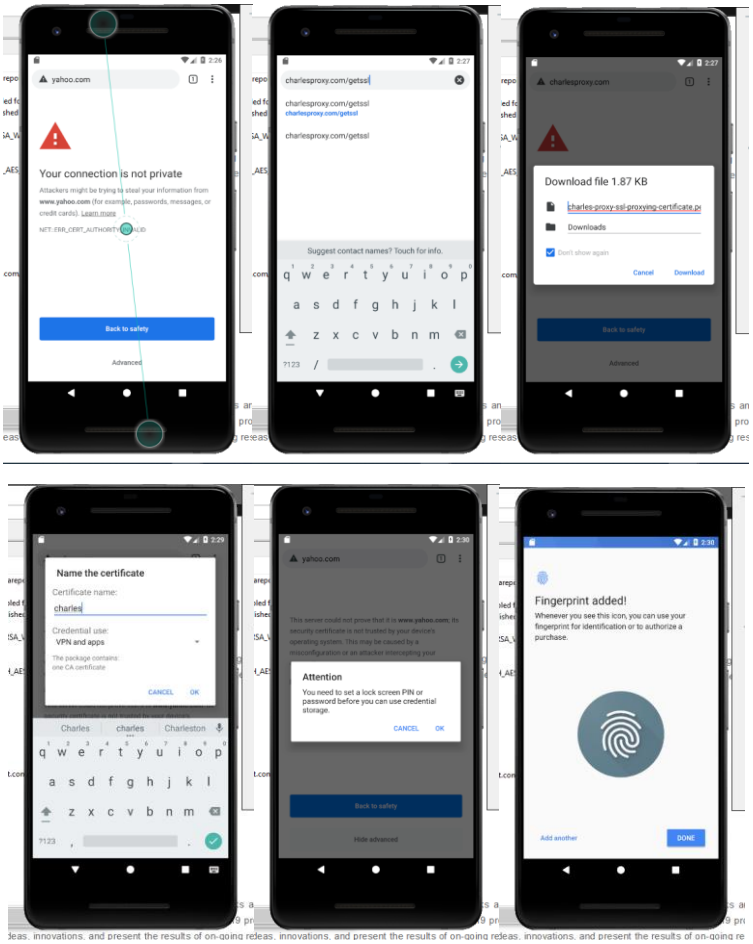


Figure 62 – Εκτέλεση πειραμάτων

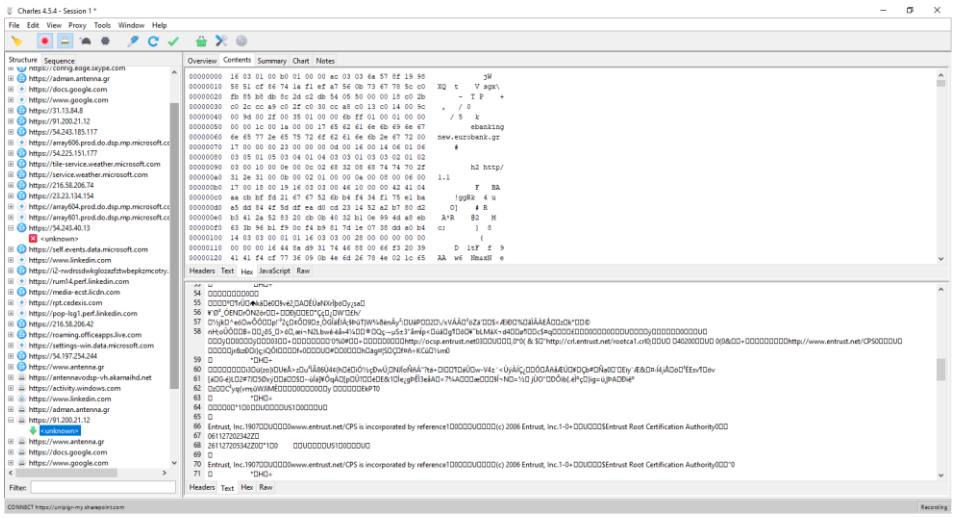


Figure 63 – Charles Proxy Scan – E mobile app

Κεφάλαιο 4ο – Συμπεράσματα και Μελλοντικές Έρευνες

Με το πέρας των πειραμάτων και αναλύσεων, στα πλαίσια της παρούσας διλωματικής εργασίας, δε μπορούν να εξαχθούν καθοριστικά συμπεράσματα για την ασφάλεια των τεχνολογιών βιομετρικής αυθεντικοποίησης σε έξυπνα κινητά Android. Βασιστήκαμε στη σχετική μεθοδολογία κατά OWASP. Ωστόσο, από το ενδεικτικό δείγμα εφαρμογών που μελετήθηκαν, καθώς και τη μεγάλη ποικιλία και εύρος πειραμάτων που εκτελέσθηκαν μπορούμε να συμπεράνουμε πολλά για την αντιμετώπιση της συγκεκριμένης τεχνολογίας από τους προγραμματιστές, τη συμμόρφωση με πρότυπα καθώς και τις άμυνες ενάντια σε επιθέσεις σε εφαρμογές που διαχειρίζονται πολύτιμα δεδομένα χρήστη. Συνοψίζοντας τα ευρήματα που προέκυψαν από τις διάφορες αναλύσεις, προκύπτουν ακόμα εύλογα προτάσεις για μελλοντική μελέτη και έρευνα επί του αντικειμένου, προκειμένου να δημιουργηθεί μια πιο πλήρης και ξεκάθαρη εικόνα του πεδίου μελέτης.

Εφαρμογή	FIDO API	Broken Fingers	Static Analysis	Dynamic Analysis	Network Analysis
A	✕	ERROR	36/100	Not allowed	Will not start
E	✕	ERROR	39/100	DoS when attempting Authentication Bypass.	Will not start
N	✕	ERROR	42/100	Not allowed	Will not start
W	✕	ERROR	39/100	Not allowed	Detects proxy and restarts, bypassed after a while.

Πίνακας 1 - Συγκεντρωτικός Συγκριτικός Πίνακας Αποτελεσμάτων

Από το reverse engineering των εφαρμογών μας προκύπτει πως καμία από τις υπό μελέτη εφαρμογές δεν ενσωματώνει το υπάρχον FIDO API στον κώδικα της βιομετρικής της αυθεντικοποίησης. Ωστόσο, είναι πιθανόν η αρχιτεκτονική που ακολούθησαν οι developers να συμμορφώνεται με τις βασικές αρχές του FIDO, δηλαδή να εφαρμόζει public key cryptography, τοπική αποθήκευση της βιομετρικής πληροφορίας κ.ο.κ.

Η μελέτη της δημοσίευσης των Bianchi et al. επί του αντικειμένου μας προσέφερε πολύτιμα δεδομένα σχετικά με τη γενικότερη αντιμετώπιση του fingerprint API στην αγορά, μελετώντας την υλοποίηση του σε μεγάλο αριθμό εφαρμογών και παρουσιάζοντας στατιστικά αποτελέσματα τα οποία είναι μάλλον απογοητευτικά, καθώς οι περισσότεροι προγραμματιστές αμελούν να ενισχύσουν την ασφάλεια της βιομετρικής αυθεντικοποίησης που ενσωματώνουν στις εφαρμογές τους. Δεν περιλαμβάνονται ισχυροί αλγόριθμοι κρυπτογράφησης ή ορθή υλοποίηση του χαρακτηριστικού του fingerprint API για χρήση «υπογραφής», όπως προτείνεται, ή αυθεντικοποίησης σε εφαρμογές. Αντ' αυτού, μόλις σε λιγότερο από 2% των εφαρμογών εφαρμόζονται όλες οι βέλτιστες πρακτικές, και πάνω από το 50% δεν χρησιμοποιεί κρυπτογράφηση

για να προστατέψει τα δεδομένα του. Όσον αφορά το εργαλείο static analysis που ανέπτυξε η ομάδα στα πλαίσια της εν λόγω δημοσίευσης, παρότι επιχειρήσαμε να το χρησιμοποιήσουμε για στοχευμένη ανάλυση της βιομετρικής αυθεντικοποίσης και γενικώς του fingerprint API στις τέσσερις υπο μελέτη εφαρμογές, στάθηκε αδύνατη η λειτουργία του καθώς, όπως διαπιστώσαμε και από τη σχτική δραστηριότητα στο GitHub του έργου, φαίνεται να υπάρχει σφάλμα στον ανοικτό κώδικα του εργαλείου Broken Fingers που ανέπτυξε και χρησιμοποίησε η ερευνητική ομάδα. Συνεπώς, δε μπορέσαμε να εξάγουμε δικά μας συμπεράσματα.

Στη συνέχεια, χρησιμοποιήσαμε δύο εργαλεία ανοικτού κώδικα για γενικότερο static analysis: το QARK και το MobSF. Από το πρώτο φαίνεται να εμφανίζονται κίνδυνοι και σημαντικές ευπάθειες στον κώδικα της εφαρμογής Ε. Συγκεκριμένα, ορισμένα activities της εφαρμογής, μεταξύ αυτών και το αρχικό (LaunchActivity) και η σύνδεση με τη βάση δεδομένων (network.push.firebase) έχουν ευπάθεια στην εξαγωγή tags, γεγονός που τις καθιστά ευπαθείς σε επιθέσεις injection και διαρροή δεδομένων. Επίσης, από το AndroidManfest της εφαρμογής προκύπτει ότι η εφαρμογή επιτρέπει το back up δεδομένων μέσω ADB, και σε περίπτωση που ενεργοποιηθεί το USB debugging στη συσκευή, ενέχεται κίνδυνος κλοπής δεδομένων. Υπενθυμίζεται πως η εφαρμογή Α δε μπόρεσε να αναλυθεί με QARK, ούτε να πραγματοποιηθεί manual reverse engineering, ωστόσο δε μπορέσαμε να συμπεράνουμε εάν ο λόγος ήταν πρόβλημα στο αρχείο .apk ή άμυνα της εφαρμογής στο reverse engineering. Με το MobSF αναλύθηκαν και οι τέσσερις εφαρμογές, ως προς το σύνολο του κώδικα τους. Στάθηκε δυνατό μόνο το static analysis παρότι το εργαλεί παρέχει τη δυνατότητα και για dynamic analysis, λόγω παύσης υποστήριξης ορισμένων λειτουργιών του εργαλείου. Τα αποτελέσματα στο εξαχθέν report δείνουν ένα ποσοστό της τάξης του περίπου 40% για όλες τις εφαρμογές ως προς την ασφάλεια του κώδικα. Δεν δίνονται όμως περισσότερες πληροφορίες σχετικά με τη βιομετρική αυθεντικοποίηση συγκεκριμένα.

Στο μέρος του dynamic analysis χρησιμοποιήθηκε το δημοφιλές εργαλείο drozer, και όλες οι δυνατότητες που παρέχει. Τα πειράματα εκτελέσθηκαν σε φυσική συσκευή (rooted) και εκτελέσθηκαν:

- Αποτίμηση και εκμετάλλευση ευπαθειών σε όλα τα διαθέσιμα activities
- Προσπάθεια authentication bypass (προσπέρασης του activity sign up/log in)
- Αποτίμηση και εκμετάλλευση ευπαθειών σε content providers, broadcast receivers & services
- SQL Injection
- Data extraction

Από τα πειράματα αυτά προέκυψε DoS στην εφαρμογή Ε κατά την προσπάθεια authentication bypass, αφού προσπερνώντας το activity εισόδου στο σύστημα, η εφαρμογή «κολλούσε» επ' αόριστον σε μια λευκή οθόνη. Οι υπόλοιπες εφαρμογές δεν παρουσίασαν κάποια ευπάθεια ή εύρημα, καθώς ανιχνεύοντας ύποπτη δραστηριότητα ή επισφαλές περιβάλλον, ζητούσαν αυθεντικοποίηση με στοιχεία e-banking και όχι PIN ή δακτυλικό αποτύπωμα.

Τέλος, ως προς το Network Analysis, επιχειρήσαμε απλό monitoring, proxying και SSL pinning/SSL bypass. Ωστόσο, δεδομένου ότι οι υπό μελέτη εφαρμογές είναι τραπεζικές, διαθέτουν ενσωματωμένες άμυνες σε επισφαλείς συνδέσεις. Όλες οι εφαρμογές δεν πραγματοποιούσαν καν εκκίνηση εάν εντόπιζαν πιστοποιητικό BurpSuite, δίνοντας και αντίστοιχο μήνυμα με push notification. Η βέλτιστη επιλογή αποδείχθηκε το Charles Proxy, όπου αν και δεν υπήρξε σχετικό μήνυμα, όλες οι εφαρμογές πλην τις W δεν πραγματοποίησαν εκκίνηση, ενώ η W παρότι εκτελούσε επαναλαμβανόμενα restarts τελικά λειτούργησε και επέτρεψε την αυθεντικοποίηση. Ωστόσο, το scanning μας δεν μπόρεσε να προβάλει κάποιο αποτέλεσμα ώστε να διαπιστώσουμε πώς εκτελειται η αυθεντικοποίηση (αν συμφωνεί με τα πρότυπα του FIDO αποθηκεύοντας τοπικά τα raw data του βιομετρικού και πραγματοποιώντας απομακρυσμένη αυθεντικοποίηση με public key cryptography.

Αξίζει να τονιστεί πως η παρούσα μελέτη πραγματοποιήθηκε με μικρό και ενδεικτικό δείγμα εφαρμογών, και δη του τραπεζικού τομέα, με ισχυρές άμυνες και μηχανισμούς ασφαλείας. Από την προτεινόμενη μεθοδολογία δεν πραγματοποιήθηκαν server attacks για νομικούς και πρακτικούς λόγους. Επιπλέον, χρησιμοποιήθηκε περιορισμένος αριθμός εργαλείων ανάλυσης, αποκλειστικά ανοικτού κώδικα αλλά άκρως αποτελεσματικών και συστημένων από μελετητές ασφαλείας και security analysts. Ωστόσο το μόνο εργαλείο που επικεντρώνεται στη λειτουργία και υλοποίηση της βιομετρικής αυθεντικοποίηση, το Broken Fingers, δε μπορεί να χρησιμοποιηθεί από το κοινό και επιπλέον εκτελεί μόνο στατική ανάλυση του κώδικα της εφαρμογής.

Μελλοντικά, θα ήταν χρήσιμη η μελέτη αντίστοιχης μεθοδολογίας σε μεγαλύτερο δείγμα εφαρμογών της αγοράς και σε όλους τους τομείς ασφαλείας που μελετήσαμε. Η ανάπτυξη πειραματικών εφαρμογών για επιστήμονες που μαθαίνουν penetration testing και εκτίμηση ασφαλείας, «Insecure Bank», δεν παρέχει βιομετρική αυθεντικοποίηση ώστε να μελετηθεί η ασφάλεια της, ούτε και υπάρχει προς το παρόν κάποια σχετική εφαρμογή. Η μελέτη και αξιολόγηση της ασφάλειας και ορθής λειτουργίας σε πειραματικό και πραγματικό περιβάλλον είναι αναγκαία για τη διασφάλιση της βέλτιστης ποιότητας και αποδοτικότητας της τεχνολογίας αυτής. Τέλος, ακόμα και με τη βελτίωση και λειτουργία της υπάρχουσας εφαρμογής αποτίμησης ασφαλείας για την υλοποίηση του fingerprint API, υπενθυμίζεται πως μελετά μόνο τον κώδικα σε non-runtime περιβάλλον. Θα ήταν συνιστώμενη η ανάπτυξη σχετικού εργαλείου και για dynamic (runtime environment) ανάλυση της λειτουργίας του συγκεκριμένου χαρακτηριστικού. Εναλλακτικά, το υπάρχουν εργαλείο θα μπορούσε, μετά την επίλυση των τεχνικών προβλημάτων του, να περιλαμβάνει μελλοντικά και λειτουργίες dynamic analysis, ή πιθανώς υπάρχοντα εργαλεία static & dynamic analysis όπως αυτά που χρησιμοποιήσαμε να επεκταθούν και στη στοχευμένη μελέτη της απόδοσης χαρακτηριστικών ασφαλείας, ακόμα και πέρα από αυτό του δακτυλικού αποτυπώματος. Τέλος, ασφαλώς, ας μην ξεχνάμε ότι η παρούσα μελέτη πραγματοποιήθηκε αποκλειστικά σε περιβάλλον Android. Θα ήταν συνιστώμενη η συνέχεια της μελλοντικά και σε περιβάλλον iOS, Windows και άλλων λειτουργικών συστημάτων έξυπνων κινητών συσκευών.

Βιβλιογραφικές Πηγές

1. Nanavati, S. Thieme, M. & Nanavati, R. 2002. Biometrics: Identity Verification in a Networked World. Wiley Computer Publishing: New York. pp. 84-85
2. International Association for Identification, et al. 2015. The History of Fingerprints. http://www.onin.com/fp/fphistory.html. Pollack, A. 1981. Technology; Recognizing the real you.
3. Technology; Recognizing the Real You – Pollack A. - The New York Times. http://www.nytimes.com/1981/09/24/business/technology-recognizing-the-real-you.html
4. Κ. Λαμπρινουδάκης, 2017. Διαφάνειες μαθήματος «Ασφάλεια Πληροφοριακών Συστημάτων», Π.Μ.Σ. «Ασφάλεια Ψηφιακών Συστημάτων» Παν. Πειραιώς
5. Bianchi, A., Fratantonio, Y., Machiry, A., Kruegel, C., Vigna, G., Chung, S. P. H., & Lee, W. (2018, February). Broken Fingers: On the Usage of the Fingerprint API in Android. In NDSS. - https://reyammer.io/publications/2018_ndss_fingerprint.pdf
6. Wikipedia – Λήμμα «FIDO Alliance» - https://en.wikipedia.org/wiki/FIDO_Alliance
7. Fascinating Ancient History Of Fingerprints - AncientPages.com - March 4, 2016 - http://www.ancientpages.com/2016/03/04/fascinating-ancient-history-of-fingerprints/
8. https://epicecommercetools.com/2016/10/07/next-step-in-mobile-security-iris-recognition/
9. How Biometrics on Smartphones is Changing our Lives – M2SYS - http://www.m2sys.com/blog/biometric-resources/biometrics-on-smartphones/
10. Biometric Security for Cell Phones - Adrian Pocovnicu - https://core.ac.uk/download/pdf/6612561.pdf
11. 5 Ways Biometric Security Will Redefine Mobile Phone Authentication - http://www.m2sys.com/blog/guest-blog-posts/5-ways-biometric-security-will-redefine-mobile-phone-authentication/
12. Android Developers Blog – «Better Biometrics in Android P» - https://android-developers.googleblog.com/2018/06/better-biometrics-in-android-p.html
13. Βιολογία Γ' Γενικού Λυκείου – Θετική κατεύθυνση – ΟΕΔΒ
14. Disaster Victim Identification (DVI) - Interpol - https://www.interpol.int/INTERPOL-expertise/Forensics/DVI
15. On Forensic Use of Biometrics, with Face and Ear Recognition - Arbab-Zavar et al. - University of Southampton, University of Warwick - https://pdfs.semanticscholar.org/9fbe/6b06b1006fbb588aca06e9f6cd2a7438e624.pdf
16. Cell phone-based biometric identification - Jennifer R. Kwapisz, Gary M. Weiss, Samuel A. Moore - https://ieeexplore.ieee.org/abstract/document/5634532
17. Mobile Biometric Applications - Mar 6, 2017 - Chris Burt – Biometric Update - https://www.biometricupdate.com/wp-content/uploads/2017/03/special-report-mobile-biometric-applications.pdf
18. Smartphone sensor data as digital evidence – A. Mylonas, V. Meletiadis, L. Mitrou, D. Gritzalis
19. Biometrics and Forensics in Law Enforcement – Current and Evolving Technologies – Lockheed Martin © - John Mears, 2015
20. Genetically identical irises have texture similarity that is not detected by iris biometrics - Hollingworth, K., Bowyer, K. & Lagree, S. et. al., 2014.
21. Adaptive Algorithms in Accelerometer Biometrics – Pisani et. al. – 2014 – IEEE Xplore Digital Library - https://ieeexplore.ieee.org/document/6984853
22. «Password-free smartphones are no longer the stuff of science fiction — they're everywhere» - Agomuoh F. – 2017 – Business Insider - https://www.businessinsider.com/smartphone-biometrics-are-no-longer-the-stuff-of-science-fiction-2017-12

23. Λήμμα EFTPOS – Wikipedia - https://en.wikipedia.org/wiki/EFTPOS
24. Λήμμα Biometrics – Wikipedia - https://en.wikipedia.org/wiki/Biometrics
25. Biometrics - Overview - https://www.tutorialspoint.com/biometrics/biometrics_overview.htm
26. FIDO Alliance - "How FIDO works" - https://fidoalliance.org/how-fido-works/
27. FIDO Alliance – Biometric Component Certification - https://fidoalliance.org/certification/biometric-component-certification/
28. Cyber Vulnerabilities of Biometrics - OWASP 2015 -https://www.slideshare.net/BojanSimic/cyber-vulnerabilities-of-biometrics-owasp-2015
29. HSBC Experiences Voice Biometrics Telephone Banking Fail - https://www.bankinfosecurity.com/interviews/hsbc-experiences-voice-biometrics-telephone-banking-fail-i-3592
30. OWASP Mobile App Authentication Architectures - https://github.com/OWASP/owasp-mstg/blob/master/Document/0x04e-Testing-Authentication-and-Session-Management.md
31. Kali Tools - dex2jar Package Description - https://tools.kali.org/reverse-engineering/dex2jar
32. OWASP - Mobile Top 10 2016-M4-Insecure Authentication - https://www.owasp.org/index.php/Mobile_Top_10_2016-M4-Insecure_Authentication
33. https://github.com/linkedin/qark
34. Write and View Logs with Logcat – Android Developers - https://developer.android.com/studio/debug/am-logcat
35. Configuring an Android Device to Work With Burp - https://support.portswigger.net/customer/portal/articles/1841101-configuring-an-android-device-to-work-with-burp

ΠΑΡΑΡΤΗΜΑ Α – Οδηγός Εγκατάστασης

QARK

```
~ git clone https://github.com/linkedin/qark
~ cd qark
~ pip install -r requirements.txt
~ pip install .
//APK analysis
~ qark --apk path/to/my.apk
//Source Code Analysis
~ qark --java path/to/parent/java/folder
~ qark --java path/to/specific/java/file.java
```

Mob SF

Υπάρχουν δύο εκδοχές. Για τη μία απαιτείται εγκατάσταση **Docker**.
Στο **terminal** κατεβάζουμε το Mob SF και δημιουργούμε το interface για να το τρέξουμε:

```
>docker pull opensecurity/mobile-security-frameworkmobsf
>docker run -it -p 8000:8000 opensecurity/mobilesecurity-
framework-mobsf:latest
```

Η εγκατάσταση σε Docker container **δεν** υποστηρίζει dynamic analysis. Εναλλακτικά:

```
>git clone https://github.com/MobSF/Mobile-Security-Framework-
MobSF.git
>cd Mobile-Security-Framework-MobSF
>./setup.sh # For Linux and Mac, setup.bat For Windows
```

Στο **browser** ανοίγουμε το GUI: https://localhost:8000
Από εκεί, ανεβάζουμε το APK και τρέχουμε τις αναλύσεις.

ΠΑΡΑΡΤΗΜΑ Β – Εντολές

Drozer

```
//Start a session
> adb forward tcp:31415 tcp:31415
> drozer console connect
//Package info
> run app.package.list -f "app name"
> run app.package.info -a "package name"
//Attack Surface
> run app.package.attacksurface "package name"
//Exploiting Activities (AUTHENTICATION BYPASS)
> run app.activity.info -a "package name" -u
> run app.activity.start --component "package name""component name"
//Exploiting Content Provider
> run app.provider.info -a "package name"
> run scanner.provider.finduris -a "package name"
> run app.provider.query "uri"
> run app.provider.update "uri" –selection "conditions" "selection
arg" "column" "data"
> run scanner.provider.sqltables -a "package name"
> run scanner.provider.injection -a "package name"
> run scanner.provider.traversal -a "package name"
//Exploiting Broadcast Receivers
> run app.broadcast.info -a "package name"
> run app.broadcast.send --component "package name" "component
name" --extra "type" "key" "value"
> run app.broadcast.sniff --action "action"
//Exploiting Service
> run app.service.info -a "package name"
> run app.service.start --action "action" --component "package
name" "component name"
> run app.service.send "package name" "component name" --msg "what"
"arg1" "arg2" --extra "type" "key" "value" --bundle-as-obj
//To attack using SQL injection:
> run app.provider.query
content://com.mwr.example.sieve.DBContentProvider/Passwords/ --
projection "* FROM SQLITE_MASTER WHERE type='table';--"
//To read the files in the file system
> run app.provider.read <URI>To download content from the file
> run app.provider.download <URI>To check for injection
vulnerabilities
> run scanner.provider.injection -a <package_name>To check for
directory traversal vulnerabilities
> run scanner.provider.traversal -a <package_name>
```