ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗ ΒΙΟΙΑΤΡΙΚΗ

Σύστημα ελέγχου λειτουργίας IoT συσκευών με την χρήση βιομετρικών χαρακτηριστικών του χρήστη

Σοφία Α. Τσαγιοπούλου

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Υπεύθυνοι
Γεώργιος Σπαθούλας
Μέλος ΕΔΙΠ

Αθανάσιος Κακαρούντας
Επίκουρος Καθηγητής

Λαμία, 2020

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΜΕ ΕΦΑΡΜΟΓΕΣ ΣΤΗ ΒΙΟΙΑΤΡΙΚΗ

Σύστημα ελέγχου λειτουργίας IoT συσκευών με την χρήση βιομετρικών χαρακτηριστικών του χρήστη

Σοφία Α. Τσαγιοπούλου

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Επιβλέποντες
Γεώργιος Σπαθούλας
Μέλος ΕΔΙΠ

Αθανάσιος Κακαρούντας
Επίκουρος Καθηγητής

Λαμία, 2020

Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις [1], που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

1. *Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί* ***χωρίς να τα περικλείω σε εισαγωγικά*** *και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάσθηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.*

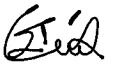
2. *Δέχομαι ότι η αυτολεξεί* ***παράθεση χωρίς εισαγωγικά****, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.*

3. *Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια*

4. *Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.*

Ημερομηνία: 10/6/2020

Ο – Η Δηλ.



(Υπογραφή)

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.

Σύστημα ελέγχου λειτουργίας IoT συσκευών με την χρήση βιομετρικών χαρακτηριστικών του χρήστη

Σοφία Α. Τσαγιοπούλου

Τριμελής Επιτροπή:

Γεώργιος Σπαθούλας, Μέλος ΕΔΙΠ (επιβλέπων)

Ιωάννης Αναγνωστόπουλος, Αναπληρωτής Καθηγητής

Αθανάσιος Κακαρούντας, Επίκουρος Καθηγητής (επιβλέπων)

ΠΕΡΙΛΗΨΗ

Το Internet of Things έχει γίνει βασικό κομμάτι της καθημερινότητας μας με τις συσκευές του να αυξάνονται εκθετικά. Αλλάζει καθημερινά την ζωή μας αυτοματοποιώντας πολλές ενέργειες, κάτι που ενισχύει το βιοτικό επίπεδο των κοινωνιών μας. Παρά τις διευκολύνσεις που προσφέρει εγκυμονεί πολλούς κινδύνους. Τα κυριότερα προβλήματα του IoT συστήματος είναι η ασφάλεια και η ιδιωτικότητα. Σε αυτή την πτυχιακή εργασία παρουσιάζεται η αρχιτεκτονική των IoT αναλύοντας τις απειλές και τρόπους ενίσχυσης της ασφάλειας. Επίσης, επικεντρώνεται στην έννοια της ιδιωτικότητας και την ενίσχυση του με διάφορες τεχνικές. Το σύστημα που προτείνεται σχετίζεται με την ιδιωτικότητα των χρηστών σε ένα έξυπνο περιβάλλον, έχοντας την δυνατότητα οι χρήστες να ελέγχουν πότε οι συσκευές θα συλλέγουν δεδομένα, που θα αποστέλλονται και την ροή αυτών. Η λειτουργία αυτή εφαρμόζεται όταν οι χρήστες βρίσκονται στον χώρο αναγνωρίζοντας την φυσική τους παρουσία μέσω του δακτυλικού τους αποτυπώματος. Το σύστημα υλοποιήθηκε με ένα Raspberry Pi 3, έναν αισθητήρα δακτυλικού αποτυπώματος και την χρήση του πακέτου Flask της Python για την υλοποίηση μιας διαδικτυακής εφαρμογής.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Internet of Things, Ιδιωτικότητα, Βιομετρία

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Internet of Things, Ιδιωτικότητα, Ασφάλεια, δακτυλικό αποτύπωμα, Raspberry Pi

ABSTRACT

The Internet of Things has become part of our lives and its devices are increasing exponentially. It changes our daily life automating many actions, something that strengthens the standard of living of our societies. Despite the facilitations it offers, it poses many risks. The main issues of the IoT system are security and privacy. This project presents the IoT's architecture, analyzing the threats and methods to enhance security. Also, focuses on the concept of privacy and its enhancement with various techniques. The proposed system is related to the privacy of users in a smart environment, allowing users to control when the devices collect data, where they send them and their flow. This function is applied when the users are in the smart environment as their physical presence is recognized through their fingerprint. The system was implemented with a Raspberry Pi 3, a fingerprint sensor and the Flask Python package to create a web application.

SUBJECT AREA: Internet of Things, Privacy, Biometrics

KEYWORDS: Internet of Things, Security, Privacy, fingerprint, Raspberry Pi

Στους αγαπημένους μου.

ΕΥΧΑΡΙΣΤΙΕΣ

Καθώς η εργασία αυτή σημάνει το τέλος των προπτυχιακών μου σπουδών, θα ήθελα να ευχαριστήσω τους ανθρώπους που συνέβαλαν στην επιτυχή ολοκλήρωση της πτυχιακής μου εργασίας και αυτούς που με στήριξαν αυτά τα χρόνια.

Πρώτα από όλους θέλω να ευχαριστήσω τους επιβλέποντες της πτυχιακής μου εργασίας. Τον κύριο Γεώργιο Σπαθούλα και τον κύριο Αθανάσιο Κακαρούντα για την εμπιστοσύνη που μου έδειξαν, την συνεχή καθοδήγηση τους και την άριστη συνεργασία μας. Είμαι πραγματικά ευγνώμων που μου δόθηκε η ευκαιρία να συνεργαστώ μαζί τους.

Επίσης, αισθάνομαι την ανάγκη να ευχαριστήσω τους καθηγητές της σχολής που με τροφοδότησαν με γνώσεις και με ώθησαν να αγαπήσω την επιστήμη της Πληροφορικής.

Τέλος, θέλω να ευχαριστήσω τους γονείς μου που με υποστήριξαν και με υποστηρίζουν με κάθε δυνατό τρόπο αλλά και την αδελφή μου που σε κάθε μου βήμα ήταν εκεί.

ΠΕΡΙΕΧΟΜΕΝΑ

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

1. ΕΙΣΑΓΩΓΗ

Το Internet of Things επιτρέπει ηλεκτρονικές συσκευές συνδεδεμένες στο διαδίκτυο να υπάρχουν στο περιβάλλον συλλέγοντας και ανταλλάσσοντας πληροφορίες με άλλες συσκευές του δικτύου. Έτσι, μπορούν να αναγνωρίσουν γεγονότα και αλλαγές στο περιβάλλον τους και να ενεργούν αυτόνομα, κυρίως χωρίς καμία αλληλεπίδραση με τον άνθρωπο. Τα πλεονεκτήματα του IoT είναι σχεδόν απεριόριστα και οι εφαρμογές του αλλάζουν τον τρόπο με τον οποίο εργαζόμαστε και ζούμε. Εξοικονομώντας χρόνο, πόρους και ανοίγοντας νέες ευκαιρίες για ανάπτυξη, καινοτομία και ανταλλαγή γνώσεων μεταξύ των συσκευών. Προβλέπεται ότι έως το 2020 το Internet of Things θα επεκτείνεται πολύ περισσότερο από 50 δισεκατομμύρια συσκευές (εκτός από υπολογιστές, tablet και smartphone). Ωστόσο, η ύπαρξη ενός τόσο μεγάλου δικτύου διασυνδεδεμένων συσκευών θα δημιουργήσει σίγουρα νέες απειλές ασφάλειας, ιδιωτικότητας και εμπιστοσύνης.

Οι περισσότερες από τις συσκευές των IoT έχουν περιορισμένη ισχύ, αποθήκευση και υπολογιστικές δυνατότητες. Συνεπώς συλλέγονται δεδομένα, διαμορφώνονται ώστε να είναι πιο εύκολη η εύρεση τους και αποθηκεύονται στο cloud. Το όφελος της πρόσβασης στα δεδομένα από οπουδήποτε και οποιαδήποτε στιγμή δημιουργεί σοβαρά ζητήματα ασφάλειας και ιδιωτικότητας και οδηγούν σε πολλά προβλήματα όπως η έκθεση των προσωπικών και ευαίσθητων πληροφοριών των χρηστών και η απώλεια εμπιστοσύνης.

Η ασφάλεια και η ιδιωτικότητα είναι μερικά από τα κρίσιμα ζητήματα του IoT. Είναι αναγκαίο να προστατεύονται οι συσκευές των συστημάτων από κακόβουλες επιθέσεις αλλά να προστατεύονται και τα προσωπικά δεδομένων των χρηστών. Σε αυτή την πτυχιακή εργασία παρουσιάζεται ένα σύστημα ενίσχυσης της ιδιωτικότητας των χρηστών. Αυτό επιτυγχάνεται μέσω του καθορισμού της συμπεριφοράς των συσκευών του χώρου βάσει των προτιμήσεων των χρηστών που είναι παρόντες κάθε στιγμή στο περιβάλλον. Ο χρήστης με αυτόν τον τρόπο νιώθει πιο ασφαλής αφού μπορεί να ελέγξει την αποστολή και την ροή των δεδομένων στον χώρο. Το υλικό που χρησιμοποιήθηκε ήταν ένα Raspberry Pi 3 συνδεδεμένο με έναν αισθητήρα δακτυλικού αποτυπώματος, με το Raspberry να λειτουργεί ως access point και firewall.

Η εργασία είναι οργανωμένη με τον εξής τρόπο. Η πρώτη ενότητα αποτελεί την εισαγωγή, η δεύτερη παρουσιάζει τις εφαρμογές των IoT, την αρχιτεκτονική, τους κινδύνους στην ασφάλεια αυτής και μεθόδους για ασφαλέστερη αρχιτεκτονική. Επίσης, αναλύεται η ιδιωτικότητα στα IoT συστήματα και τρόποι ενίσχυσης της ιδιωτικότητας. Η τρίτη ενότητα

αναφέρει άρθρα σχετιζόμενα με το θέμα. Η τέταρτη ενότητα παρουσιάζει την υλοποίηση του προτεινόμενου συστήματος και τέλος παραθέτονται τα συμπεράσματα.

2. ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΙΟΤ, ΑΣΦΑΛΕΙΑ ΚΑΙ ΙΔΙΩΤΙΚΟΤΗΤΑ

2.1 Εισαγωγή στο Internet of things

Το πεδίο που επιτρέπει σε αντικείμενα και μηχανήματα να συνδέονται και να επικοινωνούν μεταξύ τους με την παρουσία του διαδικτύου χωρίς να απαιτείται αλληλεπίδραση μεταξύ ανθρώπου με άνθρωπο και ανθρώπου με υπολογιστή, ονομάζεται IoT [30]. Η ιδέα πίσω από αυτήν τη νέα τεχνολογία είναι η αυτοματοποίηση της εργασίας και η σύνδεση των συσκευών μέσω του διαδικτύου που χρησιμοποιούμε στην καθημερινή μας ζωή. Ειδικοί τύποι αισθητήρων επισυνάπτονται σε κάθε αντικείμενο για να συλλάβουν τις πληροφορίες από τον φυσικό κόσμο. Οι πληροφορίες αναλύονται με τοπική επεξεργασία για την αφαίρεση των περιττών δεδομένων και την αποθήκευση των χρήσιμων πληροφοριών σε έναν τοπικό αποθηκευτικό χώρο. Οι πληροφορίες αποστέλλονται από εκεί στο cloud storage όπου όλα τα αντικείμενα στέλνουν τις συλλεγόμενες πληροφορίες τους. Τέλος, χρησιμοποιώντας τις συγκεντρωμένες πληροφορίες, λαμβάνεται η κατάλληλη απόφαση. Δεν είναι υποχρεωτικό να πραγματοποιείται πάντα κάποια ενέργεια χρησιμοποιώντας αυτές τις πληροφορίες, αλλά μπορούμε να διαχειριστούμε και να ελέγξουμε τα αντικείμενα και τις μηχανές από απόσταση και να χρησιμοποιήσουμε αυτές τις πληροφορίες για να διατηρήσουμε αρχεία για μελλοντική χρήση.

Υπάρχουν πολλές τεχνολογίες και αισθητήρες που χρησιμοποιούνται για την εφαρμογή της ιδέας του IoT. Οι τεχνολογίες επικοινωνίας που χρησιμοποιούνται είναι η αναγνώριση ραδιοσυχνοτήτων (RFID), η επικοινωνία κοντά στο πεδίο (NFC), το ασύρματο διαδίκτυο αισθητήρων (WSN) αλλά και άλλες [44].

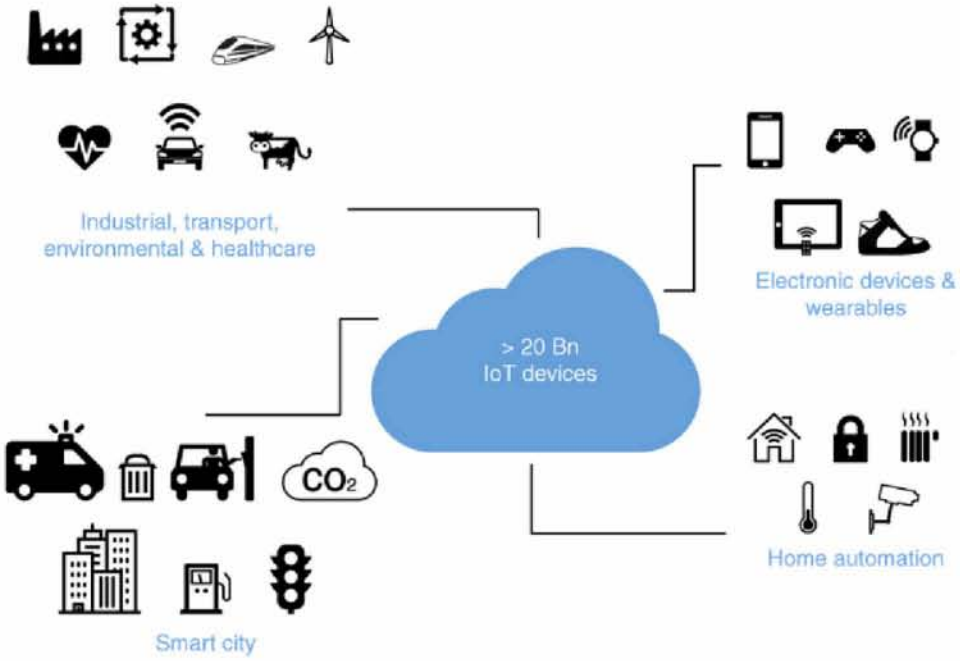
Η αναγνώριση ραδιοσυχνοτήτων RFID (RFID) είναι μικρής εμβέλειας τεχνολογία επικοινωνίας όπου μια ετικέτα RFID επικοινωνεί με συσκευή ανάγνωσης RFID μέσω ηλεκτρομαγνητικών ραδιοσυχνοτήτων. Οι ετικέτες ενδέχεται να περιέχουν διαφορετικές μορφές δεδομένων, η πιο συχνή μορφή που χρησιμοποιείται στις IoT συσκευές είναι ο Ηλεκτρονικός Κωδικός Προϊόντος (EPC).

Η NFC τεχνολογία βασίζεται στο πρότυπο RFID. Το NFC είναι μικρής εμβέλειας πρότυπο επικοινωνίας όπου οι συσκευές είναι σε θέση να αναπτύξουν ραδιοφωνική επικοινωνία μεταξύ τους όταν βρίσκονται σε κοντινή απόσταση. Κάθε ετικέτα NFC περιέχει

ένα μοναδικό αναγνωριστικό (UID) που σχετίζεται με την ετικέτα.

Οι αισθητήρες είναι συσκευές που παρακολουθούν τα χαρακτηριστικά του περιβάλλοντος ή άλλα αντικείμενα όπως θερμοκρασία, υγρασία, κίνηση και ποσότητα. Όταν πολλαπλοί αισθητήρες χρησιμοποιούνται μαζί και αλληλοεπιδρούν, συντελούν το ασύρματο δίκτυο αισθητήρων (WSN). Το οποίο μπορεί να περιέχει επίσης πύλες που συλλέγουν δεδομένα από τους αισθητήρες και τα στέλνουν σε έναν διακομιστή.

Ο αριθμός των συσκευών IoT αυξάνεται κάθε μέρα. Ο λόγος για την αύξηση του αριθμού των συσκευών IoT είναι ότι παρέχουν άνεση στην ανθρώπινη ζωή και εκτελούν εργασίες με καλύτερα αποτελέσματα από τους ανθρώπους.



Σχήμα 2.1: Εφαρμογές των IoT συσκευών [1].

Μία εφαρμογή των IoT είναι ο τομέας της υγειονομικής περίθαλψης όπου οι αισθητήρες χρησιμοποιούνται για τον έλεγχο της θερμοκρασίας του ανθρώπου, της αρτηριακής πίεσης και του καρδιακού ρυθμού. Συλλέγοντας έτσι δεδομένα για μια καλύτερη ιατρική περίθαλψη [25]. Ακόμη μια εφαρμογή είναι οι έξυπνες φάρμες. IoT συσκευές χρησιμοποιούνται για να ελέγχεται το περιβάλλον των φυτών που καλλιεργούνται ώστε να παρέχονται οι καλύτερες δυνατές συνθήκες αλλά και να αποτρέπονται ασθένειες [21],[34]. Ακόμη στα ζώα τοποθετούνται αισθητήρες οι οποίοι αντλούν πληροφορίες για την υγεία τους [24].

Με τα χρόνια τα IoT wearables γίνονται όλο και πιο γνωστά λόγω της ικανότητάς

τους να ανιχνεύουν, να υπολογίζουν και να επικοινωνούν. Είναι έξυπνα ρολόγια τα οποία μπορούν να συλλέγουν ατομικές πληροφορίες υγείας και ευεξίας, σωματική δραστηριότητα, και άλλες κρίσιμες παραμέτρους που επηρεάζουν την ποιότητα της καθημερινής ζωής [16]. Τέλος, μια κοινή χρήση των IoT συσκευών είναι στα έξυπνα σπίτια τα οποία μέσω αυτών των συσκευών αυτοματοποιούνται ενέργειες. Θερμοστάτες που μπορούν να ελέγξουν την θερμοκρασία, έξυπνες κλειδαριές οι οποίες όταν ανιχνεύουν παράνομη κίνηση ειδοποιούν τον ιδιοκτήτη, ενημέρωση του ιδιοκτήτη για κάποια συσκευή που δεν συμπεριφέρεται ως συνήθως, είναι κάποιες από τις λειτουργίες που περιλαμβάνουν τα έξυπνα σπίτια.

2.2 Ασφάλεια IoT

Τα IoT συστήματα είναι ένα πρόσφατο πεδίο, το οποίο επεκτείνει το όριο του διαδικτύου για να συμπεριλάβει μία ποικιλία από υπολογιστικές συσκευές. Η σύνδεση πολλών IoT συσκευών με αυτόνομο τρόπο στο διαδίκτυο φέρνει πολλές προκλήσεις, με την ασφάλεια να εκτίθεται σε ένα ευρύ και συχνά άγνωστο κοινό. Όμως, πολλές από τις βασικές μεθόδους ασφαλείας δεν είναι δυνατόν να εφαρμοστούν σε IoT συστήματα λόγω των εγγενών ορίων (πολύ περιορισμένης μνήμης, υπολογιστικής ισχύος και τροφοδοσίας) στις ικανότητες των low-end IoT συσκευών, οι οποίες αντιπροσωπεύουν την πλειοψηφία τους.

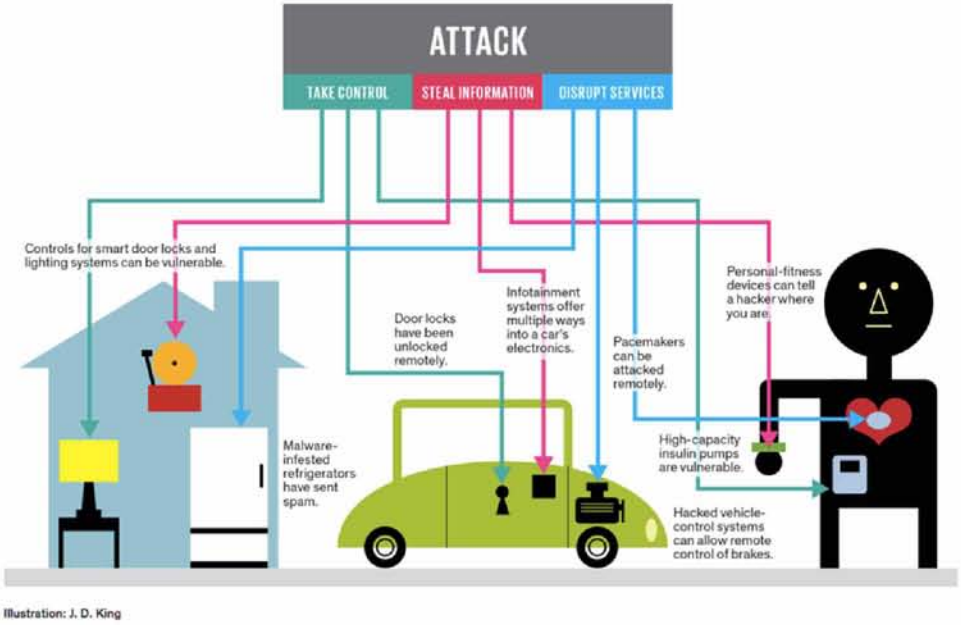
Οι περισσότερες από τις IoT συσκευές (έξυπνες τηλεοράσεις ,συστήματα φωνητικής καθοδήγησης, οικιακές συσκευές κ.α.) είναι συνδεδεμένες στο διαδίκτυο σχεδόν πάντα, ενώ άλλες συνδέονται στο κινητό ή στον υπολογιστή τουλάχιστον μία φορά την μέρα. Το γεγονός αυτό εγκυμονεί πολλούς κινδύνους από ηλεκτρονικούς εγκληματίες, για παράδειγμα:

- Να κλαπούν προσωπικά δεδομένα.
- Να σταματήσει κάποια συσκευή να δουλεύει ώσπου ο ιδιοκτήτης να πληρώσει τον εγκληματία.
- Να επιτεθεί σε άλλες συνδεδεμένες συσκευές.
- Να σταλούν spam emails .
- Έκθεση προσωπικής υγείας.

- Διασύνδεση με τις χρηματικές συναλλαγές.

Οι IoT συσκευές έχουν σχεδιαστεί για να είναι εύκολες στην χρήση και όχι ασφαλείς. Οι συσκευές αυτές χρησιμοποιούν το Universal Plug and Play Protocol το οποίο χρησιμοποιείται όταν η συσκευή συνδέεται από απόσταση. Το Universal Plug and Play (UPnP) [18] είναι ένα σύνολο πρωτοκόλλων δικτύωσης που επιτρέπουν σε δικτυωμένες συσκευές όπως laptops, εκτυπωτές, Internet gateways, σημεία πρόσβασης Wi-Fi και κινητά τηλέφωνα να ανακαλύπτουν άψογα την παρουσία του άλλου στο δίκτυο και να δημιουργούν λειτουργικές υπηρεσίες δικτύου για κοινή χρήση δεδομένων, επικοινωνίες και ψυχαγωγία. Η διαδικασία αυτή όμως, γίνεται αυτόματα και χωρίς έλεγχο ταυτότητας.

Επίσης, για τις IoT συσκευές υπάρχουν προκαθορισμένοι κωδικοί ασφαλείας με σκοπό να γινεί πιο απλή η σύνδεση τους. Αυτό, καθιστά τους κωδικούς πρόσβασης ανιχνεύσιμους με αποτέλεσμα να δοθούν σε μη εξουσιοδοτημένους χρήστες προσωπικά δεδομένα.



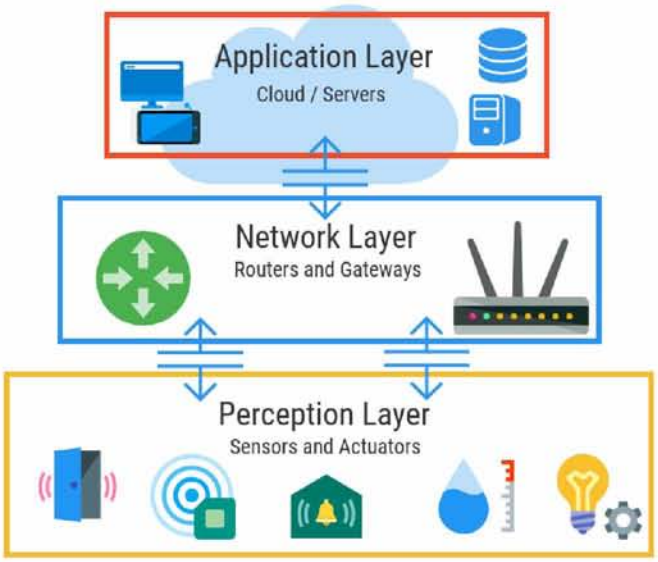
Σχήμα 2.2: Πιθανές επιθέσεις σε IoT συσκευές [22].

2.3 Αρχιτεκτονική IoT και η ασφαλεία

Οι ερευνητές έχουν διχαστεί με την αρχιτεκτονική των IoT συστημάτων. Κάποιοι

υποστηρίζουν ότι υπάρχουν τρία επίπεδα ενώ άλλοι ότι τα επίπεδα αυτά είναι τέσσερα. Πιστεύουν ότι, λόγω της βελτίωσης του IoT, η αρχιτεκτονική τριών επιπέδων δεν μπορεί να ικανοποιήσει τις απαιτήσεις των εφαρμογών. Επίσης, ακόμη μια αρχιτεκτονική που έχει προταθεί είναι αυτή των πέντε επιπέδων. Η αρχιτεκτονική των πέντε επιπέδων προτάθηκε λόγω των πολλών προκλήσεων στην ασφάλεια και στην ιδιωτικότητα.

2.3.1 Αρχιτεκτονική 3 επιπέδων και πιθανές επιθέσεις



Σχήμα 2.3: Η αρχιτεκτονική 3 επιπέδων IoT συστημάτων [7].

- **Επίπεδο Perception**: Είναι το πιο βασικό επίπεδο το οποίο συλλέγει όλα τα είδη πληροφοριών μέσω του φυσικού εξοπλισμού και εντοπίζει τον φυσικό κόσμο. Οι πληροφορίες περιλαμβάνουν ιδιότητες αντικειμένων, περιβαλλοντικές συνθήκες κ.α. Οι φυσικοί εξοπλισμοί περιλαμβάνουν αναγνώστη RFID, όλα τα είδη αισθητήρων, GPS και άλλους εξοπλισμούς. Το βασικό στοιχείο σε αυτό το επίπεδο είναι οι αισθητήρες για τη λήψη και την απεικόνιση του φυσικού κόσμου στον ψηφιακό κόσμο.

- **Επίπεδο Network**: Είναι υπεύθυνο για την αξιόπιστη μετάδοση πληροφοριών από το επίπεδο perception, την αρχική επεξεργασία πληροφοριών, την ταξινόμηση και τον πολυμερισμό. Σε αυτό το επίπεδο η μετάδοση πληροφοριών βασίζεται σε πολλά βασικά δίκτυα, τα οποία είναι το διαδίκτυο, το δίκτυο κινητής επικοινωνίας, τα δορυφορικά δίκτυα, το ασύρματο δίκτυο, η υποδομή δικτύου και τα πρωτόκολλα επικοι-

νωνίας που είναι επίσης απαραίτητα για την ανταλλαγή πληροφοριών μεταξύ συσκευών.

- **Επίπεδο Application**: Είναι το ανώτατο και το τελικό επίπεδο. Παρέχει τις εξατομικευμένες υπηρεσίες ανάλογα με τις ανάγκες των χρηστών. Οι χρήστες μπορούν να έχουν πρόσβαση στο IoT μέσω της διασύνδεσης του επιπέδου application, χρησιμοποιώντας τηλεόραση, προσωπικό υπολογιστή ή κινητό εξοπλισμό και ούτω καθεξής.

2.3.2 Απειλές ασφάλειας στο επίπεδο Perception

- **Επίθεση replay**: Είναι μια επίθεση κατά την οποία ένας εισβολέας παρακολουθεί τη συζήτηση μεταξύ αποστολέα και παραλήπτη και λαμβάνει αυθεντικές πληροφορίες από τον αποστολέα. Ο εισβολέας στέλνει τις ίδιες επικυρωμένες πληροφορίες, στο θύμα, που είχε ήδη λάβει αποδεικνύοντας την ταυτότητα και τη αυθεντικότητά του. Το μήνυμα είναι σε κρυπτογραφημένη μορφή, οπότε ο παραλήπτης μπορεί να το αντιμετωπίζει ως ένα σωστό αίτημα και να προβεί σε ενέργειες που επιθυμεί ο εισβολέας [26].

- **Ψεύτικος και κακόβουλος κόμβος**: Είναι μια επίθεση στην οποία ένας εισβολέας προσθέτει έναν κόμβο στο σύστημα και εισάγει ψεύτικα δεδομένα. Στόχος είναι να σταματήσει η μετάδοση πραγματικών πληροφοριών. Ένας κόμβος που προστίθεται από έναν εισβολέα καταναλώνει πολύτιμη ενέργεια από τους πραγματικούς κόμβους και ενδεχομένως μπορεί να καταστρέψει το δίκτυο.

- **Επίθεση routing**: Στην διαδικασία συλλογής δεδομένων υπάρχει αναμετάδοση και προώθηση δεδομένων. Έτσι, ο εισβολέας μέσω ενδιάμεσων κόμβων ενδέχεται να επιτεθεί στα δεδομένα κατά τη διάρκεια της προώθησης [45].

- **Επίθεση Timing**: Επιτρέπει σε έναν εισβολέα να ανακαλύψει ευπάθειες και να εξαγάγει μυστικά που διατηρούνται στην ασφάλεια ενός συστήματος παρατηρώντας πόσο καιρό χρειάζεται το σύστημα να ανταποκριθεί σε διαφορετικά ερωτήματα, εισόδους ή κρυπτογραφικούς αλγόριθμους.

- **Υποκλοπές και παρεμβολές**: Επειδή οι περισσότερες συσκευές στο IoT επικοινωνούν μέσω ασύρματων δικτύων, η ευπάθεια έγκειται στο γεγονός ότι οι πληροφορίες παρέχονται σε ασύρματα δίκτυα. Οι σύνδεσμοι μπορούν να παραβλεφθούν από μη

εξουσιοδοτημένους χρήστες. Ο αντίπαλος μπορεί επίσης να στείλει δεδομένα θορύβου ή σήμα για παρέμβαση στις πληροφορίες που παραδίδονται σε ασύρματες συνδέσεις [26].

2.3.3 Απειλές ασφάλειας στο επίπεδο Network

- **Denial-of-Service (DoS) Attacks**: Οι επιθέσεις DoS μπορούν να καταναλώσουν όλους τους διαθέσιμους πόρους στο IoT με το να επιτίθενται σε πρωτόκολλα δικτύου ή να κατακλύζουν το δικτύου IoT με μαζική κίνηση, καθιστώντας τις υπηρεσίες του συστήματος IoT μη διαθέσιμες.

- **Man in the Middle Attack**: Μια κακόβουλη συσκευή που ελέγχεται από τον αντίπαλο μπορεί ουσιαστικά να βρίσκεται μεταξύ δύο συσκευών επικοινωνίας στο IoT. Κλέβοντας τις πληροφορίες αναγνώρισης των δύο συσκευών, η κακόβουλη συσκευή μπορεί να είναι μια μεσαία συσκευή για αποθήκευση και προώθηση όλων των δεδομένων που ανταλλάσσουν οι δύο συσκευές. Οι δύο αυτές συσκευές δεν μπορούν να ανιχνεύσουν την ύπαρξη της κακόβουλης συσκευής, και αντ 'αυτού πιστεύουν ότι αυτές επικοινωνούν απευθείας μεταξύ τους [31], [26].

- **Μη εξουσιοδοτημένη πρόσβαση**: Ένας μεγάλος αριθμός συσκευών που βασίζονται σε RFID είναι ενσωματωμένες στο IoT, οι περισσότερες από τις RFID ετικέτες δεν διαθέτουν κατάλληλους μηχανισμούς ελέγχου ταυτότητας. Έτσι, μπορούν να γίνουν προσβάσιμες και οι πληροφορίες που αποθηκεύονται σε αυτές τις ετικέτες μπορούν να ληφθούν, να τροποποιηθούν και να διαγραφούν από τον εισβολέα [5].

2.3.4 Απειλές ασφάλειας στο επίπεδο Application

- **Επίθεση Phishing**: Ο αντίπαλος μπορεί να λάβει τα εμπιστευτικά δεδομένα των χρηστών, όπως ταυτοποίηση και κωδικούς πρόσβασης. Αυτό επιτυγχάνεται με πλαστογράφηση των διαπιστευτηρίων ελέγχου ταυτότητας των χρηστών μέσω των μολυσμένων ιστότοπων ηλεκτρονικού ταχυδρομείου και ηλεκτρονικού ψαρέματος [26].

- **Κακόβουλα σενάρια**: Τα κακόβουλα σενάρια αντιπροσωπεύουν τα σενάρια που προστίθενται σε λογισμικό, τροποποιούνται στο λογισμικό, και διαγράφονται από το αυτό με σκοπό να βλάψουν τις λειτουργίες του συστήματος IoT. Επειδή όλες οι εφαρμογές IoT είναι συνδεδεμένες στο διαδίκτυο, ο αντίπαλος μπορεί εύκολα να

επιτεθεί. Έτσι, μπορεί να δημιουργηθεί διαρροή εμπιστευτικών δεδομένα και ακόμη και να προκληθεί πλήρης τερματισμός του συστήματος.

2.4 Αρχιτεκτονική 4 επιπέδων και πιθανές επιθέσεις

Ο λόγος για την δημιουργία ενός ακόμη επιπέδου είναι η ασφάλεια στην αρχιτεκτονική του IoT. Στη αρχιτεκτονική των τριών επιπέδων αποστέλλονται οι πληροφορίες απευθείας στο επίπεδο Network και έτσι αυξάνονται οι πιθανότητες απειλών. Στην αρχιτεκτονική των τεσσάρων επιπέδων, οι πληροφορίες αποστέλλονται στο επίπεδο Support οι οποίες λαμβάνονται από το επίπεδο Perception. Το νέο επίπεδο επιβεβαιώνει ότι οι πληροφορίες αποστέλλονται από τους αυθεντικούς χρήστες και προστατεύονται από απειλές. Υπάρχουν πολλοί τρόποι επαλήθευσης των χρηστών και των πληροφοριών. Η πιο συχνά χρησιμοποιούμενη μέθοδος είναι ο έλεγχος ταυτότητας. Ακόμη, αποστέλλει τις πληροφορίες στο επίπεδο Network. Όμως υπάρχουν διάφορες επιθέσεις που μπορούν να επηρεάσουν αυτό το επίπεδο, όπως για παράδειγμα η επίθεση DoS και η μη εξουσιοδοτημένη πρόσβαση που αναλύθηκαν παραπάνω.

2.5 Ενίσχυση ασφάλειας

Για το πρώτο επίπεδο, ο έλεγχος ταυτότητας του κόμβου είναι απαραίτητος για την αποτροπή της παράνομης πρόσβασης σε αυτούς. Δεύτερον, για την προστασία της ιδιωτικότητας της μετάδοσης πληροφοριών μεταξύ των κόμβων, η κρυπτογράφηση των δεδομένων είναι σημαντική. Όσον αφορά το Network επίπεδο, οι υφιστάμενοι μηχανισμοί ασφάλειας της επικοινωνίας είναι δύσκολο να εφαρμοστούν. Ο έλεγχος ταυτότητας είναι ένα είδος μηχανισμού για την αποτροπή των παράνομων κόμβων, και είναι η προϋπόθεση του μηχανισμού ασφαλείας. Η εμπιστευτικότητα και η ολοκληρωτικότητα είναι εξίσου σημαντικές, επομένως πρέπει να εδραιωθεί μηχανισμός εμπιστευτικότητας και ολοκλήρωσης δεδομένων. Ένα ακόμη πρόβλημα που πρέπει να επιλυθεί σε αυτό το επίπεδο είναι η επίθεση DoS.

Το Support επίπεδο έχει ανάγκη από cloud computing. Δηλαδή, την χρήση ενός δικτύου απομακρυσμένων διακομιστών που φιλοξενούνται στο διαδίκτυο για την απο-

θήκευση, διαχείριση και επεξεργασία δεδομένων αντί για έναν τοπικό διακομιστή ή έναν προσωπικό υπολογιστή. Όπως επίσης, και για ασφαλή multiparty computation , δημιουργία μεθόδων ώστε κάποια μέρη να υπολογίζουν από κοινού μια λειτουργία πάνω από τις εισόδους τους διατηρώντας παράλληλα αυτές τις εισόδους ιδιωτικές. Ακόμη χρειάζεται, ισχυρούς αλγόριθμους κρυπτογράφησης και πρωτόκολλο κρυπτογράφησης, ισχυρότερη τεχνολογία ασφάλειας συστημάτων και anti-virus. Το application επίπεδο χρειάζεται συμφωνία επαλήθευσης ταυτότητας και κλειδιού σε ένα ετερογενές δίκτυο και προστασία της ιδιωτικής ζωής του χρήστη. Επιπλέον, η εκπαίδευση και η διαχείριση είναι πολύ σημαντικές για την ασφάλεια των πληροφοριών, ειδικά του κωδικού πρόσβασης [41].

2.6 Η ιδιωτικότητα στα IoT συστήματα

Πολλοί ερευνητές θεωρούν πως η ιδιωτικότητα αποτελεί κομμάτι της ασφάλειας. Αν και υπάρχει σημαντικά μεγάλος όγκος αλληλοεπικαλύψεων και μερικές διασταυρώσεις μεταξύ των δύο εννοιών, υπάρχουν αξιοσημείωτες διαφορές σε αυτούς τους όρους. Η ιδιωτικότητα είναι ένας όρος που σχετίζεται με τα άτομα και τα δεδομένα τους, ιδιαίτερα προσωπικά ή ευαίσθητα δεδομένα, πρόσβαση σε δεδομένα χωρίς άδεια ή χρήση με τρόπο που ο κάτοχος δεν εγκρίνει [37]. Επιπλέον, ιδιωτικότητα σημαίνει ότι κάθε άτομο έχει το δικαίωμα να καθορίσει την ποσότητα των δεδομένων που επιτρέπεται για δημόσια προβολή αλλά και τον βαθμό που θα αλληλοεπιδρά με το περιβάλλον [23]. Η ασφάλεια, από την άλλη, προσπαθεί να προστατεύσει κατά βάση τις συσκευές από επιθέσεις.

Η ασφάλεια και η ιδιωτικότητα αποτελούν σημαντικά ζητήματα για συσκευές IoT, εισάγοντας νέες ανησυχίες σχετικά με την ιδιωτικότητα των χρηστών στο διαδίκτυο. Αυτό συμβαίνει επειδή οι συσκευές όχι μόνο συλλέγουν προσωπικά στοιχεία όπως ονόματα και αριθμούς τηλεφώνων, αλλά μπορούν επίσης να παρακολουθούν τις δραστηριότητες των χρηστών. Όλες αυτές οι προσωπικές πληροφορίες ανταλλάσσονται μεταξύ συσκευών και αποθηκεύονται, μέσω του διαδικτύου. Έτσι, μπορεί να υπάρξει διαρροή δεδομένων από επιθέσεις, όπως αυτές που αναφέρθηκαν παραπάνω.

Για παράδειγμα, οι έξυπνες τηλεοράσεις πρόσφατης τεχνολογίας διαθέτουν ενσωματωμένους αισθητήρες κάμερας. Εάν ένας εισβολέας μπορεί να αναγνωρίσει συσκευές IoT γνωστές ως τρωτά σημεία, όπως οι έξυπνες τηλεόρασης, μπορούν όχι μόνο να ελέγχουν την συσκευή αλλά και το περιβάλλον. Δηλαδή, ένας εισβολέας μπορεί να ελέγχει την

τηλεόρασή για αλλαγή καναλιών ή ελέγχων έντασης, αλλά επίσης να παρακολουθεί τις καθημερινές κινήσεις και τις συνομιλίες των χρηστών χρησιμοποιώντας την ενσωματωμένη κάμερα και το μικρόφωνο.

Εισάγοντας κακόβουλο κώδικα σε μια εφαρμογή για κινητά που είναι συνδεδεμένα στο σύστημα IoT, ένας εισβολέας έχει την δυνατότητα να εκτελέσει επιβλαβείς λειτουργίες. Η κακόβουλη απειλή εισαγωγής κώδικα μπορεί να προκαλέσει ζημιά λόγω διαρροής γενικών δεδομένων μέσω αισθητήρων GPS, αισθητήρων κίνησης, αισθητήρων ήχου και αισθητήρων κάμερας τοποθετημένων σε συσκευές του χώρου ή σε κινητές συσκευές. Όταν ένας εισβολέας αποκτά πρόσβαση σε δεδομένα τοποθεσίας για μια κινητή συσκευή ή μια συσκευή με δυνατότητα GPS, μπορεί να συμπεράνει, για παράδειγμα, αν ο κάτοικος ενός έξυπνου σπιτιού βρίσκεται στο σπίτι.

Τέλος, ακόμη και οι έξυπνες καφετιέρες μπορούν να αποτελέσουν απειλή. Οι συσκευές αυτές ελέγχονται απο τα κινητά τηλέφωνα των χρηστών. Ο εισβολέας μπορεί να χρησιμοποιήσει τις έξυπνες καφετιέρες ως είσοδο στα κινητά τηλέφωνα. Με αυτό τον τρόπο μπορεί να υποκλέψει προσωπικά στοιχεία, όπως κωδικούς τραπεζικών λογαριασμών [32].

Οι χρήστες αναμένουν ένα συγκεκριμένο επίπεδο ιδιωτικότητας κατά τη χρήση προϊόντων οικιακού αυτοματισμού και εμπιστεύονται τα δεδομένα τους με την προϋπόθεση πως θα προστατευτούν. Καθώς δημιουργούνται δεδομένα IoT, κοινοποιούνται σε έναν πάροχο υπηρεσιών IoT και συνήθως φιλοξενούνται στο cloud από υπηρεσία τρίτου. Παρά τα μέτρα που έχουν ληφθεί για τη διασφάλιση της ιδιωτικότητας του χρήστη, συμπεριλαμβανομένης της κρυπτογράφησης της επικοινωνίας του πελάτη με τον διακομιστή, τα δεδομένων σχετικά με έναν χρήστη (συμπεριφορές, μοτίβα, προτιμήσεις) μπορούν να διαρρεύσουν.

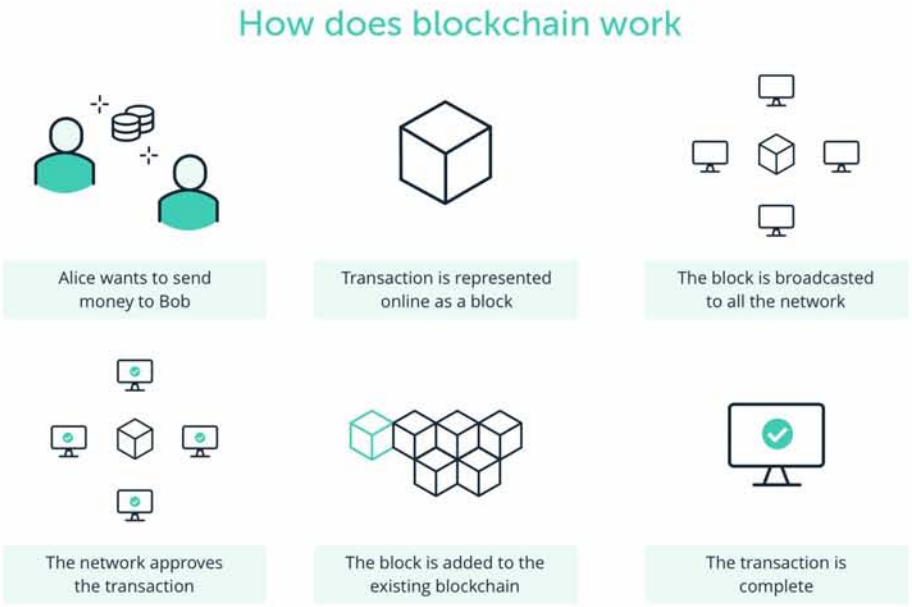
2.7 Ενίσχυση ιδιωτικότητας

Η αυθεντικοποίηση του χρήστη θεωρείται σημαντικό στοιχείο ασφάλειας στα IoT συστήματα. Βασικές αδυναμίες των IoT συστημάτων είναι οι ευάλωτοι κωδικοί πρόσβασης και η πρόσβαση με βάση το προφίλ συμπεριφοράς του χρήστη. Αυτές οι αδυναμίες των παραδοσιακών μεθόδων επαλήθευσης ταυτότητας μπορούν να διορθωθούν με την εισαγωγή βιομετρικών στοιχείων. Η βιομετρία χρησιμοποιεί τα φυσιολογικά και συμπερι-

φορικά χαρακτηριστικά ενός ατόμου, όπως μοτίβα δακτυλικών αποτυπωμάτων, μοτίβα ίριδας, μοτίβο φλεβών κλπ., για να προσδιορίσει τον χρήστη. Δεδομένου ότι αυτά τα χαρακτηριστικά είναι μοναδικά για ένα άτομο, μπορούν να χρησιμοποιηθούν ως ασφαλείς μέθοδοι πιστοποίησης. Τα βιομετρικά χαρακτηριστικά επίσης δεν αλλάζουν με το χρόνο ή την ηλικία, ούτε είναι εύκολο να αναπαραχθούν. Όλες αυτές οι ιδιότητες προσδίδουν στην βιομετρία ένα πλεονέκτημα έναντι της παραδοσιακής μεθόδου αυθεντικοποίησης.

Για παράδειγμα, οι Kashif Habib και λοιποί [14] προτείνουν έναν νέο έλεγχο ταυτότητας, ο οποίος εφαρμόζεται στην υγεία, βασισμένο σε βιομετρικούς τρόπους και μία ασύρματη συσκευή δακτυλικού αποτυπώματος. Ένα σύστημα το οποίο μπορεί να επαληθεύσει ότι τα ελεγχόμενα δεδομένα ανήκουν στον σωστό ασθενή κατά τη διάρκεια ολόκληρης της συνεδρίας, διασφαλίζοντας επίσης την ακεραιότητα και την εμπιστοσύνη των δεδομένων που ληφθήκαν.

Ακόμη, το Blockchain έχει προσελκύσει τεράστια προσοχή ως μια πολλά υποσχόμενη προσέγγιση για τον περιορισμό των κινδύνων ασφαλείας στο IoT λόγω των γνωρισμάτων του που περιλαμβάνουν διαφάνεια, αμεταβλητότητα και αποκεντρωποίση. Ένα Blockchain, είναι μια αυξανόμενη λίστα εγγραφών, που ονομάζονται block τα οποία συνδέονται χρησιμοποιώντας κρυπτογραφία. Κάθε block περιέχει ένα κρυπτογραφικό κατακερματισμό του προηγούμενου block, μια χρονική σήμανση και τα δεδομένα συναλλαγών. Σε ένα IoT σύστημα, όλες οι συναλλαγές, δηλαδή οι επικοινωνίες μεταξύ συσκευών, αποθηκεύονται μόνιμα στο Blockchain. Κάθε block περιλαμβάνει τον κατακερματισμό του προηγούμενου block στο ledger. Αυτό εξασφαλίζει ότι το ledger παραμένει αμετάβλητο [11]. Κατακερματισμός είναι ο μετασχηματισμός μιας συμβολοσειράς χαρακτήρων, σε μια μικρότερη τιμή σταθερού μήκους ή ένα κλειδί που αντιπροσωπεύει την αρχική συμβολοσειρά. Η τροποποίηση ή η κατάργηση των συναλλαγών, είναι αδύνατη, αφού ο κατακερματισμός που διατηρείται στο επόμενο block δεν θα ταιριάζει με τον κατακερματισμό του τροποποιημένου block.



Σχήμα 2.4: Πως λειτουργεί το Blockchain. Μεταφορά χρημάτων μέσω αυτού [3].

Το Blockchain βρίσκει εφαρμογή σε wearables, έξυπνα συστήματα μεταφοράς, έξυπνα σπίτια και πόλεις [12]. Για παράδειγμα, οι συγγραφείς στο [15] προτείνουν ένα σύστημα ασφάλειας για έξυπνη κλειδαριά που βασίζεται σε Blockchain. Με αυτό τον τρόπο διασφαλίζεται η ακεραιότητα των δεδομένων αφού τα δεδομένα που αποστέλλονται και λαμβάνονται από το σύστημα είναι ευάλωτα σε πλαστογράφηση και υποκλοπή. Το Blockchain χρησιμοποιείται για τη διασφάλιση της διαθεσιμότητας και της αμεταβλητότητας των δεδομένων που συλλέγονται από κινητούς αισθητήρες διασκορπισμένους σε μια ευρεία περιοχή, όπως μια αγροτική γη [8], [46]. Στο [42] παρουσιάζεται ένα σύστημα για την παρακολούθηση των κινεζικών αγροτικών ειδών διατροφής. Το σύστημα βασίζεται στη χρήση της αναγνώρισης ραδιοσυχνοτήτων (RFID) και ενός Blockchain στην αλυσίδα εφοδιασμού γεωργικών τροφίμων, που αποσκοπεί στην υποστήριξη των αγορών με σκοπό την αύξηση της ασφάλειας, της ποιότητας των τροφίμων και την μείωση των απωλειών κατά τη διάρκεια της διαδικασίας. Ακόμη, βοηθάει στην πρόληψη κινδύνων μέσω της εποπτείας των κέντρων επιτήρησης. Γενικά, σε αυτό το σύστημα, τα μέλη της αλυσίδας εφοδιασμού βασίζονται σε ένα κέντρο εποπτείας πληροφοριών για την μεταφορά και την κοινή χρήση των πληροφορίων.

3. ΣΧΕΤΙΚΗ ΒΙΒΛΙΟΓΡΑΦΙΑ

Πολλές είναι οι έρευνες που παρουσιάζουν συστήματα για την διασφάλιση των προσωπικών δεδομένων αλλά και για την ασφάλεια των συσκευών.

Στην δημοσίευση ο Naman Gupta και λοιποί [13] αναφέρουν ότι έχει αναπτυχθεί μεγάλη ανησυχία γύρω από την ασφάλεια των ευαίσθητων δεδομένων που στέλνουν οι συσκευές IoT μέσω του διαδικτύου σε μία βάση δεδομένων. Η λύση που παρουσιάζουν είναι η δημιουργία ενός firewall με την χρήση ενός Raspberry Pi ως πύλη που εξασφαλίζει την επικοινωνία με την βάση δεδομένων. Το Raspberry χρησιμοποιείται ως access point με όλες τις συσκευές να συνδέονται σε αυτό και το firewall δημιουργείται στο Raspberry με την χρήση των iptables. Οι κανόνες που χρησιμοποιούνται είναι οι εξής:

- Ενεργοποίηση της IPv4 μεταμφίεση
- Ενεργοποίηση της προστασίας Spoof
- Δεν δέχεται ICMP redirects
- Ενεργοποίηση των TCP/IP SYN cookies
- Ενεργοποίηση της προκαθορισμένης παρακολούθησης σύνδεσης που υποστηρίζεται από τους iptables

Στόχος είναι να υποστηρίξει την παροχή κυκλοφορίας από συσκευές IoT που είναι συνδεδεμένες σε οικιακό δίκτυο, για αυτό το λόγω αναλύθηκε η συμπεριφορά μια έξυπνης κάμερας ασφαλείας. Για την αποφυγή επιθέσεων DDOS αναφέρει την μελλοντική ανάπτυξη μιας έξυπνης υπογραφής και την χρήση πολλαπλών firewalls ή πυλών με έναν εξισορροπιστή φορτίου. Γιατί, όπως αναφέρεται το Raspberry ενδέχεται να μην είναι σε θέση να χειριστεί πολλές συνδέσεις, όπως σε ένα σενάριο DDOS. Ακόμη, για την διασφάλιση της ασφάλειας μελλοντικός στόχος είναι η δημιουργία ενός πίνακα ανίχνευσης κυκλοφορίας που θα λειτουργεί στο Raspberry Pi. Αυτός ο πίνακας ελέγχου θα περιέχει επίσης πληροφορίες σχετικά με τη συχνότητα αποστολής δεδομένων, τη θέση των συσκευών και θα ταξινομεί τις IoT συσκευές με βάση αυτά τα δεδομένα.

Ο Bogdan Jeliskoski και λοιποί υποστηρίζουν [20] ότι η μεγαλύτερη απειλή για τους χρήστες των IoT συσκευών είναι η προστασία των προσωπικών δεδομένων. Δεδομένου

της μεγάλης ανησυχίας ότι το κανάλι επικοινωνίας στην τεχνολογία IoT δεν πραγματοποιείται μόνο μεταξύ του ατόμου και του μηχανήματος, αλλά και μεταξύ των μηχανών κατηγοριοποιούν την ασφάλεια σε τρείς κατηγορίες:

- Ασφάλεια του συστήματος
- Ασφάλεια διαδικτύου
- Ασφάλεια των IoT εφαρμογών

και δίνουν λύση στο πρόβλημα που αφορά την ασφάλεια διαδικτύου που βασίζεται σε μια VPN πύλη. Η πύλη VPN είναι ένας τύπος συσκευής δικτύου που συνδέει δύο ή περισσότερες συσκευές ή δίκτυα σε μια κοινόχρηστη υποδομή. Είναι ένα εικονικό ιδιωτικό δίκτυο που παρέχει ένα ιδιωτικό και κρυπτογραφημένο κανάλι μέσω του οποίου επιτυγχάνεται η επικοινωνία. Η κρυπτογράφηση που χρησιμοποιείται είναι η RSA 2048. Επίσης, Ο διακομιστής VPN χρησιμοποιεί PKI (υποδομή δημόσιου κλειδιού). Υπάρχουν δύο σημαντικά πράγματα που πρέπει να αντιμετωπιστούν για το PKI :

- Δημόσιο κλειδί, (ένα ξεχωριστό πιστοποιητικό) και ένα ιδιωτικό κλειδί για το διακομιστή και τον πελάτη.
- Πιστοποιητικό κύριου πιστοποιητικού αρχής (CA) και κλειδί που χρησιμοποιείται για την υπογραφή πιστοποιητικών για το διακομιστή και τον πελάτη.

Ακόμη, ο διακομιστής υποστηρίζει αμφίδρομη πιστοποίηση, η οποία βασίζεται σε πιστοποιητικά. Ο πελάτης θα πιστοποιηθεί με το πιστοποιητικό του διακομιστή, ενώ ο διακομιστής πρέπει να πιστοποιήσει το πιστοποιητικό του πελάτη. Τα ρολόγια στο διακομιστή και στον πελάτη πρέπει να είναι στον ίδιο συγχρονισμό, διαφορετικά τα πιστοποιητικά δεν θα λειτουργούν.

Το Raspberry Pi καλείται να γίνει η πύλη VPN δίνοντας μία στατική IP σε αυτό και ενεργοποιώντας το NTP (Network Time Protocol) . Επίσης προτείνεται να εγκατασταθεί το dnsmasq για να διασφαλιστεί ότι όλη η κίνηση DNS θα περνάει από το VPN. Σε περίπτωση που το VPN πέσει αναφέρεται πως έχουν δημιουργηθεί κάποιοι κανόνες iptables ώστε καμία συσκευή να μην μπορεί να έχει εξερχόμενη ή εισερχόμενη κίνηση. Τέλος, υποστηρίζουν πως αν συνδεθούν δύο Raspberry θα διασφαλιστεί περισσότερο η ασφάλεια του IoT δικτύου.

Οι συγγραφείς στο [27] μελετώντας τα θέματα ασφαλείας που έχει η τεχνολογία των IoT, παρουσιάζουν διαφορετικούς τύπους επιθέσεων σε IoT που μπορούν να ανιχνευθούν χρησιμοποιώντας ένα απλό εργαλείο όπως το Raspberry Pi που μετατρέπεται σε access point. Εάν μια συσκευή διαπιστωθεί ότι είναι κακόβουλη βάσει ύποπτου μοτίβου δραστηριότητας, η διεύθυνση IP του αποκλείεται και δεν μπορεί πλέον να έχει πρόσβαση στο δίκτυο. Ο χρήστης μπορεί να προβεί σε ενέργειες βάσει των αρχείων καταγραφής που παρέχονται από το πρόγραμμα ανίχνευσης, το οποίο εκτελείται συνεχώς στο δρομολογητή μέσα στο Raspberry. Με αυτόν τον τρόπο, ο χρήστης μπορεί να αποφασίσει να αφήσει την IP αποκλεισμένη έως ότου η συσκευή δεν θεωρείται μολυσμένη ή να την αφαιρέσει από τη μαύρη λίστα σε περίπτωση ψευδούς συναγερμού. Αυτό επιτυγχάνεται μέσω ενός interface.

Στην συνέχεια, αναφέρουν πως μέσω των iptables στο Raspberry Pi τέθηκαν κάποιοι κανόνες για την αποφύγει επιθέσεων αλλά και για τον αποκλεισμό των IP. Έπειτα, παρουσιάζουν ότι η έρευνα των επιθέσεων έχει πραγματοποιηθεί από μια εικονική μηχανή που περιέχει το Kali Linux, τα πακέτα αποθηκεύτηκαν στο Raspberry Pi, και αυτά αργότερα απεικονίστηκαν στο Wireshark για ανάλυση των επαναλαμβανόμενων προτύπων. Βρέθηκε πως οι επιθέσεις από τις οποίες πρέπει να προστατευτεί το IoT δίκτυο είναι:

- Brute Force Attacks
- SQL Injection
- DoS and DDoS Attack

Τέλος, παρουσιάζουν πως για τον εντοπισμό κακόβουλης συμπεριφοράς, χρειάστηκαν δύο προγράμματα Python που τρέχουν παράλληλα: ένα από αυτά παρακολουθεί μη κρυπτογραφημένη δραστηριότητα δικτύου, ανίχνευση επιθέσεων όπως DoS, brute force και SQL injection, ενώ το άλλο σενάριο παρακολουθεί κρυπτογραφημένα πακέτα.

Από την άλλη πολλές δημοσιεύσεις αναφέρονται στην χρήση βιομετρικών χαρακτηριστικών για την ενίσχυση της ασφάλειας έξυπνων περιβαλλόντων και της ιδιωτικότητας των χρηστών.

Ο tanaya και λοιποί [43] παρουσιάζουν μια τεχνική ανίχνευσης προσώπου με σκοπό την ασφάλεια του σπιτιού. Αυτό επιτυγχάνεται με την χρήση του Raspberry Pi μέσω των IoT. Η κύρια ιδέα είναι ότι σε μια βάση δεδομένων αποθηκεύονται τριάντα εικόνες του κάθε χρήστη (μέγιστος αριθμός έξι χρήστες) που είναι εξουσιοδοτημένος. Όταν κάποιος συναντήσει την κάμερα η εικόνα του συγκρίνεται με αυτές από την βάση δεδομένων. Αυτές

οι εικόνες γίνονται ασπρόμαυρες για εξοικονόμηση χώρου. Το Raspberry Pi συνδέεται με ένα laptop και η κάμερα του ανοίγει αυτόματα για να δημιουργηθεί η βάση δεδομένων. Τέλος, αναφέρουν πως η ταυτοποίηση του χρήστη γίνεται μέσω του αλγορίθμου haar και ανάλογα με το αν ο χρήστης είναι εξουσιοδοτημένος ή όχι παίρνεται κάποια απόφαση.

Ακόμη, η ομάδα του πανεπιστημίου Muffakham Jah College of Engineering and Technology [2] παρουσιάζει ένα έξυπνο σύστημα ασφαλείας εισόδου το οποίο βασίζεται σε βιομετρικά χαρακτηρίστηκα του χρήστη και ενσωματώνει την λειτουργικότητα των IoT για την ένδειξη παράνομων χρηστών. Το σύστημα χρησιμοποιεί έναν αισθητήρα αποτυπώματος, έναν ηλεκτρομηχανικό διακόπτη για τον έλεγχο περιστρεφόμενου συστήματος κλειδώματος πόρτας, μία web κάμερα, IoT συσκευές και ένα Raspberry Pi 3. Ο πρώτος χρήστης που πρέπει να εγγραφεί είναι ο διαχειριστής του συστήματος. Το αποτύπωμα του χρήστη που θέλει να μπει στον χώρο συγκρίνεται με αυτά που υπάρχουν ήδη στην βάση δεδομένων, τα οποία αντιστοιχούν σε εξουσιοδοτημένους χρήστες. Αν το αποτύπωμα επαληθευθεί τότε η πόρτα ανοίγει. Επισημαίνουν πως το Raspberry πρέπει να είναι συνδεδεμένο στο διαδίκτυο. Με αυτό τον τρόπο, όταν κάποιος μη εξουσιοδοτημένος χρήστης θελήσει να μπει στον χώρο η κάμερα στέλνει φωτογραφία του στον διαχειριστή μέσω email.

Όπως και στην δημοσίευση [36], παρουσιάζεται ένα έξυπνο σύστημα ασφαλείας εισόδου βασισμένο σε βιομετρικά χαρακτηριστικά, δακτυλικό αποτύπωμα. Το όποιο προτείνεται για την αντικατάσταση των κλειδιών και των καρτών. Για να δημιουργηθεί αυτό το σύστημα χρειάστηκε ένας αισθητήρας δαχτυλικού αποτυπώματος, ένα Arduino, μια ηλεκτρομαγνητική κλειδαριά και μια οθόνη LCD. Η διαδικασία που αναφέρεται είναι η εξής:

- Ο διαχειριστής του συστήματος συνδέεται στο σύστημα βάζοντας το δακτυλικό του αποτύπωμα στον αισθητήρα, ο οποίος είναι συνδεδεμένος στο Arduino , για να εγγράψει τους εξουσιοδοτημένους χρήστες.
- Τα δακτυλικά αποτυπώματα αποθηκεύονται στην βάση δεδομένων
- Ο χρήστης όταν θελήσει να μπει στο χώρο τοποθετεί στον αισθητήρα το δάχτυλο του, αν είναι εξουσιοδοτημένος η πόρτα ξεκλειδώνει αλλιώς όχι, μέσω του Arduino .

Ακόμη, επισημαίνεται πως είναι εφικτή η διαγραφή δαχτυλικού αποτυπώματος.

Οι συγγραφείς στο [38] παρουσιάζουν ακόμη ένα σύστημα ασφάλειας με την χρήση βιομετρικών χαρακτηριστικών που μπορεί να χρησιμοποιηθεί για παράδειγμα για το ξεκλείδωμα μιας πόρτας ή για την είσοδο ενός ατόμου σε ένα κτήριο. Με κύρια διαφορά

από τα προηγούμενα που αναφέρθηκαν, την ενίσχυση της ασφάλειας των βιομετρικών χαρακτηριστικών μέσω κρυπτογράφησης. Χρησιμοποιείται ένα Raspberry ώστε να συνδεθούν σε αυτό μια κάμερα, ένας αισθητήρας κίνησης και ένας αισθητήρας δακτυλικού αποτυπώματος. Σε αυτό επίσης αναφέρεται πως γίνεται η κρυπτογράφηση και ότι στέλνει μέσω του διαδικτύου αυτές τις κρυπτογραφημένες πληροφορίες στο cloud Azure. Όταν ο αισθητήρας κίνησης εντοπίσει κίνηση μία εφαρμογή εγγραφής/σύνδεσης εμφανίζεται ώστε η κάμερα να τραβήξει ένα στιγμιότυπο του ατόμου και ο αισθητήρας δακτυλικού αποτυπώματος την εικόνα του δαχτύλου. Αυτά κρυπτογραφούνται με τον αλγόριθμο AES -256 και αποθηκεύονται στο cloud μαζί με το κρυπτογραφικό κλειδί. Αναφέρεται, επίσης, πως σε περίπτωση σύνδεσης γίνεται η διαδικασία αποκρυπτογράφησης των αποθηκευμένων στοιχείων στο cloud.

Ακόμη μια δημοσίευση της ομάδας του Department of Computer Science, University of Lleida [35] παρουσιάζει ένα σύστημα ασφαλείας πόρτας με την αναγνώριση δακτυλικού αποτυπώματος. Αυτή η εργασία σχετίζεται με την παρούσα και λόγω του αισθητήρα δακτυλικού αποτυπώματος που χρησιμοποιείται ο οποίος είναι ο GT(511C1R). Η εφαρμογή αυτή είναι ένα παράδειγμα πελάτη-διακομιστή. Ο διακομιστής είναι το Raspberry Pi με τον αισθητήρα, έναν διακόπτη και μια ηλεκτρομαγνητική κλειδαριά να είναι συνδεδεμένα σε αυτό. Οι πελάτες μπορεί να εκτελούνται σε περιηγητές, ένας διαχειριστής μπορεί να διαχειριστεί το σύστημα από μια διαδικτυακή εφαρμογή ως πελάτης. Ο διαχειριστής αναφέρεται ότι μπορεί να εγγράψει, να διαγράψει και να ενημερώσει τους χρήστες. Επιπλέον, ο διαχειριστής, μέσω μιας ιστοσελίδας, στέλνει δεδομένα χρήστη στον διακομιστή, ο οποίος αποθηκεύει και αλληλοεπιδράει με έναν αισθητήρα δακτυλικού αποτυπώματος για να εκτελέσει την ενέργεια που ζητήθηκε από τον πελάτη. Εάν ένας εγγεγραμμένος χρήστης θέλει να ξεκλειδώσει την κλειδαριά, πρέπει να πατήσει έναν διακόπτη για να ενεργοποιήσει τον αισθητήρα του δακτυλικού αποτυπώματος. Εάν το αποτύπωμα είναι έγκυρο, ο διακομιστής ξεκλειδώνει τη μαγνητική κλειδαριά. Διαφορετικά, το μαγνητικό κλείδωμα παραμένει κλειδωμένο (ενέργεια από προεπιλογή). Τέλος, αναφέρει πως η επικοινωνία πελάτη-διακομιστή γίνεται μέσω HTTP μεθόδων.

Στο σύστημα που υλοποιήσαμε χρησιμοποιήθηκε συνδυασμός τεχνικών, που αναφέρθηκαν στην παραπάνω σχετική βιβλιογραφία, για την ενίσχυση της ιδιωτικότητας του χρήστη σε ένα κοινόχρηστο IoT περιβάλλον. Οι τεχνικές που χρησιμοποιήθηκαν είναι οι εξής:

- firewall μέσω του πακέτου python-iptables

- διαδικτυακή εφαρμογή μέσω του flask
- λειτουργία του Raspberry Pi ως access point
- χρήση βιομετρικού χαρακτηριστικού

Στις δημοσιεύσεις [13], [27] χρησιμοποιούνται κάποιες από τις τεχνικές που υλοποιήσαμε όμως δεν διασφαλίζεται άμεσα η προστασία των προσωπικών στοιχείων αλλά μέσω της ασφάλειας των συσκευών. Οι δημοσιεύσεις που αναφέρουν χρήση βιομετρικού χαρακτηριστικού [43], [36], [2], [38], [35] χρησιμοποιείται ως κάρτα ή κλειδί για την δημιουργία ενός ασφαλέστερου περιβάλλοντος και την αποτροπή εισόδου σε μη εξουσιοδοτημένο χρήστη. Στη δική μας περίπτωση το δακτυλικό αποτύπωμα καθορίζει την λειτουργία των συσκευών.

4. ΠΡΟΤΕΙΝΟΜΕΝΟ ΣΥΣΤΗΜΑ

Στόχος της παρούσας εργασίας είναι η υλοποίηση ενός συστήματος ρύθμισης της λειτουργίας των IoT συσκευών σε ένα κοινόχρηστο περιβάλλον, με την χρήση βιομετρικών χαρακτηριστικών του χρήστη. Σκοπός είναι η δυναμική αλλαγή της λειτουργίας των συσκευών κάθε φορά που αλλάζουν οι χρήστες του κοινόχρηστου χώρου. Το σύστημα αυτό αποτελείται από ένα Raspberry Pi 3, έναν αισθητήρα δακτυλικού αποτυπώματος, μια διαδικτυακή εφαρμογή (web application) με την αντίστοιχη βάση δεδομένων, και τέλος έναν QR κωδικού. Ο κωδικός QR είναι μια οπτική ετικέτα αναγνώσιμη από μηχανή που περιέχει πληροφορίες σχετικά με το αντικείμενο στο οποίο είναι συνημμένο. Στην πράξη, οι κωδικοί QR περιέχουν συχνά δεδομένα που οδηγούν σε ιστοσελίδα ή εφαρμογή.

Αρχικά, ο χρήστης χρειάζεται να σαρώσει το QR για να μεταβεί στην διαδικτυακή εφαρμογή στην οποία θα εγγραφεί. Για να πραγματοποιηθεί η εγγραφή απαιτείται η εισαγωγή του ονόματος του χρήστη και του κωδικού πρόσβασης. Ο χρήστης καλείται να επιλέξει τους περιορισμούς που θα ακολουθούν οι IoT συσκευές που βρίσκονται στον χώρο. Οι συσκευές έχουν χωριστεί σε τέσσερις κατηγορίες: εικόνα, ήχο, κίνηση, περιβάλλον. Ο χρήστης ορίζει, μέσω της εφαρμογής, τον τρόπο που επιθυμεί να λειτουργούν οι συσκευές κάθε κατηγορίας όταν βρίσκεται στον κοινόχρηστο χώρο. Οι περιορισμοί που μπορεί να θέσει είναι οι εξής τρεις:

- οι συσκευές να είναι τελείως κλειστές,
- να στέλνουν δεδομένα σε μια συγκεκριμένη IP εμπιστοσύνης, ή
- να έχουν ελεγχόμενο ρυθμό μετάδοσης δεδομένων.

Στην συνέχεια, ακολουθεί η καταχώρηση του δακτυλικού αποτυπώματος του χρήστη. Κάθε φορά που ο χρήστης εισέρχεται ή εξέρχεται από τον χώρο, με την χρήση του δακτυλικού αποτυπώματος, δομείται ένας γενικός κανόνας που καθορίζει την λειτουργία των συσκευών. Ο γενικός κανόνα παράγεται από τους περιορισμούς των ατόμων που βρίσκονται εκείνη την στιγμή στο κοινόχρηστο περιβάλλον.

4.1 Υλικό και λογισμικό

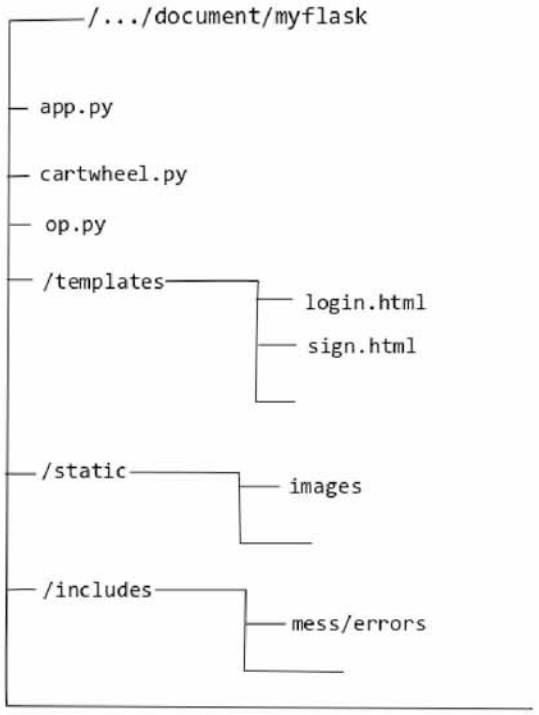
4.1.1 Flask

Το Flask είναι ένα πακέτο της γλώσσας προγραμματισμού Python το οποίο είναι ένα διαδικτυακό πλαίσιο (web microframework). Το Flask δεν παρέχει έτοιμη βάση δεδομένων ή form validation, όπου υπάρχουν ήδη διαφορετικές βιβλιοθήκες που μπορούν να το χειριστούν. Όμως, υποστηρίζει επεκτάσεις για να προσθέσει τέτοια λειτουργικότητα στην εφαρμογή.

Σχήμα 4.1: Ενεργοποίηση του Flask σε λογισμικό Windows.

Σχήμα 4.2: Ενεργοποίηση του Flask σε λογισμικό Raspbian.

Το γεγονός ότι το Flask είναι microframework δεν συνεπάγει ότι το web application πρέπει να χωρέσει σε έναν απλό φάκελο Python. Συνήθως, τα πρότυπα (templates) και τα στατικά (static) αρχεία αποθηκεύονται σε υποκαταλόγους μέσα στο δένδρο φακέλων της εφαρμογής, οι οποίοι ονομάζονται templates και static αντίστοιχα. Κάθε κώδικας της εφαρμογής βρίσκεται σε ένα συγκεκριμένο directory (myflask).



Σχήμα 4.3: Δένδρο φακέλων εφαρμογής. app.py: flask, cartwheel.py και op.py: κώδικες για το δακτυλικό αποτύπωμα, /templates: τα HTML αρχεία της εφαρμογής, /static: οι εικόνες, /includes: μηνύματα errors

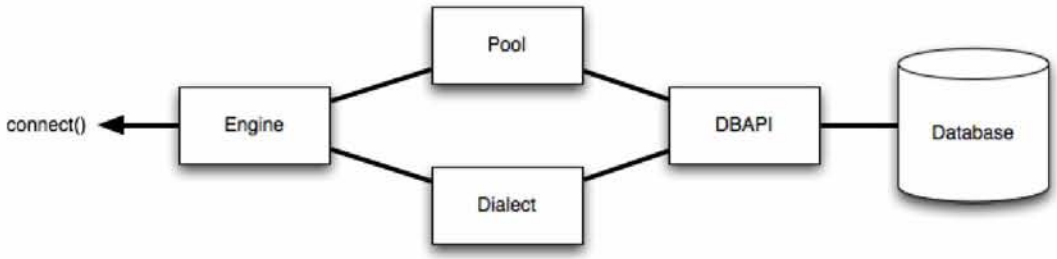
Όταν εγκατασταθεί το Flask μαζί του εγκαθίσταται και η εντολή flask. Η εντολή flask run προτρέπει το εικονικό περιβάλλον του πακέτου Flask να τρέχει σε έναν HTTP server χρησιμοποιώντας ένα αντικείμενο app.py. Αυτό το αντικείμενο λειτουργεί ως αντικείμενο διαμόρφωσης όλης της εφαρμογής και χρησιμοποιείται για να ορίσει τα σημεία αλληλεπίδρασης της εφαρμογής. Τα σημεία αλληλεπίδρασης διαμορφώνονται μέσω του app.route ώστε να οριστούν ως λειτουργικά τμήματα της εφαρμογής. Το app.route είναι μια συνάρτηση decoration. Αυτές οι συναρτήσεις χρησιμοποιούνται στην Python για να μεταμορφώσουν άλλες συναρτήσεις [33]. Δηλαδή, το app.route χρησιμοποιείται για να αντιστοιχίσει URLs σε view συναρτήσεις (Σχήμα 4.4).

```python
app = Flask(__name__)
#app is an instance of Flask, taking in the __name__ of the script file

@app.route('/')
def index():
    return render_template('index1.html')
    #user will receive the content from index1.html as a response
```

Σχήμα 4.4: Αντικείμενο app και παράδειγμα συνάρτησης decoration.

Ενώ το Flask δεν υποστηρίζει ενσωματωμένη βάση δεδομένων, το πακέτο flask-sqlalchemy συνδέει μια SQL βάση δεδομένων στην εφαρμογή. Επίσης, διαχειρίζεται τα εισερχόμενα δεδομένα αιτήσεων του χρήστη μέσω του πακέτου request.

Η σύνδεση της εφαρμογής με την βάση δεδομένων γίνεται με τη διαμόρφωση engine. Το αντικείμενο engine περιγράφει πώς επικοινωνεί η SQLAlchemy με την βάση δεδομένων (Σχήμα 4.5). Η σύνδεση αυτή πραγματοποιείται μέσω ενός Pool και ενός Dialect. Το αντικείμενο Dialect ορίζει τη συμπεριφορά συγκεκριμένης βάσης δεδομένων, της δημιουργίας ερωτημάτων SQL, της εκτέλεσης, και του χειρισμού των αποτελεσμάτων. Το αντικείμενο Pool εγκαθιστά την σύνδεση DBAPI όταν υπάρξει αίτημα σύνδεσης. Τα αντικείμενα Pool και Dialect μαζί ερμηνεύουν τις λειτουργίες της μονάδας του DBAPI καθώς και τη συμπεριφορά της βάσης δεδομένων. Το DBAPI είναι ένα API χαμηλού επιπέδου, το οποίο είναι συνήθως το σύστημα που χρησιμοποιείται σε εφαρμογές της Python για να επικοινωνήσει με μια βάση δεδομένων.



Σχήμα 4.5: Σύνδεση της βάσης δεδομένων με την εφαρμογή μέσω της διαμόρφωσης engine [10].

Η συνάρτηση create_engine() παράγει ένα αντικείμενο engine βασισμένο σε μια διεύθυνση URL. Αυτές οι διευθύνσεις περιλαμβάνουν όνομα χρήστη, κωδικό πρόσβασης, όνομα κεντρικού υπολογιστή, όνομα βάσης δεδομένων, και προαιρετικές λέξεις-κλειδιά για επιπλέον ρυθμίσεις.

Η φόρμα ενός τέτοιου URL είναι dialect+driver://username:password@host:port/database (Σχήμα 4.6).

Για να εκτελέσουμε μια εντολή SQL χρειάζεται να χρησιμοποιήσουμε το αντικείμενο Session του Flask (Σχήμα 4.7). Αυτό, καθορίζει όλες τις "συνομιλίες" με τη βάση δεδομένων και αποθηκεύει όλα τα αντικείμενα που έχει φορτωθεί ή συνδεθεί σε αυτό. Παρέχει το σημείο εισόδου για την απόκτηση ενός αντικειμένου Query που αποστέλλει ερωτήματα στη βάση δεδομένων, χρησιμοποιώντας την τρέχουσα σύνδεση βάσης δεδομένων του Session. Τοποθετεί τα αποτελέσματα των ερωτημάτων σε αντικείμενα που στη συνέχεια αποθηκεύονται στο Session, μέσα στη δομή που ονομάζεται Map Identity. Αυτή, διατηρεί μοναδικά αντίγραφα κάθε αντικειμένου, όπου "μοναδικό" σημαίνει "μόνο ένα αντικείμενο με ένα συγκεκριμένο πρωτεύον κλειδί". Το Session δημιουργείται μέσω της κλάσης Sessionmaker (Σχήμα 4.6). Αυτό, επιτρέπει τη χρήση του σε όλη την εφαρμογή, χωρίς να χρειάζεται επανάληψη της διαμόρφωσης των κανόνων.

```
engine = create_engine("mysql+pymysql://root:1080tsayo@localhost/register")
#MySQL dialect uses mysql-python as the default DBAPI
db=scoped_session(sessionmaker(bind=engine))
#the sessionmaker call creates a factory
#when called will create a new db object
#using the configurational arguments we've given the factory
```

Σχήμα 4.6: Σύνδεση της βάσης δεδομένων και δημιουργία αντικειμένου Session μέσω της κλάσης Sessionmaker.

```
countrows = db.execute("SELECT * from (select COUNT(*) from users where name=(:name)) AS T1",{"name":name})
for x in countrows:
    if x[0]==0:
        db.execute("INSERT INTO users(name,password) VALUES (:name,:password)", {"name":name, "password":secpass})
        db.execute("INSERT INTO present_u(present) VALUES('false')")
        db.commit()
```

Σχήμα 4.7: Εκτέλεση εντολών SQL μέσω του αντικειμένου db.

Επίσης, το Flask υποστηρίζει μεθόδους HTTP για την επικοινωνία των δεδομένων. Το HTTP είναι το πρωτόκολλο μεταφοράς υπερκειμένου, το οποίο θεωρείται θεμέλιο της μεταφοράς δεδομένων στον παγκόσμιο ιστό. Διαφορετικές μέθοδοι ανάκτησης δεδομένων από συγκεκριμένες διευθύνσεις URL ορίζονται σε αυτό το πρωτόκολλο. Στο Σχήμα 4.8 περιγράφονται κάποιες μέθοδοι.

FLASK HTTP METHODS

Μέθοδος	Περιγραφή
GET	Χρησιμοποιείται για να επιστρέψει έναν συγκεκριμένο πόρο
POST	Χρησιμοποιείται για να δημιουργήσει νέα δεδομένα σε έναν συγκεκριμένο πόρο
PUT	Χρησιμοποιείται για να δημιουργήσει καινούργια δεδομένα ή να αντικαστήσει τα ήδη υπάρχοντα σε έναν συγκεκριμένο πόρο
DELETE	Χρησιμοποιείται για να διαγράψει υπάρχοντα δεδομένα σε έναν συγκεκριμένο πόρο
PATCH	Χρησιμοποιείται για να ενημερώσει ή να τροποιήσει δεδομένα σε έναν πόρο

Σχήμα 4.8: 5 βασικές HTTP μέθοδοι που υποστηρίζει το Flask.

Το Flask route, από προεπιλογή, απαντάει μόνο σε αιτήματα GET, τα οποία είναι και τα πιο συνηθισμένα. Όμως, αυτή η συμπεριφορά μπορεί να μεταβληθεί παρέχοντας την μεταβλητή methods στην συνάρτηση decorator route. Ένα κοινό παράδειγμα για την χρήση του αιτήματος GET είναι η επιστροφή μιας σελίδας HTML (Σχήμα 4.9).

```
app = Flask(__name__)
#app is an instance of Flask, taking in the __name__ of the script file

@app.route('/')
def index():
    return render_template('index1.html')
    #user will receive the content from index1.html as a response
```

Σχήμα 4.9: Απάντηση στο αίτημα GET. Επιστροφή της σελίδας index1.html.

Για την διαχείριση των αιτημάτων POST, χρειάζεται πρώτα να δημιουργηθεί μια φόρμα για την λήψη ορισμένων στοιχείων του χρήστη (Σχήμα 4.10). Με αυτό τον τρόπο υπάρχει πρόσβαση σε αυτά τα δεδομένα του server χρησιμοποιώντας τα αιτήματα POST (Σχήμα 4.11). Τα αιτήματα αυτά θα πρέπει να χρησιμοποιούνται για τη δημιουργία νέων πόρων, για παράδειγμα νέοι χρήστες.

```
<form action="" method="POST">
  <div class="form-group">
    <input type="text" name="name" class="form-control" placeholder="name" required>
    </div>
    <div class="form-group">
      <input type="password" name="password" class="form-control" placeholder="password" required>
      </div>
      <div class="form-group">
        <input type="password" name="confirm" class="form-control" placeholder="confirm password" required>
        </div>
```

Σχήμα 4.10: Διαχείριση αιτήματος POST. Φόρμα λήψης δεδομένων.

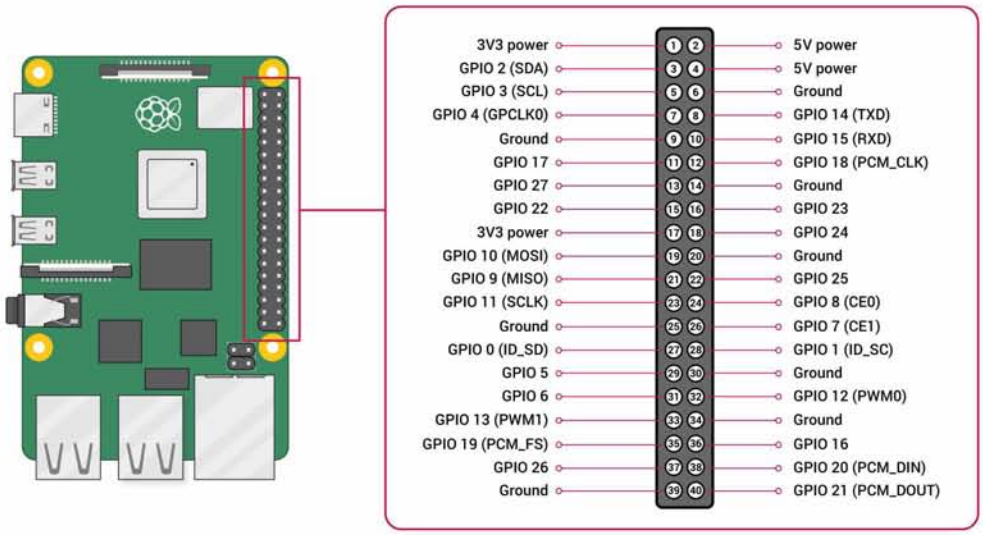
```
@app.route('/sign', methods=["GET","POST"])
def sign():
    if request.method == "POST":
        name = request.form.get("name")
        password = request.form.get("password")
        confirm = request.form.get("confirm")
        secpass= sha256_crypt.encrypt(str(password))
```

Σχήμα 4.11: Λήψη δεδομένων από την εφαρμογή.

Το request() είναι ένα αντικείμενο το οποίο είναι διαθέσιμο σε κάθε route και περιλαμβάνει τα περιεχόμενα των αιτημάτων που γίνονται σε ένα συγκεκριμένο route. Το request.method περιέχει την μέθοδο που χρησιμοποιείται για την πρόσβαση στο route, όπως οι μέθοδοι POST και GET. Το request.form είναι ο τρόπος πρόσβασης στην πληροφορία της φόρμας (Σχήμα 4.11). Αν το κλειδί (π.χ. name) δεν υπάρχει, χρησιμοποιείται το request.form.get.

4.1.2 Raspberry

Το Raspberry Pi 3 b είναι ένας πολύ μικρός υπολογιστής (μεγέθους πιστωτικής κάρτας) με δυνατότητα ασύρματης σύνδεσης LAN . Συνδέεται σε οθόνη υπολογιστή, πληκτρολόγιο και ποντίκι μέσω USB. Αυτή η συσκευή χρησιμοποιεί το Raspbian, το οποίο είναι ένα λειτουργικό σύστημα υπολογιστή με βάση το Debian ειδικά σχεδιασμένο για το Raspberry Pi. Έχει κεντρική μονάδα επεξεργασίας με 1.2GH και 64/32-bit τετραπύρηνο ARM Cortex-A53. Το οποίο έχει εσωτερική μνήμη 1 GB. Το Raspberry Pi 3 b αποτελείται από RAM, I / O, CPU / GPU, USB hube, Ethernet,2x USB, θύρα HDMI και υποδοχή κάρτας μνήμης. Επίσης, παρέχει θύρες για σύνδεση κάμερας και οθόνης αφής, σαράντα pin που επεκτείνονται σε GPIO [27],[2].



Σχήμα 4.12: Απεικόνιση του Raspberry Pi και ανάλυση των pin.

4.1.3 Fingerprint

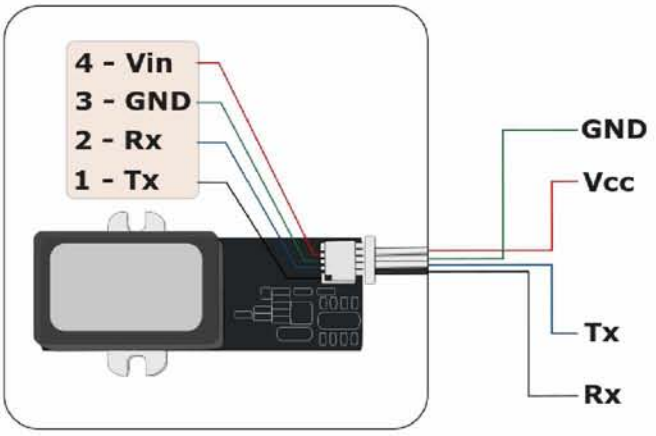
Η συσκευή που χρησιμοποιήθηκε για την οπτική αναγνώριση δακτυλικών αποτυπωμάτων είναι η Fingerprint Scanner TTL GT511C1R [35], [4]. Αποτελείται από έναν οπτικό αισθητήρα και ένα τσιπ με αλγόριθμο δακτυλικού αποτυπώματος. Έχει την δυνατότητα να εγγράφει ένα δακτυλικό αποτύπωμα και να το αναγνωρίζει με δυνατότητα αναγνώρισης 360 μοίρες.

Βασικό χαρακτηριστικό είναι η τεχνολογία υψηλής ακρίβειας και ταχύτητας για την αναγνώριση δακτυλικών αποτυπωμάτων. Επίσης, μπορεί να κατεβάζει εικόνες δακτυλικού αποτυπώματος από την συσκευή αλλά και να γράφει και να διαβάζει δείγματα από και προς αυτήν. Ο μέγιστος αριθμός αποτυπωμάτων που έχει δυνατότητα να αποθηκεύσει η συσκευή είναι είκοσι, από την θέση μηδέν ως την δεκαεννιά. Για να πραγματοποιηθεί η διαδικασία εγγραφής ενός δακτυλικού αποτυπώματος, χρειάζεται να δημιουργηθούν τρία δείγματα. Ο χρήστης πρέπει να τοποθετήσει το δάχτυλο του στον αισθητήρα τρείς φορές. Αυτά τα τρία δείγματα είναι οι τρείς εικόνες του δακτυλικού αποτυπώματος του χρήστη. Όταν δημιουργηθούν, ενώνονται, και αποθηκεύονται σαν ένα. Ακόμη, υποστηρίζει λειτουργία αντιστοίχισης, 1:1 επαλήθευση και 1:N ταυτοποίηση. Στην πρώτη λειτουργία, αντιστοιχεί το αποτύπωμα με ένα άλλο και επαληθεύει αν είναι το ίδιο. Ενώ στην δεύτερη, η αντιστοίχιση γίνεται με μια λίστα αποτυπωμάτων προσπαθώντας να εξακριβώσει αν το

αποτύπωμα υπάρχει.

Η επικοινωνία της συσκευής γίνεται σειριακά με UART πρωτόκολλο και USB. Στην παρούσα εργασία χρησιμοποιείται η UART επικοινωνία [28]. Στην επικοινωνία αυτή, δύο UARTs επικοινωνούν απευθείας μεταξύ τους. Ο UART μετάδοσης μετατρέπει τα παράλληλα δεδομένα από μια συσκευή ελέγχου, όπως μια CPU, σε σειριακή μορφή, τα μεταδίδει σε σειρά στον UART λήψης, ο οποίος στη συνέχεια μετατρέπει τα σειριακά δεδομένα πίσω σε παράλληλα δεδομένα για τη συσκευή λήψης. Απαιτούνται μόνο δύο καλώδια για τη μετάδοση δεδομένων μεταξύ δύο UART. Δεδομένα ρέουν από τον ακροδέκτη Tx του UART μετάδοσης προς τον ακροδέκτη Rx του UART λήψης.

Τα UART μεταδίδουν δεδομένα ασύγχρονα. Αυτό σημαίνει, ότι δεν υπάρχει σήμα ρολογιού για το συγχρονισμό της εξόδου των δυαδικών ψηφίων από το UART μετάδοσης προς τη δειγματοληψία των δυαδικών ψηφίων από το UART λήψης. Αντί ενός σήματος ρολογιού, το UART μετάδοσης προσθέτει bits εκκίνησης και τερματισμού στο μεταφερόμενο πακέτο δεδομένων. Αυτά τα δυαδικά ψηφία ορίζουν την αρχή και το τέλος του πακέτου δεδομένων, έτσι ώστε το UART λήψης να γνωρίζει πότε θα αρχίσει να διαβάζει τα δυαδικά ψηφία. Όταν ο UART λήψης εντοπίσει ένα bit έναρξης, αρχίζει να διαβάζει τα εισερχόμενα bits σε μια συγκεκριμένη συχνότητα, γνωστή ως ρυθμός baud. Ο ρυθμός αυτός είναι ένα μέτρο της ταχύτητας μεταφοράς δεδομένων, εκφρασμένο σε δυαδικά ψηφία ανά δευτερόλεπτο (bps). Και τα δύο UARTs πρέπει να λειτουργούν περίπου στον ίδιο ρυθμό baud.



Σχήμα 4.13: Απεικόνιση συσκευής δακτυλικού αποτυπώματος. Ανάλυση των άκρων της.

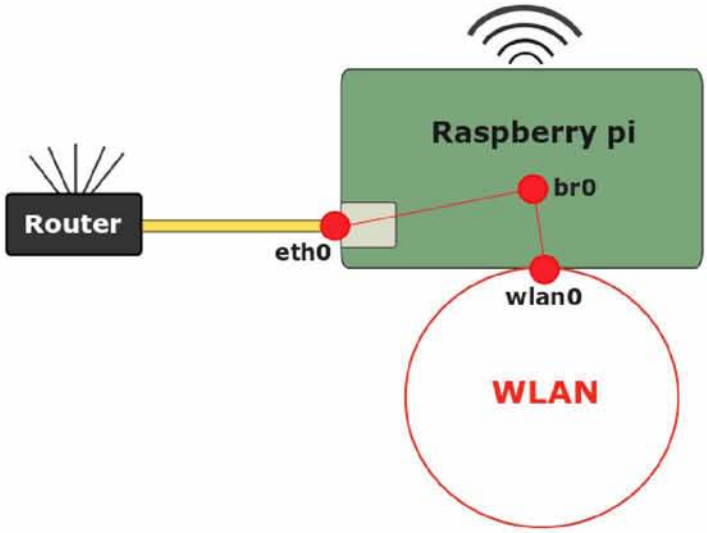
4.1.4 Access point

Μια κοινή χρήση του Raspberry Pi είναι ως access point, δηλαδή η παροχή ασύρματων συνδέσεων σε μια ενσύρματη σύνδεση Ethernet. Έτσι, όποιος συνδεθεί στο access point αποκτά πρόσβαση στο διαδίκτυο, με προϋπόθεση ότι το ενσύρματο Ethernet στο Pi μπορεί να συνδεθεί στο διαδίκτυο μέσω ενός router.

Το router είναι μια συσκευή δικτύου που εξυπηρετεί δύο κύριες λειτουργίες. Πρώτον, συνδέει πολλούς υπολογιστές, τηλέφωνα, ή άλλες συσκευές, και διαμορφώνει ένα διαχειριζόμενο τοπικό δίκτυο. Δεύτερον, παρέχει πρόσβαση στο διαδίκτυο σε όλες τις συμβατές συσκευές που συνδέονται στο δρομολογητή. Ένα τοπικό δίκτυο (LAN) μπορεί να ρυθμιστεί απλά χρησιμοποιώντας ένα router και συνδέοντας μία ή περισσότερες συσκευές σε αυτό. Οι σύγχρονες συσκευές επιτρέπουν στους χρήστες να συνδέουν συσκευές τόσο μέσω καλωδίων Ethernet είτε ασύρματα (χρησιμοποιώντας Wi-Fi).

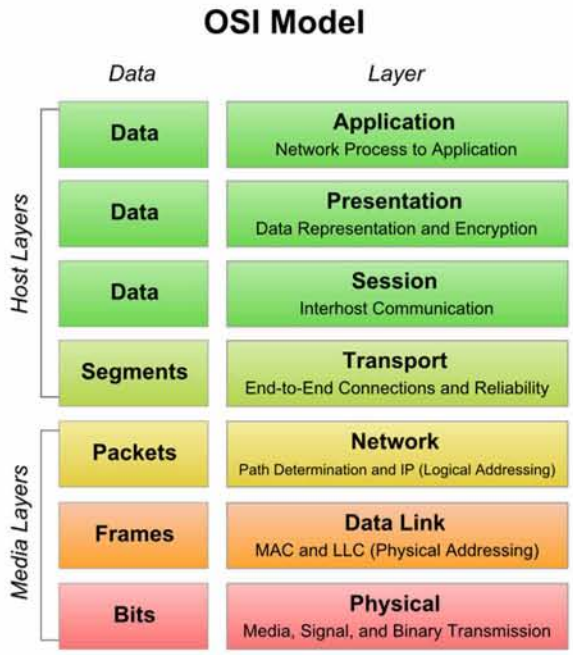
Το access point είναι μια συσκευή ασύρματου δικτύου που λειτουργεί ως πύλη για τη σύνδεση συσκευών σε τοπικό δίκτυο. Τα access point χρησιμοποιούνται για την επέκταση της ασύρματης κάλυψης ενός υπάρχοντος δικτύου και για την αύξηση του αριθμού των χρηστών που μπορούν να συνδεθούν με αυτό. Ένα καλώδιο Ethernet υψηλής ταχύτητας συνδέει ένα router σε ένα access point, το οποίο μετατρέπει το ενσύρματο σήμα σε ασύρματο. Η ασύρματη συνδεσιμότητα είναι συνήθως η μόνη διαθέσιμη επιλογή για access point, δημιουργώντας συνδέσμους με τελικές συσκευές που χρησιμοποιούν Wi-Fi.

Για να μπορέσει το Raspberry να έχει αυτή την χρήση, πρέπει να δημιουργηθεί μια γέφυρα (bridge) μεταξύ της ασύρματης συσκευής και της συσκευής Ethernet στο σημείο πρόσβασης Raspberry Pi. Αυτή η γέφυρα θα περάσει όλη την κίνηση μεταξύ των δύο διεπαφών. Ο στόχος της χρήσης του access point είναι να ελέγχεται η μεταφορά δεδομένων μέσω της γέφυρας.



Σχήμα 4.14: Απεικόνιση λειτουργίας του Raspberry Pi ως access point.

Μια γέφυρα μπορεί να χρησιμοποιηθεί σε δίκτυα υπολογιστών για τη διασύνδεση δύο τοπικών δικτύων και ξεχωριστών τμημάτων δικτύου. Η γέφυρα είναι μια συσκευή στρώματος δύο στο μοντέλο OSI (Σχήμα 4.15). Το OSI είναι ένα εννοιολογικό πλαίσιο που προσδιορίζει, καθορίζει και ρυθμίζει τις λειτουργίες επικοινωνίας ενός υπολογιστικού συστήματος, λαμβάνοντας υπόψη την εσωτερική δομή και την τεχνολογία. Εφόσον η γέφυρα ανήκει στο δεύτερο στρώμα σημαίνει ότι χρησιμοποιεί τις πληροφορίες διεύθυνσης MAC για τη λήψη αποφάσεων σχετικά με την προώθηση των πακέτων δεδομένων. Μόνο τα δεδομένα που πρέπει να αποσταλούν μέσω της γέφυρας στο γειτονικό τμήμα δικτύου διαβιβάζονται. Αυτό καθιστά δυνατή την απομόνωση ή την κατανομή της κυκλοφορίας των δεδομένων του δικτύου [19].



Σχήμα 4.15: Μοντέλο OSI [6].

4.1.5 Iptables

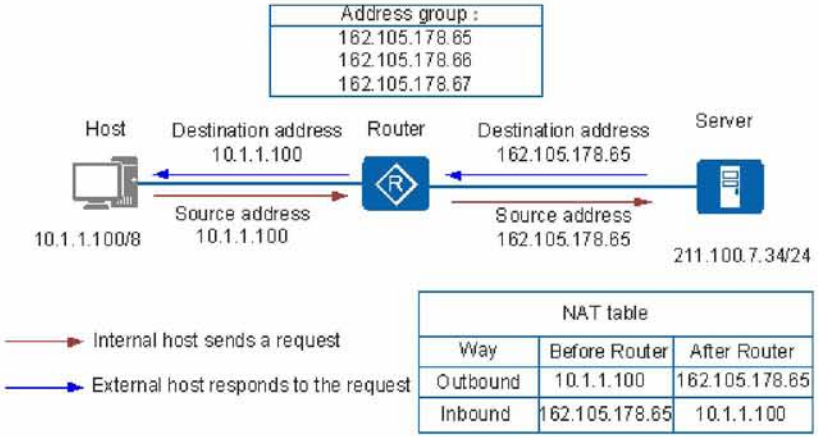
Το iptables είναι ένα εργαλείο για τη διαχείριση των κανόνων του τείχους προστασίας (firewall) σε μια μηχανή Linux. Το firewall είναι ένα σύστημα ασφάλειας δικτύου που έχει σχεδιαστεί για να εμποδίζει μη εξουσιοδοτημένη πρόσβαση σε, ή από, ένα ιδιωτικό δίκτυο. Αποτρέπουν την πρόσβαση μη εξουσιοδοτημένων χρηστών του διαδικτύου σε ιδιωτικά δίκτυα που είναι συνδεδεμένα σε αυτό. Όλα τα πακέτα πληροφορίας που εισέρχονται ή εξέρχονται από αυτό περνούν από το firewall, το οποίο εξετάζει κάθε πακέτο και αποκλείει αυτά που δεν πληρούν τα καθορισμένα κριτήρια ασφαλείας. Τα περισσότερα κριτήρια περιλαμβάνουν πληροφορίες σχετικά με την προέλευση, τον προορισμό και το πρωτόκολλο που σκοπεύει να χρησιμοποιήσει το πακέτο.

Γενικά, ένα σύνολο κανόνων iptables επεξεργάζεται από τον πυρήνα του Linux για κάθε πακέτο συγκριτικά με ένα πρόγραμμα παρτίδας. Οι κανόνες αξιολογούνται διαδοχικά, αλλά η ενέργεια (μερικές φορές ονομάζεται στόχος) εφαρμόζεται μόνο εάν το πακέτο ταιριάζει με τα κριτήρια του κανόνα. Τελικά, ο πυρήνας του Linux πρέπει να καθορίσει εάν θα γίνει το πακέτο DROP ή ACCEPT, επομένως, αυτές είναι οι κοινές ενέργειες. Άλλη πιθανή ενέργεια είναι το άλμα σε άλλες αλυσίδες (JUMP όνομα αλυσίδας), και τη συνέχιση της επεξεργασίας των πακέτων από εκεί [9].

Από προεπιλογή, υπάρχουν τρεις πίνακες στον πυρήνα (filter table, nat table και mangle table) που περιέχουν σύνολα κανόνων. Μια λίστα κανόνων αποκαλείται αλυσίδα (chain). Ο filter table χρησιμοποιείται για το φιλτράρισμα των πακέτων, ο nat table για μετάφραση διευθύνσεων και, τέλος, ο mangle table μπορεί να χρησιμοποιηθεί για ειδική επεξεργασία πακέτων [39].

Στον filter table περιέχονται οι κανόνες που καθορίζουν αν ένα πακέτο θα συνεχίσει για τον τελικό του προορισμό. Σε αυτό τον πίνακα υπάρχουν τρείς αλυσίδες: η INPUT, η FORWARD και η OUTPUT. Όλα τα πακέτα που εισέρχονται στο ιδιωτικό δίκτυο ελέγχονται από τους κανόνες που βρίσκονται στην αλυσίδα INPUT, ενώ τα πακέτα που εξέρχονται από την OUTPUT. Η αλυσίδα FORWARD χρησιμοποιείται για εισερχόμενα πακέτα που δεν προορίζονται τοπικά στην συσκευή. Για παράδειγμα, σε έναν δρομολογητή, τα δεδομένα στέλνονται πάντα σε αυτόν, όμως σπάνια προορίζονται για τον ίδιο. Η αλυσίδα FORWARD, λοιπόν, ελέγχει τα πακέτα που ο δρομολογητής απλώς δρομολογεί.

Ο πίνακας NAT χρησιμοποιείται για την εφαρμογή κανόνων μετάφρασης διεύθυνσης δικτύου. Καθώς τα πακέτα εισέρχονται στο δίκτυο, οι κανόνες στον πίνακα NAT καθορίζουν εάν, και πώς, θα τροποποιηθούν οι διευθύνσεις προέλευσης ή προορισμού του πακέτου, προκειμένου να επηρεαστεί ο τρόπος με τον οποίο δρομολογείται το πακέτο και οποιαδήποτε κίνηση απόκρισης.

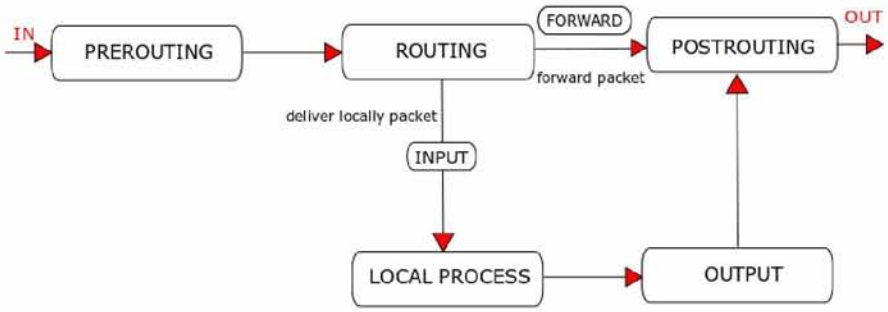


Σχήμα 4.16: Λειτουργία του NAT πίνακα [17].

Οι αλυσίδες του NAT είναι οι PREROUTING, POSTROUTING, OUTPUT. Η αλυσίδα PREROUTING είναι υπεύθυνη για τα πακέτα που μόλις έφτασαν στη διασύνδεση του δικτύου. Μέχρι στιγμής δεν έχει ληφθεί απόφαση δρομολόγησης, επομένως δεν είναι ακόμα γνωστό αν το πακέτο προορίζεται για την συσκευή, ή αν θα δρομολογηθεί σε άλλο μηχάνημα που βρίσκεται σε άλλο δίκτυο. Αφού το πακέτο έχει περάσει από την αλυ-

σίδα PREROUTING, γίνεται η απόφαση δρομολόγησης. Σε περίπτωση που το τοπικό μηχάνημα είναι ο παραλήπτης, το πακέτο θα κατευθυνθεί στην αντίστοιχη διαδικασία. Σε περίπτωση που ο παραλήπτης βρίσκεται σε διαφορετικό δίκτυο, το πακέτο θα προωθηθεί σε αυτό, με την προϋπόθεση ότι το μηχάνημα έχει ρυθμιστεί να το κάνει. Λίγο πριν φύγει το πακέτο από το μηχάνημα, περνάει από την αλυσίδα POSTROUTING και στη συνέχεια αφήνει το δίκτυο. Τα πακέτα που δημιουργούνται τοπικά δε διέρχονται από την αλυσίδα PREROUTING, αλλά από την αλυσίδα OUTPUT, και μετά μετακινούνται στην αλυσίδα POSTROUTING ώστε να φύγουν από το δίκτυο [40].
Η διέλευση των διαφορετικών πακέτων γίνεται με τους εξής τρόπους:

- εισερχόμενα πακέτα που προορίζονται για το τοπικό σύστημα
  PREROUTING -> INPUT
- πακέτα που προορίζονται για άλλο σύστημα
  PREROUTING -> FORWARD -> POSTROUTING
- πακέτα που παράγοντα τοπικά
  OUTPUT -> POSTROUTING

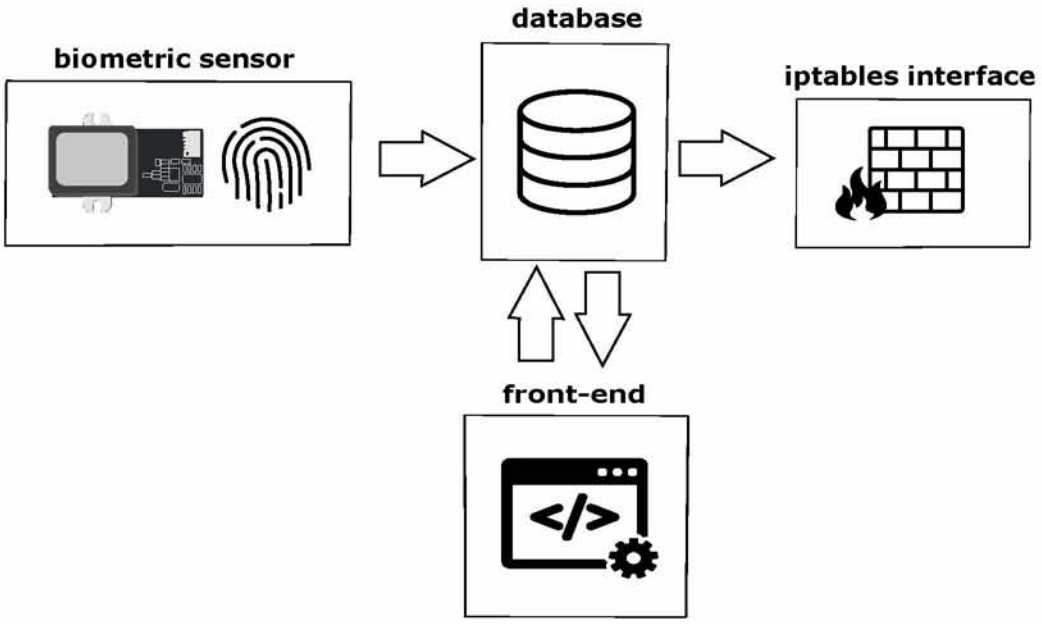


Σχήμα 4.17: Διέλευση των πακέτων ανάλογα με τον προορισμό τους.

Στους πίνακες, εκτός από τις προκαθορισμένες αλυσίδες, μπορούν να δημιουργηθούν και νέες. Στις νέες αλυσίδες τα πακέτα φτάνουν μέσω της ενέργειας JUMP από μία προκαθορισμένη αλυσίδα και ελέγχονται περαιτέρω σε αυτές. Για παράδειγμα, αν υπάρχει κανόνας στην αλυσίδα INPUT που αναφέρει ότι συγκεκριμένα πακέτα θα γίνονται JUMP στη νέα αλυσίδα, τότε αυτά τα πακέτα θα ελέγχονται και από τους κανόνες αυτής της αλυσίδας.

4.2 Υλοποίηση συστήματος

Όλα τα επιμέρους στοιχεία της αρχιτεκτονικής του συστήματος επικοινωνούν με την βάση δεδομένων (Σχήμα 4.18). Τα στοιχεία των χρηστών (όνομα, κωδικός) και οι απαντήσεις στις ερωτήσεις για τους περιορισμούς αποθηκεύονται στην βάση δεδομένων. Επίσης, τα στοιχεία των χρηστών αντλούνται από την βάση όταν θέλουν να συνδεθούν στο σύστημα, να αλλάξουν τις απαντήσεις τους ή να δουν τα διαγράμματα αυστηρότητας. Τα διαγράμματα αυστηρότητας απεικονίζουν την αυστηρότητα του χρήστη σε κάθε κατηγορία και την σύγκριση της με αυτή των άλλων χρηστών. Ο διαχειριστής αποθηκεύει και διαγράφει συσκευές στην βάση δεδομένων μέσω του front-end. Οι χρήστες εγγράφουν το δακτυλικό τους αποτύπωμα, η θέση όπου αποθηκεύτηκε, αποθηκεύεται στην βάση δεδομένων του συστήματος. Επίσης, η συσκευή δακτυλικού αποτυπώματος επικοινωνεί με την βάση για την ταυτοποίηση των δακτυλικών αποτυπωμάτων. Τέλος, οι κανόνες του iptables δομούνται με τους περιορισμούς που έχουν επιλέξει οι χρήστες, οι οποίοι είναι παρόντες στον χώρο. Αυτές οι πληροφορίες αντλούνται από την βάση δεδομένων. Για την υλοποίηση του συστήματος χρησιμοποιήθηκαν οι γλώσσες προγραμματισμού Python, HTML και CSS.

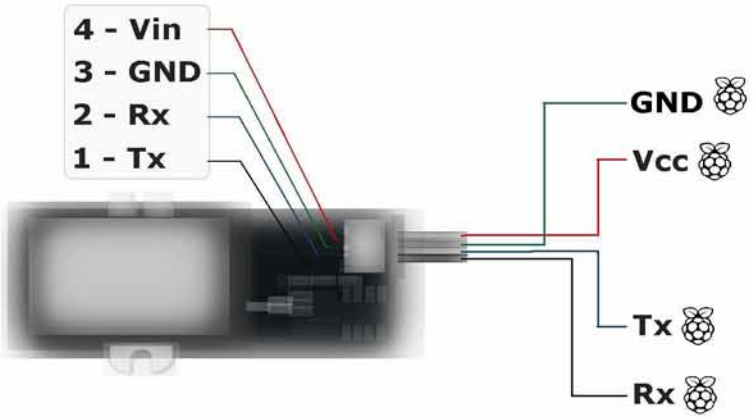


Σχήμα 4.18: Σκελετός αρχιτεκτονικής προτεινόμενου συστήματος.

4.2.1 Biometric sensor

Αρχικά, εγκαταστάθηκε στο Raspberry Pi το λειτουργικό σύστημα Raspbian μέσω του λογισμικού NOOBS. Ενεργοποιήθηκε η σειριακή επικοινωνία του Raspberry Pi για να συνδεθεί ο αισθητήρας του δακτυλικού αποτυπώματος και να είναι εφικτή η επικοινωνία τους, η οποία πραγματοποιείται με UART πρωτόκολλο (5.1.3 Fingerprint). Η σύνδεση πραγματοποιήθηκε με τον εξής τρόπο (Σχήμα 4.19):

- το πρώτο άκρο του αισθητήρα συνδέεται με το GPIO15
- το δεύτερο με το GPIO14
- τα επόμενα δύο στα αντίστοιχα PIN γείωσης και τάσης



Σχήμα 4.19: Συνδεσμολογία αισθητήρα δακτυλικού αποτυπώματος με Raspberry Pi.

Οι συναρτήσεις που υλοποιήθηκαν στην Python για την εγγραφή αλλά και για την ταυτοποίηση του αποτυπώματος είναι οι εξής:

- **CmosLed**: ελέγχει το led του αισθητήρα, που δείχνει αν είναι κλειστός ή ανοιχτός. Η προκαθορισμένη του κατάσταση είναι η κλειστή.
- **EnrollStart**: ξεκινάει η διαδικασία εγγραφής με παράμετρο τη θέση που αυτή θα γίνει. Ελέγχει αν η βάση δεδομένων είναι γεμάτη, αν η θέση που θα γίνει η εγγραφή είναι ανάμεσα στα όρια (0-19), και αν αυτή η θέση χρησιμοποιείται ήδη.
- **Enroll1**: δημιουργεί το πρώτο δείγμα για την εγγραφή.

- **Enroll2**: δημιουργεί το δεύτερο δείγμα για την εγγραφή.
- **Enroll3**: δημιουργεί το τρίτο δείγμα για την εγγραφή. Ενώνει τα τρία δείγματα σε ένα και το αποθηκεύει.
- **IsPressFinger**: ελέγχει αν ένα δάκτυλο τοποθετείται στον αισθητήρα. Αυτή η λειτουργία χρησιμοποιείται ιδιαίτερα κατά την εγγραφή.
- **Identify**: ταυτοποιεί το αποτύπωμα στη βάση δεδομένων. Αν η βάση δεδομένων είναι κενή, επιστρέφει σφάλμα.
- **CaptureFinger**: συλλαμβάνει μια εικόνα δακτυλικού αποτυπώματος (240x216), αν δεν είναι τοποθετημένο ένα δάχτυλο στον αισθητήρα, επιστρέφει με σφάλμα. Εάν η λειτουργία αυτή επιστρέψει με επιτυχία, η εσωτερική μνήμη RAM της συσκευής κρατάει έγκυρη την εικόνα για τις επόμενες εντολές. Πριν από την κλήση συναρτήσεων σχετικών με την εικόνα (Identify, Verify), πρέπει να κληθεί η CaptureFinger.

Ο αισθητήρας χρησιμοποιήθηκε σε δύο σενάρια. Το πρώτο είναι αυτό που εξετάζει εάν ένα δαχτυλικό αποτύπωμα είναι αποθηκευμένο και αν δεν είναι το εγγράφει.

Algorithm 1

```
Led is turned on
Waiting for a finger to capture its image
if  identify()== is not in database:
    for id in range(19):
        if enroll(0,id)== error:
                        led is turned off
                        time.sleep(3)
                        continue
                        enroll(0,id)
                        id=id
                    break
```

Το δεύτερο αναφέρεται στην λειτουργία ταυτοποίησης ενός αποτυπώματος.

Algorithm 2

```
Led is turned off
Waiting for a finger to capture its image
Led is turned off
time.sleep(1)
y=identify()
if y!=empty database or y!= is not in database:
    action
```

Μόλις ο πρώτος χρήστης εγγραφεί, αυτά τα δύο σενάρια τρέχουν παράλληλα με multiprocessing. Η λειτουργία της ταυτοποίησης είναι η προκαθορισμένη λειτουργία του συστήματος. Δηλαδή, ο αισθητήρας του δακτυλικού αποτυπώματος λειτουργεί περιμένοντας κάποιος χρήστης να εισέλθει στον χώρο ή να εξέλθει από αυτόν, ώστε να ταυτοποιήσει το αποτύπωμα του στην βάση δεδομένων. Αυτή η λειτουργία διακόπτεται, όταν κάποιος άλλος χρήστης επιθυμεί να εγγραφεί στο σύστημα, από την λειτουργία της εγγραφής.

4.2.2 Database

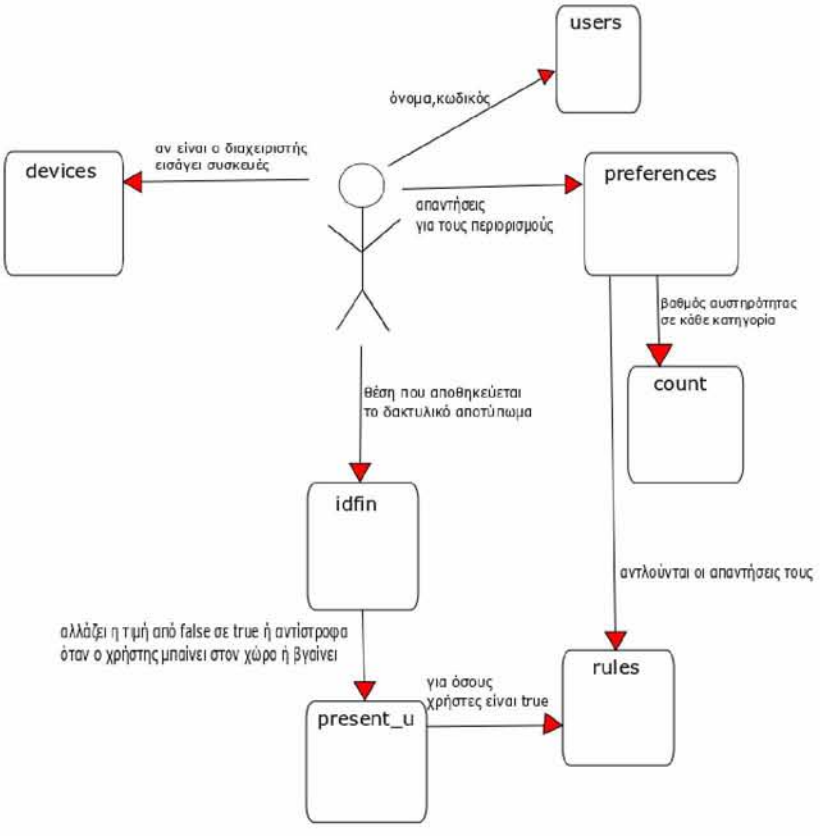
Για την υλοποίηση του συστήματος χρειάστηκε να δημιουργηθεί μια βάση δεδομένων, η οποία συνδέθηκε με την διαδικτυακή εφαρμογή του Flask. Η βάση δεδομένων MySQL εγκαθιστάται στο λογισμικού του Raspberry με τον εξής τρόπο:

- sudo apt update sudo apt upgrade
- sudo apt install mariadb-server
- sudo mysql_secure_installation
- sudo mysql -u root

[mysql] use mysql;
[mysql] update user set plugin=” where User=’root’;
[mysql] flush privileges;
[mysql] \q
Επίσης, απαραίτητο είναι το πακέτο PyMySQL.

Η βάση δεδομένων (Σχήμα 4.20) αποτελείται από επτά πίνακες: users, preferences, devices, count, idfin, present_u και rules. Στον πίνακα users αποθηκεύονται το όνομα και

ο κωδικός των χρηστών με τον κωδικό να είναι κρυπτογραφημένος, ενώ στον preferences εισάγονται οι απαντήσεις για τους κανόνες. Ο πίνακας devices είναι αυτός στον οποίο ο διαχειριστής του συστήματος εισάγει τις συσκευές του χώρου με πέντε χαρακτηριστικά: όνομα, κατηγορία στην οποία ανήκει η συσκευή, την MAC διεύθυνση, την IP διεύθυνση και την IP εμπιστοσύνης. Ο count κρατάει την τιμή αυστηρότητας για κάθε χρήστη σε κάθε κατηγορία. Ο idfin αποθηκεύει τη θέση στην οποία έχει αποθηκευτεί το fingerprint του κάθε χρήστη, από τις είκοσι θέσεις που διαθέτει συνολικά η συσκευή του δακτυλικού αποτυπώματος. Ο πίνακας present_u αντιπροσωπεύει τα άτομα που βρίσκονται μέσα στον χώρο, και ο rules είναι αυτός που καθορίζει τη λειτουργία των συσκευών. Οι πίνακες users, preferences, idfin και present_u που σχετίζονται με τους χρήστες έχουν ένα id το οποίο αντιπροσωπεύει τον κάθε χρήστη. Για παράδειγμα, ο πρώτος χρήστης που θα εγγραφεί θα έχει σε κάθε πίνακα id=1. Έτσι, μπορούν να εκτελεστούν εντολές SQL ανάμεσα σε αυτούς του πίνακες.



Σχήμα 4.20: Απεικόνιση της βάσης δεδομένων.

Η σύνδεση της βάσης δεδομένων με το Flask έγινε μέσω του αντικειμένου create_engine() από το πακέτο flask-sqlalchemy (5.1.1 Flask).

4.2.3 Front-end

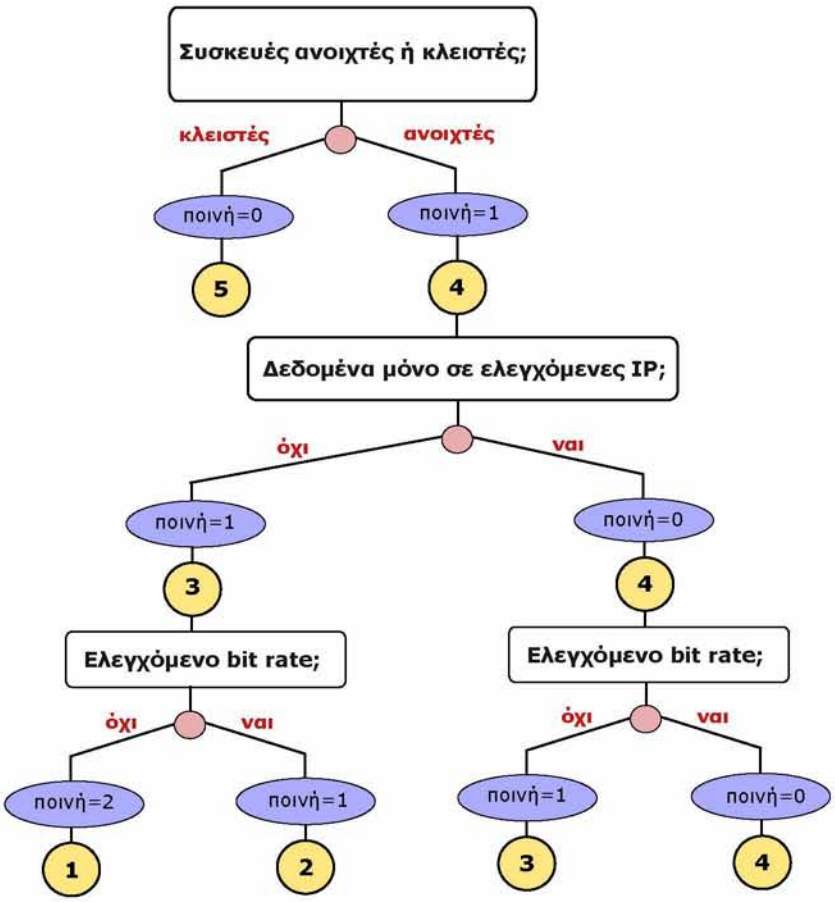
Η διαδικτυακή εφαρμογή του Flask αποτελείται από ένα αρχείο app.py και δεκατρία αρχεία HTML. Στο αρχείο app.py υλοποιήθηκαν τα εξής app.route (5.1.1 Flask): sign, q, log, delete, insert, qupdate, process2, bar2 με τις αντίστοιχες συναρτήσεις καθώς και η συνάρτηση process3().

Η sign() είναι αυτή που εμφανίζει την σελίδα που ο χρήστης εισάγει το όνομα του και τον κωδικό πρόσβασης. Τα στοιχεία αυτά αποθηκεύονται στον πίνακα users της βάσης δεδομένων, με τον κωδικό του χρήστη κωδικοποιημένο με sha 256 κρυπτογράφηση. Αν το όνομα του χρήστη υπάρχει ήδη, επιστρέφει error στην οθόνη.

Η q() επιστρέφει τη σελίδα με τις ερωτήσεις, για τους περιορισμούς των συσκευών, που καλούνται να απαντήσουν οι χρήστες και ελέγχει αν η διαδικασία process3() εκτελείται. Αν ναι, την σταματάει. Οι απαντήσεις των χρηστών αποθηκεύονται στον πίνακα preferences. Βάσει αυτών των απαντήσεων, εισάγονται στον πίνακα count οι τιμές αυστηρότητας του κάθε χρήστη, για κάθε κατηγορία (εικόνα, ήχο, κίνηση, περιβάλλον).

Η αυστηρότητα καθορίζεται από το δένδρο επιλογών (Σχήμα 4.21) και παίρνει τιμές από το ένα (καθόλου) έως το πέντε (πολύ). Οι τρείς περιορισμοί δεν έχουν την ίδια βαρύτητα για αυτό τον λόγο ιεραρχήθηκαν. Θέτοντας σημαντικότερο περιορισμό την επιλογή ανοιχτών ή κλειστών συσκευών, επόμενο την αποστολή δεδομένων σε ελεγχόμενες IP και τέλος αυτόν του ελεγχόμενου ρυθμού μετάδοσης δεδομένων. Το πρώτο που πρέπει να ορίσει ο χρήστης είναι αν οι συσκευές θα είναι ανοιχτές ή οχι, αν αυτό δεν γίνει γνωστό δεν γίνεται να τεθεί άλλος περιορισμός. Αν ο χρήστης αποφασίσει οι συσκευές να είναι ανοιχτές περνάει στον επόμενο περιορισμό ο οποίος είναι αν αυτές οι συσκευές θα στέλνουν δεδομένα μόνο σε ελεγχόμενες IP ή σε όλες τις IP. Είτε ο χρήστης θέσει τον περιορισμό να στέλνονται δεδομένα σε συγκεκριμένες IP είτε όχι πρέπει και να επιλέξει αν ο ρυθμός μεταφοράς τον δεδομένων θα είναι ελεγχόμενος ή οχι, όπου και αν στέλνουν δεδομένα οι συσκευές. Κάθε φορά που ο χρήστης επιλέγει να μην θέσει έναν περιορισμό η τιμή της αυστηρότητας του μειώνεται με μια ποινή που παίρνει την τιμή 1. Όταν ο χρήστης φτάσει στο σημείο να επιλέξει αν θα ορίσει τον περιορισμού του ελεγχόμενου ρυθμού μετάδοσης δεδομένων, ενώ έχει επιλέξει να μην εφαρμοστεί ο περιορισμός της αποστολής δεδομένων σε ελεγχόμενες IP, και δεν τον ορίσει η ποινή παίρνει την τιμή 2. Αυτό γίνεται καθώς δεν επιλέγεται να εφαρμοστεί κανένας περιορισμός και αυτό αποτελεί το πιο μη αυστηρό σενάριο. Οι τιμές αυστηρότητες αντιστοιχούν στους περιορισμούς με την εξής σειρά:

- **Τιμή αυστηρότητας = 5**: Κλειστές συσκευές
- **Τιμή αυστηρότητας = 4**: Αποστολή δεδομένων μόνο σε ελεγχόμενες IP **και** Ελεγχόμενος ρυθμός μετάδοσης δεδομένων
- **Τιμή αυστηρότητας = 3**: Αποστολή δεδομένων μόνο σε ελεγχόμενες IP
- **Τιμή αυστηρότητας = 2**: Ελεγχόμενος ρυθμός μετάδοσης δεδομένων
- **Τιμή αυστηρότητας = 1**: Κανένας περιορισμός



Σχήμα 4.21: Δένδρο επιλογών. Καθορίζει την αυστηρότητα.

Μετά την επιλογή των περιορισμών εμφανίζονται οι οδηγίες του πρέπει να ακολουθήσει ο χρήστης για να εγγράψει το δακτυλικό του αποτύπωμα. Εκεί μέσω ενός συνδέσμου καλείται η process2() η οποία εγγράφει το δακτυλικό αποτύπωμα και αποθηκεύει στο present_u το id του χρήστη και την τιμή false. Επίσης, στον idfin αποθηκεύεται το id

του χρήστη και την θέση στην οποία αποθηκεύτηκε το αποτύπωμα (από την μηδέν έως την δεκαεννιά). Εφόσον ολοκληρωθεί η process2() ξεκινάει η λειτουργία της process3().

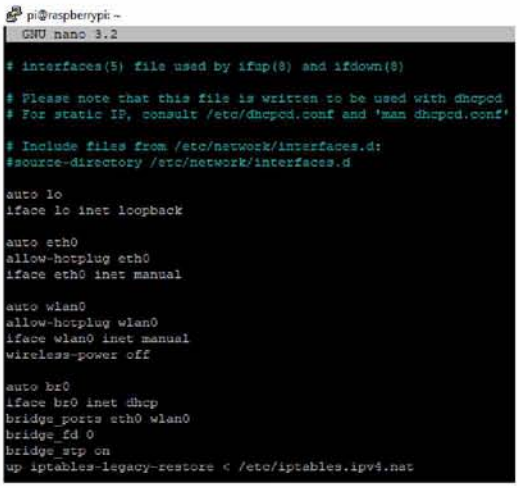
Η process3() είναι η συνάρτηση η οποία αναγνωρίζει το αποτύπωμα ενός χρήστη και αλλάζει την προκαθορισμένη τιμή false στον present_u σε true. Αυτό δηλώνει πως ο χρήστη μπήκε στον χώρο. Και αντίστοιχα όταν ένας χρήστης βγαίνει από τον χώρο αναγνωρίζει το αποτύπωμα του και η τιμή από true γίνεται false. Με βάση το πόσες τιμές είναι true στον present_u, δηλαδή πόσοι χρήστες βρίσκονται στον χώρο, δομείται ο πίνακας rules ο οποίος καθορίζει τους κανόνες iptables.

Η log() εμφανίζει την φόρμα σύνδεσης του χρήστη. Αν τα δεδομένα στα πεδία username και password είναι αποθηκευμένα στον πίνακα users τότε επιστρέφει ένα γράφημα για την αυστηρότητα του χρήστη που συνδέθηκε. Ο χρήστης μπορεί μέσω ενός συνδέσμου να αλλάξει τις επιλογές του καλώντας την qupdate(), η οποία ενημερώνει τον πίνακα preferences με τι νέες επιλογές. Εναλλακτικά, ο χρήστης μπορεί μέσω ενός άλλου συνδέσμου να δει ένα γράφημα που συγκρίνει τις τιμές αυστηρότητας του με αυτές των υπόλοιπων χρηστών, καλώντας την συνάρτηση bar2(). Αν τα στοιχεία που παρέχει ο χρήστης δεν είναι σωστά τότε επιστρέφεται μήνυμα ότι ή το όνομα ή ο κωδικός είναι λάθος.

Αν τα στοιχεία που εισάγει ο χρήστης αντιστοιχούν στον διαχειριστή του συστήματος συλλέγονται όλες οι συσκευές από τον πίνακα devices και επιστρέφονται με την μορφή ενός πίνακα που έχει δύο κουμπιά, delete και insert. Τα κουμπιά αυτά καλούν τις συναρτήσεις delete(name) και την insert() για την διαγράφη και την εισαγωγή μιας συσκευής στον devices, αντίστοιχα.

4.2.4 Iptables interface

Το Raspberry Pi χρησιμοποιήθηκε ως access point (5.1.4 Access point), για την δημιουργία ενός δίκτυο στο οποίο συνδέθηκαν όλες οι συσκευές του συστήματος με σκοπό να ακολουθούν του κανόνες του iptables. Το Raspberry Pi συνδέθηκε με ένα καλώδιο ethernet στο router. Εγκαταστάθηκαν τα πακέτα hostapd και bridge-utils. Το hostapd είναι το πακέτο που μας επιτρέπει να δημιουργήσουμε ένα ασύρματο hotspot χρησιμοποιώντας ένα Raspberry Pi. Το hostapd για να λειτουργήσει πρέπει να γίνει unmask και να ενεργοποιηθεί. Το πακέτο Bridge-Utils περιέχει ένα βοηθητικό πρόγραμμα που απαιτείται για τη δημιουργία και τη διαχείριση συσκευών γέφυρας. Επίσης, τα αρχεία /etc/network/interfaces και /etc/hostapd/hostapd.conf διαμορφώθηκαν όπως φαίνεται στο Σχήμα 4.22 και στο Σχήμα 4.23.



Σχήμα 4.22: Διαμόρφωση /etc/network/interfaces αρχείου για την λειτουργία access point.

Σχήμα 4.23: Διαμόρφωση /etc/hostapd/hostapd.conf αρχείου για την λειτουργία access point.

Οι κανόνες του iptables (5.1.5 Iptables) δημιουργήθηκαν με το πακέτο python-iptables [29]. Ο σκοπός της χρήσης του iptables είναι να περιοριστούν οι ενέργειες των συσκευών που υπάρχουν στον χώρο, όπου αυτό χρειάζεται. Οι συσκευές έχουν χωριστεί σε τέσσερις κατηγορίες, και στην κάθε κατηγορία υπάρχουν οι εξής τρείς περιορισμοί:

- οι συσκευές να είναι κλειστές,
- να στέλνουν δεδομένα μόνο σε συγκεκριμένες IP, και
- να είναι ο ρυθμός μεταφοράς τον δεδομένων ελεγχόμενος.

Όλοι οι κανόνες βρίσκονται στην αλυσίδα FORWARD του πίνακα Filter διότι το

Raspberry δρομολογεί τα πακέτα από το wlan στο διαδίκτυο. Σε όλους τους κανόνες χρησιμοποιείται το φίλτρο physdev που ταιριάζει στη θύρα της γέφυρας τις εντός και εκτός συσκευές, που είναι συνδεδεμένες στη γέφυρα. Το physdev-in ονομάζει τη θύρα της γέφυρας που θα λάβει τα πακέτα, και το physdev-out τη θύρα μέσω της οποίας τα πακέτα θα σταλούν.
Οι κανόνες που δημιουργήθηκαν ώστε να αντιπροσωπεύουν τους 3 περιορισμούς είναι της μορφής:

- **target: DROP** source: IP των συσκευών που πρέπει να είναι κλειστές destination: anywhere filter: PHYSDEV match –physdev-in wlan0 physdev-out eth0
- **target: DROP** source: IP των συσκευών που πρέπει να στέλνουν δεδομένα σε συγκεκριμένη IP destination: ! (αν δεν είναι) η IP εμπιστοσύνης filter: PHYSDEV match –physdev-in wlan0 physdev-out eth0
- **target: RATE LIMIT** source: IP των συσκευών που πρέπει να έχουν ελεγχόμενο ρυθμό μεταφοράς δεδομένων destination: anywhere filter: PHYSDEV match –physdev-in wlan0 physdev-out eth0

Όσα πακέτα ταιριάζουν με τον τελευταίο κανόνα μεταφέρονται στην αλυσίδα RATE-LIMIT. Όλα τα πακέτα που ελέγχονται από τους κανόνες αυτής της αλυσίδας έχουν ως φίλτρο το hashlimit, το οποίο εφαρμόζει ένα όριο σε μια ομάδα πακέτων (μια ομάδα μπορεί να έχει την ίδια IP προέλευσης). Το όριο αυτό είναι ότι μετά από τα πρώτα είκοσι πακέτα ο ρυθμός παραλαβής πακέτων θα είναι δέκα ανά δευτερόλεπτο.

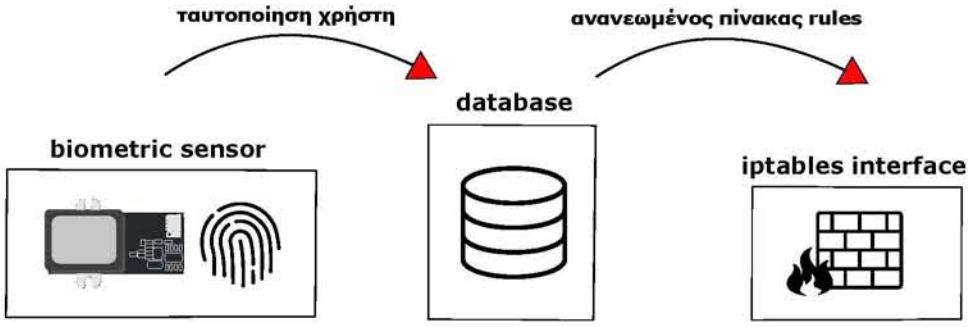
4.2.5 Επικοινωνία επιμέρους στοιχείων της αρχιτεκτονικής

Στην σελίδα με τις οδηγίες εγγραφής του δακτυλικού αποτυπώματος υπάρχει ένας σύνδεσμος. Όταν πατηθεί αυτός ο σύνδεσμος ενεργοποιείται η διαδικασία εγγραφής. Μόλις ο πρώτος χρήστης εγγραφεί στο σύστημα οι λειτουργίες ταυτοποίησης και εγγραφής τρέχουν παράλληλα. Με την υπό συνθήκη λειτουργία να είναι αυτή της ταυτοποίησης (Σχήμα 4.24). Δηλαδή, ο αισθητήρας του δακτυλικού αποτυπώματος λειτουργεί περιμένοντας κάποιος χρήστης να εισέλθει στον χώρο ή να εξέλθει από αυτόν, ώστε να ταυτοποιήσει το αποτύπωμα του στην βάση δεδομένων. Αυτή η λειτουργία διακόπτεται, όταν κάποιος άλλος χρήστης επιθυμεί να εγγραφεί στο σύστημα, από την λειτουργία της εγγραφής. Όταν ο επόμενος χρήστης που επιθυμεί να εγγραφεί μεταβεί στην σελίδα όπου

θα επιλέξει περιορισμούς, η διαδικασία ταυτοποίησης σταματάει και το led του αισθητήρα σβήνει. Μετά την εγγραφή του χρήστη ενεργοποιείται πάλι η διαδικασία της ταυτοποίησης.

Σχήμα 4.24: Επικοινωνία biometric sensor με front-end.

Ακόμη, το iptables interface επικοινωνεί έμμεσα με το biometric sensor (Σχήμα 4.25). Οι κανόνες του iptables δομούνται με βάση τον πίνακα rules της βάσης δεδομένων. Ο πίνακας αυτός δημιουργείται από τους πιο αυστηρούς περιορισμούς που έχουν θέσει οι χρήστες που είναι παρόντες στον χώρο. Οι κανόνες του iptables αναδομούνται κάθε φορά που ένας χρήστης ταυτοποιεί το δάχτυλο του στην βάση δεδομένων, δηλαδή εισέρχεται στον χώρο ή εξέρχεται από αυτόν, με βάση των ανανεωμένο πίνακα rules.



Σχήμα 4.25: Επικοινωνία biometric sensor με iptables interface.

4.3 Λειτουργία συστήματος

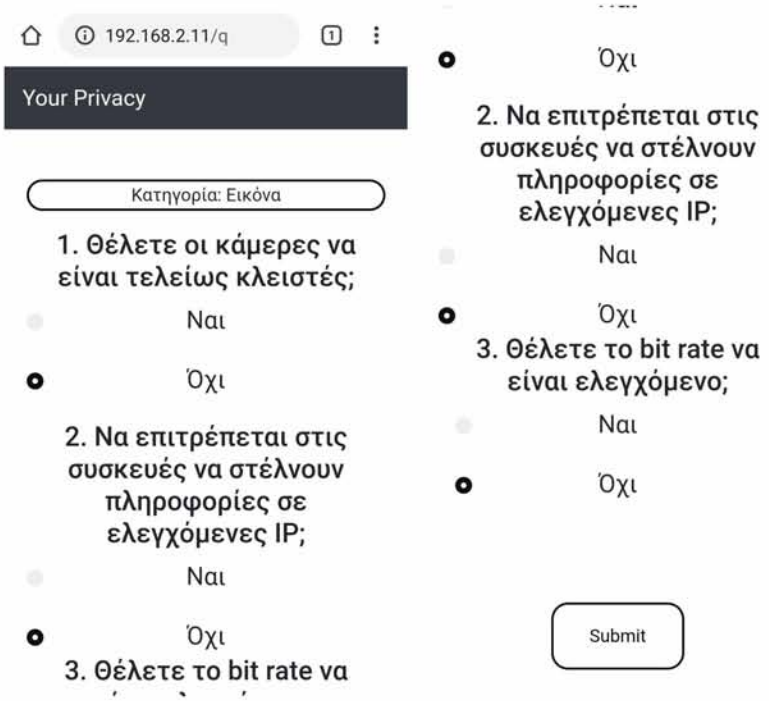
Έστω ότι η διαδικασία ξεκινάει χωρίς να έχει εγγραφεί κανένας χρήστης στο σύστημα. Ο χρήστης φτάνει στον κοινόχρηστο χώρο και καλείται να σκανάρει με το κινητό

του ένα κωδικό QR ώστε να μπει στην διαδικτυακή εφαρμογή. Το πρώτο βήμα που πρέπει να κάνει είναι να εισάγει ένα όνομα και έναν κωδικό που επιθυμεί για αν εγγραφεί στο σύστημα (Σχήμα 4.26). Αφού τα στοιχεία του αποθηκευτούν στην βάση δεδομένων, μεταβαίνει στην σελίδα όπου θα δηλώσει τις προτιμήσεις του σχετικά με τους κανόνες που θέλει να ακολουθούν οι συσκευές του συστήματος (Σχήμα 4.27). Οι συσκευές έχουν χωριστεί σε τέσσερις κατηγορίες, και στην κάθε κατηγορία υπάρχουν οι εξής τρείς περιορισμοί που ο χρήστης μπορεί να επιλέξει:

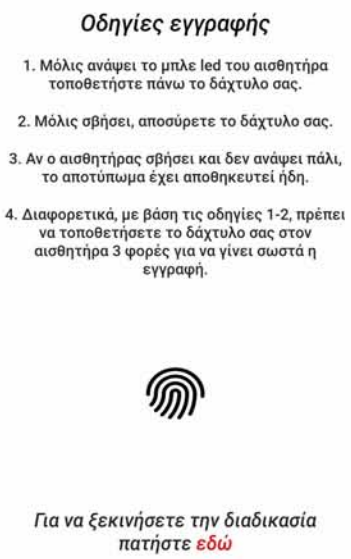
- οι συσκευές να είναι κλειστές,
- να στέλνουν δεδομένα μόνο σε συγκεκριμένες IP, και
- να είναι ο ρυθμός μεταφοράς τον δεδομένων ελεγχόμενος.

Μετά την ολοκλήρωση αυτού του βήματος παρουσιάζονται κάποιες οδηγίες ώστε να γίνει πιο εύκολη η εγγραφή του δακτυλικού αποτυπώματος (Σχήμα 4.28). Μόλις ο χρήστης πατήσει τον αντίστοιχο σύνδεσμο της σελίδας, ο αισθητήρας ανάβει. Ο χρήστης τοποθετεί πάνω το δάχτυλο του, ο αισθητήρας σβήνει. Αν το αποτύπωμα δεν είναι ήδη αποθηκευμένο, ανάβει ξανά για να ξεκινήσει η διαδικασία της εγγραφής. Όταν το led του αισθητήρα σταθεροποιηθεί ο χρήστης πρέπει να τοποθετήσει το δάχτυλο του τρείς φορές, με χρονικό διάστημα ανάμεσα στις τρείς τοποθετήσεις πέντε με έξι δευτερόλεπτα. Αφού ο πρώτος χρήστης εγγραφεί, το led του αισθητήρα είναι πάντα αναμμένο και περιμένει να ταυτοποιήσει το αποτύπωμα του χρήστη που εισέρχεται, ή εξέρχεται, από τον χώρο, ώστε να ρυθμιστούν οι περιορισμοί των συσκευών. Αυτή η διαδικασία του αισθητήρα διακόπτεται όταν ο επόμενος χρήστης θελήσει να εγγραφεί. Όταν μεταβεί στην σελίδα με τις ερωτήσεις ο αισθητήρας σβήνει έτσι ώστε όταν στην σελίδα των οδηγιών πατηθεί ο σύνδεσμος "εδώ" (Σχήμα 4.28) να ξεκινήσει η εγγραφή. Έπειτα επιστρέφει στην προηγούμενη λειτουργία του.

Σχήμα 4.26: Απεικόνιση σελίδας εγγραφής του χρήστη.

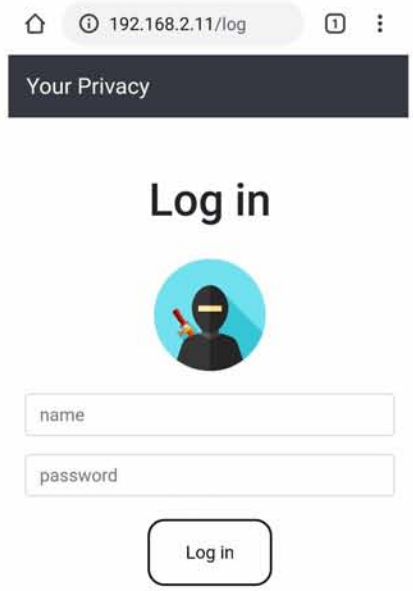


Σχήμα 4.27: Απεικόνιση σελίδας ερωτήσεων. Ο χρήστης επιλέγει περιορισμούς.

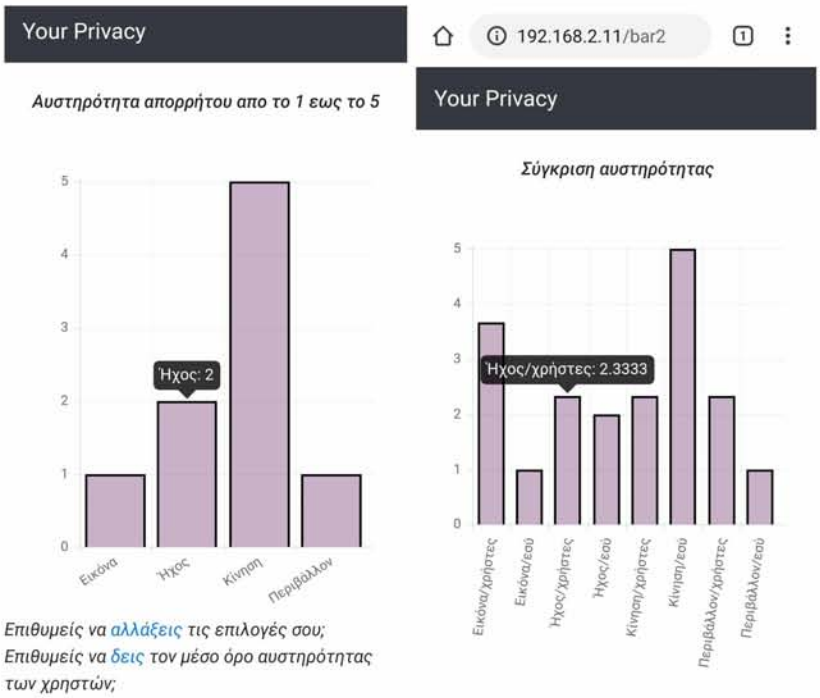


Σχήμα 4.28: Απεικόνιση σελίδας οδηγιών για την σωστή εγγραφή δακτυλικού αποτυπώματος.

Ακόμη, ο χρήστης μέσω της εφαρμογής έχει την δυνατότητα να κάνει log in (Σχήμα 4.29) και να αλλάξει τις προτιμήσεις του ως προς τους κανόνες ασφαλείας που έχει επιλέξει. Επίσης, μπορεί να δει σε γράφημα, πόσο αυστηρός ήταν στις επιλογές του με κλίμακα από ένα (καθόλου) ως πέντε (πολύ), αλλά και πώς συγκρίνεται με το σύνολο των χρηστών (Σχήμα 4.30). Υπάρχει ένας εξουσιοδοτημένος χρήστης, ο διαχειριστής, ο οποίος διαχειρίζεται τις συσκευές του συστήματος. Εφόσον ο χρήστης εισάγει στη σελίδα log in τα στοιχεία που αντιστοιχούν στον διαχειριστή θα παρουσιαστεί ένας πίνακας με τις ήδη υπάρχουσες συσκευές του συστήματος. Έχει την δυνατότητα να εισάγει μία νέα συσκευή ή να διαγράψει κάποια (Σχήμα 4.31). Αν ο διαχειριστής δεν έχει εισάγει καμία συσκευή ο πίνακας είναι κενός και έχει μόνο την επιλογή να εισάγει συσκευή.



Σχήμα 4.29: Απεικόνιση σελίδας σύνδεσης του χρήστη.



Σχήμα 4.30: Απεικόνιση, αυστηρότητας του χρήστη και σύγκρισης αυστηρότητας μεταξύ του χρήστη και των άλλων χρηστών, με διάγραμμα.

Σχήμα 4.31: Διαχείριση συσκευών του συστήματος μέσω του προφίλ του διαχειριστή.

Όταν ένας χρήστης εισέρχεται ή εξέρχεται από τον χώρο οι κανόνες που ακολουθούν οι συσκευές αλλάζουν με βάση τους κανόνες του iptables. Για παράδειγμα, στο Σχήμα 4.32, στον πρώτο πίνακα παρουσιάζονται οι προτιμήσεις δύο χρηστών που είναι εγγεγραμμένοι στο σύστημα. Οι στήλες q1, q2, q3 αφορούν την κατηγορία εικόνα, οι q4, q5,q6 τον ήχο, οι επόμενες τρεις την κίνηση, και οι τελευταίες τρεις την κατηγορία περιβάλλον. Όπως φαίνεται, ο πρώτος χρήστης έχει ορίσει περιορισμούς για τις συσκευές που βρίσκονται στις κατηγορίες εικόνα, ήχος, περιβάλλον. Ενώ ο δεύτερος επέλεξε και οι συσκευές στην κατηγορία κίνηση να ακολουθήσουν έναν κανόνα. Στον δεύτερο πίνακα δίνονται οι συσκευές του χώρου που έχουν καταχωρηθεί στη βάση δεδομένων. Μία IP camera (κατηγορία εικόνα), μία έξυπνη συσκευή ανίχνευσης φωνής (κατηγορία ήχος), μία συσκευή αναγνώρισης κίνησης (κατηγορία κίνηση) και τέλος μία συσκευή που ανιχνεύει περιβαλλοντικές συνθήκες (κατηγορία περιβάλλον).

```
MariaDB [register]> select * from preferences;
+----+------+------+------+------+------+------+------+------+------+------+------+
| id | q1   | q2   | q3   | q4   | q5   | q6   | q7   | q8   | q9   | q10  | q11  | q12  |
+----+------+------+------+------+------+------+------+------+------+------+------+------+
|  1 | y    | n    | n    | n    | y    | n    | n    | n    | n    | n    | n    | y    |
|  2 | y    | n    | n    | n    | n    | y    | n    | y    | n    | n    | n    | y    |
+----+------+------+------+------+------+------+------+------+------+------+------+------+
2 rows in set (0.000 sec)

MariaDB [register]> select * from devices;
+----+-------------------------+-----------+-------------------+-------------+-------------------+
| id | name                    | category  | mac               | IP          | IPtrust           |
+----+-------------------------+-----------+-------------------+-------------+-------------------+
|  4 | Lora-TemperatureSensor  | periballon | E4:A7:C5:DD:4B:5B | 192.168.2.6  | !123.45.67.89     |
|  5 | Xiaomi-IPcam            | eikona    | D0:FF:98:97:57:92 | 192.168.2.9  | !172.16.254.1     |
|  6 | GoogleHome-Charcoal     | ixos      | 2C:57:31:ED:CE:04 | 192.168.2.7  | !195.251.108.181  |
|  7 | Philips-Motion Sensor   | kinisi    | 78:0C:B8:A9:97:F5 | 192.168.2.10 | !195.251.108.181  |
+----+-------------------------+-----------+-------------------+-------------+-------------------+
```

Σχήμα 4.32: Προτιμήσεις δύο εγγεγραμμένων χρηστών και συσκευές του συστήματος.

Οι κανόνες του iptables εξαρτώνται από τις προτιμήσεις οι οποίες συντελούν έναν γενικό κανόνα. Ο γενικός κανόνας δημιουργείται μόνο από τους χρήστες του βρίσκονται στον χώρο μια δεδομένη χρονική στιγμή, και λαμβάνει υπόψη του την πιο αυστηρή επιλογή. Ο χρήστης για να δηλώσει αν είναι, ή όχι, στον χώρο, χρειάζεται κάθε φορά που εισέρχεται ή εξέρχεται να τοποθετεί το δάχτυλο του στον αισθητήρα. Στο Σχήμα 4.33 φαίνεται ο γενικός κανόνας με βάση τους δύο χρήστες οι οποίοι είναι παρόντες. Σύμφωνα με τα παραπάνω στοιχεία (Σχήμα 4.32) ο πρώτος χρήστης έχει επιλέξει οι συσκευές στην κατηγορία κίνηση να μην στέλνουν δεδομένα σε μία συγκεκριμένη IP. Ο άλλος χρήστης έχει επιλέξει το αντίθετο, η μεταφορά δεδομένων να γίνεται μόνο με την trust IP. Εφόσον ο γενικός κανόνας δομείται με βάση την πιο αυστηρή απάντηση, οι συσκευές αυτές θα στέλνουν δεδομένα μόνο στην trust IP τους.

```
+-----+---------+
| idu | present |
+-----+---------+
|   1 | true    |
|   2 | true    |
+-----+---------+
2 rows in set (0.001 sec)

MariaDB [register]> select * from rules;
+------+------+------+------+------+------+------+------+------+------+------+------+------+
| id   | q1   | q2   | q3   | q4   | q5   | q6   | q7   | q8   | q9   | q10  | q11  | q12  |
+------+------+------+------+------+------+------+------+------+------+------+------+------+
|    1 | y    | n    | n    | n    | y    | y    | n    | y    | n    | n    | n    | y    |
+------+------+------+------+------+------+------+------+------+------+------+------+------+
```

Σχήμα 4.33: Γενικός κανόνας με τους δύο χρήστες παρόντες στον χώρο.

Βάσει του παραπάνω γενικού κανόνα (Σχήμα 4.33), οι κανόνες του iptables θα συμπεριλαμβάνουν όλες τις συσκευές αφού υπάρχει περιορισμός σε κάθε κατηγορία. Εφόσον το q1 είναι y, η Xiaomi-IPcam θα είναι κλειστή. Για την συσκευή GoogleHome-Charcoal, που ανήκει στην κατηγορία ήχος, ισχύει η αποστολή δεδομένων μόνο σε συγκεκριμένες IP και ο ελεγχόμενος ρυθμός αποστολής δεδομένων. Ενώ οι χρήστες έχουν ορίσει η συσκευή Philips-Motion Sensor να στέλνει δεδομένων στην trust IP της. Στην κατηγορία περιβάλλον, οι χρήστες έχουν ορίσει ελεγχόμενο ρυθμό μετάδοσης δεδομένων

για την συσκευή που ανήκει σε αυτή την κατηγορία. Επομένως, οι κανόνες iptables διαμορφώνονται όπως φαίνεται στο Σχήμα 4.34.

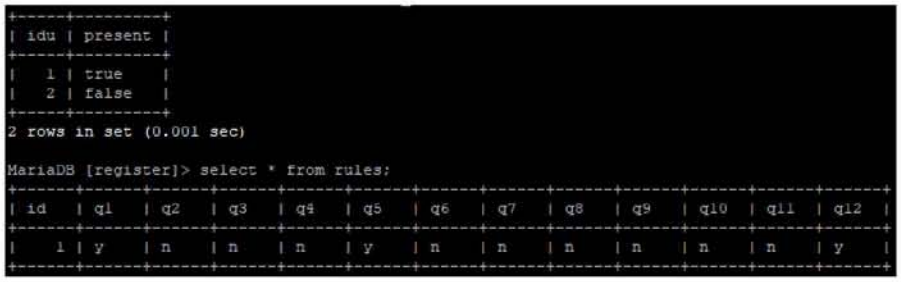
```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RATE-LIMIT  all  --  192.168.2.6          anywhere             PHYSDEV match --physdev-in wlan0 --physdev-out eth0
RATE-LIMIT  all  --  192.168.2.7          anywhere             PHYSDEV match --physdev-in wlan0 --physdev-out eth0
DROP       all  --  192.168.2.10        !181.108.251.195.in-addr.arpa  PHYSDEV match --physdev-in wlan0 --physdev-out eth0
DROP       all  --  192.168.2.7         !181.108.251.195.in-addr.arpa  PHYSDEV match --physdev-in wlan0 --physdev-out eth0
DROP       all  --  192.168.2.9          anywhere             PHYSDEV match --physdev-in wlan0 --physdev-out eth0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RATE-LIMIT (2 references)
target     prot opt source               destination
DROP       all  --  anywhere             anywhere             limit: up to 10/sec burst 20 mode srcip htable-expire 1100
ACCEPT     all  --  anywhere             anywhere             limit: up to 10/sec burst 20 mode srcip htable-expire 1100
```

Σχήμα 4.34: Κανόνες iptables που ακολουθούν οι συσκευές με τους δύο χρήστες παρών στον χώρο.

Ο πρώτος κανόνας αφορά τον ρυθμό μετάδοσης πακέτων. Όταν εντοπιστεί πακέτο που έχει IP προέλευσης, στο συγκεκριμένο παράδειγμα, την IP της συσκευής LoraTemperatureSensor και χωρίς να υπάρχει περιορισμός στην IP προορισμού μεταφέρονται στην αλυσίδα RATE-LIMIT. Στην RATE-LIMIT γίνεται περαιτέρω επερξεργασία των πακέτων θέτοντας το όριο στον ρυθμό μετάδοσης τους. Σύμφωνα με τον πέμπτο κανόνα, πακέτα με IP προέλευσης την IP της συσκευής Xiaomi-IPcam απορρίπτονται όποια και να είναι η IP προορισμού. Όπως φαίνεται, ο τρίτος και ο τέταρτος κανόνας επιτρέπει να στείλουν πακέτα μόνο στην trust IP που έχει οριστεί από τον διαχειριστή των συσκευών.

Στο παράδειγμα που παρουσιάζεται, υπάρχουν άλλες τρείς περιπτώσεις. Η πρώτη είναι να είναι παρών μόνο ο πρώτος χρήστης στον χώρο, η δεύτερη να είναι ο δεύτερος και η τρίτη να μην είναι κανείς. Ο γενικός κανόνας τροποποιείται κάθε φορά ανάλογα με τις συνθήκες όπως και οι κανόνες του iptables.

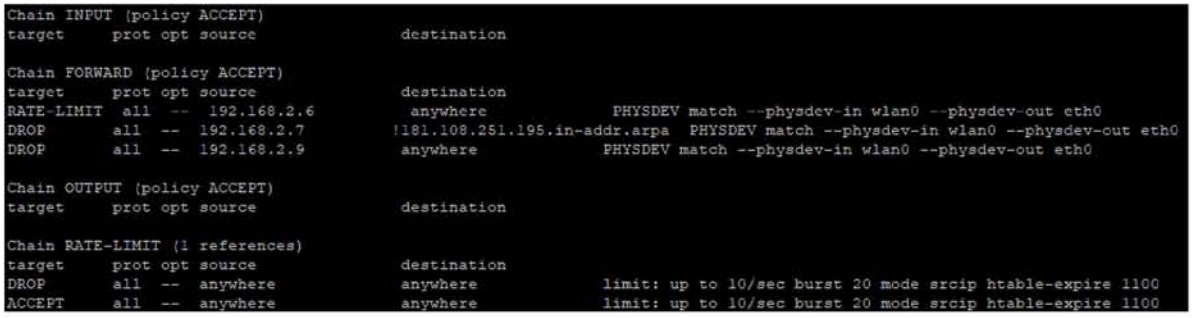
Όταν ο δεύτερος χρήστης εξέλθει από τον χώρο, ο rules διαμορφώνεται έτσι ώστε να περιλαμβάνει μόνο τις επιλογές του πρώτου χρήστη και οι κανόνες του iptables αλλάζουν αντίστοιχα (Σχήμα 4.35).



```
+-----+---------+
| idu | present |
+-----+---------+
|   1 | true    |
|   2 | false   |
+-----+---------+
2 rows in set (0.001 sec)

MariaDB [register]> select * from rules;
+------+------+------+------+------+------+------+------+------+------+------+------+------+
| id   | q1   | q2   | q3   | q4   | q5   | q6   | q7   | q8   | q9   | q10  | q11  | q12  |
+------+------+------+------+------+------+------+------+------+------+------+------+------+
|    1 | y    | n    | n    | n    | y    | n    | n    | n    | n    | n    | n    | y    |
+------+------+------+------+------+------+------+------+------+------+------+------+------+
```

Σχήμα 4.35: Γενικός κανόνας με τον πρώτο χρήστη παρών στον χώρο.

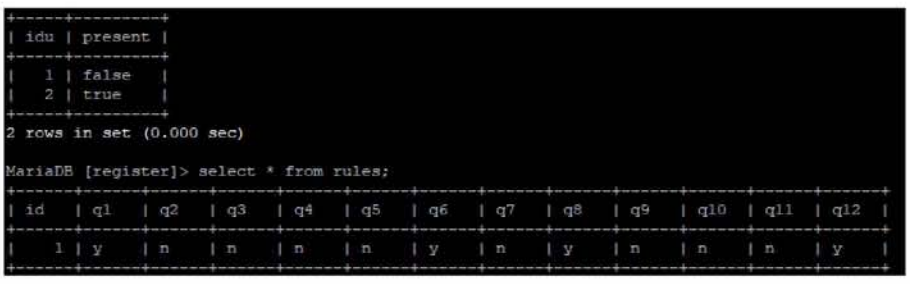
Στην κατηγορία εικόνα η συσκευή θα είναι κλειστή ενώ στην κατηγορία ήχο θα αποστέλλει δεδομένα μόνο στην trust IP της. Για την κατηγορία περιβάλλον εφαρμόζεται ο κανόνας του ελεγχόμενου ρυθμού αποστολής δεδομένων. Ο χρήστης δεν έχει ορίσει κάποιον περιορισμό για την κατηγορία κίνηση.



```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RATE-LIMIT  all  --  192.168.2.6          anywhere             PHYSDEV match --physdev-in wlan0 --physdev-out eth0
DROP       all  --  192.168.2.7          !181.108.251.195.in-addr.arpa  PHYSDEV match --physdev-in wlan0 --physdev-out eth0
DROP       all  --  192.168.2.9          anywhere             PHYSDEV match --physdev-in wlan0 --physdev-out eth0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RATE-LIMIT (1 references)
target     prot opt source               destination
DROP       all  --  anywhere             anywhere             limit: up to 10/sec burst 20 mode srcip htable-expire 1100
ACCEPT     all  --  anywhere             anywhere             limit: up to 10/sec burst 20 mode srcip htable-expire 1100
```

Σχήμα 4.36: Κανόνες iptables που ακολουθούν οι συσκευές με τον πρώτο χρήστη παρών στον χώρο.

Στην περίπτωση που παρών στον χώρο είναι μόνο ο δεύτερος χρήστης (Σχήμα 4.37) οι κανόνες που εφαρμόζονται είναι τέσσερις και αντιπροσωπεύουν τις επιλογές αυτού του χρήστη.



```
+-----+---------+
| idu | present |
+-----+---------+
|   1 | false   |
|   2 | true    |
+-----+---------+
2 rows in set (0.000 sec)

MariaDB [register]> select * from rules;
+------+------+------+------+------+------+------+------+------+------+------+------+------+
| id   | q1   | q2   | q3   | q4   | q5   | q6   | q7   | q8   | q9   | q10  | q11  | q12  |
+------+------+------+------+------+------+------+------+------+------+------+------+------+
|    1 | y    | n    | n    | n    | n    | y    | n    | y    | n    | n    | n    | y    |
+------+------+------+------+------+------+------+------+------+------+------+------+------+
```

Σχήμα 4.37: Γενικός κανόνας με τον δεύτερο χρήστη παρών στον χώρο.

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination
RATE-LIMIT  all  --  192.168.2.6          anywhere             PHYSDEV match --physdev-in wlan0 --physdev-out eth0
RATE-LIMIT  all  --  192.168.2.7          anywhere             PHYSDEV match --physdev-in wlan0 --physdev-out eth0
DROP       all  --  192.168.2.10        !181.108.251.195.in-addr.arpa  PHYSDEV match --physdev-in wlan0 --physdev-out eth0
DROP       all  --  192.168.2.9          anywhere             PHYSDEV match --physdev-in wlan0 --physdev-out eth0

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination

Chain RATE-LIMIT (2 references)
target     prot opt source               destination
DROP       all  --  anywhere             anywhere             limit: up to 10/sec burst 20 mode srcip htable-expire 1100
ACCEPT     all  --  anywhere             anywhere             limit: up to 10/sec burst 20 mode srcip htable-expire 1100
```

Σχήμα 4.38: Κανόνες iptables που ακολουθούν οι συσκευές με τον δεύτερο χρήστη παρών στον χώρο.

Όταν και οι δύο χρήστες αποχωρήσουν από τον χώρο όλες οι στήλες του πίνακα rules έχουν την τιμή n. Τότε δεν εφαρμόζεται κανένας κανόνας.

5. ΣΥΜΠΕΡΑΣΜΑΤΑ

Τα τελευταία χρόνια, ο αναδυόμενος τομέας του IoT έχει προσελκύσει σημαντικό ενδιαφέρον. Παρά την γρήγορη εξέλιξη του, συνεχώς προκύπτουν νέες και σοβαρές προκλήσεις. Τα δύο κύρια ανοιχτά ερωτήματα είναι η ασφάλεια του συστήματος και η ιδιωτικότητα του χρήστη, δύο θέματα που η παρούσα εργασία διερευνά.

Αρχικά, παρουσιάστηκε η αρχιτεκτονική των IoT συστημάτων, αναφέρθηκαν πιθανές απειλές και τρόποι προστασίας. Προτάθηκαν τεχνικές λύσεις για το θέμα της ασφάλειας και συστήματα που συμπεριλαμβάνουν Blockchain και βιομετρικά χαρακτηρίστηκα για την ενίσχυση της ιδιωτικότητας. Το σύστημα που υλοποιήθηκε στην παρούσα εργασία αφορά την ιδιωτικοτήτα του χρήστη σε ένα σύστημα IoT, δίνοντας του την δυνατότητα να ρυθμίζει εύκολα την λειτουργία των συσκευών σε έναν κοινόχρηστο χώρο.
Τα πλεονέκτημα του συστήματος είναι:

- η ευκολία στη χρήση. Ο χρήστης καλείται να χρησιμοποιήσει ένα interface και να τοποθετεί το δάχτυλο του στον αισθητήρα του δακτυλικού αποτυπώματος όταν επιθυμεί να εγγραφεί στο σύστημα, να εισέλθει ή να εξέλθει από τον χώρο.

- σέβεται την ιδιωτικότητα του χρήστη. Ο χρήστης καθορίζει πως θα λειτουργούν οι συσκευές του χώρου, αν θα συλλέγουν δεδομένα, που θα αποστέλλονται και με τι ρυθμό. Γεγονός που αποτρέπει την πρόσβαση σε προσωπικά δεδομένα και την χρήση αυτών.

- αξιοπιστία, λόγω της χρήσης βιομετρικού χαρακτηριστικού.

- χαμηλό κόστος υλοποίησης. Το υλισμικό (hardware) αποτελείται από ένα Raspberry Pi 3 και μια συσκευή δακτυλικού αποτυπώματος.

Μια μελλοντική πρόσθετη λειτουργία και στόχος αποτελεί ο έλεγχος των πακέτων που στέλνονται και λαμβάνονται από τις υπό περιορισμούς συσκευές. Για την αποφυγή επιθέσεων που σχετίζονται με το δίκτυο, είναι σημαντικό να αναγνωρίζονται οι τύποι επιθέσεων εναντίον συστημάτων και τα ζητήματα που σχετίζονται με το δίκτυο. Τα καταγεγραμμένα πακέτα μπορούν να αποκαλύψουν τις υπογραφές των επιθέσεων και αυτές οι πληροφορίες μπορούν να επιτρέψουν στους χρήστες να ανακτήσουν τα συστήματα έπειτα από ζημιές που προκαλούνται από τους πιθανούς εισβολείς.

Ένα πρόγραμμα που χρησιμοποιείται για ανάλυση πακέτων σε ένα δίκτυο είναι το Wireshark. Σε αυτή την περίπτωση όμως, το πακέτο Pyshark της Python ταιριάζει για την ανάλυση της κυκλοφορίας του δικτύου. Το Pyshark είναι ένα Python wrapper για το tshark, επιτρέποντας την ανάλυση πακέτων Python χρησιμοποιώντας διαχωριστές Wireshark. Πιο συγκεκριμένα, με αυτό τον τρόπο επιτρέπεται να εκτελούνται εντολές tshark, οι λειτουργίες του Wireshark, στην Python. Το Pyshark διαθέτει τα αντικείμενα "Capture" (Live, Remote, File, InMem). Κάθε αντικείμενο σύλληψης μπορεί να λάβει διάφορα φίλτρα έτσι ώστε να αποθηκεύονται μόνο συγκεκριμένα εισερχόμενα πακέτα. Με την προσθήκη αυτή στο σύστημα θα διαπιστώνεται αν υπάρχει κάποια κακόβουλη ενέργεια και θα αποτρέπεται. Επίσης, θα διασφαλίζεται η ασφάλεια των συσκευών και θα ενισχύεται περισσότερο η ιδιωτικότητα του χρήστη.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Towards a framework for developing extensible iot applications, 2018.

[2] Mohd Abdul Muqeet, Fabia Akbar, and Syed Hussaini. Iot assisted fingerprint based security system using raspberry pi 3. 06 2019.

[3] Ledger Academy. Blockchain applications, 2019.

[4] Ltd ADH Technology Co. Optical fingerprint recognition embedded module gt-511cr, 2016.

[5] Ioannis Andrea, Chrysostomos Chrysostomou, and George Hadjichristofi. Internet of things: Security vulnerabilities and challenges. In *2015 IEEE Symposium on Computers and Communication (ISCC)*, pages 180–187. IEEE, 2015.

[6] Melissa Brown. Osi model: Part one, 2018.

[7] Adam Calihman. The fundamental three layer iot architecture, 2019.

[8] Miguel Pincheira Caro, Muhammad Salek Ali, Massimo Vecchio, and Raffaele Giaffreda. Blockchain-based traceability in agri-food supply chain management: A practical implementation. In *2018 IoT Vertical and Topical Summit on Agriculture-Tuscany (IOT Tuscany)*, pages 1–4. IEEE, 2018.

[9] Cornelius Diekmann, Lars Hupel, Julius Michaelis, Maximilian Haslbeck, and Georg Carle. Verified iptables firewall analysis and verification. *Journal of automated reasoning*, 61(1-4):191–242, 2018.

[10] SQLAlchemy 1.3 Documentation. Engine configuration, 2017.

[11] Ali Dorri, Clemence Roulin, Raja Jurdak, and Salil S Kanhere. On the activity privacy of blockchain for iot. In *2019 IEEE 44th Conference on Local Computer Networks (LCN)*, pages 258–261. IEEE, 2019.

[12] Tiago M Fernández-Caramés and Paula Fraga-Lamas. A review on the use of blockchain for the internet of things. *IEEE Access*, 6:32979–33001, 2018.

[13] Naman Gupta, Vinayak Naik, and Srishti Sengupta. A firewall for internet of things. In *2017 9th International Conference on Communication Systems and Networks (COMSNETS)*, pages 411–412. IEEE, 2017.

[14] Kashif Habib, Arild Torjusen, and Wolfgang Leister. A novel authentication framework based on biometric and radio fingerprinting for the iot in ehealth. In *Proceedings of International Conference on Smart Systems, Devices and Technologies (SMART)*, pages 32–37, 2014.

[15] Donhee Han, Hongjin Kim, and Juwook Jang. Blockchain based smart door lock system. In *2017 International conference on information and communication technology convergence (ICTC)*, pages 1165–1167. IEEE, 2017.

[16] Shivayogi Hiremath, Geng Yang, and Kunal Mankodiya. Wearable internet of things: Concept, architectural components and promises for person-centered healthcare. In *2014 4th International Conference on Wireless Mobile Communication and Healthcare-Transforming Healthcare Through Innovations in Mobile and Wireless Technologies (MOBIHEALTH)*, pages 304–307. IEEE, 2014.

[17] Huawei. What is nat, 2019.

[18] IoTONE. Universal plug and play (upnp), 2020.

[19] Piyasat Nilkaew Jeffrey S. Beasley. Networking essentials: Interconnecting the lans, 2016.

[20] Bogdan Jeliskoski, Biljana Stojcevska, and Adrijan Bozinovski. Securing a home network by using raspberry pi as a vpn gateway.

[21] Sehan Kim, Meonghun Lee, and Changsun Shin. Iot-based strawberry disease prediction system for smart farming. *Sensors*, 18(11):4051, 2018.

[22] J.D. King. Securing the internet of things, 2016.

[23] J Sathish Kumar and Dhiren R Patel. A survey on internet of things: Security and privacy issues. *International Journal of Computer Applications*, 90(11), 2014.

[24] Seema Kumari and Sumit Kumar Yadav. Development of iot based smart animal health monitoring system using raspberry pi. *International Journal of Advanced Studies of Scientific Research*, 3(8), 2018.

[25] Phillip A Laplante and Nancy Laplante. The internet of things in healthcare: Potential applications and challenges. *It Professional*, 18(3):2–4, 2016.

[26] Jie Lin, Wei Yu, Nan Zhang, Xinyu Yang, Hanlin Zhang, and Wei Zhao. A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications. *IEEE Internet of Things Journal*, 4(5):1125–1142, 2017.

[27] Lilla Nagy and Adrian Coleşa. Router-based iot security using raspberry pi. In *2019 18th RoEduNet Conference: Networking in Education and Research (RoEduNet)*, pages 1–6. IEEE, 2019.

[28] Umakanta Nanda and Sushant Pattnaik. Universal asynchronous receiver and transmitter (uart). pages 1–5, 01 2016.

[29] Vilmos Nebehaj. python-iptables documentation release 0.4.0-dev, 2017.

[30] Marcus Oppitz and Peter Tomsu. Internet of things. In *Inventing the Cloud Century*, pages 435–469. Springer, 2018.

[31] Rabi Prasad Padhy, Manas Ranjan Patra, and Suresh Chandra Satapathy. Cloud computing: security issues and research challenges. *International Journal of Computer Science and Information Technology & Security (IJCSITS)*, 1(2):136–146, 2011.

[32] Mookyu Park, Haengrok Oh, and Kyungho Lee. Security risk measurement for information leakage in iot-based smart homes from a situational awareness perspective. *Sensors*, 19(9):2148, 2019.

[33] Robert Picard. Advanced patterns for views and routing, 2014.

[34] Minwoo Ryu, Jaeseok Yun, Ting Miao, Il-Yeup Ahn, Sung-Chan Choi, and Jaeho Kim. Design and implementation of a connected farm for smart farming system. In *2015 IEEE SENSORS*, pages 1–4. IEEE, 2015.

[35] Jordi Sapes and Francesc Solsona. Fingerscanner: Embedding a fingerprint scanner in a raspberry pi. *Sensors*, 16(2):220, 2016.

[36] CG Sarika, A Bharathi Malakreddy, and HN Harinath. Iot-based smart login using biometrics. In *International Conference on Computer Networks and Communication Technologies*, pages 589–597. Springer, 2019.

[37] Adnan Ahmed Abi Sen, Fathy Albouraey Eassa, Kamal Jambi, and Mohammad Yamin. Preserving privacy in internet of things: a survey. *International Journal of Information Technology*, 10(2):189–200, 2018.

[38] Dhvani Shah and V Haradi. Iot based biometrics implementation on raspberry pi. *Procedia Computer Science*, 79:328–336, 2016.

[39] Bhisham Sharma and Karan Bajaj. Packet filtering using ip tables in linux. *International Journal of Computer Science Issues*, 8:320–325, 07 2011.

[40] Bhisham Sharma and Sanmeet Guide Bhatia. *Packet filtering using IP tables in Linux*. PhD thesis, 2010.

[41] Hui Suo, Jiafu Wan, Caifeng Zou, and Jianqi Liu. Security in the internet of things: a review. In *2012 international conference on computer science and electronics engineering*, volume 3, pages 648–651. IEEE, 2012.

[42] Feng Tian. An agri-food supply chain traceability system for china based on rfid & blockchain technology. In *2016 13th international conference on service systems and service management (ICSSSM)*, pages 1–6. IEEE, 2016.

[43] K Vadivukarasi and S Krithiga. Home security system using iot. *International Journal of Pure and Applied Mathematics*, 119(15):1863–1868, 2018.

[44] Andrew Whitmore, Anurag Agarwal, and Li Da Xu. The internet of things—a survey of topics and trends. *Information Systems Frontiers*, 17(2):261–274, 2015.

[45] Weizhe Zhang and Baosheng Qu. Security architecture of the internet of things oriented to perceptual layer. *International Journal on Computer, Consumer and Control (IJ3C)*, 2(2):37–45, 2013.

[46] Guoqing Zhao, Shaofeng Liu, Carmen Lopez, Haiyan Lu, Sebastian Elgueta, Huilan Chen, and Biljana Mileva Boshkoska. Blockchain technology in agri-food value chain management: A synthesis of applications, challenges and future research directions. *Computers in Industry*, 109:83–99, 2019.