

ΙΟΝΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



- Πτυχιακή Εργασία -

ΘΕΜΑ: Υλοποίηση και σχεδιασμός Software στην
κυβερνοεπίθεση Sim Swap

ΣΠΟΥΔΑΣΤΗΣ: Λάζαρος Παπανικολάου (Π2017085)

ΕΠΙΒΛΕΠΩΝ: Χριστόφορος Νταντογιάν

Επιβλέπων

ΔΡ. Χριστόφορος Νταντογιάν, Επίκουρος Καθηγητής,

Ιόνιο Πανεπιστήμιο

Τριμελής Επιτροπή

ΔΡ. Χριστόφορος Νταντογιάν, Επίκουρος Καθηγητής,

Ιόνιο Πανεπιστήμιο

ΔΡ. Εμμανουήλ Μαγκος, Αναπληρωτής Καθηγητής,

Ιόνιο Πανεπιστήμιο

ΔΡ. Αγγελική Τσώχου, Επίκουρος Καθηγήτρια,

Ιόνιο Πανεπιστήμιο

Υπεύθυνη δήλωση

Βεβαιώνω ότι είμαι συγγραφέας αυτής της πτυχιακής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της αναφέρεται στην πτυχιακή εργασία. Επίσης έχω αναφέρει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης βεβαιώνω ότι αυτή η πτυχιακή εργασία προετοιμάστηκε από εμένα προσωπικά ειδικά για τις απαιτήσεις του προγράμματος σπουδών του Τμήματος Πληροφορικής του Ιονίου Πανεπιστημίου.

Περίληψη

Η παρούσα εργασία πραγματεύεται τη σημασία της κοινωνικής μηχανικής που πρυτανεύει στην προσωπική ζωή των χρηστών. Έχει ως στόχο να αναδείξει τις επιπτώσεις παραβίασης προσωπικών δεδομένων εξαιτίας της ελλιπής πληροφόρησης σχετικά με τη ασφάλεια που χρειάζεται να υφίσταται. Αναλύει μορφές επιθέσεων οι οποίες υποβαθμίζουν την ασφάλεια των υπολογιστικών συστημάτων που παρέχονται για την προστασία των ευαίσθητων προσωπικών δεδομένων των χρηστών ενώ παράλληλα παρουσιάζει την υλοποίηση λογισμικού που χρησιμοποιείται ενάντια στην υποβάθμιση αυτήν.

Αρχικά το πρώτο κεφάλαιο αναλύει τον ρόλο της κοινωνικής μηχανικής στον σύγχρονο κόσμο του διαδικτύου καθώς και τον διαφορετικό τρόπο αξιοποίησης της βάσης προτιμήσεων. Αναφέρει τις επιθέσεις που συμπεριλαμβάνονται σε αυτήν κατηγοριοποιώντας τις και παρουσιάζοντας τις μεθόδους εξαπόλυσης και απόκτησης πληροφορίας από τους χρήστες. Τέλος στηριζόμενο σε πηγές παρουσιάζει στατιστικά συνηθέστερων επιθέσεων όπως sim swap και cyber crime και αριθμητικά μεγέθη προσωπικών δεδομένων που έχουν διαρρεύσει στο διαδίκτυο.

Στο δεύτερο κεφάλαιο, πραγματοποιείται μια ομαλή εισαγωγή στο εννοιολογικό κομμάτι της επίθεσης sim swap και στον τρόπο που διαλέγεται για να εκμεταλλευτεί ευπάθειες δικλείδων ασφάλειας. Εστιάζει στην ασφάλεια των πολλών παραγόντων που παρέχει ένα αρκετά ισχυρό σύστημα προστασίας δεδομένων, καθώς επίσης παρουσιάζει την θεωρητική λειτουργία του λογισμικού που μπορεί να αντιμετωπίσει επιθέσεις sim swap.

Έπειτα το τρίτο κεφάλαιο περιγράφει αναλυτικά την λειτουργία του λογισμικού, παρουσιάζοντας την λειτουργικότητα της και τονίζοντας την ιδιαιτερότητα που το ξεχωρίζει από παρομοίου είδους λογισμικά καθώς βασίζεται στην ανάγκη απόκτησης μοναδικών κεφάλαιο αριθμών της φυσικής συσκευής.

Τέλος το τέταρτο κεφάλαιο αξιολογεί το λογισμικό με άξονες την αποδοτικότητα, την αποτελεσματικότητα και την επίπτωση που έχει στο περιβάλλον. Παρουσιάζει μελλοντικές προεκτάσεις που χρειάζεται να ληφθούν υπόψιν και να υλοποιηθούν προς όφελος του λογισμικού καθώς επίσης, συνοψίζει τα συμπεράσματα που προέρχονται μέσω αυτής της εργασίας.

Πρόλογος

Η σύγχρονη εποχή που διανύουμε έχει λάβει τον χαρακτηρισμό της εποχής της τεχνολογίας εξαιτίας της χρήσης των υπολογιστικών συστημάτων που απαιτεί για την κάλυψη των αναγκών της. Οι κίνδυνοι που ελλοχεύουν στο διαδίκτυο είναι αδιόρατοι και στηρίζονται στην ελλιπή πληροφόρηση των χρηστών αναφορικά με τα μέτρα πρόληψης που κρίνεται αναγκαίο να ληφθούν. Η εμπιστευτικότητα, η ακεραιότητα και η διαθεσιμότητα αποτελούν το τρίπτυχο της ιδιωτικότητας των προσωπικών δεδομένων κρίνοντας αναγκαία την διαφύλαξη τους και όχι την υπομόνευση τους. Ωστόσο πληροφορίες και απόρρητα στοιχεία κάθε μέρα, διαρρέονται στο διαδίκτυο παράνομα, απειλώντας τόσο την προσωπική ζωή των χρηστών όσο και την οικονομική τους ευημερία. Βάσιμη είναι λοιπόν η λήψη μέτρων και συγκεκριμένα η ανάπτυξη λογισμικών που συμβάλλουν στην μείωση των πιθανοτήτων εμφάνισης των επιθέσεων που έχουν ως υπόβαθρο να καταρρίψουν την ασφάλεια της οικονομικής δύναμης του χρήστη. Λόγω του παραπάνω, το περιεχόμενο της παρούσας πτυχιακής παρουσιάζει την επιτυχή υλοποίηση λογισμικού ασφάλειας που έχει ως στόχο να αντιμετωπίσει απειλές ασφάλειας σε ηλεκτρονικές συναλλαγές.

Κέρκυρα, 21 Σεπτεμβρίου 2021

Λάζαρος Παπανικολάου

Ευχαριστίες

Η παρούσα πτυχιακή εργασία με θέμα «Υλοποίηση και σχεδιασμός Software στην κυβερνοεπίθεση Sim Swap» πραγματοποιήθηκε στο πλαίσιο της πτυχιακής εργασίας του τμήματος Πληροφορικής του Ιονίου Πανεπιστημίου το έτος 2021. Αξιοσημείωτες ήταν οι στιγμές που διαδραμάτισαν καθαριστικό ρολό στην πορεία της συγγραφή της, από άτομα που με τον τρόπο τους βοήθησαν είτε ψυχικά είτε λειτουργικά στην εξέλιξή της. Στο σημείο αυτό αισθάνομαι την ανάγκη να εκφράσω τις ειλικρινείς και θερμές ευχαριστίες μου σε όσους συνέβαλλαν στην ολοκλήρωση αυτής της προσπάθειας. Αρχικά θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή της πτυχιακής μου εργασίας, κύριο Χριστόφορο Νταντογιαν για τη συνεχή καθοδήγηση και την αμέριστη υποστήριξη που μου παρείχε σε όλο αυτό το χρονικό διάστημα. Τέλος θα ήθελα να εκφράσω τις ευχαριστίες μου στην μητέρα μου και στην αδελφή μου για την συνεχή ενθάρρυνση και υποστήριξη στον αγώνα αυτόν αλλά και στον πατέρα μου ο οποίος έφυγε νωρίς από την ζωή.

Περιεχόμενα

Υπεύθυνη δήλωση	3
Περίληψη.....	4
Πρόλογος.....	5
Ευχαριστίες.....	6
ΚΕΦΑΛΑΙΟ 1	13
ΕΙΣΑΓΩΓΗ.....	13
SOCIAL ENGINEERING.....	13
1.1 Έννοια της κοινωνικής μηχανικής.....	14
1.2 Τύποι κοινωνικής μηχανικής.....	14
1.3 Αρχιτεκτονική κοινωνικής μηχανικής	16
1.4 Απόκτηση πληροφορίας.....	17
1.5 Κατηγοριοποίηση επιθέσεων	19
1.6 Στατιστικά διαδικτυακών επιθέσεων	21
ΚΕΦΑΛΑΙΟ 2	29
SIM SWAP	29
2.1 Χρηματικές συναλλαγές στον χώρο του διαδικτύου	29
2.2 Έννοια της Sim Swap	30
2.3 Εκμετάλλευση ευπαθειών	31
2.4 Μεθοδολογία επίθεσης sim swap	32
2.5 Ανάλυση των μεθόδων ταυτοποίησης πολλαπλών παραγόντων.....	33
2.5.1 Εκμετάλλευση μοναδικού αριθμού με την επίθεση sim swap.....	34
2.5.2 Εκμετάλλευση ευπαθειών των SMS μηνυμάτων	35
2.6 Μοναδικοί αναγνωριστικοί αριθμοί κινητών συσκευών	36
2.7 Μεθοδολογία αποστολής SMS και αριθμού OTP μέσω ταυτοποίησης πολλών παραγόντων.....	37
2.8 Αρχιτεκτονική συγχρόνων εφαρμογών.....	38
2.8.1 Ορισμός και χρήση των API	39
2.8.2 Διακριτοποίηση των API	40
2.8.3 Βασικές διαφορές μεταξύ των APIs	41
2.9 Η προτίμηση των εταιριών.....	42
2.10 Θεωρητική ανάπτυξη εφαρμογής e-banking με χρήση του DeviceID	42
ΚΕΦΑΛΑΙΟ 3	45
ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ.....	45
3.1 REST API – BACKEND	45
3.1.1 Διαδικασία εγγραφής του χρήστη.....	48

3.1.2 Διαδικασία εισόδου του χρήστη	49
3.1.3 Διαδικασία ελέγχου εγκυρότητας του OTP.....	51
3.1.4 Διαδικασία ασφαλούς αποσύνδεσης του χρήστη	53
3.1.5 Λόγοι χρήστης RESTful API και παραμετροποίηση του Laravel framework	54
3.2 USER INTERFACE– FRONTEND.....	58
3.2.1 Βασικές παραμετροποιήσεις και διαφορά προτίμησης μεταξύ των volley και retrofit βιβλιοθηκών	58
3.2.2 Σύνδεση της λειτουργίας εγγραφής στο frontend με το backend	58
3.2.3 Σύνδεση της λειτουργίας εισόδου του frontend με το backend.....	61
3.2.4 Διαδικασία αποστολής του OTP αριθμού	63
3.2.5 Σύνδεση της λειτουργίας ελέγχου εγκυρότητας του OTP στο frontend με το backend.....	65
3.2.6 Εμπλουτισμός της λειτουργίας της διεπαφής του χρήστη.....	68
ΚΕΦΑΛΑΙΟ 4	73
4.1 Αξιολόγηση εφαρμογής	73
4.2 Συζήτηση – μελλοντικές προεκτάσεις	76
4.3 Συμπεράσματα	77
Βιβλιογραφία	79

Κατάλογος εικόνων

Εικόνα 1. Αρχιτεκτονική κοινωνικής μηχανικής, Journal of information security and applications 22 (2015) 113 e122.

Εικόνα 2. Κατηγοριοποίηση τεχνικών επίθεσης κοινωνικής μηχανικής, School of Electrical Engineering and Computer Science, University of North Dakota.

Εικόνα 3. Στατιστικά συνολικών χρηστών της τεχνολογίας παγκοσμίως, THE U.N LOCAL GOVERNMENT BODIES, GSMA INTELLIGENCE, ITU, GWI, EUROSTAF, CNNNIC, APJII, SOCIAL MEDIA PLATFORMS.

Εικόνα 4. Στατιστικά συνολικών breaches παγκοσμίως, Risk Based.

Εικόνα 5. Στατιστικά διαρροής δεδομένων παγκοσμίως, Risk Based.

Εικόνα 6. Στατιστικά συχνότερων κατηγοριών επιθέσεων παγκοσμίως, Risk Based.

Εικόνα 7. Στατιστικά διαρροής δεδομένων παγκοσμίως, Risk Based.

Εικόνα 8. Στατιστικά πολυπληθέστερων κατηγοριών επιθέσεων παγκοσμίως, North American Industry Classification System (NAICS), Vocabulary for Event Recording and Incident Sharing (VERIS).

Εικόνα 9. Στατιστικά συνηθέστερων στοιχείων που διέρρευσαν παγκοσμίως, Risk Based.

Εικόνα 10. Στατιστικά αριθμητικής υπεροχής σε τύπους ηλεκτρονικού εγκλήματος παγκοσμίως, Risk Based.

Εικόνα 11. Στατιστικά χρηστών ηλικιακά που χρησιμοποιούν ηλεκτρονικές συναλλαγές, THE U.N LOCAL GOVERNMENT BODIES, GSMA INTELLIGENCE, ITU, GWI, EUROSTAF, CNNNIC, APJII, SOCIAL MEDIA PLATFORMS.

Εικόνα 12. Σχεδιάγραμμα μεθοδολογίας sim swap βάση διαγράμματος ακολουθίας.

Εικόνα 13. Διάγραμμα τεχνικής διαδικασίας εξέλιξης της ασφάλειας εφαρμογής που βασίζεται στην εγκυρότητα της κάρτας sim.

Εικόνα 14. Διάγραμμα διαδικασίας αποστολής http request και http response διάμεσο μηχανισμού API.

Εικόνα 15. Διάγραμμα τεχνικής διαδικασίας εξέλιξης της ασφάλειας εφαρμογής που βασίζεται στην εγκυρότητα του DeviceID.

Εικόνα 16. MariaDB βάση δεδομένων.

Εικόνα 17. Εικόνα δημιουργίας table μεταβλητών και αρχικοποιήσεις χαρακτήρων.

Εικόνα 18. Εικόνα προσθήκης μεταβλητών σε function προστασίας από επιθέσεις τύπου sql injection.

Εικόνα 19. Εικόνα διαδικασίας εγγραφής του χρήστη (1).

- Εικόνα 20.** Εικόνα διαδικασίας εγγραφής του χρήστη (2).
- Εικόνα 21.** Εικόνα διαδικασίας εισόδου του χρήστη με τα στοιχεία πρόσβασης.
- Εικόνα 22.** Εικόνα διαδικασίας εισόδου του χρήστη με τα στοιχεία πρόσβασης και δημιουργία OTP (1).
- Εικόνα 23.** Εικόνα διαδικασίας εισόδου του χρήστη με τα στοιχεία πρόσβασης και δημιουργία OTP (2).
- Εικόνα 24.** Εικόνα διαδικασίας ελέγχου εγκυρότητα OTP αριθμού (1).
- Εικόνα 25.** Εικόνα διαδικασίας ελέγχου εγκυρότητα OTP αριθμού (2).
- Εικόνα 26.** Εικόνα διαδικασίας ασφαλούς απενεργοποίησης χρήστη.
- Εικόνα 27.** Εικόνα διαδικασίας σύνδεσης backend-frontend.
- Εικόνα 28.** Διαδικασία οριοθέτησης headers.
- Εικόνα 29.** Όλα τα request περνάνε από header που επιτρέπει μεταφορά json δεδομένων.
- Εικόνα 30.** Διαδικασία εγκαθίδρυσης URL του API.
- Εικόνα 31.** Διαδικασία αρχικοποίησης των Middleware's που χρησιμοποιούνται.
- Εικόνα 32.** Διαδικασία πραγματοποίησης register σύνδεσης στο frontend (1).
- Εικόνα 33.** Διαδικασία πραγματοποίησης register σύνδεσης στο frontend (2).
- Εικόνα 34.** Διαδικασία σύνδεσης volley library με backend.
- Εικόνα 35.** Διαδικασία λήψης μεταβλητών από το backend με volley library.
- Εικόνα 36.** Διαδικασία αρχικοποίησης πεδίων που λαμβάνουν εντολές για το backend.
- Εικόνα 37.** Διαδικασία πραγματοποίησης login σύνδεσης στο frontend.
- Εικόνα 38.** Διαδικασία δημιουργίας notifications.
- Εικόνα 39.** Διαδικασία απόκτησης DeviceID.
- Εικόνα 40.** Διαδικασία σύνδεσης login με volley library και backend (1).
- Εικόνα 41.** Διαδικασία σύνδεσης login με volley library και backend (2).
- Εικόνα 42.** Διαδικασία αρχικοποίησης πεδίων που λαμβάνουν εντολές για το backend.
- Εικόνα 43.** Διαδικασία δημιουργίας σύνδεσης για εγκυρότητα otp.
- Εικόνα 44.** Διαδικασία δημιουργίας πεδίων ψηφίων.
- Εικόνα 45.** Διαδικασία δημιουργίας αλλαγή πεδίου ψηφίου.
- Εικόνα 46.** Διαδικασία σύνδεσης εγκυρότητας otp με volley library στο backend.

Εικόνα 47. Διαδικασία αρχικοποίησης πεδίων που λαμβάνουν εντολές για το backend.

Εικόνα 48. Διαδικασία δημιουργίας UI για Main Activity.

Εικόνα 49. Διαδικασία δημιουργίας UI για Register screen.

Εικόνα 50. Διαδικασία δημιουργίας UI για Otp screen.

Εικόνα 51. Διαδικασία δημιουργίας UI για Login screen.

Εικόνα 52. Εξαφάνιση μπάρας ειδοποιήσεων.

Εικόνα 53. Αρχική οθόνη και ζήτηση permissions.

Εικόνα 54. Αποστολή Otp και εισαγωγή του.

Εικόνα 55. Αποτελέσματα POSTMAN για Login.

Εικόνα 56. Αποτελέσματα POSTMAN για ConfirmOtp.

Εικόνα 57. Αποτελέσματα POSTMAN για Register.

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

Ανέκαθεν η τεχνολογία και ο τομέας της πληροφορικής διαδραμάτιζε σημαντικό ρολό στην επιρροή της οικονομίας και της κοινωνίας εξαιρουμένης. Με την πάροδο του χρόνου, ωστόσο η ανάγκη της για την κάλυψη βασικών αναγκών όπως τραπεζικών συναλλαγών προέτρεψε αρκετούς χρήστες να στραφούν υπέρ της και να την θεωρήσουν ως πρωτεύον εργαλείο ικανοποίησης αυτών των αναγκών. Ωστόσο η ευρεία χρήση της τεχνολογίας προϋποθέτει και την ύπαρξη πολλών δικλίδων ασφάλειας καθώς τα δεδομένα που εκτίθενται στο διαδίκτυο είναι πολυάριθμα. Την ανάγκη αυτήν έχουν συνενώσει αρκετοί οργανισμοί και εταιρίες οι οποίες κάνουν ότι είναι δυνατόν για την ανάπτυξη τεχνολογιών που θα βοηθήσουν τόσο τον χρήστη που την χειρίζεται όσο και την ίδια την εταιρία. Η προσπάθεια ανάπτυξης λογισμικών και τεχνολογιών που βοηθάνε στην ασφάλεια των χρηστών είναι αξιόλογη, όμως η κατάχρηση της κοινωνικής μηχανικής συμβάλλει στην υπομόνευση της ιδιωτικότητας των χρηστών και της ακεραιότητας των προσωπικών τους δεδομένων. Εξαιτίας της μη έγκυρης ευαισθητοποίησης για την κατάσταση που επικρατεί, κύριος στόχος είναι άτομα μέσης ηλικίας που αποτελούν χρήστες των εργαλείων τεχνολογίας για την κάλυψη των αναγκών τους. Επιθέσεις τύπου sim swapping αποτελούν μεγάλη απειλή για αυτήν την κοινωνική ομάδα καθώς βάση ερευνών έχει αποδειχθεί ότι αποτελούν τους κύριους χρήστες που χρησιμοποιούν την τεχνολογία για την πραγματοποίηση τέτοιων ηλεκτρονικών συναλλαγών. Φυσικά πολλές είναι οι εταιρίες που προσπαθούν με τον οποιοδήποτε τρόπο να βοηθήσουν στην ασφάλεια των προσωπικών δεδομένων παρέχοντας τεχνικές ασφάλειας διπλής ταυτοπροσωπίας, καθώς αποτελεί μια από τις ισχυρότερες μεθόδους ταυτοποίησης. Βάση ερευνών όμως υπάρχουν τρόποι και κενά ασφάλειας που εκμεταλλεύονται για να χρησιμοποιηθούν ενάντια στην ασφάλεια των μεθόδων αυτών. Μια πρακτικά καλή τεχνική καταπολέμησης των κενών ασφάλειας είναι η δημιουργία λογισμικών που κάνουν χρήση της διπλής ταυτοπροσωπίας αλλά βασίζονται σε μοναδικούς αριθμούς της φυσικής συσκευής του χρήστη. Η χρήση αυτών των μεθόδων είναι αρκετά πιθανό να καταπολεμήσει όσο το δυνατόν περισσότερο την πιθανότητα εμφάνισης τέτοιου είδους επιθέσεων.

SOCIAL ENGINEERING

Η έννοια της κοινωνικής μηχανικής είναι αρκετά γενικευμένη και ορίζεται διαφορετικά από τον κάθε άνθρωπο. Στην σύγχρονη εποχή, λόγω της ραγδαίας αύξησης εμφάνισης επιθέσεων στον κυβερνοχώρο, η κοινωνική μηχανική συνδέεται με μια απρόσωπη κακόβουλη πράξη που ως στόχο έχει να υποβαθμίσει, μέσω της πειθούς, τη νοημοσύνη της οντότητας του

θύματος έτσι ώστε να αποσπάσει ευαίσθητες προσωπικές πληροφορίες. Ωστόσο, ο προσδιορισμός αυτός δεν ανταποκρίνεται απολυτά στο εννοιολογικό περιεχόμενο της ορολογίας, καθώς είναι άρρηκτα συνυφασμένος και με την ικανότητα του χρήστη να πείσει κάποιον για καλό σκοπό, απορρίπτοντας έτσι κάθε δόλιο μέσο. Βέβαια όπως έχει επισημανθεί και παραπάνω, το αρνητικό περιεχόμενο του ορού είναι αυτό που πρυτανεύει. Προς επίτευξη αυτού, χρησιμοποιούνται διάφορων ειδών επιθέσεων που λαμβάνουν χώρα στο διαδίκτυο, εκμεταλλευόμενες από ανθρώπους με ιδιαίτερη γνώση στο αντικείμενο.

1.1 Έννοια της κοινωνικής μηχανικής

Η κοινωνική μηχανική έχει υιοθετήσει τον ρολό της επιβολής κυριαρχίας και της χειραγώγησης του χρήστη βασιζόμενη στην ψυχολογία του. Ωστόσο, ο καθένας έχει σχηματίσει την προσωπική του άποψη, ακολουθώντας μια ξεχωριστή κατεύθυνση για την οριοθέτηση της εν λόγω έννοιας. Η μία οπτική που βασίζεται στο λεξικό Cambridge (Cambridge, 2021) έχει ως υπόβαθρο την κακόβουλη πράξη και αναφέρεται σε αυτήν ως μια ενέργεια χειραγώγησης του χρήστη με στόχο την αποκάλυψη και υποκλοπή των προσωπικών του στοιχείων. Στην αντίθετη περίπτωση η άλλη οπτική πλευρά που βασίζεται στο λεξικό του Webster's (Merriam-Webster, 2021) αναφέρεται στην θέση και στον ρολό που πρέπει να έχει ο άνθρωπος στην καθημερινή κοινωνία προσφέροντας τις υπηρεσίες του για ευεργετικούς σκοπούς. Πλησιέστερο εννοιολογικά στην κοινωνική μηχανική βάση του (Hadnagy, 2010) είναι η σημασία της πειθούς. Σε καθημερινή βάση, η συνήθεια ενός ατόμου να λάβει την βοήθεια του συνάνθρωπου αποκαλύπτοντας του προσωπικά του στοιχεία, αποτελεί μια πρόκληση για τον ίδιο να επιτύχει τον σκοπό του πείθοντας τον. Η συγκεκριμένη μεθοδολογία, δηλαδή η απόκτηση της εμπιστοσύνης από τον συνάνθρωπο ορίζεται ως κοινωνική μηχανική. Τέτοιες περιπτώσεις υφίστανται σε όλα τα κοινωνικά στρώματα. Λαμβάνοντας ωστόσο υπόψη την καθημερινότητα και τις απαιτήσεις της, η ορολογία της κοινωνικής μηχανικής αναφέρεται κυρίως στην κοινωνία της τεχνολογίας. Για αυτόν τον λόγο αποκτά αυτόματα αρνητικές διαστάσεις, εξαιτίας των επιθέσεων που χρησιμοποιούνται για να υποκλέψουν προσωπικά στοιχεία των χρηστών του διαδικτύου χρησιμοποιώντας διάφορες μεθόδους.

1.2 Τύποι κοινωνικής μηχανικής

Όπως το διαδίκτυο έτσι και η κοινωνική μηχανική χωρίζεται σε κατηγορίες. Κατά βάση αναφέρεται στο γεγονός ότι χρησιμοποιείται για την εξασφάλιση της εμπιστοσύνης του χρήστη έχοντας ως τελικό στόχο να αποσπάσει προσωπικές του πληροφορίες. Η ενέργεια αυτήν μπορεί να λάβει είτε αρνητικές είτε θετικές διαστάσεις αναλόγως την οντότητα που θα τις εκμεταλλευτεί. Σύμφωνα με (Hadnagy, 2010), οι δρώντες μπορεί να είναι άτομα εξιδεικευμένα στην χρήση του υπολογιστή, έχοντας όμως διαφορετικούς στόχους όπως είναι οι black hat

hackers, οι spies, οι Identity thieves, οι Disgruntled employees, οι Scamartist, οι Executive recruiters και οι Salespeople. Η κάθε μια από της παραπάνω κατηγορίες διαφοροποιείται από το μοτίβο που ακολουθεί και τις δεξιότητες που κατέχει. Ωστόσο έχουν ένα κοινό γνώρισμα, το γεγονός ότι ως απώτερο σκοπό, μέσω της εξασφάλισης της εμπιστοσύνης, επιθυμούν να λάβουν χρηματικά ποσά ή προσωπικά δεδομένα των χρηστών. Βασικό γνώρισμα των επιτιθέμενων είναι η τεχνική elicitation. Αφορά στην υιοθέτηση κανόνων, προσωπικοτήτων, συμπεριφορών και ιδεών που χρειάζεται ο επιτιθέμενος για να αποσπάσει πληροφορίες έμμεσα χωρίς να το ζητήσει. Αναλύοντάς τις κατηγορίες οι black hat hackers αποκτούν περισσότερη εμπειρία εισβάλλοντας στα υπολογιστικά συστήματα των χρηστών χωρίς εξουσιοδοτημένη πρόσβαση. Επιπρόσθετα οι spies και οι Identity thieves λειτουργούν παρόμοια παρακολουθώντας και ψυχολογώντας τον χρήστη για να αποσπάσουν πληροφορίες ιδιωτικού περιεχομένου όπως είναι το ονοματεπώνυμο, η διεύθυνση κατοικίας και ο αριθμός τηλεφώνου. Έπειτα έχουμε τους Disgruntled employees οι οποίοι με τον τρόπο τους χρησιμοποιούν μεθόδους επικαλούμενοι το συναίσθημα για να κερδίσουν την ευκαιρία να σχηματίσουν ένα πιο οικείο περιβάλλον με τον εργαζόμενο ή ακροατή. Προχωρώντας, Οι Scamartist είναι εκείνοι οι οποίοι προσπαθούν να δολέψουν τον χρήστη να συμμετάσχει σε δόλιες απάτες οι οποίες τις περισσότερες φορές προϋποθέτουν και καταθέσεις χρηματικών ποσών. Στο ίδιο πλαίσιο ερμηνείας εντάσσονται και οι Executive recruiters οι οποίοι χρησιμοποιώντας μεθόδους ανάκτησης πληροφορίας χωρίς να το ζητήσουν, είναι σε θέση να μάθουν τις προσωπικές δεξιότητες και ικανότητες του χρήστη εκμεταλλευόμενες φυσικά για δικό τους συμφέρον. Τελευταία κατηγορία είναι οι sales person οι οποίοι έχουν ως στόχο να πείσουν τον χρήστη να προβεί σε αγορά ενός προϊόντος αναδεικνύοντας έτσι τις προτιμήσεις του. Για την επίτευξη του παραπάνω σκοπού χρησιμοποιούν κυρίως δυο μεθόδους όπως είναι το Pretexting δηλαδή η διαδικασία να υποδυθεί κάποιον διαφορετικό χαρακτήρα από αυτόν που είναι για να λάβει πληροφορίες που θέλει και elicitation που είναι η διαδικασία για να συλλεγεί δεδομένα από τον χρήστη έμμεσα, χωρίς δηλαδή να το ζητήσει. Οι κατηγορίες που συζητήθηκαν εντάσσονται στην κοινωνική μηχανική με αρνητικές διαστάσεις καθώς τα προσωπικά δεδομένα των χρηστών χρησιμοποιούνται καθαρά για εκμετάλλευση και προσωπικό συμφέρον. Ωστόσο στην αντίθετη περίπτωση υπάρχει η θετική πλευρά της η οποία στοχεύει στην απόκτηση προσωπικών πληροφοριών του χρήστη προς όφελος του. Τέτοιου είδους κατηγορίες είναι white hat hackers, η Governments και οι Doctors, psychologists, and lawyers. Η κάθε μια από τις παραπάνω κατηγορίες εξασφαλίζει την εμπιστοσύνη του χρήστη και συλλεγεί τις πληροφορίες του για να συνδράμει στην ασφάλεια του και όχι στην εξαπάτηση του.

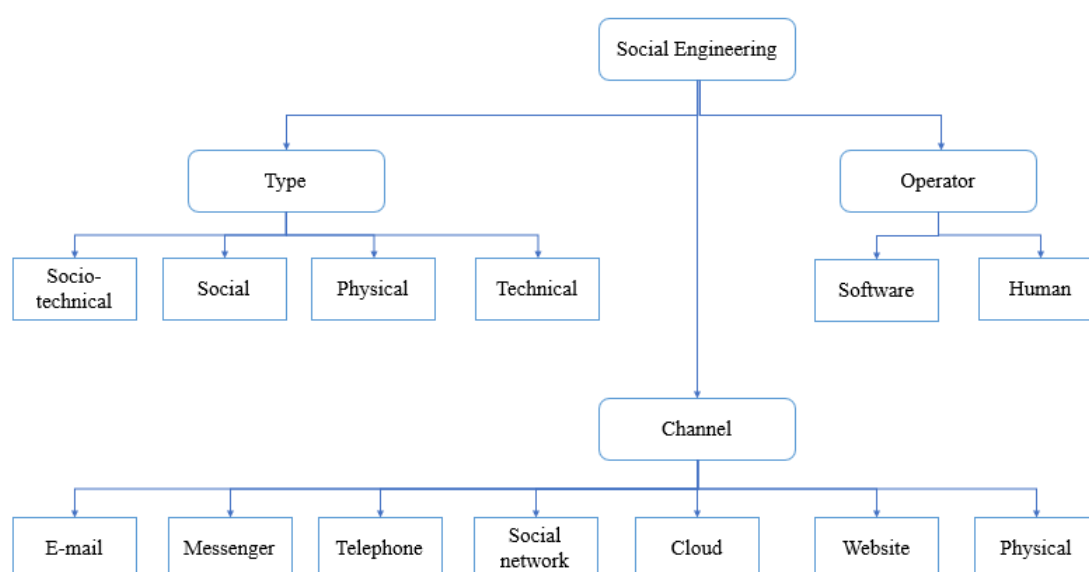
1.3 Αρχιτεκτονική κοινωνικής μηχανικής

Όπως προαναφέρθηκε η κοινωνική μηχανική έχει ένα προκαθορισμένο πλάνο και ακολουθεί ένα μοτίβο. Δίνοντας αναφορά στην κοινωνική μηχανική του κυβερνοχώρου, η σημασία της είναι άρρηκτα συνδεδεμένη με τις αρνητικές διαστάσεις που αναφέρθηκαν παραπάνω. Βάση αυτού το μοτίβο που ακολουθεί αποτελεί τον κύκλο ζωής της κοινωνικής μηχανικής και απαρτίζεται από τα στάδια investigation, hook, play και exit (Wang, Sun, and Zhu 2020). Οι οροί αυτοί δεν είναι προκαθορισμένοι καθώς εμφανίζονται και με διαφορετικές ονομασίες όπως information gathering, Establish Relationship and Rapport, Exploitation και execution (Salahdine and Kaabouch 2019). Ωστόσο το περιεχόμενο των εννοιών μεταξύ των διαφοροποιημένων ορολογιών που υπάρχει είναι παρόμοιο. Το καθένα από αυτά διαδραματίζει καθοριστικό ρόλο στην προσπάθεια απόκτησης της εμπιστοσύνης του χρήστη. Σύμφωνα με το πρώτο στάδιο, προτεραιότητα έχει η επιλογή του χρήστη, η συλλογή πληροφοριών σχετικά με τα ενδιαφέροντα του, τις προτιμήσεις του και γενικότερα πληροφορίες οι οποίες χαρακτηρίζουν την προσωπικότητα του. Βέβαια η κατάλληλη επιλογή του, γίνεται σε βάθος χρόνου καθώς το σημαντικότερο είναι να συλλεγούν όσο περισσότερα δεδομένα γίνεται. Μετά το πέρας του πρώτου στάδιο ακολουθεί το δεύτερο το οποίο στοχεύει στην απόκτηση της εμπιστοσύνης του χρήστη. Έχοντας λάβει υπόψιν τα δεδομένα που συλλεχθήκαν ακολουθεί η διαδικασία κατά την οποία ο επιτιθέμενος βασίζεται σε αυτά τα δεδομένα με σκοπό να χειραγωγήσει τον χρήστη. Έχοντας αναπτύξει μια πρώτη επαφή μαζί του, προσπαθεί να δημιουργήσει ένα ασφαλές περιβάλλον φανερώνοντας την αλήθεια του. Επιθυμητός στόχος είναι η ανάπτυξη μιας ιδανικής φιλικής σχέσης με τον χρήστη. Έπειτα σχηματίζεται θετική εντύπωση για τον επιτιθέμενο, αφού η εξασφάλιση της εμπιστοσύνης έχει ήδη επιτευχθεί. Έχοντας αποκτήσει ο επιτιθέμενος την κυρίαρχη θέση, ακολουθεί το τρίτο στάδιο όπου πραγματοποιείται η απόσπαση προσωπικών του πληροφοριών. Πλέον ο χρήστης, εμπιστευόμενος πλέον τον επιτιθέμενο αποκαλύπτει είτε προσωπικά είτε επαγγελματικά του στοιχεία, αγνοώντας οποιοδήποτε δόλιο σκοπό πίσω από αυτό. Έτσι κρίνεται ευάλωτος και είναι αρκετά επιρρεπής στο να πέσει στην παγίδα του επιτιθέμενου και να αποκτήσει πρόσβαση σε προσωπικούς του λογαριασμούς. Με το πέρας του τρίτου σταδίου ακολουθεί το τελευταίο στάδιο το οποίο έχει ως στόχο την επιτυχή ολοκλήρωση των διαδικασιών και την κάλυψή των ενεργειών. Θα μπορούσε να θεωρηθεί και ως το σημαντικότερο καθώς αποτελεί το σημείο όπου χαρακτηρίζεται από μυστικοπάθεια. Η οποιαδήποτε πράξη που λαμβάνει χώρα, χρειάζεται να μην είναι φανερή, και η ταυτότητα του επιτιθέμενου να μην εμφανίζεται πουθενά. Η κοινωνική μηχανική του κυβερνοχώρου έχει ως ζητούμενο την λανθασμένη κίνηση που θα κάνει ο χρήστης για να αποκαλύψει τα προσωπικά του στοιχεία. Βασίζεται πλήρως στην ψυχολογία του και στο κατά ποσό ανθεκτικός είναι σε τέτοιου είδους πράξεις.

1.4 Απόκτηση πληροφορίας

Στην σύγχρονη εποχή το κόστος της πληροφορίας κυμαίνεται σε υψηλά επίπεδα ενώ η προσβασιμότητά της, τις περισσότερες φορές είναι αρκετά ευάλωτη και διαμπερής. Ο καθένας ξεχωριστά διατηρεί προσωπικές και μοναδικές πληροφορίες οι οποίες ολοκληρώνουν την προσωπικότητα του. Ωστόσο η έλλειψη της εμπιστευτικότητάς, της ακεραιότητας και της διαθεσιμότητας αυτών των πληροφοριών οδηγεί στην αλλοίωση την ιδιωτικότητας της οντότητας. Διάφοροι χρήστες του διαδικτύου, μέσω της κοινωνικής μηχανικής, μάχονται για την απόκτηση πληροφοριών και προσωπικών δεδομένων τρίτων, κυρίως για την αξιοποίηση συμφερόντων που εποφθαλμιούν είτε τις χρηματικές απολαβές είτε την αναγνωσιμότητα τους. Πραγματοποιείται χρήση μεθόδων και εργαλείων που έχουν ως στόχο να υποβαθμίσουν την νοημοσύνη του χρήστη προτρέποντας τον να προβεί σε ενέργειες που θα αποκαλύψει προσωπικά του στοιχεία. Η κοινωνική μηχανική βασίζεται σε μεγάλο βαθμό στις αρχές τις επιρροής (Robert Cialdini "Influence: The Psychology of Persuasion", 2006) κατονομάζοντας τις ως αμοιβαιότητα, δέσμευση και συνέπεια, κοινωνική απόδειξη, εξουσία, αρέσκεια και έλλειψη. Συγκεκριμένα η αρχή της αμοιβαιότητας εστιάζει στην τάση του χρήστη να ανταποδίδει χάρες, η αρχή της δέσμευσης και συνέπειας στην υποχρέωση τήρησης αυτής της δέσμευσής και η κοινωνική απόδειξη στην μίμηση προτύπων και ιδεών. Παράλληλα, η αρχή της εξουσίας εστιάζει στην ανάγκη υπακοής σε ανώτερες οντότητες, η αρχή της αρέσκειας σε εξασφάλιση της εμπιστοσύνης σε άγνωστες οντότητες και τέλος η αρχή της έλλειψής οπου στοχεύει στην ανάγκη για γρήγορες αψυχολόγητες αποφάσεις. Αναμφισβήτητα η ψυχολογία του χρήστη επηρεάζεται από τις αρχές της επιρροής θέτοντας τον σε θέση να αξιολογηθεί η ανθεκτικότητα του στις ηλεκτρονικές απάτες που ευδοκιμούν. Όπως προαναφέρθηκε η κοινωνική μηχανική χωρίζεται σε κατηγορίες που περιλαμβάνουν διαφορετικές μεθόδους προσέγγισης του χρήστη, όλα όμως έχουν κοινό γνώρισμα το γεγονός ότι οδηγούν στην εξαπάτηση του. Υπάρχει η προσέγγιση Physical approaches, η Social approaches, η Reverse social engineering, η Technical approaches και η Socio-technical approaches. Το καθένα από τα παραπάνω εντάσσεται σε μια ειδική κατηγορία περιλαμβάνοντας ξεχωριστές μεθόδους απόκτησης της πληροφορίας. Συγκεκριμένα με την Physical approaches ο επιτιθέμενος αξιολογεί την ασφάλεια ενός συστήματος προσπαθώντας να αποκτήσει πρόσβαση σε αυτό κατακρατώντας πληροφορίες και δεδομένα που κρίνονται χρήσιμα. Με την Social approaches ο επιτιθέμενος προσπαθεί να εξασφαλίσει την εμπιστοσύνη του χρήστη δια μέσω τηλεφώνου ή με την βοήθεια μέσων κοινωνικής δικτύωσης. Έπειτα η Reverse social engineering οπου αποτελείται από τρία στάδια τα οποία κατονομάζονται ως σαμποτάζ, διαφήμιση και βοήθεια στοχεύει στο να συμβάλλει θετικά αναμένοντας τις περισσότερες φορές για αντάλλαγμα. Η αντιστροφή κοινωνική μηχανική καλείται όταν δημιουργείται μια κατάσταση κατά την οποία διαδραματίζει σημαντικό ρολό η αλληλεγγύη. Ουσιαστικά ο επιτιθέμενος προσπαθεί μέσω

ερωτήσεων που κάνει στον χρήστη για να τον βοηθήσει, να λάβει πληροφορίες που αφορούν τόσο εκείνον όσο και το περιβάλλον του ως αντάλλαγμα. Τέλος η Technical approaches και η Socio-technical approaches έχουν παρόμοιο υπόβαθρο αλλά λειτουργούν κάπως διαφορετικά καθώς η πρώτη χρησιμοποιείται για να λάβει πληροφορίες χρηστών έχοντας ως εργαλείο τον φυλλομετρητή ενώ η δεύτερη στοχεύει στην εγκατάσταση ενός κακόβουλου λογισμικού σε συσκευές των χρηστών με την χρήση εξωτερικών αποθηκευτικών μέσων. Κατανοώντας τις κατηγορίες που υπάρχουν για την απόκτηση της πληροφορίας, η εξαπάτηση εξίσου μπορεί να γίνει χρησιμοποιώντας την ανθρώπινη παρουσία ή την βοήθεια του υπολογιστή. Τέλος αναφέρονται τα κανάλια τα οποία χρησιμοποιούνται τα οποία έχουν άμεση επιρροή στην καθημερινότητα του χρήστη καθώς βρίσκονται σε συνεχή επαφή μαζί του. Τέτοιου είδους κανάλια είναι η τηλεφωνικές συσκευές, η ηλεκτρονική αλληλογραφία, τα μέσα κοινωνικής δικτύωσης, οι ιστοσελίδες, οι εξ αποστάσεων απρόσωπες συνομιλίες και οι διαδικτυακοί αποθηκευτικοί χώροι. Τα παραπάνω μπορούν να χρησιμοποιηθούν από κακόβουλους για να εξαπατήσουν και να υπονομεύσουν την ιδιωτικότητα του, χειραγωγώντας τον και επιβάλλοντας τον την κυριαρχία του. Στην παρακάτω φωτογραφία διακρίνεται η κατανομή σε ομάδες και κατηγορίες της αρχιτεκτονικής της κοινωνικής μηχανικής του κυβερνοχώρου.



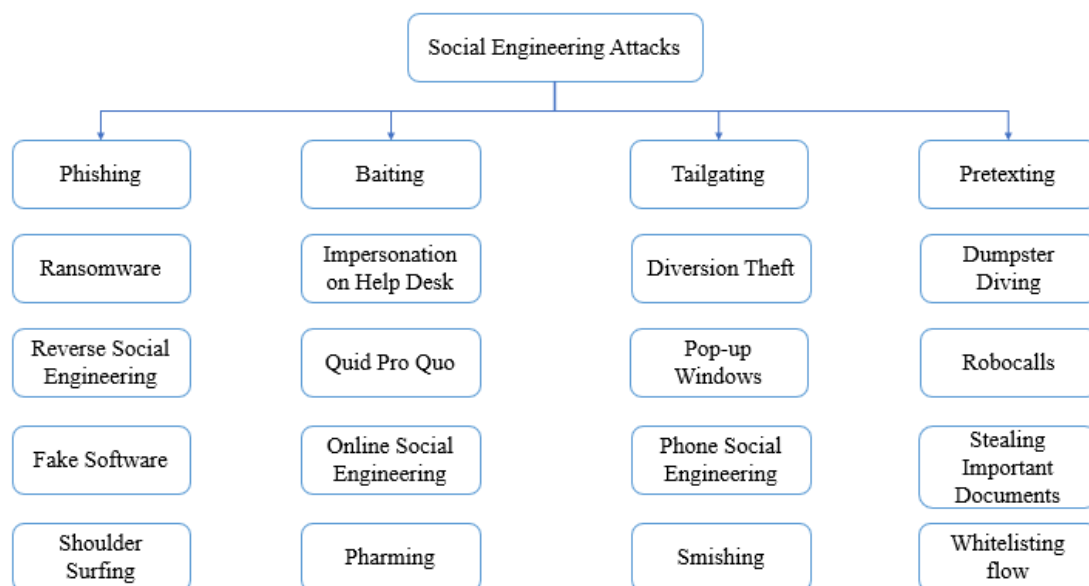
Εικόνα 1. Αρχιτεκτονική κοινωνικής μηχανικής, Journal of information security and applications 22 (2015) 113 e122.

1.5 Κατηγοριοποίηση επιθέσεων

Υπάρχουν πολλών ειδών επιθέσεων στο διαδίκτυο που εντάσσονται στο social engineering. Μερικές εξ αυτών εκτελούνται για να προκαλέσουν ένα χάος στα υπολογιστικά συστήματα των χρηστών ενώ άλλες χρησιμοποιούνται για εκμετάλλευση των προσωπικών δεδομένων. Για αυτό τον λόγο εντάσσονται σε κατηγορίες, οι οποίες ταξινομούνται βάση του τρόπου εξαπόλυσης τους και των επιπτώσεων που έχει στον χρήστη. Συγκεκριμένα υπάρχουν τεσσάρων ειδών τεχνικές επιθέσεων όπως το phishing, το baiting, το pretexting και το tailgating. Αναλυτικότερα, η επίθεση phishing είναι υπεύθυνη για τις περισσότερες περιπτώσεις εξαπάτησης χρηστών στην σύγχρονη εποχή καθώς είναι η πιο διαδεδομένη και στοχοποιεί κυρίως άτομα με ελλιπή γνώση στα υπολογιστικά συστήματα. Η μεθοδολογία που ακολουθεί είναι να εξαπατήσει τον αποδέκτη στέλνοντας του ενημερωτικά κακόβουλα μηνύματα, βασιζόμενα στις προτιμήσεις του, μέσω των καναλιών που αναφέρθηκαν προηγουμένων, προτρέποντας τον να τα ελέγξει. Ο πιο διαδιδόμενος τρόπος αποστολής τέτοιων μηνυμάτων είναι μέσω της ηλεκτρονικής αλληλογραφίας. Αυτό που ωθεί τους χρήστες να προβούν στον έλεγχο αυτού του κακόβουλου μηνύματος είναι το συναίσθημα της περιέργειας και του φόβου ότι πρόκειται για ένα σημαντικό και επείγον ηλεκτρονικό μήνυμα. Συνήθως τέτοιου είδους επιθέσεις δεν στοχοποιούν έναν μόνο χαρακτήρα αλλά μια μεγαλύτερη κοινωνική ομάδα, καθώς το όφελος των πληροφοριών που ενδεχομένως να λάβει ο επιτιθέμενος είναι απροσδιόριστο. Εμβαθύνοντας στην επίθεση, το phishing χωρίζεται σε πέντε επιπλέον υποκατηγορίες, τις spear phishing, whaling phishing, vishing phishing, Interactive voice phishing και business email compromise phishing. Η κάθε μια από τις παραπάνω υποκατηγορίες ακολουθεί διαφορετικό μοτίβο εξαπάτησης των χρηστών αλλά και οι πέντε χρησιμοποιούνται για να παραπλανήσουν τον χρήστη. Μετονομάζεται spear phishing όταν η επίθεση αφορά εξειδικευμένα και στοχευμένα ένα άτομο το οποίο εργάζεται σε μια εταιρία, με στόχο να αποσπάσει πληροφορίες που αφορούν τόσο εκείνον όσο και την εταιρία στην οποία εργάζεται. Σε άλλη περίπτωση whaling phishing αναφέρεται όταν ο στόχος είναι οι υψηλόβαθμοι εργαζόμενοι μιας εταιρίας οι οποίοι διατηρούν ιδιωτικά της δεδομένα. Η επίθεση vishing phishing αναφέρεται στην απόσπαση πληροφοριών που πραγματοποιείται μέσω τηλεφωνικής επικοινωνίας. Σε αυτήν την περίπτωση ο χρήστης υποδύεται έναν χαρακτήρα και χειραγωγώντας τον, εξασφαλίζει την εμπιστοσύνη του ακροατή και έχει την ευκαιρία να τον εξαπατήσει λαμβάνοντας ευαίσθητα προσωπικά δεδομένα. Η εναλλακτική επίθεσης Interactive voice phishing παρέχει ακριβώς τον ίδιο τρόπο εξαπάτησης του, ωστόσο η διαφορά είναι το είδος των πληροφοριών που συλλέγονται καθώς περιλαμβάνονται τραπεζικά ευαίσθητα δεδομένα. Τελευταία υποκατηγορία είναι η επίθεση business email compromise phishing η οποία λειτουργεί παρόμοια όπως η whaling phishing. Στόχος είναι και εδώ οι υψηλόβαθμοι εργαζόμενοι μιας εταιρίας οι οποίοι όμως προσεγγίζονται με διαφορετικό τρόπο. Σε αυτήν την

περίπτωση για να λάβει πρόσβαση ο επιτιθέμενος σε προσωπικούς λογαριασμούς αλληλογραφίας και σε ιδιωτικά στοιχεία της εταιρίας χρησιμοποιεί την μέθοδο της επαναποστολής μηνυμάτων. Τελικός στόχος του επιτιθέμενου είναι να αποκτήσει πρόσβαση στο δίκτυο της εταιρίας. Για την επιτυχή εξαπόλυση της επίθεσης ο κακόβουλος εκμεταλλεύεται την κρίση του απλού εργαζομένου για να αποκτήσει την πρόσβαση στο δίκτυο της εταιρίας και μετέπειτα στους τραπεζικούς λογαριασμούς των ανωτέρων. Στο ίδιο πλαίσιο ερμηνείας εντάσσεται και η επίθεση pretexting. Η ορολογία της δίνεται ως η προσπάθεια δημιουργίας ενός σεναρίου σε πλασματικές καταστάσεις κατά την διάρκεια της πρώτης φάσης κύκλου της κοινωνικής μηχανικής. Η ερμηνεία αυτήν ωστόσο δεν είναι τόσο καίρια. Pretexting είναι η διαδικασία κατά την οποία ο επιτιθέμενος υιοθετεί πλήρως τον ρολό του χαρακτήρα που θα υποδυθεί στον χρήστη. Αφομοιώνει συναισθήματα, τρόπο ομιλίας και επιχειρήματα, για να δημιουργήσει ένα φιλικό προς το χρήστη περιβάλλον που θα χρησιμοποιήσει για την εξαπάτηση του. Συνήθως η διαδικασία πραγματοποιείται χωρίς την φυσική παρουσία, αλλά μέσω τηλεφώνου, αναπαριστώντας κάποιον φορέα, αγοραστή, υπεύθυνο εργασίας οπού μέσω της συνομιλίας προσπαθεί να εξασφαλίσει την εμπιστοσύνη για να αποσπάσει δεδομένα που αφορούν την ταυτοποίηση του. Τρίτη κατηγορία αποτελεί η επίθεση baiting και βασίζεται στην περιέργεια και στην απληστία του χρήστη. Μέσω παραπλανητικών μηνυμάτων προωθούνται προϊόντα και υπηρεσίες οι οποίες δελεάζουν το ενδιαφέρον του χρήστη, υποβόσκοντας τη πιθανότητα εξαπάτησης τους. Αυτό συμβαίνει γιατί τα προϊόντα και οι υπηρεσίες που προωθούνται περιλαμβάνουν κακόβουλο πηγαίο κώδικα ο οποίος έχει ως στόχο να επηρεάσει την ακεραιότητα και την διαθεσιμότητα τόσο των υπολογιστικών συστημάτων όσο και των προσωπικών δεδομένων των χρηστών. Αναμφισβήτητα η επίθεση εξελίσσεται και με την χρήση φυσικών συσκευών όπως usb και cd, δηλαδή εξωτερικά αποθηκευτικά μέσα. Κατά την εγκατάσταση τους στην υπολογιστική συσκευή όλα λειτουργούν ομαλά ενώ στο παρασκήνιο εκτελείται ο κακόβουλος πηγαίος κώδικας ο οποίος επηρεάζει και αλλά προγράμματα του υπολογιστικού συστήματος εξαιτίας την ταχείας εξάπλωσης του. Τελευταία κατηγορία τεχνικής επίθεσης αποτελεί η tailgating, η οποία κυμαίνεται σε διαφορετικά πλαίσια σχετικά με τις προαναφερθέντες. Στην συγκεκριμένη είναι απαραίτητη η φυσική και εξ επαφής συμμετοχή του επιτιθέμενου για την επιτυχή ολοκλήρωση της. Το πρώτο στάδιο της επίθεσης είναι η τηλεφωνική επικοινωνία που διεξάγεται αναμεσα στον επιτιθέμενο και στο υπάλληλο μιας εταιρίας. Ο επιτιθέμενος υποδύεται έναν ρολό προσπαθώντας να εξασφαλίσει την εμπιστοσύνη του υπάλληλου για να αποκτήσει φυσική πρόσβαση στις εγκαταστάσεις της εταιρίας. Η ενέργεια αυτήν έχει σχεδόν παρόμοιο αντίκτυπο με την επίθεση pretexting. Έχοντας αποκτήσει πρόσβαση, ακολουθεί το δεύτερο στάδιο στο οποίο παρέχεται η δυνατότητα στον επιτιθέμενο να υποκλέψει εταιρικά στοιχεία της εταιρίας και όλων των εργαζομένων προκαλώντας καίρια τεχνικά προβλήματα στην διαχείριση και στην ασφάλεια της. Αναμφίβολα πληθώρα μεγάλων εταιριών διαχειρίζεται αυστηρό κύκλωμα παροχής

εισόδου στους εργαζομένους καθώς απαραίτητη είναι η χρήση καρτών εισόδου για την εξουσιοδότηση τους. Έχοντας αναφερθεί στις κατηγορίες των τεχνικών επίθεσης η κάθε μια αναλόγως την μεθοδολογία που ακολουθεί, συμπεριλαμβάνει διάφορων ειδών επιθέσεις η οποίες κρίνονται πιο εξειδικευμένες στην επιρροή που ασκούν στους χρήστες. Χάριν παραδείγματος η τεχνική phishing, πέραν των υποκατηγοριών στις οποίες διακρίνεται, περιέχει επιθέσεις όπως ransomware, shoulder surfing, reverse social engineering και fake software. Όλες λειτουργούν βασίζοντας στην τεχνική phishing αλλά η κάθε μια ξεχωριστά διαφέρει στον στόχο που απευθύνεται. Με τον ίδιο τρόπο λειτουργεί και η τεχνική επίθεσης pretexting η οποία συμπεριλαμβάνει τις επιθέσεις diversion theft, pop-up windows, phone social engineering, smishing. Τέλος η τεχνικές επιθέσεις baiting και tailgating, περιλαμβάνουν τις επιθέσεις impersonation on help desk, quid pro quo, online social engineering, pharming και dumpster diving, robocalls, stealing important documents και whitelisting flow αντίστοιχα. Παρακάτω το αντίστοιχο διάγραμμα κατηγοριοποίησής των τεχνικών επίθεσης.

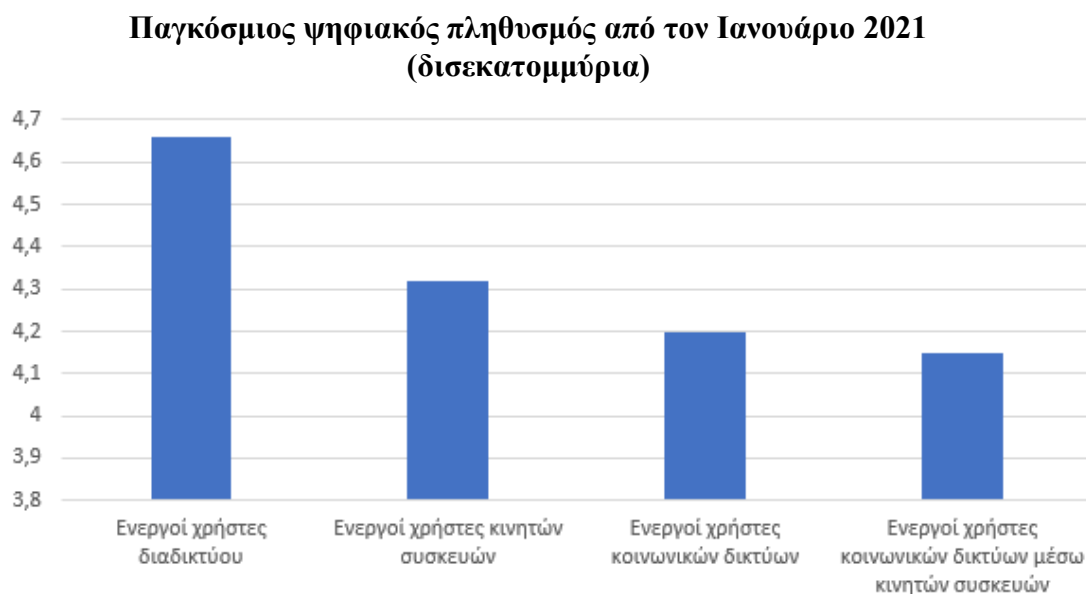


Εικόνα 2. Κατηγοριοποίηση τεχνικών επίθεσης κοινωνικής μηχανικής, School of Electrical Engineering and Computer Science, University of North Dakota

1.6 Στατιστικά διαδικτυακών επιθέσεων

Στην σύγχρονη εποχή σύμφωνα με το Statista (Statista, 2021), η ραγδαία αύξηση της τεχνολογίας και η καθημερινή χρήση του διαδικτύου επέφερε σημαντικές επιπτώσεις στην κυβερνοασφάλεια και στην ιδιωτικότητα παγκοσμίως. Σύμφωνα με την έρευνά που διεξάχθηκε στο πανεπιστήμιο αναφέρεται ότι 4.66 δισεκατομμύρια είναι ενεργοί χρήστες του διαδικτύου,

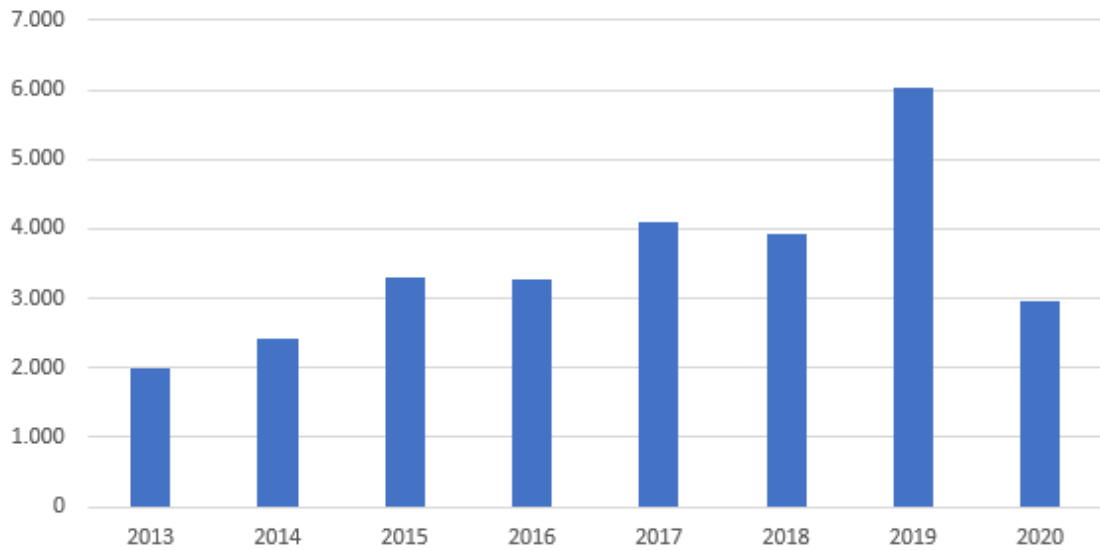
4.32 δισεκατομμύρια είναι ενεργοί χρήστες του διαδικτύου μέσω κινητής συσκευής, 4,2 δισεκατομμύρια μέσω κοινωνικής δικτύωσης και 4,15 δισεκατομμύρια χρήστες κοινωνικής δικτύωσης μέσω κινητής συσκευής όπως φαίνεται στο παρακάτω διάγραμμα.



Εικόνα 3. Στατιστικά συνολικών χρηστών της τεχνολογίας παγκοσμίως, THE U.N LOCAL GOVERNMENT BODIES, GSMA INTELLIGENCE, ITU, GWI, EUROSTAF, CNNNIC, APJII, SOCIAL MEDIA PLATFORMS.

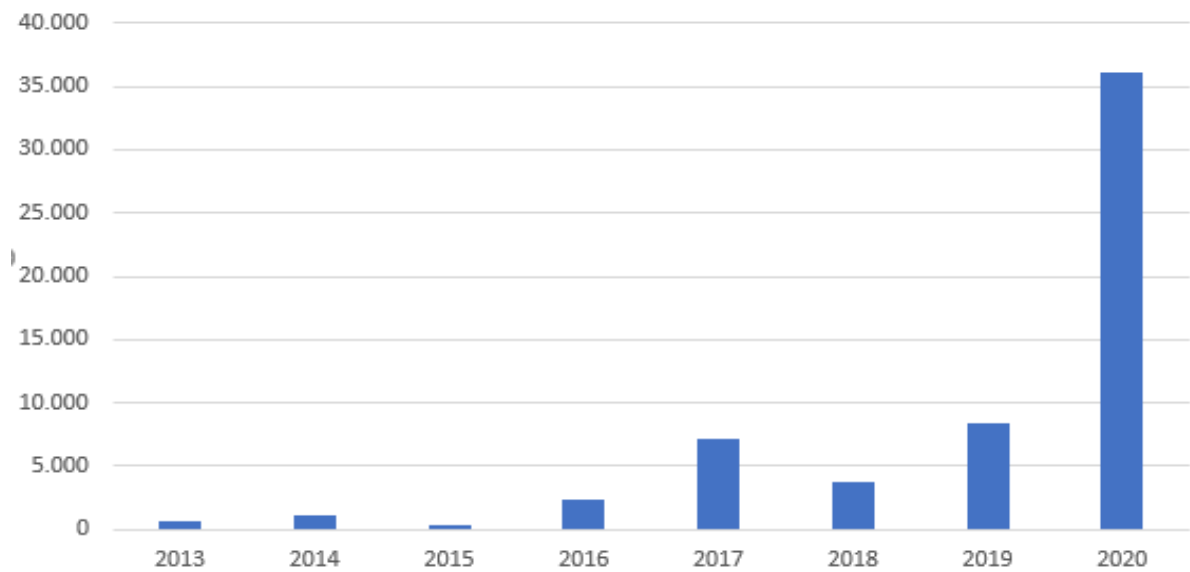
Η ανάγκη για μια εύκολη και πιο γρήγορη καθημερινότητα αυξάνει την απαίτηση κάλυψης περισσότερων πόρων της τεχνολογίας αλλά και τον κίνδυνο εξαπάτησης των χρηστών. Σημειώνεται ότι σύμφωνα με το Risk Based Security (Risk Based Security, 2020) με την τελευταία απογραφή που πραγματοποίησε συμπέρανε ότι από την χρονολογία 2019 έως το 2020 ο αριθμός των δεδομένων που κλάπηκαν αντιστοιχούν σε ποσό των 28 εκατομμυρίων ενώ τα breaches που έχουν καταγραφεί σχετικά με το 2019 είναι κατά 3.068 λιγότερα. Συμπεραίνεται ότι καθώς τα αποτελέσματα είναι δυσανάλογα, τα breaches που έγιναν το 2020 είχαν μεγαλύτερο αντίκτυπο στην κοινωνία. Τα σχετικά αποτελέσματα φαίνονται στο παρακάτω διάγραμμα.

Στατιστικά συνολικών breaches παγκοσμίως (εκατομμύρια)



Εικόνα 4. Στατιστικά συνολικών breaches παγκοσμίως, Risk Based

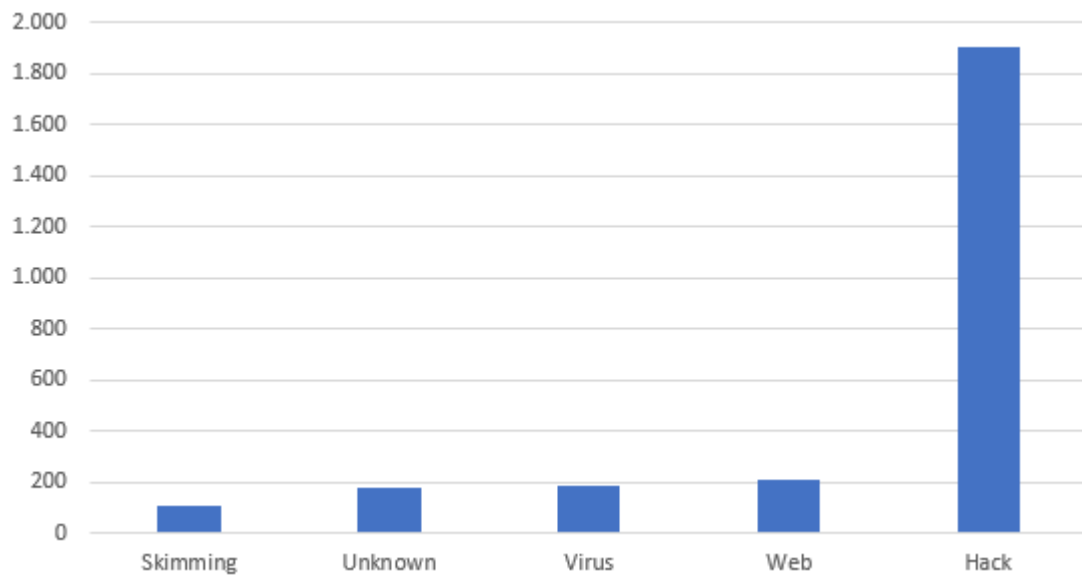
Στατιστικά συνολικής διαρροής δεδομένων παγκοσμίως (εκατομμύρια)



Εικόνα 5. Στατιστικά διαρροής δεδομένων παγκοσμίως, Risk Based

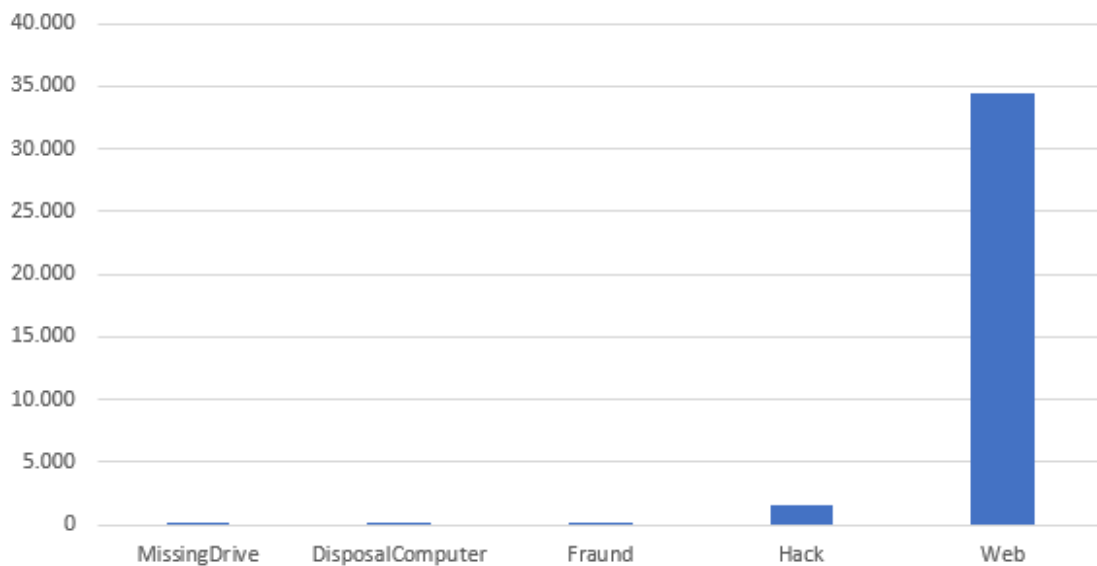
Από αυτά που σημειωθήκαν αναφέρεται το γεγονός ότι, βασιζόμενα σε ερευνα που διεξάχθηκε στο πανεπιστήμιο, τα 1.903 breaches είχαν χαρακτήρα επιθέσεων παράνομης πρόσβασης με συνολικά 34.420 εκατομμύρια δεδομένα να κλαπούν.

Στατιστικά χαρακτήρα επιθέσεων



Εικόνα 6. Στατιστικά συχνότερων κατηγοριών επιθέσεων παγκοσμίως, Risk Based

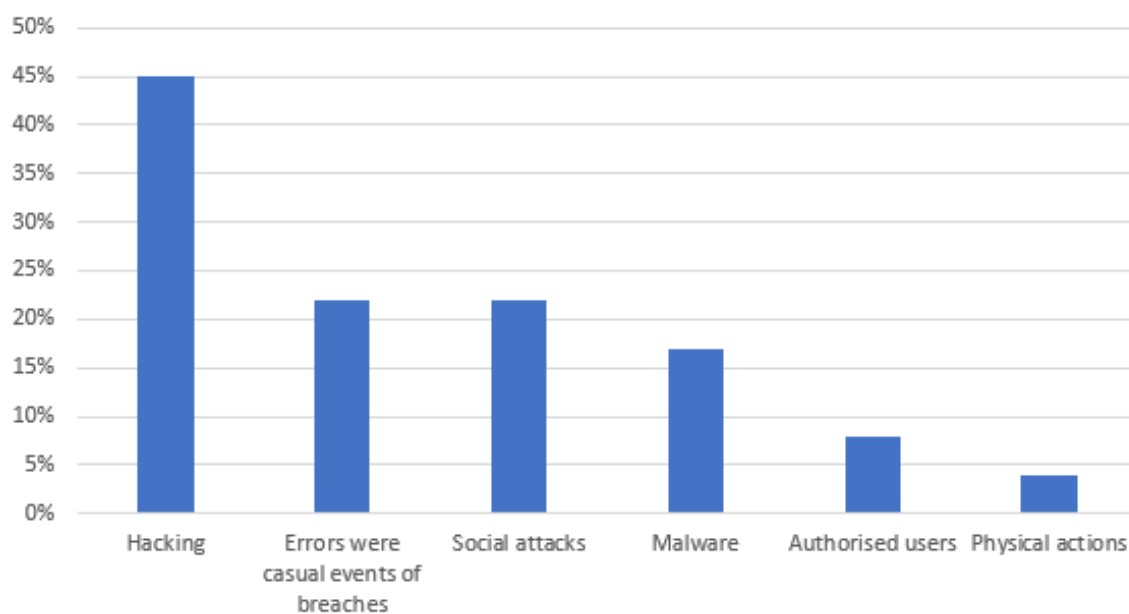
Στατιστικά διαρροής δεδομένων βάση χαρακτήρα επιθέσεων



Εικόνα 7. Στατιστικά διαρροής δεδομένων παγκοσμίως, Risk Based

Τα περισσότερα δεδομένα που κλάπηκαν αντιστοιχούν σε επιθέσεις τύπου phishing. Σύμφωνα με την έρευνά που διεξάχθηκε στο NAICS (North American Industry Classification System, 2020) σημειώνεται ότι 45% των κυβερνοεπιθέσεων που συνέβησαν το 2020 βασίζονται στο hacking βασισμένα σε social engineering ενώ με ποσοστό μόλις 22% εμφανίζεται η τεχνική επίθεσης hacking εκμεταλλευόμενα errors όπως φαίνεται στο παρακάτω διάγραμμα.

Στατιστικά ποσοστών επιθέσεων



Εικόνα 8. Στατιστικά πολυπληθέστερων κατηγοριών επιθέσεων παγκοσμίως, North American Industry Classification System (NAICS), Vocabulary for Event Recording and Incident Sharing (VERIS)

Τα δεδομένα με τα μεγαλύτερα ποσοστά κλοπής βάση ερευνών περιέχουν ηλεκτρονικές διευθύνσεις, κωδικούς και ονόματα ενώ δεδομένα τραπεζικών καρτών και ιατρικού περιεχομένου δεν έχουν μεγάλη απήχηση.

Πίνακας διαρροής δεδομένων.

Είδος δεδομένων	2020	2019	2018
Email	36%	64%	46%
Names	45%	29%	36%
Passwords	29%	58%	42%
Unknown	27%	11%	14%
Misc	28%	23%	19%
Ssn	26%	15%	19%
Address	22%	14%	20%
DoB	16%	9%	12%
Financial	16%	8%	12%
Medical	13%	7%	9%
Credit cards	12%	12%	15%
Accounts	10%	9%	7%

Εικόνα 9. Στατιστικά συνηθέστερων στοιχείων που διέρρευσαν παγκοσμίως, Risk Based

Στην εποχή της πανδημίας βάση στατιστικών του Risk Based (Risk Based, 2020) μεγαλύτερο ρολό διαδραματίζουν οι απάτες που βασίζονται στο social engineering και πιο συγκεκριμένα οι τεχνικές phishing. Οι επιθέσεις τύπου smishing έλαβαν καθοριστικό ρολό καθώς και οι επιθέσεις βασισμένες στις τηλεφωνικές επικοινωνίες δηλαδή vishing. Πολλοί είναι εκείνοι οι οποίοι βρίσκουν την εύκαιρα να εξαπατήσουν χρήστες μέσω social engineering βασίζόμενοι στην ανάγκη της τεχνολογίας που ευδοκιμεί. Βέβαια τα αποτελέσματα μέσω αυτών των καταστάσεων δεν είναι και τόσο ενθαρρυντικά καθώς άτομα μέσης και άνω ηλικίας αναγκάζονται να ολοκληρώσουν χρηματικές συναλλαγές μέσω διαδικτύου. Σύμφωνα με δεδομένα που συλλέχθηκαν, το FBI, ανέφερε ότι οι επιθέσεις τύπου Phishing, vishing, pharming και smishing εστίασαν σε πάνω από 200.000 θύματα όπως αναφέρεται στο παρακάτω πίνακα.

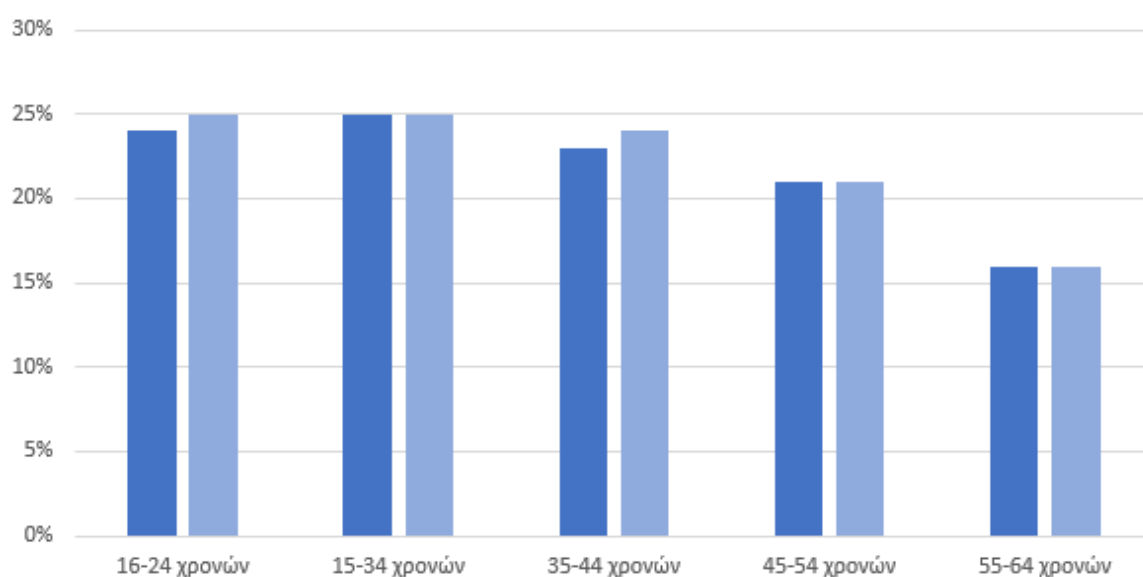
Πίνακας αριθμητικής υπεροχής σε τύπους ηλεκτρονικού εγκλήματος.

Αρίθμηση θυμάτων	
Τύπος εγκλήματος	Θύματα
Phishing/Vishing/Smishing/Pharming	241.342
Non-Payment	108.869
Extortion	76.741
Personal Data Breach	45.330
Identity Theft	43.330
Spoofing	28.218
Misrepresentation	24.276
Confidence Fraud/Romance	23.751
Harassment/Threats of Violence	20.604
BEC/EAC	19.369
Credit Card Fraud	17.614
Employment	16.879
Tech Support	15.421
Real Estate/Rental	13.638
Advanced fee	13.020
Government Impersonation	12.827
Overpayment	10.988

Εικόνα 10. Στατιστικά αριθμητικής υπεροχής σε τύπους ηλεκτρονικού εγκλήματος παγκοσμίως, Risk Based

Σύμφωνα με στατιστικά του THE U.N LOCAL GOVERNMENT BODIES (HoutSuite, 2021) της, το μεγαλύτερο ποσοστό που χρησιμοποιεί κινητές συσκευές για ηλεκτρονικές πληρωμές κυμαίνεται σε άτομα ηλικίας από 35-44 χρονών καθώς αποτελεί το 30,8% και 34,1% αντίστοιχα των συνολικών χρηστών που χρησιμοποιούν κινητές συσκευές παγκοσμίως όπως φαίνεται στο παρακάτω διάγραμμα.

Στατιστικά χρηστών ηλεκτρονικών συναλλαγών ηλικιακά



Εικόνα 11. Στατιστικά χρηστών ηλικιακά που χρησιμοποιούν ηλεκτρονικές συναλλαγές, THE U.N LOCAL GOVERNMENT BODIES, GSMA INTELLIGENCE, ITU, GWI, EUROSTAF, CNNNIC, APJII, SOCIAL MEDIA PLATFORMS.

Αυτό καθιστά αρκετά επικίνδυνο το ενδεχόμενο αρκετοί χρήστες να θεωρηθούν θύματα απάτης χρηματικών διαδικτυακών συναλλαγών. Η ελλιπής πληροφόρηση σχετικά με τους κινδύνους που προϋπάρχουν στο διαδίκτυο καθώς και με τα μετρά πρόληψης που πρέπει να παρθούν, επεκτείνουν το ενδεχόμενο παραβατικών ενεργειών. Μια αρκετά γνωστή και συνήθης επίθεση που ευδοκιμεί στην σύγχρονη εποχή είναι η sim swapping. Έχει παρατηρηθεί με την IOCTA EUROPALL (Iocta Europall, 2020) ότι η sim swapping είναι μια από τις συνηθέστερες και πιο διαδιδόμενες επιθέσεις στην εποχή του κορονοϊού παγκοσμίως, ενώ στην δεύτερη θέση βρίσκεται η παιδική πορνογραφία και οι επιθέσεις τύπου cyber crime. Αυτό το πιστοποιεί και η περίπτωση κλοπής δεδομένων χρηστών της εταιρίας T-mobile σύμφωνα με το BusinessWire και Motherboard. Το μέγεθος της απάτης αφορά 7.8 εκατομμύρια ενεργών συνδρομητών της εταιρίας κάθε και 40 εκατομμύρια μη πλέον ενεργών συνδρομητών, έχοντας όμως αιτήσει πιστοποίηση κάρτας της εταιρίας.

ΚΕΦΑΛΑΙΟ 2

SIM SWAP

Η χρήση των κινητών συσκευών έχει γίνει καθημερινή συνήθεια αρκετών χρηστών του διαδικτύου. Μέσω αυτών παρέχεται η δυνατότητα να καλύψουν πρωτεύοντες ανάγκες όπως είναι ηλεκτρονικές πληρωμές καθώς και συναλλαγές που αντιπροσωπεύουν τις απαιτήσεις της καθημερινότητας. Ο ρυθμός της σύγχρονης εποχής έχει αναγκάσει αρκετούς να δραστηριοποιηθούν και να συμβαδίσουν μαζί της εστιάζοντας στην εκμάθηση της λειτουργίας αυτών των κινητών συσκευών. Ωστόσο, η παροχή εύκολων λύσεων που δίνεται από την χρήση της τεχνολογίας περιέχει και κινδύνους που επηρεάζουν τόσο την προσωπική ζωή και ιδιωτικότητα των χρηστών όσο και της κοινωνίας εξαιρουμένης. Απαραίτητη, λοιπόν, είναι η λήψη μέτρων ασφάλειας για την αποφυγή της υπομόνευσης της ιδιωτικότητας των χρηστών.

2.1 Χρηματικές συναλλαγές στον χώρο του διαδικτύου

Τα τελευταία χρόνια η πραγματοποίηση διαδικτυακών χρηματικών συναλλαγών για την κάλυψη των καθημερινών αναγκών έχει γίνει συνήθειά αρκετών πολιτών. Πολλοί είναι εκείνοι οι οποίοι αποφεύγουν να χρησιμοποιούν φυσικά καταστήματα για εξαργύρωσή λογαριασμών ή μεταφορά χρήματος και καταφεύγουν στην εύκολη λύση μέσω της βοήθειας των κινητών συσκευών. Πολλές είναι οι επιχειρήσεις που καινοτομούν στην ανάπτυξη εφαρμογών παρέχοντας υπηρεσίες ηλεκτρονικών χρηματικών συναλλαγών κατονομαζόμενες ως e-banking. Οι αναφερόμενες υπηρεσίες χρειάζεται να χαρακτηρίζονται από πέντε διαστάσεις όπως η αξιοπιστία, η φιλική διεπαφή, οι υπηρεσίες είναι διαθέσιμες στον χρήστη, η υποστήριξη και τέλος και σημαντικότερο η ιδιωτικότητα και η ασφάλεια (Shankar, A., & Jebarajakirthy, C, 2019). Αυτές οι διαστάσεις χρειάζονται για την ολοκλήρωση του προφίλ των ηλεκτρονικών τραπεζών. Αναλυτικότερα, η διάσταση της διαθεσιμότητας ορίζεται ως η ικανότητα του e-banking να παρέχει τις υπηρεσίες της με ακρίβεια και συνέπεια. Στο εννοιολογικό περιεχόμενο των ηλεκτρονικών συναλλαγών, χρειάζεται να περιέχονται οι έννοιες όπως συνοχή, όπου αναφέρεται στην παροχή των ποιοτικών υπηρεσιών, η διαθεσιμότητα όπου τονίζει την παροχή υπηρεσιών χωρίς σφάλματα και τέλος η ακρίβεια όπου αναφέρεται στην ακριβή πραγματοποίηση του στόχου. Εν κατακλείδι το σημαντικότερο γνώρισμα της διάστασης της διαθεσιμότητας είναι η έγκαιρη πραγματοποίηση της συναλλαγής καθώς θεωρείται το κλειδί επιτυχίας της υπηρεσίας. Έπειτα, η διάσταση της φιλικής διεπαφής ορίζεται ως η δυνατότητα να παρέχονται διαφορά στοιχεία ξενάγησης στην εφαρμογή η οποία θα αλληλεπιδρά με την ευχέρεια του χρήστη να τις εκμεταλλευτεί. Αυτό προσδίδει αίσθηση

ευχαρίστησης και ενθουσιασμού με αποτελέσματα η χρήση της εφαρμογής να γίνεται συχνότερη συμβάλλοντας στην προτίμηση των χρηστών στους διαδικτυακούς παρόχους υπηρεσιών. Περαιτέρω, η διάσταση της παροχής υπηρεσιών και υποστήριξης αναφέρονται ως η ανάγκη για αναβάθμιση και up-to-date λογισμικού. Η συχνή κάλυψη των ευπαθειών που ενδεχόμενος να υπάρχουν, η προσθήκη νέων δυνατοτήτων στις υπηρεσίες, καθιστά τους χρήστες στο να διατηρήσουν σταθερή στάση στην προτίμηση της εφαρμογής. Τέλος η διάσταση της ιδιωτικότητας και της ασφαλείας που θεωρείται και η σημαντικότερη είναι εκείνη η οποία διαδραματίζει και καθοριστικό ρόλο στην ζήτηση της υπηρεσίας. Η διάσταση αυτήν, αναφέρεται ως η ανάγκη για εξασφάλιση της εμπιστοσύνης των χρηστών από τις εφαρμογές. Καθώς δεν υπάρχει η δυνατότητα της εξ επαφής αλληλεπίδρασης μεταξύ του χρήστη και του υπευθύνου χρηματικών συναλλαγών είναι θεωρητικά δύσβατη η εξασφάλιση της εμπιστοσύνης προσωπικών δεδομένων. Ωστόσο σε περίπτωση που οι διαδικτυακοί πάροχοι υπηρεσιών διασφαλίσουν την ακεραιότητα των ηλεκτρονικών συναλλαγών και την ιδιωτικότητα των προσωπικών δεδομένων, τότε η ζήτηση των υπηρεσιών θα είναι υψηλής εξαιτίας των προτιμήσεων των χρηστών. Γενικευμένα όλα τα παραπάνω συνάγουν στην αφοσίωση των χρηστών. Χάριν παραδείγματος είναι η διάδοση της ευχρηστίας των υπηρεσιών που προσφέρει η εφαρμογή σε άλλους χρήστες, αυξάνοντας με αυτόν τον τρόπο την ζήτηση. Αναμφισβήτητά η εστίαση στην ασφάλεια που χρειάζεται να παρέχει η υπηρεσία e-banking είναι αισθητή καθώς στον κυβερνοχώρο εξαιτίας του social engineering, οι χρηματικές ηλεκτρονικές συναλλαγές αποτελούν έναν από τους σημαντικότερους λογούς εξάπλωσης του cyber fraud. Πολλές είναι οι τεχνικές και οι μέθοδοι που μπορεί να ακολουθήσει ένας επιτιθέμενος για να παραβιάσει την ασφάλεια αυτών των υπηρεσιών αποσπώντας χρηματικά ποσά προς φόλες τους. Δικαιολογημένα οι περισσότερες επιθέσεις αναφέρονται στο social engineering και πιο συγκεκριμένα σε επιθέσεις phishing καθώς υπάρχει έλλειψη ευαισθητοποίησης σχετικά με τις επιθέσεις που προϋπάρχουν καθώς και με τις επιπτώσεις που προκύπτουν. Μια από τις συνηθέστερες επιθέσεις αποσπάσεις χρηματικού ποσού είναι η sim swap.

2.2 Έννοια της Sim Swap

Ως Swim Swap, βάση του International Journal of Trend in Scientific Research and Development (Awale, S. M., & Gupta, D. P, 2019) ορίζεται ως μια προσπάθεια εξαγοράς του ηλεκτρονικού λογαριασμού του θύματος, βασιζόμενο σε αδυναμία του επιπέδου ασφάλειας διπλής ταυτοπροσωπίας που παρέχεται στους χρήστες. Τεχνικά θεωρείται μια νέα μορφή κυβερνητικού εγκλήματος που στόχο έχει να αποσπάσει μεγάλα χρηματικά ποσά για προσωπικό συμφέρον. Βασίζεται σε επιθέσεις τύπου social engineering και εντάσσεται στην τεχνική επιθέσεων phishing καθώς στόχος είναι η υποκλοπή προσωπικών ευαίσθητων

δεδομένων. Πολλές φορές αναγνωρίζεται και με τις διαφορετικές ορολογίες όπως port-out scam, SIM splitting, Smishing και SIM jacking αλλά η συνηθέστερη ονομασία είναι sim swapping.

2.3 Εκμετάλλευση ευπαθειών

Βασικό χαρακτηριστικό της επίθεσης είναι η αξιολόγηση του χρήστη. Η sim swap είναι μια διαδικασία τριών βημάτων τα οποία ακολουθεί για να θεωρηθεί επιτυχής. Συγκεκριμένα περιλαμβάνει το στάδιο προσδιορισμού του στόχου, διεξαγωγής αναγνώρισης και τέλος της απόκτησης των προσωπικών δεδομένων. Εμβαθύνοντας, το πρώτο στάδιο αναφέρεται στην διαδικασία που ακολουθεί ο επιτιθέμενος για να αξιολογήσει την αξία του θύματος. Σύνηθες κέντρο προσοχής θεωρούνται αναγνωρισμένοι κυρίως χρήστες οπου προσωπικά στοιχεία όπως ονοματεπώνυμο είναι αναγνωρισμένα και διαθέσιμα μέσω διαδικτύου. Ωστόσο στην σύγχρονη εποχή στόχος είναι εκείνοι οι οποίοι διακρίνονται από έλλειψη ενημέρωσης σχετικά με τους κινδύνους της επίθεσης και γενικότερα με έλλειψη της σημασίας πρόληψης μέτρων προστασίας που χρειάζεται να ληφθούν για την εξασφάλισή της ιδιωτικότητας τους. Προσωπικά δεδομένα όπως ονοματεπώνυμο, διεύθυνση κατοικίας, διεύθυνση ηλεκτρονικού ταχυδρομείου και ενδεχομένως αριθμός τηλεφώνου διαδίδονται από χρήστη σε χρήστη ή αναρτώνται σε προσωπικούς λογαριασμούς κοινωνικών διαδικτυακών πλατφορμών καθώς αποτελούν βασικό στόχο των επιτιθέμενων και βασικό εργαλείο για την επιτυχή εξαπάτηση τους. Έπειτα ακολουθεί το στάδιο διεξαγωγής αναγνώρισης το οποίο μπορεί να παρομοιαστεί και ως ένα στάδιο παρακολούθησης της νοημοσύνης του χρήστη. Αυτό οφείλεται στην μέθοδο που ακολουθεί το συγκεκριμένο στάδιο το οποίο είναι η συλλογή όσο το δυνατόν περισσότερων προσωπικών ευαίσθητων δεδομένων είναι εφικτό. Σχεδιάζοντας ένας συνδυασμό πηγών εύρεσης πληροφοριών για την συλλογή προσωπικών δεδομένων του χρήστη, που θα κρίνει αποτελεσματική όχι μόνο στην επίθεση sim swap αλλά και σε επιθέσεις τύπου hacking. Τέλος, έχοντας συλλέξει αρκετά χαρακτηριστικά τα οποία ολοκληρώνουν την ταυτότητα του χρήστη, ακολουθεί το τελευταίο στάδιο, απόκτηση προσωπικών στοιχείων, το οποίο εκμεταλλεύεται την ευπάθεια των αγαθών.

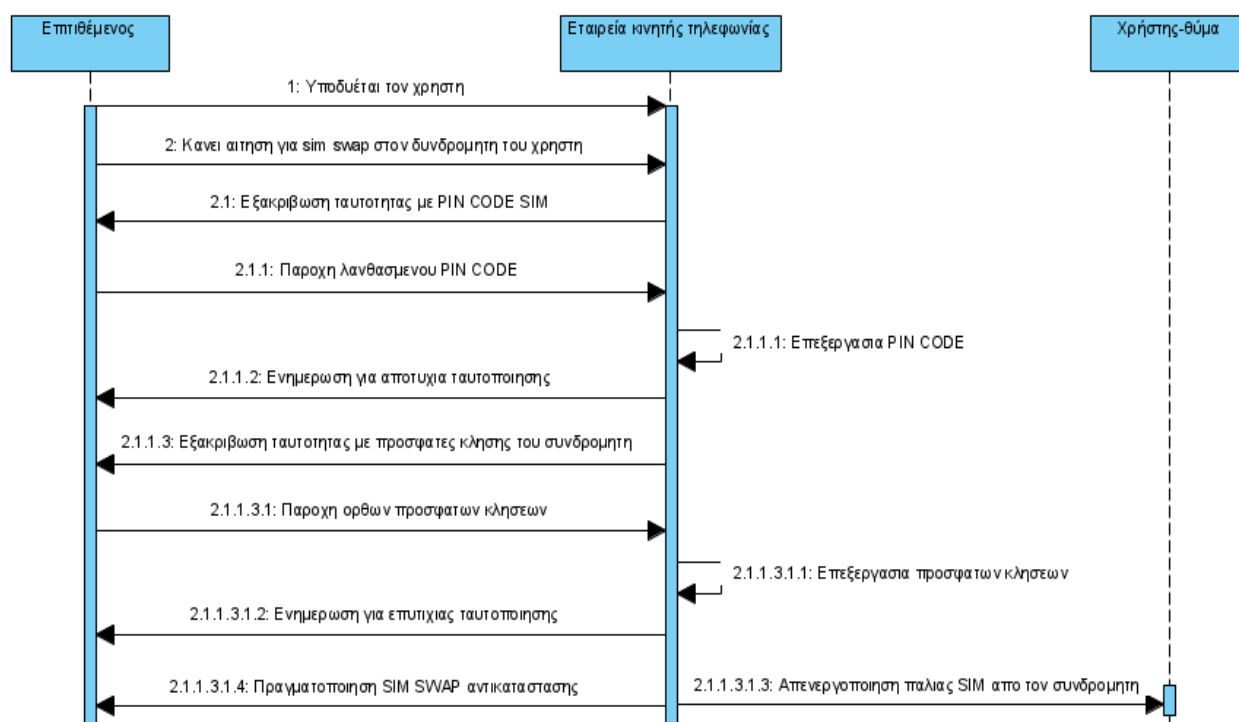
Τα παραπάνω στάδια βασίζονται στο social engineering αλλά χρησιμοποιούν δυο βασικές τεχνικές, Phishing και pretexting, συμπεριλαμβανομένων αρκετών μοτίβων επιθέσεων. Χαρακτηριστικό παράδειγμα είναι η τεχνική επίθεσης phishing η οποία χρησιμοποιείται για να υποκλέψει προσωπικά στοιχεία του χρήστη μέσω διάφορων καναλιών όπως email, ενημερωτικών φυλλαδίων, φυσικών μεθόδων και διάφορων άλλων όπως προαναφέρθηκαν στο προηγούμενο κεφάλαιο. Τέλος χρησιμοποιείται και η τεχνική επίθεσης pretexting που αποσκοπεί να εξασφαλίσει την εμπιστοσύνη του χρήστη για να αποσπάσει

ευαίσθητα προσωπικά δεδομένα. Οι προαναφερθέντες τεχνικές χρησιμοποιούν ως υπόβαθρο την αρχή elicitation για να εκμεταλλευτούν την νοημοσύνη του χρήστη και να ασκήσουν κυριαρχία πάνω τους ενώ βασική επιδίωξη της επίθεσης είναι η εκμετάλλευση των μηχανισμών ασφάλειας που παρέχουν οι υπηρεσίες e-banking.

2.4 Μεθοδολογία επίθεσης sim swap

Η επίθεση sim swapping έχει λάβει αυτήν την ονομασία καθώς στοχεύει στην εξασφάλιση της κυριότητας της κάρτας sim της κινητής συσκευής. Κάρτα sim θεωρείται η ύπαρξη ενός μικρού chip στις κινητές συσκευές, το οποίο χρησιμοποιείται για της τηλεπικοινωνίες και την αναγνώριση και αποθήκευση του προφίλ του χρήστη. Πολλές επιχειρήσεις χρησιμοποιούν την αναγνωρισιμότητα της κάρτας sim ως πρωτεύον κλειδί, δηλαδή ως κύριο μέσο αυθεντικοποίησης και εξουσιοδότησης του χρήστη. Στις κινητές συσκευές οι ασύρματες υπηρεσίες των εταιριών τηλεπικοινωνίας είναι συνδεδεμένες με αυτές των συσκευών της κάρτας sim. Οι εταιρίες κινητής τηλεφωνίας βρίσκονται άμεσα συνδεδεμένες καθώς παρακολουθούν την αντιστοίχιση μεταξύ των αριθμών τηλεφώνου των χρηστών και των καρτών sim για να διασφαλίσουν ότι οι κλήσεις, οι συνδέσεις και γενικότερα οι υπηρεσίες που δίνει μια κινητής τηλεφωνίας εταιρία, δρομολογούνται στον σωστό χρήστη. Γενικότερα η αντιστοίχιση ενός αριθμού τηλεφώνου με μια κάρτα sim διατηρεί δυαδική συσχέτιση έναν προς ένα καθώς ένας αριθμός τηλεφώνου συσχετίζεται με μια μόνο κάρτα sim επομένως με έναν και μοναδικό χρήστη. Υπάρχουν δυο σενάρια όπου η κάρτα sim δεν αντιστοιχίζεται με τον αριθμό τηλεφώνου σε περίπτωση που δεν υπάρχει εξουσιοδότηση για μια τέτοια πράξη. Οι περιπτώσεις είναι η sim swap και port out. Σχετικά με την πρώτη περίπτωση ο επιτιθέμενος, υποδύεται το θύμα και έχοντας ως στόχο να υποκλέψει την κυριότητα της κάρτας sim του, πραγματοποιεί αίτηση αλλαγής κάρτας sim με τον ίδιο αριθμό τηλεφώνου, στην κινητή του συσκευή. Ο πάροχος επικοινωνίας που είναι εγγεγραμμένος ο χρήστης, δέχεται το αίτημα και το θέτει σε επεξεργασία. Έπειτα για να επιβεβαιώσει το αίτημα ζητάει από τον υποτιθέμενο, που ισχυρίζεται τον χρήστη, να αυθεντικοποιήσει την ταυτότητα του δίνοντας στην εταιρία ευαίσθητα προσωπικά δεδομένα του χρήστη. Τα δεδομένα αυτά ωστόσο, μέσω τεχνικής Phishing, όπως προαναφέρθηκε παραπάνω, έχουν παρθεί από τον χρήστη και χρησιμοποιούνται πλέον από την πλευρά του επιτιθέμενου. Αδιάψευστα, απαραίτητο στοιχείο για την αυθεντικοποίηση του χρήστη είναι η εγκυρότητα του pin της κάρτας sim που αποτελείται από τέσσερα έως και οχτώ ψηφία. Τις περισσότερες φορές ο επιτιθέμενος δεν μπορεί να υποκλέψει μέσω μέθοδο phishing των κωδικό pin αλλά σε περίπτωση που καταφέρει να τον υποκλέψει δεν μπορεί να είναι σίγουρος για την εγκυρότητα του. Για αυτόν ακριβώς τον λόγο δηλώνεται στην υπηρεσία κινητής τηλεφωνίας λανθασμένος αριθμός pin της sim. Η εταιρία από την μεριά της είναι αναγκασμένη να ζητήσει έναν

διαφορετικό τρόπο αυθεντικοποίησης του χρήστη οπού είναι η μετονομασία προσφάτων κλήσεων που πραγματοποιήθηκαν από τον συγκεκριμένο συνδρομητή. Ο επιτιθέμενος επιβεβαιώνει ορθά, και πλέον η κυριότητα της κάρτας sim βρίσκεται στην κατοχή του επιτιθέμενου καθιστώντας την κάρτα sim του χρήστη εκτός υπηρεσιών. Η αναφερομένη διαδικασία φαίνεται στο παρακάτω διάγραμμα.



Εικόνα 12. Σχεδιάγραμμα μεθοδολογίας sim swap βάση διαγράμματος ακολουθίας.

2.5 Ανάλυση των μεθόδων ταυτοποίησης πολλαπλών παραγόντων

Οι επιθέσεις sim swap πραγματοποιούνται κυρίως για να αποσπάσουν χρηματικά ποσά από λογαριασμούς χρηστών για προσωπικό συμφέρον. Πλέον όλες οι εφαρμογές εταιρειών που εξυπηρετούν e-banking συναλλαγές, για την ασφάλη και αξιόπιστη ολοκλήρωση της συναλλαγής περιέχουν συστήματα ασφαλείς που παρέχουν εξουσιοδότηση στον χρήστη. Ανέκαθεν η χρήση passwords ήταν βασική προϋπόθεση για την ανάπτυξη μιας εφαρμογής συναλλαγών. Ωστόσο σύμφωνα με στο Department of Computer Science and Center for Information Technology Policy Princeton University (K.L., B.K., J.M., & A.N, 2020), η ύπαρξη συστημάτων με ασφάλεια μόνο κωδικών πρόσβασης, αξιολογήθηκε αρκετά ευπαθής με την πάροδο του χρόνο εξαιτίας της κυβερνητικής εγκληματικότητας που πρυτανεύει. Βάσιμο είναι η προσθήκη επιπλέον μέτρων ασφάλειας που βοηθάει τόσο την ιδιωτικότητα των χρηστών όσο και την αξιοπιστία της εταιρίας που έχει αναπτύξει την εφαρμογή. Έτσι μια

παρόμοια προσέγγιση στις e-banking συναλλαγές, που προϋποθέτει την ανάγκη για γνώση αλλά και κατοχή του χρήστη, είναι η υποστήριξη της πιστοποίησης δυο παραγόντων. Το συγκεκριμένο είδος ασφάλειας περιλαμβάνει διάφορων ειδών τύπων αυθεντικοποίησης του χρήστη. Μερικοί εξ αυτών είναι τα hardware tokens, SMS-text και voice-based OTP number, software tokens, push notifications και τέλος τα biometric tokens. Το καθένα από αυτά διαδραματίζει σημαντικό ρόλο στην ασφάλεια του χρήστη αλλά ακολουθεί διαφορετικό μοτίβο κρυπτογράφησης δεδομένων. Αναλυτικότερα τα hardware tokens περιγράφονται κυρίως ως φυσικές εξωτερικού αποθηκευτικού μέσου συσκευές οι οποίες παρέχουν πρόσβαση σε λογαριασμούς κατά την είσοδο τους στην υπολογιστική συσκευή. Αν και παρέχουν ευελιξία στην διαχείριση των λογαριασμών, δεν παρέχουν ασφάλεια καθώς είναι εύκολο σε επιτιθέμενους να δημιουργήσουν κλώνους τέτοιων συσκευών για να αποκτήσουν μη εξουσιοδοτημένη πρόσβαση. Έπειτα ακολουθούν τα software tokens τα οποία θεωρούνται μια από την σημαντικότερη πηγή ασφάλειας. Τα soft tokens ή TOTP περιέχουν έναν μοναδικό ηλεκτρονικό κωδικό ο οποίος δημιουργείται κατά την είσοδο του χρήστη στην υπηρεσία και ισχύει για ένα συγκεκριμένο χρονικό διάστημα. Επιπρόσθετα, push notifications είναι μια μέθοδος κατά την οποία δεν χρησιμοποιούνται κωδικοί μιας χρήσης και ηλεκτρονικά δεδομένα, αλλά χρησιμοποιούνται υπηρεσίες την εταιρίας και την φυσικής συσκευής του χρήστη. Έπειτα ακολουθούν τα biometrics tokens τα οποία χρησιμοποιούν βιομετρικά χαρακτηριστικά του χρήστη για την εξουσιοδότηση του. Τέλος και σημαντικότερο λαμβάνουν χώρα το SMS-text και voice-based OTP. Με αυτήν την μέθοδο για την ολοκλήρωση της συναλλαγής η υπηρεσία της εταιρίας στέλνει έναν μοναδικό αριθμό ο οποίος ισχύει για μια μόνο φορά και ορισμένο χρονικό διάστημα στον λογαριασμό του χρήστη. Με την ορθή συμπλήρωση του, θεωρείται η συναλλαγή επιτυχής.

2.5.1 Εκμετάλλευση μοναδικού αριθμού με την επίθεση sim swap

Με την επίθεση sim swap ο επιτιθέμενος προσπαθεί να παραβλέψει αυτόν τον μοναδικό αριθμό, χρησιμοποιώντας την κάρτα sim του θύματος. Όπως προαναφέρθηκε ο αριθμός τηλεφώνου, αρά και η κάρτα sim, αποτελεί πρωτεύον κλειδί για την αναγνώριση του χρήστη καθώς αποθηκεύεται στην βάση ως μοναδικό χαρακτηριστικό. Αυτό έχει ως αποτέλεσμα ότι κύριο μέσο αναγνωρισιμότητας και εξουσιοδότησης του χρήστη είναι ο αριθμός τηλεφώνου. Με αυτόν τον τρόπο το SMS-text και voice-based OTP number αποστέλλεται βάση του αριθμού τηλεφώνου μην έχοντας γνώση ποιος πραγματικά είναι δικαιούχος του αριθμού. Με την συγκεκριμένη διαδικασία οι επιτιθέμενοι και με μεθόδους υποκλοπής ευαίσθητων προσωπικών στοιχείων που προαναφέρθηκαν εξαπατούν τις τηλεφωνικές εταιρίες αλλά και τους χρήστες.

2.5.2 Εκμετάλλευση ευπαθειών των SMS μηνυμάτων

Σύμφωνα με μια έρευνά που διεξάχθηκε στο Department of Computer Science and Center for Information Technology Policy Princeton University (K.L., B.K., J.M., & A.N, 2020) αναφέρεται ότι υπάρχουν εταιρίες οι οποίες πραγματοποιούν αιτήματα sim swap αποκλειστικά απαιτώντας μόνο την εξακρίβωση προσωπικών στοιχείων. Μερικές από τις μεγαλύτερες εταιρίες τηλεπικοινωνιών όπως η AT&T, T-Mobile, Tracfone, US Mobile, και Verizon Wireless χρησιμοποιούν για την ταυτοποίηση του χρήστη πληροφορίες όπως προσωπικά δεδομένα, δεδομένα των λογαριασμών των χρηστών, πληροφορίες των φυσικών κινητών συσκευών, δεδομένα ιστορικού κλήσεων, πληροφορίες γνώσεων και στοιχεία προσωπικής απόκτησης. Αναλυτικότερα οι εταιρείες AT&T, T-Mobile, Tracfone, and Verizon χρησιμοποιούν μεθόδους πληρωμής για να τακτοποιήσουν τους χρήστες, κάτι το οποίο όμως κρίνεται ευπαθές καθώς ο επιτιθέμενος μπορεί να υποκλέψει τα στοιχεία πληρωμής του χρηστή και να τα χρησιμοποιήσει για να ταυτοποιήσει τον εαυτό του. Έπειτα οι εταιρίες AT&T, T-Mobile και Verizon χρησιμοποιούν μεθόδους επανάκτησης προσφάτων αριθμών τηλεφώνου από τον συγκεκριμένο συνδρομητή το οποίο όμως κρίνεται εξίσου ευπαθές αφού θεωρούνται πληροφορίες που αποκτώνται μέσω τεχνικών Phishing. Συνεχίζοντας οι εταιρίες Tracfone και US Mobile χρησιμοποιούν μεθόδους ταυτοποίησης με προσωπικές πληροφορίες κάτι το οποίο όμως κρίνεται ιδιαίτερα εύαλωτο καθώς τέτοιου είδους πληροφορίες είναι εκτεθειμένες στο διαδίκτυο ή κρίνονται ευκολά προστασίες με τεχνικές μεθόδων Phishing. Μετέπειτα οι εταιρίες AT&T, US Mobile, και Verizon χρησιμοποιούν μεθόδους ταυτοποίησης με την βοήθεια πληροφοριών των λογαριασμών. Παρόμοια με τις προσωπικές πληροφορίες, αυτήν η μέθοδος είναι αρκετά ευπαθής εξαιτίας της διαθεσιμότητάς που παρέχεται στο διαδίκτυο και κάτ' επέκταση στον επιτιθέμενο. Οι ερωτήσεις ασφάλειας χρησιμοποιούνται από τις εταιρίες Tracfone για ταυτοποίηση ενώ οι εταιρίες, T-Mobile, Tracfone, και Verizon τις χρησιμοποιούν για να ολοκληρώσουν την εγγραφή τους στο σύστημα τους. Οι ερωτήσεις ασφάλειας βάση ερευνών του Princeton University κρίνονται εξίσου ευπαθής μέθοδος ταυτοποίησης καθώς οι απαντήσεις που δίνονται, είναι ευκολά διαθέσιμες και αναγνώσιμες στο διαδίκτυο. Τέλος αξίζει να σημειωθεί ότι οι περισσότερες εταιρίες χρησιμοποιούν μεθόδους ταυτοποίησης μέσω ιδιωτικών πληροφοριών των συσκευών. Σημειώνεται ότι στις πληροφορίες των κινητών συσκευών, οι εταιρίες χρησιμοποιούν μόνο μοναδικούς αριθμούς που συσχετίζονται με την κάρτα SIM όπως για παράδειγμα ο αριθμός IMEI ο οποίος υποδηλώνει τον σειριακό αριθμό της συσκευής σε δίκτυο GSM και ο αριθμός ICCID ο οποίος υποδηλώνει τον σειριακό αριθμό της κάρτας SIM. Ωστόσο οι δυο αναγνωριστικοί αριθμοί συνδέονται άμεσα με το δίκτυο τηλεπικοινωνιών χωρίς να παρέχουν άμεση συσχέτιση με την κινητή συσκευή του χρήστη. Αυτός είναι ο λόγος που χαρακτηρίζει την μέθοδο ταυτοποίησης μέσω ιδιωτικών πληροφοριών

των συσκευών ως ευάλωτη καθώς και οι δυο intentifiers μπορούν να συνυπάρξουν στην κινητή συσκευή του επιτιθέμενου. Σύμφωνα (The Saturday Star. Jan. 7, 2008), οι συνδρομητές των εταιριών τηλεπικοινωνίας, σε έναν πίνακα RISK MATRIX αντιπροσωπεύουν υψηλά ποσοστά, καθώς οι επιπτώσεις είναι καταστροφικές ενώ η πιθανότητα εμφάνισης του περιστατικού είναι πιθανή. Αυτό συμβαίνει καθώς η επίθεση sim swap μπορεί να έχει ως αποτέλεσμα να εκθέσει τον χρήστη σε εξίσου κακόβουλες επιθέσεις όπως άρνησης υπηρεσίας, υποκλοπή ευαίσθητων επικοινωνιών και προσωποποιήσεις που θα οδηγήσει σε περαιτέρω περιπτώσεις συμβιβασμένων λογαριασμών. Όπως αναφέρθηκε προηγουμένως ένας επιτιθέμενος που μπορεί να λάβε την κυριότητα του αριθμού τηλεφώνου του μπορεί επίσης υποκλέψει τους κωδικούς ταυτότητας που υποκλέπτονται με SMS ή με τηλεφωνική κλήση. Αυτό ακριβώς αποτελεί την αδυναμία που εκμεταλλεύεται η επίθεση.

2.6 Μοναδικοί αναγνωριστικοί αριθμοί κινητών συσκευών

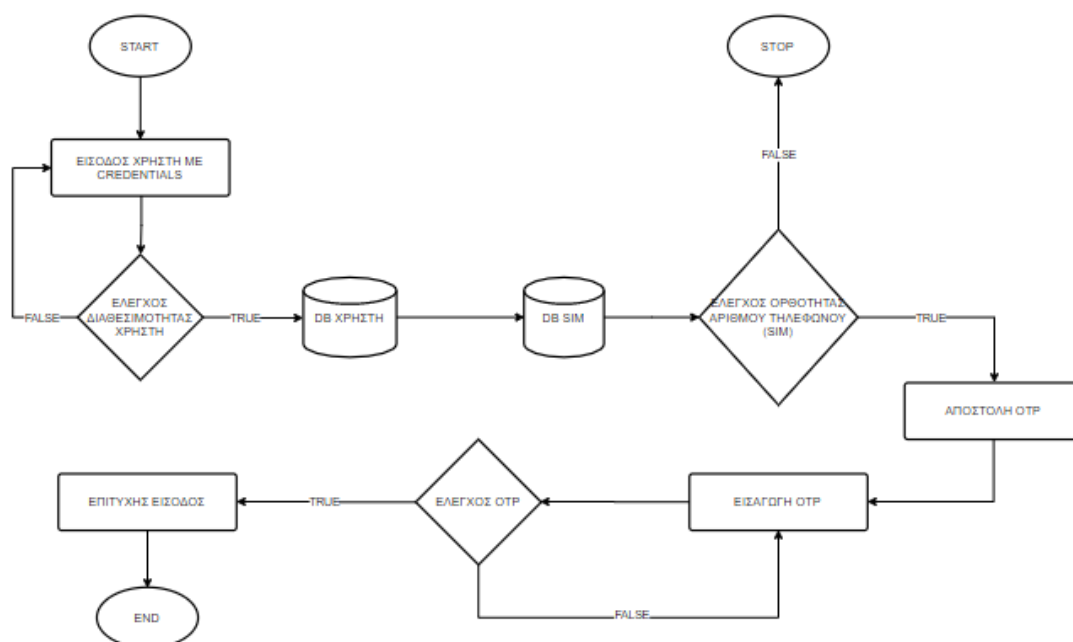
Όπως προαναφέρθηκε παραπάνω η επίθεση sim swap εκμεταλλεύει τους κωδικούς ταυτότητας OTP που στέλνονται κυρίως με SMS ή με τηλεφωνική κλήση, καθώς είναι απαραίτητοι για την επιτυχή πραγματοποίηση μιας διαδικτυακής χρηματικής συναλλαγής. Το SMS μήνυμα στέλνεται στην συσκευή του χρήστη βάση του αριθμού τηλεφώνου, κάθε η εταιρία τηλεπικοινωνίας που είναι εγγεγραμμένος ως ενεργός συνδρομητής ο χρήστης, κατέχει στην διάθεση της πληροφορίες όπως προσωπικά ευαίσθητα δεδομένα αλλά και μοναδικά αναγνωριστικά συσκευής όπως τον αριθμό IMEI και τον ICCID οι οποίοι συνδέονται άμεσα με την κάρτα SIM του χρήστη. Ωστόσο η αυθεντικοποίηση που βασίζεται σε αποστολή SMS, λόγω της συνεχής ενεργής συνδεσιμότητάς στο cellular network, υποβόσκει σοβαρά θέματα ασφάλειας. Λαμβάνοντας υπόψιν την τεχνολογική εξέλιξη της τηλεφωνικής επικοινωνίας η δεύτερη γενιά τηλεπικοινωνίας δηλαδή το 2G ή αλλιώς GSM πρωτόκολλο είναι εκείνο που διαδραμάτισε καθοριστικό ρολό στην ανάπτυξη και στην παροχή πόρων συνδεσιμότητας για πολλούς χρήστες. Σε συνδυασμό με την ευρεία διαθεσιμότητα του ανοιχτού κώδικα πρωτοκόλλου GSM είχε ως αποτέλεσμα την ανακάλυψή πολλών ευπαθειών στον σταθμό με σκοπό την εκμετάλλευσή του. Εκμεταλλεζόμενοι αυτήν την ευπάθεια τα μηνύματα SMS ήταν αρκετά εκτεθειμένα εξαιτίας του θορύβου που υπήρχε στο δίκτυο. Ως αποτέλεσμα αυτού, ορίζεται η εκμετάλλευσή του αριθμού τηλεφώνου εξαιτίας είτε του δικτύου στο οποίο βρίσκεται είτε των αναγνωριστικών αριθμών που παρέχει η κινητή συσκευή. Πέραν αυτών, οι κινητές συσκευές παρέχουν και άλλων ειδών μοναδικούς αριθμούς οι οποίοι όμως δεν συσχετίζονται με δίκτυα τηλεπικοινωνιών, κατ' επέκταση ούτε με την κάρτα sim αλλά ούτε με τον αριθμό τηλεφώνου αλλά συνδέονται μόνο με την φυσική κινητή συσκευή. Αναλυτικότερα είναι η MAC Address η οποία συνδέεται με την μητρική κάρτα της κινητής συσκευής και λαμβάνει την διεύθυνση του ασυρμάτου δικτύου Wi-Fi ή της διεύθυνση του Bluetooth. Έπειτα υπάρχει ο σειριακός αριθμός της συσκευής γνωστός ως device serial number ο οποίος είναι μοναδικός σε

κάθε συσκευή και είναι διαθέσιμος την στιγμή που δηλώνεται η συσκευή από τη εταιρία ως διαθέσιμη προς χρήση. Έπειτα λαμβάνει χώρα ο αριθμός UUID ο οποίος συμπεριφέρεται ως μοναδικός αριθμός κάθε φορά που πραγματοποιείται λήψη μιας εφαρμογής. Τέλος υπάρχει ο αριθμός secure AndroidID ο οποίος είναι διαθέσιμος την στιγμή που το κινητό κάνει εγκατάσταση εργοστασίων ρυθμίσεων. Αν και θεωρούνται ως ο ασφαλέστερος τρόπος αναγνώρισης μιας συσκευής, υπάρχουν περιπτώσεις που η χρήση τους είναι αναποτελεσματική. Συγκεκριμένα η mac address δεν βρίσκεται σε όλες τις συσκευές καθώς πολλές από αυτές δεν παρέχουν αισθητήρα σύνδεσης σε ασύρματο δίκτυο ή σύνδεσης σε Bluetooth αντίστοιχα. Από την άλλη μεριά ο σειριακός αριθμός, θεωρείται εξίσου αναποτελεσματικός καθώς δεν βρίσκεται σε όλες τις κινητές συσκευές από εργοστασιακής φύσεως. Έπειτα ο αριθμός UUID βρίσκεται σε όλες τις συσκευές καθώς δημιουργείται την στιγμή που πραγματοποιείται εγκατάσταση ενός προγράμματος. Ο χρήστης αναγνωρίζεται κάθε φορά που θα εισέλθει στην εφαρμογή καθώς ο αριθμούς αποθηκεύεται στην βάση δεδομένων. Τέλος ο αριθμός secure AndroidID θεωρείται ως η αποτελεσματικότερη μέθοδος αναγνώρισης μιας συγκεκριμένης φυσικής κινητής συσκευής ξεχωριστά καθώς υπάρχει σε όλα τις σύσκες από την στιγμή που βρίσκεται σε ενεργή λειτουργία.

2.7 Μεθοδολογία αποστολής SMS και αριθμού OTP μέσω ταυτοποίησης πολλών παραγόντων

Εξαιτίας της ανάγκης για επιπλέον ασφάλεια σε διαδικτυακές χρηματικές συναλλαγές, σχεδόν όλες οι εταιρίες χρησιμοποιούν στις εφαρμογές που έχουν αναπτύξει ασφάλεια διπλής ταυτοπροσωπίας ή αλλιώς 2FA. Μορφές ασφαλείας που περιλαμβάνονται στην TFA είναι οι hardware tokens, SMS-text και voice-based OTP number, software tokens, push notifications και τέλος τα biometric tokens όπως αναφέρονται και στην υποενότητα 2.4. Εμβαθύνοντας περισσότερο, η 2FA ή TFA αποτελεί μια μορφή ασφαλείας ενώ συνολικά υπάρχουν τρεις διαφορετικές όπως η SFA ή μονής ταυτοπροσωπίας και η MFA ή πολλαπλής ταυτοπροσωπίας. Ξεκινώντας με την SFA, οι απαιτήσεις που χρειάζονται για την αυθεντικοποίηση του χρήστη είναι η εισαγωγή ενός κωδικού πρόσβασης και το όνομα χρήστη που εισάγεται κατά την εγγραφή του στο σύστημα. Αυτό το είδος ασφαλείας συνηθιζόταν να χρησιμοποιείτε τα προηγούμενα χρόνια, ωστόσο με την έξαρση της ηλεκτρονικής εγκληματικότητας που επικρατεί στην σύγχρονη εποχή έχει αποφευχθεί η χρήση της. Έπειτα η TFA, τα τελευταία χρόνια χρησιμοποιείται από πάρα πολλούς οργανισμούς καθώς αποτελεί ένα αξιόπιστο μέσο αυθεντικοποίησης του χρήστη. Η διαδικασία απαιτεί την εισαγωγή του τετραψήφιου ή εξαψήφιου αριθμού μιας χρήσης για την ολοκλήρωσης της. Ο χρήστης έπειτα από την ορθή

συμπλήρωση των στοιχείων πρόσβασης χρειάζεται να συμπληρώσει τον αριθμό μιας χρήστης ο οποίος του αποστέλλεται μέσω SMS μήνυμα όπως φαίνεται στο παρακάτω σχήμα.



Εικόνα 13. Διάγραμμα τεχνικής διαδικασίας εξέλιξης της ασφάλειας εφαρμογής που βασίζεται στην εγκυρότητα της κάρτας sim

Τέλος η MFA, θεωρείται εξίσου εύχρηστη με την TFA με την διαφορά όμως ότι η MFA αποφεύγεται να χρησιμοποιείται κατά την πραγματοποίηση της συναλλαγής εξαιτίας της πολυπλοκότητας που προσφέρει στους χρήστες. Οι παραπάνω μορφές ασφάλειας υποστηρίζονται στις εφαρμογές διαδικτυακών εφαρμογών με την βοήθεια των API.

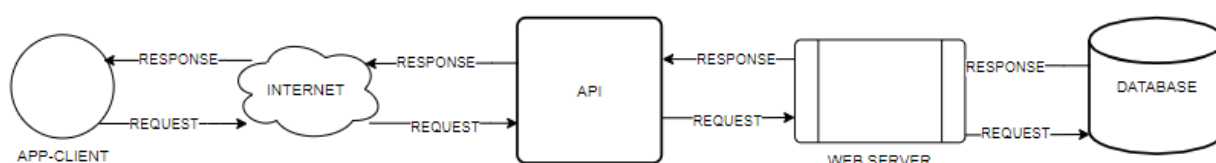
2.8 Αρχιτεκτονική συγχρόνων εφαρμογών

Πλέον όλες οι εταιρίες αναπτύσσουν εφαρμογές οι οποίες αποτελούνται από 2 στάδια, το front-end και το backend. Συνηθίζονταν τα λογισμικά ασφάλειας να προστίθεται στο front-end εξαιτίας της ευελιξίας και της έλλειψης πολυπλοκότητας που παρέχει στους προγραμματιστές. Ωστόσο η πράξη αυτήν, καθιστούσατε τα σύστημα αρκετά ευπαθή χαρακτηρίζοντας τις πληροφορίες από την πλευρά του server εκμεταλλεύσιμες από κακόβουλους προγραμματιστές. Με την πρόοδο της τεχνολογίας και τις ανοδικές αλλαγές που βιώσει, οι εφαρμογές που υποστηρίζουν λειτουργίες server και πιο συγκεκριμένα οι εφαρμογές e-banking, χρησιμοποιούν για τα λογισμικά ασφάλειας και κάθε είδους κρυφής πληροφορίας στον χρήστη, το backend κομμάτι του κώδικα δηλαδή στην μεριά του χρήστη. Στο front-end

χρησιμοποιούνται κυρίως δεδομένα και πληροφορίες οι οποίες είναι ορατές στον χρήστη όπως η διεπαφή του χρήστη της εφαρμογής. Είναι πληροφορίες που, εάν ένας επιτιθέμενος καταφέρει να εκμεταλλευτεί τις ευπάθειες της ασφάλειας του κώδικα, οι πληροφορίες αυτές δεν χαρακτηρίζονται ως σημαντικές και έτσι οι επιπτώσεις που θα έχει η υπομόνευση της ιδιωτικότητας του χρήστη και της ακεραιότητας και διαθεσιμότητά της εφαρμογής θα είναι ασήμαντες. Για αυτόν τον λόγο το front-end κομμάτι του κώδικα ονομάζεται και ως client-side rendering δηλαδή αναφέρεται στην πλευρά του χρήστη και το back-end του κώδικα ονομάζεται ως server-side rendering καθώς αναφέρεται στην πλευρά του server. Ως απόρροια αυτού το backend κομμάτι της εφαρμογής περιλαμβάνεται στο API.

2.8.1 Ορισμός και χρήση των API

Εκτενέστερα το API είναι ένας μηχανισμός ο οποίος επιτρέπει διάφορες εφαρμογές να επικοινωνούν μεταξύ τους. Συγκεκριμένα επιτρέπει λειτουργίες και δεδομένα βάσεων δεδομένων των εταιριών, να χρησιμοποιούνται από τρίτους εξωτερικούς οργανισμούς. Επιπρόσθετα περιέχει ορούς και κανόνες οι οποίοι επιτρέπουν την ανταλλαγή αυτών των δεδομένων. Η λειτουργία του API χαρακτηρίζεται από τέσσερα στάδια. Το πρώτο στάδιο είναι η αίτηση που στέλνει ο χρήστης ή client για να λάβει το δικαίωμα της μεταφοράς των δεδομένων. Δεδομένα δεν αποτελούν μόνο προσωπικές πληροφορίες που περιέχει μια επιχείρηση αλλά και πληροφορίες που περιέχει μια ιστοσελίδα. Η αίτηση που αποστέλνει ο χρήστη ονομάζεται αλλιώς και request, το οποίο λαμβάνει χωρά στον web server με την βοήθεια του URI δηλαδή του Uniform Resource Identifier που περιλαμβάνει ένα request web, headers και μερικές φορές request body. Έχοντας λάβει χωρά η αποστολή του request στο API από τον χρήστη, ακολουθεί το δεύτερο στάδιο το οποίο είναι η αίτηση που στέλνει το API με την σειρά του, στην εξωτερική πηγή πληροφοριών. Η αίτηση αυτήν καλεί να δώσει το δικαίωμα στην εξωτερική πηγή να μεταφέρει της πληροφορίες. Στην συνέχεια το τρίτο στάδιο περιλαμβάνει την απάντηση ή response που στέλνει ο server και η database στην αίτηση που έκανε το API για να λάβει πληροφορίες και να τις μεταφέρει. Τέλος το τελευταίο στάδιο, με την έγκυρη απάντηση του server στο API, αποστέλνει τα δεδομένα στο χρήστη που είχε πραγματοποιήσει το αρχικό request. Αξίζει να σημειωθεί ότι στην όλη διακόσμια περιλαμβάνονται δυο ειδών request και δυο ειδών responses όπως φαίνεται στο παρακάτω σχήμα.



Εικόνα 14. Διάγραμμα διαδικασίας αποστολής http request και http response διάμεσο μηχανισμού API.

2.8.2 Διακριτοποίηση των API

Τα API διακρίνονται στα private, στα public ή external και στα partner (What Is a REST API?, 2020). Αρχικά τα private είναι εκείνα τα οποία δημιουργούνται καθαρά για ιδιωτική χρήση. Χρησιμοποιούνται εξειδικευμένα από την εταιρία που τα δημιουργεί καθώς η λειτουργικότητα τους είναι μη διαθέσιμη σε άλλες εταιρίες. Είναι υπεύθυνα για να βελτιώσουν λύσεις και υπηρεσίες εσωτερικά στην επιχείρηση, καθώς και να αναπτύξουν νέες εφαρμογές που θα υποστηρίζουν τις παροχές του. Έπειτα τα public ή external APIs είναι εκείνα που οι υπηρεσίες που προσφέρουν είναι διαθέσιμες και σε εξωτερικές εταιρίες. Επίσης βοηθούν στην αύξηση της αναγνωρισιμότητας της δημιουργίας τους και με την πολύπλευρη χρήση τους βοηθούν στην αύξηση των πηγών εσόδων του. Βέβαια τα public ή external APIs διακρίνονται σε 2 υποκατηγορίες όπου είναι τα open και τα commercial. Τα open είναι εκείνα των οποίων οι υπηρεσίες είναι διαθέσιμες σε όλους, είτε είναι απλός χρήστης είτε είναι επιχείρηση, χωρίς την αποδοχή ορών πληρωμής ή ορών χρήσης του. Από την αντίθετη κατεύθυνσή, τα commercial προσφέρουν τις υπηρεσίες και τις βιβλιοθήκες τους μόνο υπό των ορών πληρωμής κάποιου χρηματικού ποσού. Μια αλλά οπτική που χρησιμοποιείται πολύ συχνά στις μέρες μας είναι, η υποβολή χρηματικού πόσου για την αξιοποίηση ολοκληρωμένων υπηρεσιών του API. Στην τελευταία κατηγορία βρίσκονται τα partners APIs τα οποία είναι έχουν δημιουργηθεί από μια εταιρία αλλά είναι διαθέσιμα σε τρίτους μόνο που διατηρούν είτε συνεταιρική σχέση με την εταιρία είτε διατηρούν τρέχων υπογεγραμμένο συμβόλαιο παροχής υπηρεσιών του. Τα APIs μπορούν να κατηγοριοποιούμεν βάση των συστημάτων που εξυπηρετούν. Έτσι, υπάρχουν τα database APIs τα οποία επιτρέπουν την επικοινωνία μεταξύ της βάσης δεδομένων και της φραγμούς της εκάστοτε εταιρίας. Υπάρχουν τα operating system APIs τα οποία χρησιμοποιούνται για την ομαλή και σωστή λειτουργία των λειτουργικών συστημάτων καθώς μεταφέρουν τα δεδομένα του υπολογιστή από υπηρεσία σε υπηρεσία. Έπειτα υπάρχουν τα remote APIs τα οποία βοηθάνε δυο εφαρμογές που βρίσκονται σε εικονική μηχανή να επικοινωνήσουν μεταξύ τους ανταλλάσσοντας πληροφορίες. Τέλος υπάρχουν τα web APIs τα οποία λειτουργούν με την διανομή request και response με την βοήθεια του πρωτοκόλλου HTTP ή Hypertext Transfer Protocol. Πέραν του γεγονότος ύπαρξης διάφορων κατηγοριών που υπάρχουν, τα APIs έχουν ένα κοινό χαρακτηριστικό, είναι να τυποποιήσουν την ανταλλαγή δεδομένων μεταξύ των υπηρεσιών δίκτυο. Αυτό σημαίνει να δίνεται η δυνατότητα σε διάφορα συστήματα να είναι γραμμένα σε ξεχωριστές γλώσσες προγραμματισμού, να έχουν διαφορετικές λειτουργίες σε διαφορετικά λειτουργικά συστήματα για να επικοινωνούν μεταξύ τους απρόσκοπτα. Ως αποτέλεσμα δημιουργήθηκαν τα πρωτοκόλλα των APIs τα οποία είναι το SOAP ή Simple Object Access Protocol, το XML-RPC, το JSON-RPC και το REST ή Representational State Transfer. Αναλυτικότερα το πρωτόκολλο SOAP σύμφωνα με την

Microsoft, χρησιμοποιείται για την ανταλλαγή δομημένων πληροφοριών σε κατανεμημένο περιβάλλον. Χρησιμοποιεί την γλώσσα προγραμματισμού XML ως βάση καθώς επιτρέπει τους χρήστες να στέλνουν και να λαμβάνουν δεδομένα μέσω του HTTP, όπως προαναφέρθηκε, καθώς και του SMTP το οποίο είναι πρωτόκολλο επικοινωνίας ηλεκτρονικών ταχυδρομικών μηνυμάτων. Το πρωτόκολλο SOAP χαρακτηρίζεται από την πλεονεκτική θέση που βρίσκεται καθώς η χρήση της δεν επιβαρύνει καθόλου τα υπολογιστικά συστήματα λόγω της εξοικονόμησης σε χρόνο και υπολογιστική ενέργεια που προσφέρει. Στην συνέχεια τα πρωτοκολλά XML-RPC, και JSON-RPC έχουν σχεδόν παρόμοιες ιδιότητες ενώ η χρήση του RPC περιλαμβάνει υπηρεσίες και αρχιτεκτονική του πρωτοκόλλου SOAP. Το χαρακτηριστικό που τα κάνει να διαφέρουν είναι η γλώσσα προγραμματισμού που υποστηρίζουν για να μεταφέρουν δεδομένα, καθώς το ένα στηρίζεται στην χρήση της γλώσσας XML ενώ το άλλο στην γλώσσα JSON αντίστοιχα. Τέλος, το REST ή Representational State Transfer διαφέρει από τα προηγούμενα και ως προς την λειτουργία αλλά και ως προς την δομή. Βασικό χαρακτηριστικό του REST είναι ότι δεν θεωρείται ακριβώς πρωτόκολλο αλλά υιοθετεί ένα αρχιτεκτονικό τύπο, ο οποίος λειτουργεί με τις προδιαγραφές RESTful APIs. Αξίζει να σημειωθεί ότι για να χαρακτηριστεί το REST ως RESTful χρειάζεται να πληροί ορισμένα χαρακτηριστικά όπως να ακολουθεί αρχιτεκτονική client-server δηλαδή επικοινωνία του χρήστη με http request και responses, να είναι stateless δηλαδή να μην αποθηκεύονται δεδομένα κατά την μεταφορά των requests και να είναι Cacheable δηλαδή να περιέχει δεδομένα προσωρινής αποθήκευσης που θα βελτιστοποιούν τις βελτιστοποιούν μεταξύ του πελάτη και του διακομιστή. Παράλληλα χρειάζεται να περιέχει uniform interface δηλαδή μία ομοιόμορφη διεπαφή έτσι ώστε τα δεδομένα να μεταφέρονται σε τυπική μορφή και τέλος να περιέχει layered system το οποίο βοηθάει στην οργάνωση των server. Συνοπτικά καθώς το πρωτόκολλο SOAP αλλά και το RESTful API είναι ευκολά στην χρήση, ευέλικτα, φιλικά στην ανάπτυξη υπηρεσιών ασφάλειας και πλέον πολύ διαδεδομένα έχουν ορισμένες διάφορες που τα καθιστούν διαφορετικά.

2.8.3 Βασικές διαφορές μεταξύ των APIs

Ένα εξ αυτών είναι ότι το πρωτόκολλο SOAP δεν είναι ικανό να κάνει χρήση των υπηρεσιών του RESTful καθώς το πρώτο θεωρείται πρωτόκολλο ενώ το δεύτερο αρχιτεκτονική σε αντίθεση με το RESTful που είναι ικανό. Επίσης το πρωτόκολλο SOAP μπορεί να υποστηρίξει μονάχα την γλώσσα XML για την προβολή των δεδομένων ενώ στην αντίθετη περίπτωση το RESTful υποστηρίζει γλώσσες προγραμματισμού όπως Plaintext, HTML, XML, JSON για την προβολή των δεδομένων. Τέλος και αρκετά σημαντικό είναι η πληθωρισμός δεδομένων που περιέχει ένα μήνυμα κατά την μεταφορά του εξαιτίας της πολλής πληροφορίας που διαθέτει, ενώ το RESTful προβάλλει το μήνυμα σε JSON μορφή περιλαμβάνοντας μόνο τα δεδομένα που μεταφέρονται και όχι πληροφορίες της αρχιτεκτονικής

JSON. Οι περισσότερες εταιρίες χρησιμοποιούν πλέον RESTful APIs λόγω των περιορισμών πόρων και της μικρής εμβέλειας που παρέχουν στην μεταφορά μηνυτών. Επιπλέον χάρη στο γεγονός ότι χαρακτηρίζεται Statelessness δηλαδή δεν αποθηκεύει δεδομένα σε http requests και responses ενισχύοντας την ασφάλεια και διασφαλίζοντάς την ιδιωτικότητα του χρήστη, ενώ παράλληλα μέσω caching ενισχύει την ταχύτητα που αποστέλλονται πληροφορίες μεταξύ του client και του server καθώς αποθηκεύονται προσωρινά στην μνήμη cache. Τέλος και πολύ σημαντικό, είναι το γεγονός ότι τα RESTful APIs προγραμματίζονται ευκολά εξαιτίας της πληθώρα των γλωσσών που υποστηρίζουν αλλά και την απλοϊκότητας που προσφέρουν στους χρήστες. Ως αποτέλεσμα του παραπάνω και της υπερφόρτωσης του διαδικτύου με πληροφορίες που προϋπάρχει, η χρήση των RESTful APIs είναι η προτιμότερη λύση καθώς υποστηρίζουν την αναρίθμητων δεδομένων ταυτόχρονα και με χρήση διαφορετικών servers.

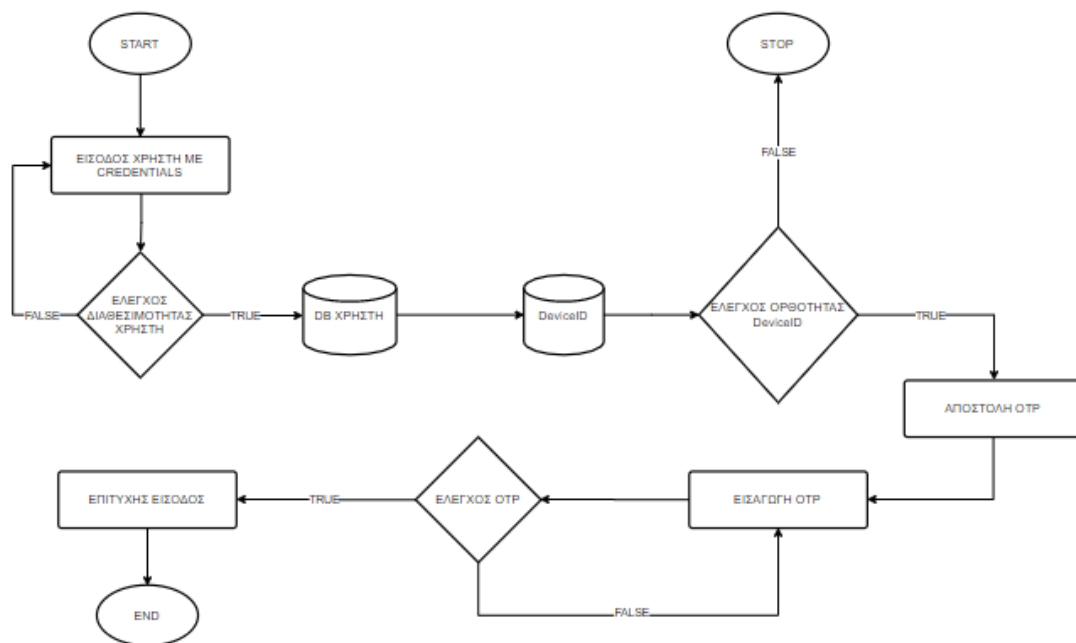
2.9 Η προτίμηση των εταιριών

Οι εταιρίες, λοιπόν, που αναπτύσσουν εφαρμογές που υποστηρίζουν παροχές ασφάλειας TFA και είναι χρήσιμες για διαδικτυακές ηλεκτρονικές συναλλαγές χρησιμοποιούν συγκεκριμένα RESTful APIs. Οι περισσότερες, έχοντας ως στόχο να αυθεντικοποιήσουν τον χρήστη, στέλνουν τον αριθμό OTP όπως προαναφέρθηκε στην προαναφέρθηκε 2.4, έχοντας ως εργαλείο προσωπικής ταυτοποίησης την κάρτα SIM και τον αριθμό του κινητού τηλεφώνου του χρήστη όπως φαίνεται στο σχήμα 2.9. Εξαιτίας ωστόσο των επιθέσεων SIM SWAP η διαδικασία αυτή κρίνεται ευάλωτη και όχι καθώς θέτει την ιδιωτικότητα των χρηστών και την ακεραιότητα των προσωπικών τους δεδομένων σε κίνδυνο. Ως αποτέλεσμα αυτού είναι η δημιουργία μιας εφαρμογής η οποία ακολουθεί το ίδιο μοτίβο αυθεντικοποίησης του χρήστη, καθώς βασίζεται στην ασφάλεια TFA αλλά έχει ως εργαλείο προσωπικής ταυτοποίησης τον κωδικό DeviceID της κινητής συσκευής ο οποίος κρίνεται ως η αποτελεσματικότερη μέθοδος όπως αναφέρεται στην αναφέρεται 2.6.

2.10 Θεωρητική ανάπτυξη εφαρμογής e-banking με χρήση του DeviceID

Όπως προαναφέρθηκε λοιπόν, εξαιτίας της συχνής εμφάνισης επιθέσεων τύπου SIM SWAP και της ανάγκης για εξασφάλιση της προστασίας ιδιωτικών δεδομένων βάσιμη είναι η ανάπτυξη μιας εφαρμογής που έχει στην διάθεση της να παρέχει όσο το δυνατόν περισσότερη ασφάλεια στους χρήστες. Η χρήση ενός unique identifier της κινητής συσκευής και η χρήση της ασφάλειας διπλής ταυτοπροσωπίας μπορεί να βοηθήσει αρκετά στην μείωση της εμφάνισης επιθέσεων που βασίζονται σε εξαπάτηση των παροχών τηλεφωνικών επικοινωνιών. Από τα unique identifiers που αναφέρθηκαν παραπάνω, προτιμότερη είναι η χρήση του

DeviceID εξαιτίας της ασφάλειας που προσφέρει και της δυνατότητας που παρέχει σε κάτοχους κινητών συσκευών , να έχουν τον πλήρη έλεγχο της συσκευής. Το μοτίβο που κρίνεται απαραίτητο να ακολουθεί η εφαρμογή αλλά και η αρχιτεκτονική στην οποία πρέπει να είναι δομημένη είναι παρόμοιο με αυτό των άλλων εφαρμογών με την διαφορά όμως ότι το μετρό αυθεντικοποίησης που χρειάζεται να χρησιμοποιεί για την αποστολή του OTP αριθμού, πρέπει να βασίζεται στην εγκυρότητα του DeviceID. Επιπρόσθετα, απαραίτητη είναι η ανάγκη ανάπτυξης ενός RESTful API, το οποίο χρησιμοποιεί τις δυνατότητες του Web API, για να μπορεί, μέσω του πρωτοκόλλου HTTP να στέλνει και δέχεται requests και responses για την μεταφορά και ανταλλαγή των δεδομένων από τον χρήστη στον server. Η διαδικασία η οποία περιγράφεται στην κάτω εικόνα αναλύει σε διάγραμμα την εξέλιξη της ταυτοποίησής και αυθεντικοποίησης του χρήστη με την βοήθεια του αριθμού DeviceID.



Εικόνα 15. Διάγραμμα τεχνικής διαδικασίας εξέλιξης της ασφάλειας εφαρμογής που βασίζεται στην εγκυρότητα του DeviceID.

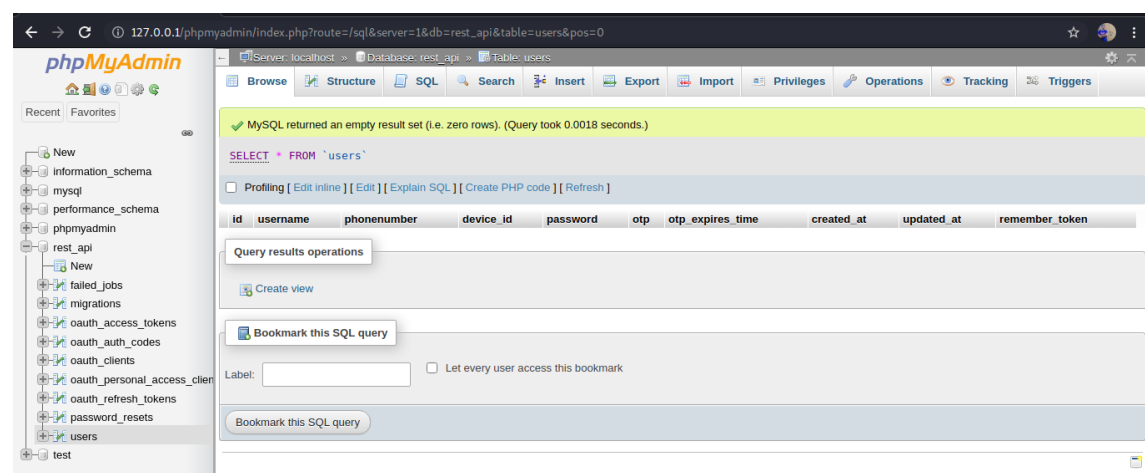
ΚΕΦΑΛΑΙΟ 3

ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ

Στο παρακάτω κεφάλαιο, βάση των όσων αναφέρθηκαν προηγουμένως κρίνεται απαραίτητο να αναφερθεί η εφαρμογή που αναπτύχθηκε με σκοπό να ελαχιστοποιήσει όσο το δυνατόν περισσότερο το ποσοστό εμφάνισης της επίθεσης SIM SWAP. Η εφαρμογή αποτελείται από δυο μέρη, το RESTful API το οποίο είναι δομημένο στο framework Laravel και αποτελεί το backend κομμάτι της εφαρμογής και στο android studio που αποτελεί το frontend κομμάτι. Οι γλώσσες προγραμματισμού οι οποίες βοήθησαν στην ανάπτυξη της εφαρμογής είναι η php για το API και η java για το android studio. Αναμφίβολα λαμβάνουν χώρα και η γλώσσα json καθώς χρησιμοποιείται για την προβολή των δεδομένων που μεταφέρονται με http requests και responses. Ως λειτουργικό σύστημα χρησιμοποιήθηκε τα kali linux λόγω της εξοικίωσης καθώς και της λειτουργικότητας που παρέχει.

3.1 REST API – BACKEND

Το backend κομμάτι της εφαρμογής αποτελείται από τον μηχανισμό RESTful API το οποίο συνδέεται με την βάση δεδομένων phpMyAdmin - MariaDB database. Η συγκεκριμένη βάση δεδομένων εμπεριέχεται στον web server XAMPP και η επιλογή του έχει γίνει καθαρά για όλους ευχρηστίας και ευεξίας με τον συγκεκριμένο ανοιχτό λογισμικό καθώς οι παροχές των υπηρεσιών είναι σε επίπεδο τοπικού δίκτυο ή localhost όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 16. MariaDB βάση δεδομένων.

Η υποστήριξη του RESTful API έγινε με την βοήθεια του Laravel framework. Η ορολογία framework αναφέρεται σε μια προκαθορισμένη δομή που υποστηρίζει για να εκτελεί ή να υποστηρίζει εργασίες. Αυτό που το κάνει να διαφέρει από τις βιβλιοθήκες είναι το γεγονός ότι το framework εκτελεί τον κώδικα που είναι ανεπτυγμένος πάνω σε αυτό ενώ οι βιβλιοθήκες εκτελούν τα συγκεκριμένα functions της γλώσσας προγραμματισμού που χρησιμοποιείται. Συγκεκριμένα το Laravel framework παρέχει μια εκφραστική και κομψή μορφή σύνταξης καθώς ακολουθεί την αρχιτεκτονική model-view-controller η οποία καθιστά την μεθοδολογία του αρκετά κατανοητή και ευέλικτη. Επίσης παρέχει προσωπικό του τοπικού δίκτυο web server ο οποίος συνδέεται άμεσα με τον server της database. Παράλληλα η χρήση του έγινε βάση της προτίμησης σε γλώσσα προγραμματισμού Php. Τέλος, ως πρόγραμμα ανάπτυξης του προγράμματος χρησιμοποιήθηκε το atom καθαρά για λογούς εξοικείωσης. Η εφαρμογή υποστηρίζει υπηρεσίες όπως εγγραφής του χρήστη, είσοδο του χρήστη στον λογαριασμό, έλεγχος εγκυρότητας του OTP αριθμού και τέλος έξοδο του χρήστη από το προφίλ του. Αρχικά λαμβάνει χώρα η δημιουργία της βάσης δεδομένων και των tables που χρειάζονται για να τοποθετηθούν οι μεταβλητές. Για καλύτερη κατανόηση της εφαρμογής προτείνεται η προσομοίωση της με το προφίλ μιας τράπεζας όπου η βάση δεδομένων phpMyAdmin - MariaDB database αντιστοιχείται στην βάση δεδομένων της τράπεζης. Αρχικά το Laravel framework δίνει την δυνατότητα δημιουργίας custom βάσης δεδομένων και tables καθώς και αρχικοποίησής μεταβλητών που αποθηκεύονται στην βάση. Έτσι δημιουργήθηκε το table users με μεταβλητές όπως id, username, phone number, device_id, password, otp, otp_expires_time, timestamps και rememberToken. αυτές οι μεταβλητές αποτελούν και τα δεδομένα που λαμβάνουν χώρα στην εξέλιξη της εφαρμογής. Παράλληλα έγινε αρχικοποίηση και των χαρακτήρων που μπορεί να λάβει η κάθε μεταβλητή ξεχωριστά όπως text, bigInteger, string, integer και dateTime ενώ το phonenummer θεωρείται ως πρωτεύον κλειδί και το otp και το otp_expires_time μπορεί να μην λάβει και κάποια τιμή όπως φαίνεται στην παρακάτω φωτογραφία.

```

public function up()
{
    Schema::create('users', function (Blueprint $table) {
        $table->id();
        $table->text('username');
        $table->bigInteger('phonenumber')->unique();
        $table->text('device_id');
        $table->string('password', 64);
        $table->integer('otp')->nullable();
        $table->dateTime('otp_expires_time')->nullable();
        $table->timestamps();
        $table->rememberToken();
    });
}

```

Εικόνα 17. Εικόνα δημιουργίας table μεταβλητών και αρχικοποιήσεις χαρακτήρων.

Η διαδικασία της αρχικοποίησής των μεταβλητών συνεχίζεται καθώς το Laravel παρέχει υπηρεσίες για προστασία μεταβλητών από τύπου επιθέσεων sql injection. Αυτό ολοκληρώνεται με το function protected που μπορεί να αναφέρεται είτε σε κάποιο writeable πεδίο, είτε σε κάποια ημερομηνία είτε σε κάποιο κρυφό πεδίο όπως κωδικοί. Για αυτόν τον λόγο χρησιμοποιείται αυτόν το function καθώς στο function writeable έχουν προστεθεί οι μεταβλητές username, password, phonenumber, device_id, otp και otp_expires_time ενώ στο function hidden έχει προστεθεί μόνο το password και στο function με τις ημερομηνίες έχουν προστεθεί το timestamps και το otp_expires_time όπως φαίνεται στην παρακάτω φωτογραφία αναλυτικότερα.

```

protected $fillable = [
    'username',
    'phonenummer',
    'password',
    'device_id',
    'otp',
    'otp_expires_time',
];

/**
 * The attributes that should be hidden for arrays.
 *
 * @var array
 */
protected $hidden = [
    'password',
];

```

Εικόνα 18. Εικόνα προσθήκης μεταβλητών σε function προστασίας από επιθέσεις τύπου sql injection.

3.1.1 Διαδικασία εγγραφής του χρήστη

Ως πρώτη διαδικασία της εφαρμογής διαδραματίζει η υπηρεσία registration. Κατά την εγγραφή του χρήστη στην βάση δεδομένων, μέσω request στέλνεται το username, phonenummer to device_id και το password και tokens. Τα tokens παρέχονται default από το Laravel ως ένδειξη ταυτοποίησης δηλαδή χαρακτηρίζουν το κάθε χρήστη ξεχωριστά. Οι τιμές αυτές όπως προαναφέρθηκε στέλνονται στην βάση μέσω request δηλαδή λαμβάνονται από τον client και το API τα στέλνει στην βάση δεδομένων. Αρχικά όμως λαμβάνει χώρα η διαδικασία ορθότητας των μεταβλητών μέσω του function validation. Το validation παρέχεται από το Laravel για να ελεγεία την ορθότητα και εγκυρότητα των στοιχείων κατά το request. Οι κανόνες του validation δίνονται default από το framework, ωστόσο για τις προσωπικές απαιτήσεις χρειάστηκε να δημιουργηθούν custom validation κανόνες όπως φαίνεται στην παρακάτω φωτογραφία. Οι κανόνες αυτοί περιλαμβάνουν εγκυρότητα χαρακτήρων και αριθμών. Έπειτα για την ολοκλήρωση της διαδικασία της ενεγράφης πραγματοποιείται ένας έλεγχος ορθότητας των κανόνων validation και στέλνεται το ανάλογο http request ή response μήνυμα στον server. Προς καλύτερη κατανόηση της χρήσης των εντολών έχουν τοποθετηθεί σχόλια στον κώδικα όπως διακρίνεται στην παρακάτω εικόνα.

```

public function register(Request $request)
{
    //edo ginontai ta validation
    $validated = $request->validate([
        'username' => 'required|max:255|regex:/^[A-Za-z0-9]+$/',
        'password' => 'required',
        'phonenumber' => 'required|digits:10',
        'device_id' => 'required'
    ]);

    //edo arxikopoio ena prokathorismeno minima
    $registerSuccessMessage = "The registration is successfull!";

    //edo o dinontai ta fields ta opoia tha stelnei request o xristis gia to registration
    $user = User::create([
        'username' => $request->username,
        'phonenumber' => $request->phonenumber,
        'device_id' => bcrypt($request->device_id),
        'password' => bcrypt($request->password),
    ]);

```

Εικόνα 19. Εικόνα διαδικασίας εγγραφής του χρήστη (1).

Σε αυτό το σημείο λαμβάνουν χώρα οι διαδικασίες που πραγματοποιούνται για την αρχικοποίηση των μεταβλητών και των κανόνων validation που χρειάζονται για την έγκυρη υποβολή των στοιχείων του χρήστη.

```

//edo me to function if elegxo an einai egkyro oi kanones validation poy exo thesei
if($validated)
{
    //edo me to function if elegxo an einai egkyra oi kanones registration poy exo steilei
    if($user)
    {
        //epistrefo ena response se morfi JSON me to prokathorismeno minima kai to minima success
        return response()->json(array(
            'registerSuccessMessage' => $registerSuccessMessage,
            'success' => true
        ));
        // 'user_secret_key' => $user_secret_key
    }
}

```

Εικόνα 20. Εικόνα διαδικασίας εγγραφής του χρήστη (2).

3.1.2 Διαδικασία εισόδου του χρήστη

Ως δεύτερη υπηρεσία της εφαρμογής αποτελεί η διαδικασία login δηλαδή είσοδος του χρήστη με τα στοιχεία πρόσβασης του στο σύστημα. Κατά την εισοδο στο σύστημα ο χρηστης απαιτείται να συμπληρώσει τα στοιχεία πρόσβασης του. το API εξαιτίας της λειτουργίας του δεχεται τα request με τα στοιχεία που έχει συμπληρώσει ο χρηστης για να εισέλθει στον

λογαριασμό του. το API, έπειτα στέλνει αυτά τα στοιχεία στην βάση και ελέγχει την εγκυρότητα τους. Η διαδικασία αυτή περιγράφεται παρακάτω. Αναλυτικότερα, ως πρώτο βήμα ελέγχονται οι κανόνες validation για τα στοιχεία πρόσβασης του και αρχικοποιούνται οι μεταβλητές που απαιτούνται για την εισοδο του χρήστη στο σύστημα. Έπειτα ξεκινάει η διαδικασία κατά την οποία ελέγχεται η εγκυρότητα των στοιχείων πρόσβασης με αυτών που είχαν σταλθεί στην βάση στο προηγούμενο στάδιο δηλαδή στην εγγραφή του χρήστη. Αφού ολοκληρωθεί ο έλεγχος εγκυρότητας των στοιχείων και του device_id τότε αποστέλνεται ένας εξαψήφιος αριθμός μιας χρήσης ο οποίος ισχύει για ένα χρονικό διάστημα σαράντα δευτερολέπτων από την τρέχουσα στιγμή. Ο αριθμός αυτός είναι ο OTP και ο χρόνος είναι ο otp_expires_time όπως φαίνονται και στις παρακάτω εικόνες. Έχοντας λοιπόν πραγματοποιηθεί με επιτυχία η ολοκλήρωση της εισόδου του χρήστη τότε αποστέλνεται ένα response από την βάση με τα στοιχεία που φαίνονται στις εικόνες. Εκτενέστερη επεξήγηση δίνεται από τα σχόλια που παρέχονται στον κώδικα παρακάτω.

```
public function login(Request $request)
{
    //orizo tous kanones validation
    $validated = $request->validate([
        'username' => 'required',
        'password' => 'required',
        'phonenumber' => 'required',
    ]);

    //arxikopoio tis metablites pou xreiazetai na stelontai mesw request gia to login
    $credentials = [

        'username' => $request->username,
        'password' => $request->password,
        'phonenumber' => $request->phonenumber,

    ];
}
```

Εικόνα 21. Εικόνα διαδικασίας εισόδου του χρήστη με τα στοιχεία πρόσβασης.

Έχοντας ως συνέχεια πραγματοποιούνται διαδικασίες οι οποίες οριοθετούν και οριοθετούν μεταβλητές που είναι υπεύθυνες για την πραγματοποίηση εισόδου του χρήστη.

```

        {
            $device_id = DB::table('users')->select('device_id')->where('username', '=', $request->username)
                ->where('phonenumber', '=', $request->phonenumber)
                ->where('password', '=', $request->password);

            //edo me to function if dhlnv se periptosi pou to device_id einai lanthasmeno tote ti ginetai
            if(!$device_id)
            {

                //epistreto error
                return ('error');

                //sthn periptosh pou to device_id einai svsto
            } else {

                //dhmioyrgv metablhth otp me 6 diaforetika tuxaia psifia
                $otp = rand(100000,999999);

                //dhmioyrgv metablhth pou dhlnv thn trexousa vra kai prostheo 40 deuterolepta
                $otp_expires_time = Carbon::now()->addSeconds(40);
            }
        }
    }
}

```

Εικόνα 22. Εικόνα διαδικασίας εισόδου του χρήστη με τα στοιχεία πρόσβασης και δημιουργία OTP (1).

```

public function confirmOtp()
{
    //edo arxikopoio ton xrono pou tha leigei o otp me to Carbon tou laravel kai prostheo 40 deyterolepta
    $otp_expires_time = Carbon::now()->addSeconds(40);

    //edo dilnv na elegxetai o xrhsths mesa sthn basi pou o otp einai idios me auton apo to request
    $user = User::where('otp', '=', request('otp'))->first();

    //edp arxizei to function if gia thn egkirotita toy otp
    if ($user)
    {
        //me auto to function if elegxo to gegonos an o xronos tou otp pou exv arxikopoihsh
        //einai megalyteros tou trexontos
        if ($user->otp_expires_time > Carbon::now())
        {
            //se periptosi pou isxuei to panv function if tote sundeo ton xrhsth me thn bash
            Auth::login($user, true);
        }
    }
}

```

Εικόνα 23. Εικόνα διαδικασίας εισόδου του χρήστη με τα στοιχεία πρόσβασης και δημιουργία OTP (2).

3.1.3 Διαδικασία ελέγχου εγκυρότητας του OTP

Τρίτη διαδικασία που λαμβάνει χώρα στην εφαρμογή είναι ο έλεγχος της εγκυρότητας του OTP αριθμού και του χρόνου λήξης του. έχοντας πραγματοποιηθεί με επιτυχία η διαδικασία της εισόδου του χρήστη με τα προσωπικά του στοιχεία και της αποστολής του OTP αριθμού, επόμενο στάδιο είναι η εγκυρότητα του με σκοπό της είσοδο του χρήστη στο προφίλ του. Στην διαδικασία αυτήν ελέγχεται με functions εάν το request που έχει στείλει ο χρήστης με τον OTP είναι έγκυρο και εντός χρονικού περιθωρίου, τότε να ολοκληρώσει την

αυθεντικοποίηση του εμφανίζοντας του μέσω response τα στοιχεία του προφίλ του όπως φαίνεται στην παρακάτω φωτογραφία.

```
public function confirmOtp()
{
    //edo arxikopoio ton xrono poy tha leigei o otp me to Carbon tou laravel kai prostheo 40 deyterolepta
    $otp_expires_time = Carbon::now()->addSeconds(40);

    //edo dilvnn na elegxetai o xrhsths mesa sthn basi pou o otp einai idios me auton apo to request
    $user = User::where('otp', '=', request('otp'))->first();

    //edp arxizei to function if gia thn egkirotita toy otp
    if ($user)
    {
        //me auto to function if elegxo to gegonos an o xronos tou otp poy exv arxikopoihsh
        //einai megalyteros tou trexontos
        if ($user->otp_expires_time > Carbon::now())
        {
            //se periptosi poy isxuei to panv function if tote sundeo ton xrhsth me thn bash
            Auth::login($user, true);
        }
    }
}
```

Εικόνα 24. Εικόνα διαδικασίας ελέγχου εγκυρότητα OTP αριθμού (1).

Ωστόσο στην περίπτωση που δεν εγκριθεί η εγκυρότητα του OTP τότε να μην προχωρήσει σε πραγματοποίηση εισόδου αλλά να θεωρηθεί μη αποτελεσματική η αυθεντικοποίηση του. Το ίδιο θα συμβεί και στην περίπτωση που εγκριθεί η εγκυρότητα του OTP αριθμού αλλά θα είναι εκτός των χρονικών πλαισίων, τότε εξίσου θα θεωρηθεί αναποτελεσματική η αυθεντικοποίηση του χρήστη εμφανίζοντας μήνυμα της αίτιας ακύρωσης της διαδικασίας όπως φαίνεται στις παρακάτω εικόνες.

```

        return response('You are logged in');

//ean den einai egkyro to function if poy elegco gia thn egkirotita tou xronou
    } else{

//epistrefo se morfi JSON to success false, kai to monima oti leiwei o xronos
        return response()->json(array(
            'success' => false,
            'message' => 'The time expired'
        ));

    }

//ean den einai egkuro ro function if gia thn egkirotita ton credentials tou xrhsth
    } else{

//epistrefo se morfi JSON to success wrong kai to katothi minima
        return response()->json(array(
            'success' => 'wrong',
            'message' => 'OTP invalid'
        ));
    }

```

Εικόνα 25. Εικόνα διαδικασίας ελέγχου εγκυρότητα OTP αριθμού (2).

3.1.4 Διαδικασία ασφαλούς αποσύνδεσης του χρήστη

Τελευταία διαδικασία που λαμβάνει μέρος στην παροχή υπηρεσιών της εφαρμογής είναι λειτουργία εξόδου του χρήστη από το προφίλ. Με την ολοκλήρωση των προηγούμενων ενεργειών δηλαδή εισόδου, εγγραφής και ελέγχου εγκυρότητας του αριθμού OTP ακολουθεί η διαδικασία ασφαλούς εξόδου από το προφίλ του χρήστη. Σε αυτό το στάδιο, μέσω functions, ελέγχεται εάν ο χρήστης είναι ήδη αυθεντικοποιημένος και διατηρεί μια ενεργή κατάσταση στο προφίλ του. Σε αυτήν την περίπτωση για την επιτυχή έξοδο και την εξασφάλιση της ιδιωτικότητας του, πραγματοποιείται ανάκληση των tokens που έχουν δημιουργηθεί κατά την εγγραφή του στην βάση δεδομένων. Μετά το πέρας της διαδικασίας αποστέλλεται μέσω response, μήνυμα επιτυχής εξόδου όπως φαίνεται στην παρακάτω εικόνα.

```

public function logout(Request $request)
{
    //function if gia to an o xristis einai authentikopoihmenos
    if (Auth::user()){
        $user = Auth::user()->token();
        $user->revoke();

        return response()->json([
            'success' => true,
            'message' => 'Logout successfully'
        ]);
    }else{
        return response()->json([
            'success' => false,
            'message' => 'Unable to logout'
        ]);
    }
}

```

Εικόνα 26. Εικόνα διαδικασίας ασφαλούς απενεργοποίησης χρήστη.

3.1.5 Λόγοι χρήστης RESTful API και παραμετροποίηση του Laravel framework

Οι παραπάνω διαδικασίες παρέχονται στην εφαρμογή για να προσομοιάσουν την λειτουργία μια εφαρμογής τραπέζης. Αξιοσημείωτο είναι να αναφερθεί και ο τρόπος σύνδεσης των παραπάνω functions με το RESTful API. Αυτό που χαρακτηρίζει τον παραπάνω κώδικα σε RESTful API είναι η μέθοδος με την οποία μεταφέρονται δεδομένα. Στην συγκεκριμένη περίπτωση, το Laravel παρέχει υπηρεσίες συνδεσιμότητάς τα λεγόμενα routes. Υπάρχουν δυο ειδών, εκείνα που χρησιμοποιούνται για την αποστολή δεδομένων σε web περιβάλλον και εκείνα που χρησιμοποιούνται για εξωτερικές εφαρμογές. Στην προκειμένη περίπτωση χρησιμοποιούνται τα routes για εξωτερικές εφαρμογές λόγω της ανάπτυξης της εφαρμογής. Στην παρακάτω εικόνα διακρίνεται η διαδικασία γεφύρωσης του RESTful API με την διεπαφή του χρήστη. Εκτενέστερα, φαίνεται οι μέθοδοι που χρησιμοποιούνται για την μεταφορά των δεδομένων. Επειδή γίνεται χρήση RESTful API, οι χρήση των PUT, GET, POST, DELETE, PATCH είναι απαραίτητη για την ανταλλαγή των δεδομένων. Στην προκειμένη περίπτωση γίνεται χρήση των POST στην μέθοδο login και register και confirmOTP καθώς είναι υπεύθυνα για την αποθήκευση δεδομένων στην βάση δεδομένων, όπως τα στοιχεία πρόσβασης αλλά και ο αριθμός OTP και ο χρόνος λήξης του. έπειτα γίνεται χρήση της GET στην μέθοδο logout και userdetails καθώς είναι υπεύθυνο για τον έλεγχο των στοιχείων εσωτερικά της βάσεως δεδομένων. Εκτός των http μεθόδων που χρησιμοποιούνται για το RESTful API, γίνεται και χρήση των παροχών middleware. Η ορολογία του ευθύνεται στην ενδιάμεση στάση που

διατηρεί μεταξύ του επιπέδου επικοινωνίας του http πρωτοκόλλου. Συγκεκριμένα middleware είναι ένας μηχανισμός ο οποίος χρησιμοποιείται με σκοπό να φιλτράρει τα requests και responses που μεταφέρονται. Το framework Laravel παρέχει default middleware αυθεντικοποίησης για ανακατεύθυνση του χρήστη σε σωστό route. Ωστόσο για τις απαιτήσεις της εφαρμογής δημιουργήθηκαν ξεχωριστά αυτοσχέδια middleware's τα οποία παρέχουν αδειές CORS-HEADERS τα οποία επιτρέπουν την χρήση των μεθόδων http από εξωτερικές σύσκες. Μέσω των headers επιτρέπονται, ανάλογα με τους περιορισμούς το κανένα header ξεχωριστά, η μεταφορά των δεδομένων. Οι παραπάνω διαδικασίες παρουσιάζονται στις κάτωθι εικόνες.

```
Route::group(['middleware' => ['cors', 'json.response']], function(){
    Route::post('login', [AuthController::class, 'login']);
    Route::post('register', [AuthController::class, 'register']);
});

Route::middleware('auth:api')->group(function (){
    Route::get('logout', [AuthController::class, 'logout']);
});

Route::middleware('TwoFactor')->group(function (){
    Route::get('userdetails', [AuthController::class, 'userdetails']);
});

Route::group(['middleware' => ['cors', 'json.response']], function(){
    Route::post('confirm0tp', [AuthController::class, 'confirm0tp']);
});
```

Εικόνα 27. Εικόνα διαδικασίας σύνδεσης backend-frontend.

Για τις ανάγκες τις προσβασιμότητας, προστέθηκαν headers τύπου X-Requested-With, Content-Type, X-Auth-Token και Authorization. Το πρώτο header χρησιμοποιείται για να αποτρέψει επιθέσεις τύπου CSRF καθώς δεν επιτρέπει τα request να προστεθούν μέσα στο AJAX request cross domain. Το δεύτερο χρησιμοποιείται για να υποδηλώσει το τύπο δεδομένων που μεταφέρονται στα requests και responses. Τέλος το τρίτο και το τέταρτο χρησιμοποιούνται για να αυθεντικοποιήσουν το request το οποίο δεν περιέχει κάποιο ασφαλές cookie.

```

public function handle(Request $request, Closure $next)
{
    return $next($request)
        ->header('Access-Control-Allow-Origin', '*')
        ->header('Access-Control-Allow-Methods', 'GET, POST, PUT, DELETE, OPTIONS')
        ->header('Access-Control-Allow-Headers', 'X-Requested-With, Content-Type, X-Token-Auth, Authorizati
}

```

Εικόνα 28. Διαδικασία οριοθέτησης headers.

Η αρχικοποίηση των ρυθμίσεων του Laravel framework περιλαμβάνουν την δημιουργία βάσης δεδομένων, την σύνδεση της βάσης με τον server του τοπικού δίκτυο καθώς και χρήσης συγκεκριμένου port. Αναμφίβολα για τον συγχρονισμό των υπηρεσιών που παρέχει το Laravel και για την επιτυχή χρήση όλων των function που προαναφέρθηκαν απαραίτητη ήταν η επιπλέον αρχικοποίηση των ρυθμίσεων όπως είναι το time zone, η αρχικοποίηση των middleware's που χρησιμοποιήθηκαν, η οριστικοποίηση χρήσης των headers που έλαβαν χωρά και τέλος η δημιουργία custom κανόνων validation για τις ανάγκες τις εφαρμογής όπως φαίνεται και στις παρακάτω εικόνες αναλυτικότερα.

```

public function handle(Request $request, Closure $next)
{
    $request->headers->set('Accept', 'application/json');
    return $next($request);
}

```

Εικόνα 29. Όλα τα request περνάνε από header που επιτρέπει μεταφορά json δεδομένων.

```

APP_ENV=local
APP_KEY=base64:NGbh8gFY8tqHMQ9kSH+Vi8UyBrZRYX3/729AE29qLEk=
APP_DEBUG=true
APP_URL=http://192.168.42.23:8000

LOG_CHANNEL=stack
LOG_LEVEL=debug

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=rest_api
DB_USERNAME=root
DB_PASSWORD=

BROADCAST_DRIVER=log
CACHE_DRIVER=file
QUEUE_CONNECTION=sync
SESSION_DRIVER=file
SESSION_LIFETIME=120

```

Εικόνα 30. Διαδικασία εγκαθίδρυσης URL του API.

Αξιοσημείωτο παράδειγμα είναι η δημιουργία και αρχικοποίηση των λειτουργιών του περιβάλλοντος του Laravel με σκοπό να λειτουργήσει για προσωπικό συμφέρον.

```

protected $routeMiddleware = [
    'twofactor' => \App\Http\Middleware\TwoFactor::class,
    'cors' => \App\Http\Middleware\Cors::class,
    'json.response' => \App\Http\Middleware\ForceJsonResponse::class,
    'auth' => \App\Http\Middleware\Authenticate::class,
    'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,
    'cache.headers' => \Illuminate\Http\Middleware\SetCacheHeaders::class,
    'can' => \Illuminate\Auth\Middleware\Authorize::class,
    'guest' => \App\Http\Middleware\RedirectIfAuthenticated::class,
    'password.confirm' => \Illuminate\Auth\Middleware\RequirePassword::class,
    'signed' => \Illuminate\Routing\Middleware\ValidateSignature::class,
    'throttle' => \Illuminate\Routing\Middleware\ThrottleRequests::class,
    'verified' => \Illuminate\Auth\Middleware\EnsureEmailIsVerified::class,
];

```

Εικόνα 31. Διαδικασία αρχικοποίησης των Middleware's που χρησιμοποιούνται.

3.2 USER INTERFACE– FRONTEND

3.2.1 Βασικές παραμετροποιήσεις και διαφορά προτίμησης μεταξύ των volley και retrofit βιβλιοθηκών

Το frontend κομμάτι της εφαρμογής αποτελεί κώδικα βασιζόμενο στο android studio. Για τις απαιτήσεις της εφαρμογής χρειάστηκαν οι εγκαταστάσεις των απαραίτητων βιβλιοθηκών που βοήθησαν είτε στην ευελιξία της, λόγω των υπηρεσιών που προσφέρει είτε στην σύνδεση της με το RESTful API που αναφέρθηκε παραπάνω. Αξίζει να σημειωθεί ότι για να μπορέσει να γίνει επιτυχής σύνδεση της εφαρμογής, χρειάστηκε η κινητή συσκευή να μετατραπεί σε κινητό δίκτυο μέσω usb καθώς ο server που χρησιμοποιεί το Laravel framework είναι απαραίτητο να συνδεθεί σε αυτό το δίκτυο και όχι στο προσωπικό του localhost που προσφέρει. Για την επιτυχή γεφύρωση της εφαρμογής με το Restful API χρησιμοποιήθηκε η βιβλιοθήκη volley library. Η σύγκρισή έγινε μεταξύ αυτής και της retrofit. Η διαφορά που λήφθηκε υπόψιν για την χρήση της είναι ότι η volley παρέχει αποδοτικότερους μηχανισμούς HTTP client σε αντίθεση με την βιβλιοθήκη retrofit και επίσης η χρήση των εντολών, βάση προσωπικών προτιμήσεων, είναι ευκολονόητη. Χρειάζεται να γίνει κατανοητό ότι το στάδιο του android studio αναφέρεται στην επικοινωνίας που γίνεται μεταξύ του χρήστη και του API, καθώς η επικοινωνίας μεταξύ του API και του server-database έχει πραγματοποιηθεί στο προηγούμενο στάδιο δηλαδή στο backend.

3.2.2 Σύνδεση της λειτουργίας εγγραφής στο frontend με το backend

Η πρώτη διαδικασία που λαμβάνει χώρα είναι αυτής της εγγραφής του χρήστη. Για την αποτελεσματική εγγραφή του, χρειάστηκε η αρχικοποίηση μεταβλητών οι οποίες θα επουλώνουν τις ενέργειες του χρήστη για την ανάλογη αποστολή δεδομένων. Επίσης απαραίτητη είναι και η αρχικοποίηση του URL που θα συνδέεται με το RESTful API για να μεταφέρει τα δεδομένα που στέλνει ο χρήστης στο API. Κατά την διαδικασία της εγγραφής, στο API είχε αναφερθεί ότι θα λαμβάνεται ο κωδικός DeviceID από την συσκευή. Έτσι για να γίνει αυτό χρειάστηκε η δημιουργία ζήτησης παροχής αδειών της συσκευής που λαμβάνει αυτόν τον κωδικό όπως φαίνεται στην παρακάτω εικόνα.

```

public class RegisterActivity extends AppCompatActivity {

    private EditText Register_with_username;
    private EditText Register_with_password;
    private EditText Register_with_phoneNumber;
    private EditText Register_with_DeviceSerialNumber;
    private TextView textView_Register;
    private Button btnRegister;

    private static final String URL_REGISTER = "http://192.168.42.23:8000/api/register";

    String [] permissions={"android.permission.READ_PHONE_STATE"};

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);

        getSupportActionBar().hide();

        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN, WindowManager.LayoutParams.FLAG_FULLSCREEN);
    }
}

```

Εικόνα 32. Διαδικασία πραγματοποίησης register σύνδεσης στο frontend (1).

Με την αποδοχή της αδείας παροχής του κωδικού, η εφαρμογή προχωρά στην διαδικασία κατά την οποία πραγματοποιείται η εγγραφή του χρήστη και η αποθήκευση των στοιχείων στην βάση δεδομένων όπως φαίνεται στις παρακάτω εικόνες.

```

getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN, WindowManager.LayoutParams.FLAG_FULLSCREEN);

Register_with_username = findViewById(R.id.Register_with_username);
Register_with_password = findViewById(R.id.Register_with_password);
Register_with_phoneNumber = findViewById(R.id.Register_with_phoneNumber);
Register_with_DeviceSerialNumber = findViewById(R.id.Register_with_DeviceSerialNumber);
textView_Register = findViewById(R.id.textView_Register);
btnRegister = findViewById(R.id.btnRegister);

textView_Register.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent LoginIntent = new Intent( packageContext: RegisterActivity.this, LoginActivity.class);
        startActivity(LoginIntent);
    }
});

btnRegister.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.Q){
            requestPermissions(permissions, requestCode: 101);
        }
    }
});

```

Εικόνα 33. Διαδικασία πραγματοποίησης register σύνδεσης στο frontend (2).

```

private void register(){

    final String Register_with_username = this.Register_with_username.getText().toString().trim();
    final String Register_with_password = this.Register_with_password.getText().toString().trim();

    final String Register_with_phoneNumber = this.Register_with_phoneNumber.getText().toString().trim();
    final String Register_with_DeviceSerialNumber = this.Register_with_DeviceSerialNumber.getText().toString().trim();

    StringRequest stringRequest = new StringRequest(Request.Method.POST, URL_REGISTER,
        new Response.Listener<String>() {
            @Override
            public void onResponse(String response) {
                try {
                    JSONObject jsonObject = new JSONObject(response);
                    String success = jsonObject.getString( name: "success");

                    if (success.equals("true")){
                        JSONObject jsonObject1 = new JSONObject(response);
                        String registerSuccessMessage = jsonObject1.getString( name: "registerSuccessMessage").trim();

                        Toast.makeText( context: RegisterActivity.this, registerSuccessMessage, Toast.LENGTH_SHORT).show();
                        Intent mainScreenIntent = new Intent( packageContext: RegisterActivity.this, OnboardActivity.class);
                        startActivity(mainScreenIntent);
                    }
                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
        }
    );
}

```

Εικόνα 34. Διαδικασία σύνδεσης volley library με backend.

Κατά την διαδικασία της εγγραφής λοιπόν, καλείται η βιβλιοθήκη volley για την επιτυχή γεφύρωση της εφαρμογής με το API. Στο στάδιο αυτό λαμβάνουν χώρα τα μηνύματα και τα δεδομένα σε μορφή JSON που στέλνονται ως απάντηση στο request, από το API και εκτενέστερα από την βάση δεδομένων. Για τον έλεγχο της ακεραιότητας δομής των δεδομένων χρησιμοποιούνται οι κωδικοί http οι οποίοι υποδηλώνουν την κατάσταση του request και response που μεταφέρονται. Στην προκείμενη περίπτωση ελέγχεται ο κωδικός 422 ο οποίος υποδηλώνει τον λόγο σφάλματος των δεδομένων που στέλνονται από την βάση όπως φαίνεται στις παρακάτω εικόνες αναλυτικότερα.

```

public void onErrorResponse(VolleyError error) {
    String DataResponseRegister = "";
    String statusCode = String.valueOf(error.networkResponse.statusCode);
    String phonenumber = "", username = "", password = "";
    if(error.networkResponse.data!=null) {
        try {
            DataResponseRegister = new String(error.networkResponse.data, charsetName: "UTF-8");
        } catch (UnsupportedEncodingException e) {
            e.printStackTrace();
        }
    }
    if (statusCode.equals("422")){
        try {
            JSONObject obj = new JSONObject(DataResponseRegister);
            String errors = obj.getString( name: "errors");

            JSONObject obj2 = new JSONObject(errors);
            phonenumber = obj2.getString( name: "phonenumber");

        } catch (JSONException e) {
            e.printStackTrace();
        }
    }
}

```

Εικόνα 35. Διαδικασία λήψης μεταβλητών από το backend με volley library.

```

@Nullable
@org.jetbrains.annotations.Nullable
@Override
protected Map<String, String> getParams() throws AuthFailureError {
    Map<String, String> params = new HashMap<>();
    params.put("username", Register_with_username);
    params.put("password", Register_with_password);
    params.put("phonenumber", Register_with_phoneNumber);
    params.put("device_id", Register_with_DeviceSerialNumber);

    return params;
}
};
RequestQueue requestQueue = Volley.newRequestQueue(context, this);
requestQueue.add(stringRequest);
}

```

Εικόνα 36. Διαδικασία αρχικοποίησης πεδίων που λαμβάνουν εντολές για το backend.

Τέλος σημαντική είναι και δέσμευση των μεταβλητών που είναι υπεύθυνες για την διαδικασία αυτήν δηλαδή τα πεδία τα οποία χρησιμοποιεί ο χρήστης για να λάβει τις ανάλογες απαντήσεις από το API όπως διακρίνεται παρακάτω.

3.2.3 Σύνδεση της λειτουργίας εισόδου του frontend με το backend

Επόμενη σειρά παίρνει η δεύτερη διαδικασία που αναφέρεται στην είσοδο του χρήστη με τα στοιχεία πρόσβασης του. Συγκεκριμένα, για την εξασφάλιση αποτελεσματικής διαδικασίας εισόδου χρειάστηκε η αρχικοποίηση των μεταβλητών που θα αλληλοεπιδρούν με τις ενέργειες του χρήστη κατά της εισαγωγή των στοιχείων. Επίσης απαραίτητη είναι η δήλωση του URL που θα χρησιμοποιηθεί για να γεφύρωση της μεταφοράς των δεδομένων μεταξύ του χρήστη και του API. Σε αυτό το σημείο απαραίτητη είναι και η αρχικοποίηση της μεταβλητής για την διεξαγωγή αποστολής του OTP αριθμού κατά την είσοδο στο λογαριασμό του χρήστη. Απαραίτητη λοιπό είναι η δημιουργία καναλιού επικοινωνίας για android 10 συσκευές όπως φαίνεται στις παρακάτω εικόνες.

```

//dhlvnv tiw metablites poy exv xrhsimopoihsei sta xml activities kai lambanoun xora edo
private NotificationManagerCompat notificationManager;
private EditText sign_in_with_username;
private EditText sign_in_with_password;
private EditText sign_in_with_phoneNumber;
private TextView textView_Login;
private Button btnLogin;
private String DeviceID;

//dhlonw to permission poy exo prosthesei sto manifest
String [] permissions={"android.permission.READ_PHONE_STATE"};

//arxikopoio to url pou xrhsimopieitai gia thn syndesh toy RESTful API me to android studio gia to login
private static final String URL_LOGIN = "http://192.168.42.23:8080/api/login";

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_login);

    //krybo to action bar kai to notification bar apo thn othoni tou kintoy
    getSupportActionBar().hide();
    getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN, WindowManager.LayoutParams.FLAG_FULLSCREEN);

```

Εικόνα 37. Διαδικασία πραγματοποίησης login σύνδεσης στο frontend

Έχοντας διασταυρωθεί τα στοιχεία του χρήστη επιτυχώς με αυτά που υπάρχουν στην βάση, τότε το σύστημα ελέγχει ξεχωριστά και την εγκυρότητα του DeviceID. Σε περίπτωση που είναι λανθασμένο φυσικά και δεν μπορεί να προχωρήσει η διαδικασία με το ενδεχόμενο ότι είναι εγκατεστημένη σε άλλη συσκευή και εισέρχεται στο σύστημα με ιδία στοιχεία σύνδεσης. Έχοντας λάβει χωρά επιτυχώς η διαδικασία αυθεντικοποίησης του χρήστη τότε πρωτεύοντα ρολό έχει η πραγματοποίηση της εισόδου και η αποστολή του OTP αριθμού για την επίτευξη της της παροχής ασφάλειας TFA όπως φαίνεται κάτωθι.

```

if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.Q) {
    NotificationChannel channel = new NotificationChannel( id: "CHANNEL_1_ID", name: "My Notification", NotificationManager.IMPORTANCE_HIGH);
    NotificationManager manager = getSystemService(NotificationManager.class);
    manager.createNotificationChannel(channel);
}

notificationManager = NotificationManagerCompat.from(this);

//orizv tiw metablhtes poy exo dilosei parapano na kavnoy interact me autes tou xml
sign_in_with_username = findViewById(R.id.sign_in_with_username);
sign_in_with_password = findViewById(R.id.sign_in_with_password);
sign_in_with_phoneNumber = findViewById(R.id.sign_in_with_phoneNumber);
textView_Login = findViewById(R.id.textView_Login);
btnLogin = findViewById(R.id.btnLogin);

//dhlonw thn leitourgia apo to button textView_Login
textView_Login.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        //se periptosi poy paththei to button tote apo to Login page pame sto Register Page
        Intent RegisterIntent = new Intent( packageContext: LoginActivity.this, RegisterActivity.class);
        startActivity(RegisterIntent);
    }
});

```

Εικόνα 38. Διαδικασία δημιουργίας notifications.

```

@SuppressLint("HardwareIds")
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);

    //ginetai lipsi to DeviceID apo thn sykeyh kathos stelbetai kai permission
    DeviceID = Settings.Secure.getString(getContentResolver(), Settings.Secure.ANDROID_ID);

    switch (requestCode)
    {
        case 101:
            if (grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                String mUsername = sign_in_with_username.getText().toString().trim();
                String mPassword = sign_in_with_password.getText().toString().trim();
                String mPhoneNumber = sign_in_with_phoneNumber.getText().toString().trim();

                if (!mUsername.isEmpty() || !mPassword.isEmpty() || mPhoneNumber.isEmpty()) {

```

Εικόνα 39. Διαδικασία απόκτησης DeviceID.

3.2.4 Διαδικασία αποστολής του OTP αριθμού

Σε αυτό το στάδιο βασικό ρολό διαδραματίζει η βιβλιοθήκη volley η οποία μεταφέρει και στέλνει μέσω post request τα δεδομένα στο API και λαμβάνει μέσω response τα δεδομένα από την βάση σε περίπτωση όμως που είναι ορθή η διαδικασία. Έπειτα μέσω του notification που έχει αρχικοποιηθεί προηγουμένως τότε αποστέλνεται ο εξαψήφιος αριθμός OTP που δημιουργείται στο RESTful API και στέλνεται σε νέα σελίδα είσοδοί του OTP όπως φαίνεται και στις παρακάτω φωτογραφίες.

```

//ksekinai h diadikasia login me to username, password, phonenumber
private void Login(String sign_in_with_username, String sign_in_with_password, String sign_in_with_phoneNumber) {

    //ginetai sindesi toy login me to rest api meso tou volley library me methodo post
    StringRequest stringRequest = new StringRequest(Request.Method.POST, URL_LOGIN,
        new Response.Listener<String>() {
            @RequiresApi(api = Build.VERSION_CODES.O)
            @Override

            //diadikasia poy lambanei xora sto response
            public void onResponse(String response) {
                try {
                    JSONObject jsonObject = new JSONObject(response);
                    String success = jsonObject.getString( name: "success");

                    if (success.equals("true")) {

                        JSONObject jsonObject1 = new JSONObject(response);

                        //sto response apothikevontai ta dedomena otp kai otp_expires_at
                        String otp = jsonObject1.getString( name: "otp").trim();
                        String otp_expires_at = jsonObject1.getString( name: "otp_expires_at").trim();

                        //axizei h diadikasia tou notification
                        NotificationCompat.Builder builder = new NotificationCompat.Builder( context: LoginActivity.this, channelId: "ΓΥΜΝΑΣΙΟ 1 ΤΗΛ").

```

Εικόνα 40. Διαδικασία σύνδεσης login με volley library και backend (1).

Σε αυτό το σημείο διεξάγεται προσπάθεια σύνδεσης των στο λογαριασμό του χρήστη με την βοήθεια των προσωπικών στοιχείων υετού, πραγματοποιώντας έλεγχο των στοιχείων που εισάγονται με αυτών που βρίσκονται ήδη αποθηκευμένα στην βάση δεδομένων κατά την διαδικασία της εγγραφής του.

```

        JSONObject jsonObject2 = new JSONObject(response);
        String error = jsonObject2.getString( name: "error");

        Toast.makeText( context: LoginActivity.this, error, Toast.LENGTH_SHORT).show();
    }
} catch (JSONException e) {
    e.printStackTrace();
    Toast.makeText( context: LoginActivity.this, text: "Error " +e.toString(), Toast.LENGTH_SHORT).show();
}
},
new Response.ErrorListener() {
    @Override
    public void onErrorResponse(VolleyError error) {
        String DataResponseLogin = "";
        String statusCode = String.valueOf(error.networkResponse.statusCode);
        String phonenumber = "", username = "", password = "";
        if (error.networkResponse.data != null) {
            try {
                DataResponseLogin = new String(error.networkResponse.data, charsetName: "UTF-8");
            } catch (UnsupportedEncodingException e) {
                e.printStackTrace();
            }
        }
        if (statusCode.equals("422")) {

```

Εικόνα 41. Διαδικασία σύνδεσης login με volley library και backend (2).

```

        @Nullable
        @org.jetbrains.annotations.Nullable
        @Override
        protected Map<String, String> getParams() throws AuthFailureError {
            Map<String, String> params = new HashMap<>();
            params.put("username", sign_in_with_username);
            params.put("password", sign_in_with_password);
            params.put("phonenumber", sign_in_with_phoneNumber);
            params.put("device_id", DeviceID);
            return params;
        }
    };

    RequestQueue requestQueue = Volley.newRequestQueue(context, this);
    requestQueue.add(stringRequest);
}

```

Εικόνα 42. Διαδικασία αρχικοποίησης πεδίων που λαμβάνουν εντολές για το backend.

Τέλος σημαντική είναι και δέσμευση των μεταβλητών που είναι υπεύθυνες για την διαδικασία αυτήν δηλαδή τα πεδία τα οποία χρησιμοποιεί ο χρήστης για να λάβει τις ανάλογες απαντήσεις από το API όπως διακρίνεται παρακάτω.

3.2.5 Σύνδεση της λειτουργίας ελέγχου εγκυρότητας του OTP στο frontend με το backend

Τρίτο στάδιο και και τελευταίο όσο αφορά την λειτουργικότητα της ανάπτυξης της εφαρμογής αποτελεί η επιβεβαίωση του OTP αριθμού. Παρομοίως και σε αυτήν την διαδικασία απαραίτητη είναι η αρχικοποίηση του URL που χρησιμοποιείται για να συνδέσει τα δεδομένα που στέλνει ο χρήστης μέσω της εφαρμογής στο API. Έπειτα λαμβάνει μέρος η αρχικοποίηση των μεταβλητών για την εισαγωγή των ψηφίων των αριθμών. Μέσω αυτού του σταδίου ελέγχεται με την βοήθεια της βιβλιοθήκης volley η εγκυρότητα του αριθμού OTP που υπάρχει αποθηκευμένος στην βάση με το αυτήν του αριθμού που συμπληρώνει ο χρήστης. Με την μέθοδο post και τα functions case που λαμβάνονται υπόψιν ελέγχονται τα μηνύματα responses που αποστέλλονται και ολοκληρώνεται η διαδικασία αναλόγως το αποτέλεσμα που έχει λάβει χωρά όπως φαίνεται στις παρακάτω εικόνες.

```

private EditText inputCode1;
private EditText inputCode2;
private EditText inputCode3;
private EditText inputCode4;
private EditText inputCode5;
private EditText inputCode6;
private Button buttonVerify;

private TextView otp_countDownTimer;

private static final String URL_ConfirmOtp = "http://192.168.42.23:8000/api/confirmOtp";

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_otp);

    getSupportActionBar().hide();

    getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN, WindowManager.LayoutParams.FLAG_FULLSCREEN);

```

Εικόνα 43. Διαδικασία δημιουργίας σύνδεσης για εγκυρότητα otp.

```

buttonVerify.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {

        String mCode1 = inputCode1.getText().toString().trim();
        String mCode2 = inputCode2.getText().toString().trim();
        String mCode3 = inputCode3.getText().toString().trim();
        String mCode4 = inputCode4.getText().toString().trim();
        String mCode5 = inputCode5.getText().toString().trim();
        String mCode6 = inputCode6.getText().toString().trim();

        String mOtp = mCode1 + mCode2 + mCode3 + mCode4 + mCode5 + mCode6;

        if (inputCode1.getText().toString().trim().isEmpty()
            || inputCode2.getText().toString().trim().isEmpty()
            || inputCode3.getText().toString().trim().isEmpty()
            || inputCode4.getText().toString().trim().isEmpty()
            || inputCode5.getText().toString().trim().isEmpty()
            || inputCode6.getText().toString().trim().isEmpty()) {
            Toast.makeText(context: OtpActivity.this, text: "Please enter valid code", Toast.LENGTH_SHORT).show();
        } else {

```

Εικόνα 44. Διαδικασία δημιουργίας πεδίων ψηφίων.

```

private void SetupOtpInputs(){
    inputCode1.addTextChangedListener(new TextWatcher() {
        @Override
        public void beforeTextChanged(CharSequence s, int start, int count, int after) {

        }

        @Override
        public void onTextChanged(CharSequence s, int start, int before, int count) {
            if (!s.toString().trim().isEmpty()){
                inputCode2.requestFocus();
            }
        }

        @Override
        public void afterTextChanged(Editable s) {

        }
    });
    inputCode2.addTextChangedListener(new TextWatcher() {
        @Override
        public void beforeTextChanged(CharSequence s, int start, int count, int after) {

```

Εικόνα 45. Διαδικασία δημιουργίας αλλαγής πεδίου ψηφιού.

Σε αυτό το σημείο, μετατοπίζεται η εικοστός του ψηφιού στην περίπτωση που ολοκληρωθεί η διαδικασία εισαγωγής του από τον χρήστη.

```

private void Verify(String mOtp) {
    StringRequest stringRequest = new StringRequest(Request.Method.POST, URL_ConfirmOtp,
        new Response.Listener<String>() {
            @Override
            public void onResponse(String response) {
                try {

                    JSONObject jsonObject = new JSONObject(response);
                    String success = jsonObject.getString( "name": "success");

                    switch (success) {
                        case "true":

                            JSONObject jsonObject1 = new JSONObject(response);
                            String dataUser = jsonObject1.getString( "name": "data").trim();

                            Toast.makeText( context: OtpActivity.this, text: "Success OTP. \nYour data :"+ dataUser, Toast.LENGTH_SHORT).show();

                            break;
                        case "false":

                            Toast.makeText( context: OtpActivity.this, text: "OutTimeLimit", Toast.LENGTH_SHORT).show();

                            break;
                        case "wrong":

```

Εικόνα 46. Διαδικασία σύνδεσης εγκυρότητας otp με volley library στο backend.

```

        @Nullable
        @org.jetbrains.annotations.Nullable
        @Override
        protected Map<String, String> getParams() throws AuthFailureError {
            Map<String, String> params = new HashMap<>();
            params.put("otp", mOtp);
            return params;
        }
    };
    RequestQueue requestQueue = Volley.newRequestQueue(context, this);
    requestQueue.add(stringRequest);
}

```

Εικόνα 47. Διαδικασία αρχικοποίησης πεδίων που λαμβάνουν εντολές για το backend.

3.2.6 Εμπλουτισμός της λειτουργίας της διεπαφής του χρήστη

Έχοντας αναλυθεί και τοποθετηθεί πλήρως η λειτουργικότητα της εφαρμογής, απαραίτητη είναι και η επισήμανση των μεθόδων που αναπτυχθήκαν με σκοπό να δημιουργήσουν στους χρήστες μια αίσθηση φιλικότητας και ευχρηστίας στον τρόπο χρήσης της εφαρμογής. Για τον λόγο αυτό η λειτουργικότητα της εφαρμογής χαρακτηρίζεται από ομοιόμορφες διεπαφές χρήστη, λειτουργίες για οθόνη αφής, σελίδες της εφαρμογής που χρησιμοποιούνται για εισαγωγή στο κυρίως περιεχόμενο της εφαρμογής και διαφορά αλλά τα οποία αποτυπώνονται στις εικόνες που ακολουθούν.

```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/custom3"
    tools:context=".MainActivity">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="270dp"
        android:src="@drawable/lp_secure">

    </ImageView>

    <ProgressBar
        android:id="@+id/progressBar"
        style="?android:attr/progressBarStyle"
        android:layout_width="89dp"
        android:layout_height="78dp"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"

```

Εικόνα 48. Διαδικασία δημιουργίας UI για Main Activity.

```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/custom3"
    android:gravity="center"
    android:orientation="vertical"
    tools:context=".RegisterActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="191dp"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="110dp"
        android:layout_marginLeft="110dp"

```

Εικόνα 49. Διαδικασία δημιουργίας UI για Register screen.

```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/custom3"
    tools:context=".OtpActivity">

    <ImageView
        android:id="@+id/imageView3"
        android:layout_width="match_parent"
        android:layout_height="139dp"
        android:layout_alignParentBottom="true"
        android:layout_marginBottom="589dp"
        app:srcCompat="@drawable/notificationlogo" />

    <Button
        android:id="@+id/buttonVerify"
        android:layout_width="77dp"
        android:layout_height="63dp"
        android:layout_alignParentEnd="true"

```

Εικόνα 50. Διαδικασία δημιουργίας UI για Otp screen.

```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/custom3"
    android:gravity="center"
    android:orientation="vertical"
    tools:context=".LoginActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="191dp"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="110dp"
        android:layout_marginLeft="110dp"

```

Εικόνα 51. Διαδικασία δημιουργίας UI για Login screen.

```

package com.example.test2app;

import ...

public class OnboardActivity extends AppCompatActivity {

    private OnBoardingAdapter onBoardingAdapter;
    private LinearLayout layoutOnBoardIndicators;
    private MaterialButton Button;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_onboard);

        getSupportActionBar().hide();

        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN, WindowManager.LayoutParams.FLAG_FULLSCREEN);

        layoutOnBoardIndicators = findViewById(R.id.layoutOnBoardIndicators);
        Button = findViewById(R.id.Button);

        setupOnBoardingAdapter();
    }
}

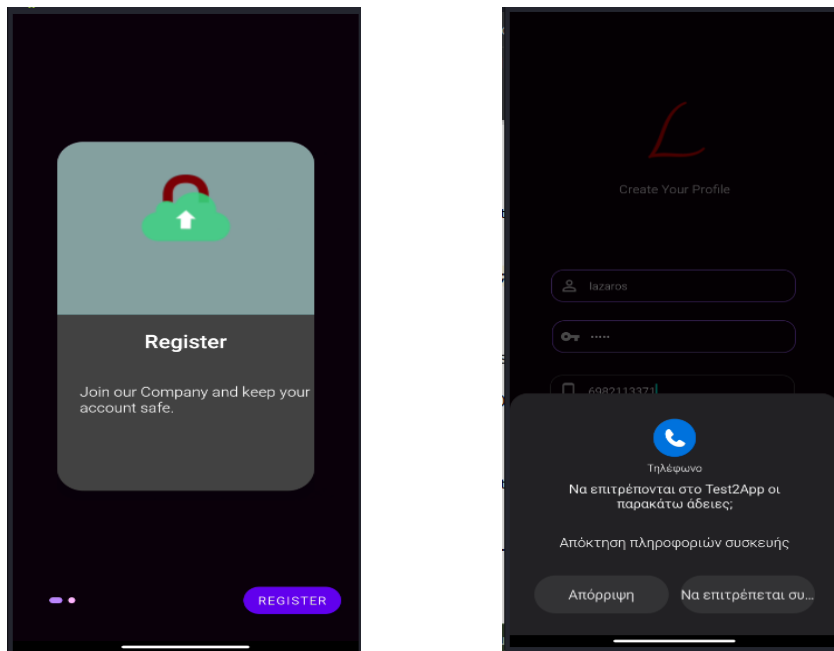
```

Εικόνα 52. Εξαφάνιση μπάρας ειδοποιήσεων.

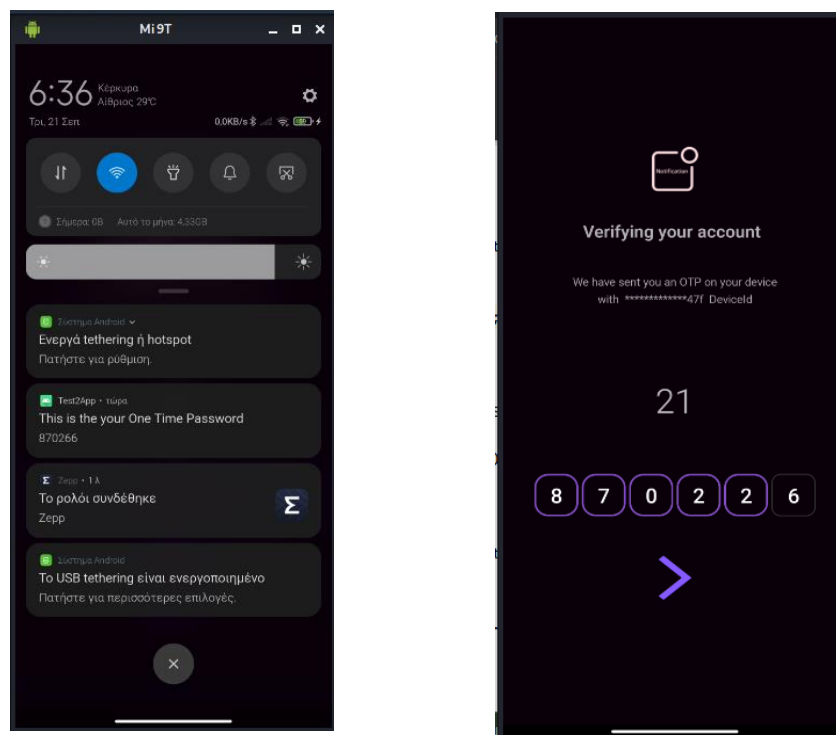
Αναφορικά με τα παραπάνω η επιλογή της εφαρμογής να αναπτυχθεί με frontend η διεπαφή του χρήστη και backend το RESTful API, έγινε ξεκάθαρα για λόγους ασφάλειας. Η χρήση της διεπαφής γίνεται μόνο για την αλληλεπίδραση του χρήστη με τα πεδία εφαρμογής των μεταβλητών και της σύνδεσης με το API, ενώ όλες οι άλλες διαδικασίες λαμβάνουν χωρά στο API.

3.3 Παρουσίαση εφαρμογής

Σε αυτό το κομμάτι παρουσιάζεται η εφαρμογή εν ώρα λειτουργίας της καθώς και όλες οι διαδικασίες που λαμβάνουν χωρά για τη επιτυχή αποστολή του Οtr αριθμού βάση του DeviceID. Μια μικρή παρουσίαση εικόνων είναι αρκετή για να υπάρξει πλήρης κατανόηση των διαδικασιών που πραγματοποιούνται με σκοπό να εξαλείφει κάθε είδους sim swap απάτης όπως φαίνεται στις παρακάτω εικόνες.



Εικόνα 53. Αρχική οθόνη και ζήτηση permissions.



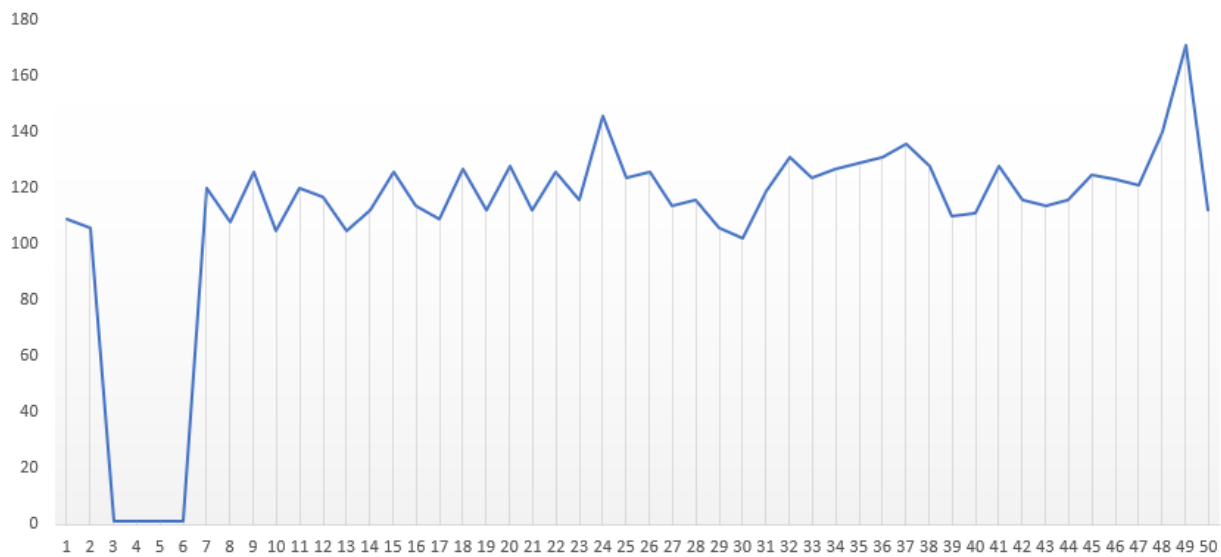
Εικόνα 54. Αποστολή Otp και εισαγωγή του.

ΚΕΦΑΛΑΙΟ 4

4.1 Αξιολόγηση εφαρμογής

Ένας βάσιμος χαρακτηρισμός που θα μπορούσε να αποδοθεί στον τρόπο εξαγωγής της εργασίας είναι σχεδίαση της επιστήμης. Η ορολογία αυτήν βάση του Wikipedia και του Gregor & Henver αποδίδεται ως μεθοδολογία εύρεσης βασιζομένη στην τεχνολογία του διαδικτύου η οποία προσφέρει guidelines, iteration και evaluation. Επίσης σύμφωνα με το Webster λεξικό η ορολογία σχεδίαση αποδίδεται για μια πράξη η οποία απαιτεί μελέτη και υλοποίηση του οτιδήποτε για συγκεκριμένη χρήση ή σκοπό. Οι πάνω ορολογίες συνήθως περιλαμβάνονται στο επίπεδο της πληροφορίας και της τεχνολογίας. Μια πολύ συγκεκριμένη προσέγγιση είναι αυτή της σχεδίας της επιστήμης η οποία αποτελεί ένα πρότυπο επίλυσης προβλημάτων με την χρήση μεθόδων, κατασκευών, μοντέλων και στιγμιότυπων. Η παρούσα εργασία εξαιτίας το δημιουργικού και υλοποιήσιμου ρολού που κατέχει, υιοθετεί το πρότυπο artifact instantiation σύμφωνα με τον Gregor & Jones. Ως αποτέλεσμα του παραπάνω η σχεδίαση της επιστήμης παρέχει το evaluation με τα κριτήρια effectiveness, efficiency και impact on the environment. Το καθένα από τα παραπάνω διαδραματίζει σημαντικό ρολό στην εξέλιξη και στην προτίμηση της εφαρμογής. Συγκεκριμένα για το efficiency δηλαδή την αποδοτικότητα, πραγματοποιήθηκαν αρκετά τεστ τα οποία εξετάζουν τον χρόνο κατά τον οποίο στέλνεται ένα request και ένα response με μια μέση τιμή από το api στον server και αντίστροφα. Η διαδικασία αυτήν πραγματοποιήθηκε με την βοήθεια του λογισμικού postman. Εξετάστηκαν οι περιπτώσεις που λαμβάνουν χωρά στην εφαρμογή δηλαδή η πραγματοποίηση εισόδου στην εφαρμογή, η διαδικασία εγγραφής καινούργιου χρήστη και τέλειος η διαδικασία ελέγχου εγκυρότητας του αριθμού OTP. Αναλυτικότερα πραγματοποιήθηκαν περίπου 300 προσπάθειες αποστολής request σε κάθε περίπτωση λαμβάνοντας υπόψιν κυρίως το χρόνο μετάδοσης των αιτημάτων σε milliseconds όπως φαίνεται στις παρακάτω εικόνες.

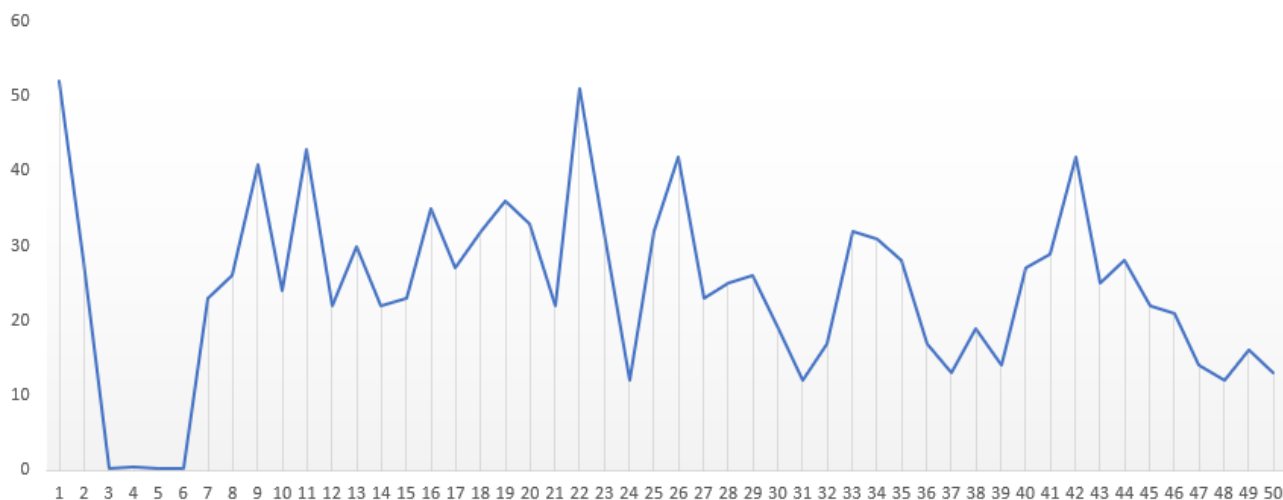
Γράφημα Http response για Login με άξονα τον χρόνο (ms).



Εικόνα 55. Αποτελέσματα POSTMAN για Login.

Σημειώνεται ότι ο μέσος ορός αποστολής αιτημάτων στην πραγματοποίηση της εισόδου του χρήστη είναι περίπου τα 131 MS με τον μεγαλύτερο χρόνο απόκρισης να αποτελείται από 158 MS και τον μικρότερο στα 98 MS. Παρόμοια αποτελέσματα αποτελεί και διαδικασία ελέγχου εγκυρότητας του αριθμού OTP. Ο μέσος ορός αποστολής αιτημάτων αποτελεί μέσο ορό 34 MS καθώς ο μεγαλύτερος χρόνος απόκρισης αποτελείται από 61 MS και τον ελάχιστο χρόνο σε 21 MS.

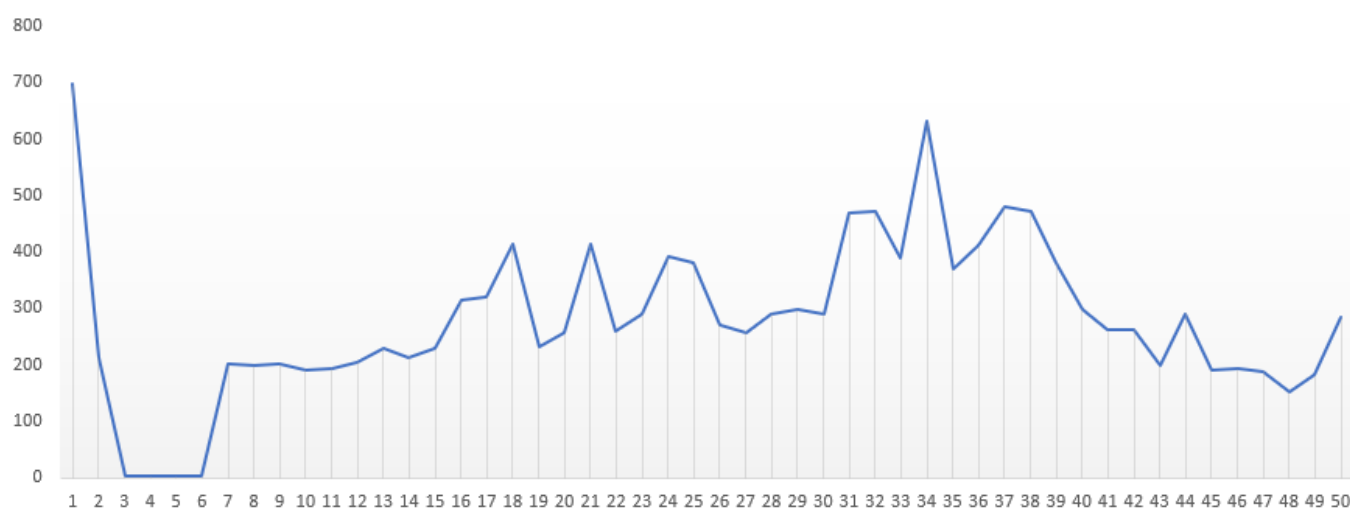
Γράφημα Http response για ConfirmOtp με άξονα τον χρόνο (ms).



Εικόνα 56. Αποτελέσματα POSTMAN για ConfirmOtp.

Τέλος, σειρά παίρνει η τελευταία διαδικασία δηλαδή η πραγματοποίηση εγγραφής του χρήστη στο σύστημα. Πραγματοποιήθηκαν, χάριν συντομίας, περίπου στις 30 συνεχόμενες εγγραφές στο σύστημα. Από αυτές ο μέσος χρόνος απόκρισης σε αιτήματα request και response ανέρχεται στα 229 MS καθώς το μεγαλύτερο χρονικά αίτημα αντιστοιχείται στα 691 MS ενώ το ελάχιστο χρονικά αίτημα αντιστοιχείται στα 31 MS όπως φαίνεται και στην παρακάτω εικόνα.

Γράφημα Http response για Register με άξονα τον χρόνο (ms).



Εικόνα 57. Αποτελέσματα POSTMAN για Register

Συνεχίζοντας σχετικά με το effectiveness δηλαδή την αποτελεσματικότητα χρειάζεται να σημειωθεί ότι είναι απαιτητή η διεξαγωγή αρκετών πειραμάτων για τον έλεγχο της λειτουργικότητας της εφαρμογής. Αξίζει να σημειωθεί όμως, σύμφωνα με των όσων αναφέρθηκαν, το TFA αποτελεί έναν από τους πιο ασφαλείς τρόπους εξασφάλισης της ιδιωτικότητας και διαφύλαξης των προσωπικών στοιχείων κατά την πραγματοποίηση διαδικτυακών συναλλαγών. Αυτό έχει ως αποτέλεσμα να χαρακτηρίζεται το ένα μέρος της εφαρμογής ως αρκετά λειτουργικό και πλήρως αναγνωρισμένο στον έξω κόσμο. Αναμφίβολα όμως χρειάζεται να εξεταστεί και η χρήση της εφαρμογής από πολλούς χρήστες. Τέλος σχετικά με το impact on the environment δηλαδή τις επιπτώσεις που έχει στο περιβάλλον η εφαρμογή χρειάζεται να σημειωθεί και τονιστεί ότι η εφαρμογή χρησιμοποιείται για να αποτρέψει επιθέσεις τύπου sim swapping. Η ασφάλεια που παρέχει στους χρήστες, εξαιτίας της αυθεντικοποίησης τους βάση του DeviceID, καθιστά αδύνατη την πιθανότητα να υποκλέψει τον αριθμό OTP και να υποβαθμίσει την ακεραιότητα του καθώς βασίζεται σε έναν αριθμό ο οποίος αποτελεί την ταυτότητα της συσκευής. Αναμφίβολα όμως και εδώ απαραίτητη είναι η χρήση της εφαρμογής για την διεξαγωγή του πορίσματος σχετικά με την ασφάλεια της.

4.2 Συζήτηση – μελλοντικές προεκτάσεις

Η διατριβή άρθρων, επιστημονικών δοκιμίων, ερευνών και διάφορων άλλων επιστημονικών πηγών εντείνει την ανάγκη επαγρύπνησης σχετικά με τους κινδύνους που κρύβει το διαδίκτυο καθώς και με την λιγοστή προσωπική ασφάλεια που υπάρχει στον κυβερνοχώρο. Η συχνή χρήση της τεχνολογίας και η ραγδαία εξέλιξη της κοινωνικής μηχανικής αυξάνει ακόμα περισσότερο την ανάγκη για ανάπτυξη εφαρμογών οι οποίες προσφέρουν την απαιτούμενη ασφάλεια. Αποτελεσματικότερο και ασφαλέστερο θα ήταν

φυσικά, η ύπαρξη αρκετών δικλίδων ασφαλείς, οι οποίες όμως δεν θα ξοδεύαν ούτε παραγωγικό κόστος ούτε χρόνο. Φυσικά αυτό είναι αναπόφευκτο έως ορισμένο βαθμό φυσικά, καθώς η οποιαδήποτε πρόοδος της τεχνολογίας επιφέρει μεγάλες αλλαγές στον οικονομικό τομέα της εκάστοτε εταιρίας αλλά και της κοινωνίας. Μια από τις αλλαγές οι οποίες μπορούν να συζητηθούν σχετικά με την βελτίωση της παρούσας εφαρμογής είναι η δημιουργία ενός πίνακα στο RESTful API ο οποίος θα αποθηκεύει όλους τους OTP έχουν δημιουργηθεί μέσα σε ένα μικρό χρονικό διάστημα. Η αλλαγή αυτήν θα εξασφαλίζει την ανωνυμία του επομένου αριθμού OTP καθώς θα αποτελείται από διαφορετικούς μη χρησιμοποιήσιμους συνδυασμούς ψηφίων. Εξίσου μια αρκετά σημαντική αλλαγή που αξίζει να αναφερθεί είναι η δημιουργία μιας χρονικής μπάρας στο frontend η οποία θα χρησιμοποιείται για την λήξη του αριθμού OTP, έχοντας ως βασικό χαρακτηριστικό την λήψη του χρόνου λήξης του αριθμού από το API οπότε και δημιουργείται. Οι παραπάνω ιδέες μπορούν να ενισχύσουν της επιτυχή εξασφάλιση της ιδιωτικότητας του χρήστη προσφέροντας του επιπλέον παροχές ασφάλειας.

4.3 Συμπεράσματα

Η ύπαρξη τόσων πηγών καταμέτρησης θυμάτων διαδικτυακής εγκληματικότητας, τα τόσα συνέδρια και ερευνητικά εργαστήρια που πραγματοποίησαν μελέτες σχετικά με τις κυβερνοεπιθέσεις, οι τόσες ιστοσελίδες που αναφέρουν συνεχώς για περιπτώσεις διαρροής δεδομένων, διασταυρώνουν για άλλη μια φορά την έξαρσή της κοινωνικής μηχανικής του κυβερνοχώρου. Μπορεί τα δεδομένα να διαφοροποιούνται από ερευνά σε ερευνά δεν παύει όμως να ανέρχονται σε υψηλά επίπεδα προκαλώντας ανησυχία στους χρήστες του διαδικτύου για την ακεραιότητα των προσωπικών τους δεδομένων. Η ύπαρξη πολυάριθμων κατηγοριών και μεθοδολογιών κυβερνητικών επιθέσεων που γίνονται χρήση καθιστά τους χρήστες αδυνάτους στο να εμπιστευτούν τον οποιοδήποτε συνομιλητή τους που βρίσκεται εξ αποστάσεως, είτε είναι γνησιά οντότητα και προσπαθεί να παρεχει τις υπηρεσίες για θέματα διαδικτυακών συναλλαγών ή προσωπικών υποθέσεων είτε είναι ψευδής. Αυτό καθιστά αυτόματα δύσκολη την συνεργασία για την ανάγκη παροχής βοήθειας που υπάρχει, ειδικά σε άτομα μέσης ηλικίας, οπου η γνώση σχετικά με την τεχνολογία είναι λιγοστή συγκριτικά με άλλες κοινωνικές ομάδες. Απαραίτητή είναι η επαγρύπνηση και η λήψη μέτρων που χρειάζονται να παρθούν για να μειωθεί το ποσοστό και η πιθανότητα επιθέσεων του κυβερνοχώρου. Μια επίθεση που συνηθίζεται να συνηθίζεται προσπαθώντας να καταβάλλει χρηματικά ποσά από λογαριασμούς χρηστών είναι η sim swapping. Αναμφίβολα εταιρές έχουν αναπτύξει εφαρμογές οι οποίες παρέχουν ασφάλεια συναλλαγών μέσω της διπλής ταυτοπροσωπίας δηλαδή αυθεντικοποίησης δυο παραγόντων, οπου αποδεδειγμένα αποτελεί μια από τις πιο ισχυρές δικλίδες ασφάλειας. Εξαιτίας όμως της επίθεσης που ευδοκιμεί υποβαθμίζεται η χρήση της κάρτας sim, κρίνοντας την επίθεση αποτελεσματική. Για την

ανάγκη παροχής όσων των δυνατών μεγαλύτερης ασφάλειας, βάσιμη είναι η δημιουργία μιας εφαρμογής η οποία θα παρεχει τεχνική ασφάλειας TFA με την διαφορά όμως ότι βασίζεται σε μοναδικούς αριθμούς που χαρακτηρίζουν την ταυτότητα της συσκευής του χρήστη. Η εφαρμογή, με την πάροδο του χρόνου, κρίνεται πολύ σημαντικό να εξαλείψει σε μεγάλο βαθμό την επίθεση sim swapping και να εκτελούνται διαδικτυακές συναλλαγές με ασφάλεια. Φυσικά η διαδικασία απαιτεί χρόνο και πολλές δοκιμές για να εξεταστεί η αποδοτικότητα και η αποτελεσματικά της. Τέλος αναγνωρίζεται ότι η κοινωνική μηχανική θέλοντάς και μη, λαμβάνει αρνητικούς χαρακτηρισμούς. Η ανάγκη ανακάλυψης της δύναμης του διαδικτύου, η χρήση της τεχνολογίας για κακόβουλους προσωπικούς σκοπούς, η περιέργεια για τις αρνητικές επιπτώσεις που επιφέρουν οι κυβερνητικές επιθέσεις και πολλών άλλων ειδών ζητημάτων ανακαλύπτει τον λόγο πρόληψης μέτρων. Απαραίτητη είναι η ευαισθητοποίηση σχετικά με την τρέχουσα κατάσταση και η επείγουσα κινητοποίηση για τη μείωση των περιπτώσεων διαδικτυακής απάτης.

Βιβλιογραφία

1. Cambridge contributors. (2021, August 23). *Social engineering (security)*. Cambridge.
2. Merriam-Webster. (n.d.). Social engineering. In Merriam-Webster.com dictionary. Retrieved August 27, 2021
3. Hadnagy, C. (2010). *Social Engineering* (1st ed.). John Wiley & Sons.
4. Federal Bureau of Investigation. (2020, January). 2020 Internet Crime Report. Paul Abbate.
5. hackmageddon.com. (2021, February). 2020 Cyber Attacks Statistics.
6. Europol European Cybercrime Centre. (2020, January). SIM SWAPPING – A MOBILE PHONE SCAM.
7. Europol European Cybercrime Centre. (2021, February). TEN HACKERS ARRESTED FOR STRING OF SIM-SWAPPING ATTACKS AGAINST CELEBRITIES. European Union Agency for Law Enforcement Cooperation.
8. S. (2021, July 7). The Impact of COVID-19 on Security. Security Through Education. <https://www.social-engineer.org/social-engineering/the-impact-of-covid-19-on-security/>
9. S. (2021b, July 7). The Rise of Phishing Attacks: P.S. I Love You. Security Through Education. <https://www.social-engineer.org/newsletter/the-rise-of-phishing-attacks-p-s-iloveyou/>
10. Bisson, D. (2020, July 16). 5 Social Engineering Attacks to Watch Out For. The State of Security. <https://www.tripwire.com/state-of-security/security-awareness/5-social-engineering-attacks-to-watch-out-for/>
11. elemind.com. (2020, February 1). The VERIS Framework. Veris. <http://veriscommunity.net/>
12. Assolini, F. (2020, September 18). Large-scale SIM swap fraud. Securelist. <https://securelist.com/large-scale-sim-swap-fraud/90353/>
13. Education, I. C. (2021, April 9). Application Programming Interface (API). IBM. <https://www.ibm.com/cloud/learn/api>
14. Editor. (2020, February 11). *What is API: Definition, Types, Specifications, Documentation*. AltexSoft. <https://www.altexsoft.com/blog/engineering/what-is-api-definition-types-specifications-documentation/>
15. Siegel, C. (2021, July 29). What is an API? Axway Blog. <https://blog.axway.com/amplify-products/api-management/what-is-an-api>
16. E. (2021a, August 27). How database API works. Educative: Interactive Courses for Software Developers. <https://www.educative.io/edpresso/how-database-api-works>
17. GeeksforGeeks. (2020, May 31). What is Web API and why we use it ? <https://www.geeksforgeeks.org/what-is-web-api-and-why-we-use-it/>
18. Walker, A. (2021, August 19). SOAP Vs. REST: Difference between Web API Services. Guru99. <https://www.guru99.com/comparison-between-web-services.html>

19. Omariba, Zachary B, and Nelson B Masese. 2012. "Security and Privacy of Electronic Banking." *International Journal of Computer Science Issues* 9(3): 432–46.
20. What is a REST API? (2018, January 2). RedHat. <https://www.redhat.com/en/topics/api/what-is-a-rest-api>
21. European Union, Agency for Law Enforcement. 2020. "Internet Organised Crime Threat Assessment." *Iocta* 1(1): 64.
22. Yin, X., He, J., Guo, Y., Han, D., Li, K. C., & Castiglione, A. (2020). An Efficient Two-Factor Authentication Scheme Based on the Merkle Tree. *Sensors*, 20(20), 5735. <https://doi.org/10.3390/s20205735>
23. Alzahrani, A. (2020). Coronavirus Social Engineering Attacks: Issues and Recommendations. *International Journal of Advanced Computer Science and Applications*, 11(5). <https://doi.org/10.14569/ijacsa.2020.0110523>
24. Jover, R. P. (2020). Security Analysis of SMS as a Second Factor of Authentication. *Queue*, 18(4), 37–60. <https://doi.org/10.1145/3424302.3425909>
25. Heartfield, R.; Loukas, G. A taxonomy of attacks and a survey of defence mechanisms for semantic social engineering attacks. *ACM Comput. Surv.* 2016, 48, 1–
26. Madain, A.; Ala, M.A.; Al-Sayyed, R. Online social networks security: Threats, attacks, and future directions. In *Social Media Shaping e-Publishing and Academia*; Springer International Publishing: New York, NY, USA, 2017; pp. 121–132.
27. Rismayani, N. (2017). Rural Population Administration Management System Based on RESTful API in Bone District (Sistem Manajemen Administrasi Kependudukan Tingkat Pedesaan Berbasis RESTful API di Kabupaten Bone). *Journal Pekommas*, 2(1), 7. <https://doi.org/10.30818/jpkm.2017.2020102>
28. Seo, B., Jang, B. S., Oh, H. S., & Park, H. J. (2019). RESTful, Redis Based API Server Platform Design for Automatic API Generation and Data Processing Performance. *The Journal of Korean Institute of Communications and Information Sciences*, 44(5), 895–903. <https://doi.org/10.7840/kics.2019.44.5.895>
29. Hassan, M. A., & Shukur, Z. (2021). A Secure Multi Factor User Authentication Framework for Electronic Payment System. 2021 3rd International Cyber Resilience Conference (CRC). Published. <https://doi.org/10.1109/crc50527.2021.9392564>
30. Hussain, A. A. (2016). A Strong Token Authentication And Cryptography Based Application And Data Access For Cloud Users. *International Journal Of Engineering And Computer Science*. Published. <https://doi.org/10.18535/ijecs/v4i10.02>
31. Gao, L., Zhang, C., & Sun, L. (2011). RESTful Web of Things API in Sharing Sensor Data. 2011 International Conference on Internet Technology and Applications. Published. <https://doi.org/10.1109/itap.2011.6006157>
32. Wylie, P. L., & Crawley, K. (2020). *The Pentester BluePrint: Starting a Career as an Ethical Hacker* (1st ed.). Wiley.
33. Park, W. S., Hwang, D. Y., & Kim, K. H. (2018). A TOTP-Based Two Factor Authentication Scheme for Hyperledger Fabric Blockchain. 2018 Tenth International

Conference on Ubiquitous and Future Networks (ICUFN). Published. <https://doi.org/10.1109/icufn.2018.8436784>

34. Eldefrawy, M. H., Alghathbar, K., & Khan, M. K. (2011). OTP-Based Two-Factor Authentication Using Mobile Phones. 2011 Eighth International Conference on Information Technology: New Generations. Published. <https://doi.org/10.1109/itng.2011.64>

35. Kumar, D., Agrawal, A., & Goyal, P. (2015). Efficiently improving the security of OTP. 2015 International Conference on Advances in Computer Engineering and Applications. Published. <https://doi.org/10.1109/icacea.2015.7164835>

36. Costa, J., & Michalas, A. (2017). Middle Man: An Efficient Two-Factor Authentication Framework. 2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA). Published. <https://doi.org/10.1109/iccubea.2017.8463686>

37. Kisore, N. R., & Sagi, S. (2015). A secure SMS protocol for implementing digital cash system. 2015 International Conference on Advances in Computing, Communications and Informatics (ICACCI). Published. <https://doi.org/10.1109/icacci.2015.7275893>

38. API Guide to set up OTP verification. (2020, January 12). Miniorange. <https://www.miniorange.com/step-by-step-guide-to-set-up-otp-verification#OTPGenerationChallengeRestAPI>

39. TIBCO Software. (2020, January). When To Use JWT Vs. OAuth2.0 Access Token. <https://docs.mashery.com/connectorsguide/GUID-3812EE8B-3770-445C-83F2-FB6D1D54C18A.html>

40. Rosencrance, L., Loshin, P., & Cobb, M. (2021, July 7). two-factor authentication (2FA). SearchSecurity. <https://searchsecurity.techtarget.com/definition/two-factor-authentication>

41. Admin, S. (2020, November 5). 29 Must-know Cybersecurity Statistics for 2020. Cyber Observer. <https://www.cyber-observer.com/cyber-news-29-statistics-for-2020-cyber-observer/>

42. Proofpoint. (2021, July). 2021 State of the Phish.

43. Security, P. (2020, December 17). 43 COVID-19 Cybersecurity Statistics. Panda Security Mediacenter. <https://www.pandasecurity.com/en/mediacenter/news/covid-cybersecurity-statistics/#country>

44. Masse, M. (2011). REST API Design Rulebook. Van Duuren Media.

45. Jelen, S. (2021, August 28). Two-Factor Authentication vs. Multi-Factor Authentication: Differences Explained. Security Trails. <https://securitytrails.com/blog/two-factor-auth-vs-multi-factor-auth>

46. Cox, J. (2021, August 15). T-Mobile Investigating Claims of Massive Customer Data Breach. Vice. https://www.vice.com/en/article/akg8wg/tmobile-investigating-customer-data-breach-100-million?utm_source=reddit.com

47. SIM Swapping – The Achilles’ Heel Of Two-Factor Authentication - The Monitor, Issue 9. | Conventus Law. (2019, December 29). Conventus Law. <https://www.conventuslaw.com/report/sim-swapping-the-achilles-heel-of-two-factor/>

48. Internet Fraud. (2020, May 25). Federal Bureau of Investigation.

<https://www.fbi.gov/scams-and-safety/common-scamsQ-and-crimes/internet-fraud>

49. Frontend vs Backend: What's The Difference? (2020, August 13). PLURALSIGHT. <https://www.pluralsight.com/blog/software-development/front-end-vs-back-end>
50. T-Mobile Shares Additional Information Regarding Ongoing Cyberattack Investigation. (2021, August 18). Business Wire. <https://www.businesswire.com/news/home/20210817005964/en/>
51. K.L., B.K., J.M., & A.N. (2020). An Empirical Study of Wireless Carrier Authentication for SIM Swaps. Department of Computer Science and Center for Information Technology Policy Princeton University. Published.
52. RiskBased SECURITY. (2020, January). 2020 Q3 Report Data Breach QuickView. Inga Goddijn.
53. Shankar, A., & Jebarajakirthy, C. (2019). The influence of e-banking service quality on customer loyalty. *International Journal of Bank Marketing*, 37(5), 1119–1142. <https://doi.org/10.1108/ijbm-03-2018-0063>
54. Awale, S. M., & Gupta, D. P. (2019). Awareness of Sim Swap Attack. *International Journal of Trend in Scientific Research and Development*, Volume-3(Issue-4), 995–997. <https://doi.org/10.31142/ijtsrd23982>