ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΛΟΠΟΝΝΗΣΟΥ

ΣΧΟΛΗ ΟΙΚΟΝΟΜΙΑΣ, ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

Πτυχιακή Εργασία

Χρήστος-Μηνάς Μαθάς

Α.Μ. 2022201300086

Αξιολόγηση του Apache Spot με χρήση δοκιμών διείσδυσης

Επιβλέπων καθηγητής: Βασιλάκης Κωνσταντίνος

Τρίπολη | Μάρτιος 2018

Περιεχόμενα

Ευρετήριο Εικόνων

Ευρετήριο Πινάκων

Περίληψη

Στην εργασία αυτή αξιολογούμε το σύστημα ανίχνευσης διεισδύσεων Apache Spot, εκτελώντας διάφορα είδη επιθέσεων ενάντια σε ένα δίκτυο που επιβλέπει το Apache Spot. Ξεκινάμε παρουσιάζοντας το Apache Spot με τις δομικές λειτουργίες του και τα επιμέρους χαρακτηριστικά του. Στη συνέχεια παρουσιάζουμε τα εργαλεία δοκιμών διείσδυσης που χρησιμοποιήθηκαν για την εκτέλεση των επιθέσεων περιγράφοντας τα χαρακτηριστικά τους, συνεχίζοντας με την περιγραφή και παρουσίαση με χρήση διαγραμμάτων της διάταξης δικτύου που χρησιμοποιήσαμε για το Apache Spot και την εκτέλεση των επιθέσεων. Παράλληλα, παρουσιάζουμε τα bash scripts που γράφτηκαν για την αυτοματοποίηση του μεγαλύτερου μέρους της διαδικασίας. Προχωρώντας, εξηγούμε τον τρόπο λειτουργίας της κάθε επίθεσης που εκτελέσαμε και παρουσιάζουμε τα αποτελέσματα που πήραμε, τα οποία στη συνέχεια συγκεντρώνουμε και συγκρίνουμε ώστε να εξάγουμε συμπεράσματα. Από τα συμπεράσματα μας μπορεί να κατανοηθεί καλύτερα ο τρόπος λειτουργίας του Apache Spot και να καταδειχθεί το πως ανταποκρίνεται ανάλογα τη διάταξη δικτύου, τη παραμετροποίηση του και το είδος της επίθεσης. Τέλος, βασιζόμενοι στα συμπεράσματα μας αναφέρουμε περαιτέρω δυνατότητες έρευνας που προκύπτουν από αυτά.

Λέξεις – κλειδιά: Apache Spot, δοκιμή διείσδυσης, αξιολόγηση απόδοσης, εργαλεία ανίχνευσης διεισδύσεων

Abstract

In this thesis, we are evaluating the intrusion detection system Apache Spot, by deploying various types of attacks against a network monitored by Apache Spot. We begin by presenting Apache Spot with its core functions and its individual characteristics. Consequently we present the penetration testing tools used for the deployment of the attacks and describing their characteristics. We also present the network setup we used for Apache Spot and the deployment of the attacks by using diagrams. In parallel, we present the bash scripts we created in order to automate the largest part of the procedure. As we move forward, we explain the way each attack we deployed works and we present the results we obtained, which we gather and compare in order to extract conclusions. These conclusions allow us to better understand the way Apache Spot works and establish how it responds depending on the network setup, its configuration and the type of the attack. Finally, based on our conclusions we propose further research possibilities that occur from them.

Keywords: Apache Spot, penetration testing, performance evaluation, intrusion detection system

Ευχαριστίες

Αισθάνομαι την ανάγκη να ευχαριστήσω ιδιαίτερα τον επιβλέποντα καθηγητή κύριο Κωνσταντίνο Βασιλάκη για τις παρατηρήσεις του και την καθοδήγηση του, καθώς και τους συναδέλφους μου στο *ΕΚΕΦΕ Δημόκριτος*, Γιώργο Ξυλούρη, Χρήστο Ξυλούρη και Δημήτρη Χριστινάκη για τη βοήθεια τους σε κρίσιμες στιγμές της εργασίας. Επίσης θα ήθελα να ευχαριστήσω θερμά τον Ricardo Barona(Oracle) για τη πολύτιμη βοήθεια του, καθώς και τον Gustavo Lujan Moreno(Intel). Για την ευκαιρία της Πρακτικής Άσκησης στο *ΕΚΕΦΕ Δημόκριτος*, ευχαριστώ ιδιαίτερα την κυρία Ζαρμπούτη και τον κύριο Μπατιστάτο.

1 Εισαγωγή

Τη σημερινή εποχή οι υπολογιστές και το διαδίκτυο αποτελούν αναπόσπαστο κομμάτι της ανθρώπινης πραγματικότητας, τόσο σε προσωπικό όσο και σε επαγγελματικό επίπεδο. Όλο και περισσότερες εργασίες εκτελούνται με χρήση των υπολογιστών και του διαδικτύου και οι υποδομές των οργανισμών βασίζονται πλέον κατά κύριο λόγο σε εξελιγμένα υπολογιστικά συστήματα αλλά και το διαδίκτυο. Το διαδίκτυο χρησιμοποιείται για όλο και περισσότερες εργασίες της καθημερινότητας, από απλές, όπως η μεταφορά και αποθήκευση δεδομένων μέχρι πιο σύνθετες, όπως τραπεζικές συναλλαγές.

Ένα από τα πιο σημαντικά ζητήματα που έχει ανακύψει από αυτή τη τεχνολογική επανάσταση είναι η ασφάλεια και, όσο πιο πολύ εξελίσσεται το διαδίκτυο και οι εργασίες που εκτελούνται μέσω αυτού, τόσο πιο επιτακτική καθίσταται η πλήρης διαχείριση του θέματος της ασφάλειας. Οι κυβερνοεπιθέσεις αυξάνονται συνεχώς και μαζί αυξάνεται η κλίμακα επίδρασης τους, καθώς πλέον υπάρχουν μέθοδοι επιθέσεων παγκόσμιας εμβέλειας. Έτσι, τα κράτη και οι οργανισμοί επενδύουν όλο και περισσότερους πόρους για την εξασφάλιση των πόρων και των δεδομένων τους και χρησιμοποιώντας εξελιγμένα συστήματα για τον εντοπισμό και την αντιμετώπιση επιθέσεων και προσλαμβάνοντας ειδικευμένο προσωπικό.

Ένα από τα πιο σημαντικά προβλήματα που προκύπτει στην ασφάλεια δικτύων σήμερα είναι τα τεράστια μεγέθη δεδομένων που καλείται να επεξεργαστεί ένα σύστημα ασφαλείας ώστε να εντοπίσει ή/και να αντιμετωπίσει μία ή περισσότερες επιθέσεις σε πραγματικό χρόνο. Το σύστημα ανίχνευσης διεισδύσεων (IDS) Apache Spot έρχεται να αντιμετωπίσει το ζήτημα του αποδοτικού εντοπισμού επιθέσεων σε ένα δίκτυο με πολύ υψηλό φόρτο κίνησης.

Στη παρούσα πτυχιακή εργασία πρόκειται να αξιολογήσουμε το σύστημα ανίχνευσης διεισδύσεων (IDS) Apache Spot, μέσω της διενέργειας δοκιμών διείσδυσης, χρησιμοποιώντας επιθέσεις διαφόρων ειδών σε ένα δίκτυο που επιτηρείται και προστατεύεται από το Apache Spot.

Η δομή της εργασίας είναι η εξής: στο κεφάλαιο 2 γίνεται η παρουσίαση του Apache Spot. Παρουσιάζουμε τα βασικά χαρακτηριστικά του, τις δομικές λειτουργίες του, τα τμήματα που το αποτελούν και περιγράφουμε τον τρόπο λειτουργίας του. Στο τρίτο κεφάλαιο παρουσιάζουμε τα εργαλεία δοκιμών διείσδυσης που χρησιμοποιήσαμε, περιγράφοντας τα χαρακτηριστικά τους και τις δυνατότητες τους. Στο τέταρτο κεφάλαιο παρουσιάζουμε την διάταξη που χρησιμοποιήθηκε για το δίκτυο υπό επίβλεψη, για το Apache Spot και την εκτέλεση των επιθέσεων. Στο πέμπτο κεφάλαιο επεξηγούμε τις επιθέσεις που εκτελέσαμε και παράλληλα παρουσιάζουμε τα αποτελέσματα που καταγράφηκαν από τη λειτουργία του Apache Spot για κάθε επίθεση. Στο έκτο κεφάλαιο αξιολογούμε τα αποτελέσματα του πέμπτου κεφαλαίου και τα συγκρίνουμε μεταξύ τους καταλήγοντας σε συμπεράσματα για την λειτουργία και την αποδοτικότητα του Apache Spot.

2 Το Apache Spot

Το Apache Spot είναι ένα λογισμικό ανοιχτού κώδικα το οποίο δημιουργήθηκε με στόχο να διευκολύνει την ανίχνευση, διερεύνηση και αποκατάσταση απειλών με χρήση μηχανικής μάθησης. Ένα από τα πλεονεκτήματα του Apache Spot έγκειται στη δυνατότητα του, α) να επεξεργαστεί δεδομένα από διαφορετικές πηγές, β) να τα συσχετίσει ώστε να εντοπίσει ανωμαλίες και έτσι γ) να παρέχει χρήσιμες πληροφορίες στους υπεύθυνους του δικτύου.

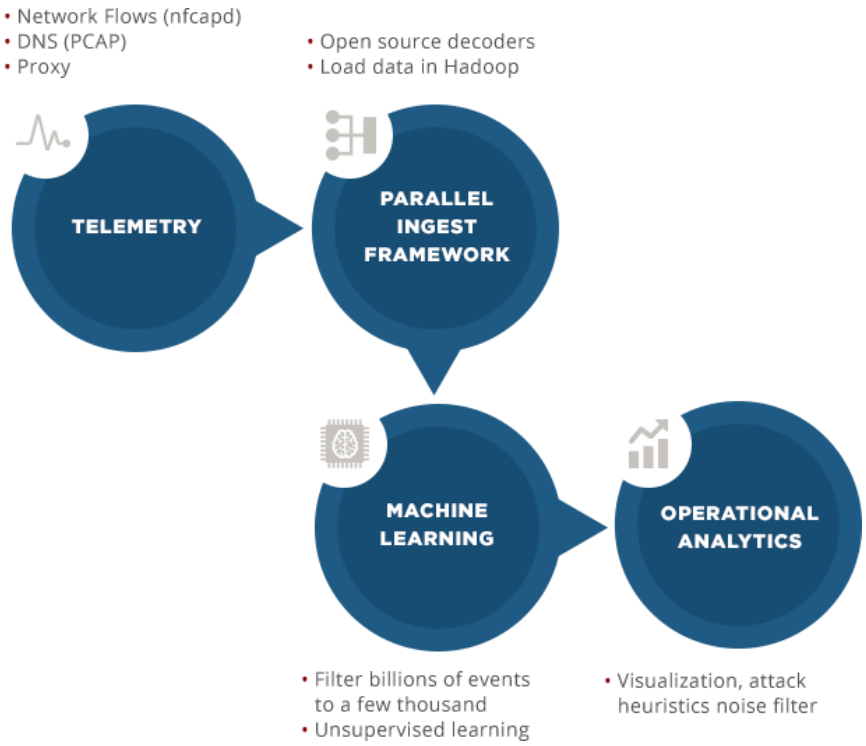
Βασικό χαρακτηριστικό του αποτελεί το γεγονός πως δεν βασίζεται σε αιτιοκρατικές τεχνικές και στη γνώση καταγεγραμμένων μέχρι τώρα απειλών (signature-based detection), αλλά κάνοντας χρήση μηχανικής μάθησης μπορεί να επεξεργαστεί αποδοτικά δισεκατομμύρια γεγονότα φιλτράροντας τα και μειώνοντας το προς περαιτέρω επεξεργασία πλήθος γεγονότων σε μερικές χιλιάδες και να ανακαλύψει επιθέσεις που δεν είναι ανιχνεύσιμες από τα καθιερωμένα εργαλεία ανίχνευσης. Η δυνατότητα του αυτή προκύπτει από πολλά χαρακτηριστικά του, με τα πιο βασικά να είναι (Intel, 2016):

- Μαθαίνει αυτόματα τι είναι «φυσιολογικό» για ένα συγκεκριμένο δίκτυο και βελτιώνεται συνεχώς.
- Δημιουργία προφίλ ρίσκου για τις κινήσεις που θεωρούνται ύποπτες.
- Οι αλγόριθμοι «εκπαιδεύονται» χρησιμοποιώντας τα δεδομένα από τα προφίλ ρίσκου, μειώνοντας τις «ψευδώς θετικές» αναφορές («false positives») και έτσι αποφεύγοντας την επανάληψη τους.

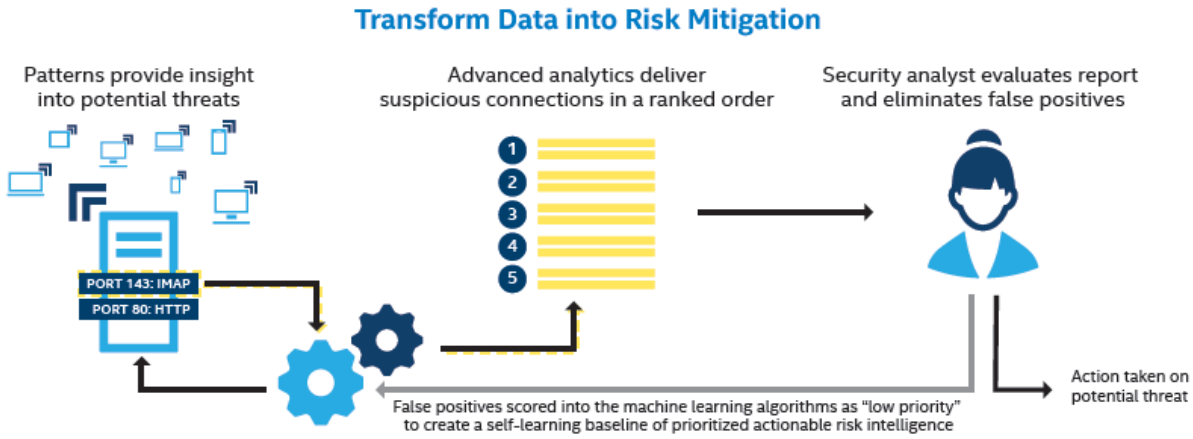
2.1 Δομικές Λειτουργίες

Το Apache Spot συλλέγει δεδομένα από την κίνηση στο δίκτυο το οποίο επιβλέπει. Συγκεκριμένα συλλέγει δεδομένα α) ροής (Flow), β) Συστήματος Ονομάτων Τομέων (DNS), και γ) διακομιστών μεσολάβησης (Proxy). Στη συνέχεια συνθέτει τα δεδομένα και αναφέρει ύποπτες συνδέσεις και μοτίβα επιθέσεων. Τα συσχετισμένα δεδομένα φορτώνονται σε ένα Apache Hadoop cluster (Wikipedia, 2018a) (βλ. σχετική παράγραφο στη συνέχεια). Το Apache Spot χρησιμοποιεί αλγόριθμους μηχανικής μάθησης και εργαλεία ανάλυσης λειτουργιών (operational analytics)[1] ώστε να εντοπίσει, βαθμολογήσει και αναφέρει τις ύποπτες κινήσεις. Μια γραφική αναπαράσταση της διαδικασίας παρουσιάζεται στην Εικόνα 1. Τα δεδομένα αυτά εξετάζονται από τους υπεύθυνους του δικτύου ώστε να πάρουν τα κατάλληλα μέτρα, ενώ παράλληλα δίνεται η δυνατότητα βαθμολόγησης του κάθε αποτελέσματος από τον χρήστη λειτουργώντας ως ανατροφοδότηση για τον αλγόριθμο μηχανικής μάθησης, όπως παρουσιάζεται στην Εικόνα 2.

[1] Το Operational Analytics είναι ένας πιο ακριβής όρος για ένα είδος επιχειρηματικών αναλύσεων οι οποίες εστιάζουν στη βελτίωση των υπαρχουσών λειτουργιών
https://www.techopedia.com/definition/29495/operational-analytics



Εικόνα 1 - Δομικές Λειτουργίες (http://spot.incubator.apache.org/get-started/)



Εικόνα 2 – Διαδικασία Ημι-Επιβλεπόμενης Μάθησης (https://goo.gl/UpRFfM)

Η διαδικασία που περιγράφηκε παραπάνω μπορεί να διαχωριστεί σε 3 δομικές λειτουργίες ή φάσεις.

- Λήψη Δεδομένων (Data Ingestion)
- Μηχανική Μάθηση (Machine Learning)
- Ανάλυση Λειτουργιών (Operational Analytics)

Πριν προχωρήσουμε στη περιγραφή των δομικών λειτουργιών, θα παραθέσουμε μια επιγραμματική περιγραφή κάποιων από τα κύρια επιμέρους εργαλεία που χρησιμοποιεί το Apache Spot ώστε να είναι πιο εύληπτη η συνέχεια στον αναγνώστη.

Apache Hadoop

Το Apache Hadoop είναι ένα πλαίσιο λογισμικού ανοιχτού κώδικα (open source software framework) που χρησιμοποιείται για κατανεμημένη αποθήκευση και επεξεργασία μεγάλων δεδομένων (big data). Το κομμάτι της αποθήκευσης ονομάζεται Hadoop Distributed File System ή HDFS που είναι και το όνομα που θα χρησιμοποιήσουμε από δω και πέρα. (Wikipedia, 2018a)

Apache Spark

Το Apache Spark είναι ένα πλαίσιο υπολογιστικής συστοιχιών (cluster computing) ανοιχτού κώδικα (open source cluster computing framework) για γρήγορη επεξεργασία μεγάλων δεδομένων. Χαρακτηρίζεται από διαλειτουργικότητα καθώς μπορεί να εκτελεστεί πάνω στο Hadoop, μόνο του, ή σε ένα πλαίσιο υπολογιστικής νέφους (cloud computing) και μπορεί να επεξεργαστεί δεδομένα από διάφορες πηγές συμπεριλαμβανομένου του HDFS. (Wikipedia, 2018b) (Apache, 2018a)

Apache Hive

Το Apache Hive είναι ένα λογισμικό διαχείρισης αποθηκών δεδομένων (data warehouse software), το οποίο έχει χτιστεί βασιζόμενο στο Apache Hadoop και παρέχει συνοπτική παρουσίαση, δυνατότητα αναζήτησης με ερωτήσεις (querying) και ανάλυση δεδομένων. (Wikipedia, 2018c)

Apache Kafka

Το Apache Kafka είναι μια πλατφόρμα επεξεργασίας ροών ανοιχτού κώδικα (open source stream processing platform) που δημιουργήθηκε με στόχο να παρέχει μία ενοποιημένη, υψηλής απόδοσης, και χαμηλής καθυστέρησης πλατφόρμα για τον χειρισμό εισροών δεδομένων πραγματικού χρόνου. (Wikipedia, 2018d)

2.2 Λήψη Δεδομένων

Επειδή το ζήτημα των μεγάλων δεδομένων (big data) και τα οφέλη της ανάλυσης τους είναι ένα από τα πιο σημαντικά ζητήματα των τελευταίων χρόνων στον κλάδο της επιστήμης των υπολογιστών, έχουν ανακύψει πολλά ζητήματα ως προς την αποδοτική διαχείριση τους. Ο κατανεμημένος αρχιτεκτονικός σχεδιασμός του πλαισίου λήψης δεδομένων (ingestion framework) του Apache Spot έχει δημιουργηθεί με στόχο την εξάλειψη της απώλειας δεδομένων και την εξασφάλιση της διαθεσιμότητας της υπηρεσίας, ακόμα και σε καταστάσεις όπου ο φόρτος δεδομένων και οι απαιτήσεις σε υπολογιστική

ισχύ είναι μεγάλα. Έτσι έρχεται να προτείνει λύσεις για αρκετά ζητήματα που έχουν ανακύψει από τον τομέα των μεγάλων δεδομένων. Βασικά παραδείγματα εργαλείων που επιτρέπουν την αποδοτική διαχείριση δεδομένων αποτελούν το Apache Spark, το Hadoop Distributed File System (HDFS) και το Cloudera Distribution for Hadoop (βλ. (Cloudera, 2018)). Στη συνέχεια περιγράφουμε πως λειτουργεί η λήψη δεδομένων.

2.2.1 Τηλεμετρία (Telemetry)

Η τηλεμετρία είναι η διαδικασία κατά την οποία συλλέγονται τα δεδομένα που θα δοθούν ως είσοδος στο πλαίσιο λήψης δεδομένων. Συγκεκριμένα, δεδομένα από την κίνηση στο δίκτυο υπό επίβλεψη συγκεντρώνονται σε διαφορετική μορφή ανάλογα με τη περίπτωση. Το Apache Spot υποστηρίζει τις εξής κατηγορίες:

- NetFlow: δεδομένα ροής δικτύου που συλλέγονται από εισερχόμενες και εξερχόμενες ροές σε μία διεπαφή παρακολούθησης (monitoring interface). Τα δεδομένα NetFlow παρέχουν πληροφορίες όπως τη πηγή και τον προορισμό της κίνησης. Στο πλαίσιο του Apache Spot, τα δεδομένα ροής δικτύου (flow) αποθηκεύονται σε αρχεία τύπου nfcapd[2]. Στη συνέχεια θα χρησιμοποιείται ο όρος Flow, όποτε γίνεται αναφορά σε αυτό το κομμάτι της Τηλεμετρίας.
- DNS: τα δεδομένα DNS αποθηκεύονται σε αρχεία pcap[3]. Τα αρχεία pcap περιέχουν δεδομένα πακέτων δικτύου και δημιουργούνται κατά την διαδικασία ζωντανής καταγραφής δικτύου (live network capture).
- Proxy: τα δεδομένα διακομιστών μεσολάβησης (proxy) αποθηκεύονται σε αρχεία καταγραφής (logs) τύπου bluecoat[4].

Οι παραπάνω λειτουργίες αναφέρονται ως Telemetry (βλ. Εικόνα 1) και αποτελούν απαραίτητη προεργασία για την εκκίνηση της λήψης δεδομένων.

2.2.2 Συλλέκτες (Collectors)

Οι συλλέκτες (collectors ή master collectors) αποτελούν διεργασίες παρασκηνίου (daemons) οι οποίες παρακολουθούν συγκεκριμένες τοποθεσίες του συστήματος αρχείων και συλλέγουν δεδομένα από αυτές. Τα δεδομένα είναι αυτά που δημιουργούνται από τη διαδικασία της τηλεμετρίας όπως περιγράφηκε παραπάνω και αποθηκεύονται στις επιλεγμένες τοποθεσίες - που παρακολουθούνται από τους συλλέκτες - προς συλλογή. Τα δεδομένα αναλύονται χρησιμοποιώντας τα κατάλληλα εργαλεία ανάλογα με τον τύπο δεδομένων. Για αρχεία nfcapd από τις ροές δικτύου (flow) χρησιμοποιείται το nfdump[5]. Για αρχεία pcap από το DNS χρησιμοποιείται το tshark[6]. Τέλος για αρχεία καταγραφής διακομιστών μεσολάβησης (proxy) χρησιμοποιείται το Spark Streaming (Apache, 2018b). Μια γραφική σύνοψη της λειτουργίας των συλλεκτών βρίσκεται στην Εικόνα 3. Τα δεδομένα αυτά σώζονται στη συνέχεια στην αναγνώσιμη μορφή τους στο HDFS, ενώ σώζονται και στη Hive σε μορφή

[2] Είναι ένα πρόγραμμα καταγραφής δεδομένων NetFlow και μέρος της συλλογής εργαλείων nfdump.
[3] Είναι ένας τύπος αρχείου για καταγραφή κίνησης δικτύου.
[4] Είναι ένας τύπος αρχείου για καταγραφή κίνησης διακομιστών μεσολάβησης σε ένα δίκτυο.
[5] Είναι ένα εργαλείο για εμφάνιση και ανάλυση δεδομένων NetFlow.
[6] Είναι ένα εργαλείο που παρέχει τη δυνατότητα καταγραφής πακέτων που στέλνονται εντός ενός δικτύου.

Avro-parquet[7] ώστε να είναι προσβάσιμα μέσω SQL queries. Το τελευταίο κομμάτι το αναλαμβάνουν οι εργάτες (workers) όπως εξηγείται παρακάτω.

2.2.3 Εργάτες (Workers)

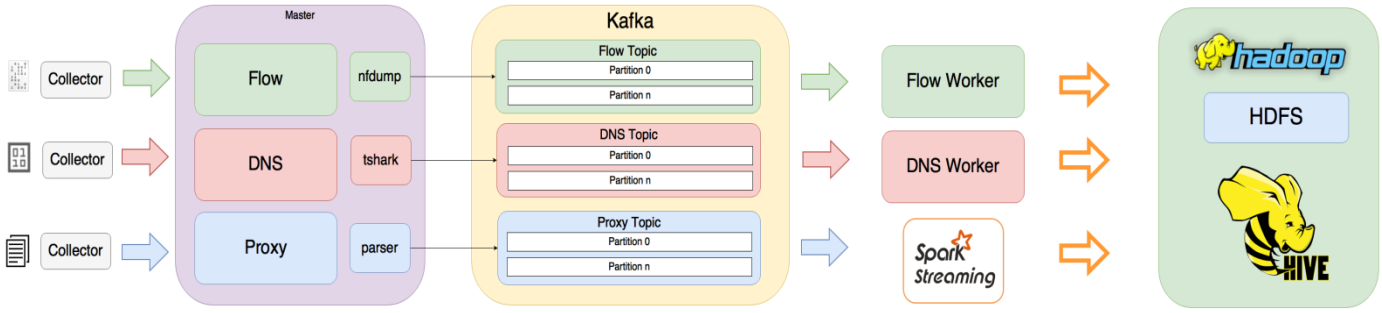
Όταν δημιουργείται ένα νέο στιγμιότυπο λήψης δεδομένων, δημιουργείται και του ανατίθεται ένα θέμα (topic), το οποίο λειτουργεί σαν αναγνωριστικό αυτού του στιγμιότυπου. Αναφέρεται και ως *Kafka topic* καθώς δημιουργείται από το Apache Kafka, στη συνέχεια ωστόσο της πτυχιακής εργασίας θα αναφέρεται απλά ως *θέμα*. Το Kafka λαμβάνει τα δεδομένα από τους συλλέκτες ώστε να τα επεξεργαστούν οι εργάτες (workers).

Οι εργάτες είναι διεργασίες παρασκηνίου (daemons) και είναι συνυφασμένοι από τη δημιουργία τους με κάποιο συγκεκριμένο θέμα και διαμέριση (partition) του Apache Kafka. Οι εργάτες επεξεργάζονται τα δεδομένα που δέχονται από τους συλλέκτες ώστε να τα μετατρέψουν σε Avro-parquet μορφή – όπως προαναφέραμε- για να αποθηκευτούν σε πίνακες της Hive, ώστε εν συνεχεία να αποτελέσουν είσοδο για τον αλγόριθμο μηχανικής μάθησης.

Υπάρχουν δύο είδη εργατών:

1. Οι εργάτες Python που χρησιμοποιούν πολυνηματισμό (multithreading) για να επεξεργαστούν τα δεδομένα τύπου Flow και DNS με τα αντίστοιχα εργαλεία.
2. Οι εργάτες Spark Streaming που εκτελούν μία εφαρμογή Spark για να διαβάσουν δεδομένα από το Kafka. Οι συγκεκριμένοι εργάτες αναλαμβάνουν τα αρχεία καταγραφής διακομιστών μεσολάβησης (proxy), όπως φαίνεται και στην Εικόνα 3.

Στην Εικόνα 3 φαίνεται μια γραφική απεικόνιση της λειτουργίας των εργατών και του Kafka, όπου τα αναφερόμενα ως partitions ορίζονται από τον αριθμό των εργατών στο topic.



Εικόνα 3 – Διαδικασία Λήψης Δεδομένων (https://goo.gl/fjN5bp)

(Apache, 2016a) (SPACE Hellas, i2cat, ubiwhere, Politecnico Di Torino, Infili, Telefonica, Orion Innovations, incites, Agenzia per l'Italia Digitale, Hewlett Packard Enterprise, NCSR Demokritos, 2017)

[7] Είναι ένας τύπος αρχείου με δεδομένα οργανωμένα σε στήλες.

2.3 Μηχανική Μάθηση

Το κομμάτι της μηχανικής μάθησης του Apache Spot περιέχει ρουτίνες για ανάλυση ύποπτων συνδέσεων από δεδομένα Flow, DNS και Proxy που έχουν συγκεντρωθεί από το δίκτυο υπό επίβλεψη κατά την φάση της λήψης δεδομένων. Στο πλαίσιο των αναλύσεων αυτών, οι ρουτίνες επεξεργάζονται τα καταγεγραμμένα συμβάντα δικτύου και παράγεται μία λίστα που περιλαμβάνει τα συμβάντα που θεωρούνται λιγότερο πιθανό να εμφανιστούν σαν νομότυπη κυκλοφορία, και αυτά είναι που θεωρούνται ως τα πιο ύποπτα.

Το Apache Spot χρησιμοποιεί θεματική μοντελοποίηση (topic modeling) για την εύρεση φυσιολογικής και μη συμπεριφοράς. Στη μηχανική μάθηση το θεματικό μοντέλο (Wikipedia, 2018e) είναι ένα στατιστικό μοντέλο για εύρεση αφηρημένων θεμάτων (abstract topics) που εμφανίζονται σε μια συλλογή εγγράφων και σημασιολογικών δομών (semantic structures) σε ένα κείμενο. Βασίζεται, δηλαδή, στην εξής λογική: δεδομένου ότι ένα έγγραφο σχετίζεται με ένα συγκεκριμένο θέμα, θα περιμέναμε συγκεκριμένες λέξεις να εμφανίζονται συχνά. Για τον σκοπό αυτόν χρησιμοποιείται ο αλγόριθμος Latent Dirichlet Allocation (LDA) (Wikipedia, 2017a), όπου στα πλαίσια του Spot έχει υλοποιηθεί με χρήση της Spark MLlib (Apache, 2018c). Το Spot τον εφαρμόζει πάνω στα δεδομένα που έχουν συλλεχθεί από το δίκτυο, μετατρέποντας τις καταχωρήσεις των αρχείων καταγραφής σε λέξεις μέσω συνάθροισης και διακριτοποίησης (discretization). Έτσι, τα έγγραφα αντιστοιχούν σε διευθύνσεις IP, οι λέξεις σε καταχωρήσεις αρχείων καταγραφής (σχετιζόμενα με κάποια IP) και τα θέματα (topics) σε προφίλ συνηθισμένης δραστηριότητας δικτύου. Το αποτέλεσμα είναι πως το Apache Spot συνάγει ένα πιθανοτικό μοντέλο για τη συμπεριφορά κάθε διεύθυνσης IP. Το μοντέλο αναθέτει σε κάθε καταχώρηση στα αρχεία καταγραφής μία εκτιμώμενη πιθανότητα. Τα γεγονότα (καταχωρήσεις) με χαμηλότερες πιθανότητες σημειώνονται ως ύποπτα για περαιτέρω ανάλυση.

(Apache, 2016b) (Apache, 2016c)

2.3.1 Ανάλυση Ύποπτων Κινήσεων (Suspicious Connects Analysis)

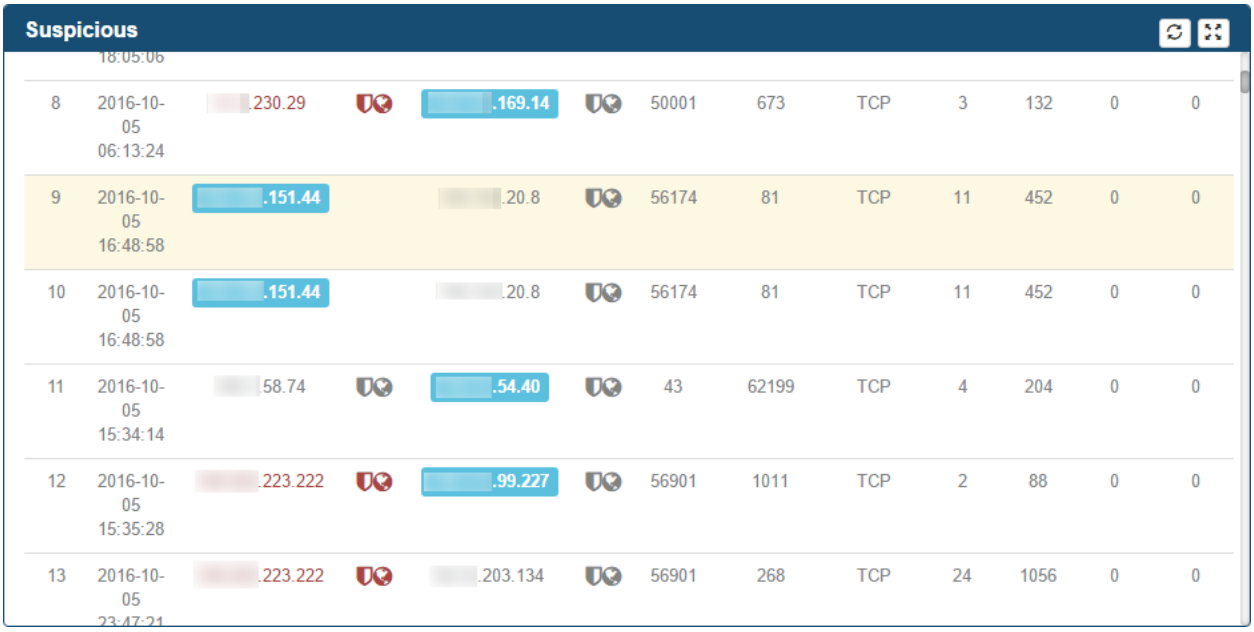
Το θεματικό μοντέλο (topic model) στον πυρήνα του Spot Machine Learning είναι ένα μη-επιβλεπόμενο μοντέλο μηχανικής μάθησης (unsupervised machine learning model). Παρ' όλα αυτά, το Spot δίνει τη δυνατότητα στον χρήστη να αξιολογήσει και να επηρεάσει το μοντέλο ως προς το τι είναι ύποπτο.

Το Spot παρέχει μια πληθώρα μεθόδων ανάλυσης ύποπτων κινήσεων οι οποίες εντοπίζουν ένα σύνολο ύποπτων ή απίθανων κινήσεων στο δίκτυο υπό επίβλεψη και τις αναφέρουν στον χρήστη για περαιτέρω εξέταση ώστε να κριθεί αν είναι ενδεικτικές κακόβουλης κίνησης ή ψευδώς θετικές αναφορές (false positives). Αυτή η διαδικασία αποτελεί μια μορφή ημι-επιβλεπόμενου εντοπισμού ανωμαλιών που χρησιμοποιεί θεματική μοντελοποίηση ώστε να συνάγει ποιες συμπεριφορές θεωρούνται φυσιολογικές για το δίκτυο υπό επίβλεψη και να κατασκευάσει ένα μοντέλο συμπεριφοράς για κάθε IP διεύθυνση. Η λειτουργία αυτή παρέχεται στα πλαίσια της ανάλυσης λειτουργίας που είναι η τρίτη και τελευταία δομική λειτουργία του Apache Spot. Για περισσότερες πληροφορίες για το πως χρησιμοποιείται το Topic Modeling στα πλαίσια του Apache Spot βλ. (Apache, 2016c).

2.4 Operational Analytics

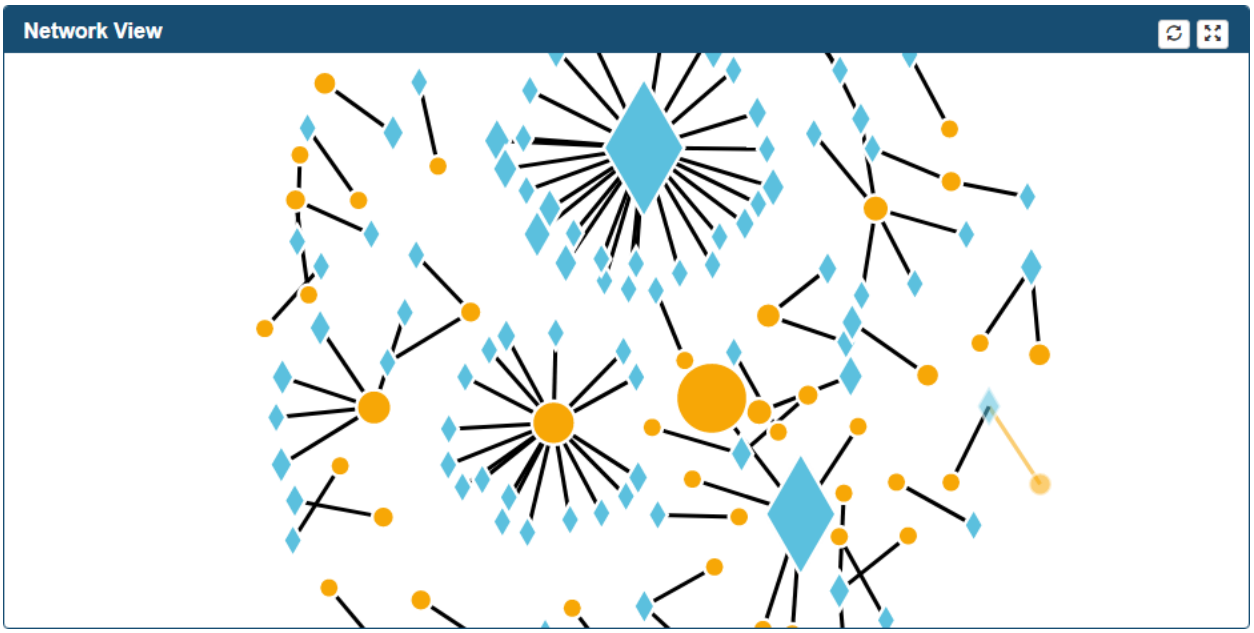
Το Apache Spot παρέχει γραφικό περιβάλλον όπου αναλύονται και παρουσιάζονται τα αποτελέσματα του αλγορίθμου μηχανικής μάθησης. Τα δεδομένα διαχωρίζονται σε NetFlow, DNS και Proxy και ανά ημέρα που ανακτήθηκαν. Ακολουθούν κάποια παραδείγματα για τις βασικές λειτουργίες που παρέχονται από το γραφικό περιβάλλον ανάλυσης του Apache Spot.

Η ανάλυση λειτουργιών μας παρέχει τη δυνατότητα να εξετάσουμε ύποπτη κίνηση δικτύου παρουσιάζοντας έναν κατάλογο με τις πιθανές απειλές που εντόπισε ο αλγόριθμος μηχανικής μάθησης. Η λίστα βρίσκεται στο παράθυρο Suspicious όπως παρουσιάζεται στην Εικόνα 4.



Εικόνα 4 – Κατάλογος Πιθανών Απειλών (https://goo.gl/9KW87T)

Στο παράθυρο Network View (βλ. Εικόνα 5) μπορούμε να δούμε ένα γράφημα του δικτύου υπό επίβλεψη και να κατανοήσουμε πως οι συσκευές του δικτύου αλληλεπιδρούν ώστε να εντοπίσουμε πιο εύκολα ύποπτες κινήσεις, μέσω οπτικοποίησης τους.

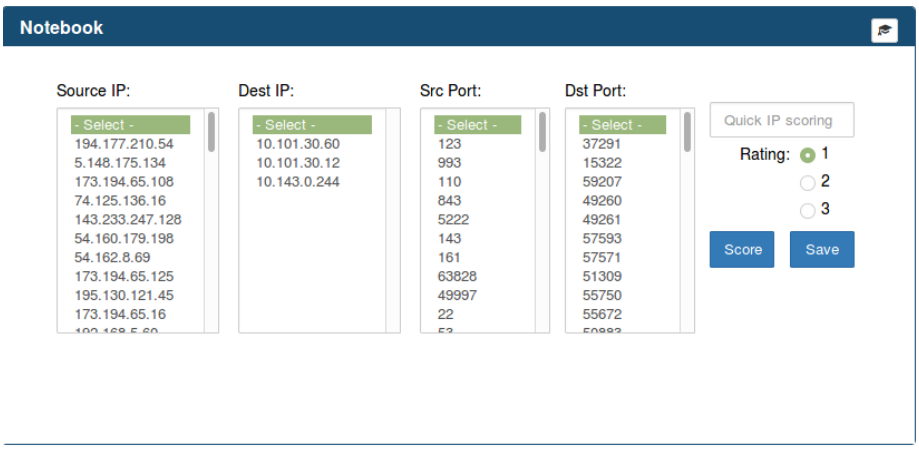


Εικόνα 5 – Γράφημα Δικτύου υπό Επίβλεψη (https://goo.gl/QSyxFN)

Το Apache Spot χρησιμοποιεί το Jupyter notebook, μία εφαρμογή διακομιστή-πελάτη (server-client application) που παρέχει δυνατότητα επεξεργασίας και εκτέλεσης εγγράφων notebook (notebook documents) μέσω ενός περιηγητή ιστού (web browser). Με το εργαλείο αυτό ουσιαστικά εισάγονται εκτελέσιμες προδιαγραφές για εφαρμογή υπηρεσιών φιλτραρίσματος και εξαίρεσης με αποτέλεσμα να παρέχεται μία πιο ξεκάθαρη εικόνα της διαδικασίας εντοπισμού ανωμαλιών στο δίκτυο, μειώνοντας έτσι τις ψευδώς θετικές αναφορές. Οι λειτουργίες που θα παρουσιάσουμε στη συνέχεια βασίζονται σε Jupyter notebooks.

2.4.1 Ημι-Επιβλεπόμενη Μάθηση

Στο παράθυρο Notebook παρέχεται στον χρήστη η δυνατότητα να βαθμολογήσει διευθύνσεις IP και θύρες (ports). Ο χρήστης μπορεί να επιλέξει έναν συνδυασμό ανάμεσα σε Source IP, Dest IP, Src Port και Dst Port, και στη συνέχεια να τον βαθμολογήσει επιλέγοντας ανάμεσα σε 1 (high risk), 2 και 3(low Risk), και πατώντας το κουμπί Score. Στη συνέχεια πατώντας το κουμπί Save οι βαθμολογίες σώζονται και οι αντίστοιχες κινήσεις απαλείφονται από το παράθυρο Suspicious Connects. Παράλληλα, οι βαθμολογίες γίνονται διαθέσιμες στον αλγόριθμο μηχανικής μάθησης για χρήση στις επόμενες εκτελέσεις του. Στην Εικόνα 6 παρουσιάζουμε το γραφικό περιβάλλον που περιγράφηκε παραπάνω.

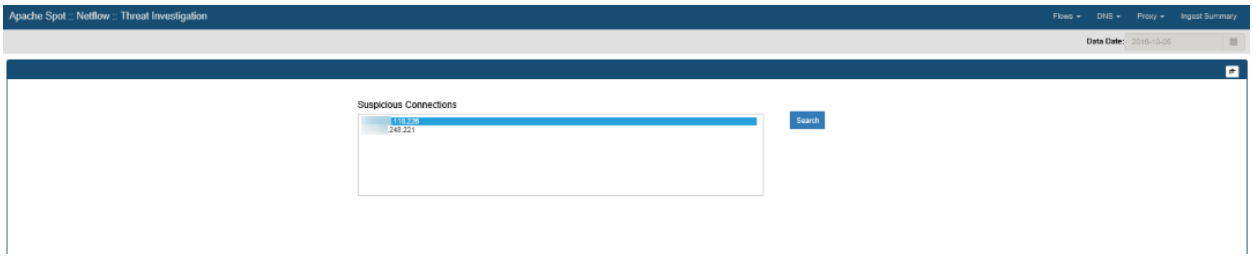


Εικόνα 6 – Παράθυρο Βαθμολόγησης Αποτελεσμάτων

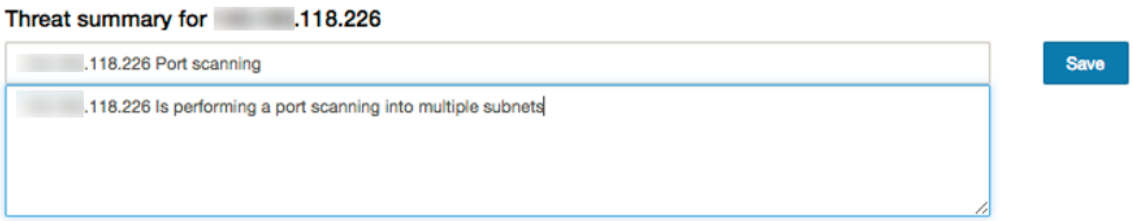
Η υλοποίηση αυτής της λειτουργίας βρίσκεται σε πειραματικό στάδιο. Συγκεκριμένα, οι βαθμολογίες του χρήστη σώζονται σε ένα αρχείο τύπου CSV, το οποίο παίρνει ως είσοδο ο αλγόριθμος μηχανικής μάθησης και όσες κινήσεις δικτύου είναι βαθμολογημένες με 3, δηλαδή θεωρούνται φυσιολογικές, προστίθενται (inject) στο αρχείο που πάρθηκε κατά τη παρούσα επίθεση ώστε να μεγαλώσει η ποσότητα της φυσιολογικής κίνησης. Επίσης, ο αλγόριθμος μηχανικής μάθησης έχει μία παράμετρο με το όνομα *dupfactor* η οποία καθορίζει πόσες φορές θα επισυναφθεί η κάθε κίνηση. Η προεπιλεγμένη τιμή είναι 1000.

2.4.2 Threat Investigation

Μετά την βαθμολόγηση των αποτελεσμάτων, μπορούμε να ανοίξουμε το Threat Investigation παράθυρο, όπου μας παρέχεται δυνατότητα να κάνουμε πιο εκτενή έρευνα πάνω στις διευθύνσεις που εντοπίσαμε σαν πιθανές απειλές (βλ. Εικόνα 7) και να αποθηκεύσουμε σχόλια για αυτές (βλ. Εικόνα 8).



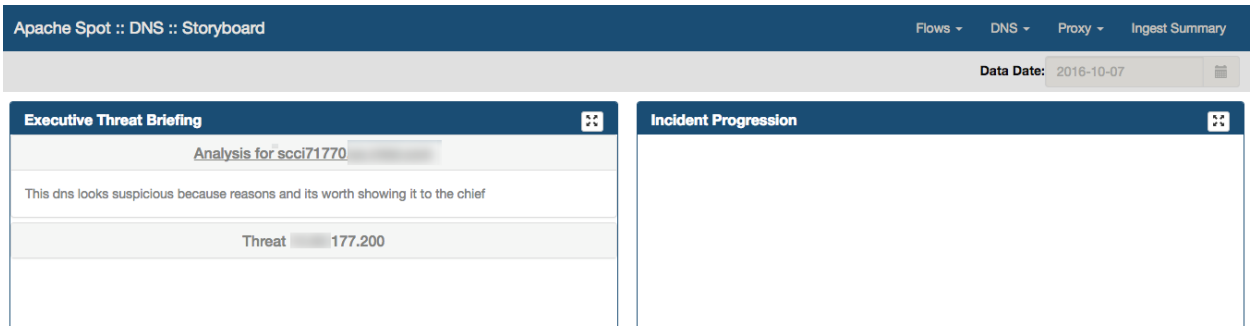
Εικόνα 7 – Παράθυρο Threat Investigation (http://spot.incubator.apache.org/doc/)



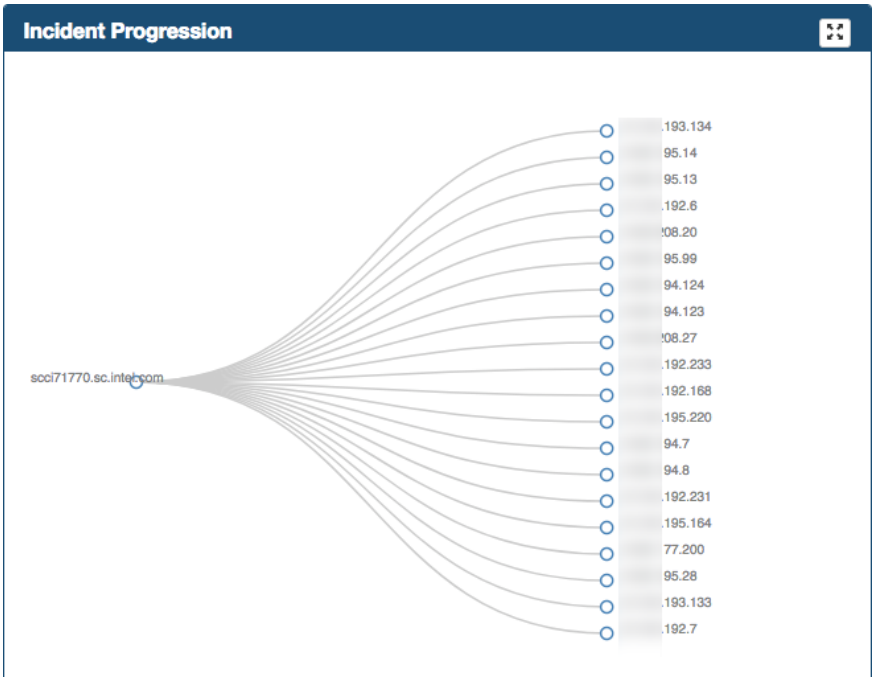
Εικόνα 8 – Σχολιασμός Απειλών (http://spot.incubator.apache.org/doc/)

2.4.3 Storyboard

Η επιλογή Storyboard παρέχει τη δυνατότητα να παρουσιάσουμε τα ευρήματα μας και σε τρίτους με τρόπο οργανωμένο και κατανοητό. Ένα παράδειγμα παρουσιάζεται στις εικόνες Εικόνα 9 και Εικόνα 10.



Εικόνα 9 – Παράθυρο Storyboard (http://spot.incubator.apache.org/doc/)



Εικόνα 10 – Παράθυρο Incident Progression (http://spot.incubator.apache.org/doc/)

(Apache, 2016d)

3 Παρουσίαση των Εργαλείων Δοκιμών Διείσδυσης

Η διαδικασία που ονομάζεται δοκιμή διείσδυσης (penetration testing), μπορεί να οριστεί σαν μια δοκιμαστική εισβολή σε ένα πληροφοριακό σύστημα για την αξιολόγηση της ασφάλειας του. Πρακτικά, περιλαμβάνει την προσομοίωση αληθινών επιθέσεων χρησιμοποιώντας τις ίδιες τεχνικές με αυτές που θα χρησιμοποιούσε ένας κακόβουλος χρήστης ώστε να εκτιμηθεί το ρίσκο που ενέχουν οι αδυναμίες που έχει ένα δίκτυο/σύστημα και να ληφθούν τα κατάλληλα μέτρα. Η διαδικασία δοκιμής διείσδυσης, μπορεί να διαχωριστεί σε φάσεις. Υπάρχουν πολλοί τρόποι διαχωρισμού στη βιβλιογραφία. Στο πλαίσιο της εργασίας αυτής υιοθετείται ο εξής διαχωρισμός:

- Αναγνώριση (Reconnaissance)
- Σάρωση (Scanning)
- Εκμετάλλευση Ευπαθειών (Exploitation)
- Διατήρηση Πρόσβασης (Maintaining Access)

Για την αξιολόγηση του Apache Spot, χρησιμοποιήθηκε μια σειρά από εργαλεία δοκιμών διείσδυσης, τα οποία εμπίπτουν στις φάσεις της Σάρωσης και της Εκμετάλλευσης Ευπαθειών. Στην ενότητα αυτή θα παρουσιάσουμε τα εργαλεία αυτά.

3.1 Σάρωση

3.1.1 Nmap

(https://home-assistant.io/images/supported_brands/nmap.png)

Το nmap -συντόμευση για network mapper- είναι ένα λογισμικό ανοιχτού κώδικα για εξερεύνηση δικτύων και έλεγχο ασφαλείας. Χρησιμοποιείται ευρέως από διαχειριστές δικτύων και δοκιμαστές διείσδυσης (penetration testers), αλλά και από κακόβουλους χρήστες. Η πιο συνηθισμένη χρήση του είναι η σάρωση θυρών (port scanning), αλλά διαθέτει πολύ περισσότερες δυνατότητες από αυτό.

Το nmap στέλνει ειδικά κατασκευασμένα πακέτα ώστε να βρει ποιες συσκευές είναι διαθέσιμες στο δίκτυο, τα ονόματα και τις εκδόσεις των υπηρεσιών που τρέχουν στις συσκευές, το λειτουργικό σύστημα τους, τι είδη μέτρων προστασίας προστατεύουν το δίκτυο (φιλτράρισμα IP/πακέτων, τείχη προστασίας κοκ.) και πολλά άλλα είδη πληροφοριών. Επιπρόσθετα, το nmap είναι ενισχυμένο με την NSE (Nmap Scripting Engine), μία συλλογή από scripts που παρέχουν δυνατότητες όπως σάρωση ευπαθειών (vulnerability scanning), εύρεση προεπιλεγμένων διαπιστευτηρίων (default credentials), εξελιγμένο εντοπισμό υπηρεσιών και πολλά άλλα. Όλα τα παραπάνω υποστηρίζονται από μία μεγάλη κοινότητα και ενημερώνονται συνεχώς.

Το nmap σχεδιάστηκε αρχικά μόνο για λειτουργικά συστήματα Linux, αλλά πλέον είναι διαθέσιμο για πάρα πολλά δημοφιλή λειτουργικά μεταξύ των οποίων τα Windows και Mac OS X.

(Lyon, Introduction, 2017) (Wikipedia, 2018f)

3.1.2 Nessus

(https://static.tenable.com/press/logos/products/Nessus-FullColor-RGB-logo.png)

Το Nessus είναι ένα εμπορικό λογισμικό της εταιρείας Tenable για σάρωση ευπαθειών (vulnerability scanning). Παρέχεται και στη δωρεάν έκδοση Nessus Home για μη-επαγγελματική χρήση. Βάσει σχετικών ερευνών, αποτελεί το πιο ευρέως χρησιμοποιούμενο εργαλείο της κατηγορίας του.

Το Nessus σαρώνει κάθε θύρα, βρίσκει τι υπηρεσία τρέχει, και στη συνέχεια ελέγχει την υπηρεσία για ευπάθειες που θα μπορούσε να εκμεταλλευτεί ένας κακόβουλος χρήστης. Μερικά από τα είδη ευπαθειών τα οποία ελέγχει είναι:

- Ευπάθειες που επιτρέπουν σε έναν κακόβουλο χρήστη να αποκτήσει έλεγχο ή πρόσβαση σε ευαίσθητα δεδομένα ενός συστήματος.
- Λανθασμένη ρύθμιση παραμέτρων συστήματος.
- Προεπιλεγμένους κωδικούς ή πρόσβαση χωρίς κωδικό.

Οι έλεγχοι ευπαθειών γίνονται με scripts γραμμένα σε NASL (Nessus Attack Scripting Language), που είναι μία γλώσσα βελτιστοποιημένη για παραμετροποιήσιμη αλληλεπίδραση με δίκτυα. Για την εκτέλεση πολλών από τις λειτουργίες του χρησιμοποιεί εξωτερικά εργαλεία, όπως το nmap που προαναφέραμε και το THC-Hydra που είναι ένα εργαλείο για «σπάσιμο» κωδικών (password cracking).

(Wikipedia, 2018g) (Wendlandt, 2004)

3.2 Εκμετάλλευση Ευπαθειών

3.2.1 Ncrack

(https://nmap.org/ncrack/images/ncrack_logo.png)

Το ncrack είναι ένα εργαλείο ανοιχτού κώδικα για «σπάσιμο» κωδικών μέσα σε ένα δίκτυο (network authentication cracking). Σχεδιάστηκε για εταιρείες και επαγγελματίες ασφαλείας ώστε να ελέγχουν τις συσκευές που βρίσκονται στο δίκτυο τους για αδύναμους κωδικούς.

Είναι βασισμένο σε μία παραμετροποιήσιμη αρχιτεκτονική και έτσι παρέχει πάρα πολλές δυνατότητες παραμετροποίησης ανάλογα με τα χαρακτηριστικά του δικτύου και ως προς την επέκταση του για υποστήριξη περαιτέρω πρωτοκόλλων. Ο τρόπος με τον οποίο λειτουργεί είναι πως εντοπίζοντας ποιες υπηρεσίες/πρωτόκολλα τρέχουν στον στόχο, προσπαθεί να βρει τα στοιχεία εισόδου σε αυτές μέσω

της τεχνικής brute force. Κάποια από τα πρωτόκολλα που υποστηρίζει μεταξύ πολλών άλλων είναι τα SSH, RDP, FTP, Telnet, HTTP(S), POP3(S) και IMAP(S).

Μία από τις δυνατότητες του ncrack είναι να λαμβάνει ως είσοδο ένα αρχείο XML που έχει παραχθεί από το nmap με τα αποτελέσματα μιας ανίχνευσης εκδόσεων υπηρεσιών (version scanning). Αυτός είναι ο τρόπος με τον οποίο χρησιμοποιήθηκε το ncrack στο πλαίσιο αυτής της εργασίας όπως θα περιγραφεί αργότερα.

Η έξοδος του ncrack είναι μια λίστα με τα στοιχεία εισόδου που κατάφερε να βρει, αν βρήκε κάποια, για κάθε στόχο της επίθεσης.

(Lyon, Ncrack, 2017) (Chantzis & Lyon, 2010)

3.2.2 t50

Το t50 είναι ένα εργαλείο έγχυσης πακέτων (packet injector), και χρησιμεύει στον έλεγχο της αντοχής ενός δικτύου υπό μεγάλη πίεση (stress testing). Αν και δεν σχεδιάστηκε γι' αυτό τον σκοπό αποτελεί ένα πολύ ισχυρό μέσο για επιθέσεις άρνησης εξυπηρέτησης (DoS attack), έτσι χρησιμοποιήθηκε και στο πλαίσιο αυτής της εργασίας.

Το t50 έχει τη δυνατότητα να στέλνει διαδοχικά 15 διαφορετικά πρωτόκολλα. Συγκεκριμένα καλύπτει συνήθη πρωτόκολλα όπως UDP, TCP και ICMP, πρωτόκολλα υποδομής (infrastructure protocols) όπως GRE, IPSec και RSVP, και πρωτόκολλα δρομολόγησης όπως RIP, EIGRP και OSFP. Επίσης μπορεί να στείλει πακέτα σε πάρα πολύ υψηλούς ρυθμούς, για παράδειγμα, σε ένα 1000BASE-T δίκτυο στέλνει 10^6 αιτήσεις SYN το δευτερόλεπτο δημιουργώντας μια SYN flood επίθεση.

Η ειδοποιός διαφορά του t50 σε σχέση με άλλα εργαλεία της κατηγορίας είναι ότι έχει τη δυνατότητα να στείλει όλα τα πρωτόκολλα, διαδοχικά, χρησιμοποιώντας μία μόνο υποδοχή δικτύου (network socket).

(Pissarra, 2017)

3.2.3 Slowloris

Το Slowloris είναι ένα εργαλείο για επιθέσεις άρνησης εξυπηρέτησης (DoS attacks) που επιτρέπει στον επιτιθέμενο να καταβάλει τον διακομιστή-στόχο εκκινώντας και διατηρώντας πάρα πολλές τμηματικές αιτήσεις HTTP, οι οποίες δεν ολοκληρώνονται ποτέ. Παράλληλα, αυτό το πετυχαίνει χρησιμοποιώντας ελάχιστο εύρος ζώνης και χωρίς παρενέργειες σε άσχετες υπηρεσίες και θύρες.

Αυτό που πετυχαίνει με τις τμηματικές αιτήσεις είναι πως δημιουργεί συνδέσεις με τον διακομιστή οι οποίες παραμένουν ανοιχτές καθώς ο εξυπηρετητής περιμένει την ολοκλήρωση των αιτήσεων. Έτσι, κάποια στιγμή, ο εξυπηρετητής φτάνει στο ανώτατο όριο συνδέσεων και αρνείται να εξυπηρετήσει περαιτέρω αιτήσεις.

Το γεγονός ότι το εργαλείο χρησιμοποιεί τμηματικά αλλά όχι παραμορφωμένα πακέτα, ελάχιστο εύρος ζώνης και δεν επηρεάζει άλλες υπηρεσίες οδηγεί στο να περνάει απαρατήρητη η επίθεση –ακόμα και

από συστήματα ανίχνευσης εισβολών (IDS)- μέχρι να είναι πολύ αργά. Τα παραπάνω καθιστούν το Slowloris ένα πολύ ενδιαφέρον εργαλείο για τον σκοπό αυτής της εργασίας.

(Wikipedia, 2018h) (Hansen, 2009) (Clouflare, -) (Incapsula, 2018)

3.2.4 BoNeSi

Το BoNeSi είναι ένα εργαλείο προσομοίωσης κατανεμημένης επίθεσης άρνησης εξυπηρέτησης (DDoS attack). Η έννοια της προσομοίωσης προκύπτει από το γεγονός πως παρέχει ρύθμιση για IP spoofing επιλέγοντας πόσες διαφορετικές IP να παραχθούν, προσομοιώνοντας έτσι ένα botnet.

Τα πακέτα που μπορεί να παραγάγει είναι τύπου ICMP, UDP και TCP (HTTP). Το χαρακτηριστικό του είναι πως όταν τρέχει σε ένα κλειστό δίκτυο όπου οι απαντήσεις του διακομιστή δρομολογούνται πίσω στο περιβάλλον που τον φιλοξενεί, μπορεί να κάνει TCP spoofing. Το TCP spoofing δουλεύει μόνο με την παραπάνω προϋπόθεση. Για τα πρωτόκολλα ICMP και UDP η επίθεση λειτουργεί και σε δημόσια δίκτυα.

(Goldstein, 2016)

3.2.5 Armitage

(https://www.hackingloops.com/wp-content/uploads/2017/01/armitage1.png)

Το Armitage είναι ένα εργαλείο διαχείρισης κυβερνο-επιθέσεων (cyber attacks) ανοιχτού κώδικα το οποίο παρέχει ένα γραφικό περιβάλλον για το εργαλείο Metasploit (Rapid7, 2018), οπτικοποιώντας τους στόχους και προτείνοντας προγράμματα εκμετάλλευσης ευπαθειών (exploits) και όχι μόνο. Το Metasploit είναι ένα εργαλείο ανοιχτού κώδικα που παρέχει μία συλλογή προγραμμάτων εκμετάλλευσης ευπαθειών ανοιχτού κώδικα που ενημερώνεται συνεχώς, παρέχοντας πολλές δυνατότητες για δοκιμές διείσδυσης.

Το Armitage εκτός από ένα γραφικό περιβάλλον για το Metasploit, παρέχει δυνατότητα διαμοιρασμού μιας συνεδρίας Metasploit μεταξύ πολλών ατόμων. Τα μέλη της ομάδας μπορούν να μοιράζονται δεδομένα που έχουν καταγραφεί, αρχεία που έχουν ανακτηθεί, τον έλεγχο των παραβιασμένων συσκευών στόχων και ένα κοινό αρχείο καταγραφής που δίνει τη δυνατότητα σε κάθε μέλος της ομάδας να ενημερώνεται για την κατάσταση της επίθεσης. Επίσης, παρέχει τη δυνατότητα εισόδου δεδομένων από άλλες πηγές όπως εργαλεία σάρωσης, χωρίς αυτό να σημαίνει πως εξαρτάται από αυτά, καθώς έχει τη δυνατότητα να σαρώσει μόνο του για ευπάθειες και να αξιοποιήσει τα αποτελέσματα. Παράλληλα, το Armitage καλύπτει και την φάση της Διατήρησης Πρόσβασης (Maintaining Access) παρέχοντας πολλά εργαλεία της κατηγορίας που αποτελούν μέρος του Meterpreter, το οποίο είναι επίσης κομμάτι του Metasploit.

Τέλος, παρέχει μία πολύ ισχυρή αλλά καθόλου εκλεπτυσμένη επίθεση που ονομάζεται Hail Mary. Πρόκειται μια αυτοματοποιημένη χρήση όλων των πιθανών προγραμμάτων εκμετάλλευσης ευπαθειών για τον στόχο, εκτελεσμένα σε βέλτιστη σειρά.

(Wikipedia, 2017b) (Mudge, 2016)

3.2.6 Iodine

Το Iodine είναι ένα εργαλείο που παρέχει τη δυνατότητα αποστολής δεδομένων IPV4 μέσω ενός εξυπηρετητή DNS, δημιουργώντας έτσι μία επίθεση DNS tunneling. Ένα παράδειγμα της χρησιμότητας του είναι δίκτυα που έχουν περιορισμένη πρόσβαση στο διαδίκτυο μέσω τείχους προστασίας, αλλά στα οποία επιτρέπονται τα αιτήματα DNS. Δύο χαρακτηριστικά που το ξεχωρίζουν από άλλα εργαλεία της κατηγορίας είναι:

- Είναι σχεδιασμένο για να τρέχει σε πολλά συστήματα UNIX αλλά και Win32. Τα «τούνελ» μπορούν να δημιουργηθούν ανεξάρτητα του λειτουργικού συστήματος εκάστου των δύο άκρων.
- Παρέχει εύκολη εγκατάσταση και ρύθμιση, καθώς αναλαμβάνει πολλά ζητήματα αυτόματα για βέλτιστη απόδοση. Επίσης, υποστηρίζει μέχρι και 16 χρήστες παράλληλα σε κάθε «τούνελ».

(Ekman & Andersson, 2014)

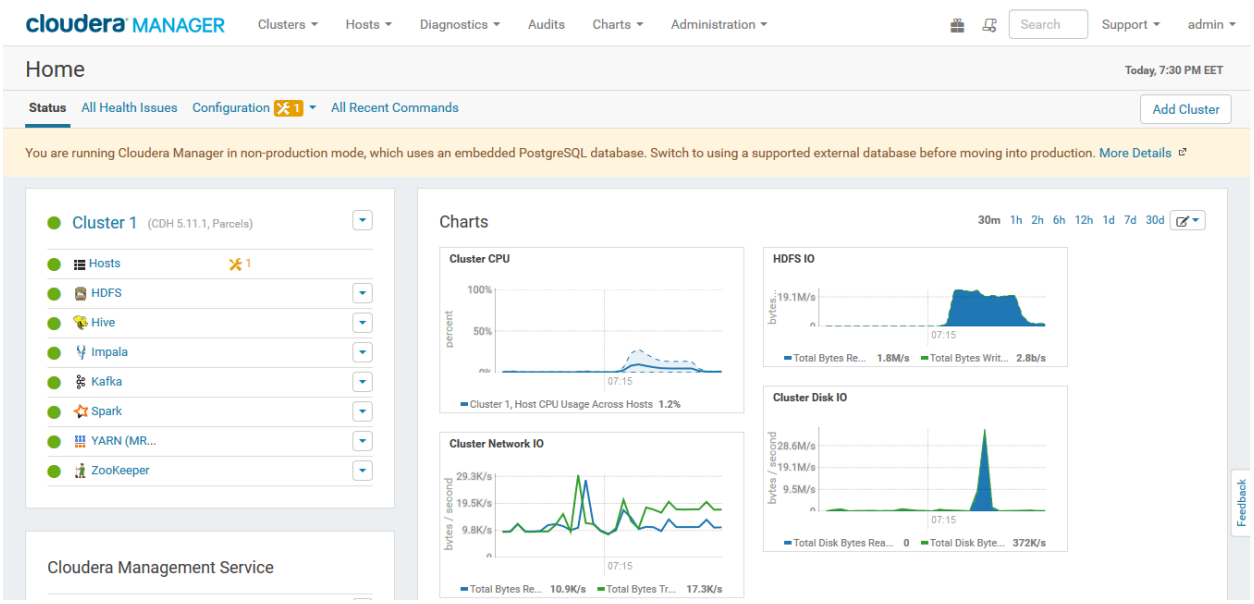
4 Διάταξη Συστήματος και Δικτύου

Στην ενότητα αυτή θα παρουσιάσουμε τη διάταξη που χρησιμοποιήθηκε για το Apache Spot αλλά και για το δίκτυο υπό επίβλεψη. Όλα τα μηχανήματα που χρησιμοποιήθηκαν είναι εικονικά και εγκατεστημένα σε έναν server στον οποίο τρέχει το VMware ESXi 6.0 που είναι ένα πρόγραμμα για χρήση και διαχείριση εικονικών υπολογιστών (virtual machines).

Για το Apache Spot χρησιμοποιήθηκαν τρεις εικονικοί εξυπηρετητές(virtual machines - VMs) με λειτουργικό σύστημα Ubuntu 14.04.4 Server, όπου σε κάθε έναν έχει ανατεθεί μία από τις τρεις δομικές λειτουργίες του Apache Spot. Παρακάτω παρουσιάζονται με τα ονόματα τους όπως θα τους αναφέρουμε στη συνέχεια της εργασίας και ποιά είναι η δομική λειτουργία που τους αντιστοιχεί.

- cloudera-host-2
  - Λήψη Δεδομένων
- cloudera-host-1
  - Μηχανική Μάθηση
- cloudera-manager
  - Ανάλυση Λειτουργίας

Όλες οι υπηρεσίες που χρειάζεται το Apache Spot για τη λειτουργία του, όπως τα Apache Hadoop, Apache Hive, Apache Kafka και Apache Spark, στα οποία έχουμε αναφερθεί νωρίτερα, έχουν εγκατασταθεί και τις διαχειριζόμαστε μέσω του CDH (Cloudera Distribution for Hadoop). Η πλατφόρμα Cloudera Manager που παρέχει το CDH μας δίνει τη δυνατότητα να διαχειριστούμε , παρακολουθήσουμε και ρυθμίσουμε όλες τις υπηρεσίες μέσα από το ίδιο περιβάλλον, όπως φαίνεται στην Εικόνα 11.



Εικόνα 11 – Περιβάλλον Cloudera Manager

Στη συνέχεια θα παρουσιάσουμε τη διάταξη που χρησιμοποιήθηκε για την αναπαραγωγή και παρακολούθηση της κίνησης στο δίκτυο αλλά και για τις επιθέσεις. Επειδή η διάταξη αλλάζει ανάλογα με το είδος τηλεμετρίας, δηλαδή ανάμεσα σε flow, dns και proxy, θα διαχωρίσουμε τη παρουσίαση μας βάσει των τριών αυτών ειδών.

4.1 Flow

Για την τηλεμετρία που αφορά δεδομένα NetFlow χρησιμοποιήθηκαν δύο εικονικοί εξυπηρετητές Ubuntu 16.04 Server και ένας εικονικός υπολογιστής Kali Linux 2017.2. Παρακάτω βλέπουμε τα ονόματα των εικονικών μηχανών όπως θα αναφέρονται στο υπόλοιπο της εργασίας και τον ρόλο τους.

- Server
  - Είναι ο εξυπηρετητής υπό επίβλεψη. Δέχεται κίνηση και τη προωθεί στον cloudera-host-2.
- Client
  - Αναπαράγει αρχεία με καταγεγραμμένη κίνηση δικτύου στο δίκτυο, με προορισμό τον Server.
- Kali
  - Εκτελεί τις επιθέσεις στον Server.

4.1.1 Αναπαραγωγή κίνησης δικτύου

Η κίνηση δικτύου που χρησιμοποιήθηκε καταγράφηκε από το δίκτυο του Ινστιτούτου Πληροφορικής και Τηλεπικοινωνιών του ΕΚΕΦΕ Δημόκριτος και αποθηκεύτηκε σε αρχεία pcap. Τα αρχεία τροποποιήθηκαν με το εργαλείο tcprewrite[8] ώστε η διεύθυνση προορισμού και η MAC διεύθυνση προορισμού να είναι αυτές του Server. Για την αναπαραγωγή τους χρησιμοποιήθηκε το εργαλείο tcpreplay[9]. Τα αρχεία pcap έχουν μέγεθος 477MB το καθένα και εμείς αναπαράγαμε δύο από αυτά από δύο φορές το καθένα, σύνολο ~1.9GB.

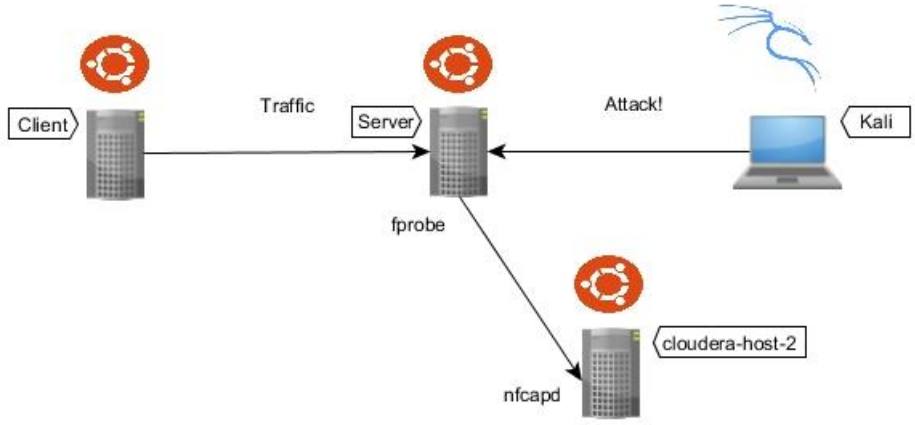
Ο Server προωθεί την κίνηση από τη διεπαφή δικτύου του (network interface), με χρήση του εργαλείου fprobe[10], στον cloudera-host-2, δηλαδή στον εξυπηρετητή υπεύθυνο για το κομμάτι λήψης δεδομένων του Spot. Αυτός με τη σειρά του καταγράφει την κίνηση που λαμβάνει με το εργαλείο nfcapd.

4.1.2 Εκτέλεση Επιθέσεων

Οι επιθέσεις γίνονται μέσω του Kali Linux σε πραγματικό χρόνο (κατά τη διάρκεια αναπαραγωγής κίνησης από τον Client) με στόχο τον Server. Έτσι προωθούνται και αυτές στο cloudera-host-2 για καταγραφή με το nfcapd.

Στην Εικόνα 12 φαίνεται μια γραφική αναπαράσταση των παραπάνω.

[8] Παρέχει τη δυνατότητα επεξεργασίας αρχείων τύπου pcap.
[9] Παρέχει τη δυνατότητα αναπαραγωγής αρχείων τύπου pcap.
[10] Συλλέγει δεδομένα κίνησης δικτύου και τα προωθεί σαν δεδομένα NetFlow στον καθορισμένο παραλήπτη.



Εικόνα 12 – Διάταξη Δικτύου – Flow

4.2 DNS

Για την τηλεμετρία που αφορά τα δεδομένα DNS χρησιμοποιήθηκαν οι Server, Client, cloudera-host-2 και ένας εικονικός εξυπηρετητής Ubuntu 16.04 Server:

- Server
  - Φορτώνει γνωστές ιστοσελίδες και καταγράφει την κίνηση σε αρχεία pcap που στη συνέχεια μεταφέρονται στον cloudera-host-2 για λήψη δεδομένων (ingestion).
- Client
  - Ίδιος ρόλος με τον Server.
- cloudera-host-2
  - Φορτώνει γνωστές ιστοσελίδες, στέλνει κίνηση με χρήση του εργαλείου iperf[11] μέσω ενός DNS tunnel που έχει εγκαθιδρύσει με τον DNS Server και αυτά καταγράφονται σε αρχεία pcap.
- DNS Server
  - Λαμβάνει τα δεδομένα που στέλνει η cloudera-host-2 με το iperf στην διεπαφή δικτύου(network interface) του DNS tunnel.

4.2.1 Δημιουργία Κίνησης DNS

Οι Server, Client και cloudera-host-2 χρησιμοποιούν το εργαλείο PhantomJS[12], το οποίο μεταξύ άλλων παρέχει ένα script που φορτώνει γνωστές ιστοσελίδες. Εκτελώντας αυτό το script δημιουργείται κίνηση και καταγράφεται σε αρχεία pcap με χρήση του εργαλείου tshark ώστε να δοθούν αργότερα για λήψη (ingestion).

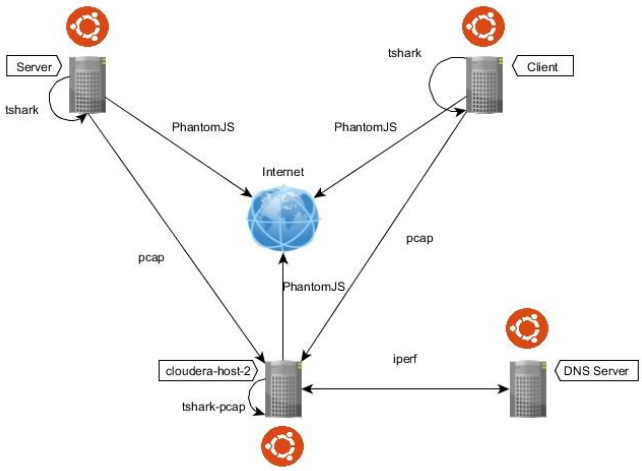
[11] Είναι ένα εργαλείο για την εκτέλεση μετρήσεων διεκπεραιωτικότητας (throughput) δικτύου.

[12] Είναι ένας περιηγητής χωρίς γραφικό περιβάλλον (headless browser) που χρησιμοποιείται για αυτόματη αλληλεπίδραση με μία ιστοσελίδα.

4.2.2 Εκτέλεση Επιθέσεων

Οι επιθέσεις γίνονται με τον cloudera-host-2 και τον DNS Server σε πραγματικό χρόνο. Είναι σημαντικό να σημειωθεί πως παρ' όλο που μπαίνει στη θέση του επιτιθέμενου σε αυτή τη περίπτωση, ο cloudera-host-2 εξακολουθεί να εκτελεί το κομμάτι λήψης δεδομένων του Apache Spot.

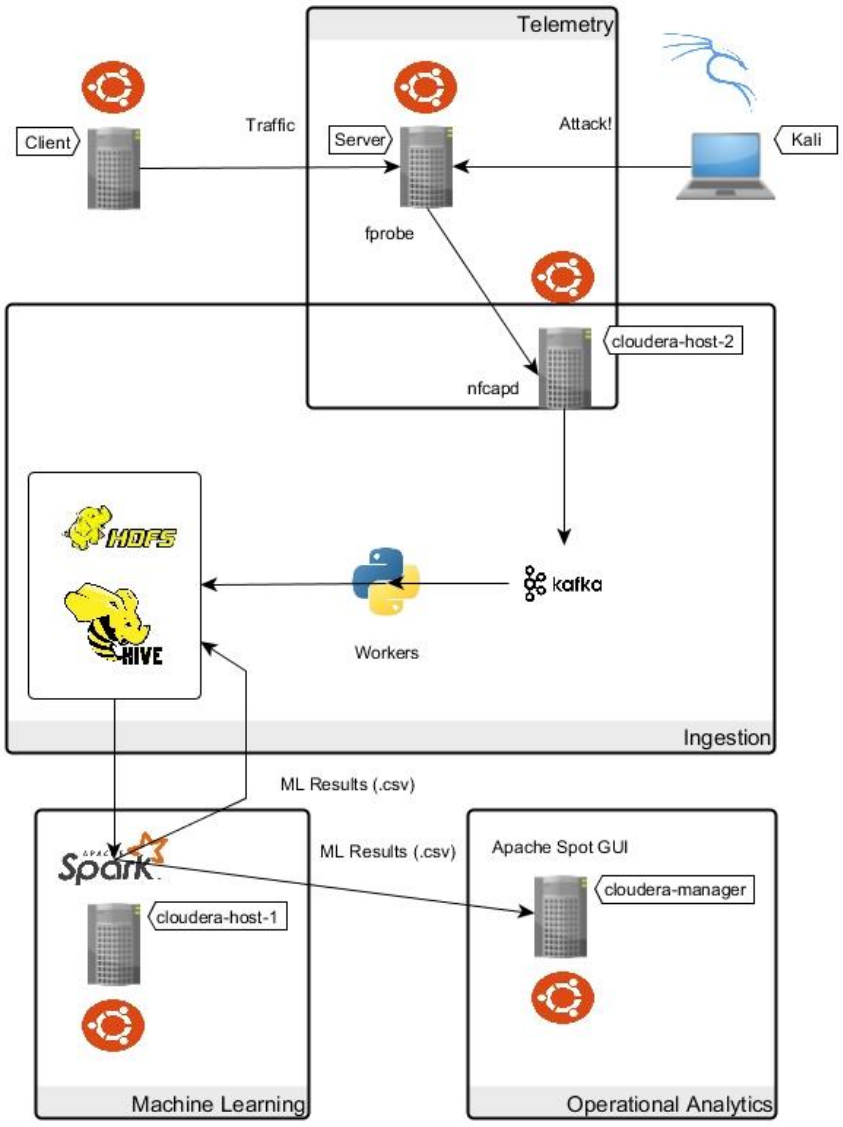
Στην Εικόνα 13 φαίνεται μια γραφική αναπαράσταση των παραπάνω.



Εικόνα 13 – Διάταξη Δικτύου DNS

4.3 Λειτουργική Διάταξη

Σε αυτή την ενότητα παρουσιάζουμε μία απεικόνιση της συνολικής λειτουργικής διάταξης του Apache Spot όπως χρησιμοποιήθηκε στο πλαίσιο της παρούσας πτυχιακής εργασίας, σε ένα συνοπτικό διάγραμμα (βλ. Εικόνα 14). Ως παράδειγμα διάταξης του κομματιού της τηλεμετρίας πάρθηκε αυτή που παρουσιάσαμε στην ενότητα 4.1 και προστέθηκε η υπόλοιπη απεικόνιση της διαδικασίας μέχρι και την ανάλυση λειτουργιών.



Εικόνα 14 – Λειτουργική Διάταξη

4.4 Αυτοματοποίηση – Bash Scripts

Για την πιο αποδοτική εκτέλεση της διαδικασίας γράψαμε δύο bash scripts, ένα για τη διαδικασία που αφορά το Flow κομμάτι και ένα για το DNS. Επιπρόσθετα παρουσιάζονται τα επιμέρους bash scripts που χρησιμοποιούνται από τα δύο βασικά. Ο λόγος που γράφτηκαν ως ξεχωριστά αρχεία είναι πως είχαν εφαρμογή και σε άλλα σενάρια χρήσης.

4.4.1 Flow Bash Script

```
#!/bin/bash

cd spot-ingest
python master_collector_s.py -t flow -w 4 &>master.out &
sleep 2s
topic=$(<topic_out)
python worker.py -t flow -i 0 --topic $topic &>workers.out &
python worker.py -t flow -i 1 --topic $topic &>workers.out &
python worker.py -t flow -i 2 --topic $topic &>workers.out &
python worker.py -t flow -i 3 --topic $topic &>workers.out &
count=0

ssh root@10.101.30.60 &>/dev/null << EOF
fprobe -i ens160 10.101.30.12:9995 &
EOF

cd ..
while [ ! $count -eq 1 ]
do

count=$((count+1))

nfcapd -D -T all -p 9995 -w -t 120 -l /home/spotuser/

if [ -n "$1" ]
then
ssh root@10.101.30.61 &>/dev/null << EOF
cd /home/localadmin/pcap/
bash traffic_temp.sh &
pid=$(ps -ef | grep traffic.sh | grep -v grep | awk '{print $2}')
ssh root@10.101.10.160 &>/dev/null 'bash $1'
wait $pid
EOF

else
read -p "Press Enter to initiate traffic"
ssh root@10.101.30.61 &>/dev/null << EOF
cd /home/localadmin/pcap/
bash traffic.sh &>/dev/null
wait
EOF

fi

read -p "Traffic done?"
kill $(pidof nfcapd)
read -p "Stop the attack"
cp nfcapd.2018* traffic/flow/
sleep 12s
bash ingestest.sh &>/dev/null
read -p "Ingestion done?"

ssh spotuser@10.101.30.11 &>/dev/null << EOF
```

```
cd spot-ml
./ml_ops.sh $(date +%Y%m%d) flow 1 &>ml.out
wait

cd csv/
mkdir $count
cd ..
hdfs dfs -get /user/spotuser/flow/scored_results/$(date
+%Y%m%d)/scores/flow_results.csv csv/$count
wait

hdfs dfs -rm -r -f /user/spotuser/flow/binary/$(date +%Y%m%d)
hdfs dfs -rm -r -f /user/spotuser/flow/hive/y=$(date +%Y)/m=$(date
+%m)/d=$(date +%d)
#hdfs dfs -rm -r -f /user/spotuser/flow/scored_results/$(date +%Y%m%d)

EOF

rm nfcapd.2018*
cd spot-ingest
>master.out
>workers.out
cd ..

done

ssh root@10.101.30.60 &>/dev/null << 'EOF'
kill $(pidof fprobe)
EOF

cd spot-ingest
bash kill.sh &>/dev/null
```

4.4.2 DNS Bash Script

```
#!/bin/bash

cd spot-ingest
python master_collector_s.py -t dns -w 3 &>master.out &
sleep 2s
topic=$(<topic_out)
python worker.py -t dns -i 0 --topic $topic &>workers.out &
python worker.py -t dns -i 1 --topic $topic &>workers.out &
python worker.py -t dns -i 2 --topic $topic &>workers.out &
count=10
cd ..

while [ ! $count -eq 11 ]
do

count=$((count+1))

tshark -i eth0 -w /home/spotuser/dns_traffic_12.pcap &>/dev/null &
cd phantomjs-2.1.1-linux-x86_64/bin/
sh generate10.sh &>/dev/null &
echo -e "tshark and generate are up on 12\n"

ssh localadmin@10.101.30.60 &>/dev/null << EOF
tshark -i ens160 -w /home/localadmin/dns_traffic_60.pcap &>/dev/null &
cd phantomjs-2.1.1-linux-x86_64/bin/
sh generate10.sh &>/dev/null &
EOF
echo -e "tshark and generate are up on 60\n"

ssh localadmin@10.101.30.61 &>/dev/null << EOF
tshark -i ens160 -w /home/localadmin/dns_traffic_61.pcap &>/dev/null &
cd phantomjs-2.1.1-linux-x86_64/bin/
sh generate10.sh &>/dev/null &
EOF
echo -e "tshark and generate are up on 61\n"

ssh root@10.101.10.35 << EOF
cd /home/localadmin/iodine/bin
./iodined -f -P secretpass 192.168.99.1 t1.iliketunnels.com &>/dev/null &
EOF

ssh -T root@10.101.10.35 << EOF
cd /home/localadmin/iodine/bin
iperf -s -B 192.168.99.1 < /dev/null > /tmp/iperf_combined_1.log 2>&1 &
EOF
echo -e "Started iodined and iperf on dns server\n"

cd /home/spotuser/iodine/bin/
echo -e "Starting iodine on 12\n"
./iodine -r 10.101.10.35 -P secretpass -f t1.iliketunnels.com &>/dev/null &
echo -e "Using iperf through the tunnel\n"
iperf -c 192.168.99.1 &>/dev/null
sleep 300s
```

```
ps -ef | grep generate | grep -v grep | awk '{print $2}' | xargs kill
pid_1=$(ps -ef | grep iodine | grep -v grep | awk '{print $2}')
kill $(pidof tshark) $pid_1
echo -e "Killed tshark, generate.sh, iodine and iperf on 12\n"

ssh root@10.101.10.35 &>/dev/null << 'EOF'
pid_1=$(ps -ef | grep iodined | grep -v grep | awk '{print $2}')
kill $pid_1
kill -9 $(pidof iperf)
EOF
echo -e "Killed iodined and iperf on dns\n"

echo "Copying pcaps from 60 and 61 to 12"
scp localadmin@10.101.30.60:/home/localadmin/dns_traffic_60.pcap
/home/spotuser/
scp localadmin@10.101.30.61:/home/localadmin/dns_traffic_61.pcap
/home/spotuser/

ssh localadmin@10.101.30.60 &>/dev/null << 'EOF'
ps -ef | grep generate | grep -v grep | awk '{print $2}' | xargs kill
kill $(pidof tshark)
rm dns_traffic_60.pcap
EOF
echo -e "Killed tshark and generate.sh on 60\n"

ssh localadmin@10.101.30.61 &>/dev/null << 'EOF'
ps -ef | grep generate | grep -v grep | awk '{print $2}' | xargs kill
kill $(pidof tshark)
rm dns_traffic_61.pcap
EOF
echo -e "Killed tshark and generate.sh on 61\n"

cd /home/spotuser/
cp dns_traffic_60.pcap traffic/dns/
cp dns_traffic_61.pcap traffic/dns/
cp dns_traffic_12.pcap traffic/dns/
echo -e "Copied pcaps for ingestion\n"


bash ingestest.sh &>/dev/null
read -p "Ingestion done?"

ssh spotuser@10.101.30.11 &>/dev/null << EOF

cd spot-ml
echo -e "Starting spot-ml\n"
./ml_ops.sh $(date +%Y%m%d) dns 1 &>ml.out
wait

cd csv/
mkdir $count
cd ..
hdfs dfs -get /user/spotuser/dns/scored_results/$(date
+%Y%m%d)/scores/dns_results.csv csv/$count
wait
```

```
hdfs dfs -rm -r -f /user/spotuser/dns/binary/$(date +%Y%m%d)
hdfs dfs -rm -r -f /user/spotuser/dns/hive/y=$(date +%Y)/m=$(date
+%m)/d=$(date +%d)
hdfs dfs -rm -r -f /user/spotuser/dns/scored_results/$(date +%Y%m%d)

EOF

rm dns_traffic_12.pcap
rm dns_traffic_60.pcap
rm dns_traffic_61.pcap

done

cd spot-ingest
>master.out
>workers.out

bash kill.sh &>/dev/null
```

4.4.3 Επιμέρους Bash Scripts

4.4.3.1 Έλεγχος Ολοκλήρωσης Λήψης Δεδομένων

```
#!/bin/bash

cd spot-ingest
wait='workers.out MODIFY'

while [ "$wait" != "" ]
do
wait=$(inotifywait -e modify -t 90 workers.out)
done
```

4.4.3.2 Τερματισμός Συλλεκτών (Collectors) και Εργατών (Workers)

```
#!/bin/bash
ps -ef | grep worker.py | grep -v grep | grep -v proxy | awk '{print $2}' |
xargs kill
ps -ef | grep master | grep -v grep | grep -v proxy | awk '{print $2}' |
xargs kill
```

5 Παρουσίαση Επιθέσεων και Αποτελεσμάτων

Στην ενότητα αυτή παρουσιάζουμε τα αποτελέσματα που πήραμε από το Apache Spot για κάθε επίθεση που εκτελέσαμε, αφού πρώτα παρουσιάσουμε τη μεθοδολογία επεξεργασίας των αποτελεσμάτων του Apache Spot. Θα διαχωρίσουμε τα αποτελέσματα ανά εργαλείο δοκιμών διείσδυσης που χρησιμοποιήθηκε. Η μεθοδολογία που ακολουθήσαμε περιλαμβάνει την εκτέλεση κάθε επίθεσης δέκα φορές, ώστε να πάρουμε πιο πλήρης στατιστικές μετρήσεις των αποτελεσμάτων που παρήγαγε το Apache Spot. Επίσης, θα δείξουμε τα αποτελέσματα που πήραμε εξετάζοντας περαιτέρω κάποιες από αυτές τις επιθέσεις αλλάζοντας κάποια δεδομένα όπως, τη ποσότητα της παραγόμενης κίνησης δικτύου και κάποιες από τις παραμέτρους του αλγορίθμου μηχανικής μάθησης. Τέλος, θα παρουσιάσουμε τα αποτελέσματα που πήραμε χρησιμοποιώντας την λειτουργία ημι-επιβλεπόμενης μάθησης που παρέχει το Apache Spot.

5.1 Επεξεργασία Αποτελεσμάτων

Στο παρασκήνιο της δομικής λειτουργίας ανάλυσης λειτουργίας και συγκεκριμένα της παρουσίασης των αποτελεσμάτων (βλ. Εικόνα 4), βρίσκεται ένα αρχείο τύπου CSV, το οποίο παράγεται από την εκτέλεση του αλγορίθμου μηχανικής μάθησης και αποθηκεύεται στο HDFS. Το αρχείο αυτό περιλαμβάνει την κατάταξη της κίνησης του δικτύου ταξινομημένη ως προς την επικινδυνότητα της, κατά φθίνουσα σειρά, παρέχοντας παράλληλα την αντίστοιχη πιθανότητα να είναι φυσιολογική κίνηση όπως υπολογίστηκε από τον αλγόριθμο μηχανικής μάθησης. Επίσης, κάθε καταχώρηση περιέχει στοιχεία ημερομηνίας και ώρας, διευθύνσεις και θύρες πηγής και προορισμού και πρωτόκολλο δικτύου. Στην Εικόνα 15 μπορούμε να δούμε ένα παράδειγμα ενός τέτοιου αρχείου.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	12/6/2017 14:10	2017	12	6	14	10	50	0.002	91.140.89.132	10.101.30.60	49310	443	TCP	10	640	0	0	1.45E-05		
2	12/6/2017 14:05	2017	12	6	14	5	10	0	192.168.1.98	10.101.30.60	889	889	UDP	3	4344	0	0	1.50E-05		
3	12/6/2017 14:06	2017	12	6	14	6	45	0	10.101.30.60	192.168.4.70	443	36665	TCP	6	240	0	0	1.74E-05		
4	12/6/2017 14:13	2017	12	6	14	13	25	8.619	192.168.100.2	10.101.30.60	0	768	ICMP	4	224	0	0	1.74E-05		
5	12/6/2017 14:12	2017	12	6	14	12	40	51.866	192.168.2.155	10.101.30.60	161	55212	UDP	6	662	0	0	1.76E-05		
6	12/6/2017 14:06	2017	12	6	14	6	20	0.017	10.101.10.160	10.101.30.60	34755	22	TCP	2	400	0	0	1.77E-05		
7	12/6/2017 14:15	2017	12	6	14	15	0	59.101	192.168.1.1	10.101.30.60	0	0	IGMP	4	144	0	0	1.78E-05		
8	12/6/2017 14:14	2017	12	6	14	14	15	0	10.0.100.57	10.101.30.60	22	40555	TCP	1	52	0	0	1.78E-05		
9	12/6/2017 14:11	2017	12	6	14	11	0	0	192.168.2.78	10.101.30.60	138	138	UDP	1	229	0	0	1.78E-05		
10	12/6/2017 14:10	2017	12	6	14	10	45	31.524	192.168.4.88	10.101.30.60	0	0	IGMP	2	56	0	0	1.78E-05		
11	12/6/2017 14:06	2017	12	6	14	6	15	0.019	192.168.4.239	10.101.30.60	55012	161	UDP	2	236	0	0	1.78E-05		
12	12/6/2017 14:05	2017	12	6	14	5	15	2.656	143.233.226.2	10.101.30.60	465	50439	TCP	4	388	0	0	1.78E-05		
13	12/6/2017 14:06	2017	12	6	14	6	45	0	10.101.10.160	10.101.30.60	33505	21	TCP	2	104	0	0	1.78E-05		
14	12/6/2017 14:05	2017	12	6	14	5	50	0	143.233.226.132	10.101.30.60	52554	69	UDP	6	360	0	0	0.000017819		
15	12/6/2017 14:05	2017	12	6	14	5	10	0	10.101.10.160	10.101.30.60	33397	26	TCP	1	44	0	0	1.78E-05		
16	12/6/2017 14:12	2017	12	6	14	12	35	0	10.101.30.60	192.168.3.79	0	768	ICMP	1	421	0	0	1.78E-05		
17	12/6/2017 14:07	2017	12	6	14	7	40	0.002	143.233.226.2	10.101.30.60	465	58892	TCP	2	202	0	0	1.78E-05		
18	12/6/2017 14:08	2017	12	6	14	8	15	8.009	192.168.100.2	10.101.30.60	0	768	ICMP	3	168	0	0	0.000017826		
19	12/6/2017 14:05	2017	12	6	14	5	10	0	10.101.10.160	10.101.30.60	33397	465	TCP	1	44	0	0	1.78E-05		
20	12/6/2017 14:06	2017	12	6	14	6	35	0	143.233.226.91	10.101.30.60	21	52435	TCP	1	60	0	0	1.78E-05		
21	12/6/2017 14:10	2017	12	6	14	10	10	0.005	195.134.65.131	10.101.30.60	110	28720	TCP	4	962	0	0	1.78E-05		
22	12/6/2017 14:06	2017	12	6	14	6	55	51.467	185.29.25.185	10.101.30.60	143	51525	TCP	6	606	0	0	1.78E-05		
23	12/6/2017 14:11	2017	12	6	14	11	15	0	143.233.226.2	10.101.30.60	993	57565	TCP	1	434	0	0	1.78E-05		
24	12/6/2017 14:12	2017	12	6	14	12	50	0	143.233.226.91	10.101.30.60	22	40638	TCP	1	60	0	0	1.78E-05		
25	12/6/2017 14:10	2017	12	6	14	10	15	0	143.233.226.132	10.101.30.60	0	2048	ICMP	3	144	0	0	0.000017831		
26	12/6/2017 14:12	2017	12	6	14	12	40	51.868	192.168.4.239	10.101.30.60	55212	161	UDP	6	606	0	0	1.78E-05		
27	12/6/2017 14:06	2017	12	6	14	6	55	0	143.233.226.2	10.101.30.60	995	52167	TCP	1	130	0	0	1.78E-05		
28	12/6/2017 14:05	2017	12	6	14	5	10	0	10.101.30.60	10.101.10.160	179	33397	TCP	1	40	0	0	1.78E-05		
29	12/6/2017 14:05	2017	12	6	14	5	15	0	10.101.10.160	10.101.30.60	33397	37	TCP	1	44	0	0	0.000017833		
30	12/6/2017 14:05	2017	12	6	14	5	45	0.008	150.140.141.143	10.101.30.60	143	56042	TCP	2	2581	0	0	1.78E-05		
31	12/6/2017 14:05	2017	12	6	14	5	10	0	10.101.30.60	10.101.10.160	37	33397	TCP	1	40	0	0	1.78E-05		
32	12/6/2017 14:05	2017	12	6	14	5	10	0	143.233.226.132	10.101.30.60	0	2048	ICMP	1	48	0	0	1.78E-05		
33	12/6/2017 14:08	2017	12	6	14	8	20	0	143.233.226.2	10.101.30.60	25	49480	TCP	1	40	0	0	1.78E-05		
34	12/6/2017 14:15	2017	12	6	14	15	0	0.011	143.233.226.2	10.101.30.60	587	52773	TCP	2	202	0	0	0.000017837		
35	12/6/2017 14:09	2017	12	6	14	9	30	99.171	143.233.226.67	10.101.30.60	445	49264	TCP	17	1480	0	0	1.78E-05		
36	12/6/2017 14:12	2017	12	6	14	12	40	8.059	143.233.226.132	10.101.30.60	52565	69	UDP	18	1080	0	0	1.78E-05		
37	12/6/2017 14:06	2017	12	6	14	6	10	0	143.233.227.144	10.101.30.60	138	138	UDP	3	687	0	0	1.78E-05		
38	12/6/2017 14:10	2017	12	6	14	10	40	0.869	192.168.2.118	10.101.30.60	49997	161	UDP	8	889	0	0	1.78E-05		

Εικόνα 15 – Αρχείο .csv με Αποτελέσματα

Στο πλαίσιο της εργασίας αυτής, χρησιμοποιήσαμε τα αρχεία CSV που δημιουργήθηκαν από κάθε επίθεση ώστε να εξάγουμε αποτελέσματα που θα μας βοηθήσουν να αξιολογήσουμε την αποτελεσματικότητα του Apache Spot ενάντια σε κάθε είδος επίθεσης. Συγκεκριμένα, για κάθε αρχείο CSV, βρήκαμε με χρήση του MS Excel τις εξής τιμές:

- Την κατάταξη της επίθεσης, δηλαδή την θέση της πρώτης καταχώρησης που αφορά την επίθεση. Στα δεδομένα αναφέρεται ως “Position”. Είναι επιθυμητό η κατάταξη να είναι όσο το δυνατόν υψηλότερη.
- Την πιθανότητα που ανατέθηκε στην πρώτη καταχώρηση που αφορά την επίθεση. Στα δεδομένα αναφέρεται ως “Probability”. Είναι επιθυμητό η πιθανότητα να είναι όσο το δυνατόν μικρότερη, υποδεικνύοντας έτσι ότι πρόκειται όντως για επίθεση.
- Το πλήθος των καταχωρήσεων που αφορούν την επίθεση στις πρώτες εκατό καταχωρήσεις του αρχείου. Στα δεδομένα αναφέρεται ως “First 100”. Είναι επιθυμητό το πλήθος αυτό να είναι όσο το δυνατόν πιο κοντά στο 100.
- Το ποσοστό των καταχωρήσεων που αφορούν την επίθεση επί του συνόλου των καταχωρήσεων. Στα δεδομένα αναφέρεται ως “Occurrence %”. Πρακτικά η μετρική αυτή υποδεικνύει το ποσοστό της κίνησης που δημιουργεί η επίθεση έναντι του συνολικού όγκου κίνησης.

Έχοντας τις παραπάνω τιμές για κάθε αρχείο CSV, πάλι με χρήση του MS Excel, δημιουργήσαμε box plots (Wikipedia, 2018i) για κάθε εργαλείο και κάθε μία από τις παραπάνω τιμές. Επιλέξαμε τα box plots ως μία από τις μορφές αναπαράστασης των αποτελεσμάτων μας, ώστε να μπορέσουμε να πάρουμε μια εικόνα της κατανομής των τιμών για κάθε εργαλείο ανάμεσα στις δέκα επιθέσεις.

Περαιτέρω, για κάθε αρχείο CSV, σχεδιάσαμε τη καμπύλη ROC (Receiver Operating Characteristic) και υπολογίσαμε την τιμή AUROC (Area Under ROC) που της αντιστοιχεί. Αυτό το κάναμε χρησιμοποιώντας ένα πρόγραμμα γραμμένο με τη γλώσσα προγραμματισμού Java το οποίο παίρνει ως είσοδο το αρχείο CSV που αναφέραμε προηγουμένως και παράγει ένα άλλο αρχείο CSV που περιέχει μόνο τις καταχωρήσεις που αφορούν τις επιθέσεις. Αυτό το αρχείο δίνεται ως είσοδος σε ένα πρόγραμμα γραμμένο με τη γλώσσα προγραμματισμού R το οποίο παράγει μία εικόνα με την ROC καμπύλη και ένα αρχείο κειμένου με την τιμή AUROC που της αντιστοιχεί. Η καμπύλη ROC και η τιμή AUROC χρησιμεύουν στην αξιολόγηση μοντέλων με δυαδικά αποτελέσματα: στην περίπτωση μας τα αποτελέσματα είναι δυαδικά καθώς μπορούν να διαχωριστούν σε true positives και false positives. Η τιμή AUROC αναπαριστά την πιθανότητα που έχει το μοντέλο να κατατάξει ένα τυχαία επιλεγμένο θετικό αποτέλεσμα σε υψηλότερη θέση από ένα τυχαία επιλεγμένο αρνητικό αποτέλεσμα. Στη περίπτωση του Apache Spot δηλαδή, αναπαριστά τη πιθανότητα να κατατάξει μία κίνηση δικτύου που αποτελεί μέρος επίθεσης ψηλότερα από μία που δεν αποτελεί μέρος επίθεσης.

(Grigorev, 2015)

5.2 Παρουσίαση Αποτελεσμάτων

Για κάθε εργαλείο δοκιμών διείσδυσης, θα εξηγήσουμε πως το χρησιμοποιήσαμε και θα παρουσιάσουμε τα αποτελέσματα που πήραμε από το Apache Spot. Συγκεκριμένα, για κάθε εργαλείο,

θα παρουσιάσουμε τα box plots που σχεδιάσαμε, τις καμπύλες ROC και τις τιμές AUROC που πήραμε μαζί τον μέσο όρο και την απόκλιση που τους αντιστοιχεί.

5.2.1 nmap

Το nmap χρησιμοποιήθηκε για την εκτέλεση μίας απλής σάρωσης θυρών (port scanning). Συγκεκριμένα οι εντολές που εκτελέστηκαν είναι:

```
nmap -Pn -n -sS 10.101.30.60

nmap -Pn -n -sT 10.101.30.60

nmap -Pn -n -sA 10.101.30.60

nmap -Pn -n -sV 10.101.30.60
```

Η παράμετρος -Pn ορίζει πως το nmap πρέπει να θεωρήσει τον στόχο διαθέσιμο και να μην κάνει ping.

Η παράμετρος -n ορίζει πως το nmap δεν θα κάνει επίλυση DNS (DNS resolution).

Το όρισμα 10.101.30.60 είναι η IP του Server.

Η παράμετρος –sS ορίζει πως το nmap θα εκτελέσει μία σάρωση τύπου SYN. Η σάρωση SYN είναι μία πολύ διαδεδομένη τεχνική η οποία είναι γρήγορη και μας επιτρέπει να καταλάβουμε αν μία θύρα είναι ανοιχτή, κλειστή ή φιλτραρισμένη(filtered). Αυτό το πετυχαίνει στέλνοντας ένα TCP πακέτο με το SYN flag σημειωμένο, έτσι, αν η πόρτα είναι ανοιχτή, ο στόχος απαντάει με SYN/ACK, και με τη σειρά του το nmap απαντάει με RST (reset) αντί για ACK καθώς δεν θέλει να ολοκληρωθεί η σύνδεση. Αν η απάντηση του στόχου αντί για SYN/ACK είναι RST, το nmap καταλαβαίνει ότι η θύρα είναι κλειστή. Τέλος, αν δεν λάβει καμία απάντηση ή κάποιο μήνυμα σφάλματος ICMP, μετά από δύο προσπάθειες, το nmap καταλαβαίνει ότι η θύρα είναι φιλτραρισμένη.

Η παράμετρος -sT ορίζει πως το nmap θα εκτελέσει μία σάρωση τύπου TCP Connect. Η σάρωση TCP Connect έχει κάποια παρόμοια χαρακτηριστικά με την SYN, αλλά δεν προτιμάται καθώς εντοπίζεται εύκολα, και έτσι εφαρμόζεται μόνο όταν δεν υπάρχει δυνατότητα για SYN scan, για παράδειγμα όταν ο χρήστης δεν έχει αρκετά δικαιώματα συστήματος. Η διαφορά είναι ότι δημιουργεί μια πλήρη σύνδεση, καθώς απαντάει με ACK αντί για RST όταν λάβει το SYN/ACK, εγκαθιδρύοντας έτσι την σύνδεση με τη θύρα. Στις περιπτώσεις που οι θύρα είναι κλειστή ή φιλτραρισμένη ισχύουν τα ίδια με τη σάρωση SYN.

Η παράμετρος -sA ορίζει πως το nmap θα εκτελέσει μία σάρωση τύπου ACK. Η σάρωση ACK στέλνει ένα TCP πακέτο με το ACK flag σημειωμένο. Σε αντίθεση με τις δύο προηγούμενες τεχνικές σάρωσης, δεν βρίσκουμε ποιες θύρες είναι ανοιχτές, μπορεί να διαχωρίσει μόνο μεταξύ μη-φιλτραρισμένη (ανοιχτή ή κλειστή) και φιλτραρισμένη. Αυτή η τεχνική σάρωσης χρησιμοποιείται για την αναγνώριση του τρόπου λειτουργίας των τειχών προστασίας (firewalls) που μπορεί να προστατεύουν το δίκτυο και για τον εντοπισμό φιλτραρισμένων θυρών.

Η παράμετρος -sV ορίζει πως το nmap θα εκτελέσει μία σάρωση τύπου Version. Ο σκοπός αυτής της σάρωσης είναι η αναγνώριση των υπηρεσιών που είναι προσβάσιμες στις θύρες, καθώς και των εκδόσεων τους. Από τη στιγμή που δεν έχουμε ορίσει τι πακέτα να στείλει στις θύρες, γίνεται μία

σάρωση τύπου SYN. Τα αποτελέσματα της σάρωσης χρησιμοποιούνται για την αναγνώριση των υπηρεσιών. Η διαδικασία αναγνώρισης υπηρεσιών είναι αρκετά μακροσκελής και δεν θα αναλυθεί περαιτέρω.

(Lyon, Nmap Network Scanning, 2008)

Αυτή η επίθεση εκτελέστηκε χρησιμοποιώντας τη διάταξη δικτύου όπως περιγράφεται στην ενότητα 4.1.

Στον Πίνακας 1 παρουσιάζουμε τις τιμές που πήραμε από τα αρχεία CSV από τις δέκα επαναλήψεις της επίθεσης όπως περιγράφηκε παραπάνω. Είναι ταξινομημένα σε χρονολογική σειρά, με τη πρώτη γραμμή να αποτελεί τη πρώτη φορά που εκτελέσαμε την επίθεση, η δεύτερη γραμμή τη δεύτερη φορά κοκ. Οι στήλες είναι οι τιμές που παίρνουμε από κάθε αρχείο CSV όπως περιγράφηκαν στη υποενότητα 5.1. Στο υπόλοιπο της ενότητας 5.2 οι αντίστοιχοι πίνακες θα ακολουθούν αυτή τη δομή.

Position	Probability	First 100	Occurrence %
4	3.41E-05	22	1.513346879
8	2.42E-05	27	1.589647903
1	3.43E-05	22	3.608247423
3	2.59E-05	39	2.371775867
1	2.26E-05	29	2.175379427
1	2.22E-05	67	5.581171463
2	1.64E-05	38	1.745356903
13	2.87E-05	34	2.41490669
3	3.58E-05	22	2.271722086
7	2.58E-05	23	1.990945571

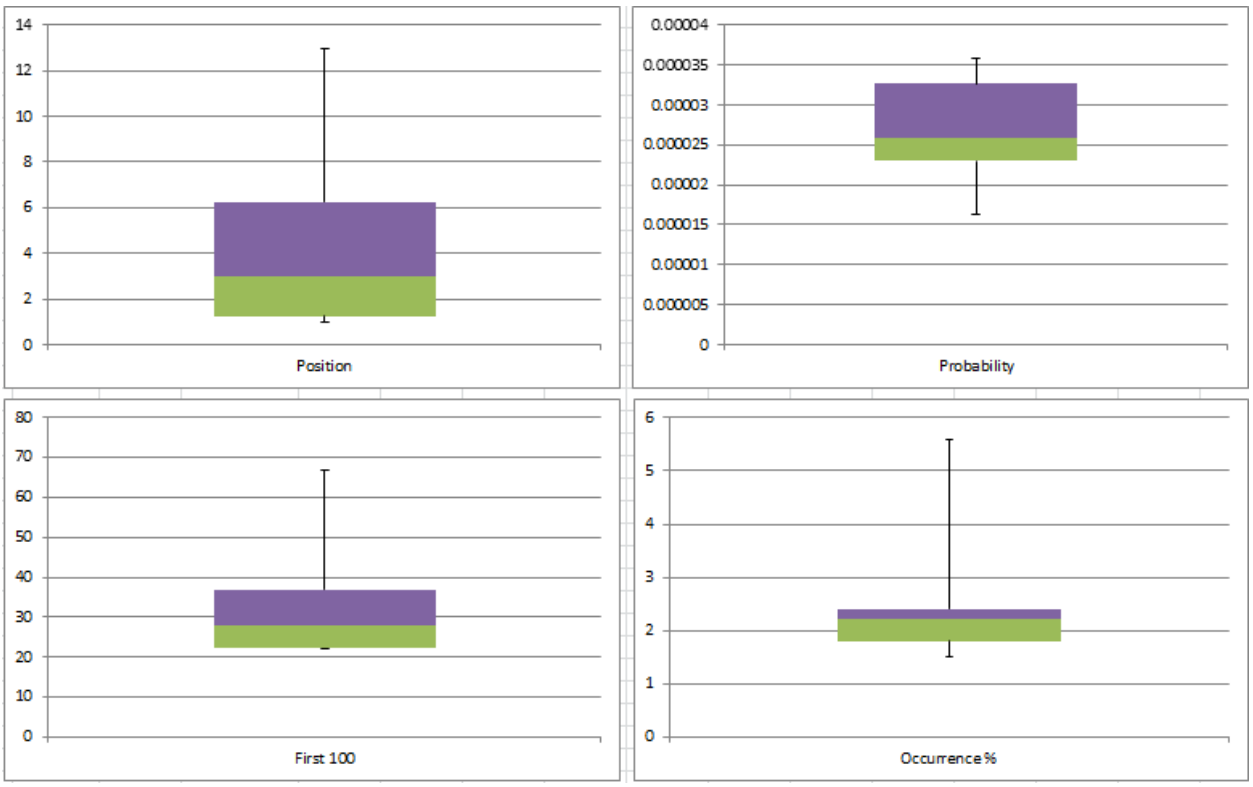
Πίνακας 1 – Αποτελέσματα nmap

Στον Πίνακας 2 βλέπουμε τις τιμές που χρειαζόμαστε για τον σχεδιασμό των box plots της Εικόνα 16.

	Position	Probability	First 100	Occurrence %
Minimum	1	1.64E-05	22	1.513346879
1st Quartile	1.25	2.3E-05	22.25	1.80675407
Median	3	2.59E-05	28	2.223550756
3rd Quartile	6.25	3.27E-05	37	2.404123985
Maximum	13	3.58E-05	67	5.581171463

Πίνακας 2 – Δεδομένα για τα Box Plots του nmap

5.2.1.1 Box Plots



Εικόνα 16 – nmap Box Plots

Η διάμεσος (median) της κατάταξης (Position) είναι η τιμή 3 και το κεντρικό 50% των τιμών ορίζεται από το 1.25 έως το 6.25. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 6.25 ενώ η μέγιστη τιμή είναι το 13.

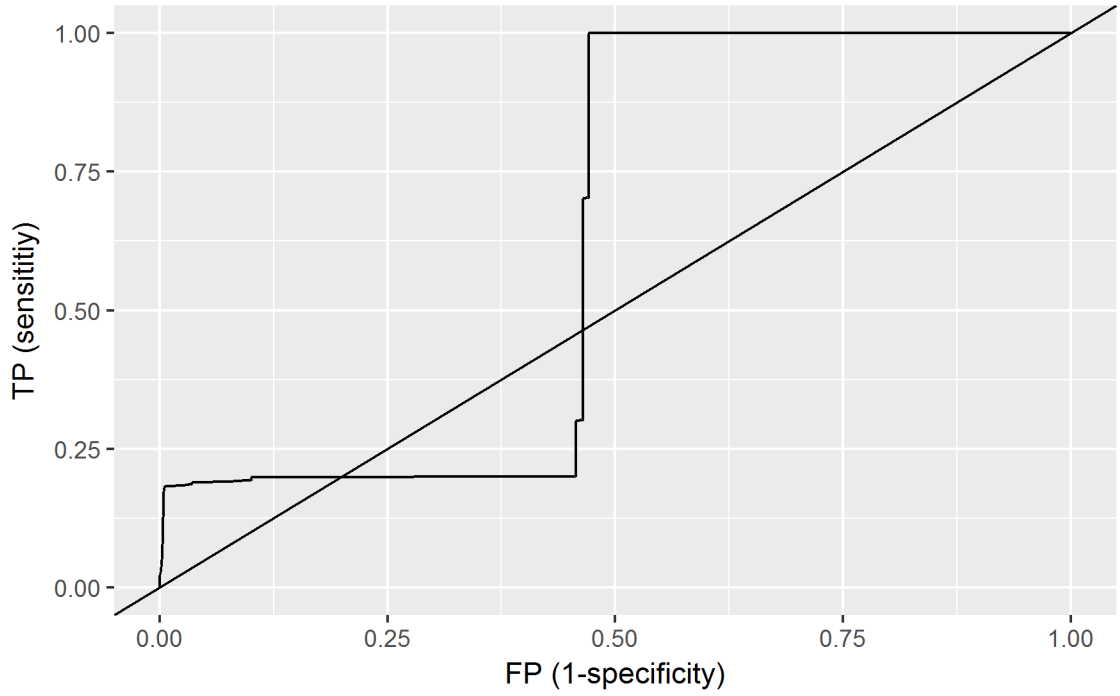
Στο γράφημα των πιθανοτήτων η διάμεσος είναι το 2.59E-05 και το κεντρικό 50% των τιμών ορίζεται από το 2.3E-05 έως το 3.27E-05. Η κατανομή είναι ασύμμετρη αφού το άνω 75% των τιμών είναι συγκεντρωμένο από το 2.3E-05 και πάνω, ενώ η ελάχιστη τιμή είναι το 1.64E-05.

Στο γράφημα First 100 η διάμεσος είναι το 28 και το κεντρικό 50% των τιμών ορίζεται από το 22.25 έως το 37. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 37 ενώ η μέγιστη τιμή είναι το 67.

Στο γράφημα Occurrence % η διάμεσος είναι το ~2.22 και το κεντρικό 50% των τιμών ορίζεται από το ~1.8 έως το ~2.4. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι μέχρι το ~2.4 ενώ η μέγιστη τιμή είναι το ~5.58.

5.2.1.2 ROC-AUROC

Στην Εικόνα 17 παρουσιάζουμε μία από τις δέκα καμπύλες ROC που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Παρατηρούμε πως ενώ η αρχή της καμπύλης είναι καλή που σημαίνει πως είναι υψηλά καταταγμένη η επίθεση, η συνέχεια δεν είναι ανάλογη και έτσι συνολικά ο διαχωρισμός δεν είναι καλός. Στον Πίνακας 3 φαίνονται οι δέκα τιμές AUROC που πήραμε.



Εικόνα 17 – ROC nmap

	Δοκιμή #1	Δοκιμή #2	Δοκιμή #3	Δοκιμή #4	Δοκιμή #5	Δοκιμή #6	Δοκιμή #7	Δοκιμή #8	Δοκιμή #9	Δοκιμή #10
AUROC	0.62615	0.4602	0.38775	0.50772	0.53571	0.44557	0.52156	0.435	0.56905	0.60473
Μέση Τιμή: 0.509343					Απόκλιση: 0.005994					

Πίνακας 3 – AUROC nmap

Η μέση τιμή της τιμής AUROC που παίρνουμε για τις δέκα επιθέσεις με το εργαλείο nmap είναι 0.509343 και η απόκλιση 0.005994.

5.2.2 Nessus

Το Nessus χρησιμοποιήθηκε για την εκτέλεση ενός Network Scan, το οποίο είναι μια λειτουργία για εύρεση ευπαθειών πολλών ειδών στα συστήματα που είναι συνδεδεμένα σε ένα δίκτυο. Δώσαμε ως στόχο τον Server. Σε αυτή την επίθεση, το Nessus κάνει μία σάρωση στις θύρες που συνήθως τρέχουν γνωστές υπηρεσίες ώστε να συλλέξει πληροφορίες για το σύστημα στόχο, αντίστοιχα όπως και το nmap που περιγράψαμε προηγουμένως. Η εύρεση ευπαθειών γίνεται με χρήση πρόσθετων scripts τα οποία είναι διαθέσιμα μέσω της Nessus Plugin Database. Εκτελεί τα scripts που ταιριάζουν στο προφίλ που κατασκευάστηκε από τη σάρωση και είναι πιθανό να εντοπίσουν αδυναμίες στο σύστημα στόχο. Μερικές από τις κατηγορίες των ευπαθειών που εξετάζουν τα scripts είναι:

- Windows
- Backdoors
- Firewalls
- FTP
- Remote File Access
- SMTP
- DoS

Στον Πίνακας 4 παρουσιάζουμε τις τιμές που πήραμε από τα αρχεία CSV από τις δέκα επαναλήψεις της επίθεσης αυτής όπως περιγράφηκε παραπάνω.

Position	Probability	First 100	Occurrence %
1	2.07E-05	31	7.52785449
15	2.13E-05	27	6.104079364
1	1.97E-05	35	6.843956858
1	1.69E-05	66	5.955370943
2	2.48E-05	37	6.672744309
1	1.68E-05	68	6.852530046
3	2.41E-05	30	6.350715644
1	2.52E-05	54	3.698355632
2	2.76E-05	28	7.654278895
1	1.28E-05	33	5.860771179

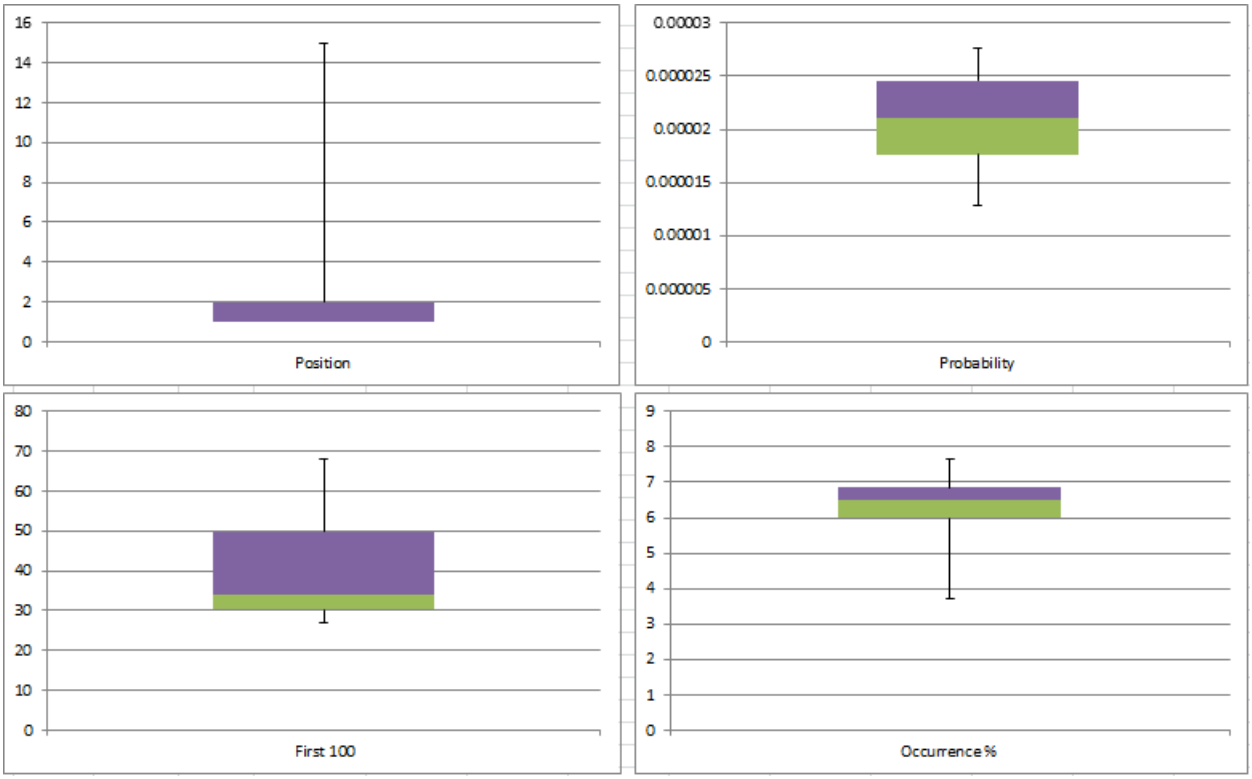
Πίνακας 4 – Αποτελέσματα Nessus

Στον Πίνακας 5 βλέπουμε τις τιμές που χρειαζόμαστε για τον σχεδιασμό των box plots της Εικόνα 18.

	Position	Probability	First 100	Occurrence %
Minimum	1	1.28E-05	27	3.698355632
1st Quartile	1	1.76E-05	30.25	5.992548048
Median	1	2.1E-05	34	6.511729976
3rd Quartile	2	2.46E-05	49.75	6.850386749
Maximum	15	2.76E-05	68	7.654278895

Πίνακας 5 – Δεδομένα για τα Box Plots του Nessus

5.2.2.1 Box Plots



Εικόνα 18 – Nessus Box Plots

Η διάμεσος (median) της κατάταξης (Position) είναι η τιμή 1 και το κεντρικό 50% των τιμών ορίζεται από το 1 έως το 2. Παρατηρούμε στον Πίνακας 4 πως το κάτω 60% των τιμών είναι 1, και αυτό εξηγεί γιατί δεν εμφανίζεται το κάτω 50% των τιμών στο box plot. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 2 ενώ η μέγιστη τιμή είναι το 15.

Στο γράφημα των πιθανοτήτων η διάμεσος είναι το 2.1E-05 και το κεντρικό 50% των τιμών ορίζεται από το 1.76E-05 έως το 2.46E-05. Η κατανομή είναι σχετικά συμμετρική, αφού η απόκλιση του κεντρικού 50% των τιμών είναι ελάχιστα προς τα πάνω.

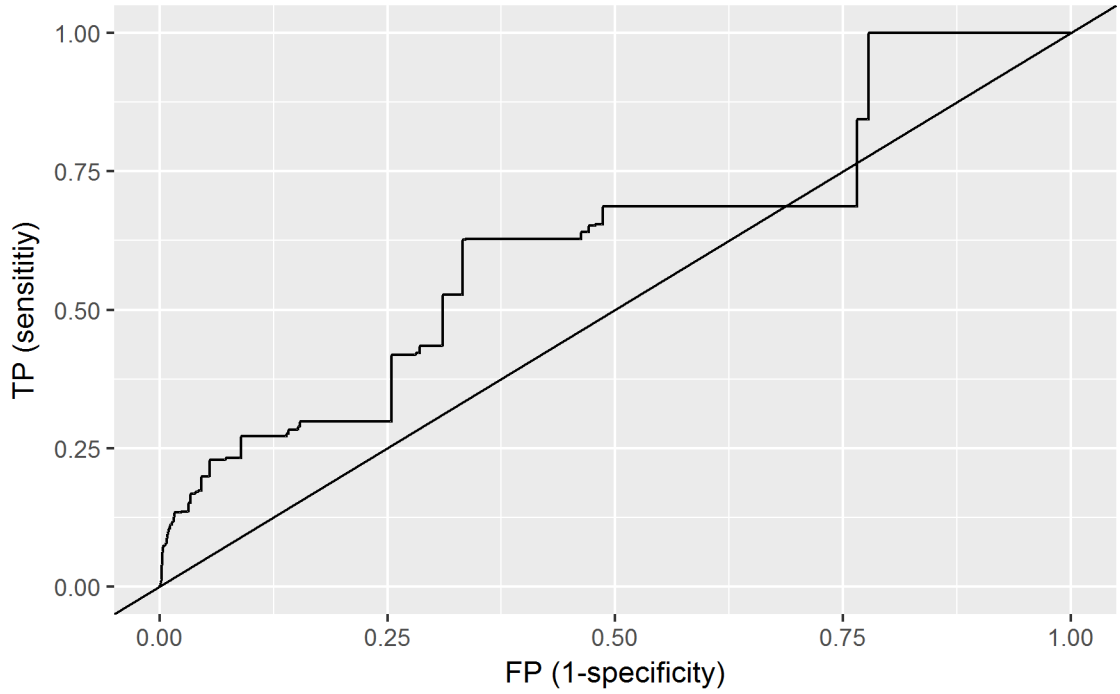
Στο γράφημα First 100 η διάμεσος είναι το 34 και το κεντρικό 50% των τιμών ορίζεται από το 30.25 έως το 49.75. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 49.75 ενώ η μέγιστη τιμή είναι το 68.

Στο γράφημα Occurrence % η διάμεσος είναι το ~6.51 και το κεντρικό 50% των τιμών ορίζεται από το ~5.99 έως το ~6.85. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το άνω 75% των τιμών είναι πάνω από το ~5.99 ενώ η ελάχιστη τιμή είναι το ~3.7.

5.2.2.2 ROC – AUROC

Στην Εικόνα 19 παρουσιάζουμε ενδεικτικά την μία καμπύλη ROC από τις δέκα που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Παρατηρούμε πως δεν απέχει πολύ από την ευθεία που αναπαριστά την

τυχαιότητα, το οποίο σημαίνει πως ο διαχωρισμός του μοντέλου σε αυτή τη περίπτωση δεν είναι καλός. Στον Πίνακας 6 φαίνονται οι δέκα τιμές AUROC που πήραμε.



Εικόνα 19 – ROC Nessus

	Δοκιμή #1	Δοκιμή #2	Δοκιμή #3	Δοκιμή #4	Δοκιμή #5	Δοκιμή #6	Δοκιμή #7	Δοκιμή #8	Δοκιμή #9	Δοκιμή #10
AUROC	0.60812	0.62061	0.60407	0.63386	0.64114	0.64889	0.62023	0.42797	0.59145	0.64213
Μέση Τιμή: 0.603848					Απόκλιση: 0.00416					

Πίνακας 6 – AUROC Nessus

Η μέση τιμή της τιμής AUROC που παίρνουμε για τις 10 επιθέσεις με το εργαλείο Nessus είναι 0.603848 και η απόκλιση 0.00416.

5.2.3 Ncrack

Το ncrack χρησιμοποιήθηκε για την εκτέλεση μιας επίθεσης λεξικού (dictionary attack). Χρησιμοποιήσαμε μία λειτουργία που παρέχει το ncrack, η οποία παίρνει ως είσοδο ένα αρχείο xml που έχει παραχθεί από μία σάρωση με το nmap. Εκτελέσαμε μία σάρωση τύπου SYN με το nmap και εξάγαμε τα αποτελέσματα σε αρχείο xml.

```
nmap -Pn -n -sS -oX scan.xml 10.101.30.60
```

Το ncrack εκτελέστηκε ως εξής:

```
ncrack -v --user localadmin -P /usr/share/wordlists/rockyou.txt -iX scan.xml
```

Η παράμετρος –v είναι για να αναφέρει λεπτομερώς τα αποτελέσματα του.

Η παράμετρος --user είναι το όνομα χρήστη που θα χρησιμοποιήσει στις υπηρεσίες που θα προσπαθήσει να βρει τον κωδικό τους.

Η παράμετρος –P παίρνει ως είσοδο ένα λεξικό με πολύ συχνούς κωδικούς που παρέχεται από το λειτουργικό σύστημα Kali Linux.

Τέλος, η παράμετρος –iX παίρνει ως είσοδο το αρχείο scan.xml με τα αποτελέσματα της σάρωσης με το nmap, ώστε να ξέρει σε ποιες θύρες εκτελούνται ποιες υπηρεσίες.

Η διάρκεια εκτέλεσης του ήταν ίση με αυτή που χρειάστηκε για την αναπαραγωγή της κίνησης δικτύου.

Στον Πίνακας 7 παρουσιάζουμε τις τιμές που πήραμε από τα αρχεία CSV και αφορούν τις δέκα επαναλήψεις της επίθεσης αυτής, όπως περιγράφηκε παραπάνω.

Position	Probability	First 100	Occurrence %
46	3.6E-05	1	14.95744833
12	2.44E-05	15	13.2936125
90	5.72E-05	3	14.49143378
11	3.05E-05	16	14.41986264
61	3.13E-05	9	14.99532856
4	3.03E-05	9	13.67442148
52	5.96E-05	8	13.6004411
10	3.11E-05	3	14.30937291
9	3.06E-05	6	13.56166913
18	6.51E-05	3	15.06248773

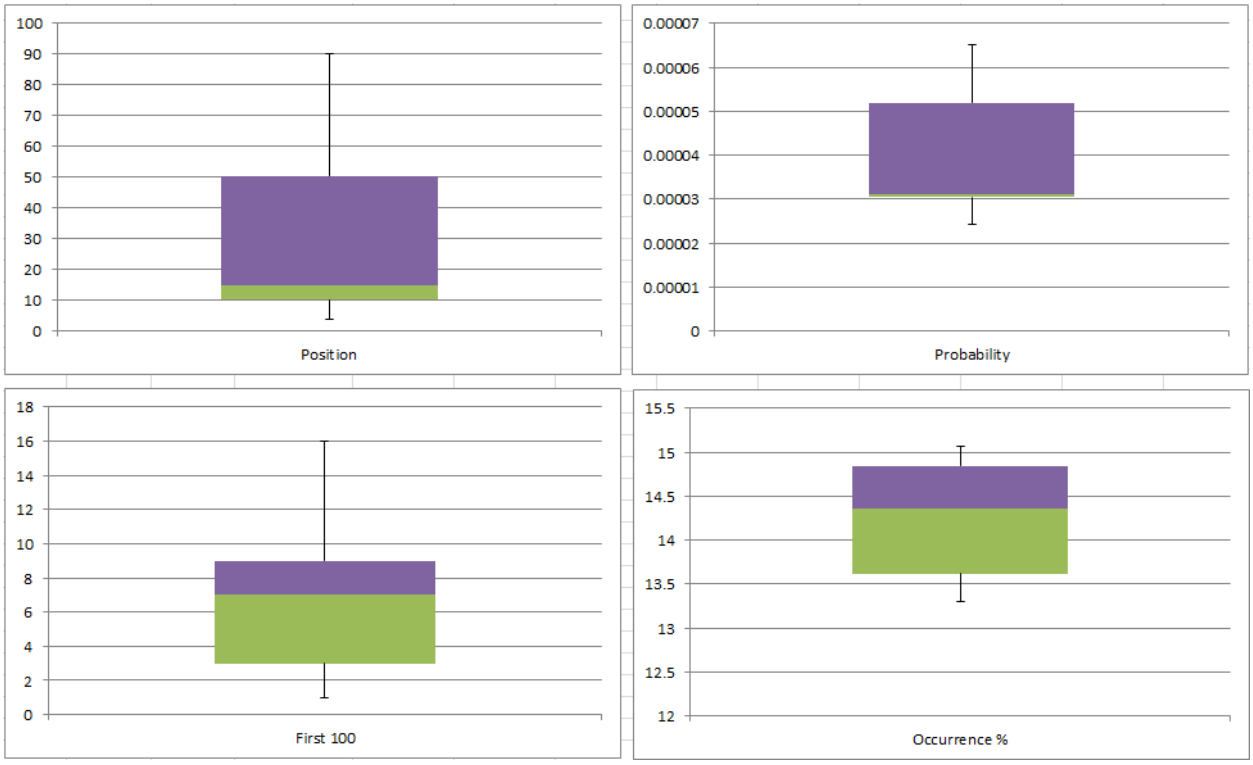
Πίνακας 7 – Αποτελέσματα ncrack

Στον Πίνακας 8 βλέπουμε τις τιμές που χρειαζόμαστε για τον σχεδιασμό των box plots της Εικόνα 20.

	Position	Probability	First 100	Occurrence %
Minimum	4	2.44E-05	1	13.2936125
1st Quartile	10.25	3.05E-05	3	13.61893619
Median	15	3.12E-05	7	14.36461777
3rd Quartile	50.5	5.19E-05	9	14.84094469
Maximum	90	6.51E-05	16	15.06248773

Πίνακας 8 – Δεδομένα για τα Box Plots του ncrack

5.2.3.1 Box Plots



Εικόνα 20 – ncrack Box Plots

Η διάμεσος (median) της κατάταξης (Position) είναι η τιμή 15 και το κεντρικό 50% των τιμών ορίζεται από το 10.25 έως το 50.5. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 50.5 ενώ η μέγιστη τιμή είναι το 90.

Στο γράφημα των πιθανοτήτων η διάμεσος είναι το 3.12E-05 και το κεντρικό 50% των τιμών ορίζεται από το 3.05E-05 έως το 5.19E-05. Η κατανομή είναι σχετικά συμμετρική, αφού η απόκλιση του κεντρικού 50% των τιμών είναι ελάχιστα προς τα κάτω.

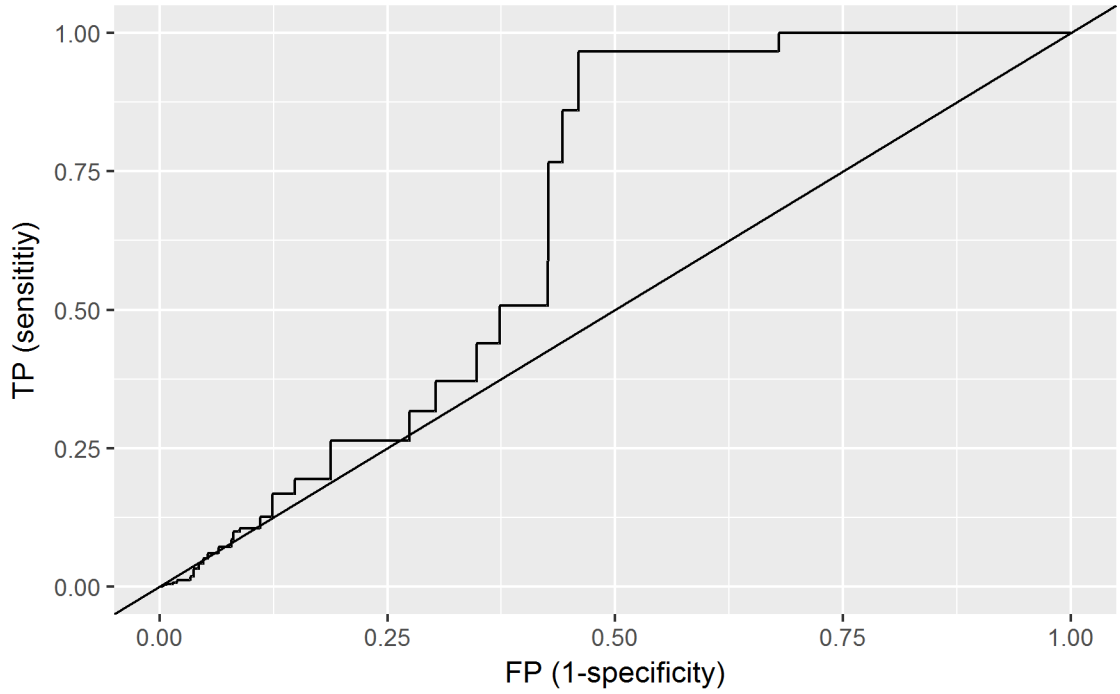
Στο γράφημα First 100 η διάμεσος είναι το 7 και το κεντρικό 50% των τιμών ορίζεται από το 3 έως το 9. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 9 ενώ η μέγιστη τιμή είναι το 16.

Στο γράφημα Occurrence % η διάμεσος είναι το ~14.36 και το κεντρικό 50% των τιμών ορίζεται από το ~13.61 έως το ~14.84. Η κατανομή είναι σχετικά συμμετρική, αφού η απόκλιση του κεντρικού 50% των τιμών είναι ελάχιστα προς τα πάνω.

5.2.3.2 ROC – AUROC

Στην Εικόνα 21 παρουσιάζουμε ενδεικτικά την μία καμπύλη ROC από τις δέκα που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Παρατηρούμε πως δεν απέχει πολύ από την ευθεία που αναπαριστά την τυχαιότητα για τα υψηλά καταταγμένα αποτελέσματα, το οποίο σημαίνει πως ο διαχωρισμός του μοντέλου σε αυτή τη περίπτωση δεν είναι καλός, αλλά όσο προχωράμε στα πιο χαμηλά καταταγμένα

αποτελέσματα ο διαχωρισμός βελτιώνεται. Στον Πίνακας 9 φαίνονται οι δέκα τιμές AUROC που πήραμε.



Εικόνα 21 – ROC ncrack

	Δοκιμή #1	Δοκιμή #2	Δοκιμή #3	Δοκιμή #4	Δοκιμή #5	Δοκιμή #6	Δοκιμή #7	Δοκιμή #8	Δοκιμή #9	Δοκιμή #10
AUROC	0.67162	0.71358	0.67215	0.69311	0.66626	0.68485	0.68792	0.68084	0.68811	0.67242
Μέση Τιμή: 0.683086					Απόκλιση: 0.000193					

Πίνακας 9 – AUROC ncrack

Η μέση τιμή της τιμής AUROC που παίρνουμε για τις 10 επιθέσεις με το εργαλείο ncrack είναι 0.683086 και η απόκλιση 0.000193.

5.2.4 t50

Το εργαλείο t50 χρησιμοποιήθηκε για να εκτελέσουμε μία επίθεση DoS. Το εκτελέσαμε ως εξής:

```
t50 10.101.30.60 --flood -s 10.101.10.160 --protocol T50 --shuffle
```

Η παράμετρος --flood ορίζει πως θα στείλει όσο πιο πολλά πακέτα μπορεί ξεπερνώντας το προεπιλεγμένο όριο των 1000 πακέτων.

Η παράμετρος –s ορίζει πως η ip που θα φαίνεται ως αποστολέας των πακέτων θα είναι αυτή που αναγράφεται, αλλάζοντας τη προεπιλεγμένη λειτουργία κατά την οποία παράγονται τυχαίες ip διευθύνσεις.

Οι παράμετροι --protocol T50 και --shuffle ορίζουν πως θα σταλθούν πακέτα όλων των πρωτοκόλλων, με τα πακέτα που αντιστοιχούν σε έκαστο από τα διαθέσιμα πρωτόκολλα να τοποθετούνται σε τυχαία σειρά εντός της ακολουθίας των πακέτων επίθεσης.

Η διάρκεια εκτέλεσης της επίθεσης ήταν ίση με τη διάρκεια που χρειάστηκε για την αναπαραγωγή της κίνησης δικτύου.

Στον Πίνακας 10 παρουσιάζουμε τις τιμές που πήραμε από τα αρχεία CSV από τις δέκα επαναλήψεις της επίθεσης αυτής όπως περιγράφηκε παραπάνω.

Position	Probability	First 100	Occurrence %
1	1.06E-06	85	99.60015754
1	1.1E-06	89	99.51736548
4	5.72E-07	62	99.3844986
1	3.74E-07	71	99.40605164
2	5.53E-07	74	99.31688309
3	1.04E-06	91	99.49811997
1	1.09E-06	80	99.32231903
3	9.23E-07	85	99.48971425
1	5.04E-07	69	99.24812317
1	5.4E-07	65	99.27425385

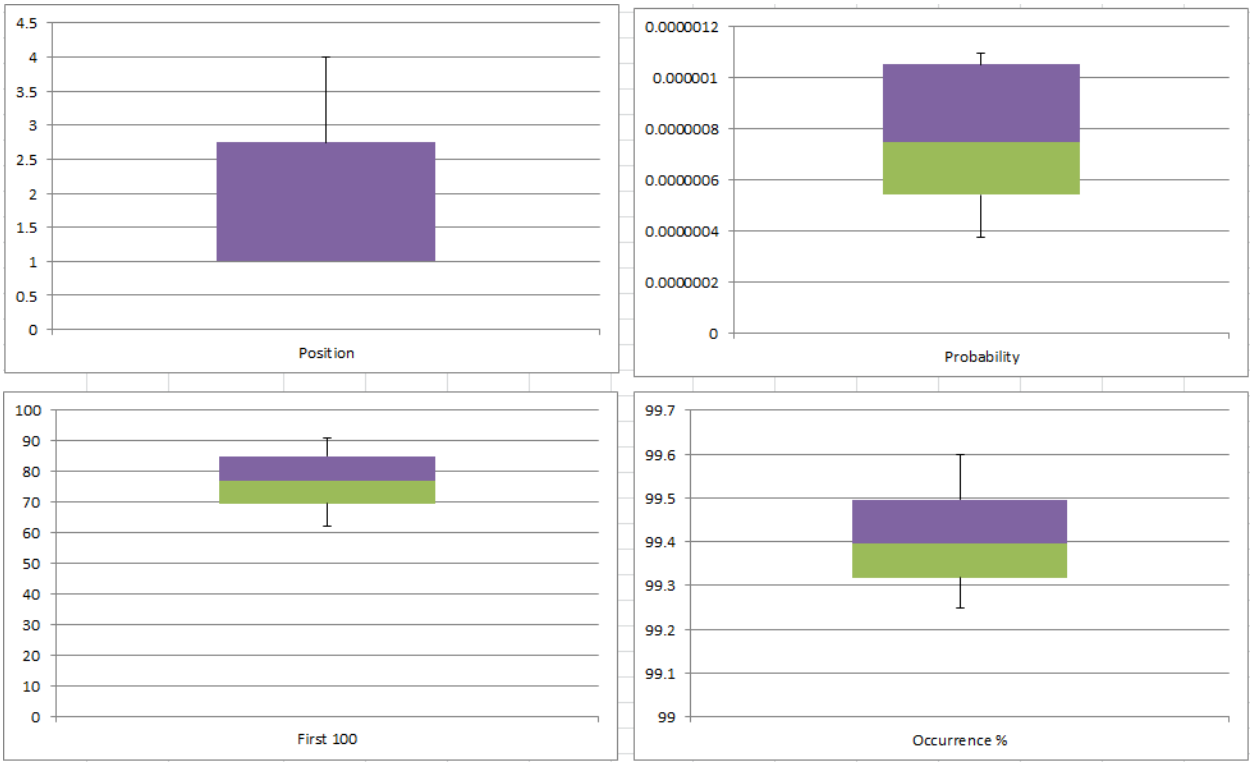
Πίνακας 10 – Αποτελέσματα t50

Στον Πίνακας 11 βλέπουμε τις τιμές που χρειαζόμαστε για τον σχεδιασμό των box plots της Εικόνα 22.

	Position	Probability	First 100	Occurrence %
Minimum	1	3.74E-07	62	99.24812317
1st Quartile	1	5.43E-07	69.5	99.31824207
Median	1	7.47E-07	77	99.39527512
3rd Quartile	2.75	1.05E-06	85	99.49601854
Maximum	4	1.1E-06	91	99.60015754

Πίνακας 11 – Δεδομένα για τα Box Plots του t50

5.2.4.1 Box Plots



Εικόνα 22 – t50 Box Plots

Η διάμεσος (median) της κατάταξης (Position) είναι η τιμή 1 και το κεντρικό 50% των τιμών ορίζεται από το 1 έως το 2.75. Παρατηρούμε στον Πίνακας 10 πως το κάτω 60% των τιμών είναι 1, αυτό εξηγεί γιατί δεν εμφανίζεται το κάτω 50% των τιμών στο box plot και επίσης κάνει τη κατανομή ασύμμετρη.

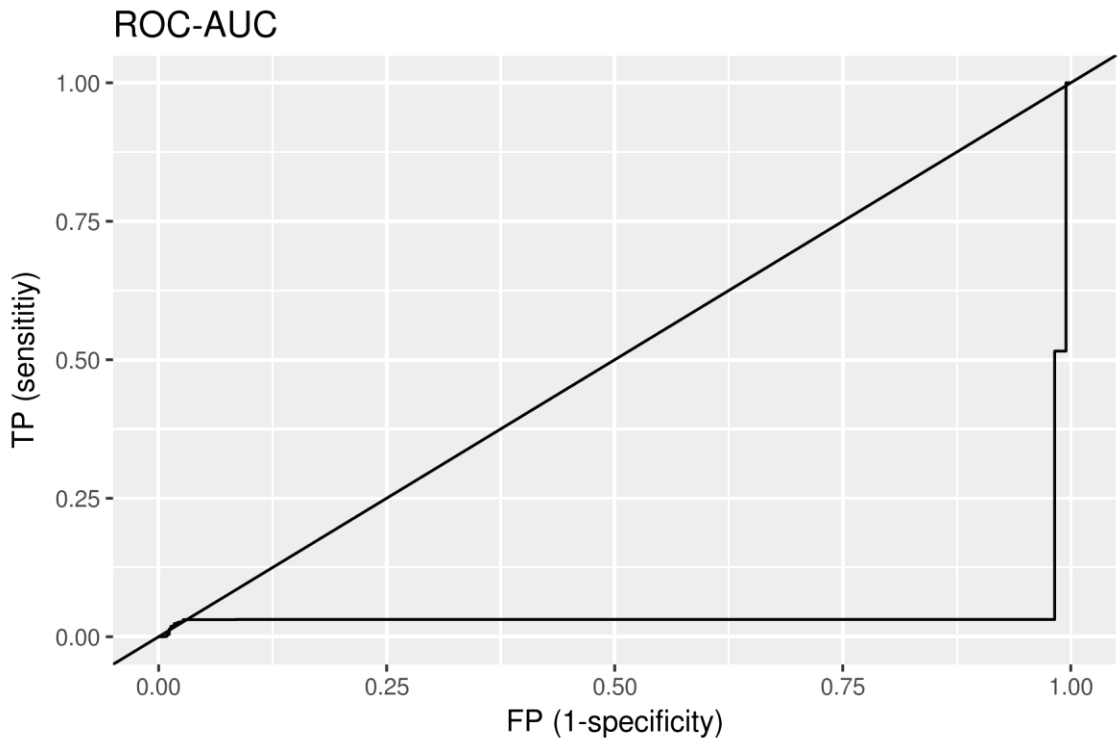
Στο γράφημα των πιθανοτήτων η διάμεσος είναι το 7.47E-07 και το κεντρικό 50% των τιμών ορίζεται από το 5.43E-07 έως το 1.05E-06. Η κατανομή είναι σχετικά συμμετρική, αφού η απόκλιση του κεντρικού 50% των τιμών είναι ελάχιστα προς τα πάνω.

Στο γράφημα First 100 η διάμεσος είναι το 77 και το κεντρικό 50% των τιμών ορίζεται από το 69.5 έως το 85. Η κατανομή είναι σχετικά συμμετρική, αφού η απόκλιση του κεντρικού 50% των τιμών είναι ελάχιστα προς τα πάνω.

Στο γράφημα Occurrence % η διάμεσος είναι το ~99.39 και το κεντρικό 50% των τιμών ορίζεται από το ~99.32 έως το ~99.49. Η κατανομή είναι σχετικά συμμετρική, αφού η απόκλιση του κεντρικού 50% των τιμών είναι ελάχιστα προς τα κάτω.

5.2.4.2 ROC – AUROC

Στην Εικόνα 23 παρουσιάζουμε ενδεικτικά την μία καμπύλη ROC από τις δέκα που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Παρατηρούμε πως είναι πολύ κάτω από τη γραμμή της τυχαιότητας, έτσι ο διαχωρισμός είναι ο αντίθετος από αυτόν που θα θέλαμε. Στον Πίνακας 12 φαίνονται οι δέκα τιμές AUROC που πήραμε.



Εικόνα 23 – ROC t50

	Δοκιμή #1	Δοκιμή #2	Δοκιμή #3	Δοκιμή #4	Δοκιμή #5	Δοκιμή #6	Δοκιμή #7	Δοκιμή #8	Δοκιμή #9	Δοκιμή #10
AUROC	0.04171	0.05151	0.03245	0.04038	0.03672	0.05104	0.0481	0.05503	0.0442	0.03285
Μέση Τιμή: 0.0434					Απόκλιση: 6.33E-05					

Πίνακας 12 – AUROC t50

Η μέση τιμή της τιμής AUROC που παίρνουμε για τις 10 επιθέσεις με την επίθεση t50 είναι 0.0434 και η απόκλιση 6.33E-05.

5.2.5 Slowloris

Η επίθεση Slowloris εκτελέστηκε χρησιμοποιώντας μία υλοποίηση της σε python από το github (Yaltirakli, 2017). Εκτελέστηκε με τις προεπιλεγμένες ρυθμίσεις παρέχοντας μόνο την ip-στόχο και ως θύρα-στόχο την 80. Η λειτουργία της επίθεσης Slowloris εξηγήθηκε προηγουμένως στην ενότητα 3.2.3.

Στον Πίνακας 13 παρουσιάζουμε τις τιμές που πήραμε από τα αρχεία CSV από τις δέκα επαναλήψεις της επίθεσης αυτής όπως περιγράφηκε παραπάνω.

Position	Probability	First 100	Occuerrence %
1	3.03E-05	1	3.388494878
45	4.09E-05	2	3.109037018
3	2.95E-05	4	3.802653298
46	3.1E-05	1	2.958544408
2	2.31E-05	4	2.6737318
1	1.95E-05	4	2.842614412
1	1.93E-05	2	2.782220821
3	2.38E-05	6	2.995452056
6	2.48E-05	1	2.626526191
1	1.95E-05	3	2.647144165

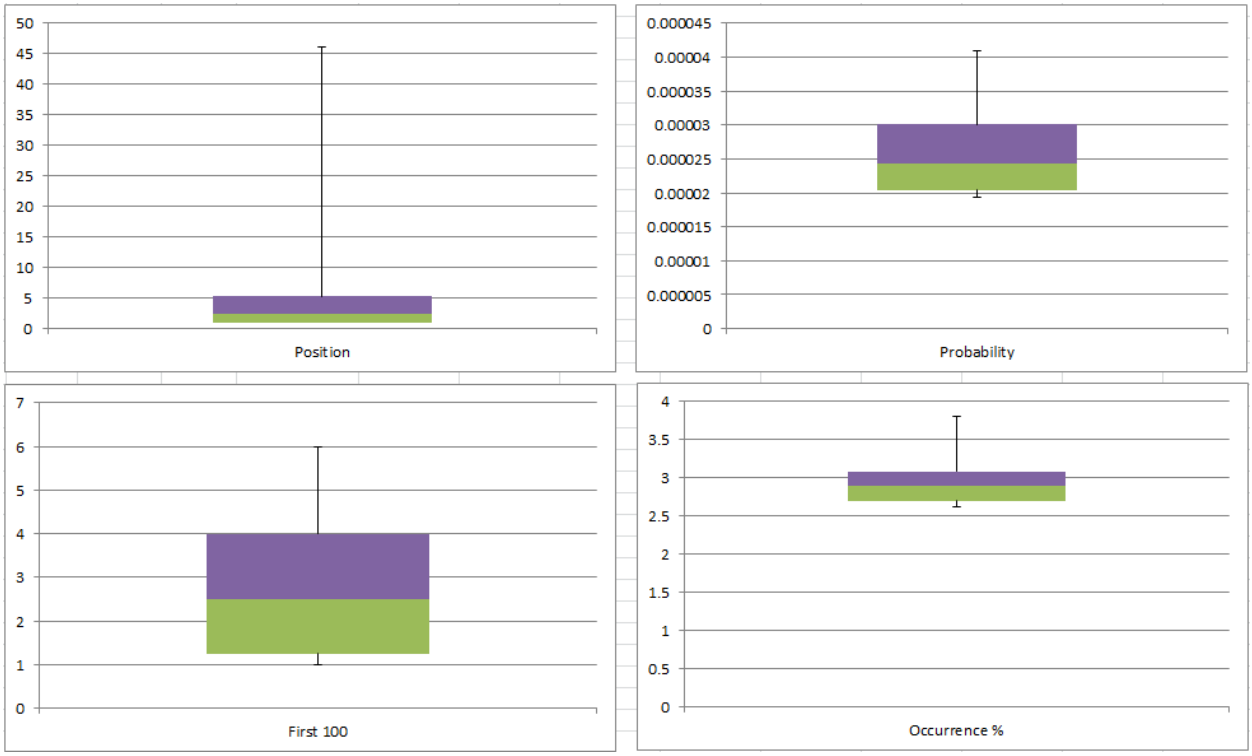
Πίνακας 13 – Αποτελέσματα Slowloris

Στον Πίνακας 14 βλέπουμε τις τιμές που χρειαζόμαστε για τον σχεδιασμό των box plots της Εικόνα 24.

	Position	Probability	First 100	Occuerrence %
Minimum	1	1.93E-05	1	2.626526191
1st Quartile	1	2.04E-05	1.25	2.700854055
Median	2.5	2.43E-05	2.5	2.90057941
3rd Quartile	5.25	3.01E-05	4	3.080640778
Maximum	46	4.09E-05	6	3.802653298

Πίνακας 14 – Δεδομένα για τα Box Plots του Slowloris

5.2.5.1 Box Plots



Εικόνα 24 – Slowloris Box Plots

Η διάμεσος (median) της κατάταξης (Position) είναι η τιμή 2.5 και το κεντρικό 50% των τιμών ορίζεται από το 1 έως το 5.25. Παρατηρούμε στον Πίνακας 13 πως το κάτω 40% των τιμών είναι 1, αυτό εξηγεί γιατί δεν εμφανίζεται το κάτω 25% των τιμών στο box plot. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 5.25 ενώ η μέγιστη τιμή είναι το 46.

Στο γράφημα των πιθανοτήτων η διάμεσος είναι το 2.43E-05 και το κεντρικό 50% των τιμών ορίζεται από το 2.04E-05 έως το 3.01E-05. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 3.01E-05 ενώ η μέγιστη τιμή είναι το 4.09E-05.

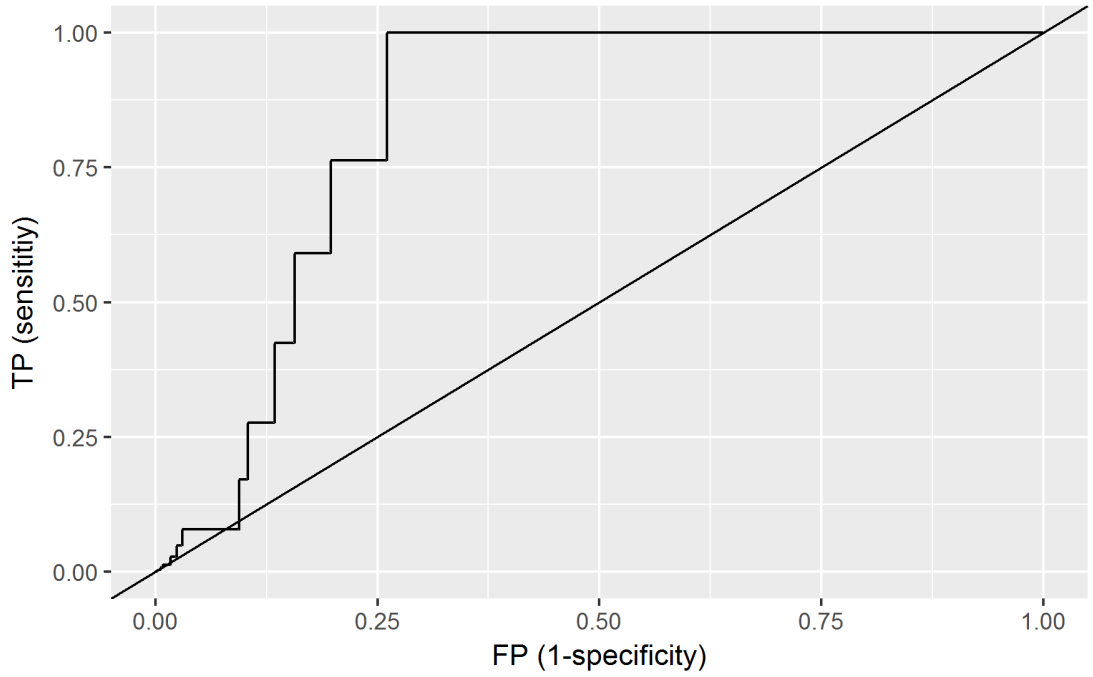
Στο γράφημα First 100 η διάμεσος είναι το 2.5 και το κεντρικό 50% των τιμών ορίζεται από το 1.25 έως το 4. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 4 ενώ η μέγιστη τιμή είναι το 6.

Στο γράφημα Occurrence % η διάμεσος είναι το ~2.90 και το κεντρικό 50% των τιμών ορίζεται από το ~2.70 έως το ~3.08. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το ~3.08 ενώ η μέγιστη τιμή είναι το ~3.8.

5.2.5.2 ROC - AUROC

Στην Εικόνα 25 παρουσιάζουμε ενδεικτικά την μία καμπύλη ROC από τις δέκα που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Παρατηρούμε πως απέχει πολύ από την ευθεία που αναπαριστά την

τυχαιότητα, το οποίο σημαίνει πως ο διαχωρισμός του μοντέλου σε αυτή τη περίπτωση είναι καλός. Στον Πίνακας 15 φαίνονται οι δέκα τιμές AUROC που πήραμε.



Εικόνα 25 – ROC Slowloris

	Δοκιμή #1	Δοκιμή #2	Δοκιμή #3	Δοκιμή #4	Δοκιμή #5	Δοκιμή #6	Δοκιμή #7	Δοκιμή #8	Δοκιμή #9	Δοκιμή #10
AUROC	0.70613	0.83707	0.71053	0.80984	0.78203	0.76789	0.75087	0.79304	0.82088	0.82342
Μέση Τιμή: 0.780171					Απόκλιση: 0.002128					

Πίνακας 15 – AUROC Slowloris

Η μέση τιμή της τιμής AUROC που παίρνουμε για τις 10 επιθέσεις με την επίθεση Slowloris είναι 0.780171 και η απόκλιση 0.002128.

5.2.6 BoNeSi

Χρησιμοποιήσαμε το εργαλείο BoNeSi για την εκτέλεση μίας επίθεσης DoS. Εκτελέστηκε με τις προεπιλεγμένες ρυθμίσεις παρέχοντας μόνο την IP στόχο και ως θύρα-στόχο την 80. Το BoNeSi έχει ως προεπιλογή να στέλνει πακέτα τύπου UDP αν δεν ζητηθεί άλλο πρωτόκολλο από τον χρήστη.

Στον Πίνακας 16 παρουσιάζουμε τις τιμές που πήραμε από τα αρχεία CSV από τις δέκα επαναλήψεις της επίθεσης αυτής όπως περιγράφηκε παραπάνω.

Position	Probability	First 100	Occurrence %
40	3E-05	2	76.27055362
36	1.98E-05	3	76.12198357
63	3.92E-05	2	72.17966475
28	2.08E-05	2	79.11053165

124	5.88E-05	0	72.08022243
132	0.000118	0	61.87038159
47	3E-05	2	76.16703952
72	3.92E-05	3	81.13165516
29	1.31E-05	3	78.40206661
105	3.15E-05	0	78.17122694

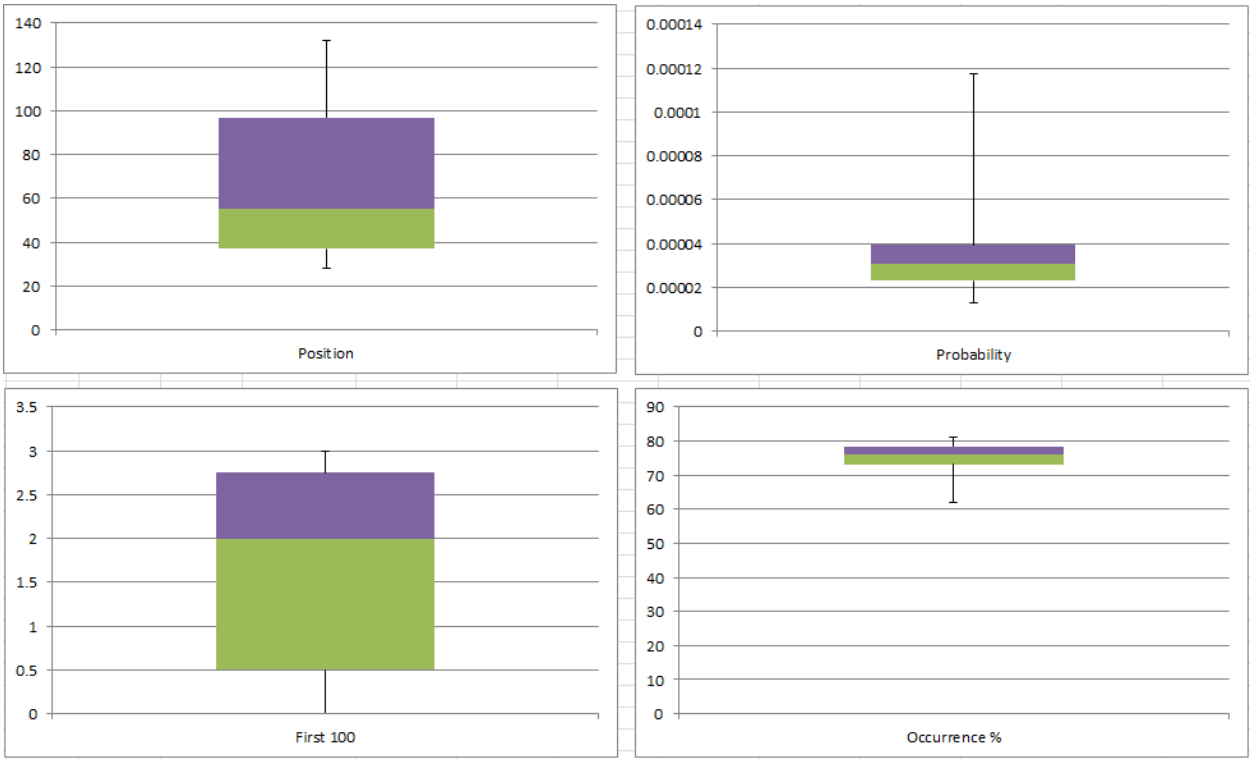
Πίνακας 16 – Αποτελέσματα BoNeSi

Στον Πίνακας 17 βλέπουμε τις τιμές που χρειαζόμαστε για τον σχεδιασμό των box plots της Εικόνα 26.

	Position	Probability	First 100	Occurrence %
Minimum	28	1.31E-05	0	61.87038159
1st Quartile	37	2.31E-05	0.5	73.16524446
Median	55	3.08E-05	2	76.21879657
3rd Quartile	96.75	3.92E-05	2.75	78.34435669
Maximum	132	11.8E-05	3	81.13165516

Πίνακας 17 – Δεδομένα για τα Box Plots του BoNeSi

5.2.6.1 Box Plots



Εικόνα 26 – BoNeSi Box Plots

Η διάμεσος (median) της κατάταξης (Position) είναι η τιμή 55 και το κεντρικό 50% των τιμών ορίζεται από το 37 έως το 96.75. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 96.75 ενώ η μέγιστη τιμή είναι το 132.

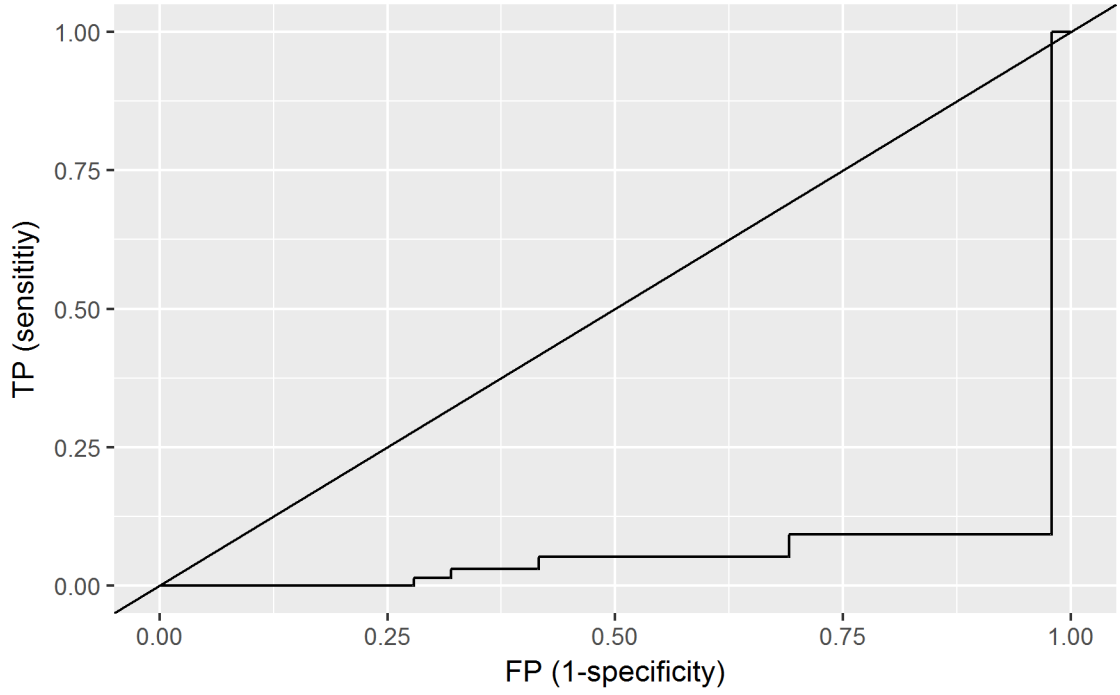
Στο γράφημα των πιθανοτήτων η διάμεσος είναι το 3.08E-05 και το κεντρικό 50% των τιμών ορίζεται από το 2.31E-05 έως το 3.92E-05. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 3.92E-05 ενώ η μέγιστη τιμή είναι το 11.8E-05.

Στο γράφημα First 100 η διάμεσος είναι το 2 και το κεντρικό 50% των τιμών ορίζεται από το 0.5 έως το 2.75. Η κατανομή είναι σχετικά συμμετρική, αφού η απόκλιση του κεντρικού 50% των τιμών είναι ελάχιστα προς τα πάνω.

Στο γράφημα Occurrence % η διάμεσος είναι το ~76.22 και το κεντρικό 50% των τιμών ορίζεται από το ~73.16 έως το ~78.34. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το άνω 75% των τιμών είναι πάνω από το ~73.16 ενώ η ελάχιστη τιμή είναι το ~61.87.

5.2.6.2 ROC – AUROC

Στην Εικόνα 27 παρουσιάζουμε ενδεικτικά την μία καμπύλη ROC από τις δέκα που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Παρατηρούμε πως ο διαχωρισμός του μοντέλου σε αυτή τη περίπτωση δεν είναι καθόλου καλός, καθώς ο διαχωρισμός είναι πλήρως λανθασμένος. Στον Πίνακας 18 φαίνονται οι δέκα τιμές AUROC που πήραμε.



Εικόνα 27 – ROC BoNeSi

	Δοκιμή #1	Δοκιμή #2	Δοκιμή #3	Δοκιμή #4	Δοκιμή #5	Δοκιμή #6	Δοκιμή #7	Δοκιμή #8	Δοκιμή #9	Δοκιμή #10
AUROC	0.04333	0.04168	0.03832	0.05949	0.06492	0.05606	0.04501	0.06449	0.03994	0.0405
Μέση Τιμή: 0.049373					Απόκλιση: 0.000114					

Πίνακας 18 – AUROC BoNeSi

Η μέση τιμή της τιμής AUROC που παίρνουμε για τις 10 επιθέσεις με το εργαλείο BoNeSi είναι 0.049373 και η απόκλιση 0.000114.

5.2.7 Armitage

Το εργαλείο Armitage χρησιμοποιήθηκε για την εκτέλεση της αυτοματοποιημένης επίθεσης Hail Mary. Αρχικά πραγματοποιήσαμε μία σάρωση με το nmap, μέσα από το Armitage που παρέχει αυτή τη δυνατότητα. Στη συνέχεια επιλέγοντας τη λειτουργία Hail Mary, το Armitage βρίσκει τα κατάλληλα προγράμματα εκμετάλλευσης ευπαθειών, βασισμένο στις πληροφορίες που πήρε από τη σάρωση με το nmap, τα ταξινομεί σε βέλτιστη σειρά εκτέλεσης και εκκινεί την επίθεση.

Στον Πίνακας 19 παρουσιάζουμε τις τιμές που πήραμε από τα αρχεία CSV από τις δέκα επαναλήψεις της επίθεσης αυτής όπως περιγράφηκε παραπάνω.

Position	Probability	First 100	Occurrence %
4	2.94E-05	3	34.11743793
6	1.77E-05	27	30.09741291
1	2.23E-05	22	23.82069263
1	2.66E-05	19	21.84865961
1	2.1E-05	8	25.91255617
1	1.32E-05	34	36.02020686
1	1.6E-05	23	37.77348422
1	2.78E-05	4	24.79637076
1	9.72E-06	39	29.41497044
1	1.75E-05	12	23.96520023

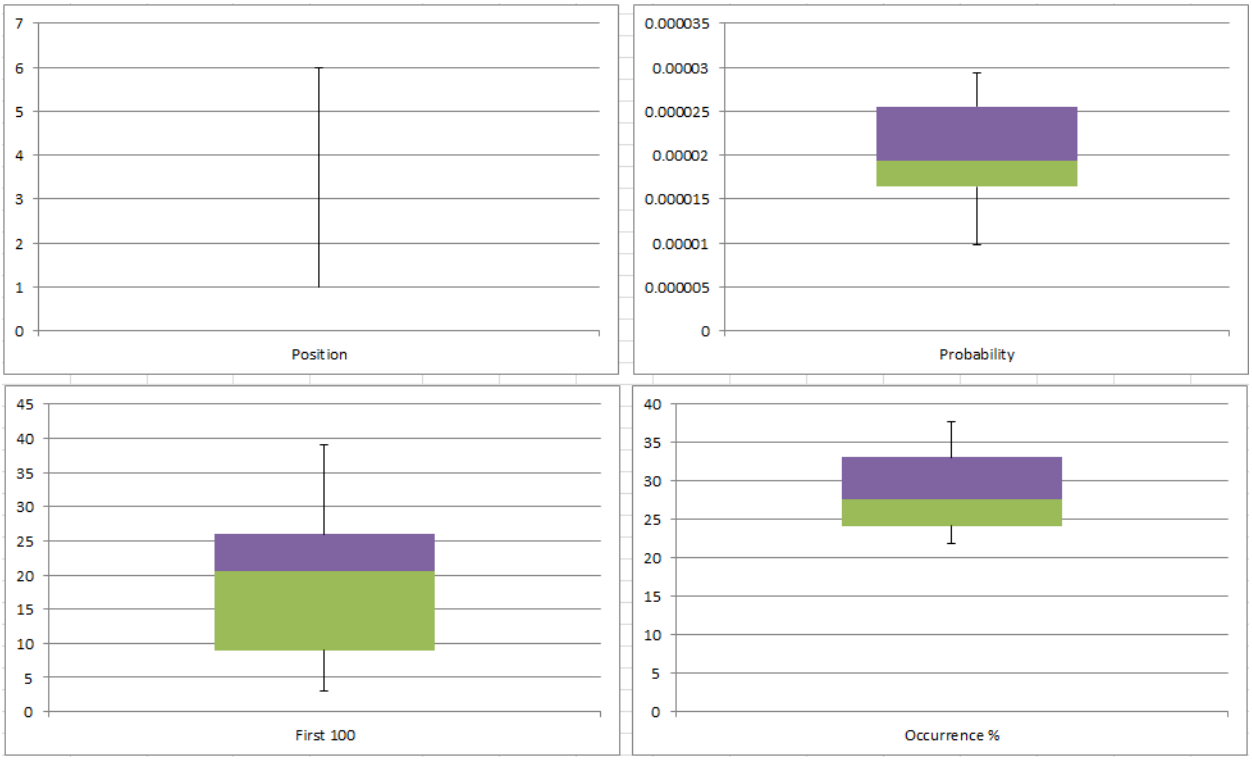
Πίνακας 19 – Αποτελέσματα Armitage

Στον Πίνακας 20 βλέπουμε τις τιμές που χρειαζόμαστε για τον σχεδιασμό των box plots της Εικόνα 28.

	Position	Probability	First 100	Occurrence %
Minimum	1	9.72E-06	3	21.84865961
1st Quartile	1	1.64E-05	9	24.17299286
Median	1	1.93E-05	20.5	27.6637633
3rd Quartile	1	2.55E-05	26	33.11243168
Maximum	6	2.94E-05	39	37.77348422

Πίνακας 20 – Δεδομένα για τα Box Plots του Armitage

5.2.7.1 Box Plots



Εικόνα 28 – Armitage Box Plots

Η διάμεσος (median) της κατάταξης (Position) είναι η τιμή 1. Παρατηρούμε στον Πίνακας 19 πως το κάτω 80% των τιμών είναι 1, αυτό εξηγεί γιατί εμφανίζεται μόνο το άνω 25% των τιμών στο box plot. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 80% των τιμών έχει τιμή 1 ενώ η μέγιστη τιμή είναι το 6.

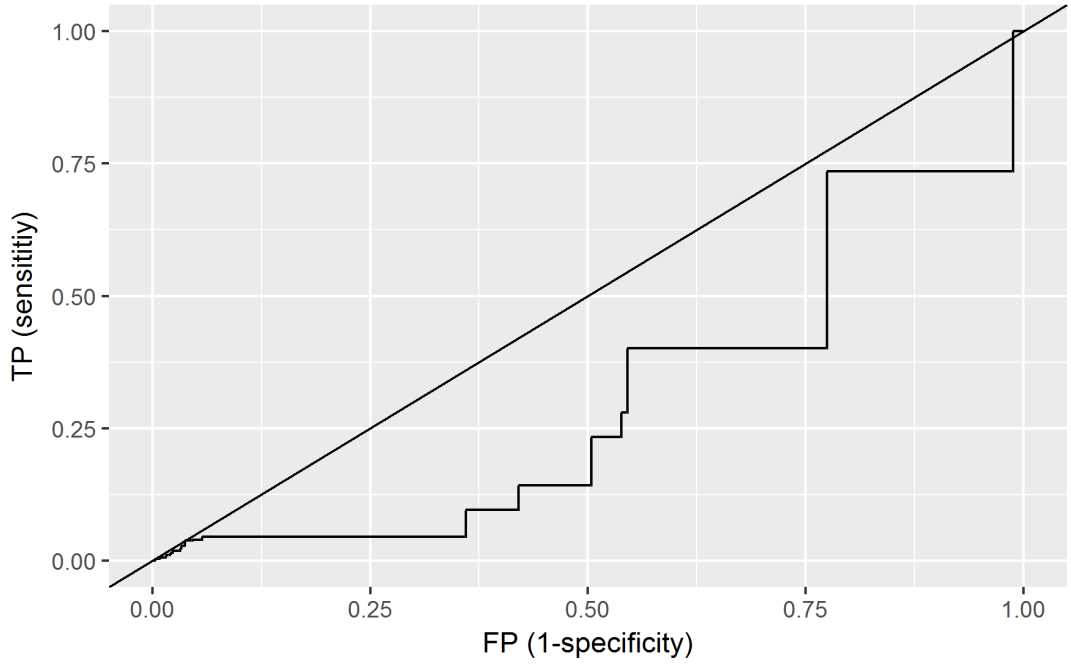
Στο γράφημα των πιθανοτήτων η διάμεσος είναι το 1.93E-05 και το κεντρικό 50% των τιμών ορίζεται από το 1.64E-05 έως το 2.55E-05. Η κατανομή είναι σχετικά συμμετρική, αφού η απόκλιση του κεντρικού 50% των τιμών είναι ελάχιστα προς τα πάνω.

Στο γράφημα First 100 η διάμεσος είναι το 20.5 και το κεντρικό 50% των τιμών ορίζεται από το 9 έως το 26. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 26 ενώ η μέγιστη τιμή είναι το 39.

Στο γράφημα Occurrence % η διάμεσος είναι το ~27.66 και το κεντρικό 50% των τιμών ορίζεται από το ~24.17 έως το ~33.11. Η κατανομή είναι σχετικά συμμετρική, αφού η απόκλιση του κεντρικού 50% των τιμών είναι ελάχιστα προς τα κάτω.

5.2.7.2 ROC – AUROC

Στην Εικόνα 29 παρουσιάζουμε ενδεικτικά την μία καμπύλη ROC από τις δέκα που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Παρατηρούμε πως ο διαχωρισμός του μοντέλου σε αυτή τη περίπτωση δεν είναι καλός, αφού η καμπύλη είναι κάτω από την ευθεία που αναπαριστά τη τυχαιότητα. Στον Πίνακας 21 φαίνονται οι δέκα τιμές AUROC που πήραμε.



Εικόνα 29 – ROC Armitage

	Δοκιμή #1	Δοκιμή #2	Δοκιμή #3	Δοκιμή #4	Δοκιμή #5	Δοκιμή #6	Δοκιμή #7	Δοκιμή #8	Δοκιμή #9	Δοκιμή #10
AUROC	0.10014	0.14771	0.28873	0.30337	0.19817	0.15256	0.19965	0.19698	0.2366	0.23892
Μέση Τιμή: 0.206282					Απόκλιση: 0.003998					

Πίνακας 21 – AUROC Armitage

Η μέση τιμή της τιμής AUROC που παίρνουμε για τις 10 επιθέσεις με το εργαλείο Armitage είναι 0.206282 και η απόκλιση 0.003998.

5.2.8 Iodine

Το εργαλείο Iodine χρησιμοποιήθηκε για να εκτελεστεί μία επίθεση DNS Tunneling. Στην επίθεση αυτή χρησιμοποιήσαμε την διάταξη που παρουσιάσαμε στην ενότητα 4.2. Στα μηχανήματα Client, Server και cloudera-host-2 εκτελέσαμε ένα script που παρέχει το PhantomJS, το οποίο φορτώνει γνωστές ιστοσελίδες, παράγοντας έτσι φυσιολογική κίνηση στο δίκτυο. Το cloudera-host-2 θα παίξει τον ρόλο του επιτιθέμενου δημιουργώντας ένα DNS tunnel με τον DNS Server (βλ. Εικόνα 13). Συγκεκριμένα στον DNS Server εκτελείται η παρακάτω εντολή:

```
./iodined -f -P secretpass 192.168.99.1 t1.iliketunnels.com
```

Η παράμετρος –f ορίζει πως το iodined θα εκτελείται στο προσκήνιο.

Η παράμετρος -P ορίζει τον κωδικό που θα χρησιμοποιήσουν ο DNS Server και το cloudera-host-2.

Το iodined κατά την εκτέλεση του δημιουργεί μία διεπαφή δικτύου (network interface) και το όρισμα “192.168.99.1” ορίζει την IP για αυτή τη διεπαφή δικτύου.

Το όρισμα “t1.iliketunnels.com” ορίζει τον τομέα (domain) του DNS Server. Αυτό σημαίνει πως όποιο DNS ερώτημα (DNS query) τελειώνει με αυτό τον τομέα (domain) θα προωθείται στον DNS Server.

Για τη παραγωγή και μεταφορά δεδομένων από το cloudera-host-2 στον DNS Server χρησιμοποιήσαμε το εργαλείο iperf. Μετά την εκτέλεση του iodined, εκτελέσαμε την εξής εντολή στον DNS Server:

```
iperf -s -B 192.168.99.1
```

Η παράμετρος –s ορίζει πως εκτελείται σε λειτουργία εξυπηρετητή (server mode) και η παράμετρος –B ορίζει τη διεπαφή δικτύου στην οποία θα δεσμευθεί (bind), δηλαδή τη διεπαφή δικτύου στην οποία θα περιμένει να λάβει τα δεδομένα από το cloudera-host-2.

Στο cloudera-host-2 εκτελούμε το iodine ως εξής:

```
./iodine -r 10.101.10.35 -P secretpass -f t1.iliketunnels.com
```

Το όρισμα –r ορίζει πως θα πρέπει να χρησιμοποιηθεί DNS Tunneling σε κάθε περίπτωση, καθώς μπορεί να επιτρέπεται το UDP tunneling και το iodine να χρησιμοποιήσει αυτό αντί για DNS tunneling.

Το όρισμα “10.101.10.35” είναι η IP του DNS Server και το όρισμα t1.iliketunnels.com είναι ο τομέας (domain) του DNS Server. Οι άλλες δύο παράμετροι εξηγήθηκαν προηγουμένως.

Το iperf στο cloudera-host-2 εκτελέστηκε ως εξής:

```
iperf -c 192.168.99.1
```

Το –c ορίζει πως θα εκτελεστή σε λειτουργία πελάτη (client mode) και το όρισμα “192.168.99.1” είναι η διεπαφή δικτύου (network interface) που δημιουργήσαμε στον DNS Server.

Η παραπάνω επίθεση εκτελέστηκε 5 φορές όπως περιγράφηκε παραπάνω και άλλες 5 χωρίς να χρησιμοποιήσουμε τα μηχανήματα Client και Server για την παράγωγη κίνησης με το PhantomJS. Χρησιμοποιήθηκε μόνο το cloudera-host-2, για σκοπούς σύγκρισης της ανταπόκρισης του Apache Spot μεταξύ των δύο σεναρίων. Θα ξεκινήσουμε από τα αποτελέσματα του σεναρίου που περιλαμβάνει τα μηχανήματα Client και Server.

Στον Πίνακας 22 παρουσιάζουμε τις τιμές που πήραμε από τα αρχεία CSV από τις πέντε επαναλήψεις της επίθεσης αυτής όπως περιγράφηκε παραπάνω.

Position	Probability	First 100	Occurrence %
4396	0.00121388	0	2.654931102
4832	3.72E-04	0	0.96459498
1323	6.80E-04	0	3.266025207
48	1.95E-04	5	4.329268293
6599	3.50E-04	0	0.887902331

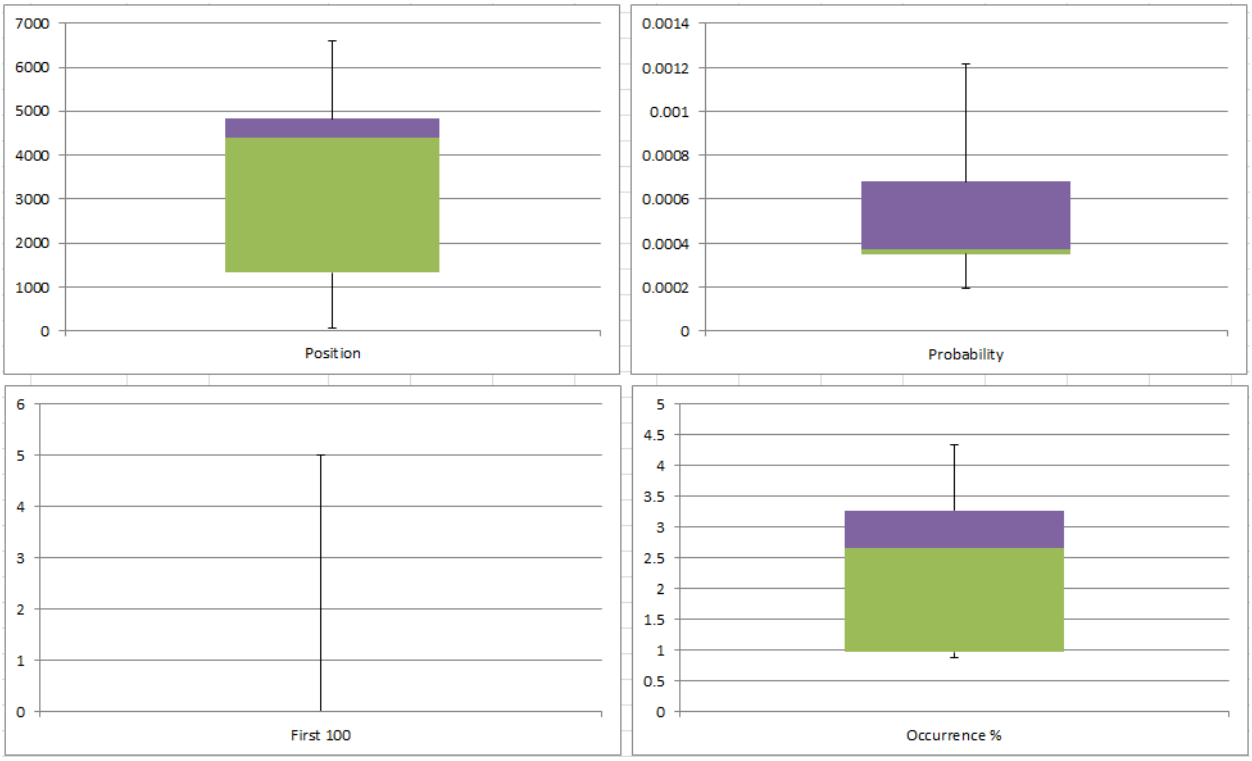
Πίνακας 22 – Αποτελέσματα Iodine 1

Στον Πίνακας 23 βλέπουμε τις τιμές που χρειαζόμαστε για τον σχεδιασμό των box plots της Εικόνα 30.

	Position	Probability	First 100	Occurrence %
Minimum	48	1.95E-04	0	0.887902331
1st Quartile	1323	3.5E-04	0	0.96459498
Median	4396	3.72E-04	0	2.654931102
3rd Quartile	4832	6.8E-04	0	3.266025207
Maximum	6599	12.1E-04	5	4.329268293

Πίνακας 23 – Δεδομένα για τα Box Plots του Iodine 1

5.2.8.1 Box Plots



Εικόνα 30 – Iodine 1 Box Plots

Η διάμεσος (median) της κατάταξης (Position) είναι η τιμή 4396 και το κεντρικό 50% των τιμών ορίζεται από το 1323 έως το 4832. Η κατανομή είναι σχετικά συμμετρική, αφού η απόκλιση του κεντρικού 50% των τιμών είναι ελάχιστα προς τα κάτω.

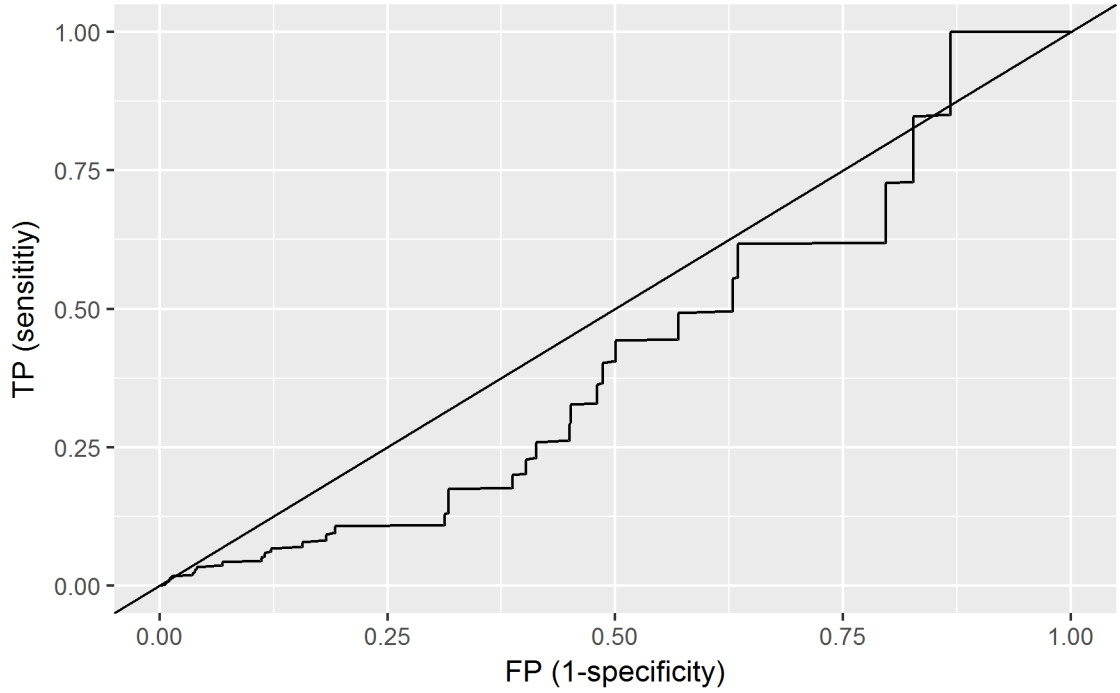
Στο γράφημα των πιθανοτήτων η διάμεσος είναι το 3.72E-04 και το κεντρικό 50% των τιμών ορίζεται από το 3.5E-04 έως το 6.8E-04. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 6.8E-04 ενώ η μέγιστη τιμή είναι το 12.1E-04.

Στο γράφημα First 100 η διάμεσος είναι το 0. Παρατηρούμε στον Πίνακας 22 ότι το 80% των τιμών είναι 0, αυτό εξηγεί γιατί εμφανίζεται μόνο το άνω 25% των τιμών στο box plot και επίσης λόγω αυτού η κατανομή είναι ασύμμετρη.

Στο γράφημα Occurrence % η διάμεσος είναι το ~2.65 και το κεντρικό 50% των τιμών ορίζεται από το ~0.96 έως το ~3.26. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι μέχρι το ~3.26 ενώ η μέγιστη τιμή είναι το ~4.32.

5.2.8.2 ROC-AUROC

Στην Εικόνα 31 παρουσιάζουμε ενδεικτικά την μία καμπύλη ROC από τις πέντε που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Παρατηρούμε πως ο διαχωρισμός του μοντέλου σε αυτή τη περίπτωση δεν είναι καλός, αφού η καμπύλη βρίσκεται κάτω από την ευθεία που αναπαριστά τη τυχαιότητα. Στον Πίνακας 24 φαίνονται οι πέντε τιμές AUROC που πήραμε.



Εικόνα 31 – ROC Iodine 1

	Δοκιμή #1	Δοκιμή #2	Δοκιμή #3	Δοκιμή #4	Δοκιμή #5
AUROC	0.10635	0.14384	0.14195	0.42177	0.27865
Μέση Τιμή: 0.218512			Απόκλιση: 0.017237		

Πίνακας 24 – AUROC Iodine 1

Η μέση τιμή της τιμής AUROC που παίρνουμε για τις 5 επιθέσεις με το εργαλείο Iodine είναι 0.218512 και η απόκλιση 0.017237.

Στη συνέχεια παρουσιάζουμε τα αποτελέσματα από την εκτέλεση της επίθεσης χωρίς τη συμμετοχή των Client και Server.

Στον Πίνακας 25 παρουσιάζουμε τις τιμές που πήραμε από τα αρχεία CSV από τις πέντε επαναλήψεις της επίθεσης αυτής.

Position	Probability	First 100	Occurrence %
21	1.78E-03	5	19.68911917
77	1.75E-03	1	19.12013536
6	1.74E-03	2	19.21768707
54	1.68E-03	3	18.81188119
1	1.72E-04	4	21.79930796

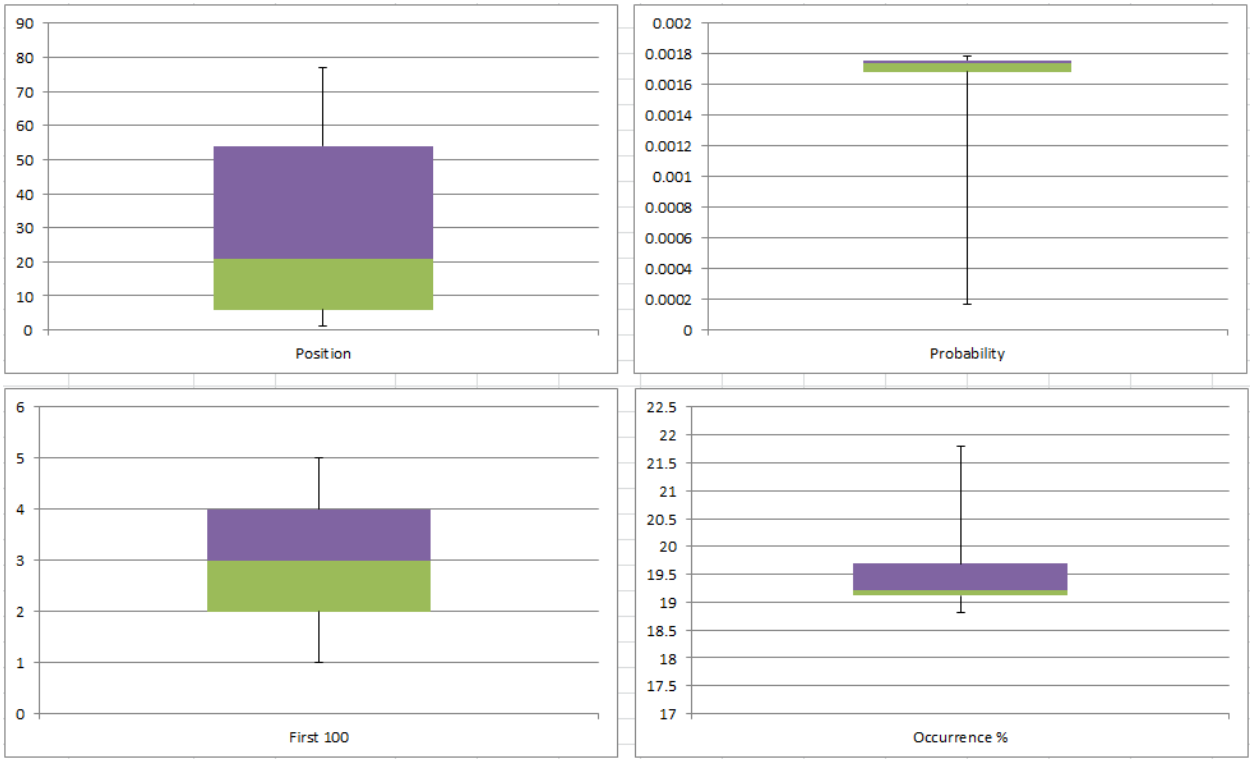
Πίνακας 25 – Αποτελέσματα Iodine 2

Στον Πίνακας 26 βλέπουμε τις τιμές που χρειαζόμαστε για τον σχεδιασμό των box plots της Εικόνα 32.

	Position	Probability	First 100	Occurrence %
Minimum	1	1.72E-04	1	18.81188119
1st Quartile	6	1.68E-03	2	19.12013536
Median	21	1.74E-03	3	19.21768707
3rd Quartile	54	1.75E-03	4	19.68911917
Maximum	77	1.78E-03	5	21.79930796

Πίνακας 26 – Δεδομένα για τα Box Plots του Iodine 2

5.2.8.3 Box Plots



Εικόνα 32 – Iodine 2 Box Plots

Η διάμεσος (median) της κατάταξης (Position) είναι η τιμή 21 και το κεντρικό 50% των τιμών ορίζεται από το 6 έως το 54. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το 54 ενώ η μέγιστη τιμή είναι το 77.

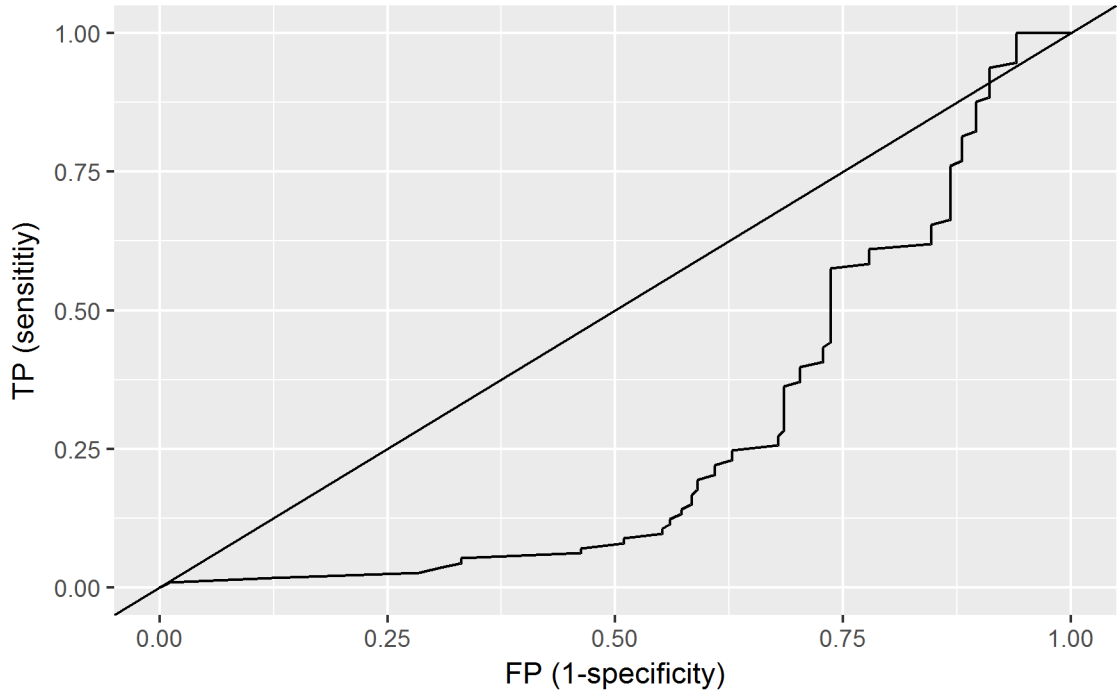
Στο γράφημα των πιθανοτήτων η διάμεσος είναι το 1.74E-03 και το κεντρικό 50% των τιμών ορίζεται από το 1.68E-03 έως το 1.75E-03. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το άνω 75% των τιμών είναι πάνω από το 1.68E-03 ενώ η ελάχιστη τιμή είναι το 1.72E-04.

Στο γράφημα First 100 η διάμεσος είναι το 3 και το κεντρικό 50% των τιμών ορίζεται από το 2 έως το 4. Η κατανομή είναι απόλυτα συμμετρική.

Στο γράφημα Occurrence % η διάμεσος είναι το ~19.21 και το κεντρικό 50% των τιμών ορίζεται από το ~19.12 έως το ~19.68. Η κατανομή είναι ασύμμετρη, καθώς παρατηρούμε πως το κάτω 75% των τιμών είναι συγκεντρωμένο μέχρι το ~19.68 ενώ η μέγιστη τιμή είναι το ~21.8.

5.2.8.4 ROC - AUROC

Στην Εικόνα 33 παρουσιάζουμε ενδεικτικά την μία καμπύλη ROC από τις πέντε που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Παρατηρούμε πως ο διαχωρισμός του μοντέλου σε αυτή τη περίπτωση δεν είναι καλός, καθώς η καμπύλη βρίσκεται κάτω από την ευθεία που αναπαριστά τη τυχαιότητα. Στον Πίνακας 27 φαίνονται οι πέντε τιμές AUROC που πήραμε.



Εικόνα 33 – ROC Iodine 2

	Δοκιμή #1	Δοκιμή #2	Δοκιμή #3	Δοκιμή #4	Δοκιμή #5
AUROC	0.18752	0.18433	0.277	0.15018	0.18541
Μέση Τιμή: 0.196889			Απόκλιση: 0.002244		

Πίνακας 27 – AUROC Iodine 2

Η μέση τιμή της τιμής AUROC που παίρνουμε για τις 5 επιθέσεις με το εργαλείο Iodine στο σενάριο όπου δεν συμμετέχουν οι Client και Server είναι 0.196889 και η απόκλιση 0.002244.

5.2.9 Αλλαγή Κίνησης και Παραμέτρων

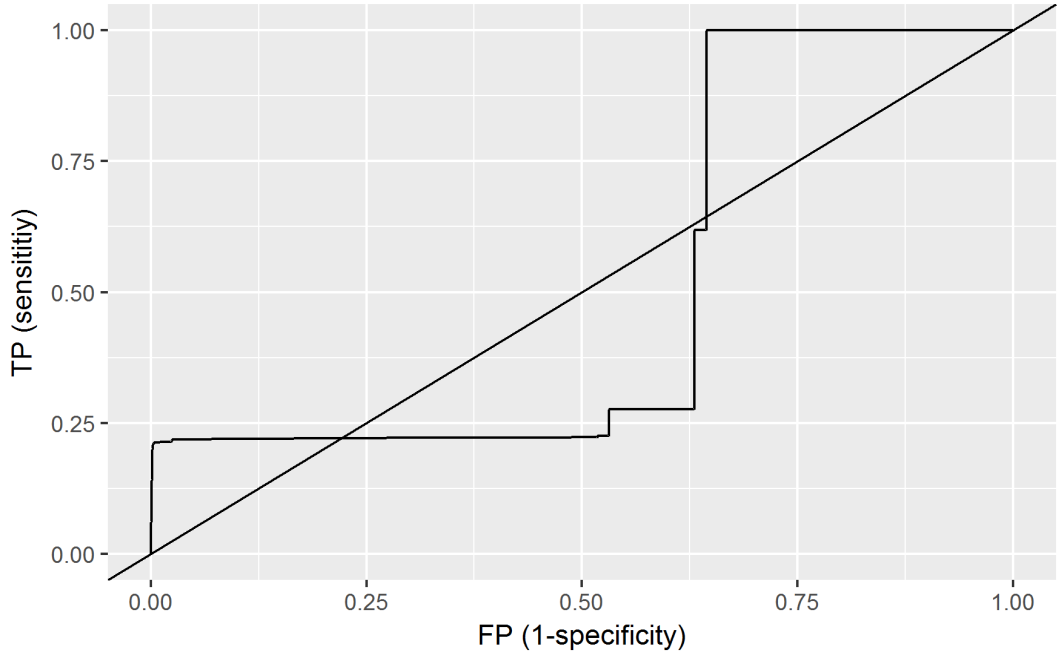
Επιλέξαμε τις επιθέσεις με τα εργαλεία nmap, Nessus και Slowloris να συμμετάσχουν στη δοκιμή αύξησης της παραγόμενης κίνησης και αλλαγής παραμέτρων στον αλγόριθμο μηχανικής μάθησης. Συγκεκριμένα, αντί για ~1.9GB κίνησης όπως αναφέραμε στην ενότητα 4.1.1, αναπαράγαμε 8 pcap των 477MB από μία φορά το καθένα διπλασιάζοντας ουσιαστικά την κίνηση σε ~3.8GB. Παράλληλα, βασιζόμενοι σε αναφορές από εμπειρικά στοιχεία, οι οποίες υποστηρίζουν πως ο αλγόριθμος μηχανικής μάθησης αρχίζει να συγκλίνει στις 100 επαναλήψεις (ldamaxiterations), αυξήσαμε τις επαναλήψεις από τις 20, που είναι η προεπιλεγμένη τιμή, στις 100. Τέλος, δοκιμάσαμε διάφορες τιμές για την παράμετρο που ορίζει το πλήθος των θεμάτων (topics) (TOPIC_COUNT), καθώς ανάλογα με τη κίνηση δικτύου και την επίθεση, το πλήθος των θεμάτων (topics) που έχει επιλεγεί μπορεί να επηρεάσει την ακρίβεια των αποτελεσμάτων του αλγορίθμου μηχανικής μάθησης.

Κάθε εργαλείο χρησιμοποιήθηκε με τον ίδιο τρόπο όπως περιγράφηκαν στις ενότητες 5.2.1, 5.2.2 και 5.2.5 για τα εργαλεία nmap, Nessus και Slowloris αντίστοιχα. Τα δεδομένα που πήραμε στην Hive μετά

το πέρας της αναπαραγωγής κίνησης, της εκτέλεσης της επίθεσης και της διαδικασίας της λήψης δεδομένων, δόθηκαν ως είσοδο στον αλγόριθμο μηχανικής μάθησης στον οποίο είχαμε αλλάξει τις επαναλήψεις από 20 σε 100. Τα δεδομένα στη Hive από κάθε επίθεση αποθηκεύτηκαν, ώστε να εκτελέσουμε ξανά τον αλγόριθμο μηχανικής μάθησης, αλλάζοντας και τη παράμετρο που ορίζει το πλήθος των θεμάτων (topics), σε διάφορες τιμές. Συγκεκριμένα τον εκτελέσαμε για τις τιμές 5, 10, 20 (προεπιλεγμένη), 30, 40. Συνοψίζοντας, για κάθε ένα εργαλείο, σε όλες τις εκτελέσεις του αλγόριθμου μηχανικής μάθησης έχουμε τα ίδια δεδομένα στη Hive και η παράμετρος ldamaxiterations είναι 100. Αυτό που αλλάζει μεταξύ των εκτελέσεων του αλγορίθμου είναι η παράμετρος που ορίζει το πλήθος των θεμάτων (topics).

5.2.9.1 nmap

Η εκτέλεση της επίθεσης είναι ακριβώς η ίδια με αυτή που περιγράφηκε στην ενότητα 5.2.1. Για κάθε αποτέλεσμα των εκτελέσεων του αλγορίθμου μηχανικής μάθησης έχουμε σχεδιάσει την καμπύλη ROC και έχουμε υπολογίσει την αντίστοιχη AUROC.



Εικόνα 34 – ROC nmap με Αλλαγή Κίνησης/Παραμέτρων

Στην Εικόνα 34 παρουσιάζουμε μία από τις πέντε καμπύλες ROC που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Παρατηρούμε πως ενώ η αρχή της καμπύλης είναι καλή που σημαίνει πως είναι υψηλά καταταγμένη η επίθεση, η συνέχεια δεν είναι ανάλογη και έτσι συνολικά ο διαχωρισμός δεν είναι καλός.

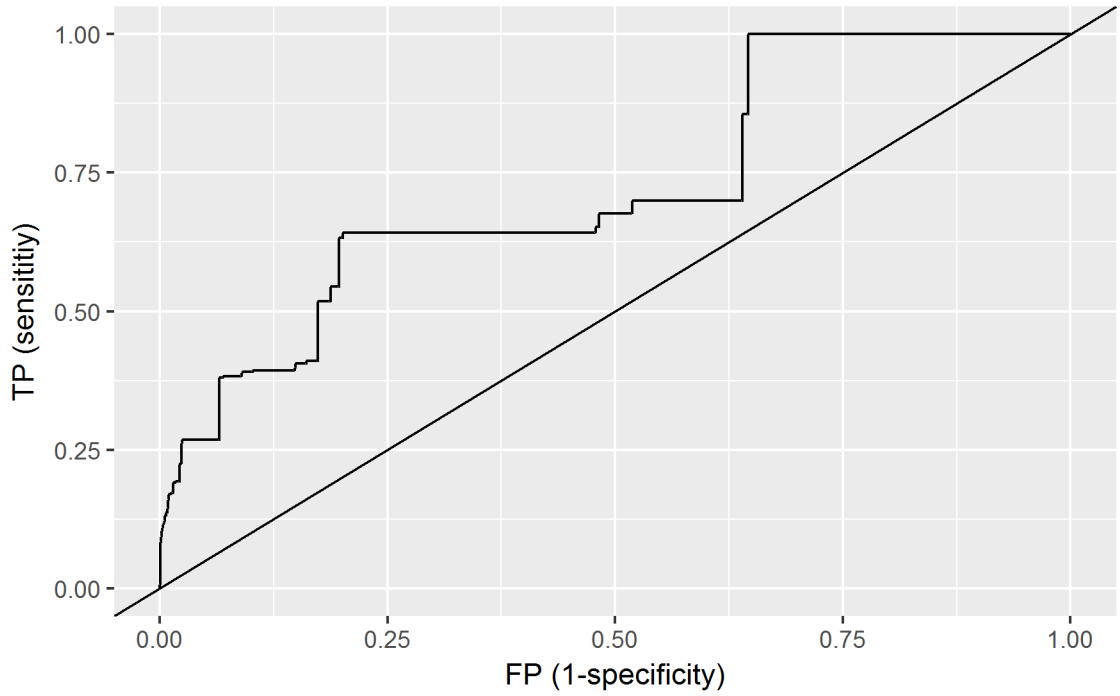
Στον Πίνακας 28 παρουσιάζουμε τις τιμές AUROC που πήραμε για κάθε πλήθος topics.

#Topics	AUROC
5	0.4949097
10	0.5122475
20	0.5108934
30	0.5096394
40	0.4975442

Πίνακας 28 – AUROC nmap με Αλλαγή Topics

5.2.9.2 Nessus

Η εκτέλεση της επίθεσης είναι ακριβώς η ίδια με αυτή που περιγράφηκε στην ενότητα 5.2.2. Για κάθε αποτέλεσμα των εκτελέσεων του αλγορίθμου μηχανικής μάθησης έχουμε σχεδιάσει την καμπύλη ROC και έχουμε υπολογίσει την αντίστοιχη AUROC.



Εικόνα 35 – ROC Nessus με Αλλαγή Κίνησης/Παραμέτρων

Στην Εικόνα 35 παρουσιάζουμε μία από τις πέντε καμπύλες ROC που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Η καμπύλη είναι πάνω από την ευθεία που αναπαριστά τη τυχαιότητα αλλά όχι πολύ, ο διαχωρισμός δεν είναι ικανοποιητικός.

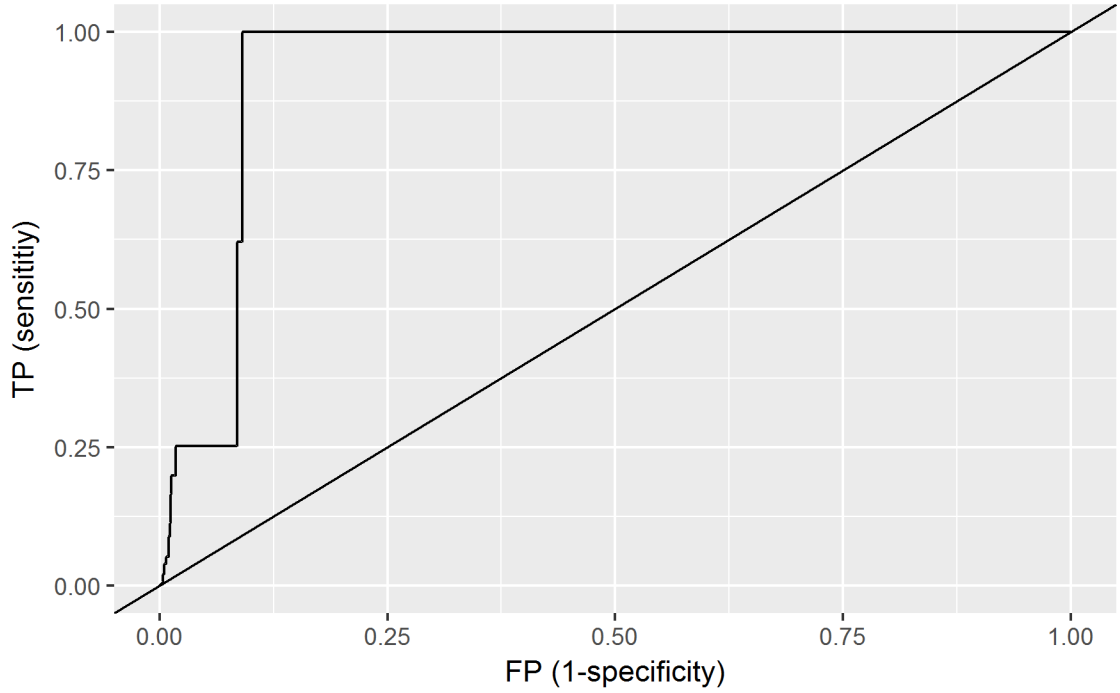
Στον Πίνακας 29 παρουσιάζουμε τις τιμές AUROC που πήραμε για κάθε πλήθος topics.

#Topics	AUROC
5	0.7133836
10	0.705946
20	0.7188948
30	0.7166957
40	0.721895

Πίνακας 29 – AUROC Nessus με Αλλαγή Topics

5.2.9.3 Slowloris

Η εκτέλεση της επίθεσης είναι η ίδια με αυτή που περιγράφηκε στην ενότητα 5.2.5, με μόνη διαφορά ότι η διάρκεια της επίθεσης ήταν περίπου όσο χρειάστηκε για την αναπαραγωγή των δύο από τα οκτώ αρχεία pcap σε αντίθεση με την ενότητα 5.2.5 που εκτελέστηκε καθ' όλη τη διάρκεια της αναπαραγωγής κίνησης (τέσσερα αρχεία pcap). Για κάθε αποτέλεσμα των εκτελέσεων του αλγορίθμου μηχανικής μάθησης έχουμε σχεδιάσει την καμπύλη ROC και έχουμε υπολογίσει την αντίστοιχη AUROC.



Εικόνα 36 – ROC Slowloris με Αλλαγή Κίνησης/Παραμέτρων

Στην Εικόνα 36 παρουσιάζουμε μία από τις πέντε καμπύλες ROC που σχεδιάσαμε, καθώς οι υπόλοιπες είναι παρόμοιες. Η καμπύλη είναι πολύ πάνω από την ευθεία που αναπαριστά τη τυχαιότητα και πλησιάζει την ιδανική καμπύλη, ο διαχωρισμός είναι πολύ καλός σε αυτή τη περίπτωση.

Στον Πίνακας 30 παρουσιάζουμε τις τιμές AUROC που πήραμε για κάθε πλήθος topics.

#Topics	AUROC
5	0.931611
10	0.929403
20	0.929273
30	0.931365
40	0.929171

Πίνακας 30 – AUROC Slowloris με Αλλαγή Topics

5.2.10 Ημι-Επιβλεπόμενη Μάθηση

Με στόχο την αξιολόγηση της λειτουργίας της ημι-επιβλεπόμενης μάθησης(βλ. 2.4.1), επιλέξαμε να την εφαρμόσουμε στα αποτελέσματα που πήραμε με το εργαλείο iodine. Συγκεκριμένα, βαθμολογήσαμε τα αποτελέσματα της τελευταίας επίθεσης από αυτές που περιλαμβάνουν τον Client και τον Server, και εκτελέσαμε ξανά την ίδια επίθεση πέντε φορές, κάθε φορά βαθμολογώντας τα αποτελέσματα. Επιλέχθηκε το εργαλείο iodine λόγω του ότι έχει χειρότερα αποτελέσματα σε σχέση με τα υπόλοιπα εργαλεία όσον αφορά τη κατάταξη της επίθεσης. Τα αποτελέσματα παρουσιάζονται στους πίνακες Πίνακας 31 και Πίνακας 32.

Position	Probability	First 100	Occurrence %
5465	2.00E-04	0	0.6206229
4597	2.44E-04	0	2.990908318
3389	7.29E-05	0	1.301630736
26	2.13E-04	1	1.648403109
1484	5.72E-04	0	3.680468423

Πίνακας 31 – Αποτελέσματα Iodine 1 με Ημι-Επιβλεπόμενη Μάθηση

	Δοκιμή #1	Δοκιμή #2	Δοκιμή #3	Δοκιμή #4	Δοκιμή #5
AUROC	0.16965	0.07816	0.2038	0.27406	0.11588
Μέση Τιμή: 0.168309			Απόκλιση: 0.00583		

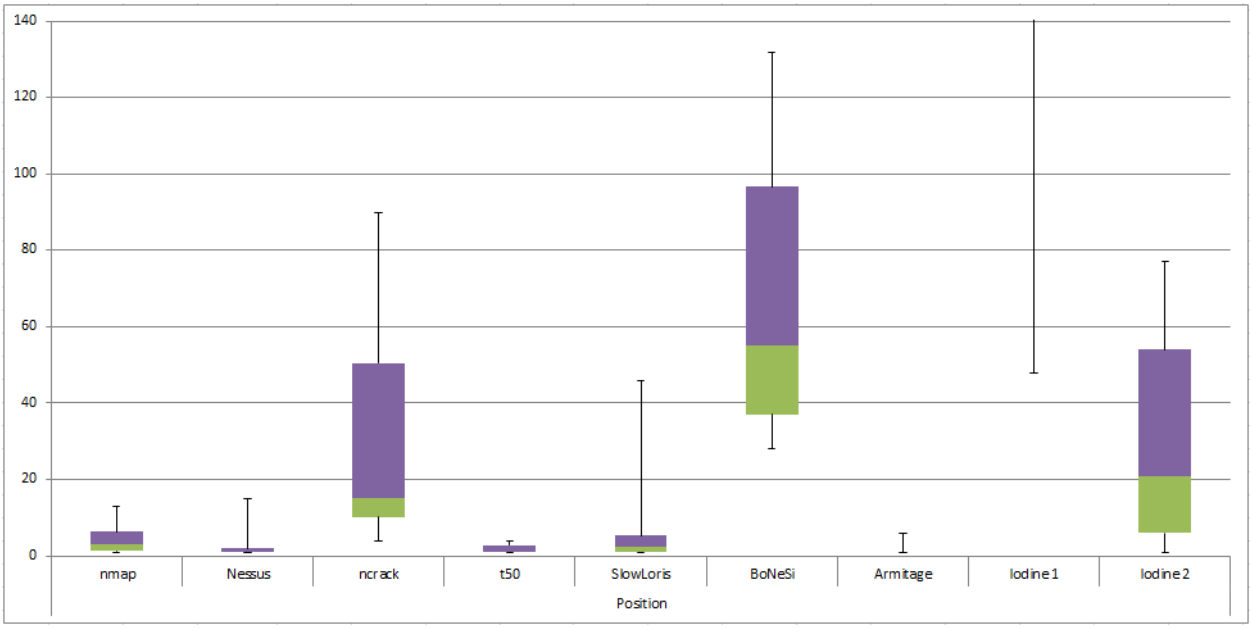
Πίνακας 32 – AUROC Iodine 1 με Ημι-Επιβλεπόμενη Μάθηση

6 Σύγκριση Αποτελεσμάτων και Συμπεράσματα

Στην ενότητα αυτή, αρχικά πραγματοποιούμε σύγκριση των δεδομένων που παρουσιάσαμε στην ενότητα 5.2. Συγκεκριμένα συγκρίνουμε τα box plots για τις μετρικές Position, Probability και First 100, και τις τιμές AUROC, μεταξύ των εργαλείων δοκιμών διείσδυσης. Επίσης, συγκρίνουμε τις τιμές AUROC που πήραμε μετά την αλλαγή της κίνησης και των παραμέτρων (βλ. 5.2.9) με τις τιμές πριν την αλλαγή, και την επίδραση της αλλαγής του πλήθους θεμάτων (topics) στον αλγόριθμο μηχανικής μάθησης. Τέλος, εξάγουμε συμπεράσματα βασιζόμενοι στις παρατηρήσεις μας από την διαδικασία αξιολόγησης και σύγκρισης των δεδομένων. Σημειώνουμε πως η επίθεση με το Iodine που περιλαμβάνει τα Client και Server θα αναφέρεται ως “Iodine 1” και η επίθεση χωρίς αυτά θα αναφέρεται ως “Iodine 2”.

6.1 Σύγκριση Box Plots

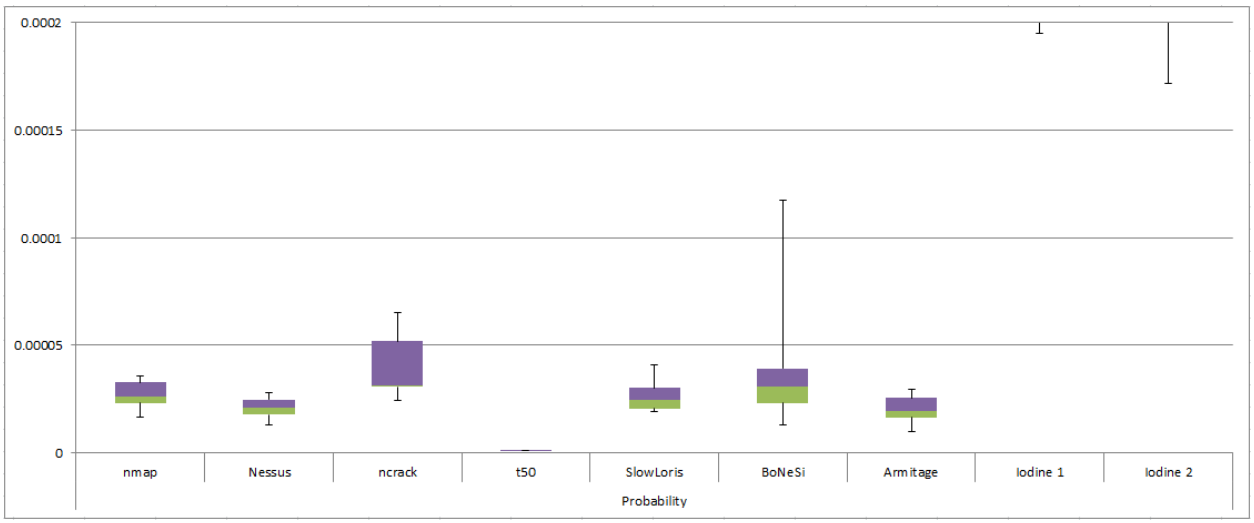
Στην Εικόνα 37 παρουσιάζουμε τα box plots για τη μετρική Position για κάθε εργαλείο. Τα καλύτερα αποτελέσματα παρατηρούνται στο εργαλείο t50 και το Armitage. Ακολουθούν τα Nessus και nmap. Το Slowloris έχει πολύ καλά αποτελέσματα στο 75% της κατανομής, αλλά έχει ένα 25% που απέχει από τις υψηλές θέσεις. Τα ncrack, Iodine 2 και BoNeSi έχουν μεγάλο εύρος τιμών και άρα δεν είναι αξιόπιστη η κατάταξη για αυτά τα εργαλεία. Το Iodine 1 έχει πολύ χαμηλή κατάταξη και έτσι εμφανίζουμε μόνο ένα μέρος του κάτω 25% των τιμών ώστε να είναι ευδιάκριτο το υπόλοιπο διάγραμμα.



Εικόνα 37 – Σύγκριση των Position Box Plots

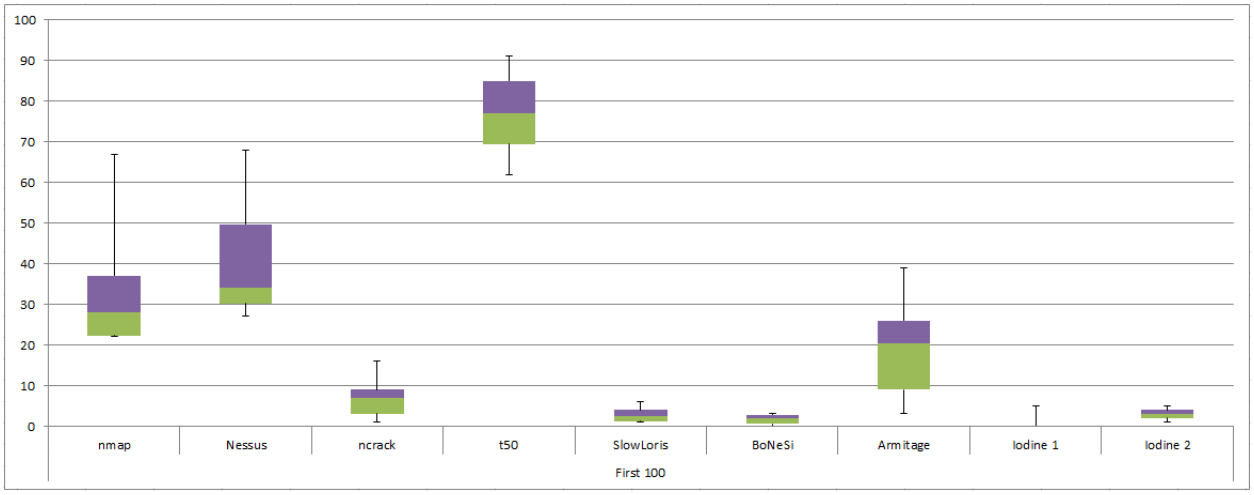
Στην Εικόνα 38 βλέπουμε τα box plots για τη μετρική Probability. Τα καλύτερα αποτελέσματα παρατηρούνται στο εργαλείο t50 το οποίο απέχει πολύ από τα υπόλοιπα. Ακολουθούν τα εργαλεία nmap, Nessus, Slowloris και Armitage τα οποία κυμαίνονται σε παρόμοια επίπεδα. Το BoNeSi έχει μεγάλη ασυμμετρία με το άνω 25% των τιμών να περιέχει υψηλές τιμές, ενώ το ncrack είναι πιο συμμετρικό αλλά και πάλι έχει πιο υψηλές τιμές από τα υπόλοιπα εκτός από το BoNeSi. Τα Iodine έχουν έως και 1 ολόκληρη τάξη μεγαλύτερες πιθανότητες από τα υπόλοιπα εργαλεία, γι’ αυτό και στο

διάγραμμα φαίνεται μόνο μέρος του κάτω 25% των τιμών τους. Υπενθυμίζουμε πως η πιθανότητα που δίνει το Apache Spot, είναι η πιθανότητα να είναι φυσιολογική κίνηση, δηλαδή όσο πιο μικρή τόσο πιο πιθανό να είναι επίθεση (βλ. 5.1).



Εικόνα 38 – Σύγκριση των Probability Box Plots

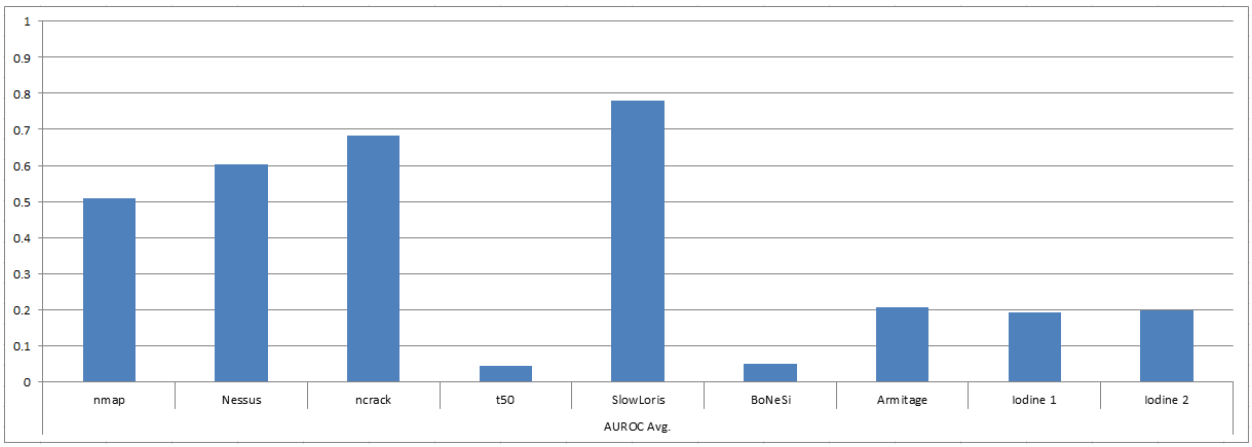
Στην Εικόνα 39 παρουσιάζονται τα box plots για τη μετρική First 100. Τα καλύτερα αποτελέσματα παρατηρούνται στο t50. Ακολουθούν τα Nessus και nmap και μετα το Armitage. Τα εργαλεία ncrack, Slowloris, BoNeSi και Iodine είναι πολύ κάτω σε σχέση με τα προηγούμενα.



Εικόνα 39 – Σύγκριση των First 100 Box plots

6.2 Σύγκριση Μέσης AUROC

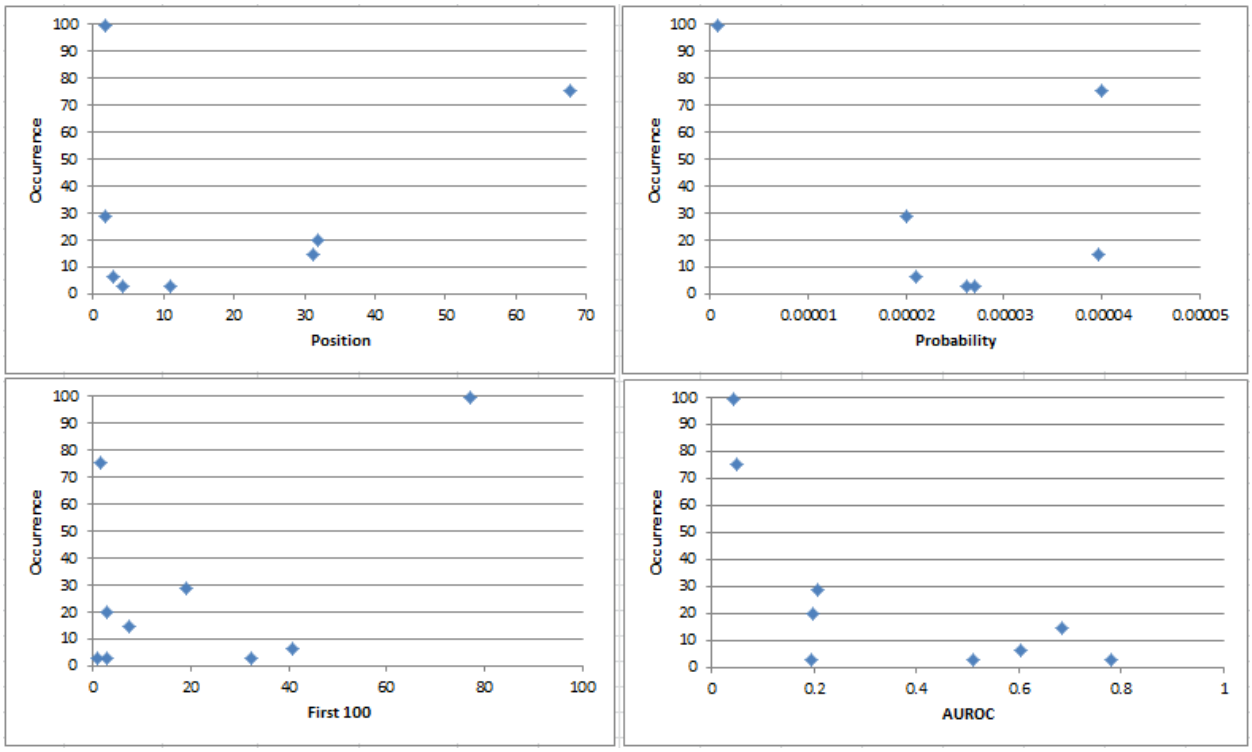
Στην Εικόνα 40 παρουσιάζουμε ένα ραβδόγραμμα στο οποίο συγκρίνονται οι μέσες τιμές AUROC που πήραμε από τα εργαλεία, όπως παρουσιάστηκαν στην ενότητα 5.2. Παρατηρούμε πως μόνο για το Slowloris προκύπτει μία αποδεκτή τιμή με μέσο όρο AUROC ~0.78. Το ncrack απέχει αρκετά από τη τυχαιότητα με μέση AUROC ~0.68, αλλά δεν αποτελεί αποδεκτή τιμή. Τα υπόλοιπα εργαλεία έχουν AUROC κοντά στη τυχαιότητα ή χειρότερα.



Εικόνα 40 - Σύγκριση Μέσων Τιμών AUROC

6.3 Σχέσεις μεταξύ Occurrence και υπόλοιπων δεδομένων

Στην Εικόνα 41 παρουσιάζονται διαγράμματα διασποράς (scatter plots) όπου σε κάθε ένα φαίνεται η σχέση μεταξύ της μέσης τιμής Occurrence με τη μέση τιμή των Position,Probability,First 100,AUROC αντίστοιχα που πήραμε για κάθε εργαλείο.



Εικόνα 41 – Σχέσεις Μεταξύ Occurrence και Υπόλοιπων Δεδομένων

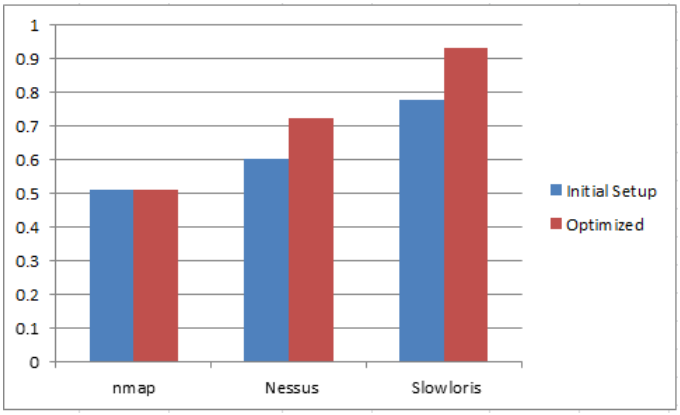
Παρατηρούμε πως δεν προκύπτει κάποια καθολική σχέση ανάμεσα στο Occurrence και κάποια από τις άλλες μετρικές. Διαισθητικά, θα λέγαμε πως, όσο πιο μικρό το Occurrence τόσο πιο μη-φυσιολογική θα πρέπει να φαίνεται μία επίθεση μέσα στο σύνολο της κίνησης δικτύου που επεξεργάζεται ο αλγόριθμος μηχανικής μάθησης. Βλέπουμε όμως, ότι αυτό δεν ισχύει καθολικά και μπορεί να επηρεαστεί από το

είδος της επίθεσης, με πιο χαρακτηριστικό παράδειγμα αυτό του εργαλείου t50, το οποίο έχει Occurrence κοντά στο 100% και παρ' όλα αυτά το Apache Spot το εντοπίζει σαν επίθεση βάσει όλων των μετρικών εκτός από την AUROC.

6.4 Αποτελέσματα Αλλαγής Κίνησης και Παραμέτρων

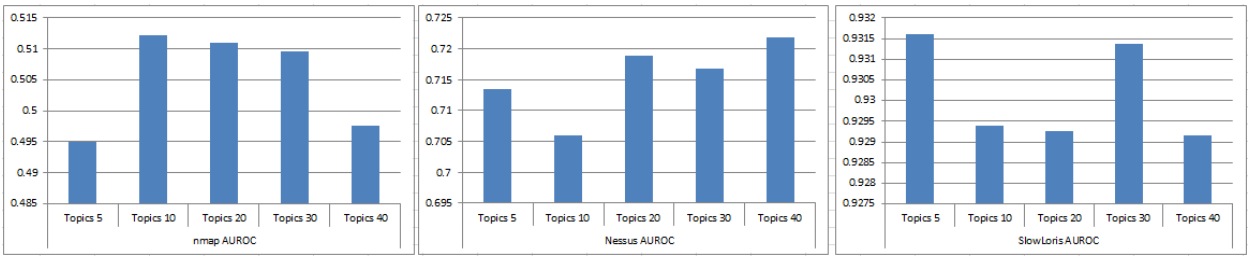
Στην Εικόνα 42 παρουσιάζουμε μια σύγκριση ανάμεσα στη μέση τιμή AUROC με την αρχική διάταξη και στη τιμή AUROC(δεν είναι μέση τιμή, καθώς πάρθηκε μόνο ένα δείγμα) που πάρθηκε μετά την αλλαγή της κίνησης και των παραμέτρων, όπως περιγράφηκε στην ενότητα 5.2.9.

Παρατηρούμε πως υπήρξε μεγάλη αύξηση της τιμής AUROC των εργαλείων Nessus και Slowloris. Στο nmap δεν υπήρξε καμία βελτίωση καθώς η καμπύλη ROC ακολούθησε το ίδιο μοτίβο που είχε ακολουθήσει και με την αρχική διάταξη, όπου ένα μέρος της επίθεσης έχει πολύ υψηλή κατάταξη, αλλά η υπόλοιπη κίνηση που αφορά την επίθεση είναι καταταγμένη πολύ χαμηλά. (βλ. Εικόνα 17 και Εικόνα 34).



Εικόνα 42 - Σύγκριση AUROC με Αλλαγή Κίνησης και Παραμέτρων

Στην Εικόνα 43 συγκρίνουμε τις τιμές AUROC που παίρνουμε αλλάζοντας τη παράμετρο που αφορά το πλήθος των θεμάτων (topics) στον αλγόριθμο μηχανικής μάθησης. Βλέπουμε πως οι διαφορές είναι αμελητέες και άρα συμπεραίνουμε πως η παράμετρος αυτή δεν επηρεάζει σημαντικά την αποδοτικότητα του για το μέγεθος κίνησης δικτύου που χρησιμοποιήσαμε εμείς.



Εικόνα 43 - Σύγκριση AUROC με Αλλαγή Θεμάτων (Topics)

6.5 Ημι-Επιβλεπόμενη Μάθηση

Στον Πίνακας 33 παρουσιάζουμε μια σύγκριση ανάμεσα στα αποτελέσματα του Iodine 1 χωρίς ημι-επιβλεπόμενη μάθηση και τα αποτελέσματα του με ημι-επιβλεπόμενη μάθηση. Οι πρώτες πέντε γραμμές που βρίσκονται πάνω από τη διπλή διαχωριστική γραμμή είναι τα αποτελέσματα που είχαμε πάρει χωρίς την ημι-επιβλεπόμενη μάθηση από τον Πίνακας 22. Από την 6η γραμμή και μετά είναι τα αποτελέσματα που πήραμε βαθμολογώντας αρχικά τα αποτελέσματα της 5ης επίθεσης και στη συνέχεια κάθε επίθεση που ακολούθησε.

Position	Probability	First 100	Occurrence %
4396	0.00121388	0	2.654931102
4832	3.72E-04	0	0.96459498
1323	6.80E-04	0	3.266025207
48	1.95E-04	5	4.329268293
6599	3.50E-04	0	0.887902331
5465	2.00E-04	0	0.6206229
4597	2.44E-04	0	2.990908318
3389	7.29E-05	0	1.301630736
26	2.13E-04	1	1.648403109
1484	5.72E-04	0	3.680468423

Πίνακας 33 – Αποτελέσματα του Iodine 1 με Ημι-Επιβλεπόμενη Μάθηση

Αντίστοιχα, στον Πίνακας 34 βλέπουμε τη σύγκριση για τις αντίστοιχες τιμές AUROC και στους πίνακες Πίνακας 35 και Πίνακας 36 τη σύγκριση για τη μέση τιμή και την απόκλιση των αντίστοιχων τιμών του Πίνακας 34.

AUROC
0.106351
0.143839
0.141951
0.421774
0.278647
0.169646
0.078159
0.203801
0.274058
0.115883

Πίνακας 34 – AUROC του Iodine 1 με Ημι-Επιβλεπόμενη Μάθηση

Μέση Τιμή AUROC	
Χωρίς Ημι-Επιβλεπόμενη	Με Ημι-Επιβλεπόμενη
0.191481	0.168309

Πίνακας 35 – Μέση Τιμή AUROC του Iodine 1 Με και Χωρίς Ημι-Επιβλεπόμενη Μάθηση

Απόκλιση Μέσης Τιμής AUROC	
Χωρίς Ημι-Επιβλεπόμενη	Με Ημι-Επιβλεπόμενη
0.01432	0.00583

Πίνακας 36 – Απόκλιση Μέσης Τιμής AUROC του Iodine 1 Με και Χωρίς Ημι-Επιβλεπόμενη Μάθηση

Παρατηρούμε πως η λειτουργία της ημι-επιβλεπόμενης μάθησης δεν είχε καμία θετική επιρροή στα αποτελέσματα που παίρνουμε για την επίθεση με το Iodine. Υπενθυμίζουμε πως η λειτουργία της ημι-επιβλεπόμενης μάθησης βρίσκεται σε πειραματικό στάδιο.

6.6 Συμπεράσματα

Στην ενότητα αυτή θα αναφέρουμε τα συμπεράσματα που έχουν προκύψει από τα αποτελέσματα και τη σύγκριση τους.

Πριν προχωρήσουμε σε συμπεράσματα, είναι σημαντικό να γίνει η εξής παρατήρηση: οι μετρικές Position, Probability και First 100 χρησιμεύουν για να αξιολογήσουμε τα αποτελέσματα του Apache Spot όπως θα τα έβλεπε ένας υπεύθυνος δικτύου, ο οποίος προφανώς δεν είναι σε θέση να εξετάσει το σύνολο των αποτελεσμάτων. Έτσι είναι σημαντικό πόσο ψηλά κατατάσσεται μία επίθεση και ας έχει χαμηλή τιμή AUROC. Η τιμή AUROC αφορά την εξέταση των αποτελεσμάτων όπως θα γινόταν από ένα υπολογιστικό σύστημα καθώς λόγω της υπολογιστικής του ισχύος θα είναι σε θέση να εξετάσει το σύνολο των αποτελεσμάτων αποδοτικά. Έτσι παύει να αποτελεί τον μόνο παράγοντα το πόσο υψηλά κατατάχθηκε η επίθεση και αξιολογούμε με μαθηματική ακρίβεια την αποδοτικότητα του μοντέλου.

Στις τρεις μετρικές Position, Probability και First 100 έχουμε λάβει τα καλύτερα αποτελέσματα με το εργαλείο t50. Ακολουθούν τα Armitage, Nessus και nmap που έχουν και αυτά καλά αποτελέσματα. Συμπεραίνουμε πως ο αλγόριθμος μηχανικής μάθησης επηρεάζεται από το είδος της επίθεσης δεδομένου της χαμηλής κατάταξης που παρατηρούμε στα υπόλοιπα εργαλεία και της μεγάλης απόκλισης που υπάρχει στις τιμές όπως φαίνεται από τα αντίστοιχα box plots.

Για την μετρική AUROC, έχουμε λάβει μέτρια αποτελέσματα με την αρχική διάταξη, όπου το Slowloris είναι το μόνο για το οποίο έχουμε λάβει αποδεκτές τιμές (~0.78 avg.). Παρ' όλα αυτά παρατηρήσαμε μεγάλη αύξηση της AUROC στα εργαλεία Slowloris και Nessus, παραμετροποιώντας διαφορετικά τον αλγόριθμο μηχανικής μάθησης και αυξάνοντας τη κίνηση δικτύου. Συγκεκριμένα, για το εργαλείο

Slowloris λάβαμε μία AUROC ~0.93 μετά από αύξηση κατά ~0.15 σε σχέση με την αρχική διάταξη, το οποίο είναι ένα πολύ καλό αποτέλεσμα. Αυτό σημαίνει, πως για κάποιες επιθέσεις θα μπορούσαμε να πάρουμε καλά αποτελέσματα και για τις ανάγκες ενός υπεύθυνου δικτύου αλλά και ενός υπολογιστικού συστήματος.

Από την παραμετροποίηση του αλγορίθμου ως προς το πλήθος των θεμάτων (topics) συμπεραίνουμε πως δεν επηρεάζει σε σημαντικό βαθμό την αποδοτικότητα του για το μέγεθος της κίνησης που χρησιμοποιήσαμε εμείς.

Από την βαθμολόγηση των αποτελεσμάτων του Iodine για την αξιολόγηση της λειτουργίας της ημι-επιβλεπόμενης μάθησης, συμπεραίνουμε πως η παρούσα πειραματική υλοποίηση δεν λειτουργεί σωστά και θα πρέπει να βελτιωθεί.

Συνολικά μπορούμε να συμπεράνουμε τα εξής:

- Το Apache Spot βρίσκεται στα πρώτα του βήματα (incubating) και αυτό φαίνεται από τα αποτελέσματα του, αφού έχουμε λάβει κάποια πολύ κακά αλλά και πολύ καλά δείγματα.
- Εντοπίζει επιθέσεις που άλλα συστήματα ανίχνευσης διεισδύσεων (IDS) δεν εντοπίζουν, με χαρακτηριστικό παράδειγμα την επίθεση Slowloris. Όμως, δεν εντοπίζει επιθέσεις που εντοπίζονται πολύ εύκολα από τα καθιερωμένα συστήματα ανίχνευσης διεισδύσεων (IDS) όπως το DNS Tunneling με το Iodine.
- Παρατηρούμε πως από αρκετά είδη επιθέσεων έχουν προκύψει πολύ καλά αποτελέσματα στις μετρικές Position, Probability και First 100 και άρα καθιστούν τα αποτελέσματα αποδεκτά αν πρόκειται να αξιολογηθούν από έναν υπεύθυνο δικτύου. Παρ’ όλα αυτά, όλα εκτός από την επίθεση Slowloris δεν έχουν αποδεκτή τιμή AUROC και τα καθιστά ακατάλληλα για αξιολόγηση από ένα υπολογιστικό σύστημα. Αυτό το θέμα, φαίνεται να επιλύεται σε κάποιες περιπτώσεις όπως δείξαμε με αύξηση της φυσιολογικής κίνησης και παραμετροποίηση του αλγορίθμου μηχανικής μάθησης.
- Είναι πολύ σημαντική η ποσότητα της φυσιολογικής κίνησης δικτύου - ώστε να είναι σε θέση ο αλγόριθμος μηχανικής μάθησης να «καταλάβει» τι είναι φυσιολογικό για το δίκτυο υπό επίβλεψη - και η παραμετροποίηση του αλγορίθμου, καθώς αυτοί οι δύο παράγοντες μπορούν να έχουν καθοριστικό ρόλο στην αποτελεσματικότητα του, όπως φάνηκε στα παραδείγματα των εργαλείων Nessus και Slowloris.
- Έχοντας γνώση των επιθέσεων για τις οποίες λειτουργεί αποδοτικά, παραμετροποιώντας τον αλγόριθμο μηχανικής μάθησης βάσει του δικτύου υπό επίβλεψη και έχοντας ένα δίκτυο στο οποίο η επίθεση θα είχε ένα μηδαμινό ποσοστό εμφάνισης επί του συνόλου της κίνησης δικτύου, μπορεί να αποτελέσει ένα πολύ χρήσιμο εργαλείο για έναν οργανισμό.
  Παρ’ όλα αυτά, η βιβλιογραφία του Apache Spot είναι ελλιπής προς το παρόν και αυτό το καθιστά ένα δύσκολο στον χειρισμό εργαλείο που δεν λειτουργεί αποδοτικά με τις προεπιλεγμένες ρυθμίσεις.

7 Μελλοντικές Επεκτάσεις

Βασιζόμενοι στα συμπεράσματα της παρούσας εργασίας, βλέπουμε πως θα μπορούσε να επεκταθεί η αξιολόγηση του Apache Spot σε επόμενες έρευνες. Συγκεκριμένα, μεγάλο ενδιαφέρον προκύπτει από την παρατήρηση της βελτίωσης των αποτελεσμάτων αυξάνοντας την κίνηση και αλλάζοντας τις παραμέτρους. Βάσει αυτού, θα είχε νόημα να αναπαράγουμε όσο το δυνατόν περισσότερη κίνηση δικτύου, με στόχο να προσεγγίσουμε μία προσομοίωση ενός πραγματικού δικτύου κάποιου μεγάλου οργανισμού. Παράλληλα, θα μπορούσαν να συμπεριληφθούν περισσότερα υπολογιστικά συστήματα στο δίκτυο, τόσο από τη πλευρά του δικτύου στόχου όσο και από τη πλευρά του επιτιθέμενου. Έχοντας μία τέτοια διάταξη στη διάθεση μας, θα μπορούσαμε στη συνέχεια να εξετάσουμε διαφορετικές παραμετροποιήσεις του αλγορίθμου μηχανικής μάθησης, ώστε να αξιολογήσουμε τη λειτουργία του Apache Spot σε μία μεγαλύτερη κλίμακα.

8 Πηγές – Βιβλιογραφία

Apache. (2016, - -). *Project Components - Ingestion*. Ανάκτηση February 22, 2018, από spot.incubator.apache.org: http://spot.incubator.apache.org/project-components/ingestion/

Apache. (2016a, - -). *Project Components - Ingestion*. Ανάκτηση February 22, 2018, από spot.incubator.apache.org: http://spot.incubator.apache.org/project-components/ingestion/

Apache. (2016b, - -). *Project Components - Machine Learning*. Ανάκτηση February 22, 2018, από spot.incubator.apache.org: http://spot.incubator.apache.org/project-components/machine-learning/

Apache. (2016c, - -). *Project Components - Suspicious Connects Analysis*. Ανάκτηση February 22, 2018, από spot.incubator.apache.org: http://spot.incubator.apache.org/project-components/suspicious-connects-analysis/

Apache. (2016d, - -). *Visualization*. Ανάκτηση February 22, 2018, από spot.incubator.apache.org: http://spot.incubator.apache.org/project-components/visualization/

Apache. (2018a, - -). *Apache Spark*. Ανάκτηση February 22, 2018, από spark.apache.org: https://spark.apache.org/

Apache. (2018b). *Spark Streaming*. Ανάκτηση March 10, 2018, από Apache Spark: https://spark.apache.org/streaming/

Apache. (2018c, - -). *MLlib*. Ανάκτηση February 22, 2018, από spark.apache.org: https://spark.apache.org/mllib/

Chantzis, F., & Lyon, G. (2010, - -). *ncrack*. Ανάκτηση February 22, 2018, από manpages.ubuntu.com: http://manpages.ubuntu.com/manpages/bionic/en/man1/ncrack.1.html

Cloudera. (2018, - -). *Developers - Inside CDH*. Ανάκτηση February 22, 2018, από cloudera.com: https://www.cloudera.com/developers/inside-cdh.html

Clouflare. (-, - -). *learning - ddos - ddos attack tools - slowloris*. Ανάκτηση February 22, 2018, από cloudflare.com: https://www.cloudflare.com/learning/ddos/ddos-attack-tools/slowloris/

Ekman, E., & Andersson, B. (2014, - -). *iodine*. Ανάκτηση February 22, 2018, από code.kryo.se: http://code.kryo.se/iodine/

Goldstein, M. (2016, - -). *Markus-Go - bonesi*. Ανάκτηση February 22, 2018, από github.com: https://github.com/Markus-Go/bonesi

Grigorev, A. (2015, January 18). *ROC Analysis*. Ανάκτηση February 22, 2018, από mlwiki.org: http://mlwiki.org/index.php/ROC_Analysis

Hansen, R. (2009, August 22). *slowloris*. Ανάκτηση February 22, 2018, από web.archive.org: https://web.archive.org/web/20090822001255/http://ha.ckers.org/slowloris/

Incapsula. (2018, - -). *SLOWLORIS*. Ανάκτηση February 22, 2018, από incapsula.com: https://www.incapsula.com/ddos/attack-glossary/slowloris.html

Intel. (2016, - -). *Financial Services IT - Apache Spot*-A More Effective Approach to Cyber Security.* Ανάκτηση January 3, 2018, από intel.com: https://www.intel.com/content/www/us/en/financial-services-it/apache-spot-a-more-effective-approach-to-cyber-security-solution-brief.html

Lyon, G. (2008). *Nmap Network Scanning.* Sunnyvale, CA: Insecure.Com LLC.

Lyon, G. (2017, - -). *Introduction*. Ανάκτηση February 22, 2018, από nmap.org: https://nmap.org/

Lyon, G. (2017, - -). *Ncrack*. Ανάκτηση February 22, 2018, από nmap.org: https://nmap.org/ncrack/

Mudge, R. (2016, - -). *rsmudge - armitage*. Ανάκτηση February 22, 2018, από github.com: https://github.com/rsmudge/armitage

Pissarra, F. (2017, - -). *fredericopissarra - t50*. Ανάκτηση February 22, 2018, από github.com: https://github.com/fredericopissarra/t50

Rapid7. (2018). *Metasploit*. Ανάκτηση March 10, 2018, από Metasploit: https://www.metasploit.com/

SPACE Hellas, i2cat, ubiwhere, Politecnico Di Torino, Infili, Telefonica, Orion Innovations, incites, Agenzia per l'Italia Digitale, Hewlett Packard Enterprise, NCSR Demokritos. (2017, May 31). *Documents - Project Deliverables.* Ανάκτηση January 12, 2018, από shield-h2020.eu: https://www.shield-h2020.eu/documents/project-deliverables/SHIELD_D4.1_Specifications,_Design_and_Architecture_for_the_Usable_Information-Driven_Engine_v.1.0.pdf

Wendlandt, D. (2004, - -). *~dwendlan - Personal - Nessus*. Ανάκτηση February 22, 2018, από cs.cmu.edu: http://www.cs.cmu.edu/~dwendlan/personal/nessus.html

Wikipedia. (2017a, December 28). *Latent Dirchlet Allocation*. Ανάκτηση February 22, 2018, από wikipedia.org: https://en.wikipedia.org/wiki/Latent_Dirichlet_allocation

Wikipedia. (2017b, May 29). *Armitage (computing)*. Ανάκτηση February 22, 2018, από wikipedia.org: https://en.wikipedia.org/wiki/Armitage_(computing)

Wikipedia. (2018a, February 22). *Apache Hadoop*. Ανάκτηση February 22, 2018, από wikipedia.org: https://en.wikipedia.org/wiki/Apache_Hadoop

Wikipedia. (2018b, February 20). *Apache Spark*. Ανάκτηση February 22, 2018, από wikipedia.org: https://en.wikipedia.org/wiki/Apache_Spark

Wikipedia. (2018c, January 23). *Apache Hive*. Ανάκτηση February 22, 2018, από wikipedia.org: https://en.wikipedia.org/wiki/Apache_Hive

Wikipedia. (2018d, February 21). *Apache Kafka*. Ανάκτηση February 22, 2018, από wikipedia.org: https://en.wikipedia.org/wiki/Apache_Kafka

Wikipedia. (2018e, February 4). *Topic Model*. Ανάκτηση February 22, 2018, από wikipedia.org: https://en.wikipedia.org/wiki/Topic_model

Wikipedia. (2018f, February 7). *Nmap*. Ανάκτηση February 22, 2018, από wikipedia.org: https://en.wikipedia.org/wiki/Nmap

Wikipedia. (2018g, February 6). *Nessus (software)*. Ανάκτηση February 22, 2018, από wikipedia.org: https://en.wikipedia.org/wiki/Nessus_(software)

Wikipedia. (2018h, February 18). *Slowloris (computer security)*. Ανάκτηση February 22, 2018, από wikipedia.org: https://en.wikipedia.org/wiki/Slowloris_(computer_security)

Wikipedia. (2018i, January 8). *Box plot*. Ανάκτηση February 22, 2018, από wikipedia.org: https://en.wikipedia.org/wiki/Box_plot

Yaltirakli, G. (2017, October 3). *gkbrk - slowloris*. Ανάκτηση February 24, 2018, από github.com: https://github.com/gkbrk/slowloris