Ατομική Διπλωματική Εργασία

ΑΠΟΚΑΤΑΣΤΑΣΗ ΚΟΜΒΟΥ ΚΑΙ ΔΙΚΤΥΟΥ ΣΤΟ ΔΙΑΔΙΚΤΥΟ ΤΩΝ ΠΡΑΓΜΑΤΩΝ

Πάρης Κωνσταντινίδης

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2020

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Αποκατάσταση Κόμβου και Δικτύου στο Διαδίκτυο των Πραγμάτων

Πάρης Κωνσταντινίδης

Επιβλέπων Καθηγητής
Βάσσος Βασιλείου

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2020

Ευχαριστίες

Η παρούσα εργασία αποτελεί διπλωματική εργασία στα πλαίσια του προπτυχιακού προγράμματος Πληροφορικής της σχολής Θετικών Επιστημών του Πανεπιστημίου Κύπρου.

Με την ολοκλήρωση της διπλωματικής μου εργασίας, θα ήθελα να ευχαριστήσω ορισμένους από τους ανθρώπους που γνώρισα, συνεργάστηκα μαζί τους και έπαιξαν πολύ σημαντικό ρόλο στην πραγματοποίηση της.

Ευχαριστώ θερμά τον επιβλέποντα καθηγητή της διπλωματικής μου εργασίας Δρ. Βάσσο Βασιλείου που μου εμπιστεύτηκε την παρούσα διπλωματική εργασία, για την ώθηση που μου έδωσε να ασχοληθώ με ένα τόσο ενδιαφέρον θέμα, αλλά και την συνεχή υποστήριξη. Ακόμη θα ήθελα να τον ευχαριστήσω για την καθοδήγηση που μου παρείχε καθ' όλη την διάρκεια των σπουδών μου.

Ευχαριστώ επίσης την διδάκτορα Χριστιάνα Ιωάννου ερευνήτρια στο NetRL Lab του Πανεπιστημίου Κύπρου για τις κατευθυντήριες γραμμές για την ολοκλήρωση της προγραμματιστικής πτυχής της διπλωματικής εργασίας αλλά και την συνεχή ανταπόκριση της σε όποια ζητήματα προέκυπταν και στην βοήθεια που μου παρείχε για την επίλυση τους.

Περίληψη

Η ασφάλεια αποτελεί καθοριστικό παράγοντα στο τομέα του Διαδικτύου των Πραγμάτων (Internet of Things) για να διασφαλιστεί η αξιόπιστη και χρονικά ομαλή λειτουργία του. Αυτή η διπλωματική εργασία έχει ως επίκεντρο τις επιθέσεις που γίνονται στο επίπεδο της δρομολόγησής στα δίκτυα IoT (Internet of Things), με στόχο την αντιμετώπιση τέτοιων επιθέσεων όταν γίνουν αντιληπτές. Ένας κακόβουλος κόμβος μπορεί να επηρεάσει την απόδοση ενός δικτύου των πραγμάτων με την πραγματοποίηση διάφορων επιθέσεων. Η παρουσία ενός αναξιόπιστου κόμβου σε ένα δίκτυο με τις ιδιαιτερότητες του δικτύου IoT έχει την δυνατότητα να μειώσει την αποδοτικότητα της εφαρμογής που εκτελείται στο δίκτυο με το να αποτρέπει την αποστολή δεδομένων στον προορισμό τους.

Όταν εντοπισθεί σε ένα δίκτυο ένας κακόβουλος κόμβος, μια σουίτα πρωτοκόλλων ενεργοποιείται για να αποτραπεί η λειτουργία του ανεπιθύμητου κόμβου στο δίκτυο και να επανέλθει το δίκτυο στην ομαλότητα. Πρόκειται για μια τοπική μέθοδο, δηλαδή εκτελείται σε κάθε κόμβο στο δίκτυο και ενεργοποιείται όταν αυτό καταστεί αναγκαίο, με δύο επιμέρους υπορουτίνες. Συγκεκριμένα η πρώτη υπορουτίνα είναι να ειδοποιηθούν οι γειτνιάζοντες κόμβοι έτσι ώστε να αποτραπεί η αποστολή δεδομένων στον εισβολέα, ενώ κατά την εκτέλεση της δεύτερης υπορουτίνας ο αναξιόπιστος κόμβος επαναφέρεται στην κατάσταση ομαλή λειτουργίας.

Η πιο πάνω μέθοδος υλοποιήθηκε και ελέγχθηκε σε προσομοιωτή δικτύου IoT. Τα πειράματα που εκτελέστηκα σε περιβάλλον προσομοίωσης έδειξαν ότι η σουίτα πρωτοκόλλων δύναται να επαναφέρει την συνδεσιμότητα του δικτύου αλλά και τον κακόβουλο κόμβο σε ομαλή λειτουργία σε σύντομο χρονικό διάστημα και με χαμηλή ροή απώλειας δεδομένων.

Περιεχόμενα

Κεφάλαιο 1

1 Εισαγωγή

1.1 Διαδίκτυο των Πραγμάτων (Internet of Things)

Το Internet of Things (IoT) έχει ως κύριο μέλημα να απλοποιήσει την καθημερινότητα αυτοματοποιώντας καθημερινές λειτουργίες. Είναι ένα διαφοροποιημένο δίκτυο το οποίο εξαρτάται από πολλές τεχνολογίες και τρόπους επικοινωνίας για να κτίσει μια προσαρμοσμένη IoT εφαρμογή. Οι έξυπνες αυτές συσκευές εξοπλισμένες με διάφορους αισθητήρες και ηλεκτρικούς ενεργοποιητές, γνωστούς και ως actuators, είναι υπεύθυνες για να παρακολουθούν ένα περιβάλλον και να λαμβάνουν δράση όταν αυτό καταστεί αναγκαίο. Συχνά αυτές οι συσκευές είναι συνδεδεμένες κατευθείαν στο διαδίκτυο ενώ πολλές φορές δημιουργούν οι ίδιες μιας μορφής δίκτυο και αποστέλλουν/λαμβάνουν πληροφορίες δια μέσου μια κεντρικής πύλης (Gateway) ή ενός Sink node. Επομένως ο τελικός χρήστης μπορεί να έχει πρόσβαση μέσου του διαδικτύου σε αυτές.

1.2 Ασφάλεια στο Διαδίκτυο των Πραγμάτων

Η ασφάλεια αποτελεί ένα σημαντικό πυλώνα στα IoT δίκτυα για να διασφαλίσει την εμπιστευτικότητα, την ακεραιότητα και την διαθεσιμότητα των πόρων. Ένας εισβολέας μπορεί να παρακολουθήσει την ροή των πληροφοριών (eavesdrop) ή ακόμη να αποστείλει μη έγκυρες πληροφορίες και να αλλοιώσει δεδομένα που στέλνουν άλλοι στο δίκτυο. Η εμπιστευτικότητα διασφαλίζεται εάν ο επιτιθέμενος δεν μπορεί να κατανοήσει την πληροφορία που αποστέλλεται, κύριος τρόπος που επιτυγχάνεται είναι η κρυπτογράφηση της πληροφορίας. Για να αποτραπεί η αλλοίωση δεδομένων και η αυθεντικότητα τους χρησιμοποιούνται συνήθως τεχνικές ψηφιακής υπογραφής και συναρτήσεις κατακερματισμού κρυπτογραφίας. Τέλος για

την διασφάλιση της διαθεσιμότητας, τοίχος προστασίας εγκαθίσταται στην κεντρική πύλη που συνδέεται με το διαδίκτυο.

Ένας εκτεθειμένος κόμβος σε ένα δίκτυο μπορεί να προκαλέσει πολλά προβλήματα. Για παράδειγμα, μπορεί να αποτρέψει πληροφορίες από το να μεταδοθούνε στο υπόλοιπο δίκτυο, να συμβάλει στην μετάδοση λανθασμένης πληροφορίας ή ακόμη και να συντείνει στην εξάντληση των πόρων των υπόλοιπων συσκευών στο δίκτυο. Αυτό συμβαίνει λόγω των περιορισμένων πόρων που έχουν οι συσκευές αυτές, καθώς λειτουργούν με μπαταρία και έχουν περιορισμένη υπολογιστική δύναμη. Τα IoT δίκτυα μπορούν να εκτεθούν και σε επιθέσεις που γίνονται σε παραδοσιακά δίκτυα όντας συνδεδεμένα με το διαδίκτυο όπου χρησιμοποιούν παραδοσιακά πρωτόκολλα.

Ωστόσο, λόγω της φύσης του, οι συσκευές αυτές δύναται να υποστούν και άλλου τύπου επιθέσεις. Τα τρέχων IoT συστήματα, εφαρμόζουν μια μεγάλη γκάμα από επιλογές σύνδεσης δικτύου, πρωτόκολλα (ιδιωτικά ή βασισμένα σε πρότυπα) και επικοινωνιακές μεθόδους. Η απουσία ασφαλών προτύπων επικοινωνίας και πρωτοκόλλων, μαζί με την αύξηση των έξυπνων συσκευών καθιστούν την ασφάλεια στον χώρο ένα δύσκολο έργο. Οι επιθέσεις στην δρομολόγηση που εκμεταλλεύονται την αδυναμία των αλγορίθμων δρομολόγησης, υποχρεώνουν την ασφάλεια να τις αντιμετωπίσει, κάτι που πολλές φορές δεν είναι ευκατόρθωτο αφού αποτρέπουν την πηγή ή την κεντρική πύλη από το να παραλάβει όλο το επιθυμητό φάσμα δεδομένων. Αναλόγως της τοποθεσίας του εκτεθειμένου κόμβου, η πηγή μπορεί να παραλαμβάνει μέχρι και τα μισά δεδομένα [7]. Επομένως κρίνεται επιτακτική η ανάγκη για την δημιουργία μιας μεθοδολογίας που θα αυξάνει την άμυνα του δικτύου απέναντι σε τέτοιες επιθέσεις.

Υπάρχουν κυρίως τρία επίπεδα ασφάλειας σε όλα τα πληροφοριακά συστήματα: (α) της αποτροπής, (β) της ανίχνευσης και (γ) της αντίδρασης. Το επίπεδο της αποτροπής περιλαμβάνει τεχνικές όπως το τείχος προστασίας που επιβλέπει το δίκτυο και δεν επιτρέπει σε ύποπτα πακέτα να εισέλθουν σε αυτό. Στο επίπεδο ανίχνευσης, ανιχνεύεται πιθανή κακόβουλη συμπεριφορά, από δεδομένα που κατάφεραν να εισέλθουν στο δίκτυο από το πρώτο επίπεδο. Η τοποθεσία των συστημάτων ανίχνευσης εξαρτάται από τον περιορισμό των έξυπνων συσκευών σε ενέργεια και μνήμη, καθώς και του περιβάλλοντος εγκατάστασης τους [5] [7]. Στην τοπική ανίχνευση απαλείφονται οι όποιες εξαρτήσεις από απώλεια δεδομένων στο φυσικό επίπεδο, παρακολουθώντας μόνο την λειτουργία του κόμβου. Στις περιπτώσεις όπου αυτό το σύστημα εγκαθίσταται στην κεντρική πύλη του δικτύου που υπάρχουν πιο πολλές δυνατότητες υπολογιστικής ισχύς, ο κόμβος δύναται να παρακολουθεί τα δεδομένα που αποστέλλονται μεταξύ των έξυπνων συσκευών και του διαδικτύου. Όταν καταστεί

ανιχνεύσιμη μια κακόβουλη συμπεριφορά, το επίπεδο αντίδρασης ενεργοποιείται. Το είδος της προειδοποίησης και το μέγεθος της βλάβης που προκαλείται αξιολογείται είτε χειροκίνητα είτε αυτόματα και οι διαδικασίες αντιδράσεις ενεργοποιούνται.

1.3 Συνεισφορά

Σε ένα περιβάλλον έξυπνων συσκευών, τεχνικές αντίδρασης σε επιθέσεις μπορεί να είναι είτε η χειροκίνητη, είτε η δυναμική επαναφορά του προβληματικού κόμβου, ή ακόμη και η φυσική αντικατάσταση του με ένα άλλο κόμβο. Η δυναμική αντίδραση σε μια επίθεση θεωρείται καλύτερη καθώς είναι πιο αποτελεσματική σε θέμα χρόνου και κόστους, μιας και οι έξυπνες συσκευές μπορεί να είναι εγκατεστημένες σε μη προσβάσιμο και μη ελεγχόμενο περιβάλλον.

Στην παρούσα διπλωματική εργασία προτείνεται μια αντιδραστική (reactive) λύση αποκατάστασης ασφαλείας τοπικού κόμβου για επιθέσεις στο επίπεδο δρομολόγησης στα δίκτυα IoT, η οποία ενεργοποιείται όταν ανιχνευτεί κακόβουλη συμπεριφορά. Ο μηχανισμός της επαναδρομολόγησης ειδοποιεί τους γείτονες της έξυπνης συσκευής για την ύπαρξη κακόβουλης συμπεριφοράς, και ακολούθως, τους αναγκάζει να επιλέξουν ένα εναλλακτικό ασφαλές μονοπάτι για την δρομολόγηση των δεδομένων τους. Ο μηχανισμός της επαναφοράς του κόμβου τον επαναφέρει στην αρχική ομαλή επιθυμητή λειτουργία του, δηλαδή πριν την ύπαρξη κακόβουλης συμπεριφοράς.

Η πιο πάνω μέθοδος υλοποιήθηκε και εγκαταστάθηκε σε κόμβους με περιορισμένη ενεργειακή και υπολογιστική ισχύ. Μετά την ανάλυση διαπιστώθηκε ότι με την χρήση της μειώθηκε μέχρι και 53,66% η απώλεια δεδομένων σε σχέση με σενάρια που η μέθοδος δεν χρησιμοποιείτο.

1.4 Δομή Διπλωματικής Εργασίας

Η υπόλοιπη διπλωματική εργασία ακολουθεί την πιο κάτω δομή: το Κεφάλαιο 2 παρουσιάζει σχετική δουλεία. Το Κεφάλαιο 3 παρουσιάζει τις επιθέσεις που αναλύθηκαν και χρησιμοποιήθηκαν στα πλαίσια της ατομικής διπλωματικής εργασίας. Το Κεφάλαιο 4 παρουσιάζει την προτεινόμενη λύση και μεθοδολογία. Το Κεφάλαιο 5 παρουσιάζει το πειραματικό περιβάλλον και τα αποτελέσματα. Τέλος στο Κεφάλαιο 6 παρουσιάζονται τα συμπεράσματα και η μελλοντική δουλεία.

Κεφάλαιο 2

2 Σχετική Δουλειά

2.1 Επισκόπηση

Σε αυτό το κεφάλαιο αναλύονται άλλες προτεινόμενες υλοποιήσεις για το επίπεδο ανταπόκρισης, αλλά και ένα σύστημα ανίχνευσης που μπορεί να χρησιμοποιηθεί για την ενεργοποίηση του προτεινόμενου συστήματος ανταπόκρισης.

2.2 Συστήματα Ανίχνευσης

Τα συστήματα ανίχνευσης εισβολής (IDS) χρησιμοποιούνται ευρέως για τον εντοπισμό μη εξουσιοδοτημένης ή κακόβουλης συμπεριφοράς σε ένα δίκτυο. Κακόβουλη συμπεριφορά ορίζεται ως η συμπεριφορά δικτύου που δημιουργείται από συμβιβασμένους κόμβους με σκοπό να διακόψει ή / και να θέσει σε κίνδυνο την αποστολή ενός δικτύου [11]. Η συμπεριφορά των επιθέσεων εξαρτάται από το επίπεδο δικτύου στο οποίο στοχεύουν και τον στόχο τους. Υπάρχουν δύο κύριες τεχνικές ανίχνευσης. Συγκεκριμένα η ανίχνευση μοτίβου, που προσδιορίζει γνωστές επιθέσεις και η ανίχνευση ανωμαλιών, η οποία προσδιορίζει γνωστές, αλλά το πιο σημαντικό, νέες επιθέσεις, μέσα στο δίκτυο.

Το mIDS [4], είναι ένα σύστημα ανίχνευσης (IDS) που χρησιμοποιεί ανίχνευση ανωμαλιών, με βάση το Binary Logistic Regression (BLR), για να ανιχνεύσει την παρουσία μιας επίθεσης τοπικά σε κάθε περιορισμένο κόμβο σε ένα δίκτυο των πραγμάτων. Το mIDS χρησιμοποιεί καλοήθη και κακόβουλα δεδομένα από το επίπεδο δικτύου δρομολόγησης κάθε κόμβου για να αντλήσει τις μετρικές ανίχνευσης.

Το mIDS, μπορεί να χρησιμοποιηθεί για το επίπεδο ανίχνευσης κάποιας κακόβουλης συμπεριφοράς και με την σειρά του να ενεργοποιήσει το επίπεδο αντίδρασης που παρουσιάζεται στην παρούσα ατομική διπλωματική εργασία.

2.3 Άλλες υλοποιήσεις και προτάσεις για το επίπεδο αντίδρασης

Όταν στο επίπεδο ανίχνευσης εντοπισθεί μια μη ομαλή συμπεριφορά στο δίκτυο, το επίπεδο αντίδρασης ενεργοποιείται για να επαναφέρει το δίκτυο στην αρχική του ομαλή λειτουργία. Πολλά πρωτόκολλα έχουν προταθεί [15] [18] [19] [23] [25] [26] [29] – [31] τα οποία έχουν ως σκοπό την δημιουργία εναλλακτικών μονοπατιών για ανταποκριθούν σε επιθέσεις που γίνονται στο επίπεδο της δρομολόγησης. Κάποιες από αυτές προκαθορίζουν από πριν το μονοπάτι ως εναλλακτικό ενώ άλλες δημιουργούν το νέο μονοπάτι όταν παραστεί ανάγκη. Στις περιπτώσεις που προκαθορίζεται το μονοπάτι δηλαδή έχουμε μια προληπτική αντίδραση οι κόμβοι αποστέλλουν προς όλες τις διαδρομές τα πακέτα τους δημιουργώντας αχρείαστη συμφόρηση στο δίκτυο. Οι αντιδραστικές μέθοδοι ενεργοποιούνται δυναμικά όταν χρειαστεί και το πρωτόκολλο δρομολόγησης ψάχνει για καινούργια μονοπάτια. Παρόλα αυτά στις ήδη προτεινόμενες λύσεις οι αντιδραστικές μέθοδοι είναι είτε επεκτάσεις υφιστάμενων πρωτοκόλλων είτε καινούργια πρωτόκολλα δρομολόγησης.

Πολλά πρωτόκολλα αντίδρασης έχουν προταθεί για επιθέσεις στο επίπεδο δρομολόγησης αλλά πολλά από αυτά αφορούν τα Mobile Ad Hoc δίκτυα όπου δεν υφίστανται περιορισμοί στο εύρος ζώνης του φυσικού μέσου, την κατανάλωση ενέργειας και την υπολογιστική ισχύ [29] – [31]. Κάποια από αυτά αφορούν τα IoT δίκτυα αλλά δεν αμύνονται σε όλα τα πιθανά σενάρια επιθέσεων [19].

Το M-RPL (Multipath - RPL) είναι μια επέκταση του υφιστάμενου πρωτοκόλλου δρομολόγησης RPL, από την τοπολογία κατευθυνόμενου άκυκλου γραφήματος στην τοπολογία ιεραρχικής ομαδοποίησης [19] [28]. Στο M-RPL σχηματίζονται πολλαπλές ομάδες που προσφέρουν υψηλότερα ποσοστά άφιξης δεδομένων έναντι κοινών επιθέσεων δρομολόγησης. Κάθε θυγατρικός κόμβος έχει περισσότερους από έναν κόμβο κεφαλής ως γονέα, επομένως υιοθετεί μια ιεραρχική τοπολογία και μια στρατηγική δρομολόγησης πολλαπλών διαδρομών για να αντιστέκεται σε κοινές επιθέσεις δρομολόγησης. Δηλαδή, κάθε θυγατρικός κόμβος επιλέγει δύο γονικούς κόμβους κεφαλής και στέλνει τα πακέτα του μέσω δύο διαδρομών προς τον προορισμό για να είναι ανθεκτικό όταν υπάρχει κάποια επίθεση σε μια από τις δύο διαδρομές. Έχει δοκιμαστεί κατά των επιθέσεων Blackhole και Wormhole και έχει δείξει ότι μπορεί να αμυνθεί όταν υπάρχουν τέτοιες επιθέσεις σε ένα κόμβο. Ωστόσο, οι

επιθέσεις σε πολλαπλούς κόμβους κλειδιά στην τοπολογία του δικτύου μπορεί να δημιουργήσουν πρόβλημα. Για παράδειγμα, σε ένα σενάριο όπου ένας θυγατρικός κόμβος στέλνει ένα πακέτο μέσω των δύο προεπιλεγμένων διαδρομών του και οι δύο γονείς του είναι κάτω από την επιρροή κάποια επίθεσης, δεν υπάρχει μέθοδος ανάκτησης. Επιπλέον, αυτό έρχεται με το κόστος ότι κάθε φορά που ένα πακέτο αποστέλλεται στο δίκτυο, χρήσιμο εύρος ζώνης και ισχύ σπαταλιέται, αφού κάθε πακέτο αποστέλλεται δύο φορές ακόμη και σε κανονική συμπεριφορά στο δίκτυο. Η προτεινόμενη λύση ασφαλείας που εφαρμόστηκε σε αυτή την διπλωματική εργασία έχει την δυνατότητα όχι μόνο να υπερασπιστεί το δίκτυο ενάντια σε επιθέσεις όπου πολλοί κακόβουλοι κόμβοι τοποθετούνται σε βασικές θέσεις σε αυτό, αλλά και δεν διαδίδει τόση συμφόρηση.

Μια άλλη ενδιαφέρουσα προσέγγιση στο [30] είναι ότι οι κόμβοι αισθητήρων λαμβάνουν ανατροφοδότηση από τους γείτονές τους και την κεντρική πύλη για να αποφασίσουν τον επόμενο κόμβο δρομολόγησης τους. Με την αξιολόγηση της εργασίας τους σε έναν προσομοιωτή δικτύου έχουν δείξει ότι οι κόμβοι αποφεύγουν τη συμφόρηση χρησιμοποιώντας αυτήν την ανατροφοδότηση και ότι μπορούν επίσης να αποφύγουν τις επιθέσεις Sinkhole και Wormhole χρησιμοποιώντας ανατροφοδότηση από την κεντρική πύλη. Αυτή η μέθοδος όχι μόνο καθιστά τη λύση κεντρικοποιημένη, αλλά περιορίζει επίσης το πρωτόκολλο δρομολόγησης για την άμυνα μόνο έναντι αυτών των συγκεκριμένων επιθέσεων. Άλλες επιθέσεις δρομολόγησης δεν αντιμετωπίζονται από την συγκεκριμένη προσέγγιση.

Στο άρθρο με τίτλο «Risk-Aware Response for Mitigating MANET Routing Attacks» [31], οι συγγραφείς πρότειναν έναν μηχανισμό συνειδητοποίησης κινδύνου για την απομόνωση κακόβουλων κόμβων. Ο μηχανισμός τους αξιολογεί τον κίνδυνο απομόνωσης του κόμβου από το δίκτυο και εάν το δίκτυο θα χωριστεί απομονώνοντας τον κόμβο, δηλαδή εάν κάποιος κόμβος δεν θα μπορεί να επικοινωνήσει με κάποιο άλλο, δεν το πράττει. Παρόλα αυτά, το να βασίζεσαι σε έναν κακόβουλο κόμβο για να δρομολογήσεις τα πακέτα σου δεν είναι λύση στο πρόβλημα για την αποφυγή πρόσθετων ζημιών στο δίκτυο, καθώς οι επιθέσεις τύπου eavesdropping μπορούν να το εκμεταλλευτούν. Στην εργασία μου παρουσιάζεται μια λύση που όχι μόνο απομονώνει την κακόβουλη συμπεριφορά αλλά και ανακτά από αυτήν.

Συνοψίζοντας, έχουν ήδη προταθεί τεχνικές απομόνωσης του κακόβουλου κόμβου και από άλλες ερευνητικές εργασίες [20] [26]. Σε αυτή την εργασία αξιολογείται ο χρόνος αποκατάστασης ενός δικτύου εφαρμόζοντας την τεχνική επαναδρομολόγησης δηλαδή της αποφυγής του απομονωμένου κόμβου, συνδυάζοντας το με μια τεχνική επανεκκίνησης προκειμένου ο κακόβουλος κόμβος να ανακάμψει πλήρως, χρησιμοποιώντας το Contiki OS

[13] και εκτελέσεις πραγματικού χρόνου στο προσομοιωτή COOJA [17] αλλά και εκτελέσεις σε πραγματικούς κόμβους με την χρήση της πλατφόρμας FIT-IOT Lab στην Γαλλία [2].

Κεφάλαιο 3

3 Επιθέσεις στο επίπεδο δρομολόγησης στα IoT δίκτυα

3.1 Επισκόπηση

Οι επιθέσεις στο επίπεδο δρομολόγησης έχουν ως κύριο μέλημα την διάσπαση του κανονικού μονοπατιού από τους κόμβους προς τον τελικό προορισμό. Αυτό μπορεί να επιτευχθεί με την εκμετάλλευση του πρωτόκολλου δρομολόγησης για να προσελκύσουν πακέτα προς αυτούς χωρίς πραγματικά να είναι οι παραλήπτες για να δουν το περιεχόμενο, να το αλλοιώσουν ή ακόμη και για να τα απορρίψουν. Σε αυτό το κεφάλαιο αναλύονται κάποιες από τις επιθέσεις στο επίπεδο δρομολόγησης στις οποίες έγινε η αξιολόγηση της προτεινόμενης μεθόδου.

3.2 Sinkhole

Η επίθεση Sinkhole είναι μια από τις πιο σοβαρές επιθέσεις στο επίπεδο δρομολόγησης. Ένας Sinkhole κόμβος παρασύρει όλα τα δεδομένα προς αυτόν προσποιούμενος την κεντρική πύλη, επομένως οι υπόλοιποι κόμβοι δεν γνωρίζουν ότι τα πακέτα τους αποστέλλονται προς κάποιο κακόβουλο κόμβο. Οι επιθέσεις τέτοιου είδους μπορούν να πραγματοποιηθούν είτε με επανάληψη πακέτων του επιπέδου δρομολόγησης που χρησιμοποιεί η ίδια η κεντρική πύλη είτε με αλλοίωση τέτοιων πακέτων κάνοντας πιο ελκυστικό τον εκτεθειμένο κόμβο από τους υπόλοιπους. Ο εισβολέας μπορεί σε μεταγενέστερο στάδιο να παραβιάσει την δρομολόγηση των πακέτων, να μεταδώσει ψευδείς πληροφορίες ή ακόμη και ψευδείς αναφορές επίθεσης για να διαταράξει περαιτέρω το δίκτυο [10] [24] [27]. Μια τέτοια επίθεση μπορεί ακόμη να προκαλέσει εξάντληση ενέργειας στους γειτονικούς κόμβους, με αποτέλεσμα κομμάτια του δικτύου να αποκοπούν από την κεντρική πύλη [1]. Η καλύτερη τοποθεσία για ένα τέτοιο κόμβο είναι κοντά στην πηγή για να προσελκύσει όσο το δυνατό περισσότερα δεδομένα. Αποδείχθηκε

στο άρθρο [7] ότι, ανάλογα με την τοπολογία του δικτύου, ακόμη και όταν ο κακόβουλος κόμβος βρίσκεται πιο μακριά από την κεντρική πύλη, μπορεί να στερήσει από την κεντρική πύλη τη δυνατότητα λήψης πακέτων από μεγάλα τμήματα του δικτύου.

3.3 Selective Forward

Σε μια επίθεση τέτοιου είδους όπως αναφέρεται και στο όνομα, η κακόβουλη συσκευή επιλέγει ποια πακέτα θα προωθήσει και ποια όχι. Αυτό είναι εφικτό λόγω του ότι σε πολλές τοπολογίες οι ενδιάμεσοι κόμβοι προωθούν τα πακέτα των γειτόνων τους προς τον τελικό τους προορισμό που συνήθως αυτός είναι η κεντρική πύλη. Η επιλογή για το ποια πακέτα θα προωθηθούν και ποια θα απορριφθούν, μπορεί να είναι τυχαία ή να αποκλείει συγκεκριμένο γείτονα. Στόχος της επίθεσης είναι να αυξήσει τα ποσοστά απώλειας του δικτύου και να κρατήσει άλλες κρίσιμες πληροφορίες.

Στην παρούσα διπλωματική εργασία χρησιμοποιήθηκαν δυο παραλλαγές της επίθεσης: Forwarding Radio (FR) και Block Node (BN). Η παραλλαγή Forwarding Ratio απορρίπτει πακέτα σε μια προκαθορισμένη πιθανότητα *r*. Για τα συγκεκριμένα πειράματα το *r* έχει την τιμή *0.5*. Δηλαδή όταν ένα κόμβος πρέπει να προωθήσει κάποιο πακέτο σε ένα γειτονικό κόμβο το προωθεί με πιθανότητα (*1-r*) ή το απορρίπτει με πιθανότητα *r*. Στην παραλλαγή Block Node, στοχοποιείτε ένας συγκεκριμένος γείτονας για να μην προωθούνται τα πακέτα του. Η επιλογή του γείτονα είναι τυχαία και αλλάζει κατά την εκτέλεση του πειράματος, και ο επιτιθέμενος μπορεί να αλλάζει τα θύματα περιοδικά. Για την συγκεκριμένη εργασία, ο χρόνος αποκλεισμού ορίστηκε ως η μετάδοση δέκα πακέτων, δηλαδή όταν αποτρέψει ένα θύμα από το να στείλει δέκα πακέτα, προχωρά σε επόμενο γείτονα για να αποκλείσει.

3.4 Blackhole

Η επίθεση τύπου Blackhole (BH) κατευθύνει τα πακέτα του δικτύου προς τον εκτεθειμένο κόμβο διαφημίζοντας μηδενικό ή χαμηλό κόστος δρομολόγησης. Έτσι οι γείτονες του κόμβου αυτού δρομολογούν τα πακέτα τους προς το εκτεθειμένο κόμβο όπου επιθέσεις τύπου eavesdropping, ή denial of service μπορεί να λαμβάνουν χώρα. Συνήθως συνδυάζεται με την επίθεση Selective Forward για να την κάνει πιο ισχυρή και να επηρεάσει μεγαλύτερο εύρος στο δίκτυο με αποτέλεσμα να απορρίψει πιο μεγάλο αριθμό πακέτων [7]. Στη συγκεκριμένη εργασία οι κόμβοι με επίθεση Blackhole διαφημίζουν ότι είναι άμεση γείτονες με την κεντρική πύλη.

Κεφάλαιο 4

4 Μεθοδολογία

4.1 Επισκόπηση

Προτείνονται δύο τεχνικές αποκατάστασης, η επαναδρομολόγηση και η επανεκκίνηση. Η τεχνική εκ νέου δρομολόγησης απομακρύνει τα πακέτα από τον κακόβουλο κόμβο ενημερώνοντας τους γείτονές του για την ύπαρξη του. Η τεχνική επανεκκίνησης ενεργοποιείται μετά την τεχνική της εκ νέου δρομολόγησης και επαναφέρει τον κακόβουλο κόμβο στην καλοήθη συμπεριφορά. Τα πιο πάνω υλοποιήθηκαν στο Contiki OS [13] και αξιολογήθηκαν στον προσομοιωτή COOJA [17]. Το τρέχον κεφάλαιο περιγράφει τις προτεινόμενες τεχνικές ανάκτησης και το πειραματικό πλαίσιο που χρησιμοποιήθηκε για την αξιολόγηση τους.

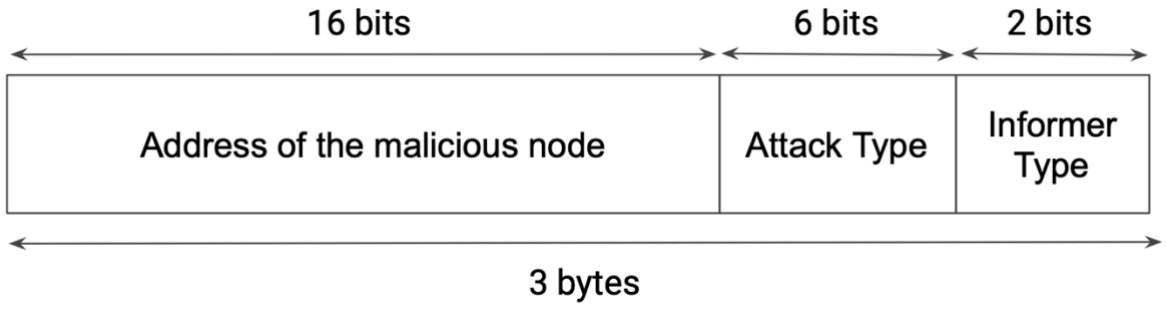
4.2 Weighed Shortest Path Routing Protocol

Οι τεχνικές αποκατάστασης αξιολογήθηκαν χρησιμοποιώντας ένα απλό πρωτόκολλο δρομολόγησης που ονομάζεται WSP και βασίζεται στην έννοια του συντομότερου μονοπατιού. Έχουν προταθεί πολλά πρωτόκολλα δρομολόγησης για IoT δίκτυα. Για να αξιολογηθούν οι τεχνικές ανάκτησης ανεξάρτητα από τους περιορισμούς ή/και τις ιδιαιτερότητες του πρωτοκόλλου δρομολόγησης, επιλέχθηκε ένα απλό πρωτόκολλο δρομολόγησης με ελάχιστη επιβάρυνση στην αποστολή δεδομένων για την δημιουργία της τοπολογίας.

Το WSP επιλέγει τον γείτονα για να προωθήσει τα πακέτα του με βάση την απόσταση του προς την κεντρική πύλη και την ισχύ του λαμβανόμενου σήματος. Στην αρχή του σχηματισμού της τοπολογίας του δικτύου κάθε κόμβος δημιουργεί ένα πίνακα γειτνίασης, ακούγοντας «ανακοινώσεις» από γειτονικούς κόμβους. Οι «ανακοινώσεις» αυτές αποστέλλονται περιοδικά από κάθε κόμβο, για την επαλήθευση της θέσης του στο δίκτυο. Ο πίνακας γειτνίασης ενημερώνεται όταν υπάρξει κάποια αλλαγή. Μια καταχώρηση στον πίνακα περιλαμβάνει την μοναδική ταυτότητα του κόμβου, στην συγκεκριμένη περίπτωση το Rime Address του, τον αριθμό των ενδιάμεσων κόμβων που είναι μεταξύ αυτού και της κεντρικής πύλης καθώς και την τιμή ένδειξης ισχύος του σήματος (RSSI). Για τους σκοπούς της τρέχουσας εργασίας προστέθηκε ένα bit ασφαλείας στον πίνακα, το οποίο λαμβάνει τις τιμές *1* για ένα αξιόπιστο κόμβο και *0* για μη αξιόπιστο. Οι κόμβοι αρχικοποιούνται με το bit ασφαλείας σε *1* και αλλάζει σε *0* όταν ληφθεί το λεγόμενο πακέτο συνδέσμου ασφαλείας και υποδεικνύει ότι ο γείτονάς τους δεν είναι αξιόπιστος και δεν πρέπει να εμπιστεύεται για την δρομολόγησης των πακέτων τους.

4.3 Πακέτο Συνδέσμου Ασφαλείας

Ως πρώτο βήμα της προτεινόμενης λύσης ανάκτησης, υλοποιήθηκε ένα αποκλειστικό μήνυμα ασφαλείας, που ονομάστηκε πακέτο συνδέσμου ασφαλείας. Κάθε κόμβος μεταδίδει περιοδικά ένα πακέτο συνδέσμου ασφαλείας για να ενημερώνει τους γείτονες του ότι δεν έχει κακόβουλη συμπεριφορά. Όταν αυτή η κατάσταση αλλάξει, δηλαδή όταν εντοπισθεί από το σύστημα ανίχνευσης κακόβουλης συμπεριφοράς μια επίθεση, ο κόμβος μεταδίδει για μια προκαθορισμένη περίοδο ένα πακέτο συνδέσμου ασφαλείας ανά δευτερόλεπτο για να ενημερώσει σχετικά με την παραβίαση. Για την τρέχουσα ρύθμιση, η περίοδος ορίστηκε σε δέκα δευτερόλεπτα.



Εικόνα 4.1: Αρχιτεκτονική Πακέτου Ασφαλείας

Το μέγεθος του πακέτου είναι τρία bytes και έχει τρία μέρη: την μοναδική ταυτότητα του κακόβουλου κόμβου, τον τύπο επίθεσης και τον τύπο πληροφοριοδότη (φαίνεται στην Εικόνα

4.1). Τα δύο πρώτα bytes αντιπροσωπεύουν τη διεύθυνση ή το μοναδικό αναγνωριστικό του προβληματικού κόμβου. Τα επόμενα έξι bits αποτελούν τον τύπο της επίθεσης. Μια καλοήθης κατάσταση αντιπροσωπεύεται με την τιμή *000000*. Κάθε επίθεση αντιπροσωπεύεται από ένα διαφορετικό κωδικό. Όταν έχει ορισθεί μόνο το τελευταίο bit σημαίνει ότι ο κόμβος πάσχει από επίθεση τύπου Sinkhole. Όταν έχει την τιμή *1* το τρίτο σε σειρά bit αντιπροσωπεύει μια επίθεση τύπου Blackhole ενώ το τέταρτο bit χρησιμεύει για την αναγνώριση επιθέσεων Selective Forward. Λόγω του ότι επιθέσεις τύπου Sinkhole και Selective Forward δεν μπορούν να συνυπάρχουν, το τελευταίο bit επαναχρησιμοποιείται στον καθορισμό της παραλλαγής της επίθεσης Selective Forward. Πιο συγκεκριμένα, τα τελευταία bit με τιμή *01* υποδηλώνουν επίθεση τύπου Forwarding Ratio, και η τιμή *10* επίθεση τύπου Block Node, ενώ η τιμή *11* μπορεί να χρησιμοποιηθεί για άλλες παραλλαγές. Τα δύο πρώτα bits στο παρόν στάδιο δεν έχουν κάποια εφαρμογή αλλά μπορούν να χρησιμοποιηθούν για τον καθορισμό άλλου τύπου επιθέσεων. Μια πιο λεπτομερή εικόνα των επιθέσεων σε συνδυασμό με τα αντιπροσωπευόμενα bits αναπαριστάτε στον Πίνακας 4.1.

ATTACK'S IDENTIFIERS

Attack Type	Bits Representation
Normal Behaviour	000000
Sinkhole	000001
Selective Forward	000100
Blackhole	001000
Selective Forward Forwarding Ratio	000101
Selective Forward Block Node	000110
Selective Forward Forwarding Ratio & Blackhole	001101
Selective Forward Block Node & Blackhole	001110
Unknown Abnormal Behaviour	111111

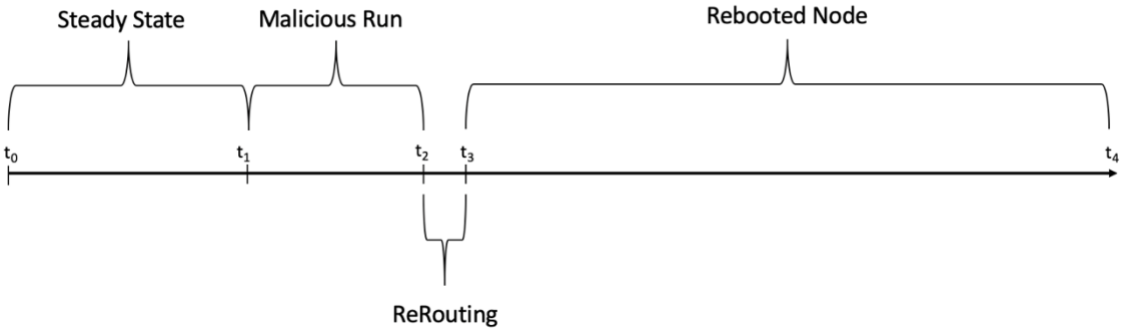
Πίνακας 4.1: Αναγνωριστικό Επίθεσης

Τα δύο τελευταία bit του πακέτου συνδέσμου ασφαλείας προσδιορίζουν τον τύπο του πληροφοριοδότη. Ο τύπος πληροφοριοδότη αποκαλύπτει την θέση του συστήματος ανίχνευσης (IDS) που έχει εντοπίσει την επίθεση. Όταν ο τύπος πληροφοριοδότη έχει τιμή *00,* υποδηλώνει ότι η ανίχνευση έγινε από τον ίδιο τον κόμβο, δηλ. από έναν τοπικό σύστημα ανίχνευσης, η ταυτότητα του οποίου βρίσκεται στα δύο πρώτα bytes του πακέτου. Η τιμή 01 δείχνει ότι το πρόβλημα εντοπίστηκε από έναν ειδικό κόμβο παρακολούθησης ασφάλειας, δηλαδή έναν αποκεντρωμένο παράγοντα ασφαλείας εντός του δικτύου. Μια τιμή 10 δείχνει ότι η ανίχνευση έγινε από έναν καθολικό κόμβο, π.χ. η κεντρική πύλη. Για την συγκεκριμένη

εργασία χρησιμοποιούνται μόνο τοπικά συστήματα ανίχνευσης, επομένως ο τύπος πληροφοριοδότη είναι πάντα *00*.

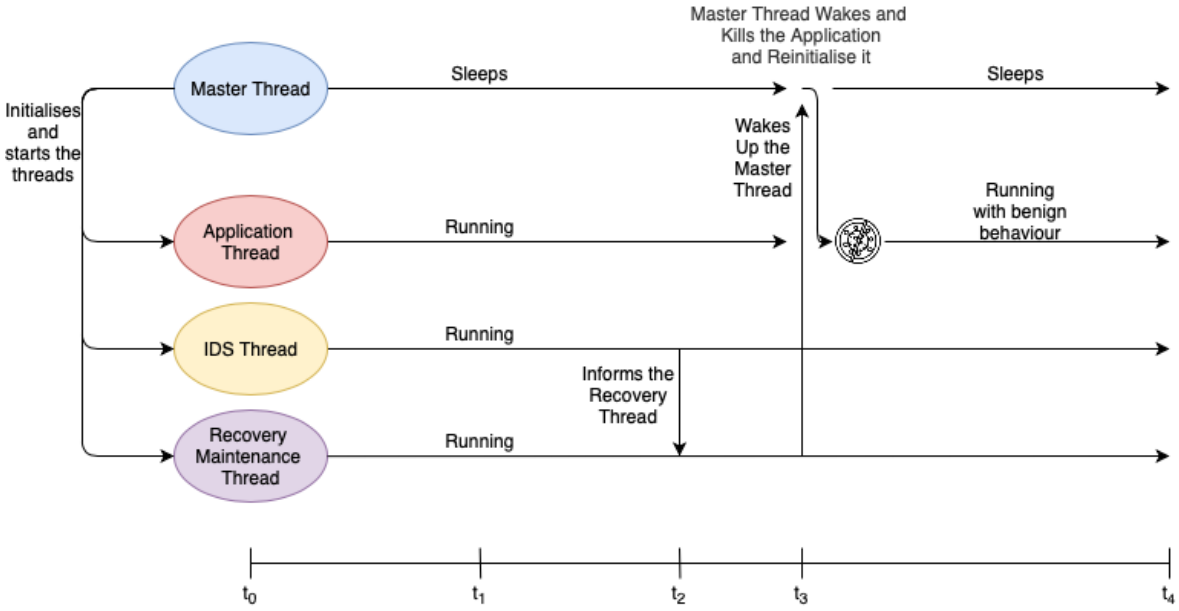
4.4 Επαναδρομολόγηση

Η τεχνική επαναδρομολόγησης αποσκοπεί στην ειδοποίηση των γειτόνων του κακόβουλου κόμβου και την εκτροπή των μονοπατιών που τον συμπεριλαμβάνουν μέσω μιας εναλλακτικής, πιο ασφαλούς διαδρομής. Όταν το τοπικό σύστημα ανίχνευσης ανιχνεύσει μια μη φυσιολογική συμπεριφορά, ενεργοποιείται ένας συναγερμός που με την σειρά του ενεργοποιεί την εκ νέου δρομολόγηση. Ένα πακέτο συνδέσμου ασφαλείας μεταδίδεται προειδοποιώντας τους γειτονικού κόμβους για την νέα κατάσταση ασφαλείας του κόμβου. Δηλαδή, ότι ο κόμβος πλέον δεν είναι αξιόπιστος, καθώς διακυβεύεται με μια επίθεση δρομολόγησης. Όταν η εκ νέου δρομολόγηση χρησιμοποιείται ως το μόνο μέτρο απόκρισης ασφαλείας, ο επηρεαζόμενος κόμβος συνεχίζει να μεταδίδει το πακέτο για συνολικό χρόνο δέκα δευτερολέπτων σε ρυθμό ενός πακέτου ανά δευτερόλεπτο. Μετά από αυτό, το πακέτο αποστέλλεται μία φορά ανά λεπτό για να ειδοποιεί νέους κόμβους που συνδέονται στο δίκτυο. Έχοντας λάβει ένα πακέτο συνδέσμου ασφαλείας, ο γειτονικός κόμβος θέτει το bit ασφαλείας του αντίστοιχου γείτονα στο *0*. Εάν το bit ασφαλείας ενός γείτονα έχει οριστεί στο *0*, τότε ο κόμβος δεν θεωρείται έγκυρος και δεν λαμβάνεται υπόψη όταν εκτελείτε ο αλγόριθμος δρομολόγησης.



Εικόνα 4.2: Χρονοδιάγραμμα ανάκτησης ασφάλειας

4.5 Επανεκκίνηση



Εικόνα 4.3: Τα Threads της Τεχνικής Επανάκτησης

Η δεύτερη τεχνική ανάκτησης είναι η επανεκκίνηση του κακόβουλου κόμβου μετά την τεχνική της επαναδρομολόγησης, για να αποφευχθεί η απώλεια πόρων από το δίκτυο. Στο τέλος της τεχνικής εκ νέου δρομολόγησης, ο κακόβουλος κόμβος επισημαίνεται ως μη αξιόπιστος και δεν αποτελεί πλέον μέρος του δικτύου. Η τεχνική επανεκκίνησης επαναφέρει τον κακόβουλο κόμβο στην αρχική του λειτουργία. Η εφαρμογή κόμβου χρησιμοποιεί μια κύρια διαδικασία που είναι υπεύθυνη για την εκκίνηση και την προετοιμασία τριών διαδικασιών: το τοπικό IDS, τη διαδικασία ανάκτησης ασφαλείας και την κύρια εφαρμογή (Εικόνα 4.3). Μόλις ξεκινήσουν οι διαδικασίες, η κύρια διαδικασία μεταβαίνει σε κατάσταση αναστολής λειτουργίας για να μειώσει την κατανάλωση ενέργειας. Μόλις το σύστημα ανίχνευσης εντοπίσει μια ανωμαλία, η διαδικασία ανάκτησης ασφαλείας ξεκινά τη διαδικασία εκ νέου δρομολόγησης (βλ. t2 στην Εικόνα 4.2) για δέκα δευτερόλεπτα. ‘Οταν όλοι οι κόμβοι ενημερωθούν με την χρήση του πακέτου συνδέσμου ασφαλείας, η κύρια διαδικασία ανάκτησης ασφαλείας σταματά την εκτέλεση της κακόβουλης εφαρμογής και ξεκινά το εφεδρικό αντίγραφο της εφαρμογής που είναι φορτωμένο στην κύρια μνήμη του κόμβου. Μόλις ο κακόβουλος κόμβος ανακτήσει την καλοήθη συμπεριφορά, χρησιμοποιεί το πακέτο συνδέσμου ασφαλείας για να ανακοινώσει ότι μπορεί να θεωρηθεί ξανά αξιόπιστος. Για να καθοριστεί γρήγορα η θέση του κόμβου στο δίκτυο, όταν οι γειτονικοί κόμβοι ενημερώνονται για μια αλλαγή στο bit ασφαλείας, μεταδίδουν αμέσως μια «ανακοίνωση» με τις προδιαγραφές δρομολόγησης, στην προκειμένη περίπτωση, τους κόμβους που απέχουν από την κεντρική πύλη και την τιμή της ισχύς του σήματος (RSSI).

4.6 Μετρήσεις Απόδοσης

Χρησιμοποιήθηκαν δύο μετρήσεις απόδοσης για την αξιολόγηση της αποτελεσματικότητας των λύσεων ανάκτησης από άποψη χρόνου και το ποσοστό παράδοσης πακέτων στη κεντρική πύλη. Τα πακέτα που ελήφθησαν από την κεντρική πύλη χρησιμοποιήθηκαν για να δείξουν την απώλεια πακέτων από την επίθεση και κατά την ενεργοποίηση των λύσεων ανάκτησης. Το ποσοστό των πακέτων που λαμβάνονται είναι η αναλογία του μέσου όρου των συνολικών πακέτων που λαμβάνονται από την κεντρική πύλη όταν δεν υπήρχε κακόβουλη επίθεση στο δίκτυο σε σχέση με των πακέτων που έλαβε η κεντρική πύλη όταν υπήρχε μια επίθεση.

Η επιβάρυνση χρόνου χρησιμοποιείται ως μέτρηση εκτίμησης κόστους για κάθε μηχανισμό ανάκτησης. Η επιβάρυνση του χρόνου μετριέται ξεκινώντας όταν ενεργοποιείται ο μηχανισμός ανάκτησης έως ότου το δίκτυο επανέλθει στην κανονική του κατάσταση. Για την τεχνική επαναδρομολόγησης, υποθέτουμε ότι το δίκτυο ανακτάται όταν ενημερωθούν όλοι οι γείτονες του κακόβουλου κόμβου για την ύπαρξή του (φαίνεται στην Εικόνα 4.2 ως t2 μέχρι το t3). Για την τεχνική επανεκκίνησης μετράμε το χρόνο που απαιτείται για να επιστρέψει ο κακόβουλος κόμβος στο δίκτυο και να είναι σε θέση να στείλει ή να λάβει το πρώτο του πακέτο (φαίνεται στην Εικόνα 4.2 ως t3 μέχρι το t4).

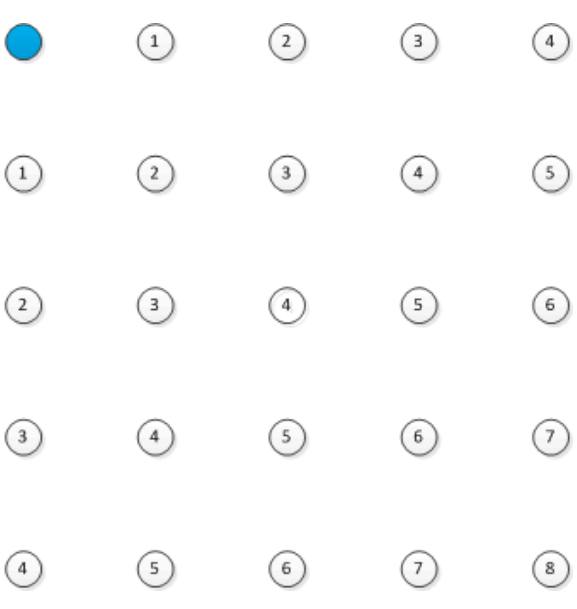
Κεφάλαιο 5

5 Πειραματικό Περιβάλλον

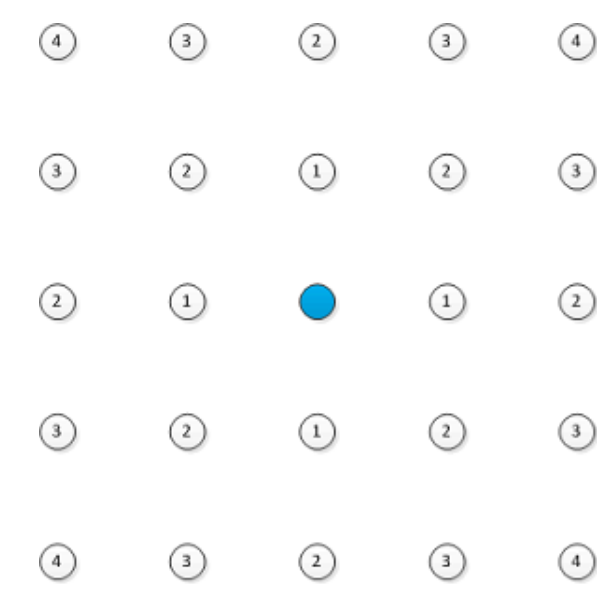
5.1 Επισκόπηση

Σε αυτό το κεφάλαιο αναλύονται οι πειραματικές παράμετροι και διαμόρφωσης του περιβάλλοντος καθώς και τα αποτελέσματα. Πιο συγκεκριμένα στο υποκεφάλαιο Τοπολογία αναλύονται οι τοπολογίες που χρησιμοποιήθηκαν για την αξιολόγηση. Στο υποκεφάλαιο Σενάρια αξιολόγησης και χρόνοι εκτέλεσης αναφέρονται οι χρόνοι εκτέλεσης του κάθε πειράματος καθώς και τα πειράματα που εκτελέστηκαν, ενώ στα υποκεφάλαια Αποτελέσματα από προσομοιώσεις COOJA παρουσιάζονται τα αποτελέσματα και στο Περιβάλλον FIT-IOT Lab αναλύεται η εμπειρία και οι γνώσεις που αποκομίστηκαν από το εν λόγο περιβάλλον.

5.2 Τοπολογία



Εικόνα 5.1: Κεντρική Πύλη στην Κορυφή



Εικόνα 5.2: Κεντρική Πύλη στη Μέση

Για τα πειράματα χρησιμοποιήθηκαν δύο τοπολογίες. Πρόκειται για δυο πλέγματα με 25 κόμβους με την μόνη διαφορά την τοποθεσία της κεντρική πύλης, όπως φαίνονται στις Εικόνα 5.1 και Εικόνα 5.2. Ο κάθε κόμβος μπορεί να αποστείλει και να λάβει πακέτα από τους κόμβους που είναι δεξιά, αριστερά, πάνω και κάτω από αυτόν. Δηλαδή ένας κόμβος στο δίκτυο είναι γείτονας με το πολύ τέσσερεις άλλους κόμβους. Στις εικόνες με μπλε χρώμα αναπαρίσταται η κεντρική πύλη του δικτύου. Οι αριθμοί αντιπροσωπεύουν το από πόσους ενδιάμεσους κόμβους χρειάζεται να αποσταλθεί το μήνυμα μέχρι να φτάσει στην κεντρική πύλη. Η πρώτη η τοπολογία που η κεντρική πύλη βρίσκεται στο άκρο του δικτύου μπορεί να λάβει μηνύματα μόνο από δύο κόμβους ενώ η δεύτερη τοπολογία έχει την κεντρική πύλη του δικτύου στο κέντρο, και έχει τέσσερεις κόμβους που μπορούν να τις προωθούν μηνύματα.

5.3 Σενάρια αξιολόγησης και χρόνοι εκτέλεσης

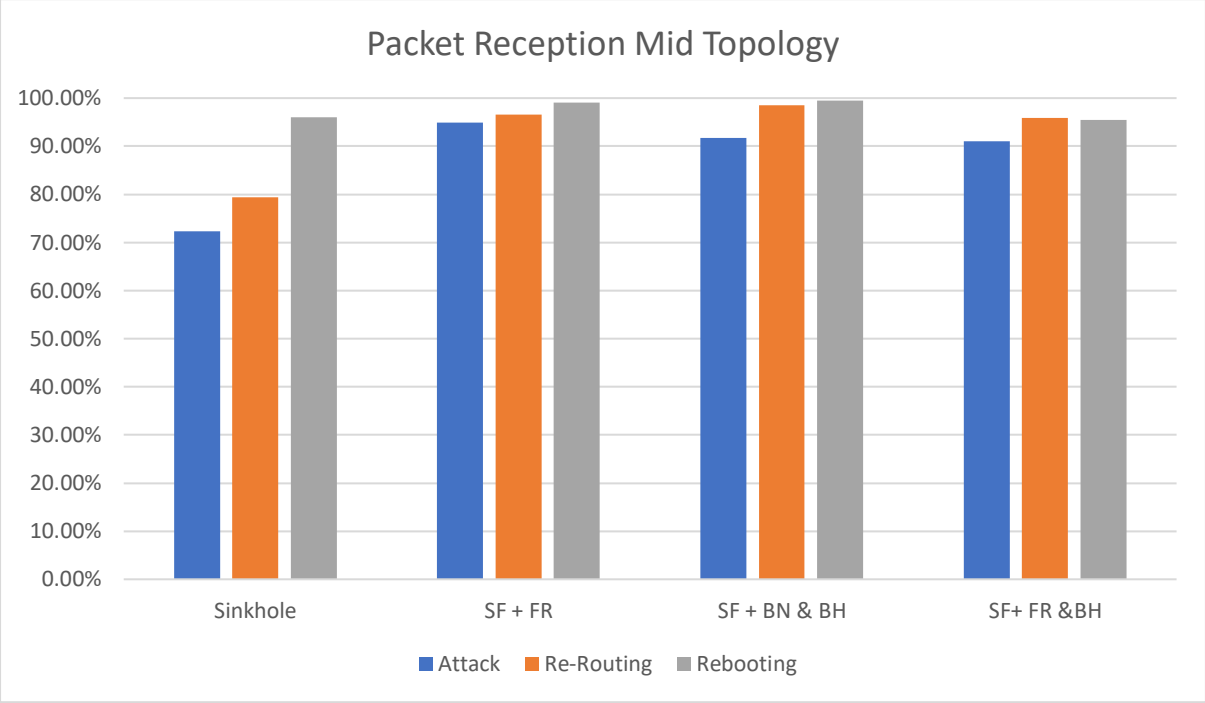
Για τις δύο τοπολογίες, χρησιμοποιήθηκαν τέσσερις τύποι πειραματικών σεναρίων: τα καλοήθη, τα κακόβουλα, της επαναδρομολόγησης και της επανεκκίνησης. Το καλοήθη σενάριο θεωρήθηκε το βασικό σενάριο στο οποίο όλοι οι κόμβοι εκτέλεσαν την καλοήθη εφαρμογή και χρησιμοποιήθηκαν για την καταγραφή των πακέτων που έλαβε η κεντρική πύλη. Συνολικά εκτελέστηκαν 20 καλοήθη σενάρια (10 για κάθε τοπολογία), για τη δημιουργία διαφορετικών μονοπατιών επικοινωνίας και χρησιμοποιήθηκαν για τον υπολογισμό των μέσων πακέτων που λαμβάνονται από την κεντρική πύλη. Στα κακόβουλα σενάρια υπάρχει ένας κόμβος εντός του δικτύου που θα παραβιαστεί από μια επίθεση επιπέδου δρομολόγησης. Για κάθε μία από τις 4 επιθέσεις δρομολόγησης, εκτελέστηκαν 24 κακόβουλα σενάρια, η κεντρική πύλη θεωρείται ότι δεν έχει παραβιαστεί. Τα σενάρια επαναδρομολόγησης είναι τα κακόβουλα σενάρια με ενεργοποιημένο τον μηχανισμό επαναδρομολόγησης. Τα σενάρια επανεκκίνησης χρησιμοποίησαν τεχνικές επαναδρομολόγησης και επανεκκίνησης. Ο μέσος όρος του αριθμού των πακέτων που ελήφθησαν από την κεντρική πύλη καταγράφηκαν για τα κακόβουλα σενάρια, επανεγκατάστασης και επανεκκίνησης και χρησιμοποιήθηκαν ως μετρική απόδοσης.

Ο χρόνος προσομοίωσης κάθε σεναρίου ορίστηκε σε 15 λεπτά, τα δύο πρώτα λεπτά χρησιμοποιούνται ως σταθερή κατάσταση στην οποία δημιουργείται το δίκτυο. Κάθε κόμβος επιλέγει έναν τυχαίο ακέραιο r_0 μεταξύ του εύρους 0-20. Η εφαρμογή αρχίζει να στέλνει πακέτα δεδομένων μετά τα πρώτα δύο λεπτά και μετά από r_0 δευτερόλεπτα. Αυτό γίνεται για να επιτευχθεί λιγότερη συμφόρηση στο φυσικό στρώμα. Επιτρέπουμε ένα χρονικό διάστημα ανίχνευσης 1 λεπτού για τον τοπικό IDS να ανιχνεύσει την παρουσία της επίθεσης και στο 3ο

λεπτό του χρόνου προσομοίωσης, οι λύσεις αποκατάστασης ξεκινούν την εκτέλεση (βλ. t2 στην Εικόνα 4.2).

5.4 Αποτελέσματα από προσομοιώσεις COOJA

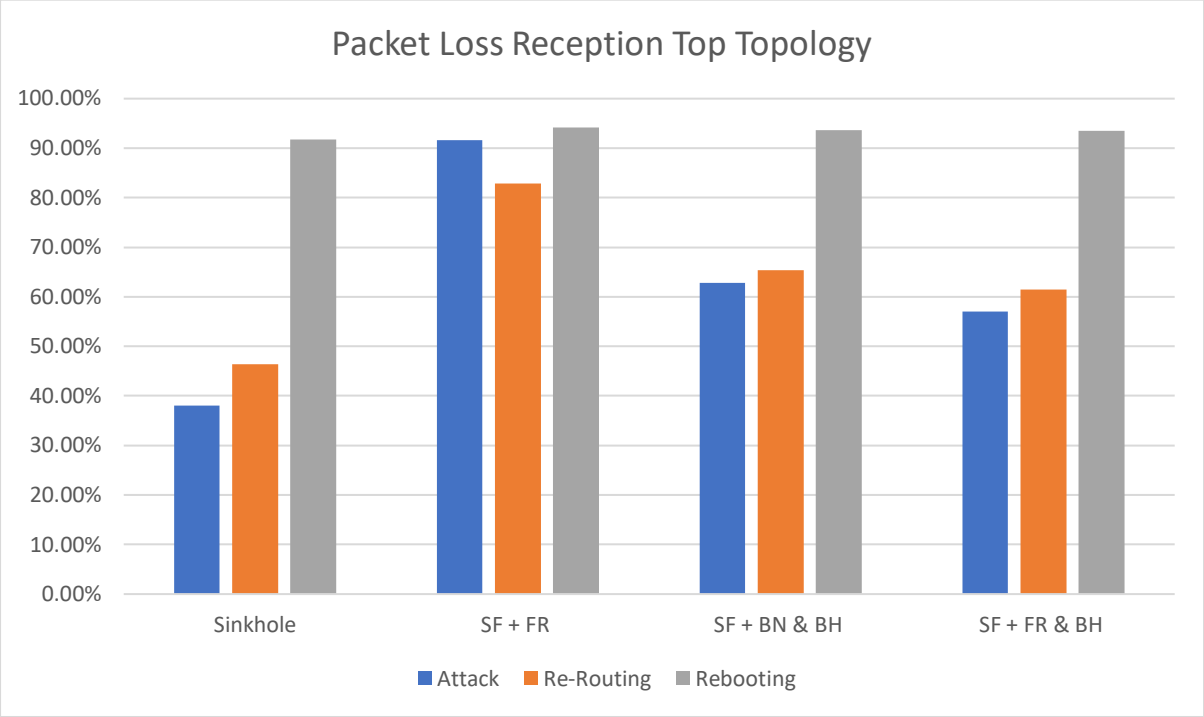
Το Cooja παρέχεται από το Contiki OS ως προσομοιωτής δικτύου και έχει γίνει ένα ευρέως χρησιμοποιούμενο εργαλείο στον τομέα των ασύρματων δικτύων αισθητήρων. Το COOJA παρέχει την δυνατότητα προσομοίωσης του φυσικού μέσου. Όλοι οι κόμβοι που χρησιμοποιήθηκαν για την προσομοίωση είναι Tmote-Sky με εύρος λήψης ακτίνας 25 μέτρων. Κάθε κόμβος μεταδίδει ένα πακέτο δεδομένων 24 byte κάθε 20 δευτερόλεπτα με την κεντρική πύλη ως τον τελικό παραλήπτη.



Εικόνα 5.3: Πακέτα που παραλήφθηκαν από την Κεντρική Πύλη με Τοπολογία όπου βρισκόταν στο κέντρο

Στην Εικόνα 5.3 παρουσιάζονται τα δεδομένα όπου η κεντρική πύλη βρισκόταν στο κέντρο της τοπολογίας (βλ. Εικόνα 5.2). Όπως παρατηρείτε σε όλες τις επιθέσεις και οι δύο μέθοδοι βοηθούν στην αναπλήρωση της απώλειας πακέτων που συμβαίνει από την εκάστοτε επίθεση. Στην επίθεση Sinkhole υπάρχει η μεγαλύτερη απώλεια δεδομένων, με την κεντρική πύλη να λαμβάνει μόλις 72,29% των αναμενόμενων πακέτων. Με την μέθοδο της επαναδρομολόγησης ανακτάται περίπου 8% του χαμένου όγκου δεδομένων ενώ όταν ο κακόβουλος κόμβος επαναφέρεται στην αρχική του καλοήθη του λειτουργία η κεντρική πύλη λαμβάνει 95,96% του αναμενόμενου όγκου δεδομένων. Στις υπόλοιπες επιθέσεις βλέπουμε ότι δεν επηρεάστηκε σε

τόσο ψηλό βαθμό η απώλεια πακέτων αλλά και πάλι στην επίθεση Selective Forward – Forwarding Ratio υπήρχε ανάκτηση δεδομένων της τάξης του 2% σε κάθε τεχνική ανάκτησης. Όπως φαίνεται στην γραφική στην επίθεση Selective Forward – Block Node & Blackhole η τεχνική της επαναφοράς κατάφερε να πετύχει το μεγαλύτερο όγκο παραληφθέντων δεδομένων με 99,48%.



Εικόνα 5.4: Πακέτα που παραλήφθηκαν από την Κεντρική Πύλη με Τοπολογία όπου βρισκόταν στο κέντρο

Στην Εικόνα 5.4 παρουσιάζονται τα πειράματα από την εκτέλεση όπου η κεντρική πύλη βρισκόταν στην κορυφή της τοπολογίας (βλ. Εικόνα 5.1). Σε όλες τις περιπτώσεις παρουσιάζεται μεγαλύτερη απώλεια δεδομένων σε σχέση με την τοπολογία όπου η κεντρική πύλη βρισκόταν στο κέντρο της τοπολογίας. Αυτό είναι αναμενόμενο καθώς η κεντρική πύλη έχει μόνο δυο γείτονες που μπορούν να της προωθήσουν πακέτα. Επομένως αν αυτοί οι γείτονες επηρεαστούν από κάποια επίθεση μειώνονται οι πόροι του δικτύου. Επίσης το ότι πολλοί κόμβοι είναι μακριά από την πηγή μπορεί να τους κατευθύνει στο να στέλνουν τα πακέτα τους προς κάποιο κακόβουλο κόμβο που είναι πιο κοντά τους. Στο άρθρο \cite{} αναφέρετε ότι η επίθεση Sinkhole σε αυτή την τοπολογία μπορεί να επηρεάσει την κεντρική πύλη στο να λαμβάνει μέχρι και τα μισά πακέτα. Αυτό παρουσιάζεται και από τα πιο πάνω αποτελέσματα όπου η πηγή λαμβάνει μόλις το 38,04% των αναμενόμενων πακέτων στην επίθεση Sinkhole. Σε αυτή την επίθεση, η τεχνική ανάκαμψης επαναφοράς, επιτυγχάνει βελτίωση 53.66% των παραληφθέντων πακέτων έναντι 8,39% της επαναδρομολόγησης. Στις επιθέσεις Selective Forward και Blackhole, τα πακέτα που λαμβάνονται από την κεντρική πύλη είναι ακόμη πιο κοντά στο καλοήθη σενάριο. Η επίθεση Selective Forward - Forwarding Ratio

επιτυγχάνει το 94,20% των πακέτων που ελήφθησαν κατά μέσο όρο στο καλοήθες σενάριο, ενώ στα σενάρια Selective Forward & Blackhole - Forwarding Ratio και Selective Forward & Blackhole – Block Node είναι 93,54% και 93,67% αντίστοιχα.

Χρόνος που ενημερώθηκε ο Τελευταίος Γείτονας (t_2 -t_3)		
Επίθεση	**Χρόνος σε Δευτερόλεπτα**	
	Ελάχιστος	Μέγιστος
Sinkhole	0,26	0,80
SF + FR	0,33	1,21
SF + BN & BH	0,25	1,03
SF + FR & BH	0,34	0,95
Μέσος Όρος	0,58	

Πίνακας 5.1: Κεντρική Πύλη στην Μέση

Χρόνος που ενημερώθηκε ο Τελευταίος Γείτονας (t_2 -t_3)		
Επίθεση	**Χρόνος σε Δευτερόλεπτα**	
	Ελάχιστος	Μέγιστος
Sinkhole	0,23	0,93
SF + FR	0,24	1,08
SF + BN & BH	0,68	1,42
SF + FR & BH	0,43	0,98
Μέσος Όρος	0,73	

Πίνακας 5.2: Κεντρική Πύλη στην Κορυφή

Στους Πίνακας 4.1 και Πίνακας 5.2 αναγράφονται οι χρόνοι που χρειάστηκε για να ειδοποιηθεί ο τελευταίος γείτονας του κακόβουλου κόμβου από το πακέτο ασφάλειας συνδέσμου στα σενάρια που χρησιμοποιήθηκε μόνο η τεχνική της επαναδρομολόγησης. Στην στήλη με το ελάχιστο και μέγιστο χρόνο αναφέρετε στο ελάχιστο και μέγιστο χρόνο που χρειάστηκε σε κάποιο πείραμα αναλόγως της επίθεσης να ειδοποιηθεί ο τελευταίος γείτονας του κακόβουλου κόμβου αντίστοιχα. Παρατηρείται ότι ο μέγιστος χρόνος για την τοπολογία με την κεντρική πύλη στην μέση σημειώνεται για την επίθεση Selective Forward – Forwarding Ratio με 1,21 δευτερόλεπτα, ενώ η αντίστοιχη μέγιστη τιμή για την τοπολογία με την κεντρική πύλη στην κορυφή είναι 1,42 δευτερόλεπτα για την επίθεση Selective Forward & Blackhole – Block Node. Αντίστοιχα οι μέσοι όροι για της τοπολογίες είναι 0,58 και 0,73 δευτερόλεπτα.

Χρόνος που χρειάστηκε για επαναφορά (t_3 -t_4)		
Επίθεση	**Χρόνος σε Δευτερόλεπτα**	
	Ελάχιστος	Μέγιστος
Sinkhole	0,24	7,87
SF + FR	0,01	0,02
SF + BN & BH	0,37	6,87
SF + FR & BH	0,24	6,50
Μέσος Όρος	1,32	

Πίνακας 5.3: Κεντρική Πύλη στην Μέση

Χρόνος που χρειάστηκε για επαναφορά (t_3 -t_4)		
Επίθεση	**Χρόνος σε Δευτερόλεπτα**	
	Ελάχιστος	Μέγιστος
Sinkhole	0,37	10,99
SF + FR	0,01	0,02
SF + BN & BH	0,10	3,74
SF + FR & BH	0,24	6,24
Μέσος Όρος	1,51	

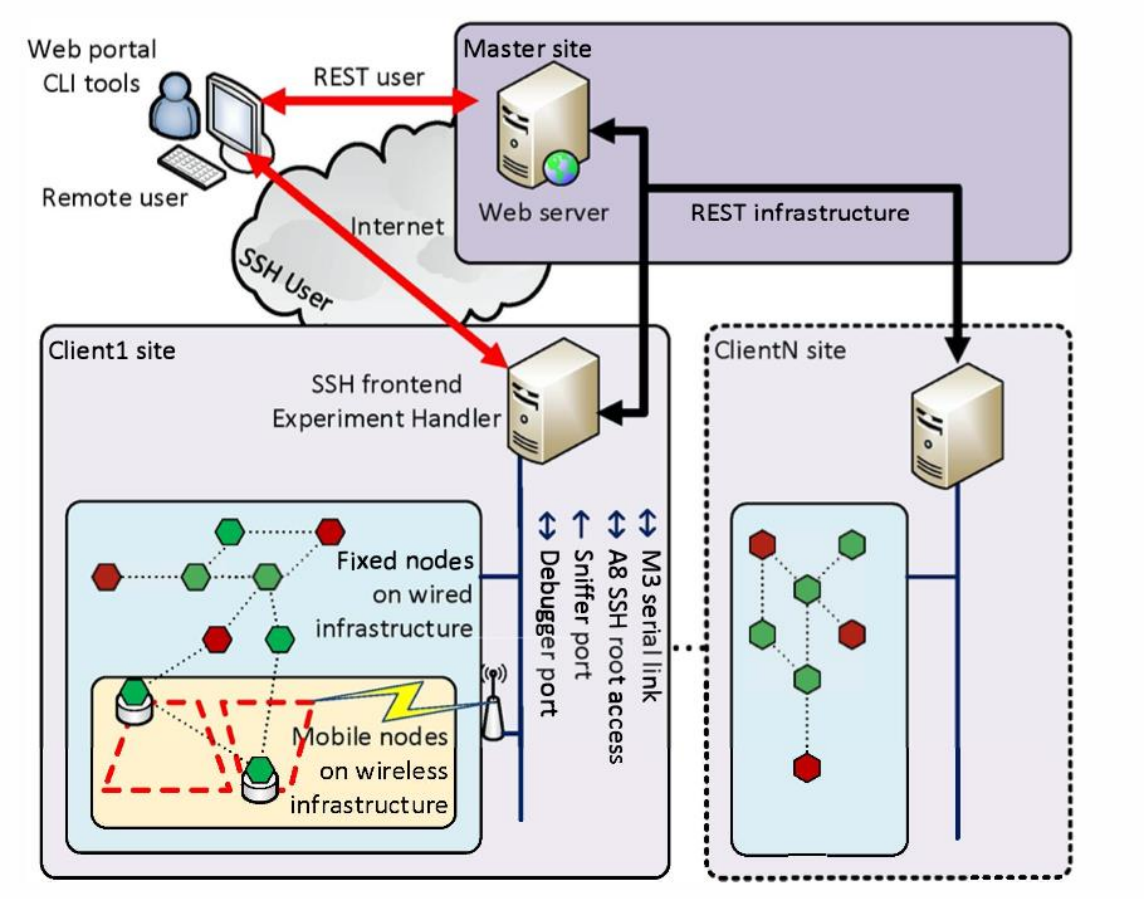
Πίνακας 5.4: Κεντρική Πύλη στην Κορυφή

Στους Πίνακας 5.3 και Πίνακας 5.4 αναγράφονται οι χρόνοι που χρειάστηκε για να μπορέσει ο κακόβουλος κόμβος να επανέλθει στο δίκτυο, δηλαδή να αποστέλλει και να λαμβάνει δεδομένα γνωρίζοντας τις παραμέτρους δρομολόγησης των γειτόνων του. Οι χρόνοι που αναφέρονται υποδεικνύουν το χρόνο μετά τα πρώτα δέκα δευτερόλεπτα όπου η επαναδρομολόγηση λαμβάνει χώρα. Στην στήλη με το ελάχιστο και μέγιστο χρόνο αναφέρετε στο ελάχιστο και μέγιστο χρόνο που χρειάστηκε σε κάποιο πείραμα αναλόγως της επίθεσης να αποστείλει ή να λάβει πακέτο ο κακόβουλος κόμβος αφού επαναφέρθηκε στην καλοήθη συμπεριφορά του. Παρατηρείται ότι ο μέγιστος χρόνος για την τοπολογία με την κεντρική πύλη στην μέση σημειώνεται για την επίθεση Sinkhole με 7,87 δευτερόλεπτα, ενώ η αντίστοιχη μέγιστη τιμή για την τοπολογία με την κεντρική πύλη στην κορυφή είναι 10,99 δευτερόλεπτα για την ίδια επίθεση. Αντίστοιχα οι μικρότεροι χρόνοι είναι 0,01 και για τα δύο σενάρια στην επίθεση Selective Forward- Forwarding Ratio. Τέλος οι μέσοι όροι για της τοπολογίες είναι 1,32 και 1,51 δευτερόλεπτα.

5.5 Περιβάλλον FIT-IOT Lab

Το FIT IoT-LAB testbed [2], είναι ένα ανοιχτό testbed που αποτελείται από 2728 ασύρματους κόμβους χαμηλής ισχύος και 117 φορητά ρομπότ διαθέσιμα για πειραματισμό με ασύρματες τεχνολογίες IoT μεγάλης κλίμακας, που κυμαίνονται από πρωτόκολλα χαμηλού επιπέδου έως προηγμένες υπηρεσίες Internet. Η πλατφόρμα είναι ένα ακριβές επιστημονικό εργαλείο ανοιχτής πρόσβασης και ανοιχτού κώδικα πολλαπλών χρηστών. Το δοκιμαστικό IoT-LAB αναπτύσσεται σε 6 τοποθεσίες σε όλη τη Γαλλία και προσφέρεται για εκτελέσεις πειραμάτων σε πραγματικές συσκευές IoT.

Οι κόμβοι είναι είτε στατικοί είτε κινητοί, και μπορούν να κατανεμηθούν σε διάφορες τοπολογίες σε όλους τους ιστότοπους. Τα ασύρματα δίκτυα είναι συχνά κινητά και η κινητικότητα μπορεί να επηρεάσει σημαντικά την απόδοση των πρωτοκόλλων δικτύου. Η ικανότητα μετακίνησης της θέσης των ρομπότ κατά τη διάρκεια ενός πειράματος επιτρέπει στον χρήστη να ποσοτικοποιήσει την επίδραση της κινητικότητας των κόμβων. Όπως και τα στατικά αντίστοιχα, οι κόμβοι μπορούν να δεσμευτούν, να επαναπρογραμματιστούν και να παρακολουθούνται. Ο χρήστης μπορεί επίσης να ελέγξει την κίνηση των ρομπότ.



Εικόνα 5.5: Υποδομή FIT-IoT Lab [2]

Διατίθεται ποικιλία ασύρματων αισθητήρων, με διαφορετικές αρχιτεκτονικές επεξεργαστών και ασύρματα τσιπ. Οι κόμβοι είναι πλήρως προγραμματίσιμοι. Η διεπαφή διαχείρισης επιτρέπει στον χρήστη να φορτώνει αυθαίρετα εκτελέσιμα αρχεία (binary files) στις συσκευές και να έχει άμεση πρόσβαση στις πύλες στις οποίες συνδέονται οι κόμβοι. Χρησιμοποιώντας εργαλεία βάσει διαδικτύου ή command-line, ένας χρήστης μπορεί να δεσμεύσει έναν αυθαίρετο αριθμό κόμβων σε έναν ή περισσότερους ιστότοπους για την εκτέλεση πειραμάτων. Μια πιο αναλυτική επεξήγηση της ροής και της υποδομής του συστήματος αναπαριστάτε στην Εικόνα 5.5. Μόλις δεσμευτεί, ο χρήστης αναπτύσσει το δικό του firmware στους κόμβους. Πρόσθετα εργαλεία επιτρέπουν στον χρήστη να διαμορφώσει ορισμένους κόμβους ώστε να λειτουργούν ως κόμβοι πύλης συνδεδεμένοι στο Διαδίκτυο, να παρακολουθεί την κατανάλωση ενέργειας κάθε κόμβου και να μετρά άλλες μετρήσεις, όπως καθυστέρηση από άκρο σε άκρο ή απόδοση.

Στο περιβάλλον του FIT-IOT Lab πολλοί χρήστες μπορούν να εκτελέσουν τον κώδικά τους. Επομένως, πολλοί χρήστες μπορεί να συναγωνίζονται για τους πόρους, και να θέλουν να εκτελέσουν στους ίδιους κόμβους. Όταν ένας χρήστης, καταθέσει μια εκτέλεση στο σύστημα, το σύστημα αξιολογεί εάν όλοι οι κόμβοι είναι διαθέσιμοι, και εάν είναι εκτελείται κάποιο άλλο πείραμα σε κάποιο από αυτούς. Στην περίπτωση που κάποιος χρήστης εκτελεί πείραμα σε αυτούς τους κόμβους, πρέπει να περιμένει το σύστημα να προγραμματίσει την εκτέλεση του πειράματος. Το σύστημα χρησιμοποιεί ένα άπληστο αλγόριθμο για τον προγραμματισμό των πειραμάτων, ελέγχει από όλα τα πειράματα των χρηστών ποιο από αυτά θα τελειώσει ποιο γρήγορα και το προγραμματίζει πρώτο. Επομένως, πολλές φορές μπορεί να χρειαστεί μέρες μέχρι να ολοκληρωθούν οι εκτελέσεις που τοποθέτησες στο σύστημα.

Για την δημιουργία της επιθυμητής τοπολογίας και την λειτουργία του Contiki OS επιλέξαμε τους σταθερούς κόμβους M3 οι οποίοι είναι συμβατοί με το Contiki OS και τον ιστότοπο Lille καθώς η φυσική τοποθέτηση των κόμβων είναι τέτοια που κάνει εφικτή την δημιουργία πλέγματος σε αντίθεση με άλλου ιστότοπους όπου οι κόμβοι είναι τοποθετημένοι σε σειρά. Αρχικά προσπάθησα να απομονώσω πέντε κόμβους σε μια σειρά έτσι ώστε να ρυθμίσω την ενέργεια αποστολής και το όριο αποδοχής πακέτων ώστε οι κόμβοι να μπορούν να ακούσουν μόνο τους άμεσα διπλανούς τους κόμβους. Μετά από πολλά πειράματα, αναδείχθηκε ότι ανεξάρτητος της φυσική απόστασης το σήμα δεν ταξίδευε προς όλες τις κατεύθυνσης με την ίδια δύναμη λόγω του φυσικού περιβάλλοντος και της παρεμπόδισης του σήματος. Επομένως για την χρησιμοποιήθηκε ένα εργαλείο που σχεδιάστηκε για την κατασκευή τοπολογιών σε τέτοιας μορφής δίκτυα [16]. Το εργαλείο αυτό εκτελεί κάποιες προσομοιώσεις στους κόμβους, και στον ιστότοπο που επιλέγει ο χρήστης και με το εργαλείο μπορεί να σου καθορίσει ποιους κόμβους και με ποιες ρυθμίσεις να χρησιμοποιήσεις για την δημιουργία συγκεκριμένου

δέντρου. Η χρήση του εργαλείου κατέδειξε ότι δεν μπορούσε να δημιουργηθεί το πλέγμα που παρουσιάζεται στις Εικόνα 5.1 και Εικόνα 5.2, απλά ρυθμίζοντας τις παραμέτρους αποστολής και παραλαβής στο φυσικό επίπεδο.

Για να καταστεί δυνατή η δημιουργία του πλέγματος (βλ. Εικόνα 5.1 και Εικόνα 5.2) οι παράμετροι διαρρύθμισης της κεραίας του κάθε κόμβου έπρεπε να τροποποιηθούν. Συγκεκριμένα λόγο των μικρών αποστάσεων που βρίσκονται εγκατεστημένη οι κόμβοι, για να μειωθεί η συμφόρηση στο φυσικό επίπεδο έπρεπε να μειωθεί η ενέργεια αποστολής των πακέτων σε -6dBm και το όριο που γίνεται αποδεκτό ένα πακέτο σε -69dBm. Επίσης λόγω της πυκνότητας των κόμβων στην πλατφόρμα αλλά και της φυσικής τοποθέτησης του δεν ήταν εφικτό μόνο με αυτή την διαρρύθμιση της ενέργειας αποστολής να επιτευχθούν οι συγκεκριμένες τοπολογίες. Για κάθε κόμβο, δημιουργήθηκε ένας στατικός πίνακας με τα μοναδικά αναγνωριστικά των επιθυμητών γειτόνων που θα δημιουργούσαν την τοπολογία αξιολόγησης (βλ. Εικόνα 5.1 και Εικόνα 5.2). Επομένως, για να δημιουργηθεί το πλέγμα, κάθε κόμβος κατά την παραλαβή των «ανακοινώσεων» (τα πακέτα που χρησιμοποιεί ο WSP για να δημιουργήσει την τοπολογία) έλεγχε ένα στατικό πίνακα με το μοναδικό αναγνωριστικό του αποστολέα. Εάν αυτό βρισκόταν στον στατικό πίνακα του, σήμαινε ότι με βάση την πειραματική τοπολογία είναι γείτονας του και συνέχισε την διαδικασία με την ενημέρωση του πίνακα γειτνίασης. Εάν ο αποστολέας δεν βρισκόταν στον στατικό πίνακα, το πακέτο απορριπτόταν και αγνοούταν από όλες τις διαδικασίες (δρομολόγησης, υπολογισμού καλύτερου γείτονα, αποστολής πακέτων ασφαλείας συνδέσμου, αποστολής δεδομένων).

Μετά των προγραμματισμό όλων των πειραμάτων και την κανονική εκτέλεση του σε πραγματικό χρόνο και ολοκλήρωση τους, όταν το σύστημα τα χρονολόγησε για εκτέλεση, ακολούθησε η διαδικασία της αξιολόγηση τους. Κατά την αξιολόγηση των πειραμάτων, εντοπίστηκε προγραμματιστικό λάθος σπατάλης μνήμης κάτι που δεν διαφάνηκε στον προσομοιωτή COOJA, καθώς και ότι σε κάποιες εκτελέσεις δεν μπορούσαν οι καλοήθεις κόμβοι να ξεκινήσουν την κανονική λειτουργία τους, τα οποία κατέστησε τα πειράματα που εκτελέστηκαν μη αξιολογήσιμα. Μετά την επίλυση των προβλημάτων, υπήρξε ακόμη ένα τεχνικό πρόβλημα στην πλατφόρμα FIT-IOT. Ένας από τους κόμβους που χρησιμοποιούταν, μετά τον καθορισμό της τοπολογίας, για την αξιολόγηση των μεθόδων χρειαζόταν φυσική αντικατάσταση από τους διαχειριστές του συστήματος. Λόγω του ότι η δημιουργία νέας τοπολογίας και η εκτέλεση των πειραμάτων είναι χρονοβόρα, και η αντικατάσταση του κόμβου δεν έγινε άμεσα από τους διαχειριστές δεν κατέστη δυνατή η αξιολόγηση της μεθοδολογίας στην πλατφόρμα FIT-IOT.

Κεφάλαιο 6

6 Συμπεράσματα

6.1 Επισκόπηση

Σε αυτό το κεφάλαιο αναλύονται τα συμπεράσματα που πηγάζουν από αυτή την ατομική διπλωματική εργασία. Συγκεκριμένα στο υποκεφάλαιο Μελλοντική Δουλεία αναγράφονται κάποιες προτάσεις για περεταίρω αξιολόγηση της ανάκτησης ασφαλείας αλλά και δουλεία που θα συνεισφέρει στην καλύτερη αντιμετώπιση επιθέσεων στο επίπεδο δρομολόγησης. Τέλος στο υποκεφάλαιο Τελικά Συμπεράσματα αναγράφονται τα συμπεράσματα που εξάγονται από αυτή την διπλωματική εργασία.

6.2 Μελλοντική Δουλεία

Στο μέλλον μπορεί να γίνει η χρήση της πλατφόρμας FIT-IOT για την αξιολόγηση των μεθόδων ανάκτησης. Υπάρχει ήδη η υποδομή για την εκτέλεση πειραμάτων στην πλατφόρμα καθώς και η διαρρύθμιση των κόμβων για την χρήση της ίδια τοπολογίας με τον προσομοιωτή COOJA για την σύγκριση των αποτελεσμάτων.

Κατά την υλοποίηση της επανεκκίνησης χρησιμοποιήθηκαν τα protothreads που παρέχονται από το λειτουργικό σύστημα Contiki OS. Συγκεκριμένα, για την αξιολόγηση της μεθόδου χρησιμοποιήθηκε κώδικας με την καλοήθη συμπεριφορά που ήταν φορτωμένος στην μνήμη του κόμβου αλλά αδρανοποιημένος. Δηλαδή, το protothread που εκτελούσε την κακόβουλη δραστηριότητα διακοπτόταν, και ξεκινούσε ένα καινούργιο protothread με την καλοήθη συμπεριφορά για την αξιολόγηση της μεθόδου. Καλύτερη πρακτική για αυτή την συμπεριφορά είναι να υλοποιηθεί χρησιμοποιώντας την βιβλιοθήκη elf_loader [12] του Contiki OS. Αυτή η βιβλιοθήκη επιτρέπει στην έξυπνη συσκευή να φορτώσει κώδικα από την δευτερεύουσα μνήμη

του κόμβου, που δεν είναι ήδη στην κύρια μνήμη. Έτσι κατά την εκτέλεση εξοικονομείται μνήμη του κύριου κόμβου, κάτι που σε τέτοιο περιβάλλον θα είναι πάρα πολύ σημαντικό. Επίσης, ο εισβολέας κατά την προσπάθεια του να παραβιάσει τον κόμβο θα βρίσκει ακόμη ένα επίπεδο ασφαλείας, αφού το πρόγραμμα που θα βρίσκεται στην δευτερεύουσα μνήμη μπορεί να είναι κρυπτογραφημένο, κάνοντας το επίπεδο της ανάκτησης πιο ανθεκτικό σε κάποιες επιθέσεις που εκμεταλλεύονται ευπάθειες της κύριας μνήμης που δεσμεύονται από τον κόμβο κατά την εκτέλεση. Συνοψίζοντας, υλοποίηση της τεχνικής επανεκκίνησης με την χρήση δυναμικού επαναπρογραμματισμού του κόμβου με την βοήθεια της βιβλιοθήκης elf_loader [12], θα κάνει το μηχανισμό ανάκτησης πιο ασφαλές, αλλά θα εξοικονομείται και μνήμη του κύριου κόμβου από το να υπάρχει καθ’ όλη την εκτέλεση δεσμευμένη μνήμη για το αντίγραφο ασφαλείας.

Ακόμη εξίσου σημαντική δουλεία που θα ήταν χρήσιμο να υλοποιηθεί στο μέλλον είναι η αξιολόγηση των τεχνικών με διαφορετικές τοπολογίες και με περισσότερους από ένα κακόβουλους κόμβους στο δίκτυο. Δεν εξετάστηκε σε αυτή την διπλωματική εργασία εάν η τοπολογία έχει κάποιο αντίκτυπο στα αποτελέσματα. Επομένως, θα ήταν χρήσιμη η αξιολόγηση από διαφορετικές τοπολογίες με κόμβους πιο απομακρυσμένους από την κεντρική πύλη αλλά και με κόμβους που είναι κοντά στην κεντρική πύλη και έχουν περισσότερες επιλογές δρομολόγησης. Διαφορετικά μονοπάτια δρομολόγησης ενδέχεται να επηρεάσουν είτε της επιθέσεις, είτε τις μεθόδους ανάκτησης ασφαλείας, επομένως να υπάρχουν είτε καλύτερα είτε χειρότερα αποτελέσματα. Ακόμη ένα στοιχείο που θα πρέπει να αξιολογηθεί είναι το πόσους γείτονες έχει η κεντρική πύλη. Για παράδειγμα, σε τοπολογία όπου η κεντρική πύλη έχει ένα μόνο γείτονα, πόσο επηρεάζουν επιθέσεις και πόσο καταφέρνουν οι μέθοδοι ανάκτησης να ανταποκριθούν στην επαναφορά του δικτύου και με ποιο κόστος;

Η διαδικασία της ανάκτησης ασφαλείας σχεδιάστηκε και υλοποιήθηκε για να εφαρμοστεί και σε άλλες μορφές ανίχνευσης επιθέσεων. Δηλαδή, σε ένα δίκτυο που υπάρχουν αποκεντρωμένοι παράγοντες ασφαλείας, κόμβοι που παρακολουθούν την ροή των πακέτων και δεδομένων αλλά δεν συμβάλλουν στην αποστολή πακέτων στο δίκτυο. Επομένως, αυτοί οι κόμβοι θα είναι υπεύθυνη για την ενεργοποίηση της τεχνικής ανάκτησης ασφαλείας, υπάρχει και σχετική πρόνοια στο πακέτο ασφαλείας με το τύπο πληροφοριοδότη και δεσμευμένα τα bit 10 για αυτό το σκοπό.

Τέλος, καλό είναι στο μέλλον οι τεχνικές αυτές να αξιολογηθούν και με άλλα πρωτόκολλα δρομολόγησης. Ένα ευρέως διαδομένο πρωτόκολλο δρομολόγησης που χρησιμοποιείται στα δίκτυα IoT είναι το RPL (Routing Protocol for Low-Power and Lossy Networks) [28]. Η διαδικασία διαμόρφωσης του δέντρου δρομολόγησης έχει αρκετές διαφορές με το πρωτόκολλο

WSP που χρησιμοποιήθηκε για την αξιολόγηση της προτεινόμενης λύσης ανάκτησης ασφαλείας. Αυτό μπορεί να επηρεάσει είτε θετικά είτε αρνητικά τα ποσοστά παραληφθέντων πακέτων αλλά και τους χρόνους ανταπόκρισης. Ακόμη θα μπορεί να συγκριθεί καλύτερα με άλλες προτεινόμενες λύσεις ασφαλείας που προτείνονται και έχουν ως υπόβαθρο αυτό το πρωτόκολλο δρομολόγησης.

6.3 Τελικά Συμπεράσματα

Σε αυτή την διπλωματική εργασία προτείνονται, εφαρμόζονται και αξιολογούνται δύο μηχανισμοί ανάκτησης, αυτός της επαναδρομολόγησης και αυτός της επανεκκίνησης. Η λύση της εκ νέου δρομολόγησης διαφημίζεται η παρουσία του κακόβουλου κόμβου χρησιμοποιώντας το πακέτο συνδέσμου ασφαλείας που προσδιορίζει το μοναδικό αναγνωριστικό κόμβου, τον τύπο της επίθεσης και τον τύπο του πληροφοριοδότη που έχει εντόπιση την επίθεση. Στην πρώτη εφαρμογή επιτεύχθηκε να αναδειχθεί το εύρος των καθυστερήσεων που αναμένονται για κάθε μηχανισμό ανάκτησης αλλά και το μέγεθος της συνεισφοράς τους στην αντιμετώπιση των επιθέσεων.

7 Βιβλιογραφία

[1] A.-u. Rehman, S. U. Rehman και H. Raheem, «Sinkhole Attacks in Wireless Sensor Networks: A Survey,» *Wireless Personal Communications*, τόμ. 106, p. 2291–2313, 01 6 2019.

[2] Adjih, Cedric and Baccelli, Emmanuel and Fleury, Eric and Harter, Gaetan and Mitton, Nathalie and Noel, Thomas and Pissard-Gibollet, Roger and Saint-Marcel, Frederic and Schreiner, Guillaume and Vandaele, Julien and others, «FIT IoT-LAB: A Large Scale Open Experimental IoT Testbed,» σε *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*, 2015.

[3] B. Rossi, «Gartner's Internet of Things predictions,» *Information Age, Vitesse Media, January*, 2015.

[4] C. Ioannou και V. Vassiliou, «An Intrusion Detection System for Constrained WSN and IoT Nodes Based on Binary Logistic Regression,» σε *Proceedings of the 21st ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Montreal, 2018.

[5] C. Ioannou και V. Vassiliou, «Classifying Security Attacks in IoT Networks Using Supervised Learning,» σε *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, 2019.

[6] C. Ioannou και V. Vassiliou, «Security Agent Location in the Internet of Things,» *IEEE Access*, τόμ. 7, pp. 95844-95856, 2019.

[7] C. Ioannou και V. Vassiliou, «The Impact of Network Layer Attacks in Wireless Sensor Networks,» σε *International Workshop on Secure Internet of Things (SIoT 2016)*, Crete, 2016.

[8] C. Ioannou, V. Vassiliou και C. Sergiou, «An Intrusion Detection System for Wireless Sensor Networks,» σε *2017 24rd International Conference on Telecommunications (ICT)*, 2017.

[9] C. Ioannou, V. Vassiliou και C. Sergiou, «RMT: A Wireless Sensor Network Monitoring Tool,» σε *Proceedings of the 13th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, &\ {U}{b}{i}quitous Networks*, Malta, 2016.

[10] C. Karlof και D. Wagner, «Secure routing in wireless sensor networks: attacks and countermeasures,» *Ad Hoc Networks*, τόμ. 1, pp. 293-315, 2003.

[11] Da Silva, A. P. R. and Martins, M. H. T. and Rocha, B. P. S. and Loureiro, A. A. F. and Ruiz, L. B. and Wong, H. C., «Decentralized Intrusion Detection in Wireless Sensor Networks,» σε *Proceedings of the 1st ACM International Workshop on Quality of Service &Amp; Security in Wireless and Mobile Networks*, New York, NY, USA, 2005.

[12] Dunkels, Adam and Finne, Niclas and Eriksson, Joakim and Voigt, Thiemo, «Run-Time Dynamic Linking for Reprogramming Wireless Sensor Networks,» σε *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, New York, NY, USA, 2006.

[13] Dunkels, Adam and Gronvall, Bjorn and Voigt, Thiemo, «Contiki - A Lightweight and Flexible Operating System for Tiny Networked Sensors,» σε *Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks*, USA, 2004.

[14] Dunkels, Adam, «The ContikiMAC Radio Duty Cycling Protocol,» 2011.

[15] E. Stavrou και A. Pitsillides, «Recovering from the selective forwarding attack in WSNs - enhancing the recovery benefits of blacklisting and rerouting using directional antennas,» σε *2014 International Wireless Communications and Mobile Computing Conference (IWCMC)*, 2014.

[16] F. Kauer και V. Turau, «Constructing Customized Multi-Hop Topologies in Dense Wireless Network Testbeds,» *International Conference on Ad-Hoc Networks and Wireless*, pp. 319-331, 5 September 2018.

[17] F. Österlind, J. Eriksson και A. Dunkels, «Cooja TimeLine: A Power Visualizer for Sensor Network Simulation,» σε *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, New York, NY, USA, 2010.

[18] F. Song και B. Zhao, «Trust-Based LEACH Protocol for Wireless Sensor Networks,» σε *2008 Second International Conference on Future Generation Communication and Networking*, 2008.

[19] G. Ma, X. Li, Q. Pei και Z. Li, «A Security Routing Protocol for Internet of Things Based on RPL,» σε *2017 International Conference on Networking and Network Applications (NaNA)*, 2017.

[20] I. Hegazy, R. Safavi-Naini και C. Williamson, «Towards Securing Mintroute in Wireless Sensor Networks,» σε *2010 IEEE International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM)*, 2010.

[21] I. Krontiris, T. Dimitriou, T. Giannetsos και M. Mpasoukos, «Intrusion Detection of Sinkhole Attacks in Wireless Sensor Networks,» σε *Algorithmic Aspects of Wireless Sensor Networks*, τόμ. 4837, M. Kutyłowski, J. Cichoń και P. Kubiak, Επιμ., Springer Berlin Heidelberg, 2008.

[22] J. Moy και others, «OSPF Version 2,» 1998.

[23] Nadeem, Adnan and Howarth, Michael P, «An Intrusion Detection & Adaptive Response Mechanism for MANETs,» *Ad Hoc Networks*, τόμ. 13, p. 368–380, 2014.

[24] S. Kaplantzis, A. Shilton, N. Mani και Y. A. Sekercioglu, «Detecting Selective Forwarding Attacks in Wireless Sensor Networks using Support Vector Machines,» σε *Intelligent Sensors, Sensor Networks and Information, 2007. ISSNIP 2007. 3rd International Conference on*, 2007.

[25] S. Raza, L. Wallgren και T. Voigt, «SVELTE: Real-time Intrusion Detection in the Internet of Things,» *Ad hoc networks*, τόμ. 11, p. 2661–2674, 2013.

[26] S. Tanachaiwiwat, P. Dave, R. Bhindwale και A. Helmy, «Location-centric isolation of misbehavior and trust routing in energy-constrained sensor networks,» σε *IEEE International Conference on Performance, Computing, and Communications, 2004*, 2004.

[27] S. Y. Moon και T. H. Cho, «Intrusion Detection Scheme against Sinkhole Attacks in Directed Diffusion Based Sensor Networks,» *IJCSNS International Journal of Computer Science and Network Security*, τόμ. 9, p. 118–122, 2009.

[28] T. Winter, P. Thubert, A. Brandt, J. W. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J.-P. Vasseur και R. K. Alexander, «RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks.,» *rfc*, τόμ. 6550, p. 1–157, 2012.

[29] Wang, Shiau-Huey and Tseng, Chinyang Henry and Levitt, Karl and Bishop, Matthew, «Cost-Sensitive Intrusion Responses For Mobile Ad Hoc Networks,» σε *International Workshop on Recent Advances in Intrusion Detection*, 2007.

[30] Z. Cao, J. Hu, Z. Chen, M. Xu και X. Zhou, «Feedback: Towards Dynamic Behavior and Secure Routing forWireless Sensor Networks,» 2006.

[31] Z. Zhao, H. Hu, G. Ahn και R. Wu, «Risk-Aware Response for Mitigating MANET Routing Attacks,» σε *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, 2010.

[32] Zhao, Ziming and Hu, Hongxin and Ahn, Gail-Joon and Wu, Ruoyu, «Risk-Aware Mitigation For MANET Routing Attacks,» *IEEE Transactions on dependable and secure computing*, τόμ. 9, p. 250–260, 2011.

Παράρτημα Α

Σε αυτό το παράρτημα παρατίθεται κομμάτι του κώδικα που χρησιμοποιήθηκε για την ενημέρωση των κόμβων ώστε να ξεκινήσει η διαδικασία της επαναδρομολόγησης από τους υπόλοιπους κόμβους.

```
/*---------------------------------------------------------------------------*/
/*
 * This function is called when we receive a packet from broadcast that is responsible for telling about a malicious node in network.
*/
static void
broadcast_recv(struct broadcast_conn *broadcast, const linkaddr_t *from)
{
        if(packetbuf_datalen()==sizeof(struct packet_infected)){
          struct packet_infected *infected_announcement = (struct packet_infected *) packetbuf_dataptr();


                struct table_entry_neighbor *current_entry;

                for(current_entry=list_head(neighbor_table);current_entry!=NULL; current_entry=current_entry->next) {
                        if(linkaddr_cmp(&infected_announcement->addr, &current_entry->addr)){
                                if(current_entry->trusted==0 && infected_announcement->attack_type ==0){
                                        announcement_bump(&example_announcement);
                                }
                                if(infected_announcement->attack_type == 0 ){
                                        current_entry->trusted=1;
                                }else{
                                        current_entry->trusted=0;
                                }
                                if(DEBUG){
```

```
                                        printf("-%d.%d - trusted bit %d\n",current_entry->addr.u8[0],current_entry->addr.u8[1],current_entry->trusted);
                                    }
                                }
                            }
                        }
                    }
                }
                static const struct broadcast_callbacks broadcast_call = {broadcast_recv};
                static struct broadcast_conn broadcast;


                PROCESS(broadcasting_warnings, "broadcasting_warnings");


                PROCESS_THREAD(broadcasting_warnings,ev,data){

                    PROCESS_EXITHANDLER(broadcast_close(&broadcast);)

                    PROCESS_BEGIN();
                    //Sets the neighbor table
                    struct arguments{
                        struct announcement a;
                        void * b;
                    };
                    struct arguments * args = (struct arguments *)data;
                    example_announcement=(struct announcement)args->a;
                    neighbor_table=args->b;
                    displayNeighborTable();

                    /* Open a broadcast connection on Rime channel */
                    broadcast_open(&broadcast, CHANNEL, &broadcast_call);
                    while(1){
                        struct packet_infected pqt= {
                            .addr.u8[0]=linkaddr_node_addr.u8[0],
                            .addr.u8[1]=linkaddr_node_addr.u8[1],
                            .attack_type= mids_wsp_attack.alert,
                            .informer_type=0,
```

```
        };
        if(DEBUG){
           printf("my rime[0] %d \n",pqt.addr.u8[0]);
           printf("my rime[1] %d \n",pqt.addr.u8[1]);
           printf("attack_type : %d\n", mids_wsp_attack.alert);
           printf("sizeof buffer : %d\n",sizeof(struct packet_infected));
        }
        printf("Broadcasting Alert Benign\n");

        packetbuf_copyfrom(&pqt,sizeof(struct packet_infected));
        broadcast_send(&broadcast);

        static struct etimer et_For_Broadcast;
        etimer_set(&et_For_Broadcast, CLOCK_SECOND*60);
        PROCESS_WAIT_EVENT_UNTIL(etimer_expired(&et_For_Broadcast));
    }
    PROCESS_END();

}
AUTOSTART_PROCESSES(&broadcasting_warnings);
```

Παράρτημα Β

Σε αυτό το παράρτημα παρατίθεται κομμάτι του κώδικα που χρησιμοποιήθηκε για τον επαναπρογραματισμό των κόμβων στην επίθεση Sinklhole μετά την αλλαγή του reboot flag από το σύστημα ανίχνευσης (IDS). Σε αυτή την προσομοίωση, αυτόματα ξεκινά η επίθεση αντί του κανονικού προγράμματος, αλλά θεωρείται ότι θα ξεκινά το καλοήθεις πρόγραμμα και όταν εντοπισθεί επίθεση στο δίκτυο από το σύστημα ανίχνευσης (IDS) και αλλάξει το reboot flag, ο υπόλοιπος ο κώδικα όπως φαίνεται πιο κάτω αντικαθιστά την κακόβουλη ενέργεια.

```
/*
 * [WSP Modified]
 * This is the struct of a neigbor table entry.
 *  - next => pointer to the next table entry
 *  - addr => neigbor rime address
 *  - rssi => rssi between current sensor and its neighbor
 *  - hops => hops from neighbor to sink
 *  - trusted => if set to 1
 */
struct table_entry_neighbor {
   struct table_entry_neighbor *next;
   linkaddr_t addr;
   uint16_t rssi;
   int hops;
   uint8_t trusted:1;
};

volatile extern reboot_flag reboot;

MEMB(neighbor_mem, struct table_entry_neighbor, MAX_NEIGHBORS);
LIST(neighbor_table);

/*---------------------------------------------------------------------------*/
PROCESS(start_SinkHole, "SinkHole master process");

/*---------------------------------------------------------------------------*/
PROCESS_THREAD(start_SinkHole, ev, data){
```

```
    PROCESS_EXITHANDLER(printf("Stopping start_SinkHole\n");)

    PROCESS_BEGIN();
    // Set sink info
    static linkaddr_t myRime;
    myRime.u8[0] =linkaddr_node_addr.u8[0];
    myRime.u8[1] =linkaddr_node_addr.u8[1];

    /* Initialize the memory for the neighbor table entries. */
    memb_init(&neighbor_mem);

    /* Initialize the list used for the neighbor table. */
    list_init(neighbor_table);

    process_start(&sinkhole,neighbor_table);
    process_start(&recovery_malicious_transimision_reboot,neighbor_table);

    /* Allow some time for the network to settle. */
    static struct etimer et;

    etimer_set(&et, 120 * CLOCK_SECOND);
    PROCESS_WAIT_UNTIL(etimer_expired(&et));

    /*mids*/
    process_start(&mids_all, NULL);
    mids_apps_init(&mids_app);

    PROCESS_WAIT_UNTIL(reboot.flag==1);
    printf("Start rebooting\n");

    process_exit(&sinkhole);
    process_exit(&recovery_malicious_transimision_reboot);

    printf("process sinkhole : %d\n", process_is_running(&sinkhole));
    printf("process recovery_malicious_transimision_reboot : %d\n",
process_is_running(&recovery_malicious_transimision_reboot));
```

```
    uint8_t longaddr[8];
    uint16_t shortaddr;

    // set rime address back to original
    linkaddr_set_node_addr(&myRime);

    shortaddr = (linkaddr_node_addr.u8[0] << 8) + linkaddr_node_addr.u8[1];
    memset(longaddr, 0, sizeof(longaddr));

    // set the mac address to original
    cc2420_set_pan_addr(0xABCD, shortaddr, longaddr);

    printf("my new rime :%d.%d\n",linkaddr_node_addr.u8[0],linkaddr_node_addr.u8[1]);


    process_start(&rerouting_benign,neighbor_table);
    struct arguments{
        struct announcement a;
        void * b;
    };
    struct arguments args;
    args.a=(const struct announcement ){0};
    args.b=neighbor_table;
    process_start(&broadcasting_warnings,&args);
    reboot.flag=0;
    process_exit(&start_SinkHole);

    printf("process exit\n");

    PROCESS_END();
}
AUTOSTART_PROCESSES(&start_SinkHole);
```

Παράρτημα Γ

Σε αυτό το παράρτημα παρουσιάζεται πρόγραμμα για την εύκολη καταχώριση πειραμάτων στο περιβάλλον FIT-IOT Lab. Για να καταχωρήσει πειράματα εύκολα κάποιος με για τις τοπολογίες που περιεγράφηκαν πιο πάνω μπορεί να κάνει χρήση του πιο κάτω προγράμματος. Με αυτό το πρόγραμμα μπορείς να καταχωρήσεις πειράματα στην πλατφόρμα, έτσι ώστε με την σειρά όλοι οι κόμβοι να τρέξουν τον κακόβουλο κώδικα. Άρα κάθε φορά που εκτελείται το πιο κάτω script καταχωρούνται συνολικά 24 πειράματα στην πλατφόρμα. Το πρόγραμμα αυτό είναι ένα bash script που δέχεται 3 παραμέτρους εισόδου. Η πρώτη είναι το όνομα του binary αρχείου που θα τρέξουν όλοι οι κόμβοι, ένας σε κάθε πείραμα (χωρίς την προσθήκη της επέκτασης). Η δεύτερη είσοδος είναι το ID του Sink node που θα χρησιμοποιηθεί για τα πειράματα, και τέλος η διάρκεια σε λεπτά που θα τρέξει το κάθε πείραμα. Επίσης ελέγχει εάν έγινε η καταχώρηση του πειράματος, εάν δεν έγινε ξανά προσπαθεί μέχρι να τα καταφέρει.

```
#!/bin/bash

# ./run ForwardingAlgorihtm_SF_BlockNode SINK_ID DURATION

if [ $# -ne 3 ]
then
	echo "USAGE"
	echo "./run Malicious_With_Out_.c SINK_ID DURATION_IN_MIN"
	echo "./run benign SINK_ID DURATION_IN_MIN"
	exit
fi
rm run_ids.txt
touch run_ids.txt
runs_id=0
nodes="152 153 154 155 156 169 170 171 172 173 191 198 193 194 195 205 206 207 208
209 225 226 227 228 229"
nameOfRun=`echo $1 | cut -d '_' -f1 --complement`
if [ "$1" == "benign" ]
then
	nodesForward=`echo $nodes| tr ' ' '+'`
	experiment=`echo "iotlab-experiment submit -n $1-sink$2 -d $3 -l
lille,m3,$nodesForward,ForwardingAlgorithmGrid_5_5.iotlab-m3 --site-association
lille,script=aggregator_script" `
	echo "$experiment"
	eval $experiment
	echo "Total submits: 1"
	exit
fi
for node in $nodes
do
	if [ "$node" != "$2" ]
	then
		nodesForward=`echo $nodes| tr ' ' '+' |sed "s/$node//g" |sed "s/++/+/g"`
```

```
                echo ""
                experiment=`echo "iotlab-experiment submit -n $nameOfRun-sink$2_m$node -d
$3 -l lille,m3,$nodesForward,ForwardingAlgorithmGrid_5_5.iotlab-m3 -l
lille,m3,$node,$1.iotlab-m3 --site-association lille,script=aggregator_script 2>&1" |sed
"s/,+/,/g" |sed "s/+,/,/g"`
                echo "$experiment"
                output=`eval "$experiment"`
                while [[ "$output" == *"HTTP"* ]];
                do
                        sleep 1
                        output=`eval "$experiment"`
                done
                echo "$output">>run_ids.txt
                runs_id=$(( runs_id+1 ))
        fi
done
echo "Total submits: $runs_id "
```